Instrukcja oprogramowania

ITKP-27-01-06-20-PL

EtherNet/IP™

Protokół komunikacji:
PUE HY10

radwag.com
Zeskanuj kod QR, aby obejrzeć dodatkowe materiały naukowe, które mogąCię zainteresować.
Znajdziesz tam więcej przydatnych informacji w przystępnej formie!

	EtherNet/IP™ to znak towarowy firmy ODVA, Inc.

CZERWIEC 2020

"Użyte rysunki, zdjecia, grafiki maja charakter poglądowy."

SPIS TREŚCI

1. STRUKTURA DANYCH 4
1.1. Adres wejściowy 4
1.1.1. Wykaz zmiennych wejściowych 4
1.1.2. Opis rejestrów wejściowych 5
1.2. Adres wyjściowy 7
1.2.1. Wykaz zmiennych wejściowych 7
1.2.2. Opis rejestrów wyjściowych 7
2. KONFIGURACJA MODUŁU EtherNet/IP™ W ŚRODOWISKU RS LOGIX 10
2.1. KONFIGURACJA RSLinxs 10
2.2. Projekt RSLogix 11

1. STRUKTURA DANYCH

1.1. Adres wejściowy

1.1.1. 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	integer
Status platformy 1	10	1	integer
Próg Lo platformy 1	12	2	float
Masa platformy 2	16	2	float
Tara platformy 2	20	2	float
Jednostka platformy 2	24	1	integer
Status platformy 2	26	1	integer
Próg Lo platformy 2	28	2	float
Masa platformy 3	32	2	float
Tara platformy 3	36	2	float
Jednostka platformy 3	40	1	integer
Status platformy 3	42	1	integer
Próg Lo platformy 3	44	2	float
Masa platformy 4	48	2	float
Tara platformy 4	52	2	float
Jednostka platformy 4	56	1	integer
Status platformy 4	58	1	integer
Próg Lo platformy 4	60	2	float
Status procesu (Stop, Start)	64	1	integer
Stan wejść	66	1	integer
Min	68	2	float
Max	72	2	float
Numer serii	84	2	long
Operator	88	1	integer
Towar	90	1	integer
Kontrahent	92	1	integer
Opakowanie	94	1	integer
Magazyn źródłowy	96	1	integer
Magazyn docelowy	98	1	integer
Receptura/proces dozowania	100	1	integer

1.1.2. 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.

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.

Magazyn źródłowy - zwraca wartość kodu magazynu źródłowego.

Magazyn docelowy - zwraca wartość kodu magazynu docelowego.

Receptura/proces dozowania – zwraca wartość kodu wybranej receptury.

Proces dozowania – zwraca wartość kodu wybranego procesu dozowania.

1.2. Adres wyjściowy

1.2.1. Wykaz zmiennych wejściowych

Zmienna	Offset	Długość [WORD]	Typ danych
Komenda	0	1	integer
Komenda z parametrem	2	1	integer
Platforma	4	1	integer
Tara	6	2	float
Próg LO	10	2	float
Stan wyjść	14	1	integer
Min	16	2	float
Max	20	2	float
Numer serii	32	2	long
Operator	36	1	integer
Towar	38	1	integer
Kontrahent	40	1	integer
Opakowanie	42	1	integer
Magazyn źródłowy	44	1	integer
Magazyn docelowy	46	1	integer
Receptura/proces dozowania	48	1	integer

1.2.2. 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 procesu
5	Zatrzymanie procesu

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 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.

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 stanu wyjść
3	Ustawienie wartości progu MIN
4	Ustawienie wartości progu MAX

	Komenda złożona wymaga ustawienia odpowiedniego parametru (offset od 4 do 50 – patrz tabela rejestrów wyjściowych)
	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.

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)

Magazyn źródłowy - parametr komendy złożonej: kod magazynu źródłowego (tylko numeryczny).

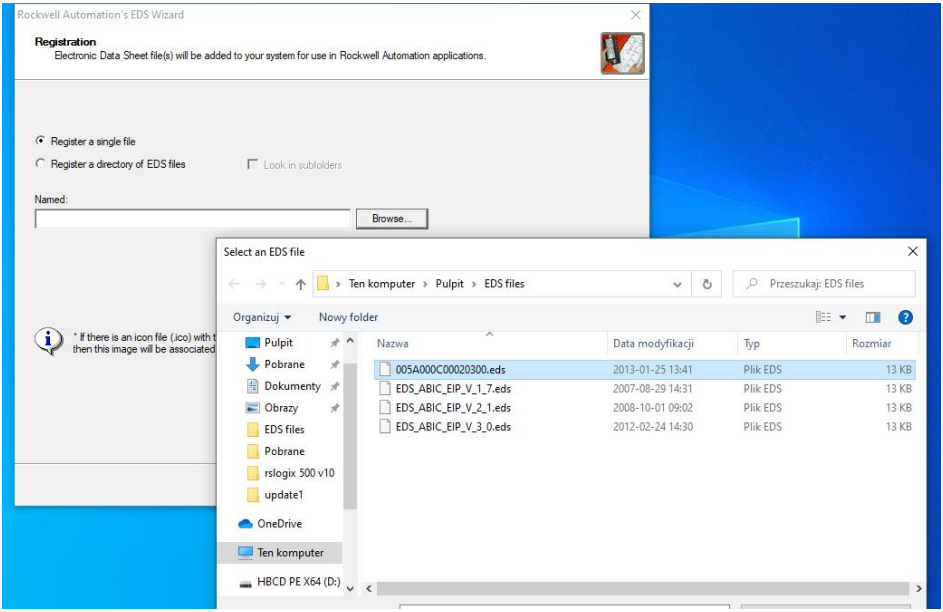
Magazyn docelowy - parametr komendy złożonej: kod magazynu docelowego (tylko numeryczny).

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

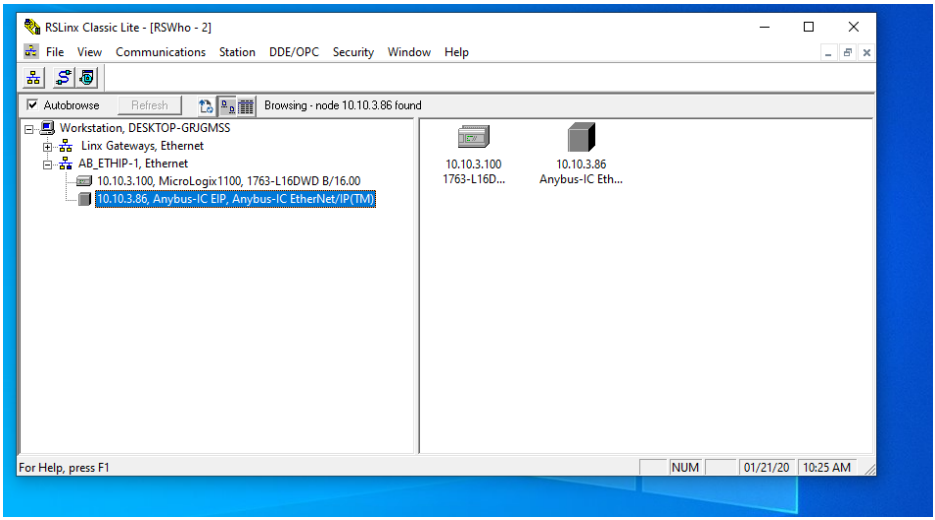
2. KONFIGURACJA MODUŁU EtherNet/IP™ W ŚRODOWISKU RS LOGIX

2.1. KONFIGURACJA RSLinxs

Pracę w środowisku należy rozpocząć od konfiguracji urządzeń w oprogramowaniu RSLinx. W tym celu należy dodać moduł EtherNet/IP wagi korzystając z pliku EDS oraz narzędzia EDS Hardware Installation Tool.

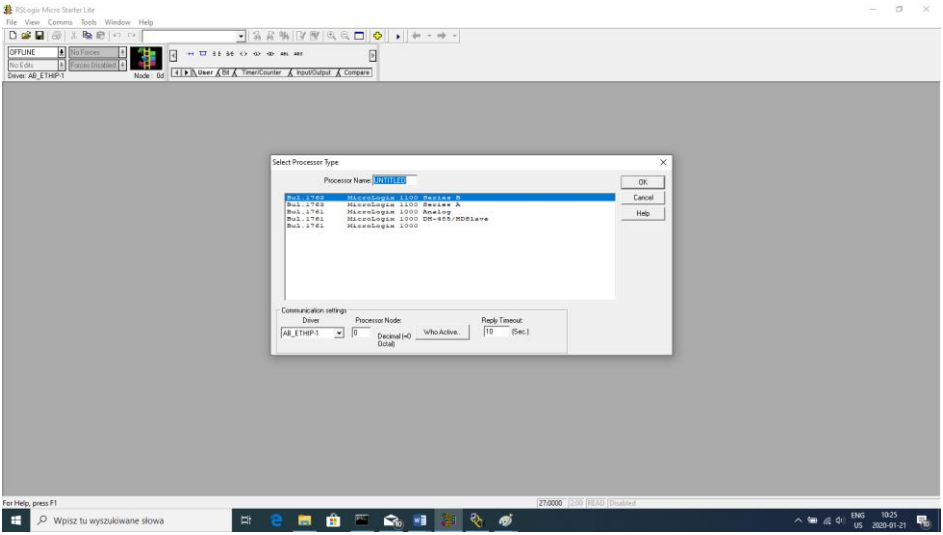


Po podłączeniu wagi oraz sterownika Master do sieci do sieci (należy zadbać żeby wszystkie urządzenia oraz komputer PC znajdowały się w tej samej podsieci) powinny być one widoczne jak na rysunku poniżej.



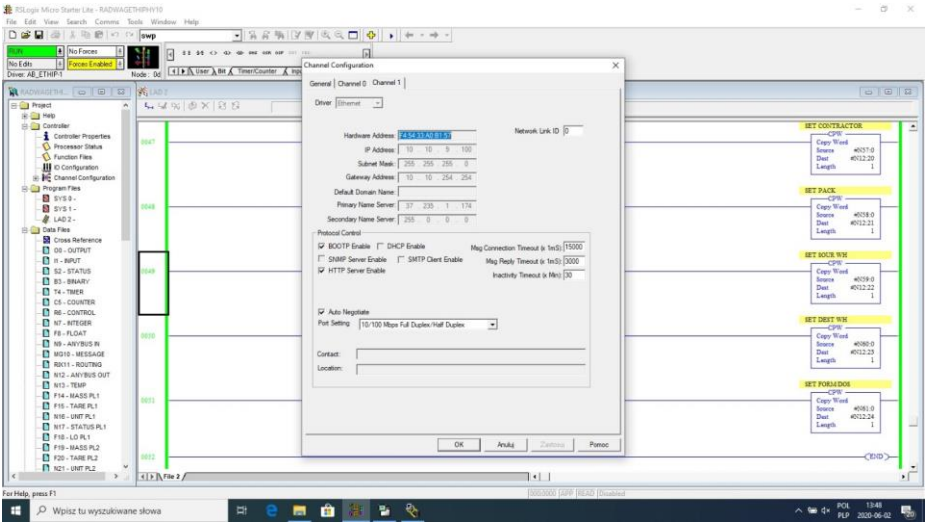
2.2. Projekt RSLogix

Pracę w środowisku rozpoczynamy od założenia nowego projektu. W oknie wyboru sterownika zaznaczamy PLC, który będzie komunikował się z wagą.



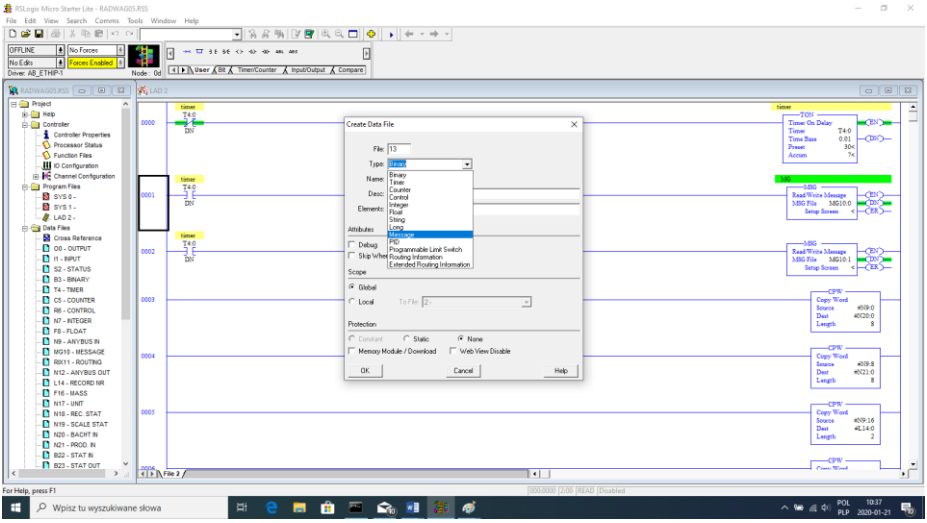
Po zatwierdzeniu wyboru przechodzimy do okna projektu. Następnie należy dokonać konfiguracji interfejsu komunikacyjnego sterownika. W tym celu w drzewie projektu wybieramy CHANNEL CONFIGURATION>CHANNEL 1.

W tym miejscu możemy zadeklarować właściwości tego kanału komunikacji takie jak adres IP czy maska podsieci.

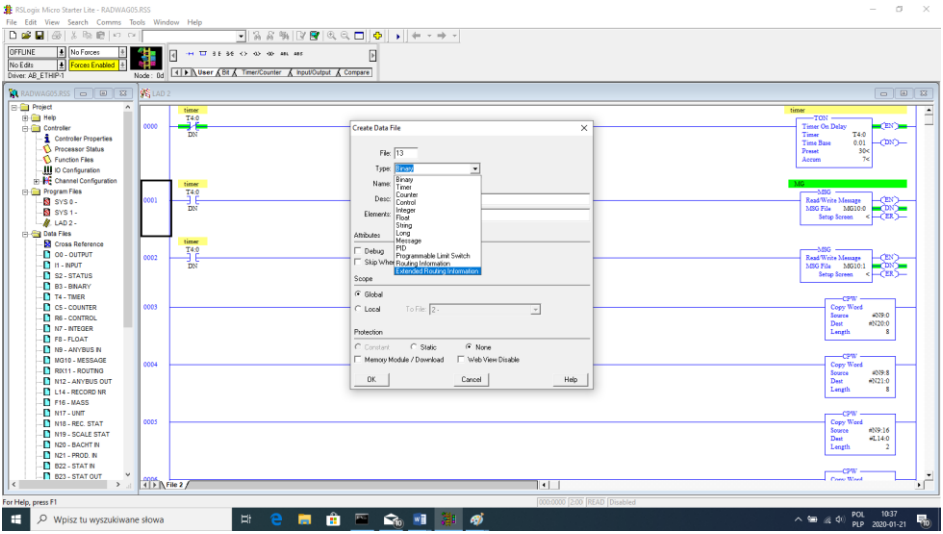


Po dokonaniu konfiguracji warto sprawdzić czy jesteśmy w stanie połączyć się z PLC (online) i załadować projekt (download).

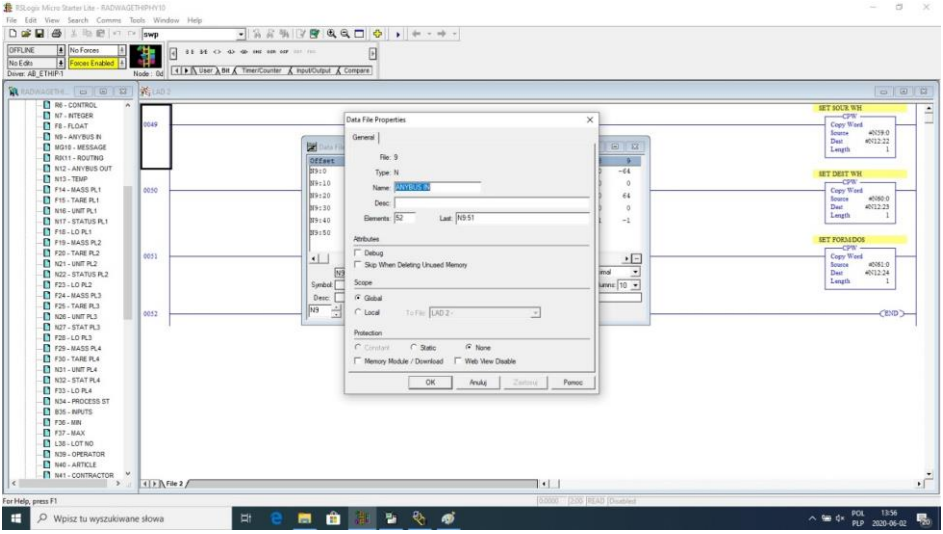
Następnym krokiem będzie dodanie nowego szczebla w drabinie projektu (rung) i umieszczenie w nim funkcji MSG służącej do odczytu danych z wagi. Zanim dodamy funkcję należy w drzewie projektu dodać nowe pliki danych (data files). Będą to dwuelementowe pliki typu MG (message).

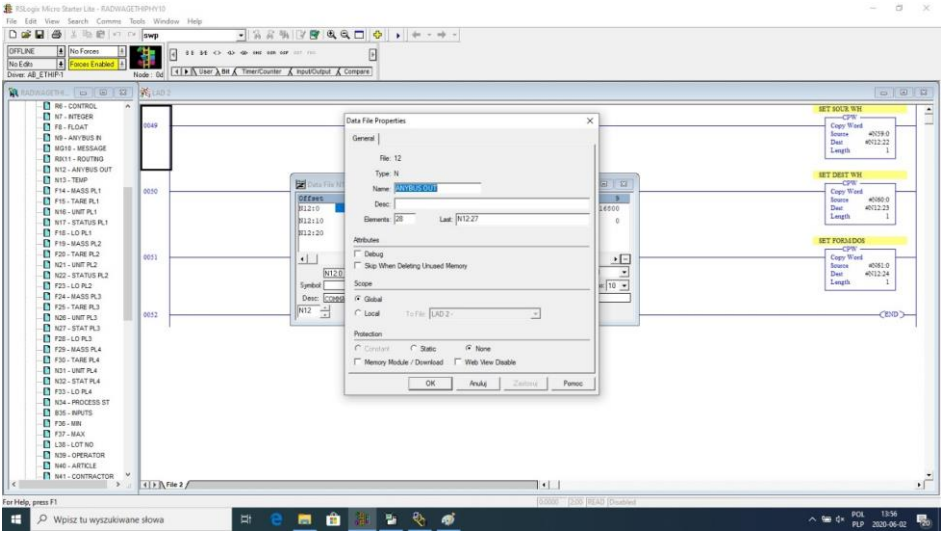


oraz RIX.

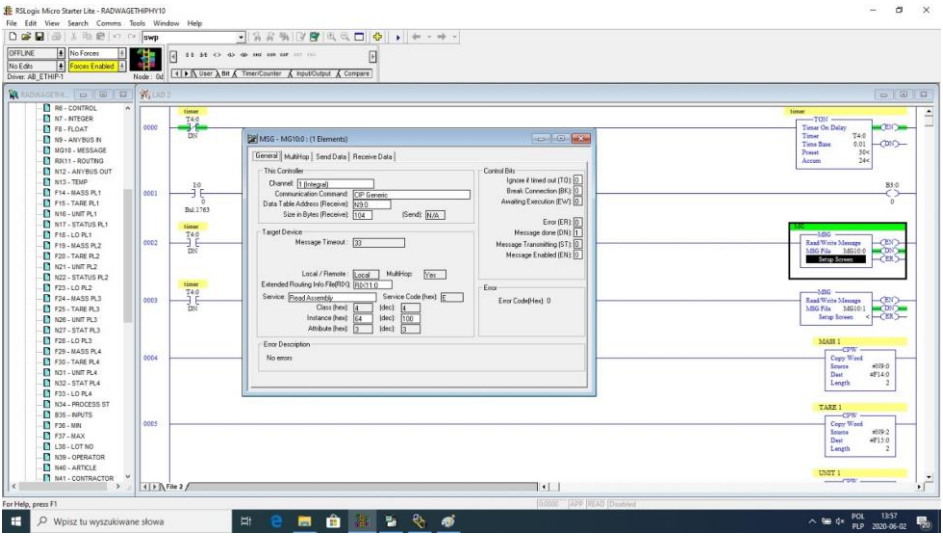


Należy również dodać 2 pliki typu INTEGER, w których będą przechowywane dane odczytane z wagi oraz te, które do wagi będą wysyłane. W przykładzie stworzono plik ANYBUS IN (N9) o rozmiarze 104 bajty oraz ANYBUS OUT (N12) o rozmiarze 56 bajtów.





Możemy już dodać funkcje MSG, jedną do odczytu danych i drugą do zapisu.

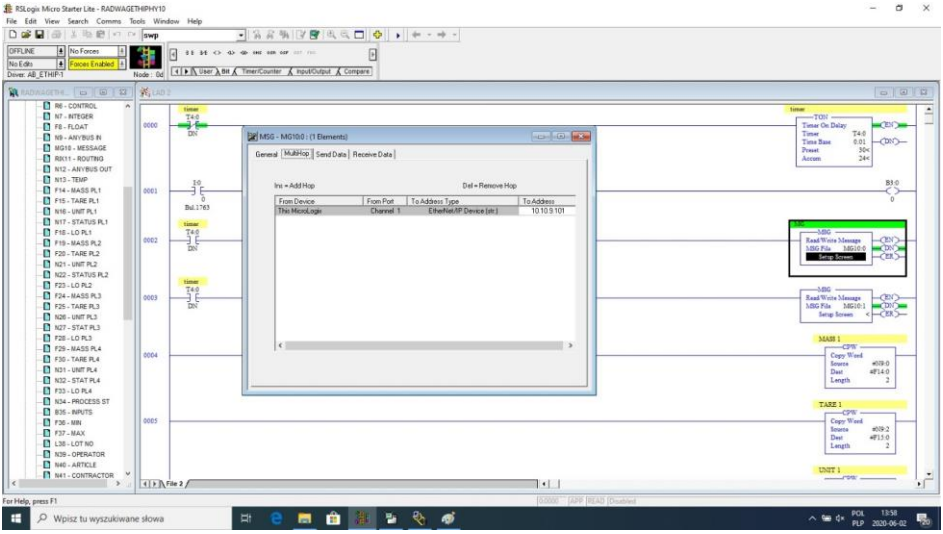


Konfiguracja sprowadza się do podania:

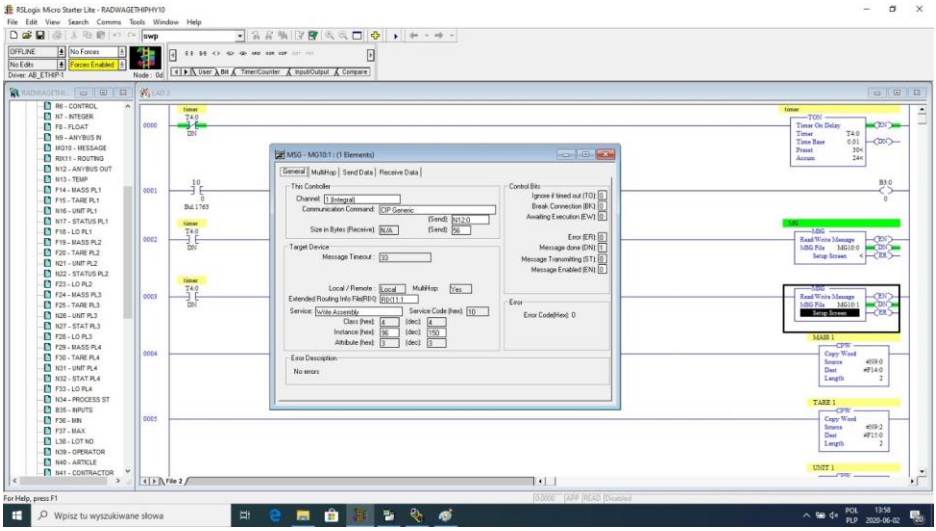
- Channel – wybieramy 1 (integral) co odpowiada EtherNet/IP.
- Communication Command – CIP Generic.
- Data Table Address – N9:0 – to nasz plik do odczytu danych.
- Size in Bytes – 104 – rozmiar tablicy rejestrów wejściowych.

- Extended Routing Info File – RIX11:0 – wskazujemy plik RIX.
- Service: Read assembly.
- Instance : 64.
- MulitHop: Yes.

Następnie przechodzimy do zakładki MultiHoop gdzie wpisujemy adres IP wagi.



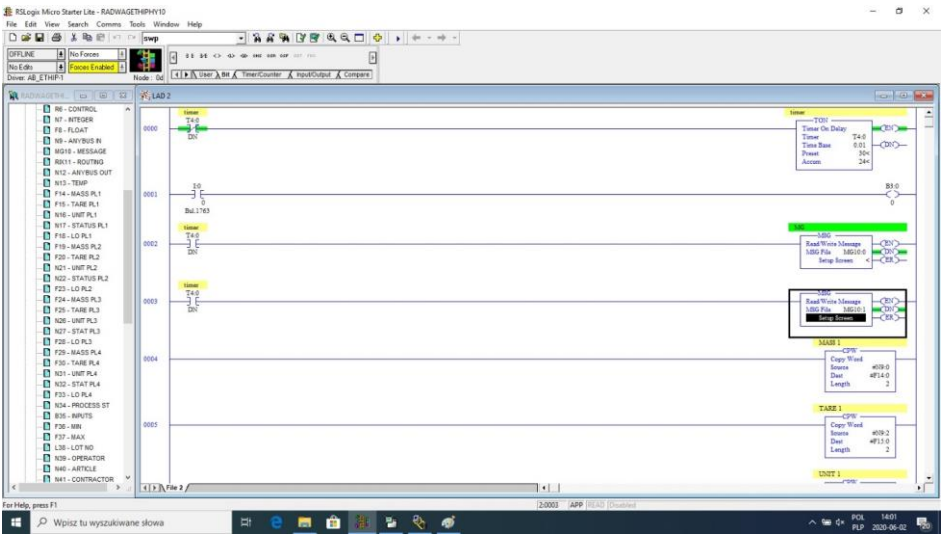
Analogicznie tworzymy funkcje do zapisu danych w wadze:



- Channel – wybieramy 1 (integral) co odpowiada EtherNet/IP™.
- Communication Command – CIP Generic.
- Data Table Address – N24:0 – to nasz plik do zapisu danych.
- Size in Bytes – 56 – rozmiar tablicy rejestrów wyjściowych.
- Extended Routing Info File – RIX11:1 – wskazujemy plik RIX.
- Service: Read assembly.
- Instance : 96.
- MulitHop: Yes.

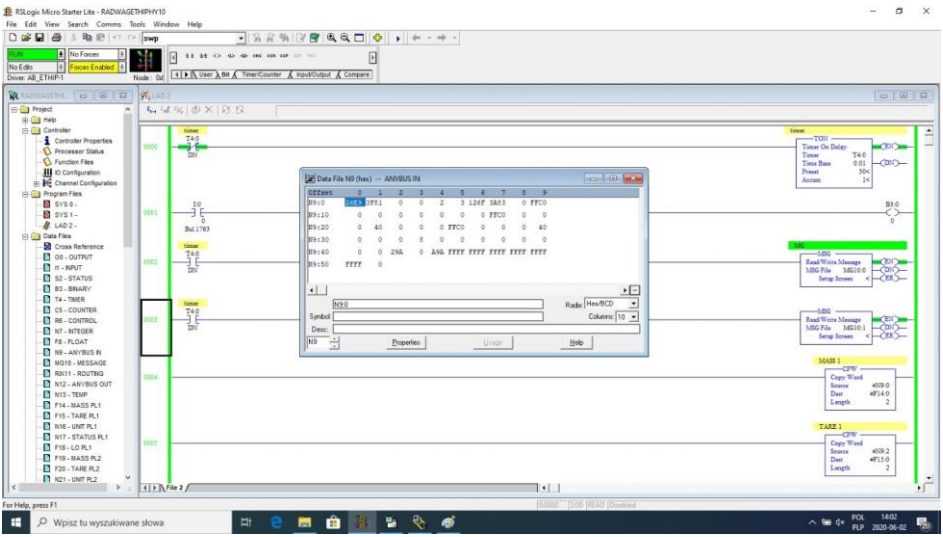
Następnie przechodzimy do zakładki MultiHoop gdzie wpisujemy adres IP wagi.

W przykładzie funkcje wyzwalane są poprzez timer, co pozwala regulować częstotliwość zapytań wysyłanych do wagi.



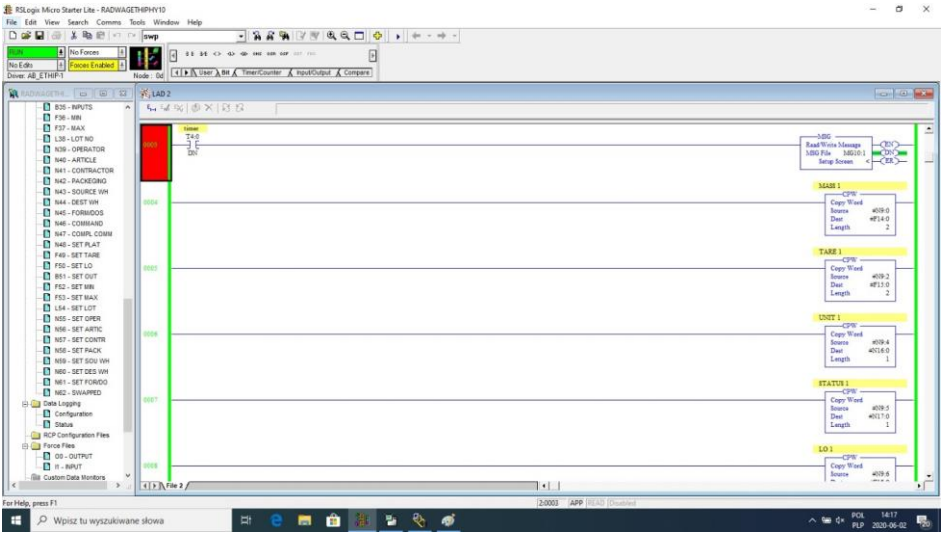
Można już załadować program do sterownika i uruchomić program.

Po połączeniu się z PLC (online) w pliku N9 powinniśmy już czytać dane a funkcja MSG nie powinna zwracać błędów.



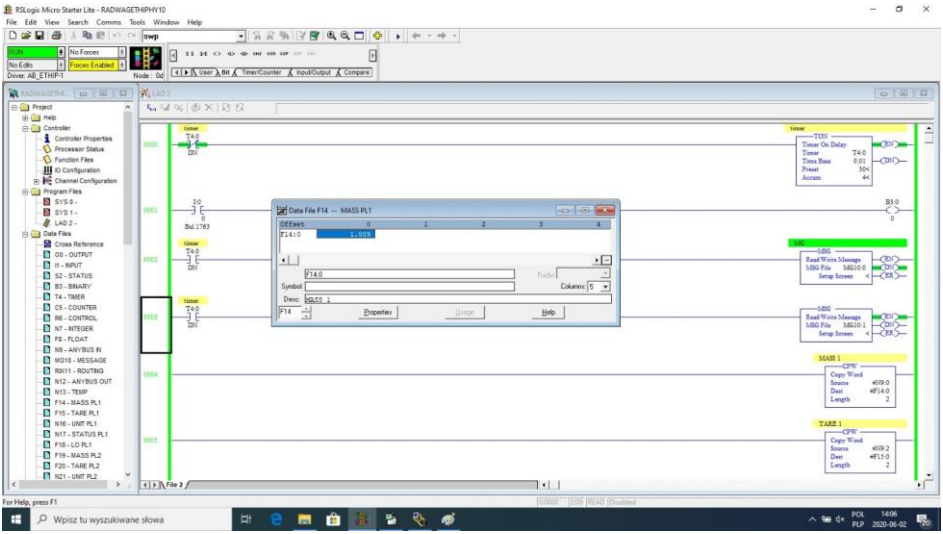
Dla zachowania porządku można stworzyć oddzielne pliki dla każdej zmiennej wagi.

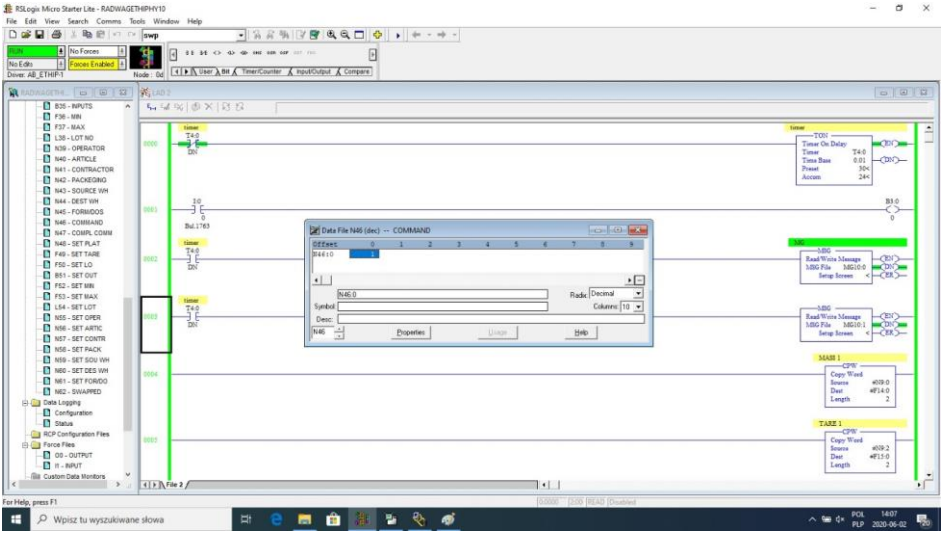
Dane pomiędzy plikami N9, N24 i plikami zmiennych przepisywane są za pomocą funkcji CPW. I tak dla przykładu funkcja do odczytu masy wygląda tak:



W efekcie w odpowiednich plikach powinniśmy czytać poprawnie dane z wagi.

Przykład odczytu masy:



Zapisując odpowiednie wartości w plikach odpowiadających rejestrom wyjściowym uruchamiamy poszczególne funkcje wagi.

Przykład zerowania wagi: