ROBOCHALLENGE

TwGateway Ethernet/IP - Profinet

Spis treści:

1. Opis zadania
    1.1 C6025
    1.2 FANUC Robot R30iB Plus
    1.3 EK9300
2. Etap 1 – konfiguracja EK9300
    2.1 Konfiguracja DIP Switchy
    2.2 Dodanie skanera Profinet w środowisku XAE
    2.3 Parametryzacja komunikacji Profinet
    2.4 Skanowanie sieci Profinet
    2.5 Skanowanie modułów
    2.6 Test komunikacji
3. Etap 2 – konfiguracja Ethernet/IP
    3.1 Dodanie Ethernet/IP Scanner
    3.2 Parametryzacja skanera Ethernet/IP
    3.3 Dodanie pliku EDS
    3.4 Skanowanie sieci Ethernet/IP
    3.5 Konfiguracja komunikacji z robotem
    3.6 Dodanie wejść i wyjść
    3.7 Sprawdzenie komunikacji
4. Etap 3 – Konfiguracja programu PLC
    4.1 Przygotowane środowisko
    4.2 Tworzenie zmiennych wejść / wyjść
    4.3 Linkowanie zmiennych
    4.4 Program główny
    4.5 Testy działania programu
5. Etap 4 – funkcja skalująca
    5.1 Wymagania dotyczące funkcji
    5.2 Testy działania funkcji
6. System oceniania

1 Opis zadania

Zadanie polega na stworzeniu bramki (Gateway), która pozwoli na przekazanie sygnałów IO pomiędzy Bus Couplerem EK9300 (Profinet) a robotem FANUC działającym po protokole Ethernet/IP.

Do dyspozycji w zadaniu jest:

→ Beckhoff C6025
→ FANUC Robot R30iB Plus
→ Beckhoff Profinet Bus Coupler EK9300

Uczestnik, w finalnej wersji zadania, ma na celu stworzenie aplikacji, która po restarcie sterownika:

→ Uruchomi robota w tryb gotowości do pracy
→ Przekaże sygnały z Joysticka do Robota w celu poruszania się
→ Przekaże sygnał z potencjometru do Robota w celu zmiany prędkości ruchu

Zadanie zostało podzielone na 4 etapy, gdzie każdy z nich jest punktowany 0-1pkt. Dokładny opis etapów znajduje się w późniejszej części instrukcji.

→ Etap 1:

Konfiguracja komunikacji z modułem EK9300 – Profinet RT.

https://infosys.beckhoff.com/content/1033/tf6281_tc3_ethernetipscanner/index.html

→ Etap 2:

Konfiguracja komunikacji z robotem FANUC – Ethernet/IP Scanner.

https://infosys.beckhoff.com/content/1033/tf6271_tc3_profinet_rt_controller/index.html

→ Etap 3:

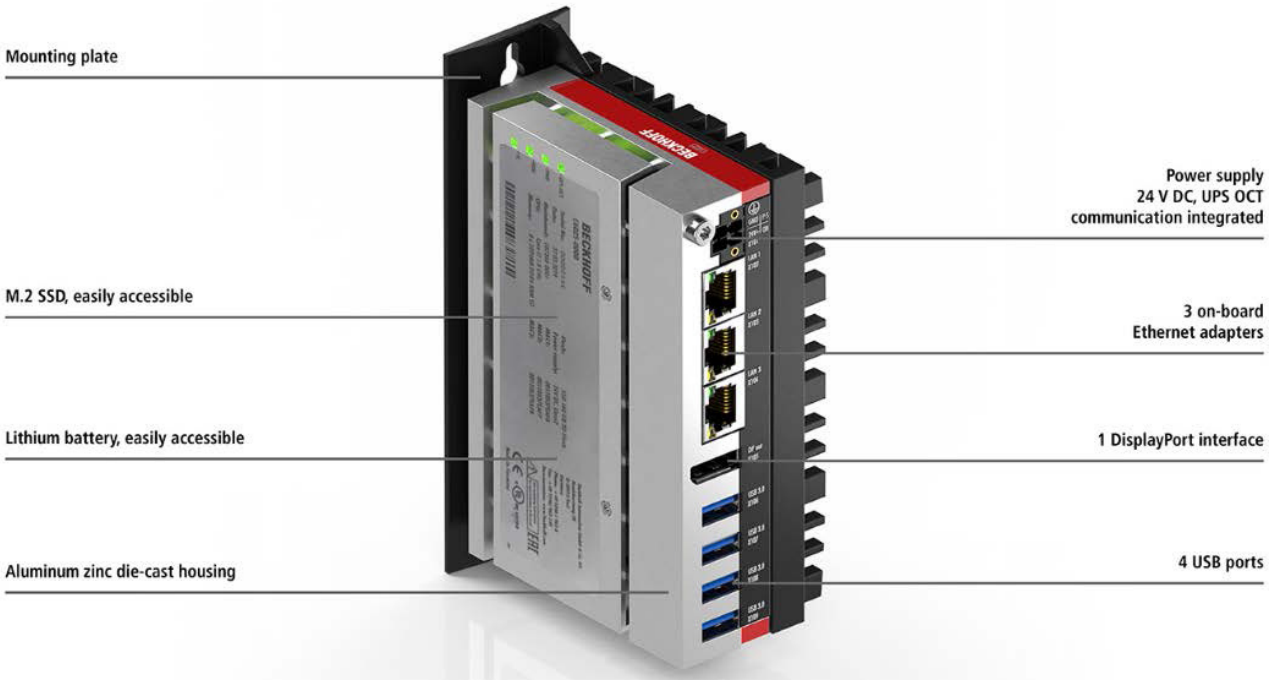
Konfiguracja programu w środowisku XAE.

→ Etap 4:

Stworzenie funkcji skalującej sygnał analogowy z potencjometru na wartość wysyłaną do robota.

https://infosys.beckhoff.com/english.php?content=../content/1033/tc3_plc_intro/2530284939.html

1.1 C6025



Sterownik C6025 pełni w tym zadaniu rolę Gateway'a.

Jest to bardzo wydajny sterownik, który posiada w sobie system operacyjny Windows 10 LTSC. Pozwala to wykorzystywać go jako komputer inżynierski. W tym celu zainstalowano na nim oprogramowanie TwinCAT 3 XAE w wersji 3.1.4024.25, na którym wykonywane będzie zadanie.

Posiada on 3 karty sieciowe, które pozwalają na odpowiednią konfigurację sieci. W obecnej aplikacji wykorzystywane są dwie z nich:

X102 – Komunikacja z EK9300 po protokole Profinet RT

X103 – Komunikacj z robotem FANUC po protokole Ethernet/IP

Identyfikację karty można dokonać w ustawieniach adapterów sieciowych.

Pełna dokumentacja znajduje się pod linkiem:

https://download.beckhoff.com/download/document/ipc/industrial-pc/c6025en.pdf

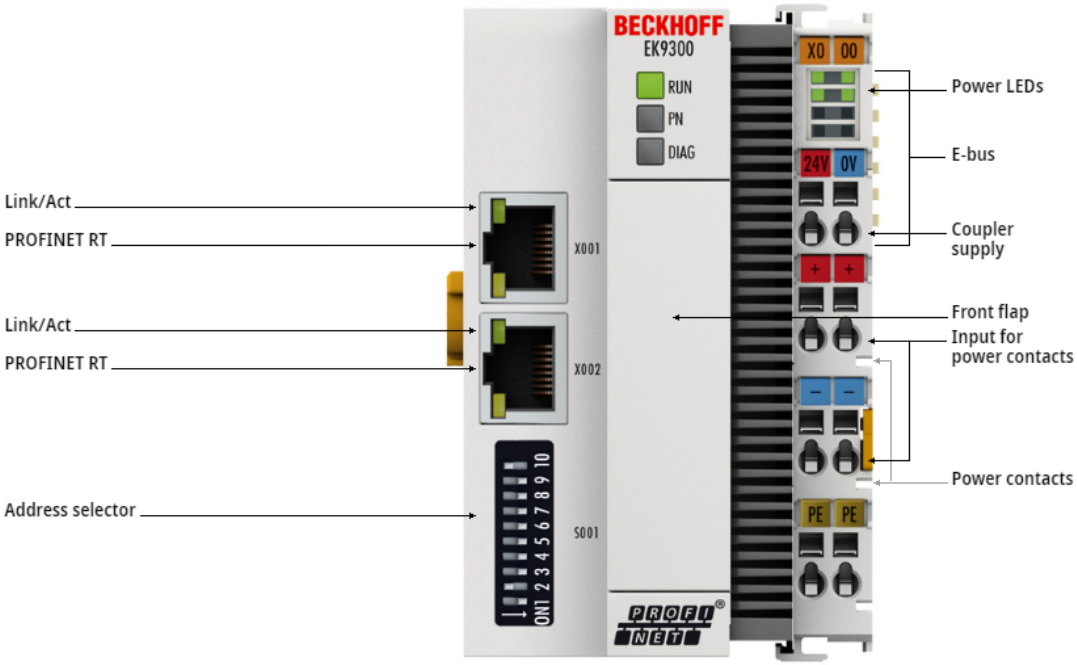
1.2 FANUC Robot R30iB Plus

Robot FANUC jest wyposażony w kontroler R-30iB Plus, który pozwala na wykorzystywanie oprogramowania należącego do firmy FANUC.

Komunikację z kontrolerem można uzyskać przy użyciu protokołu komunikacyjnego Ethernet/IP. Kontroler jest skonfigurowany w taki sposób, aby pełnił rolę adaptera.

W tej aplikacji nie ma konieczności dodatkowej konfiguracji oraz programowania robota.

1.3 EK9300



Bus Coupler EK9300 jest konfigurowalnym couplerem (łącznikiem sieci), który może komunikować się po protokole Profinet. Dodatkowo, jest on wyposażony w magistralę E-bus, która pozwala na dołączenie modułów EtherCAT'owych (ELxxxx).

Do EK9300 dołączone są moduły IO:

→ **EL1008** – Digital Input

→ **EL3008** – Analog Input

Moduły pozwolą na odczytywanie sygnałów z Joysticka oraz potencjometru zamieszczonych na stanowisku, a sygnały zostaną przekazane do sterownika po protokole Profinet.

Na module widoczne są 3 główne diody LED, które informują nas o stanie urządzenia. Zgodnie z dokumentacją:

Labelling	Meaning	Color	Meaning
RUN	Indicates the status of the coupler	red	May only light up during the start-up phase
		Green	Coupler is ready
		Blue (If red DIP switch 1 is set to on when starting the coupler)	The internal Flash can be reached via USB (firmware update)

LED PN	PROFINET status		Meaning
	green	red	
Power On	off	200 ms flashing	Start-up phase
No name	200 ms flashing	off	no Profinet name
No IP	1 s off, 200 ms on	off	No IP address
Run	on	off	OK

LED DIAG	PROFINET diagnosis		Meaning
	green	red	
Flashing, PN controller identification	500 ms	500 ms	The PN controller is transmitting an identification signal
No AR established	off	200 ms flashing	The establishment of a connection with the controller has not been completed
Device is in IO exchange Error display of Outputs CR is set to module differences	1 s off, 200 ms on	off	Problem with establishment of a connection or nominal and actual configuration differ
Device is in IO exchange but provider is in stop	200 ms	off	Coupler is in data exchange, but PLC is in stop
Device is in IO exchange	on	off	OK

Pełna dokumentacja znajduje się pod linkiem:
https://download.beckhoff.com/download/Document/io/ethercat-terminals/EK9300en.pdf

2 Etap 1 – konfiguracja EK9300

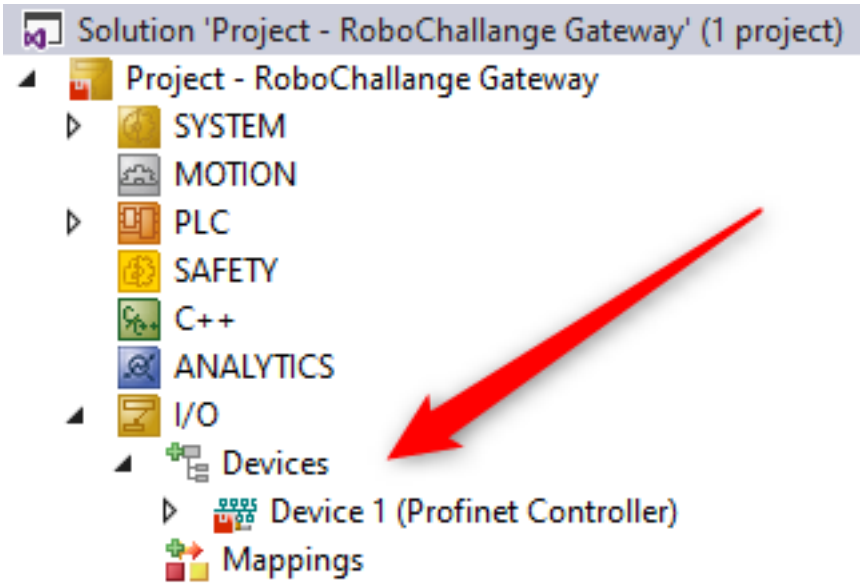
2.1 Konfiguracja DIP Switchy

Należy tak skonfigurować DIP Switche, aby łatwo zidentyfikować moduł w środowisku TwinCAT 3. Moduł należy uruchomić ze statycznym adresem IP: **172.16.17.123**.

DIP 9	DIP 10	Description DIP 1..8	Restart behavior	Behavior with factory settings
0	0	Last byte of the IP address via DIP switches 1 to 8	• PN name from memory • IP address via DIP switches 172.16.17.xxx (xxx DIP switch) SNM 255.255.0.0	• PN name becomes empty string • IP address via DIP switches 172.16.17.xxx (xxx DIP switch) SNM 255.255.0.0
0	1	DHCP DIP switch 1 to 8 set to OFF	• PN name from memory • IP address and SNM via DHCP	• PN name becomes empty string • IP address and SNM via DHCP
		DHCP DIP switch 1 to 8 set to ON	• PN name from memory • IP address from memory	• PN name becomes empty string • IP address 0.0.0.0
1	0	Reserved		
1	1	PROFINET-compliant DIP switch 1 to 8 set to OFF	• PN name from memory • IP address from memory	• PN name becomes empty string • IP address 0.0.0.0
		PROFINET with fixed name DIP switch 1 to 8 set to ON	• PN name via DIP switch 1 to 8 • IP address from memory	• PN name via DIP switch 1 to 8 • IP address 0.0.0.0

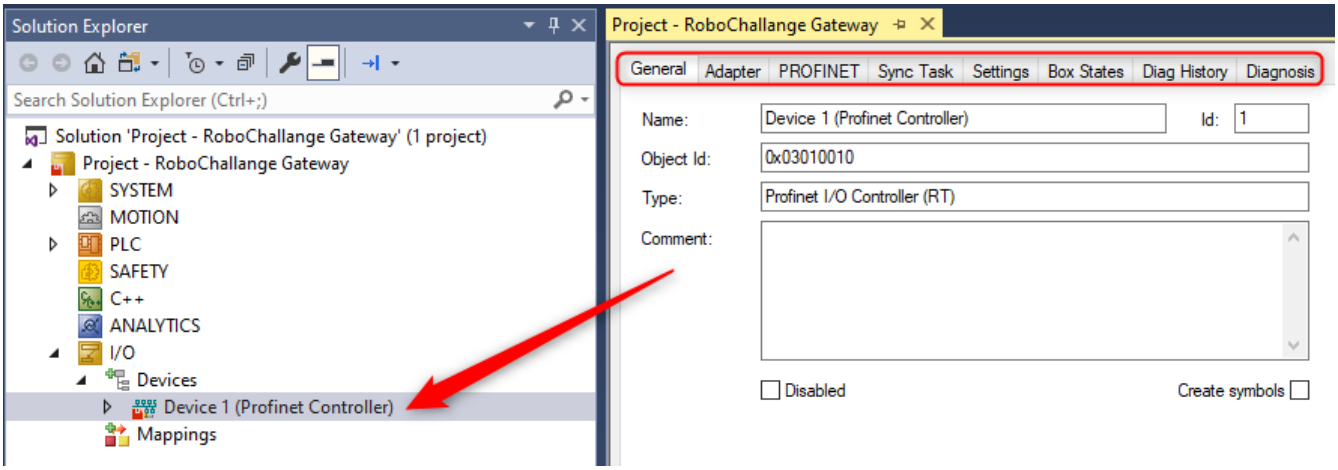
Konfiguracja zostanie zapisana po restarcie zasilania na module.

2.2 Dodanie skanera Profinet w środowisku XAE



2.3 Parametryzacja komunikacji Profinet

Pełnej parametryzacji komunikacji należy dokonać na kartach adaptera sieciowego:

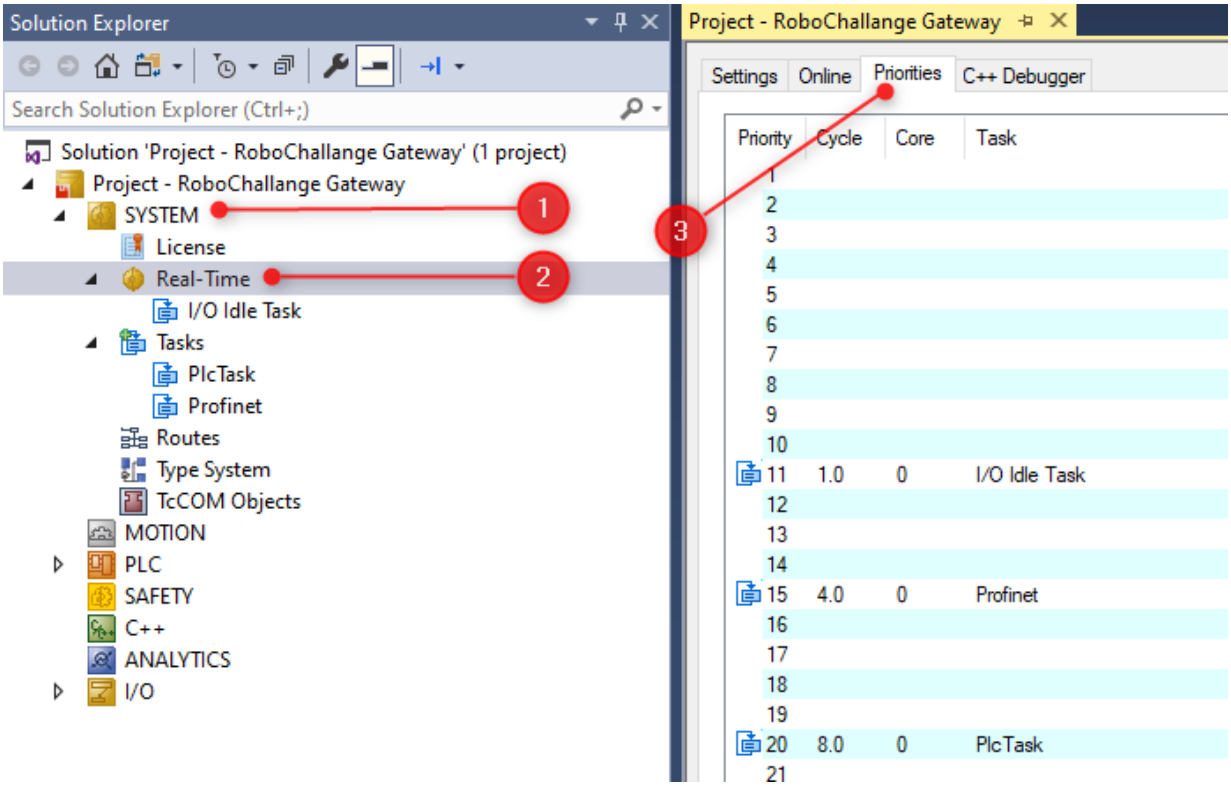


Podczas parametryzacji kontrolera należy:

→ Wybrać odpowiedni adapter sieciowy, na którym ma działać komunikacja Profinet. (Karta: **Adapter**)

→ Dodanie nowego i przypisanie tasku synchronizującego (Karta: **Sync Task**)
- o Nazwa tasku: **Profinet**
- o Czas cyklu powinien być z równy **2^n**
- o Priorytet znajdować się pomędzy **I/O Idle Task a PLC Task.**

Skonfigurowane priorytety oraz taski można zobaczyć tutaj:



→ Skonfigurowanie adresu IP oraz maski. Profinet Controller powinien znajdować się w tej samej podsieci co adapter sieciowy. Ostatni oktet powinien być unikalny dla sieci. **Maska:** 255.255.255.0. (Karta: **Settings**)

2.4 Skanowanie sieci Profinet

Skanowanie powinno odbywać się momencie, kiedy sterownik znajduje się w trybie **config**. W przeciwnym wypadku, opcja ta nie będzie dostępna.

Podczas skanowania, powinien zostać znaleziony moduł EK9300, który będzie posiadał już nadaną nazwę sieciową.

Należy nadać inną, unikalną nazwę sieciową. Zostanie ona podana przez opiekuna stanowiska. Przykładowa nazwa to: **robochallange01**.

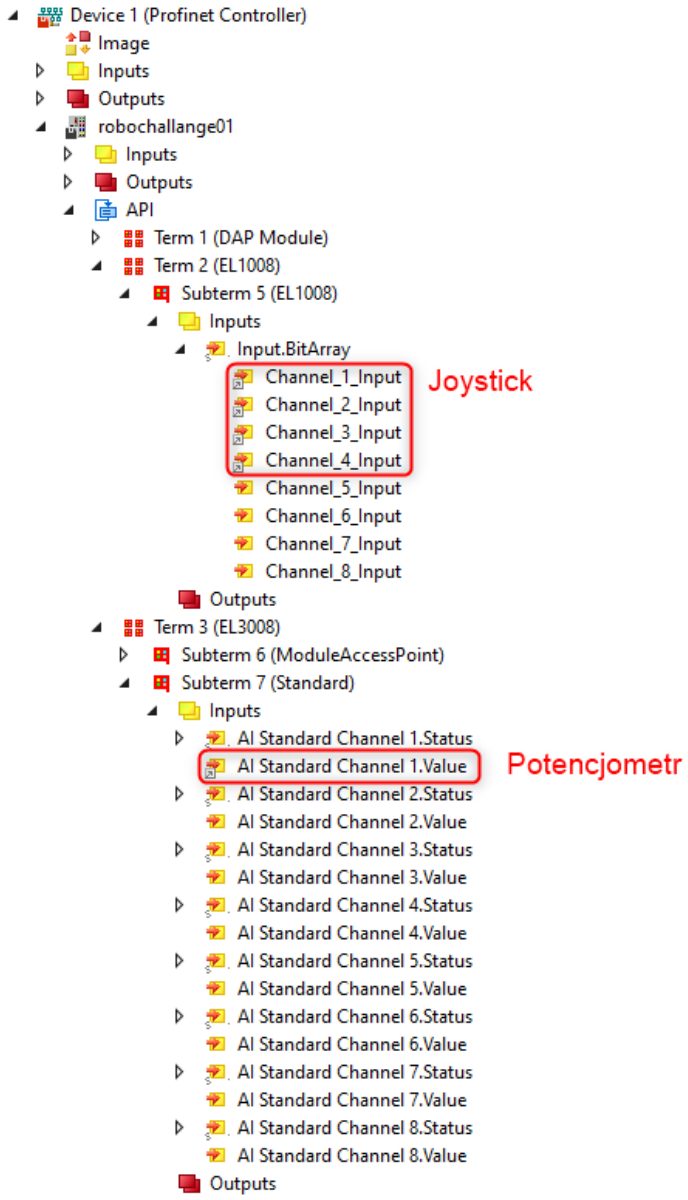
2.5 Skanowanie modułów

Po dodaniu modułu EK9300 do konfiguracji, oprogramowanie TwinCAT 3 automatycznie zapyta się o skanowanie w celu dodania konfiguracji urządzeń. Dzięki temu automatycznie zostanie uzupełniony plik GSDML.

2.6 Test komunikacji

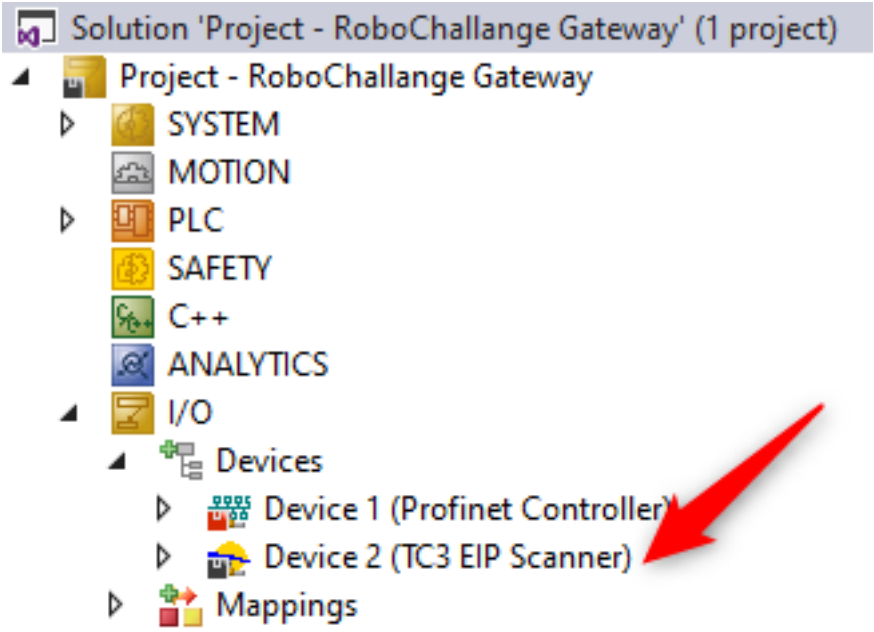
Po takim skonfigurowaniu, należy sprawdzić, czy sygnały wysyłane przez Joystick oraz potencjometr są odpowiednio odczytywane w programie.

Należy pamiętać, że aby odczytywać te sygnały należy mieć uruchomioną opcję **Free Run** podczas bycia w stanie **config**.



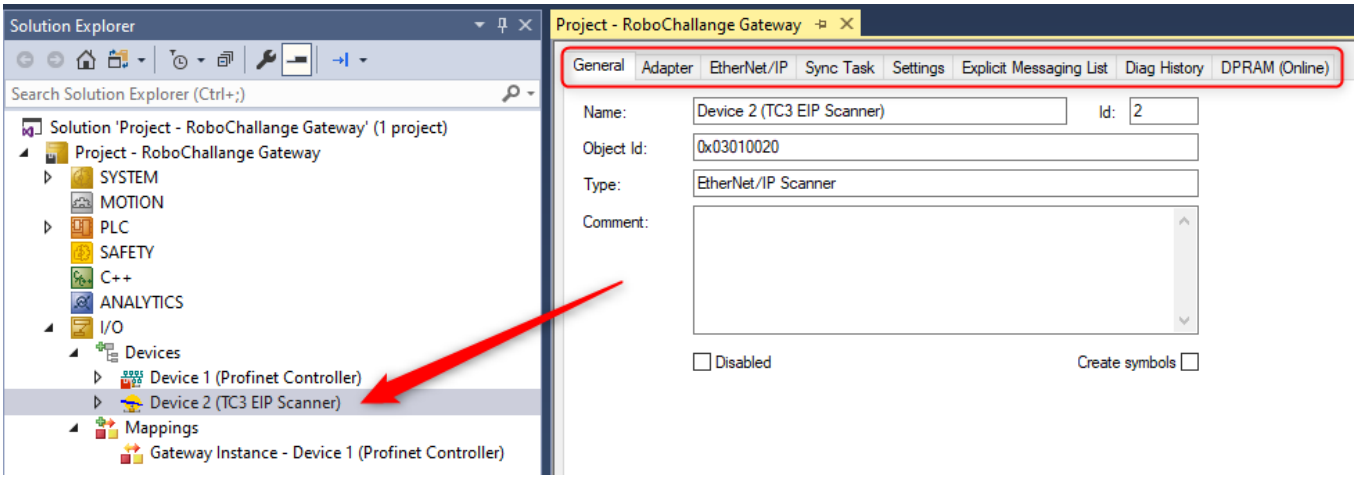
3 Etap 2 – konfiguracja Ethernet/IP

3.1 Dodanie Ethernet/IP Scanner



3.2 Parametryzacja skanera Ethernet/IP

Pełnej parametryzacji komunikacji należy dokonać na kartach adaptera sieciowego:



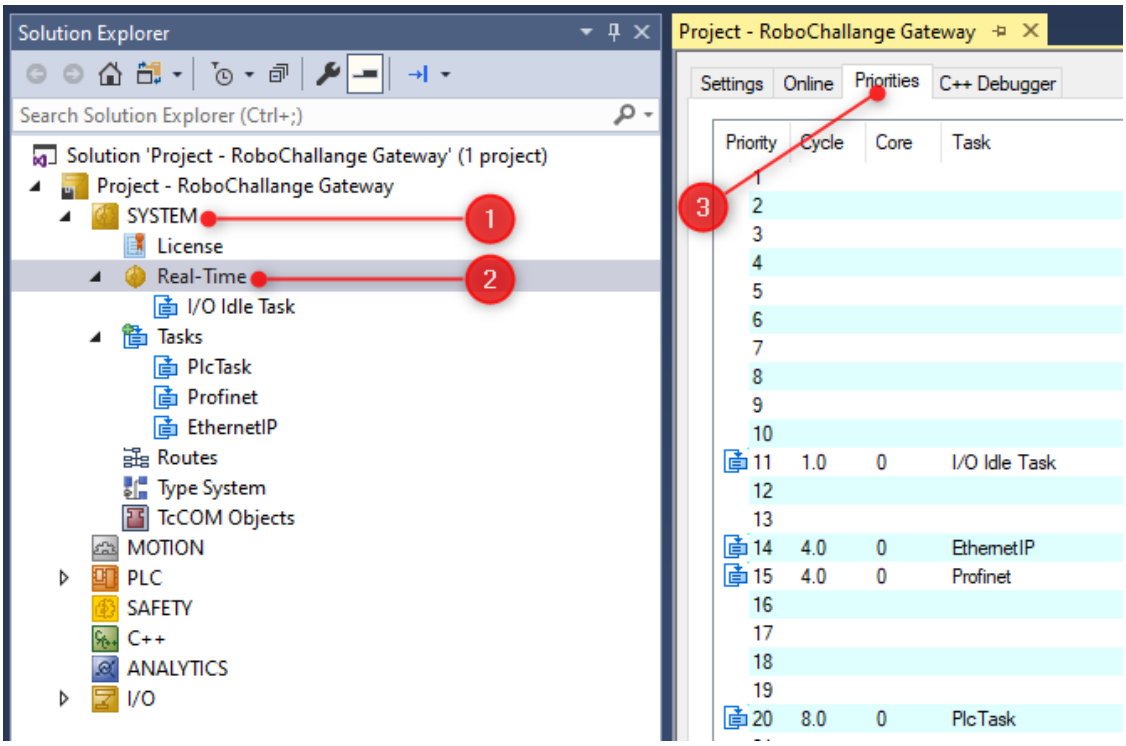
W celu sparametryzowania kontrolera należy:

→ Wybrać odpowiedni adapter sieciowy, na którym ma działać komunikacja Ethernet/IP. (Karta: **Adapter**)

→ Dodanie nowego i przypisanie tasku synchronizującego (Karta: Sync Task)

- o Nazwa tasku: **EthernetIP**
- o Czas cyklu powinien wynosić **4ms**
- o Priorytet znajdować się pomędzy **I/O Idle Task a PLC Task**.

Skonfigurowane priorytety oraz taski można zobaczyć tutaj:



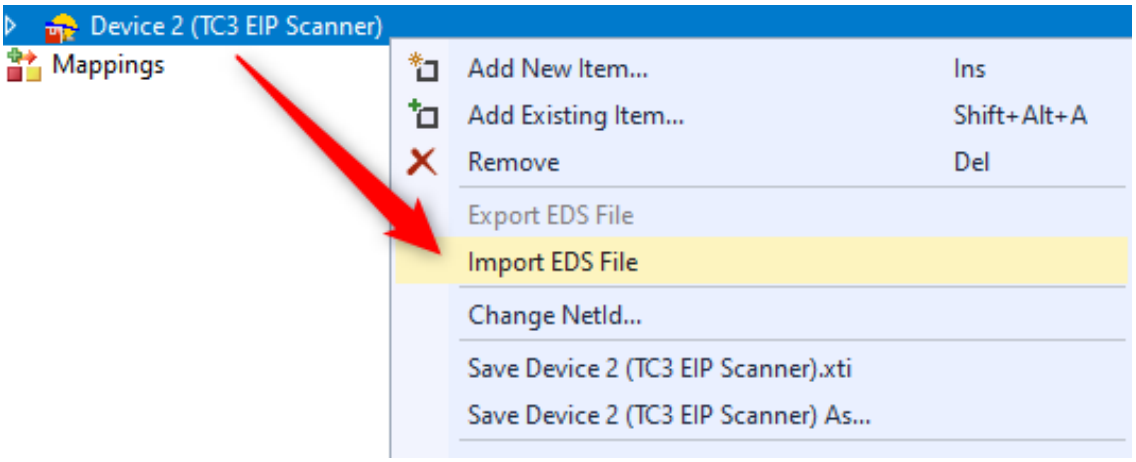
→ Skonfigurowanie adresu IP oraz maski. Ethernet/IP Scanner powinien znajdować się w tej samej podsieci co adapter sieciowy. Ostatni oktet powinien być unikalny dla sieci. Maska: **255.255.255.0**. (Karta: **Settings**)

3.3 Dodanie pliku EDS

W celu poprawnej komunikacji z robotem FANUC, należy zaimportować dedykowany plik o rozszerzeniu .eds. Plik znajduje się na pulpicie. Plik można dodać na dwa sposoby:

→ Dodanie do głównego katalogu:
C:\TwinCAT\3.1\Config\Io\EtherNetIP

→ Dodanie z poziomu TwinCAT 3



3.4 Skanowanie sieci Ethernet/IP

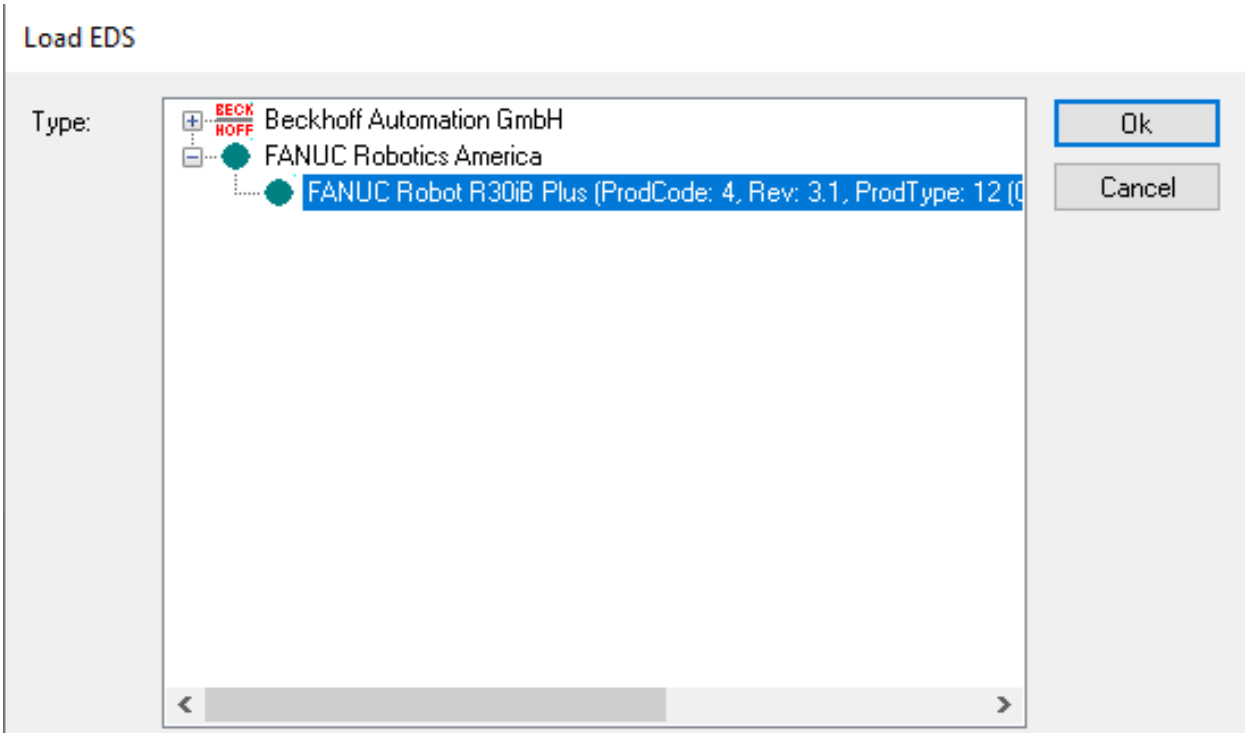
Skanowanie powinno odbywać się momencie, kiedy sterownik znajduje się w trybie config. W przeciwnym wypadku, opcja ta nie będzie dostępna.

Podczas skanowania, powinien zostać znaleziony robot FANUC, który będzie posiadał swój ad-

res IP. Znalezione urządzenie należy dodać do konfiguracji.

3.5 Konfiguracja komunikacji z robotem

Znalezioną konfigurację robota należy skonfigurować tak, aby była zgodna z plikiem EDS. W tym celu, należy załadować dodany wcześniej plik EDS do konfiguracji.



3.6 Dodanie wejść i wyjść

Do wymiany informacji pomiędzy urządzeniami należy dodać konfigurację IO zgodną z plikiem EDS:

→ PPM na zeskanowane urządzenie: **FANUC Robot R30iB**

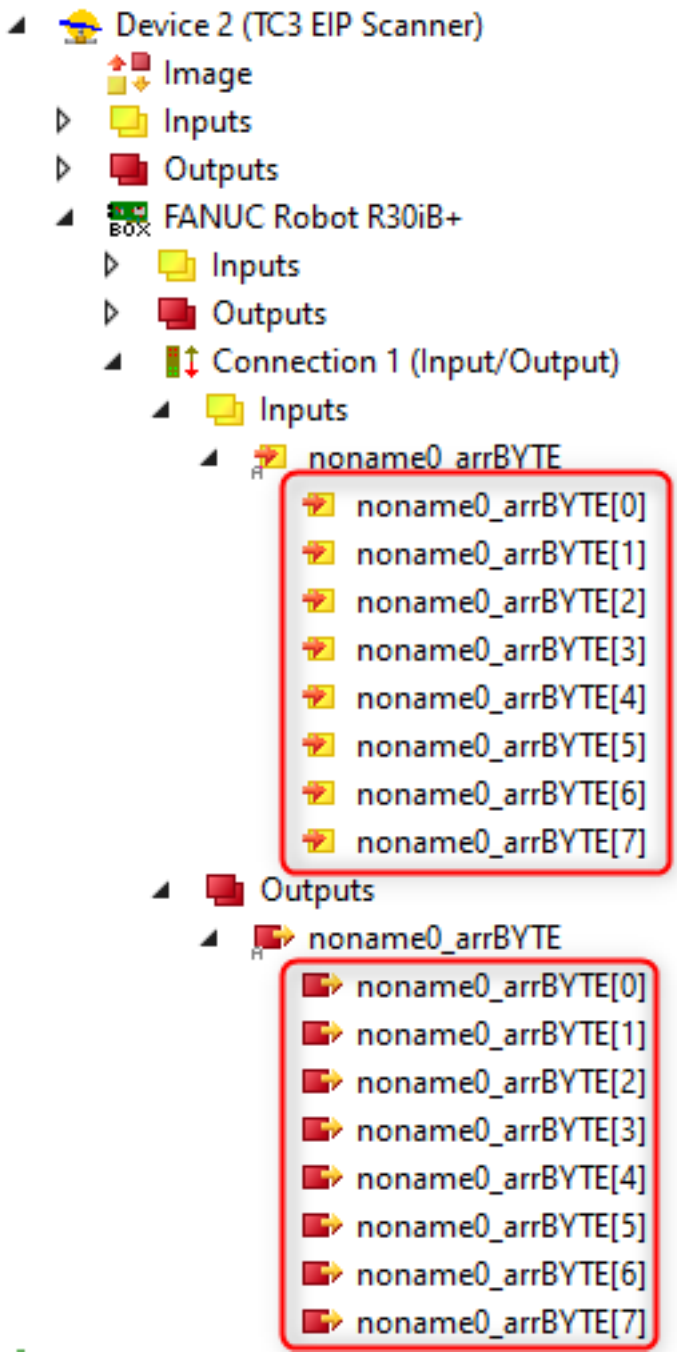
→ Rozwinąć opcję **Append IO Connection**

→ Wybrać opcję **Adaptor on Slot 1**

Uzyskujemy wtedy możliwość wymiany informacji przez:

→ 8 x Byte na wejściu

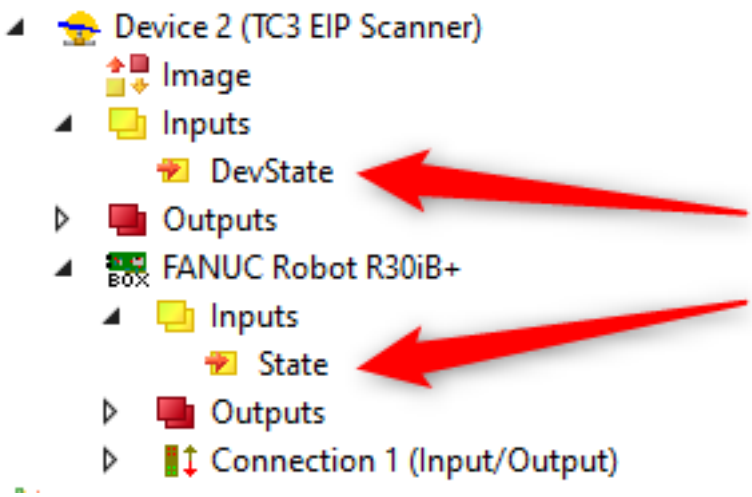
→ 8 x Byte na wyjścu



Pełny opis funkcjonalności bitów jest opisany w tabelach na końcu rozdziału.

3.7 Sprawdzenie komunikacji

Po przeładowaniu sterownika w tryb **config**, wartości na wejściach powinny wskazywać **0**:



W momencie braku komunikacji z robotem, należy dokonać jego restartu. Jeżeli po restarcie nadal wartości nie są równe 0, należy ponownie sprawdzić konfigurację parametrów.

Konfiguracja wyjść:

FANUC	Beckhoff Ethernet/IP	BIT	Nazwa
WORD 1	BYTE 0	0	IMSTP
		1	HOLD
		2	SFSPD
		3	CSTOPI
		4	RESET
		5	START
		6	ENBL
		7	
	BYTE 1	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
WORD 2	BYTE 2	0	Sygnał kierunku
		1	Sygnał kierunku
		2	Sygnał kierunku
		3	Sygnał kierunku
		4	
		5	
		6	
		7	
	BYTE 3	0	Prędkość
		1	Prędkość
		2	Prędkość
		3	Prędkość
		4	Prędkość
		5	Prędkość
		6	Prędkość
		7	Prędkość
WORD 3	BYTE 4	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
	BYTE 5	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
WORD 4	BYTE 6	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
	BYTE 7	0	
		1	
		2	SetProgram
		3	
		4	
		5	
		6	PNS_STROPE
		7	PRG_START

Konfiguracja wyjść:

FANUC	Beckhoff Ethernet/IP	BIT	Nazwa
WORD 1	BYTE 0	0	
		1	
		2	
		3	
		4	
		5	
		6	Enable
		7	
	BYTE 1	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
WORD 2	BYTE 2	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
	BYTE 3	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
WORD 3	BYTE 4	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
	BYTE 5	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
WORD 4	BYTE 6	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	
	BYTE 7	0	
		1	
		2	
		3	
		4	
		5	
		6	
		7	

4 Etap 3 – Konfiguracja programu PLC

4.1 Przygotowane środowisko

W programie PLC znajduje się:

→ Program **RobotStartup**, który służy do rozpoczęcia pracy robota

→ Struktury komunikacyjne z robotem:
- o ST_FanucInputState
- o ST_FanucOutputState
- o ST_FanucOutputProgram

→ Pliki zmiennych globalnych:

o GVL_EthernetIP

W tym pliku należy zadeklarować wszystkie zmienne linkowane z **EthernetIP Scanner**.

o GVL_Profinet

W tym pliku należy zadeklarować wszystkie zmienne linkowane z **Profinet Controller**

4.2 Tworzenie zmiennych wejść / wyjść

Do deklaracji zmiennych wejściowych lub wejściowych wykorzystuje się specjalny sposób zapisu:

→ Wejście:

bInput AT %I* : BOOL;

→ Wyjście:

bOutput AT %Q* : BOOL;

Struktury, które odpowiadają za inicjalizację robota są już zadeklarowane w pliku:

```
1  {attribute 'qualified_only'}
2  VAR_GLOBAL
3      FanucStateFBByte        AT %I* : ST_FanucInputState;
4
5      FanucStateByte          AT %Q* : ST_FanucOutputState;
6      FanucProgramByte        AT %Q* : ST_FanucOutputProgram;
7  END_VAR
```

Dodatkowo należy zadeklarować:

→ GVL_EthernetIP

Zmienne wyjściowe, które pozwolą na sterowanie odpowiednimi bitami zgodnymi z tabelą w rozdziale 3. Zmienne te zaznaczono kolorem **czerwonym**. Pozostałe zmienne znajdują się już w zadeklarowanych strukturach, które również należy podlinkować.

→ GVL_Profinet

Zmienne wejściowe, które będą odczytywane z modułów EL1008 oraz EL3008 po protokole Profinet.

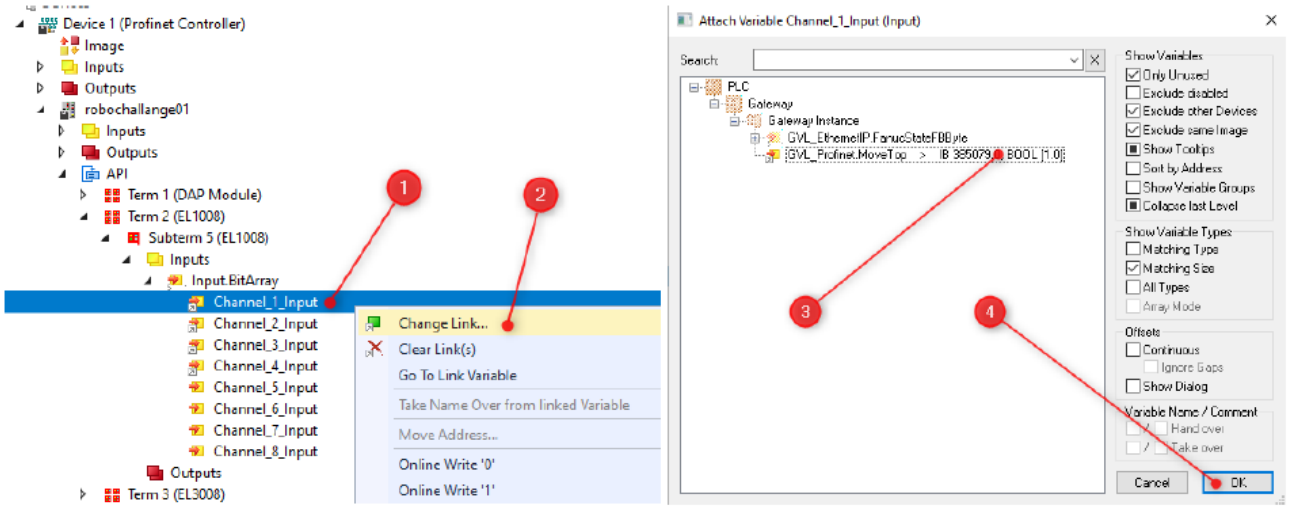
Będą to:

- o 4x zmienna BOOL
- o 1x zmienna INT

4.3 Linkowanie zmiennych

Przed linkowaniem zmiennych, należy upewnić się, że projekt został przebudowany (Build -> Build Solution).

Linkowanie odbywa się poprzez kliknięcie na zmienną w karcie IO, a następnie wybranie z listy. Na liście, domyślnie pokazują się tylko te zmienne, które posiadają taką samą wielkość. Wyjątkiem są zmienne **BOOL i BIT**.



Raz podlinkowana zmienna, nie będzie widoczna dla kolejnych linkowań.

Należy podlinkować **WSZYSTKIE** niezbędne zmienne, które zostały zadeklarowane w plikach **GVL_EthernetIP** oraz **GVL_Profinet**.

Po zalinkowaniu zmiennych należy aktywować konfigrację.

4.4 Program główny

W programie głównym– MAIN – należy:

→ Wywołać program inicjalizujący pracę robota: **RobotStartup**

→ Przepisać sygnały z wejść **Profinet Controller** na odpowiednie bity na wyjściu **EthernetIP** zgodne z tabelą z rozdziału 3. Są to:

- o 4 sygnały cyfrowe określające kierunek poruszania się
- o 1 sygnał analogowy określający prędkość *

*** Z uwagi na różny zakres wartości otrzymywanych z potencjometru od zakresu prędkości, bezpośrednie przepisanie będzie możliwe dopiero po wykonaniu ostatniego etapu – Funkcja skalująca. W tym momencie, należy ustawić sztywną wartość z zakresu <1-30>.**

Jeżeli istnieje potrzeba odwołania się do pojedynczego bitu, należy to zrobić w taki sposób:

byVar.0 := TRUE; - zapis ten oznacza, że zerowy bit (najmniej znaczący) zmieni swoją wartość z 0 na 1.

→ Uzależnić działanie programu, czyli przepisywanie zmiennych na wejścia robota, od stanu sygnału **Enable** struktury **ST_FanucInputState** przy użyciu instrukcji warunkowej **IF**.

4.5 Testy działania programu

Po zakończonej konfiguracji, należy wgrać program na sterownik w celu przetestowania funkcjonalności. W przypadku niepowodzenia, należy odpowiednio zmodyfikować program.

5 Etap 4 – funkcja skalująca

Funkcja skalująca będzie przeliczała wartość analogową, uzyskaną na wejściu modułu EL3008, na taką która będzie znajdowała się w podanym zakresie. Pozwoli to uzależnić prędkość poruszania się robota od zadanej wartości na potencjometrze.

5.1 Wymagania dotyczące funkcji

Wymogi stawiane funkcji skalującej:

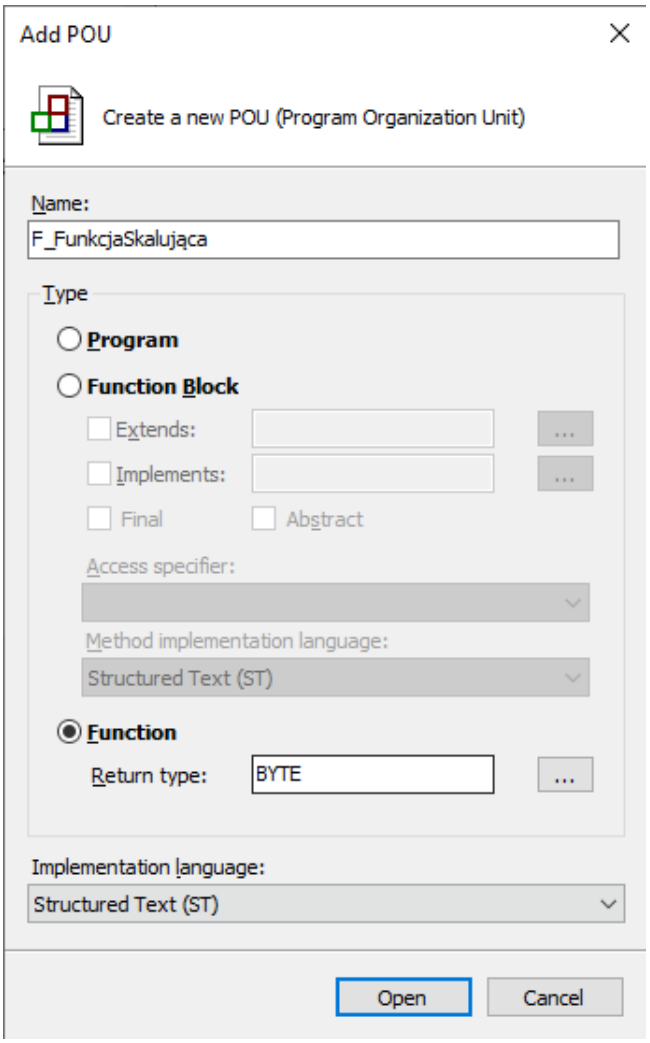
→ Nazwa funkcji: **F_FunkcjaSkalująca**

→ Na wejściu podawana jest wartość analogowa **INT** z zakresu **<0-32767>**

→ Na wyjściu otrzymujemy przeskalowaną wartość **BYTE** z zakresu **<1-30>**

→ Funkcja zabezpiecza się przed podaniem wartości ujemnych, ponieważ na wejściu EL3008, sygnał z potencjometru może pokazywać wartości **<0**

Funkcja powinna zostać napisana w języku **ST**.



5.2 Testy działania funkcji

Po napisaniu funkcji skalującej, należy odpowiednio uzupełnić program MAIN, aby przy użyciu potencjometru sterować prędkością robota. [05] Gateway

Dodajemy na końcu punkt

6. System oceniania

1. Osiągnięcie celu zadania – 1 pkt

Zadanie polega na stworzeniu bramki (Gateway), która pozwoli na przekazanie sygnałów IO pomiędzy Bus Couplerem EK9300 (Profinet) a robotem FANUC działającym po protokole Ethernet/IP.

2. Zaliczenia etapów – łącznie 4 pkt

a) Konfiguracja komunikacji z modułem EK9300 – Profinet RT. **(1 pkt.)**

b) Konfiguracja komunikacji z robotem FANUC – Ethernet/IP Scanner. **(1 pkt.)**

c) Konfiguracja programu w środowisku XAE. **(1 pkt.)**

d) Stworzenie funkcji skalującej sygnał analogowy z potencjometru na wartość wysyłaną do robota. **(1 pkt.)**

3. Czas wykonania po zapoznaniu się z instrukcją – maksymalnie 3 pkt

a) Ukończenie zadania poniżej 30 min **(3 pkt)**

b) Ukończenie zadania poniżej 35 min **(2 pkt)**

c) Ukończenie zadania poniżej 40 min **(1 pkt)**