

PRACA DYPLOMOWA

MAGISTERSKA

**Układ sterowania czasu rzeczywistego położeniem kulki na
pochylni z wykorzystaniem sieci PROFINET**

Autor: mgr inż. Marcin Faszczewski

Współautor: mgr inż. Mariusz Czajkowski



SPIS TREŚCI

Spis treści	3
Wstęp	5
Cel i zakres.....	6
1. Zagadnienia związane z programowaniem sterownika PLC	7
1.1. Oprogramowanie narzędziowe Simatic Manager	8
1.2. Operacje na blokach w projekcie	10
1.3 Typy danych w programie Step 7	13
1.4 Adresowanie pamięci systemowej sterownika PLC	14
2. Koncepcja układu sterowania PLC.....	15
2.1. Budowa układu sterowania	17
2.2. Charakterystyka programowa elementów układu sterowania	19
2.2.1. Charakterystyka programowa układu przetwarzania sygnałów.....	19
2.2.2. Charakterystyka programowa elementów pomiarowych i elementu wykonawczego.....	23
2.2.3. Konfiguracja i parametryzacja stacji PLC	24
3. Projektowanie sieci PROFINET dla komunikacji w czasie rzeczywistym	31
3.1. Ustawianie cyklu wymiany danych i cyklu odświeżania informacji przez urządzenia w sieci PROFINET	32
3.2. Obsługa czasu rzeczywistego.....	34
4. Programowanie obsługi wejść/wyjść.....	39
4.1. Obsługa programowa peryferii	39
4.1.1. Wymiana danych w sieci PROFINET	39
4.1.2. Sterowanie peryferyjnymi wejściami i wyjściami cyfrowymi	42
4.1.3. Forsowanie zmiennych.....	45
4.1.4. Skalowanie wartości analogowych	46
4.2. Linearyzacja charakterystyki czujnika optycznego	47
4.3. Sterowanie PWM serwomechanizmem	53
5. Programowanie algorytmów sterowania cyfrowego	57
5.1. Algorytmy różnicowe regulatora PID dla potrzeb sterowania PLC	58
5.2. Funkcje do obsługi PID w PLC	60
6. Testowanie algorytmów sterowania	68

6.1. Dobór nastaw regulatora	71
6.2. Jakość dynamiczna układu regulacji PLC	79
Podsumowanie	87
Literatura	89
Załącznik 1 – Listing programu sterowania	Błąd! Nie zdefiniowano zakładki.
Z 1.1. Blok OB1	92
Z 1.2. Blok FB7 – sterowanie przyciskami i diodami	96
Z 1.3. Blok FB8 – Forsowanie zmiennych	100
Z 1.4. Blok FB3 – Linearyzacja i sterowanie odczytem czujników	104
Z 1.5. Blok OB35 – Cykliczne przerwania	107
Z 1.6. Blok FB4 – Generowanie sygnału PWM	110
Z 1.7. Blok OB100 – restart regulatora PID (STL)	111

WSTĘP

System kulki na pochylni jest popularnym przykładem niestabilnego układu spotykanego w teorii sterowania [1], który składa się z kulki, pochylni, czujników odległości i silnika. Podstawowym założeniem jest wykorzystanie wychylenia wału silnika, do kontroli położenia kulki na pochylni, po której kula toczy się swobodnie. Dzięki zastosowaniu technik matematycznych, informacje z czujnika mogą być odczytane jako realne położenie kulki na pochylni z dokładnością 5mm i porównane z pożądaną pozycją kulki. Różnica podawana jest na wejście regulatora PID, który na wyjściu zwraca sygnał sterujący silnikiem w celu uzyskania pożądanej pozycji kulki na pochylni.

Układ sterowania zaprojektowany został w oparciu o sterownik przemysłowy PLC Siemens, odpowiedzialny za algorytm sterowania, wyspę wejść/wyjść peryferyjnych ET200S *Siemens* połączoną ze sterownikiem za pośrednictwem sieci PROFINET, umożliwiającej działanie układu sterowania w czasie zbliżonym do rzeczywistego oraz skonstruowaną w ramach potrzeb projektu konsoli sterującej.

Układ sterowania kulką na pochylni, mimo swojej prostoty, stanowi podstawę dla wielu typowych realnych systemów [2,3], jak pozioma stabilizacja samolotu podczas lądowania w turbulentnym przepływie powietrza. Jednak podstawową zaletą budowy takiego układu jest możliwość zbadania zachowania się obiektu pod wpływem zmian nastaw regulatora PID, które odbywają się w czasie rzeczywistym. Obiekt klarownie odzwierciedla działanie regulatora, dzięki czemu użytkownik stanowiska łatwiej przyswaja zasadę jego działania.

Ze względu na wiele niedogodności powodujących niestabilność układu, takich jak nieidealna jednorodność kulki, układ staje się trudnym do stabilizacji. Pomocne okazuje się narzędzie zawarte w pakiecie programowym sterownika, służące do wizualizacji charakterystyki odległościowo czasowej kulki na pochylni.

Do pracy dodano załącznik zawierający listing całego programu sterowania:
Załącznik 1- Listing programu sterowania.

CEL I ZAKRES

Celem pracy jest projekt układu sterowania położeniem kulki na pochylni w czasie rzeczywistym z wykorzystaniem sieci PROFINET. Przedstawione zostaną poszczególne funkcje algorytmu automatycznej regulacji, konfiguracja i parametryzacja stacji PLC i sieci PROFINET oraz badania układu sterowania. Projekt został opracowany z myślą o możliwości przeprowadzania na nim późniejszych badań w celach dydaktycznych.

Zakres pracy obejmuje:

- przedstawienie teoretyczne pakietu narzędziowego do programowania sterowników PLC *SIMATIC* marki *Siemens*,
- przedstawienie zagadnień związanych z programowaniem sterownika PLC,
- opis konfiguracji i parametryzacji stacji PLC oraz sieci PROFINET,
- przedstawienie elementów układu sterowania z perspektywy programowej,
- przedstawienie algorytmów sterowania układu automatycznej regulacji,
- testowanie algorytmów sterowania oraz dobór parametrów regulatora PID.

W części teoretycznej przedstawiono aplikacje oraz zagadnienia związane z programowaniem sterownika PLC dla potrzeb projektu. W dalszej kolejności zamieszczono dokładny opis układu sterowania zarówno z perspektywy konstrukcyjnej jak i programowej. Przedstawiono również sposób konfiguracji i parametryzacji stacji PLC oraz sieci PROFINET.

W rozdziale czwartym zaprezentowano obsługę programową wejść i wyjść peryferyjnych dla potrzeb układu sterowania. Przedstawiono również rozwiązanie problemu nieliniowej charakterystyki napięciowo-odległościowej czujnika optycznego oraz sposób generowania sygnału sterującego silnikiem.

Rozdział piąty zawiera algorytmy różnicowe regulatora PID dla potrzeb sterowania PID oraz dokładne opisy funkcji do obsługi regulatora PID.

W rozdziale szóstym przedstawione zostały sposoby doboru nastaw parametrów regulatora oraz wyniki badań zachowania układu sterowania pod wpływem automatycznej regulacji.

1. ZAGADNIENIA ZWIĄZANE Z PROGRAMOWANIEM STEROWNIKA PLC

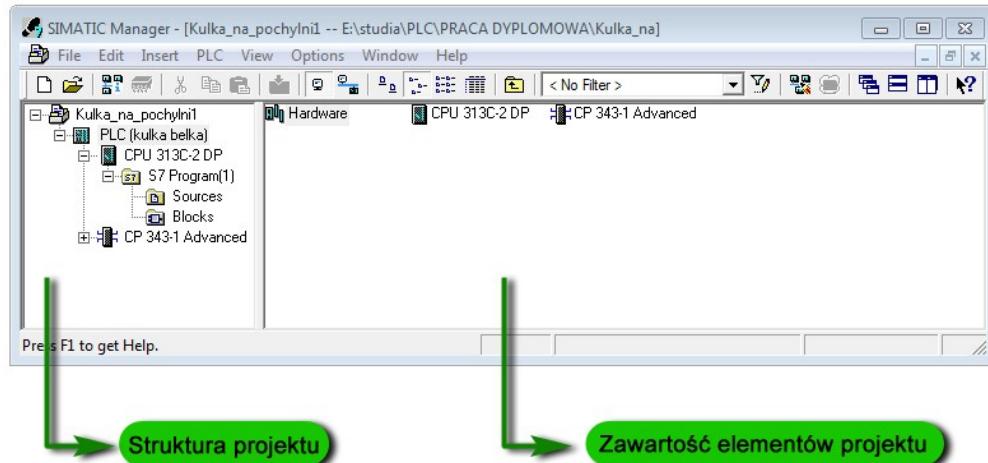
W projekcie układu sterowania czasu rzeczywistego położeniem kulki na pochylni w sieci PROFINET wykorzystano oprogramowanie narzędziowe *Simatic Manager*, pełniącego rolę programu nadziednego w środowisku *Step7*, służącego do zarządzania projektem. Pakiet *Step7* (Tabela 1.1.) składa się ze zbioru narzędzi, potrzebnych do programowania i obsługi sterowników PLC. Proces programowania, konfiguracji i parametryzacji sterownika PLC składa się z mniejszych podprocesów, które są wzajemnie powiązane. Zarządzanie wszystkimi składnikami sprzętowymi i programowymi procesu automatyzacji odbywa się wewnątrz projektu *Simatic Manager* (Rys. 1.1) [4,5].

Tabela 1.1. Narzędzia Simatic Manager.

Element Step7		Opis
NCM S7	Diagnostics	Konfigurowanie Komunikacji
	Firmware Loader	
Configure Simatic Workspace		Konfigurowanie systemu wielu użytkowników
Converting S5 Files		Konwertowanie programów napisanych w <i>Step 5</i> na odpowiadające im programy w <i>Step 7</i>
HW Config		Konfiguracja i parametryzacja sprzętowa
LAD, STL, FBD – Programming S7 Blocks		Uruchomianie edytora program użytkownika
Memory Card Parameter Assignment		Kopiowanie program użytkownika na zewnętrzny moduł pamięci
NetPro – Configuring Networks		Konfigurowanie sieci
PID Control Parameter Assignment		Wstawianie parametrów wbudowanych w pakiet podstawowy <i>Step 7</i> , przeznaczonych do projektowania programowych regulatorów PID
S7 SCL – Programming S7 Blocks		Programowanie strukturalne wyższego poziomu oparte na zasadzie działania języka <i>Pascal</i>
S7-GRAPH – Programming Sequential Control Systems		Programowanie sterowań sekwencyjnych. Polega ono na krokowym wykonywaniu rozkazów, czekanie na wykonanie rozkazów oraz na spełnieniu warunków procesowych niezbędnych do przejścia do następnych kroków
S7-PDIAG – Configuring Process Diagnostics		Diagnostyka procesowa, dzięki której możliwe jest szybkie wykrycie i usunięcie awarii
S7-PLCSIM – Simulating Modules		Szybkie i łatwe przetestowanie programu sterowania bez kompletnego sprzętu w postaci fizycznego sterownika
Setting the PG-PC Interface		Ustawienia parametrów portu komunikacyjnego programator-sterownik

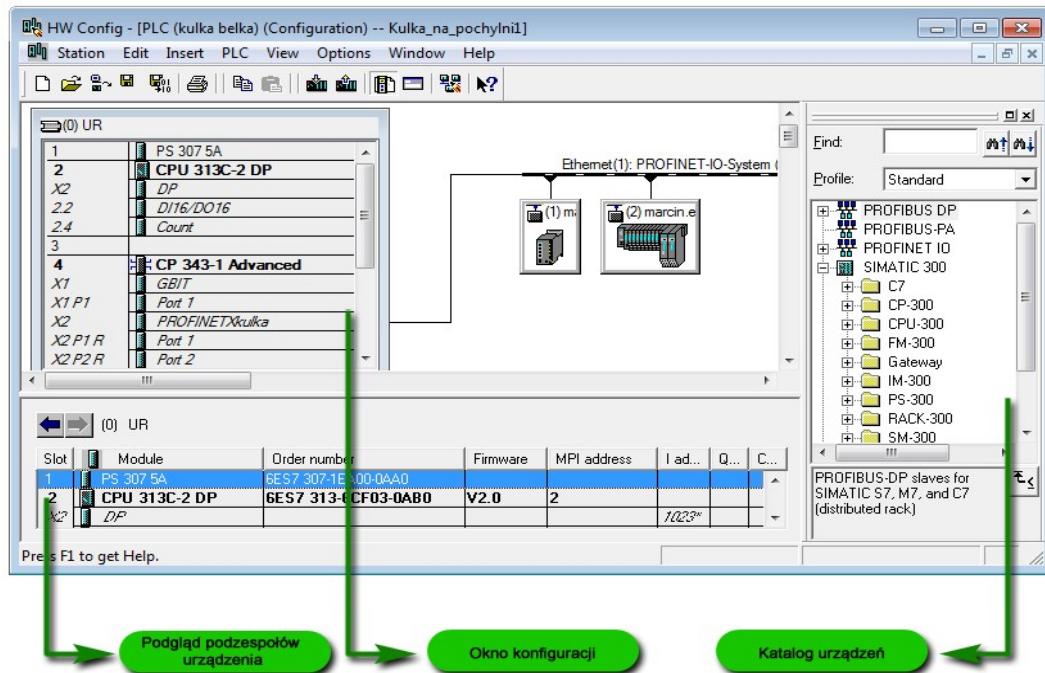
1.1. Oprogramowanie narzędziowe Simatic Manager

Każdy program jest zapisywany w postaci projektu. W projekcie poza częścią programową jest również zawarta deklaracja konfiguracji sprzętowej *HW Config*, która powinna zadany program realizować.



Rys. 1.1. Okno projektu Simatic Manager.

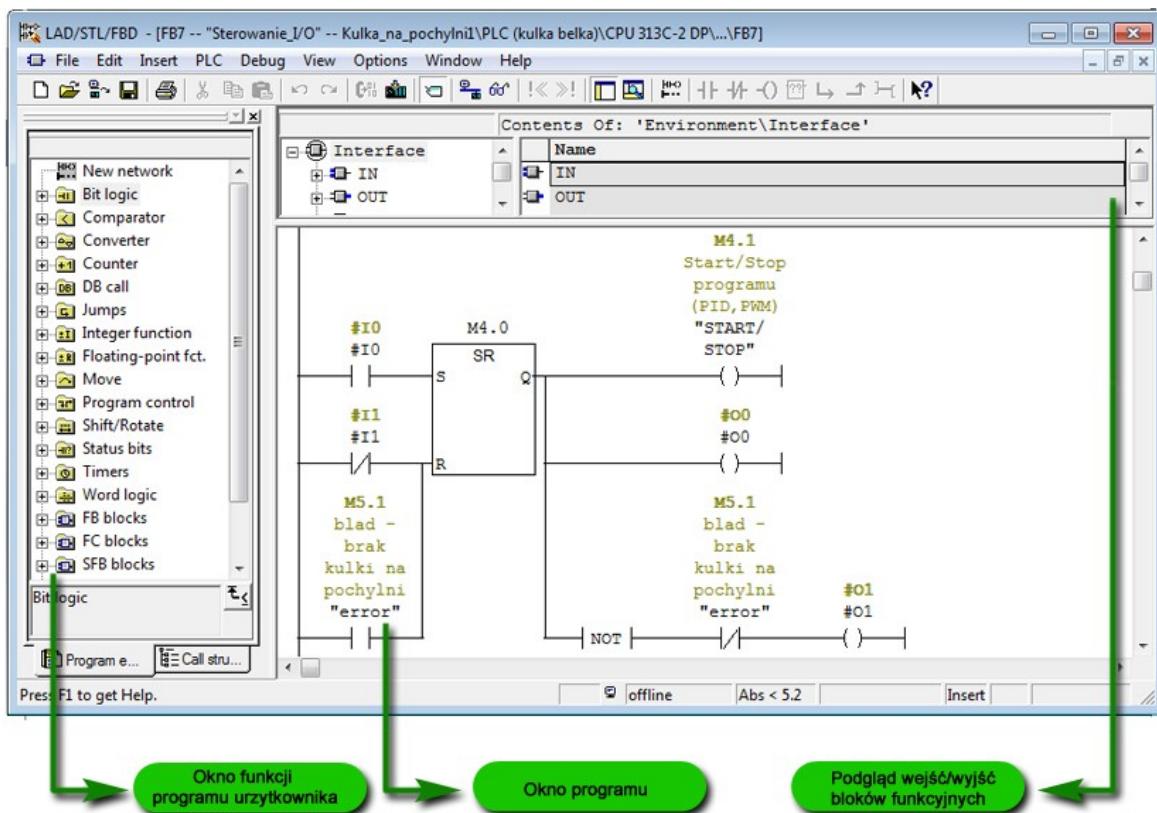
Aplikacja HW Config (Rys. 1.2.) podzielona jest na 3 części. Okno konfiguracji przedstawiające aktualną konfigurację projektu, katalog urządzeń możliwych do dodania w projekcie oraz podgląd podzespołów poszczególnych urządzeń. Każde urządzenie posiada indywidualny numer katalogowy i musi zachodzić zgodność numerów zarówno konfiguracji sprzętowej w programie *HW Config* jak i w fizycznej konfiguracji urządzeń.



Rys. 1.2. Program HW Config.

Edytor programu użytkownika *LAD/STL/FBD* służy do wprowadzania, modyfikowania i podglądu *online* programu sterownika PLC. Algorytm sterowania może zostać zapisany w kilku rodzajach językach programowania takich jak [4]:

- *Statement List* – STL – lista instrukcji,
- *Ladder Diagram* – LAD – diagram (schemat) drabinkowy,
- *Function Block Diagram* – FBD – diagram (schemat) funkcji.



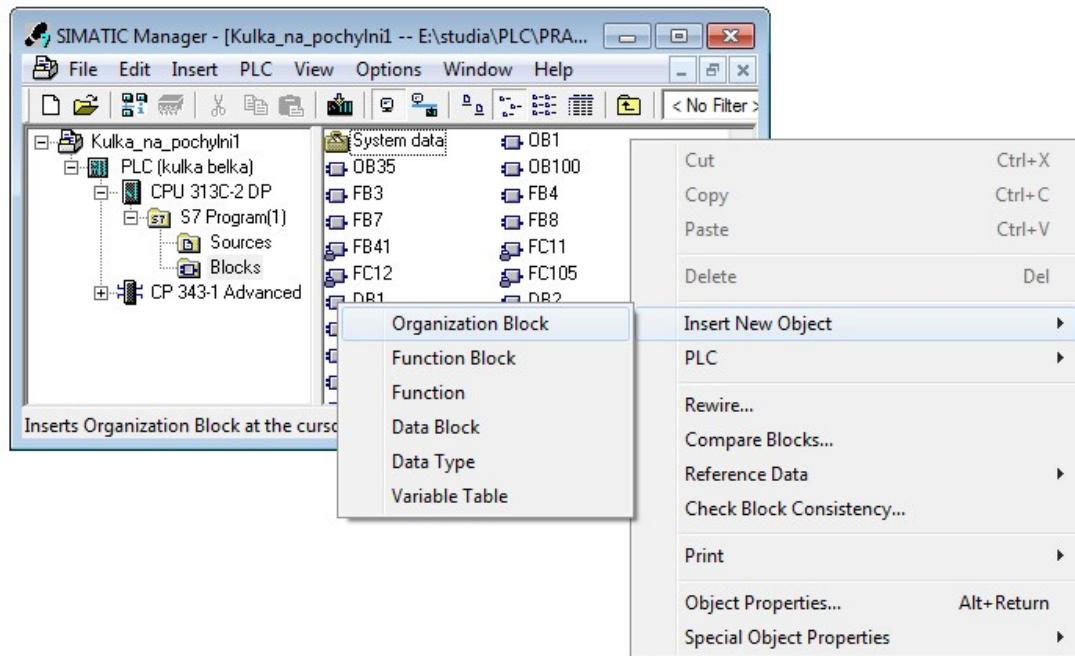
Rys. 1.3. Edytor programu użytkownika *LAD/STL/FBD*.

Okno funkcji programu zawiera podstawowe możliwe do wykorzystania w programie funkcje spełniające określone zadania (np. styki, cewki, komparatory, operacje matematyczne itp.). Okno programu wyświetla aktualnie stworzony program sterowania z możliwością podglądu znaczników pamięci programu, symboli i komentarzy. Podgląd wejść/wyjść bloków funkcyjnych stwarza wykaz wejść/wyjść użytych w blokach funkcyjnych parametryzowanych.

1.2. Operacje na blokach w projekcie

Do operacji na blokach w projekcie zaliczane jest wstawianie, zapisywanie, wywoływanie i modyfikowanie bloków. Wstawianie bloków S7 odbywa się przy użyciu opcji z paska menu: *Insert → S7 Blocks*. Opcja ta będzie aktywna tylko w momencie zaznaczenia w drzewku hierarchicznym gałęzi *Blocks* (Rys. 1.4). Po wykonaniu tych czynności zostanie wyświetlona lista bloków [4]. Są to:

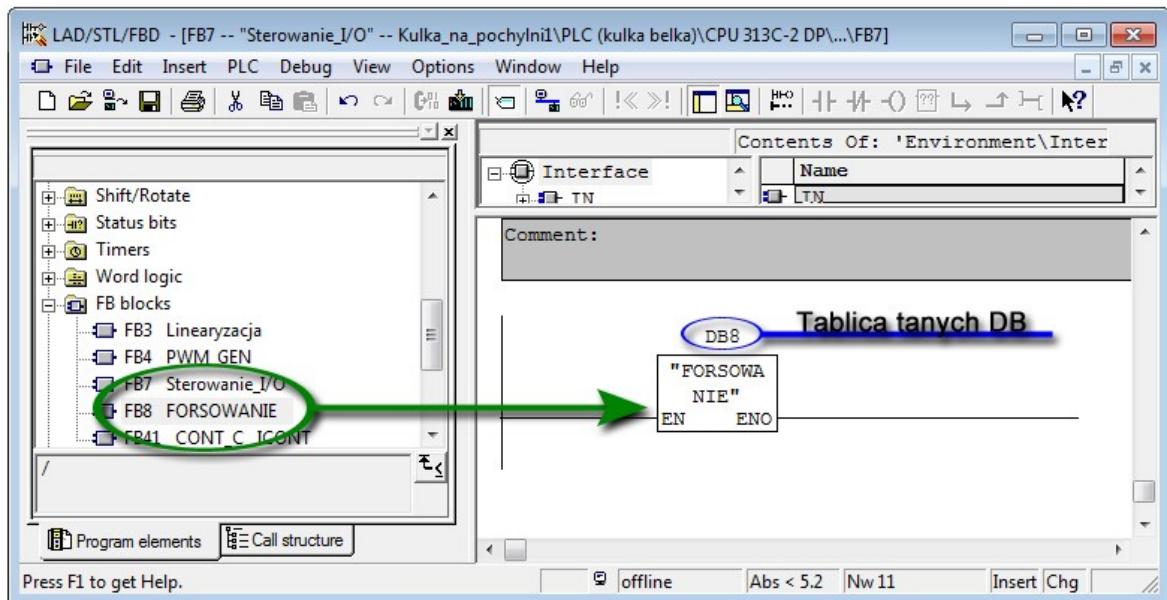
- *Organization Block* – blok OB wywoływany przez system operacyjny. Jest on formą komunikacji pomiędzy systemem operacyjnym a programem użytkownika składającego się z funkcji i bloków funkcji,
- *Function Block* – blok funkcyjny FB zawierający rzeczywisty program użytkownika. Umożliwia on złożenie podzielonego programu na małe, łatwiejsze do śledzenia części oraz może korzystać z bloków danych,
- *Function* – funkcja FC zawierająca rzeczywisty program użytkownika. Umożliwia on złożenie podzielonego programu na małe, łatwiejsze do śledzenia części,
- *Data Block* – Blok danych DB zawierający dane użytkownika,
- *Data Type* – UDT; typy danych,
- *Variable Table* – tablica zmiennych VAT. Umożliwia podgląd i modyfikację zmiennych w trybie *online*.



Rys. 1.4. Wstawianie nowego bloku do zawartości projektu.

Po wybraniu bloku wyświetcone zostanie okno edycji, w którym wybrać można numer bloku, symbol bloku oraz komentarz opisujący blok. Po zakończonej edycji bloku należy go zapisać na dysku twardym programatora (PC) przy użyciu opcji *Save*.

Aby blok był wywoływany, należy przeciągnąć go z okna funkcji programu użytkownika do segmentu w bloku OB. Następnie w przypadku bloków funkcyjnych FB należy przypisać indywidualną tablice danych (Rys. 1.5.)

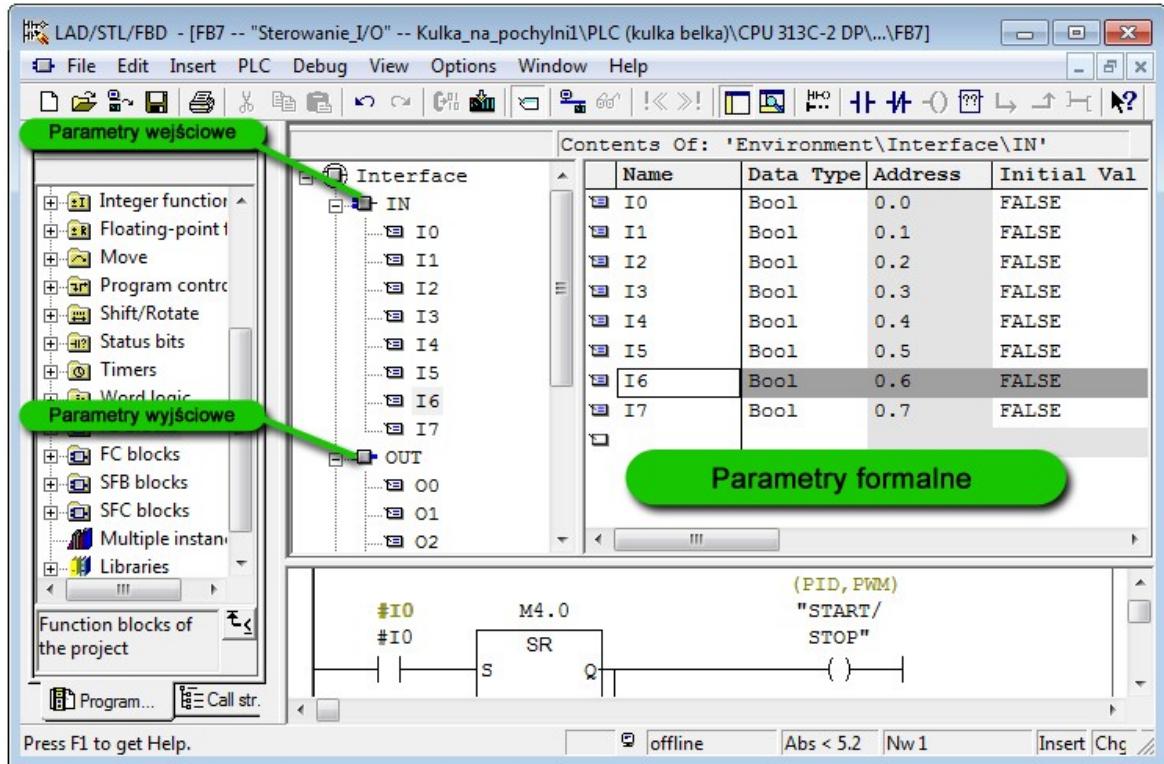


Rys. 1.5. Wywoływanie bloku FB w bloku OB.

W celu przejrzystości algorytmu sterowania dzieli się program na mniejsze i umieszcza w blokach funkcyjnych FB. Bloki funkcyjne można parametryzować nadając mu trzy parametry, parametr wejściowy (IN – tylko odczyt), parametr wyjściowy (OUT – tylko zapis) oraz parametr przejściowy (IN_OUT – odczyt/zapis). W efekcie parametryzacji bloku otrzymujemy zaprogramowany blok funkcyjny. Wykorzystanie takich bloków ma wiele zalet. Podstawową jest to, że program bloku jest tworzony tylko raz, dzięki czemu oszczędza się miejsce w pamięci. Ponadto blok ten może być wywoływany dowolną liczbę razy w dowolnym miejscu programu z różnymi zestawami parametrów aktualnych, których wartości są podstawiane w miejsce parametrów formalnych w instrukcji wywoływania bloku (Rys. 1.7). Tworzenie bloku parametryzowanego przebiega w trzech krokach [6]:

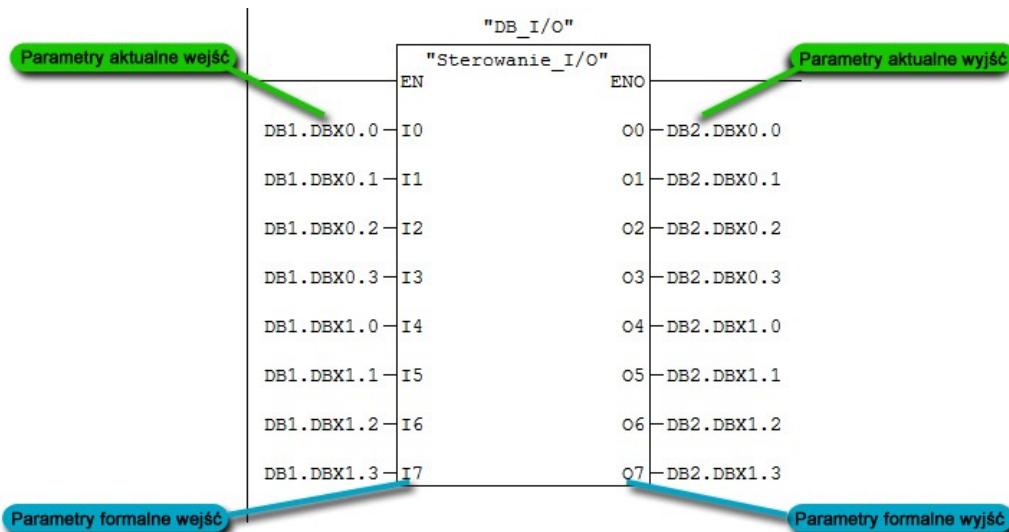
1. Utworzenie nowego bloku funkcyjnego FB w folderze *Block* (Rys. 1.4),
2. Wywołanie bloku w programie i przypisanie tablicy danych DB (Rys. 1.5),
3. Zdefiniowanie parametrów formalnych w tablicy deklaracji bloku (Rys. 1.6).

Kolejność wykonywanych kroków tworzenia bloku parametryzowanego musi być zachowana. W przypadku wcześniejszego zdefiniowania parametrów formalnych nie będzie możliwe utworzenie niezbędnej tablicy danych.



Rys. 1.6. Definiowanie parametrów formalnych bloku parametryzowanego FB.

Efekt końcowy tworzenia bloku parametryzowanego przedstawiono na poniższym rysunku:



Rys. 1.7. Przykładowy blok parametryzowany.

Po utworzeniu bloku można przystąpić do jego programowania. Wewnątrz bloku użyte parametry formalne będą odpowiadać parametrom aktualnym na zewnątrz bloku.

1.3 Typy danych w programie Step 7

Typy danych charakteryzują własności danych, tzn. sposób ich reprezentacji, dopuszczalny zakres wartości oraz metody przetwarzania (rodzaje instrukcji). Każdy typ danych charakteryzuje się wielkością wymaganego obszaru pamięci, np. typ WORD zajmuje w pamięci użytkownika długość 16 bitów. Elementarne typy danych mają długość nie większą niż 32 bity, dlatego mogą być załadowane do akumulatorów procesora w całości i przetwarzane przez instrukcje języka Step 7. Dane typu BOOL mają długość 1 bitu. Dane typu BYTE, WORD, DWORD mają długość odpowiednio 8, 16 i 32 bitów. Specyficznymi formatami tych typów danych są liczby w kodzie BCD, wartości liczników oraz znaki ASCII typu CHAR [4].

Dane typu INT, DINT, REAL zawierają liczby, które można przetwarzać w operacjach matematycznych:

- **INT (INTeger – liczba całkowita)** – identyfikuje liczbę całkowitą o długości 16 bitów. Wartość 16-bitowej liczby całkowitej mieści się w zakresie -32768...+32768. Wartość najstarszego bitu (lewy 15 bit), określa znak liczby (0 – liczba dodatnia, 1 – liczba ujemna).
- **DINT (Double INTeger – podwójna liczba całkowita)** - identyfikuje liczbę całkowitą o długości 32 bitów, gdzie najstarszy bit określa znak. Zakres liczby typu DINT wynosi L# -2 147 483 648...L# + 2 147 483 648.
- **REAL (liczba rzeczywista)** - określa liczbę rzeczywistą. Zakres liczb rzeczywistych wynosi $-1,17549510^{38} \dots 3,40282310^{38}$. W programie liczbę wykładniczą $1,1*10^9$ zapisuje się w postaci 1,1e9, gdzie „e” oznacza podstawę potęgi równą 10. Liczba rzeczywista zajmuje w pamięci dwa słowa (32 bity). Najstarszy bit określa znak liczby rzeczywistej, pozostałe bity reprezentują część ułamkową i wykładnik potęgi o podstawie 2.

Dana typu TIME ma długość podwójnego słowa i zawiera wartość interwału czasowego, interpretowanego jako liczba całkowita (dodatnia lub ujemna) długości (DINT), w milisekundach.

1.4 Adresowanie pamięci systemowej sterownika PLC

Pamięć systemowa jest zorganizowana w obszary adresowe. Poprzez odpowiednie instrukcje w programie możemy zaadresować dane bezpośrednio z odpowiedniego obszaru. Ze względu na obszerność adresowania pamięci systemowej w programie *Step 7*, wyjaśnione zostały tylko sposoby adresowania dla potrzeb programu układu sterowania :

- **Obszar znaczników M** – typu M (1 bit), MB (8 bitów), MW (16 bitów), MD (32 bity),
- **Bloki danych DB** – typu DBX (1 bit), DBB (8 bitów), DBW (16 bitów), DBD (32 bity),

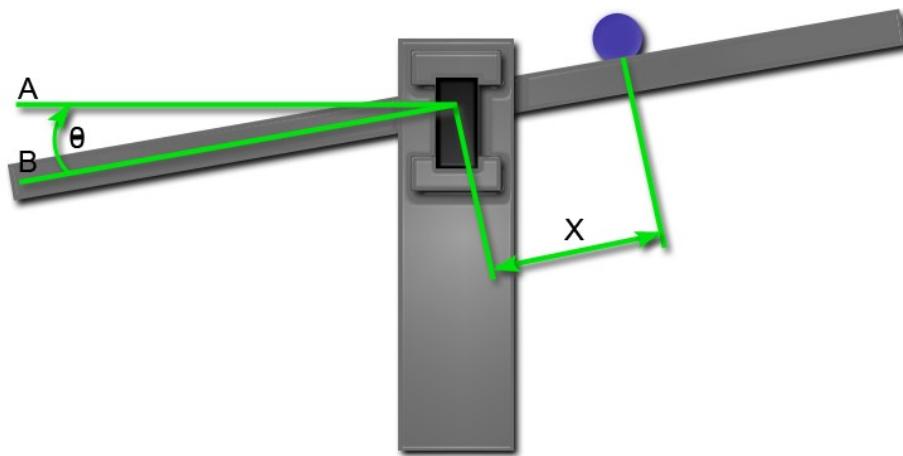
Tabela 1.2. Przykładowe adresowanie pamięci systemowej sterownika PLC.

Typ	Przykład	Liczba bitów (przykład)
M - BOOL	M 6.3 = 1	1
MB - BYTE	MB 6 (M6.0..M6.7)	0000 1000
MW - WORD	MW 5 (MW5..MW6)	0000 1000 0000 0000
MD - DWORD	MD 4 (MW4..MW6)	0000 0000 0000 1000 0000 0000 0000 0000
DBX - BOOL	DB1.DBX 6.3 = 1	1
DBB - BYTE	DB1.DBB 6	0000 1000
DBW - WORD	DB1.DBW 5	0000 1000 0000 0000
DBD - DWORD	DB1. DBD4	0000 0000 0000 1000 0000 0000 0000 0000

Analizując powyższą tabelę zauważyc można że obszar o większej pamięci zawiera obszary o mniejszej pamięci. Przykładowo znacznik MD20 (32 bity) zawiera w sobie znaczniki MW20 (16 bitów) oraz MW 22 (16 bitów), z kolei MW20 zawiera w sobie MB20 (8 bitów), MB21 (8bitów), a MW22 – MB22 (8 bitów), MB23 (8 bitów), obszar pamięci MB składa się z 8 bitów np. MB22 składa się ze znaczników o rozmiarze jednego bitu M22.0, M22.1 M22.7. Analogicznie należy adresować bloki danych DB [4].

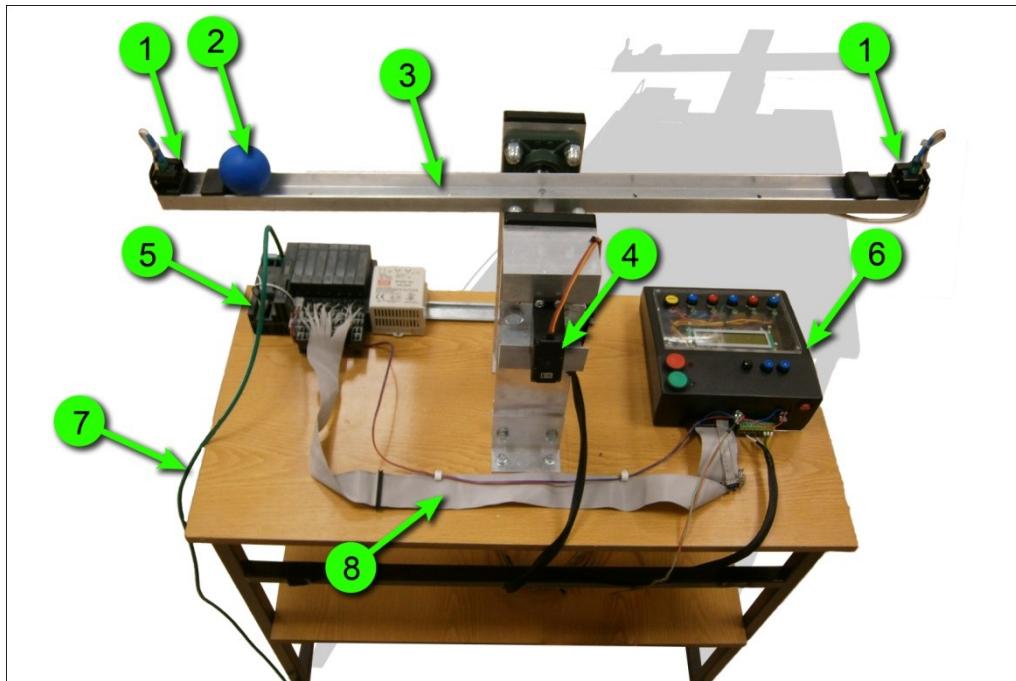
2. KONCEPCJA UKŁADU STEROWANIA PLC

Układ sterowania ma za zadanie doprowadzić do stabilności kulki w jednym wyznaczonym miejscu znajdującym się w dopuszczalnym zakresie na pochylni (Rys. 2.1). Zadanie układu sterowania można sformułować w inny, bardziej precyzyjny sposób: należy regulować pochylnią w taki sposób, aby wartość kąta Θ pomiędzy osią A i prostą B oraz odległość (X) między punktem styku kulki z pochylnią oraz pożdanym punktem stabilności kulki dążyły do zera.



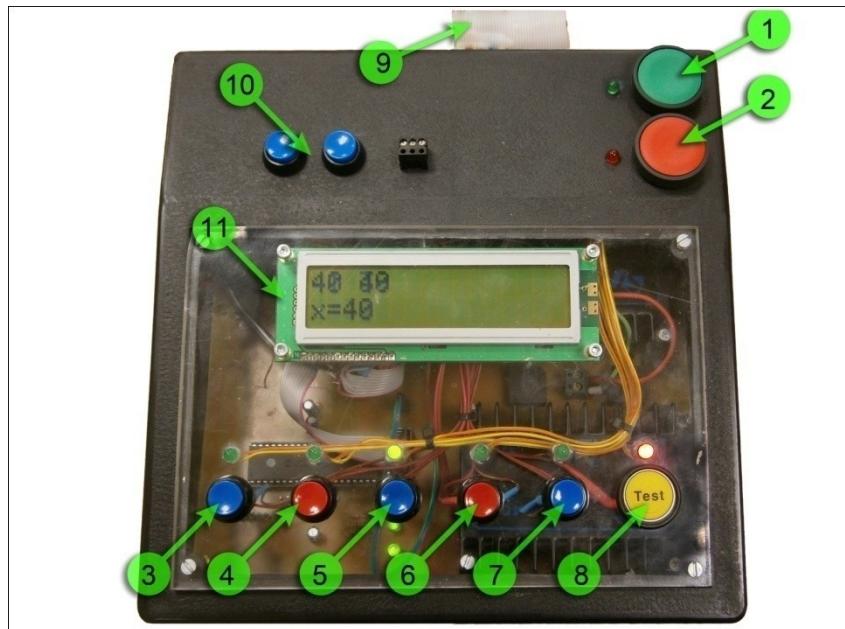
Rys. 2.1. Schemat procesu stabilizacji kulki na pochylni.

Poniższe rysunki przedstawiają budowę stanowiska do stabilizacji kulki na pochylni. Na rysunku 2.2 umieszczono układ kulka-pochylnia zawierający poszczególne oznaczenia podzespołów i elementów układu sterowania. Rysunek 2.3 przedstawia budowę zewnętrzną konsoli sterującej z oznaczeniami programowymi diod i przycisków, wyświetlacza pokazującego aktualną pozycję kulki na pochylni oraz przycisków mikrokontrolera sterującego wyświetlaczem.



Rys. 2.2. Stanowisko - kula na pochylni.

Oznaczenia do rysunku 2.2: 1 – czujniki optyczny, 2 – kulka, 3 – pochylnia, 4 – serwomechanizm, 5 – wyspa ET200S, 6 – Konsola sterująca, 7 – przewód do komunikacji w sieci PROFINET, 8 – taśma przewodowa łącząca wyspę ET200S z konsolą sterującą.

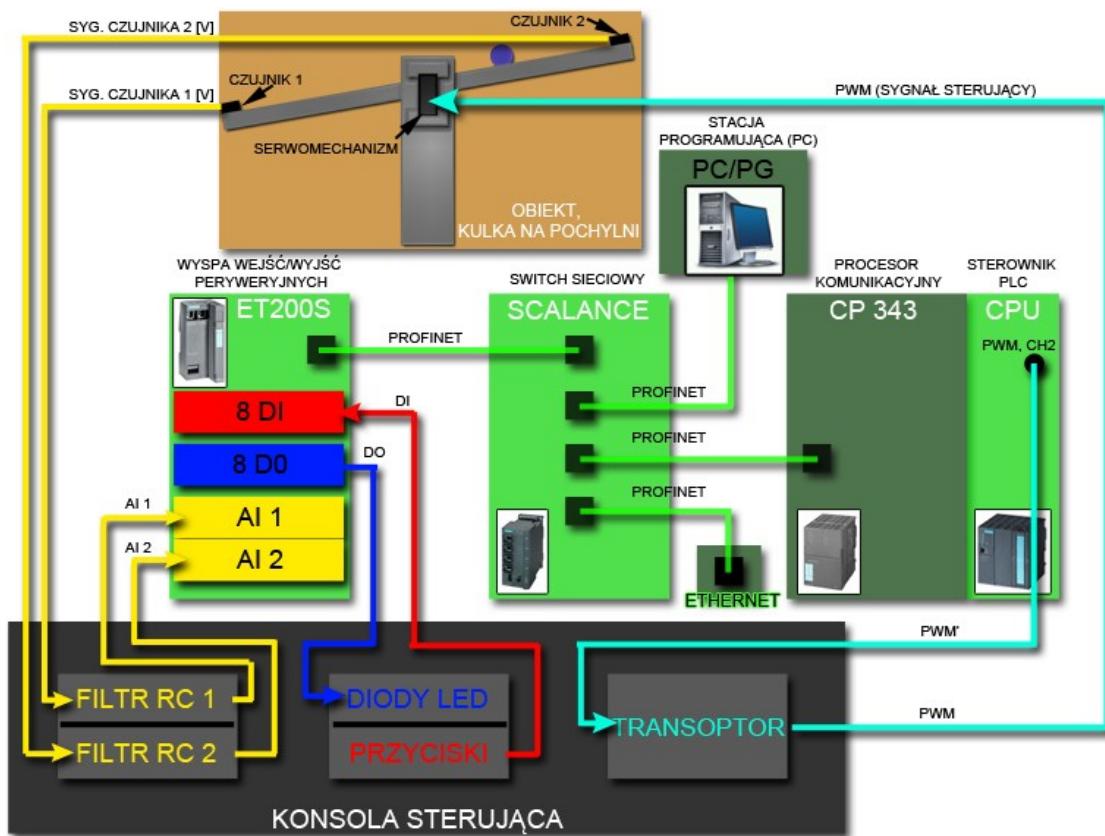


Rys. 2.3. Konsola sterująca.

Oznaczenia do rysunku 2.3 : 1 – przycisk I0 / dioda O0, 2 – przycisk I1 / dioda O1, 3 – przycisk I2 / dioda O2, 4 – przycisk I3 / dioda O3, 5 – przycisk I4 / dioda O4, 6 – przycisk I5 / dioda O5, 7 – przycisk I6 / dioda O6, 8 – przycisk I7 / dioda O7, 9 – taśma przewodowa łącząca wyspę ET200S z konsolą sterującą., 10 – przyciski mikrokontrolera, 11 – wyświetlacz mikrokontrolera.

2.1. Budowa układu sterowania

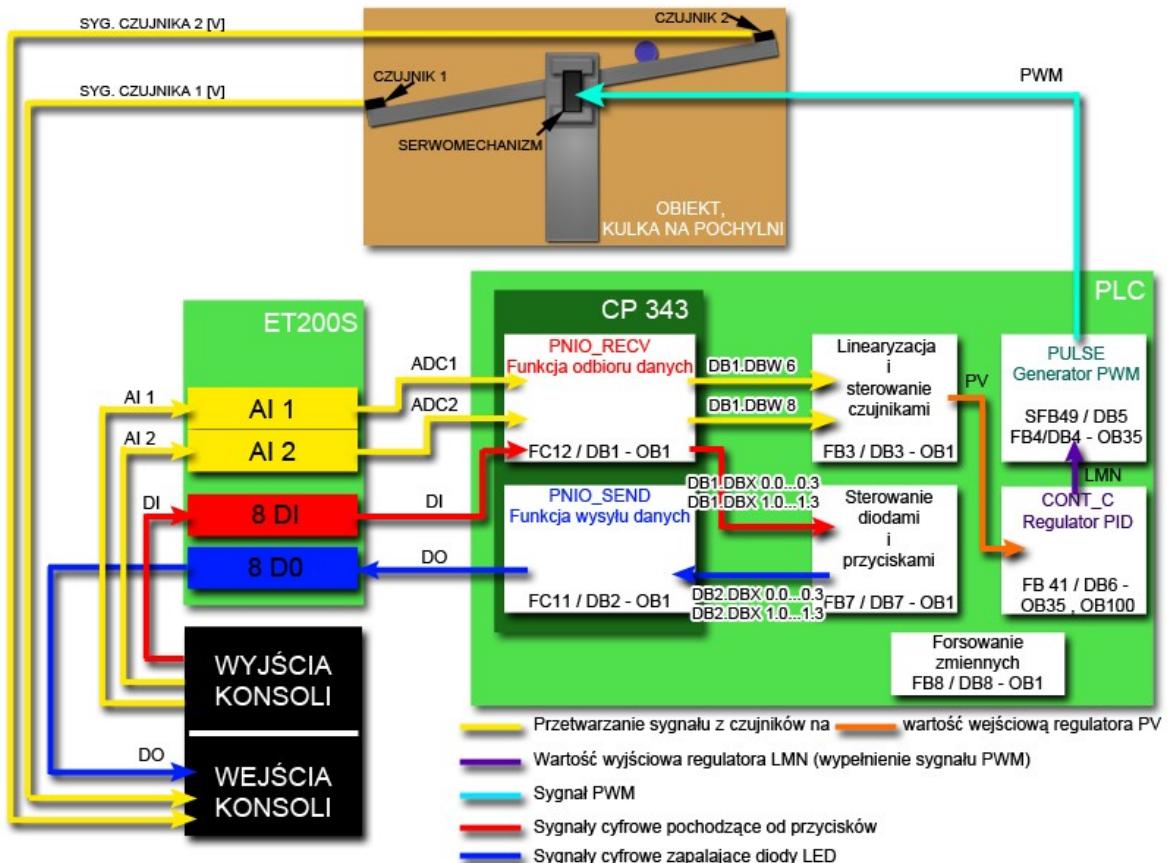
Budowę układu sterowania przedstawiono zarówno z perspektywy konstrukcyjnej (Rys. 2.4) jak i z perspektywy programowej (Rys. 2.5). Perspektywa konstrukcyjna przedstawia układ sterowania uwzględniając urządzenia, elementy, sygnały pomiarowe, sygnał sterujący oraz połączenia w sieci PROFINET. Perspektywa programowa przedstawia układ sterowania uwzględniając systemowe i parametryzowane bloki funkcyjne programu sterowania oraz sposób przetwarzania sygnałów dla potrzeb programu.



Rys. 2.4. Schemat ukłdu sterowania z perspektywy konstrukcyjnej.

Z perspektywy konstrukcyjnej układ sterowania można przedstawić następująco: sygnały w postaci napięciowej z czujników cyfrowych, zanim zostaną przetworzone przez przetworniki analogowo cyfrowe (moduły AI wyspy ET200S) są filtrowane (RC) w konsoli sterującej, w celu wyeliminowania występujących skoków napięcia sygnału wyjściowego czujnika. Wyjścia cyfrowe (moduły DO wyspy ET200S) podłączone są do diod LED sygnalizujących o stanie obiektu, natomiast wejścia cyfrowe (moduły DI wyspy ET200S) odbierają sygnały z przycisków. Wymiana danych sygnałowych między wyspą wejśc/wyjśc peryferyjnych ET200S a sterownikiem PLC odbywa się poprzez sieć

PROFINET za pośrednictwem switcha sieciowego SCALANCE a następnie poprzez procesor komunikacyjny CP połączony bezpośrednio ze sterownikiem PLC. Ponadto do switcha sieciowego podłączona jest stacja programująca PC/PG oraz lokalna sieć Ethernet. Ze sterownika cyfrowego PLC generowany sygnał PWM osiąga napięcie 24 V, natomiast serwomechanizm przyjmuje sygnał sterujący o napięciu ok. 5V. Przełożenie napięcia z 24V na 5V zapewnia transoptor na którego wejście podany jest sygnał PWM ze sterownika (24V), a na wyjściu znajduje się sygnał sterujący serwomechanizmem (5V).



Rys. 2.5. Schemat układu sterowania z perspektywy programowej.

Z perspektywy programowej układ sterowania można przedstawić następująco: wejściowe sygnały analogowe AI (od czujników) zostają przetworzone na postać cyfrową ADC a następnie zapisane w pamięci tablicy DB1. Przetworzone sygnały analogowe z czujników zostają poddane obróbce w bloku *FB3 – Linearyzacja i sterowanie czujnikami* w taki sposób aby na wyjściu tego bloku otrzymać realną odległość kulki od początku pochylni. Odległość ta oznaczona jako PV (Rys. 2.5) jest wartością wejściową regulatora PID umieszczonego w bloku *FB41 – CONT_C*, który na wyjściu podaje wartość wyjściową regulatora LMN dla generatora sygnału PWM umieszczonego w bloku *SFB49 – PULSE*. Generator sygnału PWM na podstawie wartości wyjściowej regulatora

PID, generuje sygnał sterujący serwomechanizmem. Odebrane sygnały cyfrowe z przycisków (DI) zapisywane w pamięci tablicy DB1 przekazywane są do bloku *FB7 – Sterowanie przyciskami i diodami*. W bloku FB7 odbywa się sterowanie przyciskami i diodami LED dla potrzeb układu sterowania. Z wyjścia bloku FB7 wychodzą sygnały sterujące diodami LED, zapisywane w pamięci bloku DB2 a następnie wysyłane do modułu wyjść cyfrowych DO wyspy ET200S skąd wychodzą sygnały napięciowe zapalające diody. Za odbiór danych z modułów wejściowych wyspy ET200S odpowiedzialna jest funkcja *FC12 – PNIO_RECV*, z kolei za wysyłanie danych do modułów wyjściowych wyspy ET200S służy funkcja *FC11 – PNIO_RECV*. W bloku FB8 – *Forsowanie zmiennych* odbywa się nadpisywanie odpowiednich parametrów programu dla potrzeb układu sterowania.

2.2. Charakterystyka programowa elementów układu sterowania

Układ sterowania składa się z następujących elementów:

- **Układ przetwarzania sygnałów** – sterownik PLC, procesor komunikacyjny CP, switch sieciowy Ethernet, wyspa wejść/wyjść periferyjnych ET200S,
- **Elementów pomiarowych** – czujnik optyczny Sharp GP2Y0A02,
- **Elementu wykonawczego** – serwomechanizm Tower Pro.

2.2.1. Charakterystyka programowa układu przetwarzania sygnałów

Układ przetwarzania sygnałów składa się z następujących elementów:

- zasilacza PS 24V,
- jednostki centralnej CPU,
- procesora komunikacyjnego CP,
- switch'a sieciowego Ethernet,
- wyspy wejść/wyjść ET200S.



Rys. 2.6. Stacja PLC.

Oznaczenia do rysunku 2.6 : 1 – zasilacz przemysłowy PS307, 2a – sterownik PLC CPU313C -2DP, 2b – zintegrowany moduł we/wy cyfrowych sterownika, 3 procesor komunikacyjny CP 343-1 Advanced, 4 – switch sieciowy SCALANCE X204 IRT, 5 – przewody do komunikacji w sieci PROFINET.

Zasilacz PS 307 5A 24VDC (Rys. 2.6 ozn. 1) marki Siemens zasila sterownik PLC, procesor komunikacyjny CP oraz switch sieciowy Ethernet [7].

Jako jednostkę centralną układu sterowania wykorzystano kompaktowy sterownik PLC 313C-2DP (Rys. 2.6 ozn. 2a,b) marki Siemens ze zintegrowanymi wejściami/wyjściami cyfrowymi oraz interfejsem PROFIBUS DP. Do najważniejszych cech sterownika PLC wykorzystanych w projekcie należy posiadanie 3 szybkich liczników (30KHz) do generowania sygnału PWM sterującego serwomechanizmem oraz możliwości regulacji PID z wykorzystaniem wbudowanych funkcji. Poza tym sterownik PLC wykonuje prace obliczeniowe związane z regulacją PID, linearyzacją charakterystyki odległościowej czujnika oraz prace związane z wymianą danych poprzez procesor komunikacyjny z wyspą peryferyjnych wejść/wyjść ET200S. Dokładny opis wykonywanych prac związanych z wymianą danych, linearyzacją charakterystyki odległościowej czujnika oraz generowania sygnału PWM umieszczono w rozdziale 4, natomiast funkcje do obsługi regulatora PID w rozdziale 5[7].

Procesor komunikacyjny CP343-1 Advanced (Rys. 2.6 ozn. 3) zapewnia komunikację sterowników serii S7-300 w sieci PROFINET pomiędzy programatorami, urządzeniami wizualizacyjnymi, wyspami wejść/wyjść oraz innymi sterownikami rodzin S7. Umożliwia także wysyłanie maili, diagnostykę poprzez http oraz tworzenie stron

internetowych. Obsługuje funkcje internetowe, takie jak klient FTP i Java. W projekcie układu sterowania zapewnia wymianę danych między sterownikiem PLC a wyspą ET200S.

Switch sieciowy SCALANCE X204 IRT (Rys. 2.6 ozn. 4) wyposażony w cztery porty Ethernet (RJ45) przeznaczony jest do tworzenia sieci opartej na pracy izochronicznej w trybie rzeczywistym w topologii linii, gwiazdy i pierścienia z prędkością 10/100 Mb/s.

Stacja ET200S (Rys. 2.7) [8] pod wpływem rozkazów otrzymanych ze sterownika PLC odpowiednio steruje wyjściami cyfrowymi i analogowymi, natomiast sygnały otrzymane z modułów wejściowych przekazuje z powrotem do jednostki centralnej. Dodatkowo stacja ET 200S ma wbudowany switch sieciowy wyposażony w dwa gniazda Ethernet (RJ45) do komunikacji w sieci PROFINET. Wyspa ET200S składa się z następujących urządzeń :

- **wyspy peryferyjnych wejść/wyjść ET 200S IM 151-3 PN STANDARD** – (Rys. 2.7 ozn. 2) odpowiedzialna jest za przetwarzanie sygnałów z modułów wejściowych na sygnały możliwe do wysyłania w sieci PROFINET oraz za przetwarzanie danych z sieci PROFINET na sygnały do modułów wyjściowych,
- **modułu zasilającego PM-E DC24V** (Rys. 2.7 ozn. 3) – dostarcza zasilanie do modułów wejść/wyjść,
- **dwóch modułów wejść cyfrowych 4DI DC24V ST** (Rys. 2.7 ozn. 4a,b) – do modułów wejść cyfrowych zostało podłączone 7 przycisków NO (normalnie otwartych) i 1 przycisk NZ (normalnie zamknięty),
- **dwóch modułów wyjść cyfrowych 4DO DC24V/0.5A ST** (Rys. 2.7 ozn. 5a,b) – do modułów wyjść cyfrowych zostało podłączone 8 diod sygnalizujących o stanie obiektu,
- **modułu wejść analogowych 2AI U** (Rys. 2.7 ozn. 6) – do modułów wejść analogowych zostały podłączone dwa czujniki optyczne Sharp,
- **modułu wyjść analogowych 2AO U** (Rys. 2.7 ozn. 7) – wyjścia analogowe zostały wyprowadzone opcjonalnie np. do odczytu żądanych sygnałów przez oscyloskop lub inne urządzenie przetwarzające sygnały.



Rys. 2.7. Wyspa wejść wyjść perforejnych ET200S.

Oznaczenia do rysunku 2.7 : 1 – przewód do komunikacji w sieci PROFINET, 2 – wyspa ET200S, 3 – moduł PM-E przekazujący napięcie do modułów we/wy, 4a/4b – moduły DI wejść cyfrowych , 5a/5b – moduły DO wyjść cyfrowych, 6 – moduł AI wejść analogowych, 7 – moduł AO wyjść analogowych, 8 - taśma przewodowa łącząca wyspę ET200S z konsolą sterującą.

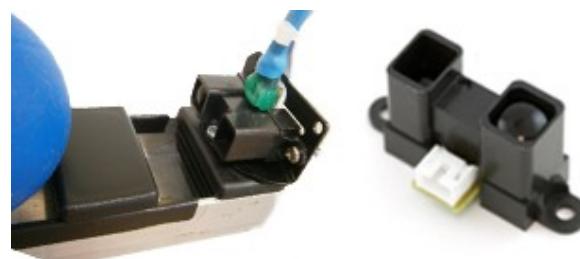
Na poniższej tabeli umieszczono spis urządzeń stacji PLC oraz wyspy ET200S wraz z oznaczeniami na poszczególnych rysunkach, opisami oraz numerami katalogowymi, niezbędnymi do prawidłowej konfiguracji sprzętowej w oprogramowaniu SIMATIC.

Tabela 3. Opis urządzeń stacji PLC oraz wyspy ET200S.

Nazwa urządzenia	Numer katalogowy	Rysunek/ oznaczenie	Opis
PS 307 5A	307-1EA00-0AA0	R 2.6/1	Zasilacz 24 VDC
CPU 313C-2DP	313-6CF03-0AB0	R 2.6/2a,b	PLC S7-300
CP343-1 ADVANCED	343-1GX30-0XE0	R 2.6/3	Procesor komunikacyjny CP
SCALANCE X 204 IRT	204-0BA00-2BA3	R 2.6/4	Switch sieciowy Ethernet
ET 200S IM 151-3PN STANDARD	6ES7 151-3AA23-0AB0	R 2.7/2	Wyspa ET 200S
Moduły wyspy ET200S	PM-E DC 24V	6ES7 138-4CA01-0AA0	Zasilacz DC 24V
	4 DI DC 24V	6ES7 131-4BD01-0AA0	4 wejścia cyfrowe DC24V
	4 DO DC24V /0.5A	6ES7 132-4BD01-0AA0	4 wyjścia cyfrowe DC24V
	2 AI U	6ES7 134-4FB01-0AB0	2 wejścia analogowe napięciowe
	2 A0 U	6ES7 135-4FB01-0AB0	2 wejścia analogowe napięciowe

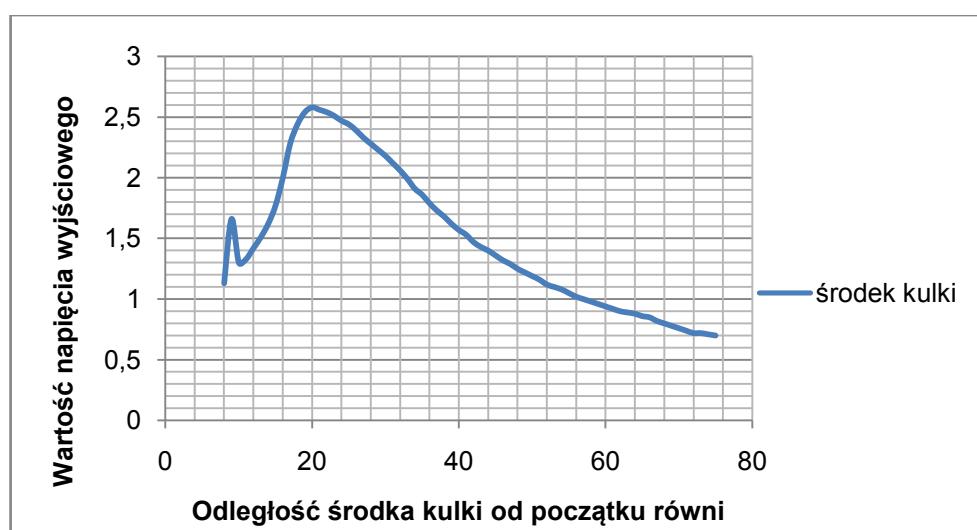
2.2.2. Charakterystyka programowa elementów pomiarowych i elementu wykonawczego

Elementami pomiarowymi w układzie kulki na pochylni są czujniki fotoelektryczne Sharp GP2Y0A02 (Rys. 2.8) o wyjściu analogowym i zakresie pomiarowym 20 cm – 150 cm. Czujniki wykorzystywane są do pomiaru odległości od przeszkody. Zintegrowana dioda czujnika pełni rolę nadajnika światła podczerwonego, która wysyła światło o odpowiedniej długości fali. Jeśli przeszkoda znajdzie się w linii prostej przed czujnikiem, odbita od niej wiązka światła podczerwonego dotrze do odbiornika czujnika. Odbierane światło przetwarzane jest przez fotorezystor, dzięki czemu odległość czujnika od przeszkody odczytywane jest poprzez napięcie. Wyjście czujnika może być podłączone bezpośrednio do przetwornika analogowo cyfrowego i nie potrzebuje żadnego zewnętrznego układu sterującego.



Rys. 2.8. Czujnik optyczny Sharp GP2Y0A02.

Wadą czujników optycznych Sharp jest nieliniowa charakterystyka stosunku napięcia wyjściowego do odległości przeszkody od czujnika. W celu wyznaczenia takiej charakterystyki zamontowano czujnik optyczny na brzegu pochylni i oddalając kulkę co 1cm rejestrowano wartość napięcia wyjściowego. Otrzymaną charakterystykę napięciowo odległościową przedstawiono na poniższym rysunku.



Rys. 2.9. Charakterystyka napięciowo odległościowa czujnika optycznego Sharp GP2Y0A02.

Rozwiążanie problemu nieliniowej charakterystyki czujnika przedstawiono w podrozdziale 4.2.

Jako element wykonawczy w układzie sterowania położeniem kulki na pochylni wykorzystano serwomechanizm Tower Pro MG9805.

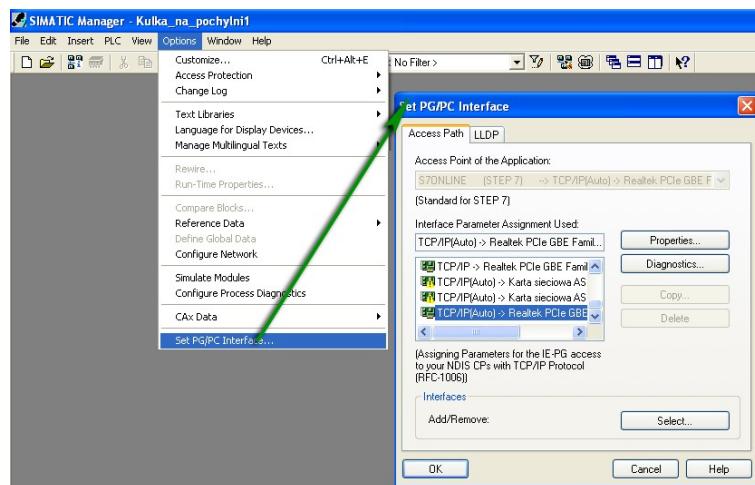


Rys. 2.10. Serwomechanizm Tower Pro MG9805.

Problem sterowania serwomechanizmem odnosi się do generowania sygnału PWM (*ang. Pulse With Modulation*). Sterowanie serwomechanizmem polega na modulacji szerokości wypełnienia sygnału prostokątnego o częstotliwości 50Hz (puls co 20 ms). Wychylenie serwomechanizmu zależy więc od wypełnienia pulsu sygnału. Dokładny opis sterowania serwomechanizmem zamieszczono w podrozdziale 4.3.

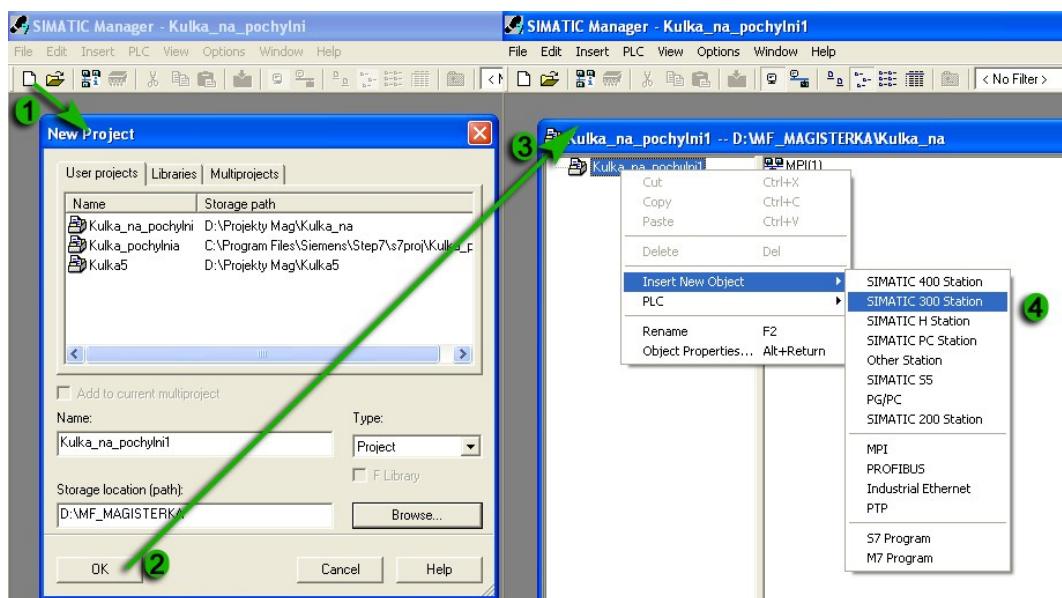
2.2.3. Konfiguracja i parametryzacja stacji PLC

Pierwszym możliwym krokiem konfiguracji stacji PLC jest wybranie sposobu komunikacji stacji programującej PC ze sterownikiem PLC (Rys 2.11). W tym przypadku portem komunikacyjnym jest karta sieciowa PC. We właściwościach (*Options / Set PG/PC Interface / Properties*) należy zdefiniować odpowiedni adres IP oraz maskę podsieci według istniejącej infrastruktury sieci [9].



Rys. 2.11. Wybór portu komunikacyjnego – *Options / Set PG/PC Interface*.

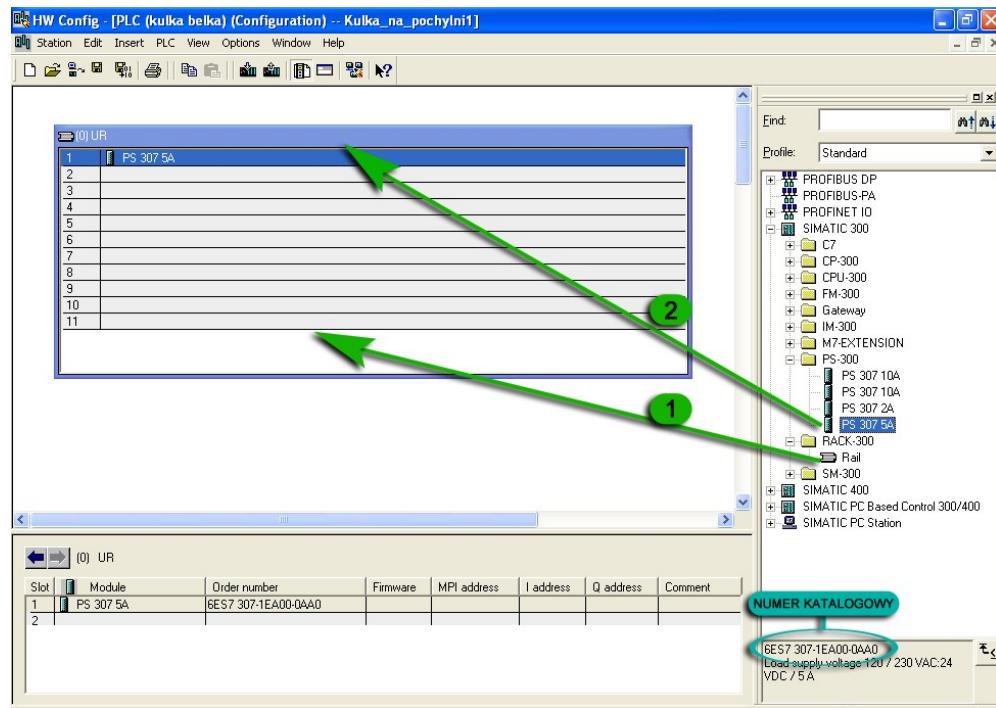
Kolejnym krokiem jest utworzenie nowego projektu w oknie *SIMATIC Manager/File/NewProject* (Rys. 2.12 ozn. 1). W oknie New Project należy określić nazwę projektu (*Name*) oraz jego lokalizację zapisu na dysku twardym (*Storage location*). Po wciśnięciu OK (Rys. 2.12 ozn. 2) pojawi się okno projektu (Rys. 2.12 ozn. 3). Pierwszą czynnością w oknie projektu jest dodanie stacji SIMATIC 300 poprzez kliknięcie prawym przyciskiem myszy na nazwie projektu i wybór opcji *Insert New Object / SIMATIC 300 Station* (Rys. 2.12 ozn. 4). Po wybraniu tej opcji można dowolnie nazwać stację.



Rys. 2.12. Okno nowego projektu - *File / New Project*.

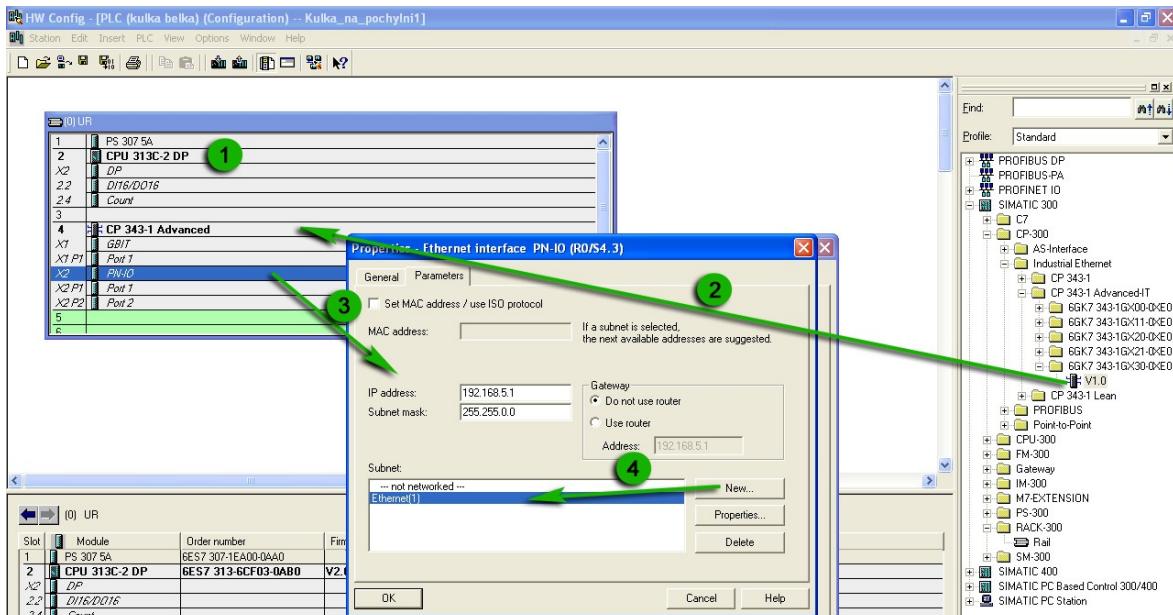
Następnie należy otworzyć aplikację konfiguracji sprzętowej poprzez wybranie ścieżki *Kulka_na_pochylni1/PLC(kulka_belka)/HW Config* (*Nazwa projektu/Nazwa stacji/HW Config*). W efekcie pojawi się okno konfiguracji sprzętowej.

W oknie HW Config (Rys. 2.13) pierwszą czynnością jest dodanie z katalogu urządzeń szyny montażowej (*Rail*) poprzez przeciągnięcie jej do okna aktualnej konfiguracji (Rys. 2.13 ozn. 1). Następnie dodanie pierwszego urządzenia które znajduje się w fizycznej konfiguracji stanowiska PLC, zazwyczaj jest to zasilacz PS (Rys. 2.13 ozn. 2). Przy dodawaniu urządzeń należy przestrzegać aby numer katalogowy (Rys. 2.14) oraz wersja dodawanego urządzenia był zgodny z numerem katalogowym fizycznego urządzenia. Nieprzestrzeganie tej zasady spowoduje wywołanie błędu sterownika PLC i uniemożliwi jego pracę.



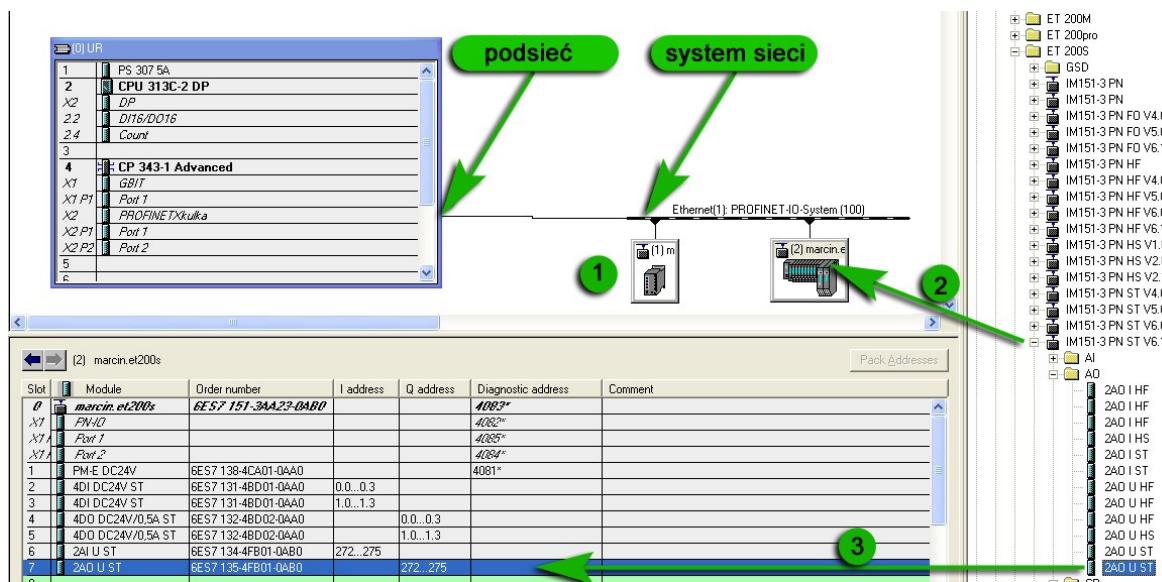
Rys. 2.14. Okno konfiguracji sprzętowej *HW Config* – dodawanie szyny montażowej.

Podobnie jak w przypadku zasilacza PS dodajemy do szyny montażowej sterownik CPU (Rys. 2.15 ozn. 1) oraz procesor komunikacyjny CP (Rys. 2.15 ozn. 2). Następnie we właściwościach (Rys. 2.15 ozn. 3) procesora komunikacyjnego CP (*CP 343-1 Advanced/PN-IO/Properties – Ethernet Interface*) należy ustawić odpowiedni adres IP oraz maskę podsieci, zgodną z ustawieniami portu komunikacyjnego (*Set PG/PC Interface*) oraz dodać podsieć PROFINET (Rys. 2.15 ozn. 4), ewentualnie zmienić nazwę podsieci (*Subnet*).



Rys. 2.15. Okno konfiguracji sprzętowej *HW Config* - dodawanie CPU, CP, tworzenie wejścia sieci PROFINET.

Z racji tej że zarówno switch sieciowy SCALANCE jak i wyspa ET200S nie są bezpośrednio podłączone do sterownika PLC należy utworzyć system sieci PROFINET w procesorze komunikacyjnym CP (*CP 343-1 Advanced/PN-IO/Insert PROFINET IO System*) a następnie dodać urządzenia podłączone w sieci nie do szyny montażowej *Rack* a do utworzonego systemu sieciowego PROFINET (Rys. 2.16. ozn. 1,2). Kolejnym krokiem jest dodawanie odpowiednich modułów stacji ET200S poprzez przeciąganie ich do okna podzespołów urządzenia (Rys. 2.16. ozn. 3).

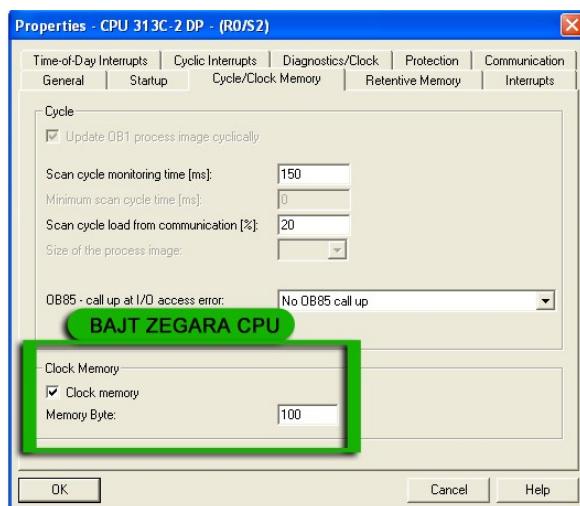


Rys. 2.16. Okno konfiguracji sprzętowej HW Config - dodawanie urządzeń podłączonych w sieci PROFINET

Po skonfigurowaniu stanowiska PLC należy sprawdzić poprawność poprzez komplikację i zapis (*HW Config/Station/Save and Compile*), jeśli komplikacja nie wykazała żadnych błędów należy wgrać konfigurację do sterownika poprzez wybranie opcji: *HW Config/PLC/Download*. Na procesorze komunikacyjnym CP pojawi się błąd sieci ponieważ należy zdefiniować i przypisać nazwy urządzeń podłączonych w sieci. Odbywa się to kolejno, poprzez wpisanie we właściwościach danego urządzenia indywidualnej nazwy (*Device name*), zaznaczeniu wszystkich urządzeń podłączonych do sieci PROFINET z przytrzymaniem klawisza *Shift*, wybraniu opcji: *HW Config/PLC/Ethernet/Assign Device Name*, następnie w oknie *Assign device name* z pola wyboru wybrać nazwę oraz z pola *Available devices* wybrać odpowiednie urządzenie i zatwierdzić przyciskiem *Assign name*. Czynność należy powtórzyć dla kolejnych urządzeń. Po przypisaniu nazw urządzeniom należy sprawdzić poprawność tej operacji poprzez wybranie opcji: *HW Config/PLC/Ethernet/Verify Device Name*, jeśli w kolumnie *Status* wszystkie urządzenia są oznaczone na zielono można ponownie wgrać

konfigurację do sterownika. Jeśli żadne urządzenie nie zwraca błędu, konfiguracja przebiegła prawidłowo, można zatem przystąpić do parametryzacji stacji PLC.

Jednym z punktów parametryzacji stacji PLC potrzebnych do zrealizowania projektu układu sterowania położeniem kulki na pochylni jest zdefiniowanie komórki pamięci dla zegara CPU (*HW Config/CPU Properties/Cycle/Clock Memory*). *Clock Memory* umożliwia sekwencyjne wywoływanie (aktywację) poszczególnego bitu, bajtu zegara CPU z różną częstotliwością (Rys. 25.). Częstotliwości poszczególnych bitów bajtu *Clock Memory* przedstawiono w tabeli 4 [6].

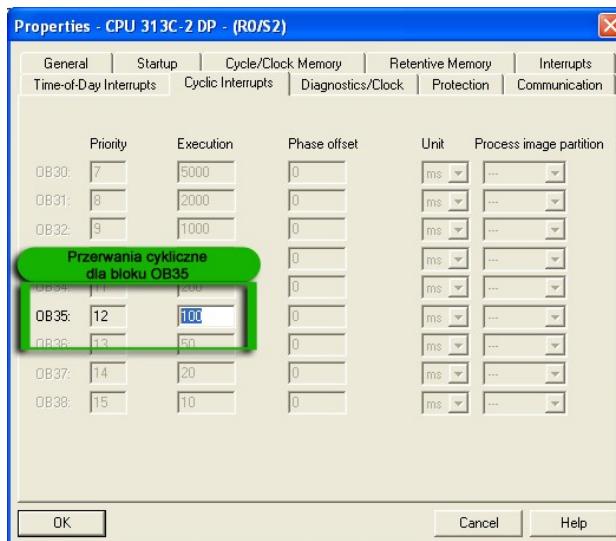


Rys. 2.17. Bajt zegara CPU - *HW Config/CPU Properties/Cycle/Clock Memory*.

Tabela 4. Działanie poszczególnych bitów *Clock Memory*.

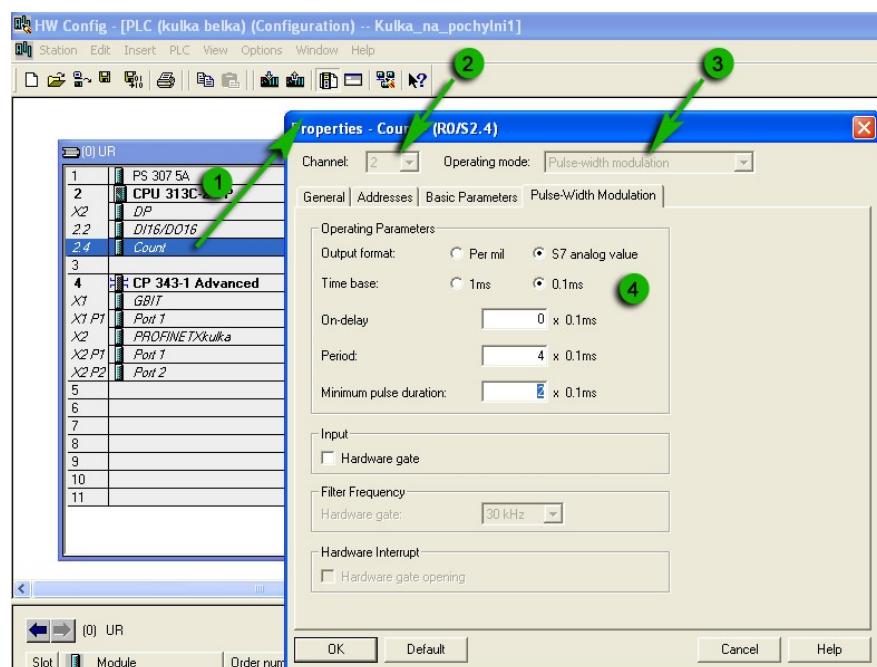
Bit bajtu zegara CPU	7	6	5	4	3	2	1	0
Okres [s]	2.0	1.6	1.0	0.8	0.5	0.4	0.2	0.1
Częstotliwość [Hz]	0.5	0.625	1	1.25	2	2.5	5	10

Kolejnym niezbędnym do zdefiniowania parametrem jest ustawienie cyklicznych przerwań co 100ms dla bloku OB35 (Rys. 2.18) w którym umieszczona została funkcja odpowiedzialna za regulację PID (*HW config/CPU Properties/Cyclic Interrupts/OB 35*). Dokładny opis bloku OB 35 oraz jego parametru został opisany w rozdziale 5.



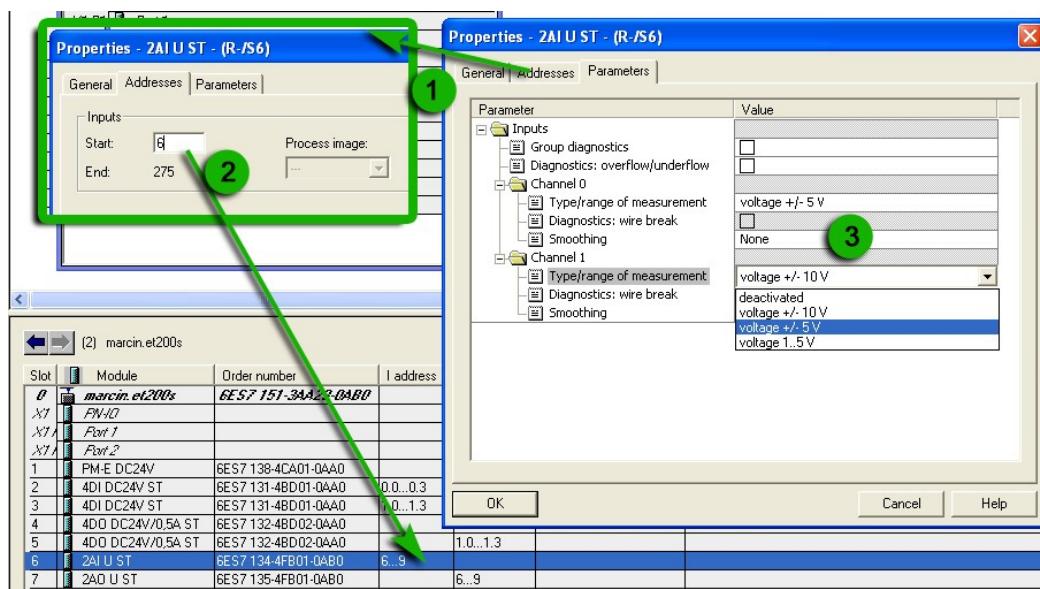
Rys. 2.18. Przerwania cykliczne dla bloku OB35 - *HW config/CPU Properties/Cyclic Interrupts/OB 35*.

Aby sterować serwomechanizmem potrzeba generować sygnał z możliwością modulacji szerokości impulsu (PWM). Do realizacji tego zadania wykorzystano szybki licznik zintegrowany w sterowniku PLC. Parametryzacja szybkiego licznika dla potrzeb PWM polega na: wejściu we właściwości szybkiego licznika *HW Config/CPU Count Properties* (Rys. 2.19 ozn. 1), zarezerwowaniu wyjścia do generowania sygnału (Rys. 2.19 ozn. 2), zdefiniowaniu rodzaju sygnału dla wybranego wyjścia (Rys. 2.19 ozn. 3) i odpowiedniej parametryzacji. Dokładny opis parametryzacji sygnału PWM we właściwościach CPU opisano w podrozdziale 4.3.



Rys. 2.19. Parametryzacja szybkiego licznika - *HW Config/CPU Count Properties*.

Kolejnym krokiem jest parametryzacja modułów wyspy ET200S. W celu przejrzystości programu należy zdefiniować adresy poszczególnych modułów (Rys. 2.20. Ozn. 1,2) oraz zakresy napięć dla modułów analogowych (Rys. 2.20. Ozn. 1,2). Wejścia analogowe służą do odczytu wartości napięcia z czujników optycznych, które pracują w zakresie 0.1V – 2.9V w związku z czym zakresy napięć wejściowych modułów analogowych należy ustawić na +/- 5V.



Rys. 1. Parametryzacja modułów wyspy ET200S.

3. PROJEKTOWANIE SIECI PROFINET DLA KOMUNIKACJI W CZASIE RZECZYWISTYM

Standard PROFINET IO wyróżnia trzy typy urządzeń: IO-Controller, IO-Device, IO-Supervisor [9]:

- **IO-Controller:** jest to sterownik, na którym wykonywany jest program sterujący instalacją (np. CPU 313),
- **IO-Device:** jest to urządzenie obiektowe, przypisane (zdalnie) do sterownika IO-Controller (np. ET200S),
- **IO-Supervisor:** urządzenie programujące (lub komputer PC) z funkcjami diagnostycznymi itp.

Dane mogą być przesyłane pomiędzy sterownikiem (IO-Controller), a urządzeniami obiektowymi (IO-Devices) przez następujące kanały:

- cykliczne dane wej/wyj poprzez kanał czasu rzeczywistego,
- alarmy i zdarzenia przez kanał czasu rzeczywistego,
- parametryzacja, konfiguracja i odczyt informacji diagnostycznych poprzez kanał standardowy oparty na UDP/IP.

Komunikacja w systemie PROFINET [10,11] może być elastycznie skalowana. Ma ona trzy poziomy wydajności:

- **TCP, UDP oraz IP** dla danych niekrytycznych czasowo, jak np. dane konfiguracyjne i parametry,
- **Real Time (RT)** dla danych krytycznych czasowo wykorzystywanych w automatyce przemysłowej,
- **Izochroniczny tryb Real Time (IRT)** do wyjątkowo wymagających zastosowań jak np. sterowanie numeryczne (Motion Control).

Komunikacja standardowa TCP/UDP. Podstawowym protokołem komunikacyjnym w systemie PROFINET jest TCP/IP. TCP/IP stał się standardem dla wszelkich aplikacji wykorzystujących technologie IT. Trzeba zauważyć, że w odniesieniu do współpracy pomiędzy różnymi aplikacjami nie wystarczy tylko ustanowienie wspólnego kanału komunikacyjnego (w warstwie 4) w oparciu o TCP lub UDP pomiędzy urządzeniami. TCP/IP dostarcza tylko podstawowych funkcji do przesyłania danych w sieci lokalnej lub rozproszonej. Do wymiany danych wymagane są tzw. protokoły aplikacyjne leżące w warstwach wyższych niż TCP czy UDP. Współpraca urządzeń

możliwa jest tylko wtedy, kiedy ten sam protokół aplikacyjny jest wykorzystywany przez wszystkie urządzenia. Przykładami popularnych protokołów są np. wykorzystywany w Internecie SMTP (poczta e-mail), FTP (przesyłanie plików), HTTP (używany do przesyłania stron internetowych).

Komunikacja Real Time. W automatyce przemysłowej aplikacje czasu rzeczywistego wymagają czasów odświeżania i odpowiedzi na poziomie 5 - 10 ms. Czas odświeżania jest to czas jaki upływa od momentu utworzenia zmiennej w aplikacji urządzeniu, następnie przesłania jej do partnera przez system komunikacyjny oraz ponowne jej udostępnienie dla aplikacji w stacji partnerskiej. Dla zapewnienia pierwszoplanowego wykonywania programu aplikacyjnego w sterowniku, komunikacja czasu rzeczywistego może jedynie w minimalnym stopniu obciążać procesor urządzenia. Jak pokazuje doświadczenie czasy transmisji poprzez sieć zbudowaną na bazie standardu *Fast Ethernet* (100 Mbps) lub szybszych są pomijalne w stosunku do czasów wykonania programu w sterownikach. Oznacza to, że czas potrzebny na udostępnienie danych w aplikacji dostawcy nie ma praktycznie wpływu komunikacja. To samo odnosi się do przetwarzania danych przez aplikację od-biorcy. Wynika z tego, że wszelka poprawa w czasie odświeżania i przez to w odpowiedzi czasu rzeczywistego osiągane są przede wszystkim przez właściwą optymalizacją stosu komunikacyjnego odbiorcy i dostawcy.

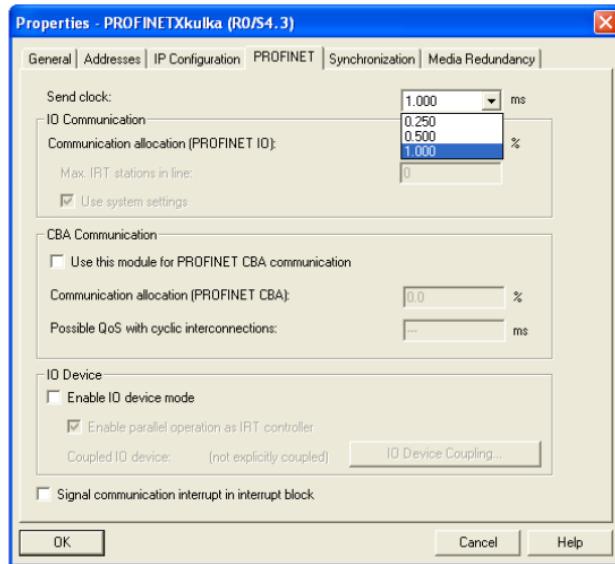
Komunikacja IRT (Isochronous Real Time). Jest stosowany w aplikacjach z synchronizacją okresu czasu przesyłania danych. W trybie IRT czasu cyklu sieci jest dużo poniżej 1ms. Do tego typu aplikacji głównie zalicza się napędy/sterowanie ruchem (motion control). Główne cechy trybu IRT to:

- Zdeterminowane przesyłanie danych w czasie rzeczywistym,
- Czas odświeżania informacji <1ms
- Nierównomierność czasu cyklu <1um
- Synchronizacja czasu,

3.1. Ustawianie cyklu wymiany danych i cyklu odświeżania informacji przez urządzenie w sieci PROFINET

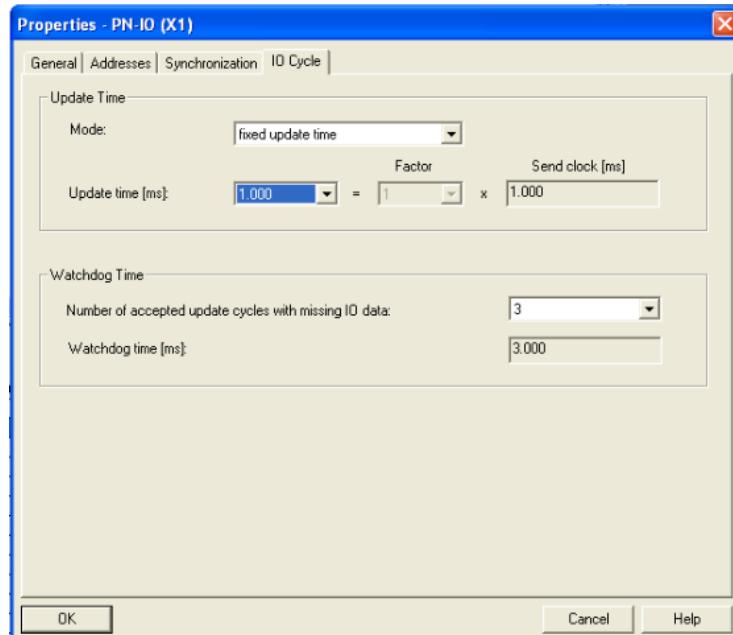
Ustawienie cyklu/zegara nadawania danych w urządzeniu IO-Controller tzw. zegar nadawania (*ang. send clock*) umożliwia okno *PN-IO/Object Properties/PROFINET/Send clock* (Rys. 3.1). Zegar nadawania może być edytowany tylko wtedy, gdy urządzenie IO-Controller nie pracuje w trybie synchronicznej wymiany danych czasu rzeczywistego IRT (zakładka *PN-IO/Object Properties/Synchronization/Synchronization role = not*

synchronized). Wartość zegara nadawania może wynosić poniżej 1ms tylko w trybie pracy czasu rzeczywistego RT lub IRT [6,9].



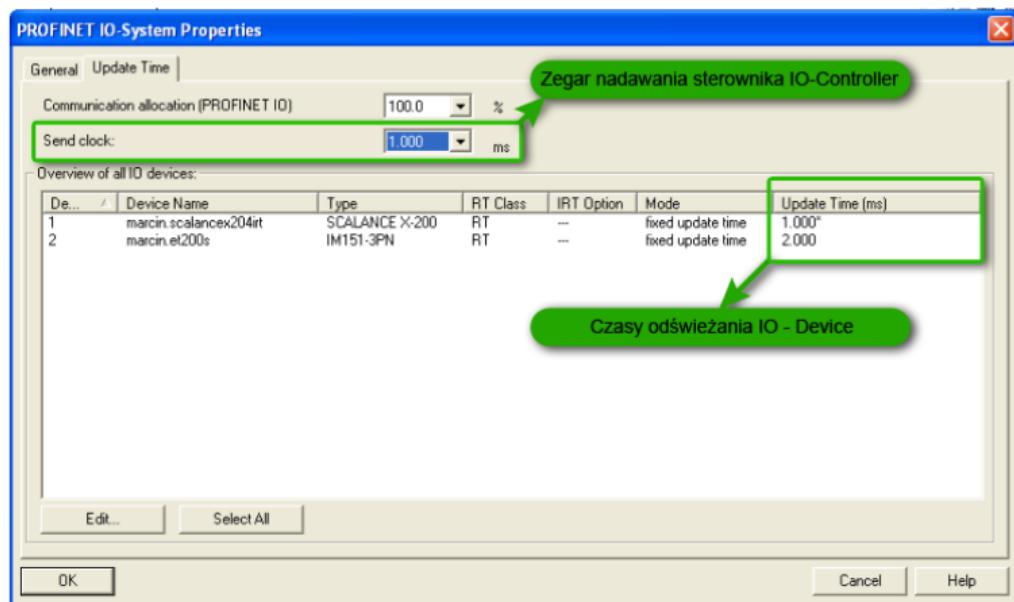
Rys. 3.1. Edycja zegara nadawania stacji PLC, zakładka *PN-IO/Object Properties/PROFINET*.

Ustawienie cyklu odświeżania informacji dla przykładowej stacji IO-Device jest ustawiany w oknie *Properties/IO Cycle/Update Time* (Rys. 3.2)



Rys. 3.2. Ustawianie czasu odświeżania stacji IO-Device, zakładka *Properties/IO Cycle/Update Time*.

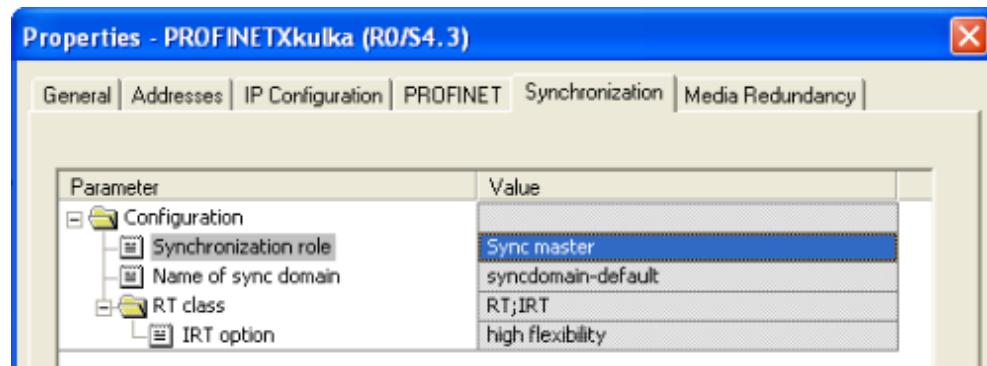
Podgląd czasów odświeżania wszystkich stacji IO-Device oraz zegara nadawania sterownika IO-Controller jest możliwy poprzez wybór właściwości magistrali sieci, zakładka Object Properties/ Update Time (Rys. 3.3).



Rys. 3.3. Okno właściwości magistrali sieci, zakładka *Object Properties/Update Time*.

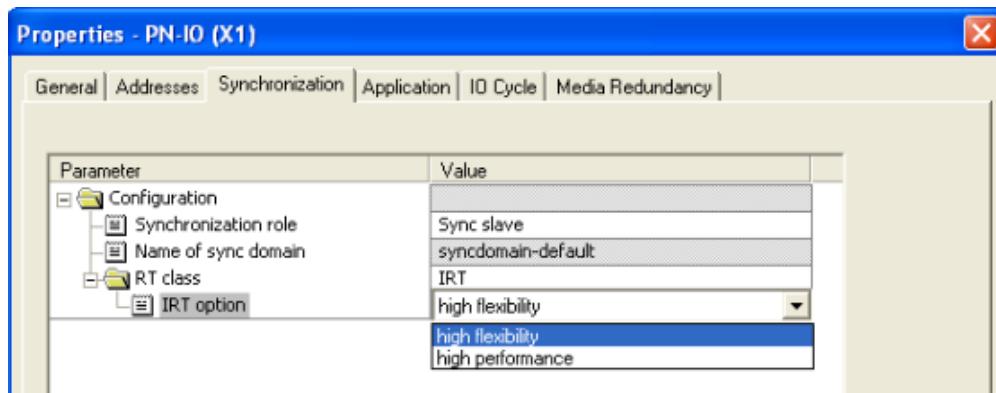
3.2. Obsługa czasu rzeczywistego

Konfiguracja komunikacji IRT polega na włączeniu tego trybu pracy w sterowniku IO-Controller poprzez ustawienie *RT class* na IRT. Polega to na włączeniu opcji *Sync master* (Rys. 3.4). Zaznaczenie opcji *Sync master* dla procesora CP oznacza że, okres czasu przesyłania danych pozostałych urządzeń (*Sync slave*) będzie się synchronizował według procesora CP [6,9].



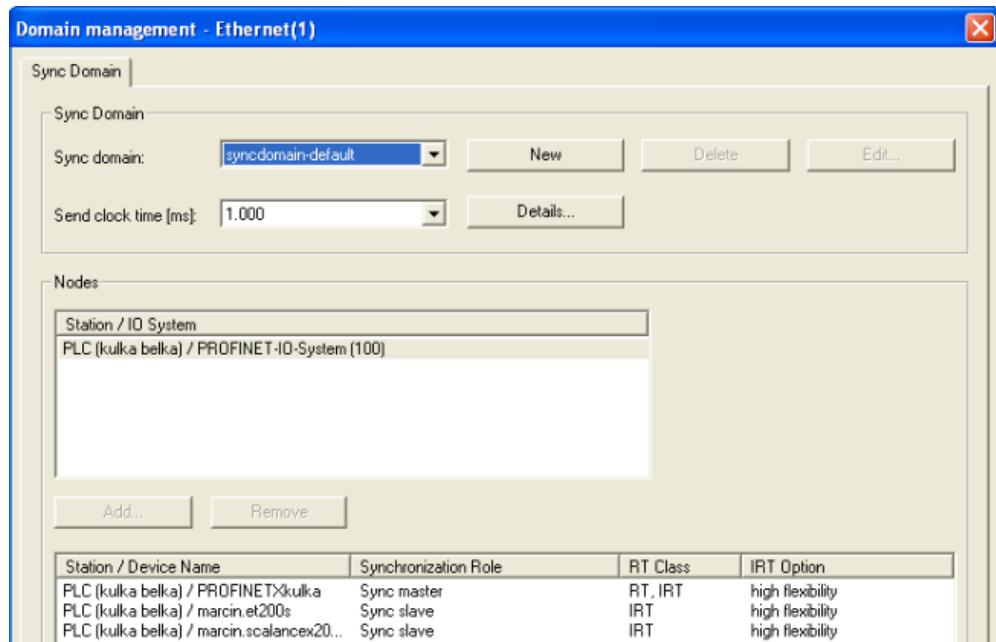
Rys. 3.4. Parametry urządzenia *IO-Controller*, zakładka *PN-IO/Object Properties/Synchronization*.

Kolejnym krokiem jest konfiguracja IRT na dodanym urządzeniu do sieci typu IO-Device. W tym przypadku polega to na włączeniu opcji *Sync slave* (Rys.3.5). W obu przypadkach istnieją dwie opcje pracy, pierwsza czasu rzeczywistego RT i druga czasu rzeczywistego z synchronizacją zegara w wszystkich urządzeniach IRT.



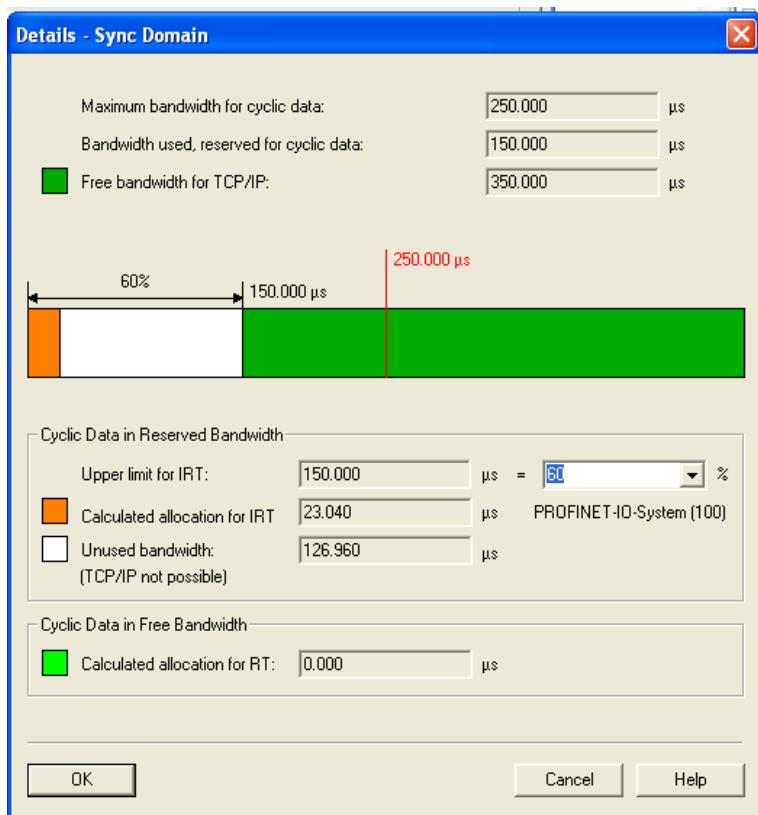
Rys. 3.5. Parametry urządzenia IO-Device, zakładka *PN-IO/Object Properties/Synchronization*.

Sprawdzenie ustawienia sieci PROFINET IO do pracy w trybie IRT można sprawdzić poprzez otwarcie zakładki *Edit/PROFINET IO/Domain Management* (Rys. 3.6). Okno pozwala na ustawienie zegara nadawania dla trybu pracy RT lub IRT w ramach urządzeń danej domeny. Wartość zegara nadawania może być mniejsza od 1ms. Nazwa domeny *Sync domain* jest nadawana automatycznie przez środowisko STEP 7 jako wartość *syncdomain-default*. Okno pozwala na tworzenie nowych domen i nadawania im własnych nazw.

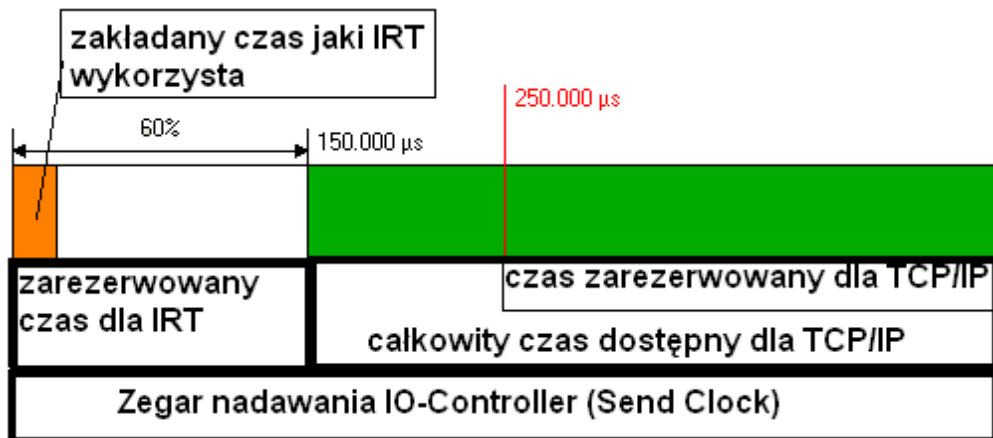


Rys. 3.6. Domena sieci PROFINET IO, zakładka *Edit/PROFINET IO/DOMAIN Managment*.

Kolejnym krokiem konfiguracji pracy sieci w czasie rzeczywistym jest ustawienie zarezerwowanego pasma transmisji IRT (przycisk Details, rys. 3.6 / 3.7). Pasmo czasu dla transmisji IRT można ustawić maksymalnie do 50% wartości zegara nadawania. Pozostała połowa jest zarezerwowana dla transmisji TCP/IP.

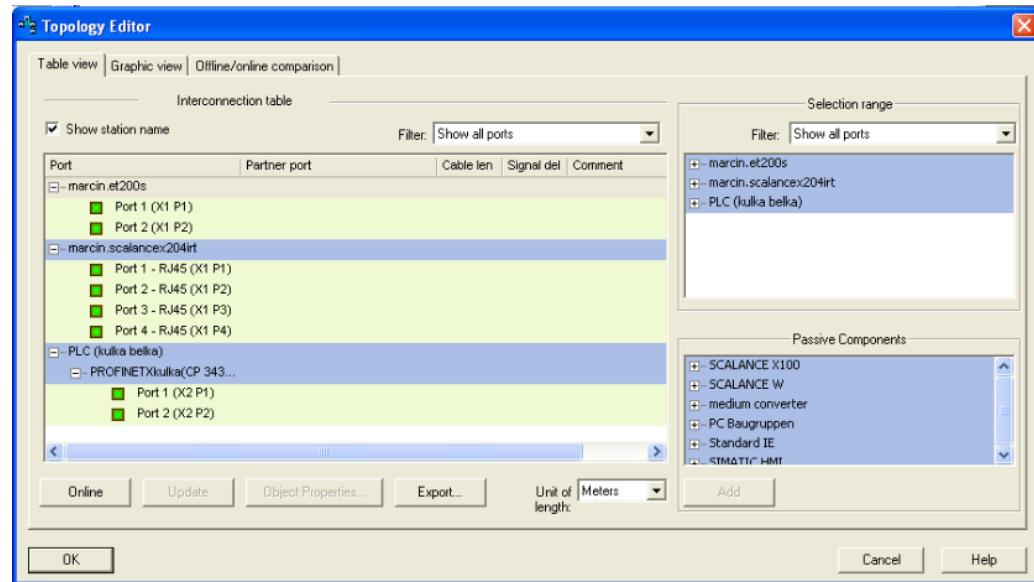


Rys. 3.7. Domena sieci PROFINET IO, zakładka *Edit/PROFINET IO/ Domain Management/Details*.



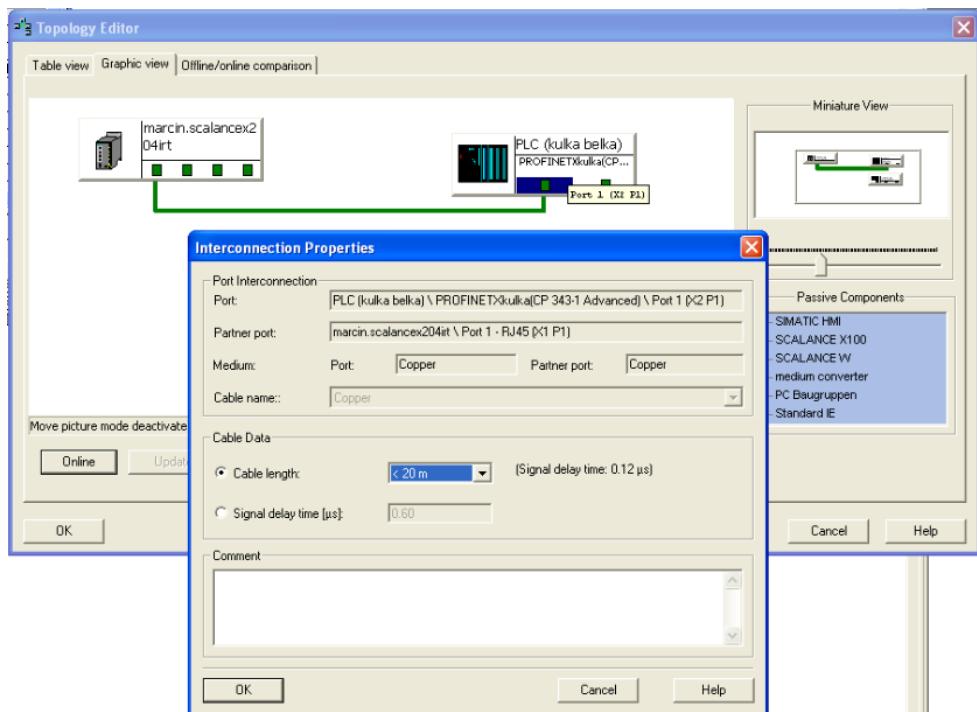
Rys. 3.8. Znaczenie *Send Clock*.

Wykorzystanie trybu pracy IRT 'high performance' wymaga zdefiniowania topologii sieci. Jest to wymagane ze względu na kompilator urządzenia IO-Controller, który musi generować listę zadań dla poszczególnych urządzeń IO-Device z okresem, który i w jakim momencie powinna wysyłać dane. Definicja topologii sieci jest możliwa poprzez uruchomienie edytora topologii dostępnego w zakładce *Edit/PROFINET IO/ Topology*. Okno topologii posiada kilka interfejsów. Pierwszy *Table view* (Rys. 3.9) wyświetla zestawienie urządzeń i ich portów w sieci. Pozwala na wyświetlenie podłączonych i nie podłączonych portów.



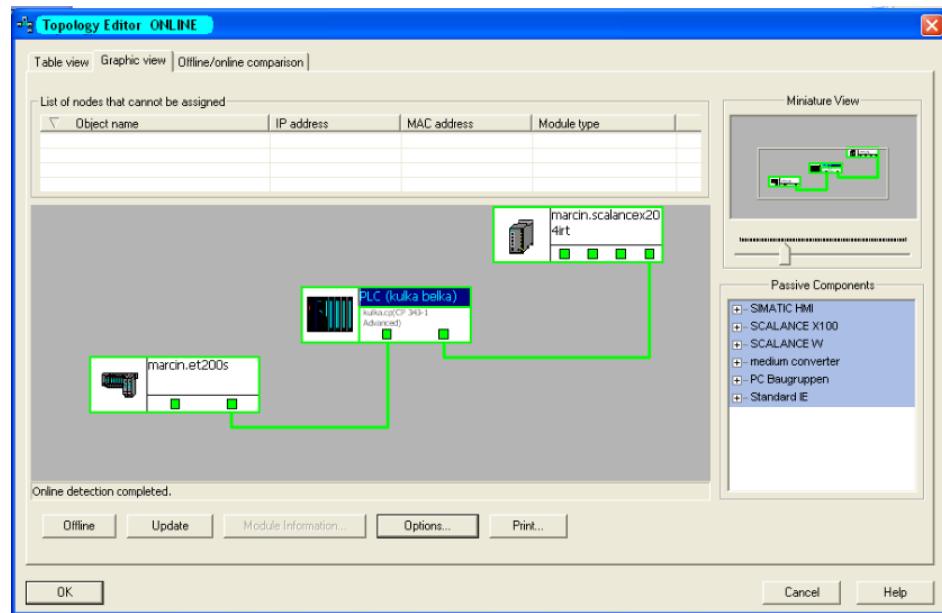
Rys. 3.9. Topologia sieci *Table View*, zakładka *Edit/PROFINET IO/ Topology/Table View*.

Drugi to interfejs graficzny *Graphic view*, który pozwala na graficzne zestawienie/połączenie poszczególnych portów urządzeń zgodnie z konfiguracją sprzętową. Podczas wykonywania połączeń w oknie *Interconnection Properties* (Rys. 3.10) można sprecyzować parametry medium transmisji (rodzaj i długość kabla komunikacyjnego).



Rys. 3.10. Topologia sieci, zakładka *Edit/PROFINET IO/ Topology/Graphic View/ Interconnection Properties*.

Zdefiniowana topologia sieci musi być załadowana do pamięci urządzenia IO-Controller. Interfejs graficzny okna topologii sieci pozwala także na pracę w trybie *online*. Co może być wykorzystane do identyfikacji poprawności połączeń urządzeń. Jeśli cała konfiguracja topologii sieci przebiegła poprawnie, po załadowaniu jej do pamięci sterownika i włączeniu podglądu *online* w podglądzie graficznym powinno uzyskać się efekt widoczny na poniższym rysunku (podświetlenie połączeń zielonym kolorem).



Rys. 3.11. Podgląd online topologii sieci, zakładka *Edit/PROFINET IO/ Topology/Graphic View*.

4. PROGRAMOWANIE OBSŁUGI WEJŚĆ/WYJŚĆ

Wymiana danych w sieci PROFINET wymaga użycia odpowiednich funkcji systemowych odpowiedzialnych za wysyłanie i odbieranie pakietów danych w sieci. W układzie sterowania wymiana danych następuje pomiędzy sterownikiem CPU 313 a wyspą ET200S. Do wyspy ET200S podłączono 8 przycisków sterujących obiektem oraz 8 diod sygnalizujących o stanie obiektu, potrzebne więc jest odpowiednie sterowanie wejściami i wyjściami cyfrowymi dla potrzeb układu. Ponadto stacja ET200S odbiera dane analogowe wyrażone w woltach i przetwarza je na postać cyfrową, aby w programie wyświetlić wartość w woltach niezbędną jest funkcja skalująca. Jeden z przycisków aktywuje blok funkcyjny odpowiedzialny za forsowanie zmiennych, operacja ta ma na celu nadpisywanie zmiennych aby wyeliminować każdorazowe wprowadzanie ich po uruchomieniu sterownika.

Czujnik optyczny Sharp w odpowiedzi na odległość od niego przeszkody zwraca napięcie, charakterystyka napięciowo odległościowa czujnika jest nieliniowa. Układ sterowania, aby pracował prawidłowo wymaga jak najbardziej liniowej charakterystyki napięciowo odległościowej. Problem ten rozwiązano linearyzując tę charakterystykę metodą aproksymacji.

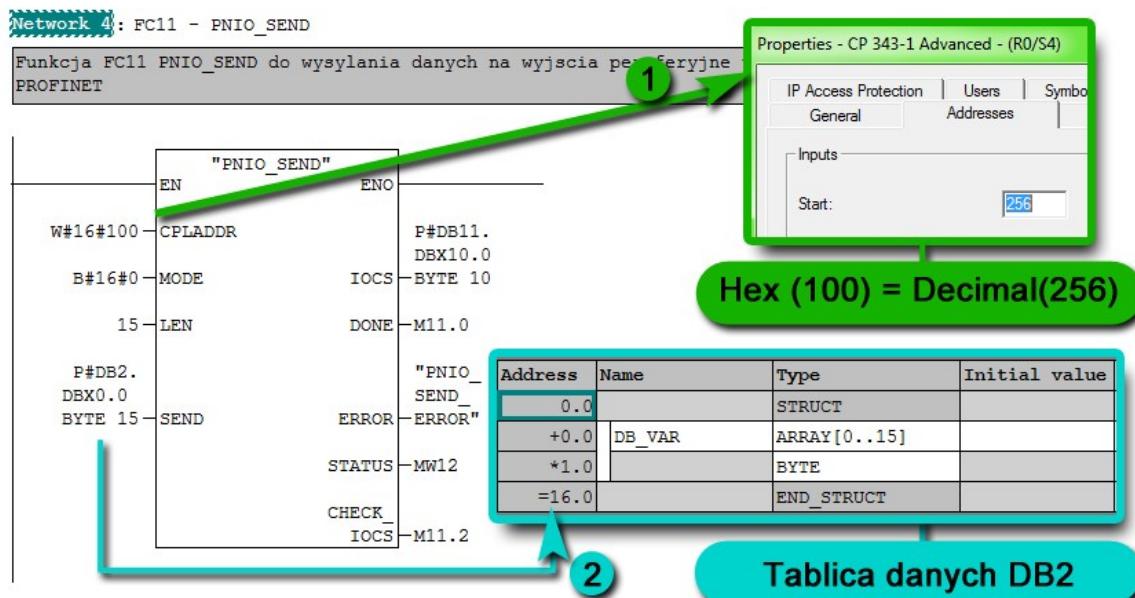
Do generowania sygnału sterującego serwomechanizmem wykorzystano jeden z trzech szybkich liczników sterownika PLC i wbudowanej funkcji generującejowy sygnał.

4.1. Obsługa programowa peryferii

4.1.1. Wymiana danych w sieci PROFINET

Wymiana danych w sieci PROFINET odbywa się za pomocą systemowych funkcyjnych bloków za pośrednictwem procesora komunikacyjnego CP. Za wysyłanie danych odpowiedzialna jest funkcja *FC11 – PNIO_SEND* (Rys. 4.1), natomiast za odbieranie *FC12 – PNIO_RECV* (Rys. 4.2) [6,9,12].

Przed przystąpieniem do parametryzacji bloku FC11, FC12 należy zdefiniować tablice danych zarezerwowane do przesyłu/odbioru danych (Podrozdział 1.2; Rys. 4.1 ozn. 2). Po utworzeniu bloku należy go otworzyć i zdefiniować przestrzeń oraz typ danych (*ARRAY[0..15]/BYTE*).



Rys. 4.1. Funkcja FC11 - PNIO_SEND; 1 – HW Config/CP 343-1 Advanced Properties/Adresses; 2 – parametryzacja tablicy danych DB2.

Opis parametrów funkcji FC11 został umieszczony w tabeli 4. Należy zwrócić szczególną uwagę na 3 parametry funkcji FC11: CPLADDR, LEN, SEND.

Parametr CPLADDR – w miejsce tego parametru należy wpisać adres procesora komunikacyjnego CP, który odczytać możemy we właściwościach procesora CP (HW Config/CP 343-1 Advanced Properties/Adresses). Adres w konfiguracji sprzętowej HW Config (Rys. 4.1 ozn. 1) wyrażony jest w postaci decimalnej (256) natomiast w programie w bloku FC11 należy wpisać adres w postaci heksadecymalnej (100).

Parametr LEN – parametr LEN określa maksymalną długość przestrzeni danych do wysłania. Parametr ten musi spełniać zależność $0 < LEN \leq \max SEND$ (15).

Parametr SEND – parametr SEND określa przestrzeń danych do wysłania. Przestrzeń danych należy określić w sposób kompatybilny z przestrzenią wejść peryferyjnych wyspy ET200S. W projekcie przestrzeń wejść (Rys. 4.2 ozn. 1) wyspy ET200S mieści się w przedziale bitów 0.0 – 9.7 co łącznie daje 10 bajtów, przestrzeń danych do wysyłania zdefiniowano na 15 bajtów więc zakres przestrzeni danych z wejść wyspy ET200S mieści się w przestrzeni danych do wysłania SEND. Adresowanie P#DB2.DBX0.0 BYTE 15 oznacza przestrzeń danych od bitu 0.0 tablicy danych DB2 o długości 15 bajtów, a więc do bitu 15.7.

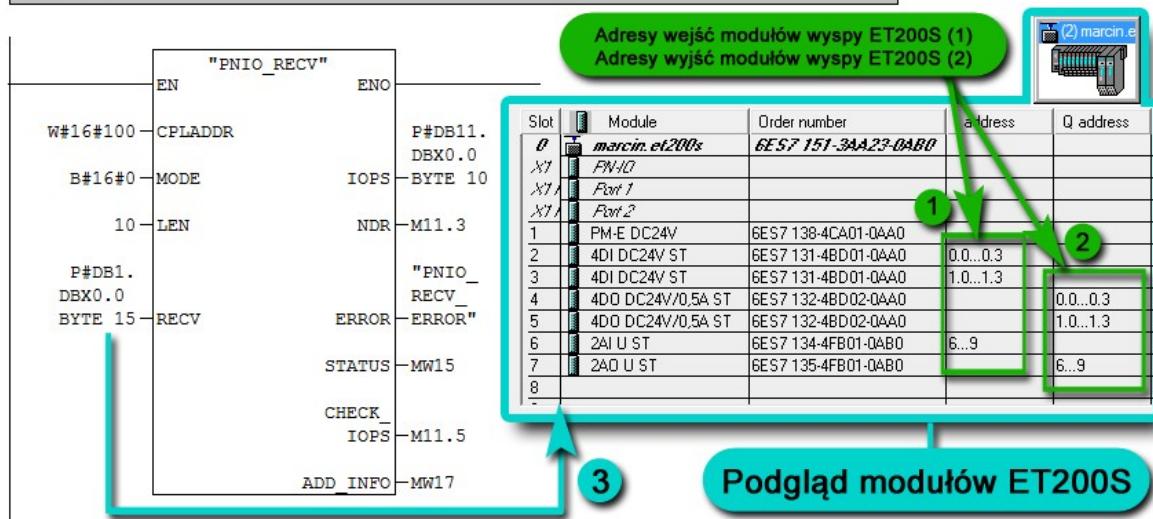
Tabela 5. Opis parametrów funkcji FC11 PNIO_SEND.

FC11 PNIO SEND		
Parametr	Typ danych	Opis
Parametry wejściowe		
CPLADDR	WORD (hex)	Adres procesora komunikacyjnego CP w HW Config
MODE	BYTE	Tryb pracy; zależy od wersji procesora CP
LEN	INT (bajt)	maksymalna długość przestrzeni danych do wysłania
SEND	Wszystkie (DB)	Wskaźnik do wysyłanych danych np. P#M4.0 BYTE 10, P#DB2.DBX0.0 BYTE 15
Parametry wyjściowe		
IOCS	Wszystkie (DB)	Status odbiorcy; dozwolone tylko bajty
DONE	BOOL	Określa status pracy
ERROR	BOOL	Określa błąd komunikacji
STATUS	WORD	Kod statusu; spis kodów w help FC11
CHECK_IOCS	BOOL	Określa status IOCS

W przypadku funkcji FC12 – PNIO_RECV należy postępować analogicznie jak w funkcji FC11.

Network 5 : FC12 - PNIO_RECV

Funkcja FC11 PNIO_RECV do odbierania danych z wejścia peryferyjnych w sieci PROFINET



Rys. 4.2. Funkcja FC12 - PNIO_RECV; podgląd modułów wyspy ET200S w HW Config

Opis parametrów funkcji FC12 został umieszczony w tabeli 5. Należy zwrócić szczególną uwagę na 3 parametry funkcji FC12: CPLADDR, LEN, RECV. W Przypadku Parametrów CPLADDR oraz LEN należy postępować w sposób analogiczny jak przy funkcji FC11.

Parametr RECV – parametr RECV określa przestrzeń danych do odebrania z procesora CP. Przestrzeń danych należy określić w sposób kompatybilny z przestrzenią wyjść peryferyjnych wyspy ET200S. W projekcie przestrzeń wyjść (Rys. 4.2 ozn. 2)

wyspy ET200S mieści się w przedziale bitów 0.0 – 9.7 co łącznie daje 10 bajtów, przestrzeń danych do odebrania zdefiniowano na 15 bajtów więc zakres przestrzeni danych z wyjścia wyspy ET200S mieści się w przestrzeni danych do wysłania RECV. Adresowanie P#DB1.DBX0.0 BYTE 15 oznacza przestrzeń danych od bitu 0.0 tablicy danych DB1 o długości 15 bajtów, a więc do bitu 15.7.

Tabela 6. Opis parametrów funkcji FC11 PNIO_RECV.

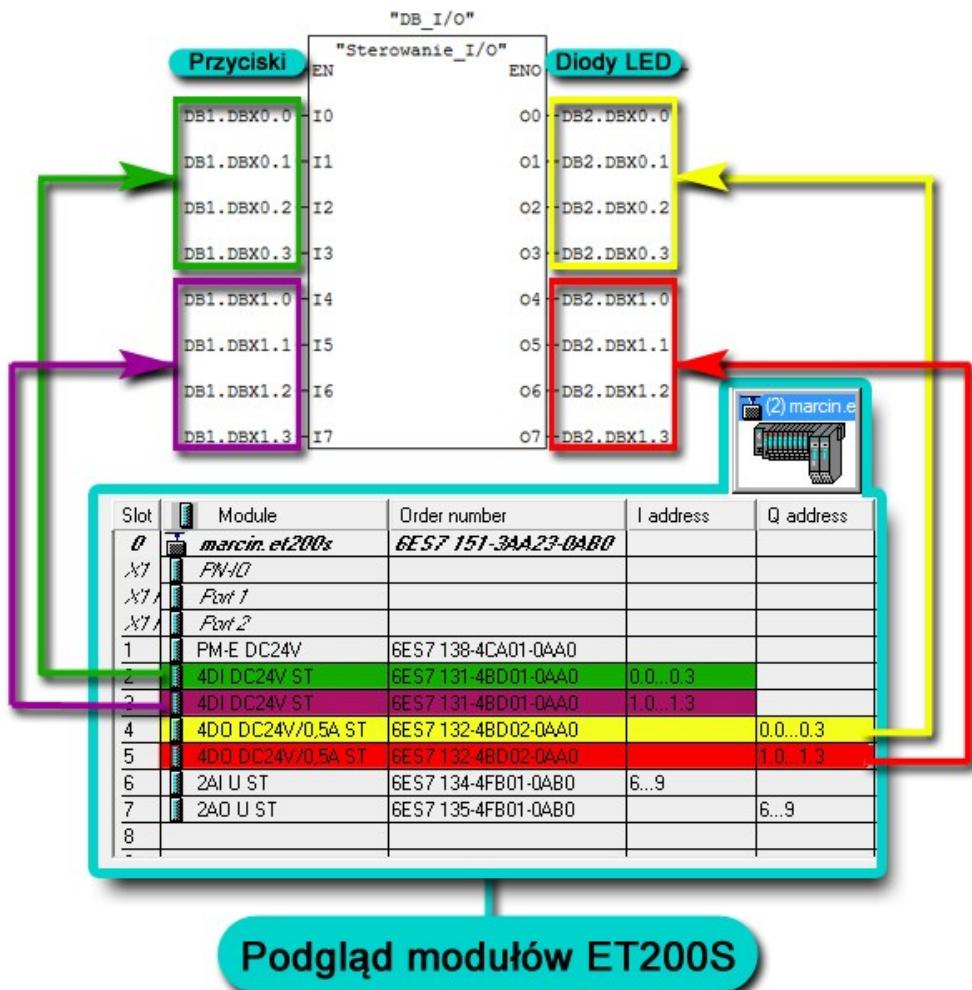
FC11 PNIO_RECV		
Parametr	Typ danych	Opis
Parametry wejściowe		
CPLADDR	WORD (hex)	Adres procesora komunikacyjnego CP w HW Config
MODE	BYTE	Tryb pracy; zależy od wersji procesora CP
LEN	INT (bajt)	maksymalna długość przestrzeni danych do odebrania
RECV	Wszystkie (DB)	Wskaźnik do odebrania danych np. P#M4.0 BYTE 10, P#DB1.DBX0.0 BYTE 15
Parametry wyjściowe		
IOCS	Wszystkie (DB)	Status nadawcy
DONE	BOOL	Określa status pracy
ERROR	BOOL	Określa błąd komunikacji
STATUS	WORD	Kod statusu; spis kodów w help FC11
CHECK_IOCS	BOOL	Określa status IOCS
ADD_INFO	WORD	Dodatkowa informacja diagnostyczna

4.1.2. Sterowanie peryferyjnymi wejściami i wyjściami cyfrowymi

Przyciski i diody konsoli sterującej są podłączone wejścia i wyjścia modułów cyfrowych wyspy ET200S (Rys. 4.3.). W programie przestrzeń danych odpowiadających za przyciski (I0, I1, I2, ...) zdefiniowana jest w tablicy danych DB1, natomiast diody (O0, O1, O2, ...) sygnalizacyjne LED w tablicy danych DB2. Działanie poszczególnych przycisków i diod umieszczone w tabeli 7.



Rys. 4.3. Połączenie konsoli sterującej do wyspy ET200S, gdzie I –przycisk, O – dioda obok przycisku.
Dokładne przypisanie poszczególnych adresów tablic DB do przycisków i diod zostało wyjaśnione na rysunku 4.4.



Rys. 4.4. Przypisanie adresów tablic DB dla przycisków i diod.

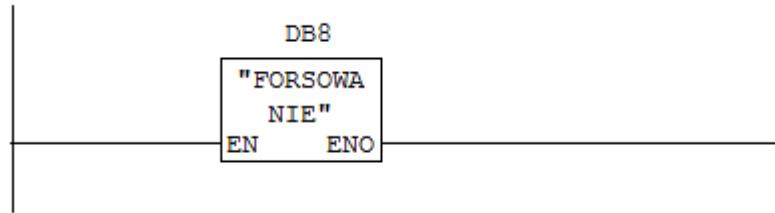
Sterowanie wejściami i wyjściami peryferyjnymi umieszczono w sparametryzowanym bloku funkcyjnym FB 7. Parametry aktualne (np. DB1.DBX0.0) na zewnątrz bloku odpowiadają parametrom formalnym (Rys. 4.4 / Rys 1.7) wewnątrz bloku (np. I0) dzięki czemu program jest bardziej czytelny.

Tabela 7. Funkcje poszczególnych przycisków i diod.

Parametr formalny (I/O)	Parametr aktualny (DB)	Funkcja
Przyciski (kolory w tabeli odpowiadają kolorom przycisków)		
I0	DB1.DBX0.0	Start programu (funkcje PWM i PID); ponowne uruchomienie programu po likwidacji błędu braku kulki na pochylni;
I1	DB1.DBX0.1	Stop programu (funkcje PWM i PID);
I2	DB1.DBX0.2	Dodanie +1 do wartości zadanej regulatora PID
I3	DB1.DBX0.3	Nadpisanie wartości zadanej =55 regulatora PID
I4	DB1.DBX1.0	Nadpisanie wartości zadanej =40 regulatora PID
I5	DB1.DBX1.1	Nadpisanie wartości zadanej =25 regulatora PID
I6	DB1.DBX1.2	Odejście -1 od wartości zadanej regulatora PID
I7 (Test)	DB1.DBX1.3	Forsowanie zmiennych; likwidacja błędu braku kulki na pochylni (wraz z forsowaniem)
Diody (kolory w tabeli odpowiadają kolorom diod)		
O0	DB2.DBX0.0	Świeci w trakcie działania programu, nie świeci w przypadku stopu programu lub błędu
O1	DB2.DBX0.1	Świeci w przypadku stopu programu; nie świeci w przypadku błędu lub działania programu
O2	DB2.DBX0.2	Miga tylko wtedy gdy kulka znajdzie się w zakresie 22=>x>5
O3	DB2.DBX0.3	Miga tylko wtedy gdy kulka znajdzie się w zakresie 35=>x>22
O4	DB2.DBX1.0	Miga tylko wtedy gdy kulka znajdzie się w zakresie 45=>x>35
O5	DB2.DBX1.1	Miga tylko wtedy gdy kulka znajdzie się w zakresie 55=>x>45
O6	DB2.DBX1.2	Miga tylko wtedy gdy kulka znajdzie się w zakresie x>55
O7	DB2.DBX1.3	Miga tylko w przypadku błędu braku kulki na pochylni

Ze względu na obszerność programu sterowania przyciskami i diodami nie umieszczono go w projekcie a w załączniku 1 „Program układu sterowania/FB7-Sterowanie I/O” wraz z komentarzami.

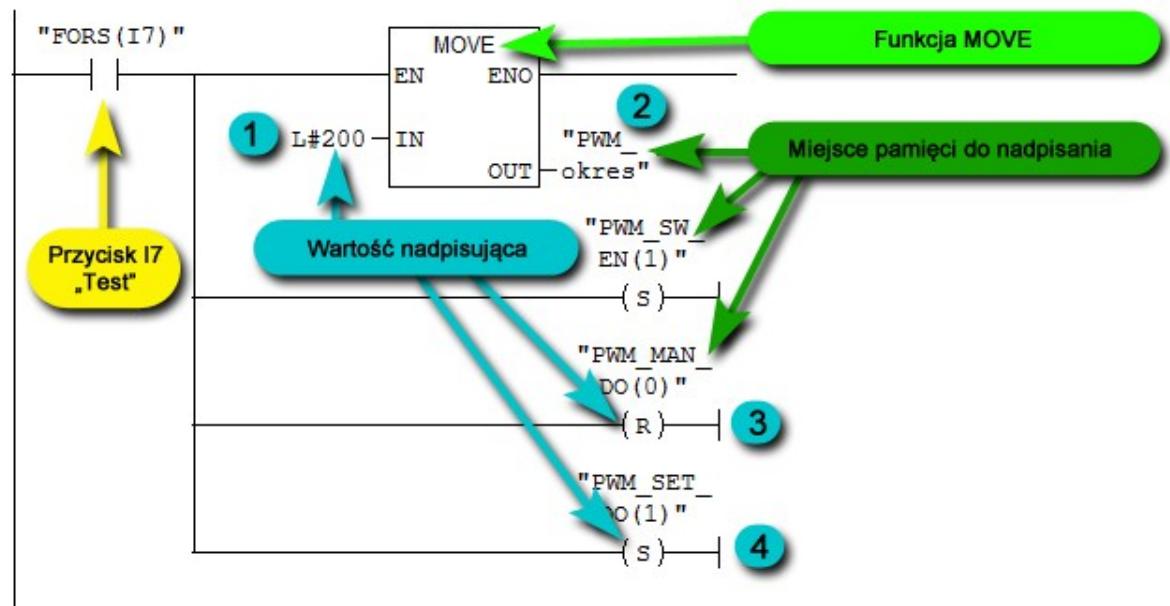
4.1.3. Forsowanie zmiennych



Rys. 4.5. Blok funkcyjne FB 8 - Forsowanie zmiennych.

Poprzez forsowanie zmiennych należy rozumieć nadpisywanie parametrów programu przez wcześniej zdefiniowane wartości. Ma to na celu wyeliminowanie każdorazowego wprowadzania zmiennych po uruchomieniu sterownika lub powrócenie do wartości początkowych parametrów po ich modyfikacji. Forsowanie odbywa się po każdym wcisnięciu przycisku *Test* (I7). Kod programu został umieszczony w bloku funkcyjnym *FB8 – Forsowanie*.

Na rysunku 4.6 pokazano przykładowe forsowanie dla funkcji generowania PWM. W czasie wcisnięcia przycisku *Test* (I7), zostaje aktywowana funkcja MOVE oraz funkcje RESET (ozn. 3) i SET (ozn. 4). Funkcja MOVE podstawia wartość nadpisującą większą od jednego bitu tj. BYTE, INT, WORD, DWORD, REAL itp. (ozn. 1) w miejsce pamięci do nadpisania (ozn. 2). Funkcja SET ustawia stan wysoki dla miejsca pamięci do nadpisania wyrażoną (bit), natomiast RESET stan niski.

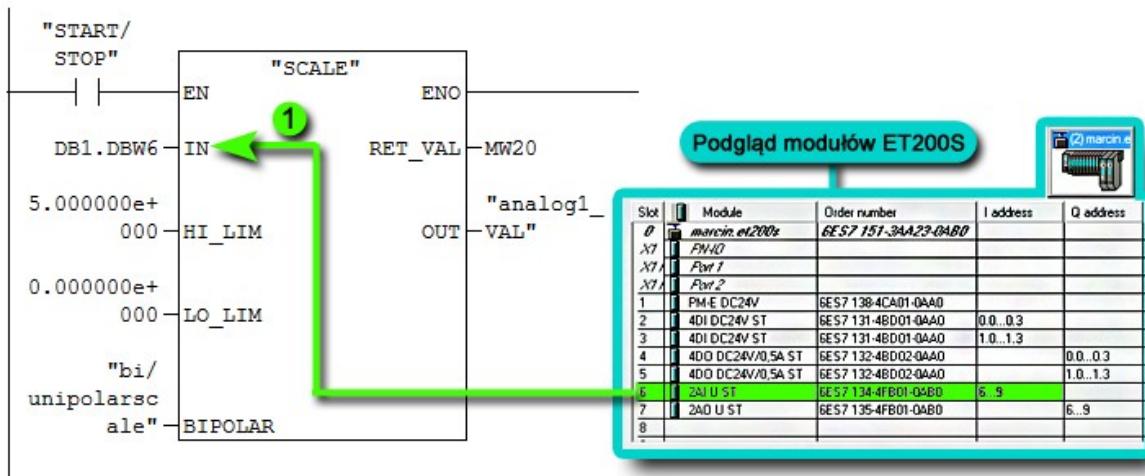


Rys. 4.6. Forsowanie parametrów dla funkcji generowania PWM.

Parametry dla pozostałych funkcji są forsowane analogicznie jak w powyższym przypadku. Cały kod programu odpowiedzialny za forsowanie zmiennych został umieszczony w załączniku 1 „Program układu sterowania/FB8-Forsowanie zmiennych” wraz z komentarzami.

4.1.4. Skalowanie wartości analogowych

Do skalowania wartości analogowych służy systemowy blok funkcyjny *FC105- SCALE CONVERT* (Rys. 4.7). Blok ten znajduje się w bibliotece *Standard Library* w zakładce *TI-S7 Converting Blocks* [4,6].



Rys. 4.7. Funkcja *FC105 SCALE CONVERT*

Dla wejść analogowych w aplikacji *HW Config* zostały przypisane adresy 6..9 (abajtów), po 2 bajty na każde wejście analogowe, w związku z czym dla pierwszego wejścia analogowego przypadają bajty 6 i 7 (czujnik optyczny 1; Rys. 4.7 ozn. 1) natomiast dla drugiego bajty 8 i 9 (czujnik optyczny 2). Funkcja *FC12 – PNIO_RECV* (Rys. 4.2) wysyła dane z wyspy ET200S do sterownika PLC zapisane w tablicy DB1. Zarezerwowana przestrzeń w tablicy danych DB1 dla wejść analogowych zdefiniowana jest więc następująco:

- pierwsze wejście analogowe (czujnik 1) – **DB1.DBW6** (WORD6 = BYTE6,7),
- drugie wejście analogowe (czujnik 2) – **DB1.DBW8** (WORD8 = BYTE 8,9).

Opis parametrów funkcji FC105 *SCALE CONVERT* umieszczono w tabeli 8.

Tabela 8. Opis parametrów funkcji FC105 SCALE CONVERT

FC105 SCALE CONVERT		Opis
Parametr	Typ danych	
Parametry wejściowe		
IN	INT	Wartość analogowa odczytana z modułu analogowego
HI_LIM	REAL	Górna granica skalowania
LO_LIM	REAL	Dolna granica skalowania
BIPOLAR	BOOL	1 – tryb bipolarny; 0 – tryb unipolarny
Parametry wyjściowe		
RET_VAL	WORD	Zwraca wartość 0 w przypadku bezbłędnego skalowania
OUT	REAL	Przeskalowana wartość fizyczna

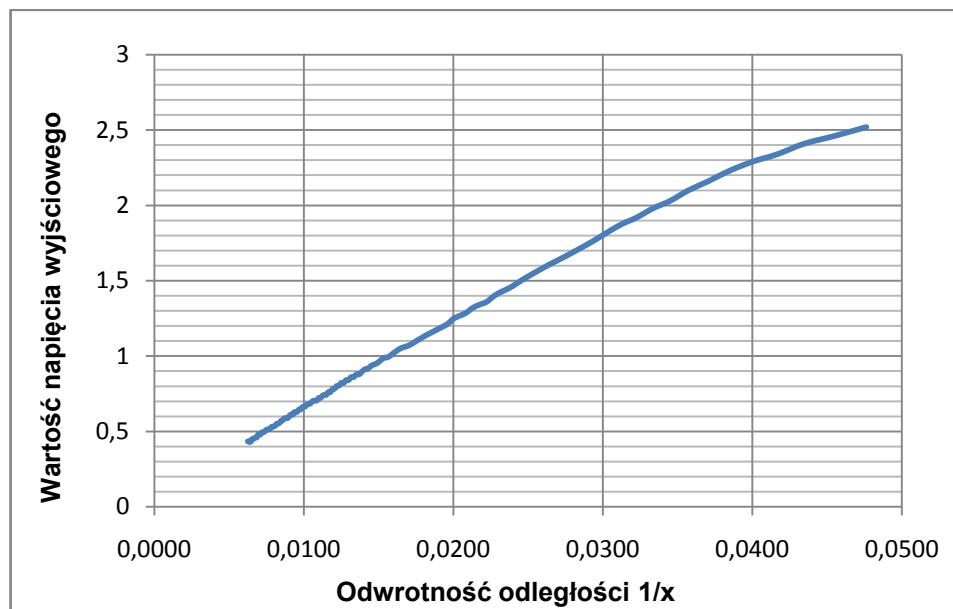
Czujnik optyczny pracuje w zakresie napięć 0.1V-2.9V natomiast zakres napięć wejściowych dla modułów analogowych w edytorze konfiguracji *HW Config* ustawione są na +/-5V (Rys. 2.20 ozn. 3). Przetwornik analogowo cyfrowy tj. moduł AI wyspy ET200S na wyjściu zwraca wartość cyfrową od -27648 (-5V) do 27648 (+5V). Tryb bipolarny służy do skalowania wartości wejściowej w zakresie od -27648 do 27648, natomiast tryb unipolarny w zakresie od 0 do 27648. Dlatego dla prawidłowego skalowania wartości cyfrowej na wartość wyrażoną w woltach należy zdefiniować parametry funkcji FC105 następująco:

- HI_LIM = 5.0 – w związku z zakresem modułu analogowego,
- LO_LIM = 0.0 – ponieważ napięcie wyjściowe czujnika nie spada poniżej 0,
- BIPOLAR = 0 – ponieważ cyfrowa wartość wejściowa jest nieujemna.

4.2. Linearyzacja charakterystyki czujnika optycznego

Jak przedstawia rysunek 2.9, czujnik odległości posiada nieliniową charakterystykę odległości od wartości napięcia. Układ pomiarowy wymaga linearyzacji funkcji charakterystyki napięciowo odległościowej w celu łatwej implementacji oraz otrzymania jak najbardziej zbliżonej do rzeczywistości odległości kulki od początku pochylni.

Tworząc charakterystykę zależności napięcia wyjściowego od odwrotności odległości $\frac{1}{x}$ (Rys. 4.8) otrzymujemy linię zbliżoną do prostej z czego wynika że można charakterystykę napięciowo odległościową aproksymować linią prostą [16,17].



Rys. 4.8. Zależność napięcia na wyjściu czujnika od odwrotności odległości.

Aby wykonać aproksymację liniową przekształca się wartości osi rzędnych:

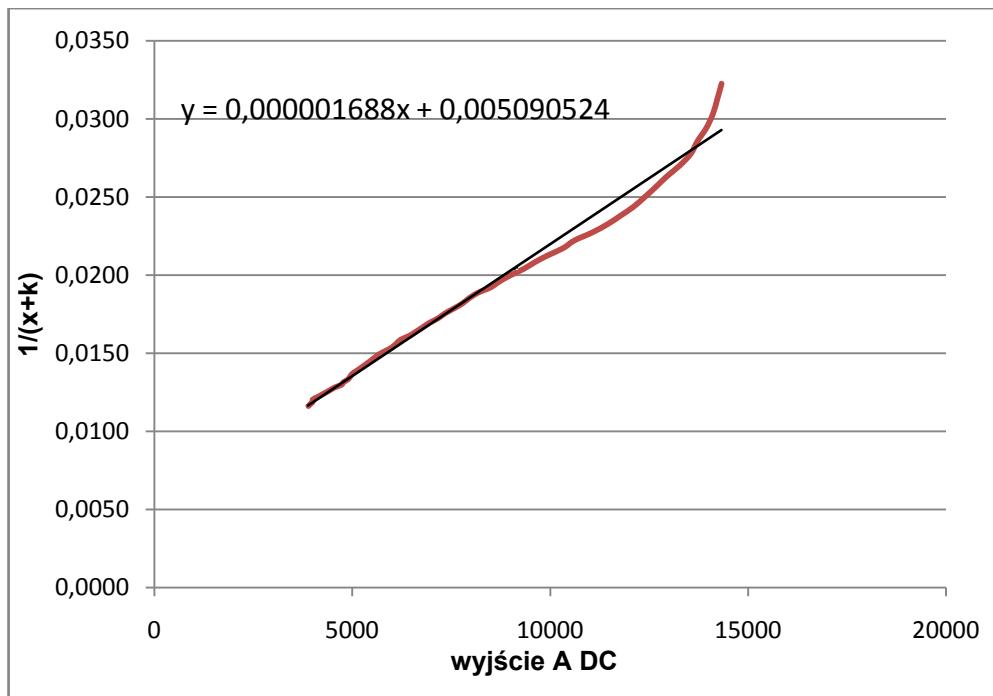
$$\frac{1}{x + k}$$

gdzie:

x - zmierzona odległość,

k - stała zależna od modelu sensora ($k=11$).

Przekształconą charakterystykę przedstawia rysunek 4.9:



Rys. 4.9. Zależność przekształcenia $1/(x+k)$ do wartości wyjścia z przetwornika.

Kolejnym krokiem jest znalezienie linii prostej najbardziej zbliżonej do tak przekształconej charakterystyki. Wyznaczenie aproksymacji liniowej umożliwia program Excel (Rys. 4.9).

Ogólne równanie prostej ma postać:

$$y = a * X + b \quad (3.1)$$

Podstawiając za X wyjściową wartość cyfrową napięcia z przetwornika oraz zależność ze wzoru (3.1) otrzymujemy:

$$\frac{1}{x+k} = a * ADC + b \quad (3.2)$$

Przekształcając równanie (3.2) do postaci odległości x otrzymujemy:

$$x = \frac{\frac{1}{a}}{ADC + \left(\frac{b}{a}\right)} - k \quad (3.3)$$

Gdzie:

x - dystans kulki od początku pochylni w centymetrach,

k - stała współczynnika czujnika (k=11),

ADC – cyfrowa wartość napięcia wyjściowego czujnika,

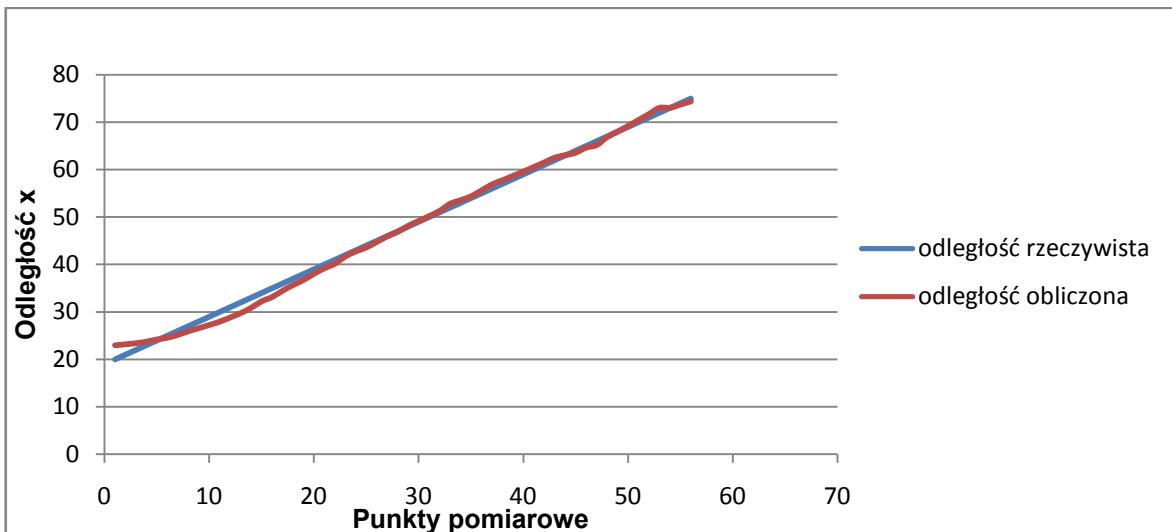
a – stała aproksymacji liniowej odczytana z trendu Excela (Rys. 4.9),

b - stała aproksymacji liniowej odczytana z trendu Excela (Rys. 4.9).

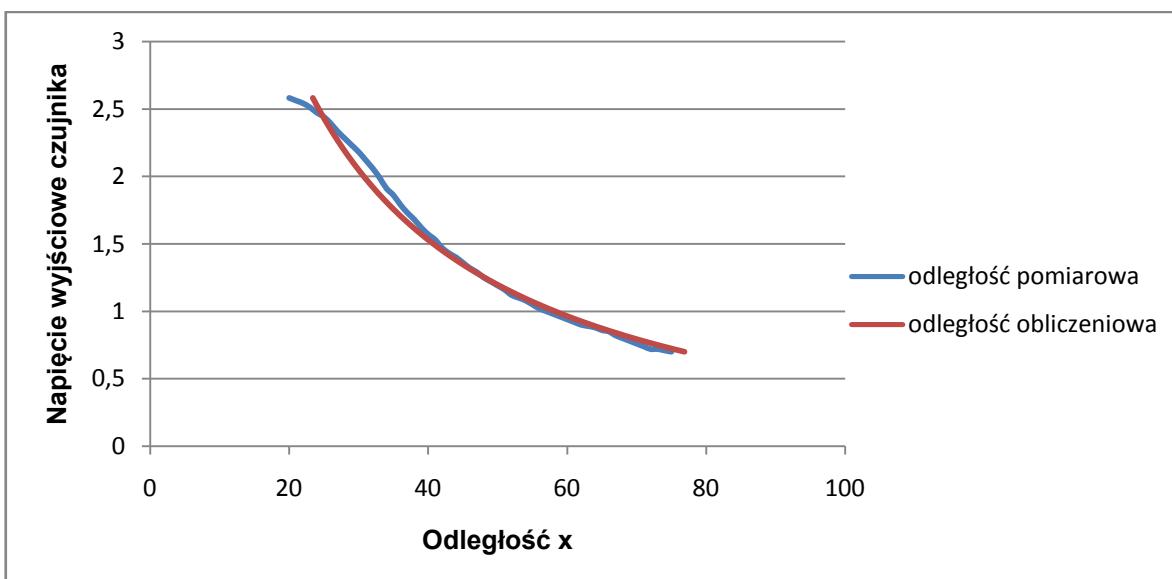
Podstawiając stałe liczbowe do wzoru (3.3) ostatecznie otrzymujemy:

$$x = \frac{588235}{ADC + (3000)} - 11 \quad (3.4)$$

Na podstawie równania (3.4) wyznaczono zależność odległości rzeczywistej i odległości obliczonej do kolejnych punktów pomiarowych przedstawioną na rysunku 4.10 oraz charakterystykę napięciowo odległościową dla odległości pomiarowej i obliczeniowej (Rys. 4.11).

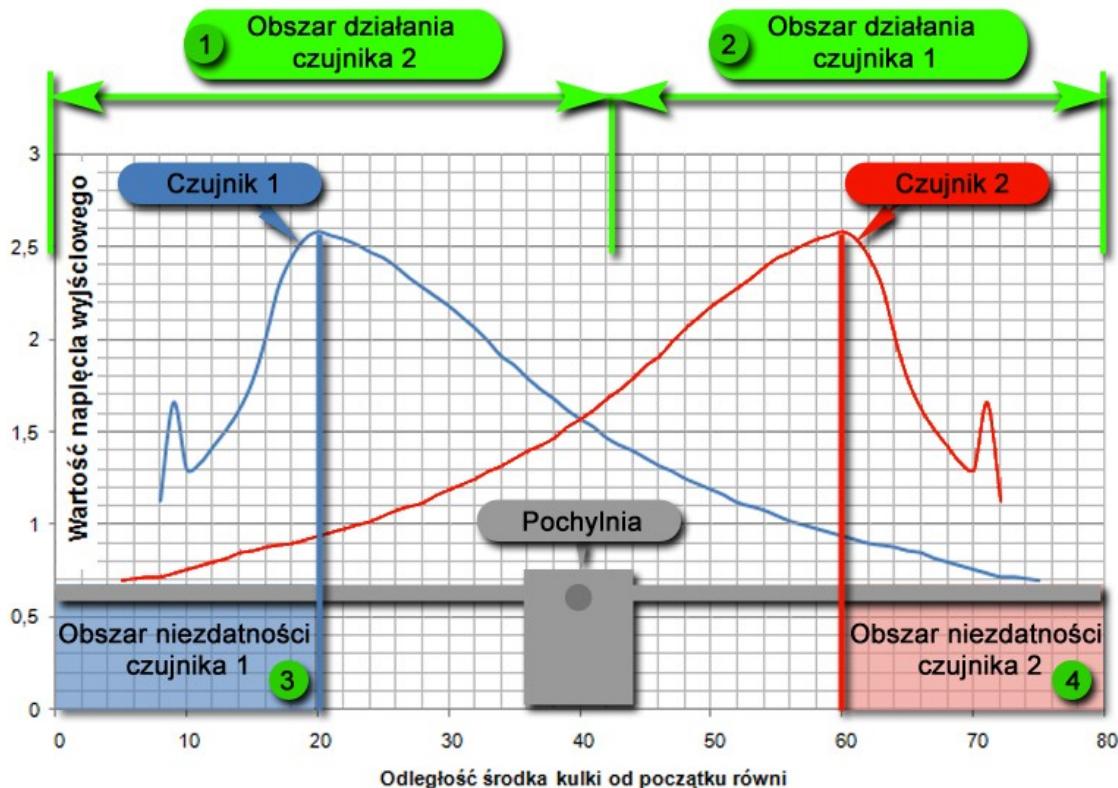


Rys. 4.10. Zależność odległości do kolejnych punktów pomiarowych.



Rys. 4.11. Charakterystyka napięciowo odległościowa na podstawie pomiarów i obliczeń czujnika optycznego.

Jak widać z rysunku 4.10 odległość obliczona ze wzoru (3.4) nie odbiega zbytnio od rzeczywistej odległości kulki i jest akceptowalna przez algorytm sterowania lecz nie w pełnym zakresie pochylni. Z rysunku 2.9 zauważać można że charakterystyka napięciowo odległościowa czujnika w zakresie od 0 cm do 20 cm (obszar niezdolności czujnika) jest na tyle nieliniowy że linearyzacja charakterystyki metodą aproksymacji w tym przedziale nie będzie odzwierciedlała realnego położenia kulki na pochylni. Aby zapewnić prawidłowy odczyt położenia kulki w pełnym zakresie kulki na pochylni zastosowano dwa czujniki optyczne, dla których zasadę sterowania przedstawiono na rysunku 4.12.



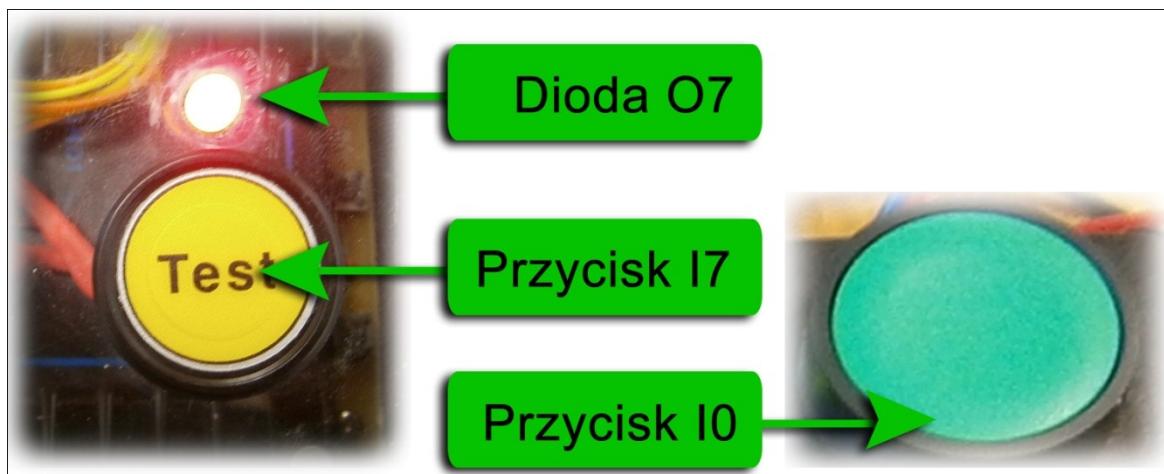
Rys. 4.12. Zasada sterowania dwoma czujnikami optycznymi Sharp.

Rozwiązywanie problemu obszaru niezdolności czujnika polega na takim sterowaniu dwoma czujnikami tak aby jeden czujnik eliminował obszar niezdolności drugiego czujnika i odwrotnie. Gdy kula znajduje się w obszarze 0..43 (Rys. 4.12 ozn. 1), odczyt odległości kulki od początku pochylni odbywa się tylko za pośrednictwem czujnika 2 eliminując ewentualny błędny odczyt czujnika 1 w jego obszarze niezdolności (Rys. 4.12 ozn. 3), natomiast gdy kula znajduje się w obszarze 43..80 (Rys. 4.12 ozn. 2), odczyt odbywa się za pośrednictwem czujnika 1 eliminując ewentualny błędny odczyt czujnika 2 w jego obszarze niezdolności (Rys. 4.12 ozn. 4). Punkt przełączania pracy czujników ustalono w 43 centymetrze od początku pochylni aby uniknąć przełączania w punkcie środkowej stabilizacji kulki na pochylni.

Program linearyzacji charakterystyki napięciowo odległościowej czujników głównie składa się z funkcji matematycznych zmiennoprzecinkowych zawartych w zakładce *Floating-point fct*. Wartość cyfrowa z modułu analogowego wyrażona jest w postaci INT, więc należy ją skonwertować na postać REAL, odbywa się to przez konwersję INT na DINT a następnie na REAL. Do konwersji użyto bloków *I_DI* oraz *DI_R* zawartych w zakładce *Converter*.

Sterowanie odczytem dwóch czujników odbywa się poprzez określenie obszarów działania poszczególnych czujników. Do realizacji tej funkcji użyto komparatorów zawartych w zakładce *Comparator*.

Błąd braku kulki na pochylni:



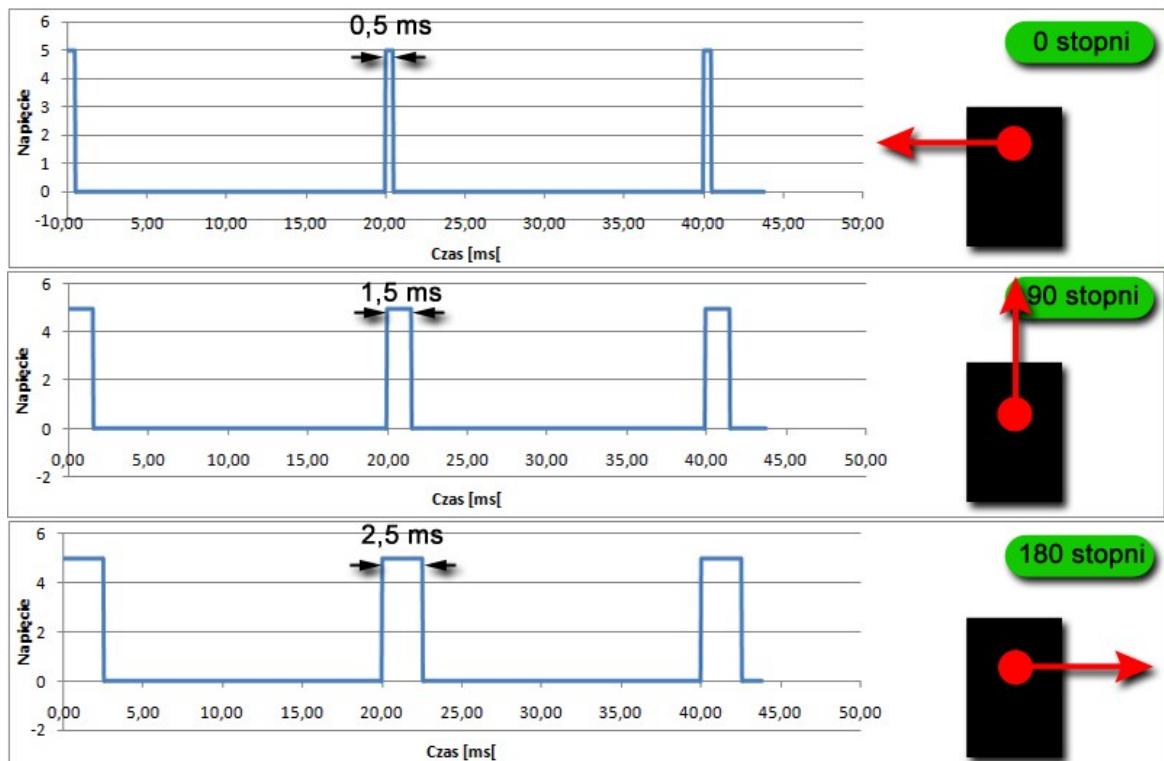
Rys. 4.13. Sygnalizacja błędu braku kulki na pochylni.

Czujniki są umieszczone na pochylni naprzeciw sobie, w związku z tym że działają wykorzystując fale podczerwone, po zdjęciu kulki z pochylni czujniki zakłócają się w efekcie dając zakłóconą wartość. Jeden z czujników po linearyzacji jego odpowiedzi w trakcie zakłócania zwraca wartość ujemną. Zjawisko to wykorzystano do wywołania błędu i zatrzymania wychylania pochylni. Kod programu błędu braku kulki na pochylni wykorzystuje komparator $CMP>R$, gdy odczyt z czujnika osiągnie wartość mniejszą niż 5cm zostanie uruchomiony znacznik *error* (RESET) powodujący zatrzymanie programu. Po wywołaniu błędu następuje miganie diody O7 co 0.1s i nie jest możliwe uruchomienie programu przyciskiem I0 (*START*), należy najpierw umieścić kulkę na pochylni, wcisnąć przycisk I7 (*TEST*) i dopiero uruchomić program przyciskiem I0 (*START*).

Kod programu linearyzacji, sterowania czujnikami oraz wywołania błędu braku kulki na pochylni został umieszczony w załączniku 1 „Program układu sterowania/FB3-Linearyzacja” wraz z komentarzami.

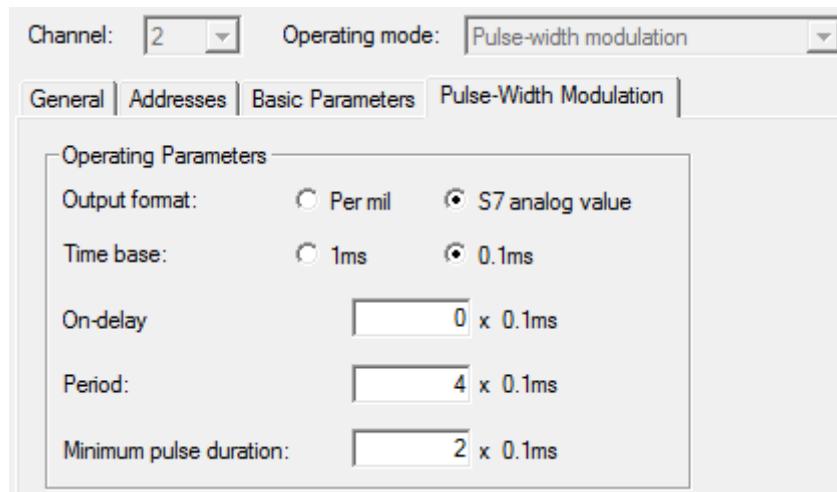
4.3. Sterowanie PWM serwomechanizmem

Sterowanie serwomechanizmem polega na podaniu mu prostokątnego sygnału sterującego o częstotliwości 50Hz (puls co 20ms) z modulacją szerokości wypełnienia pulsu. Położenie wału serwa jest więc zależne od wypełnienia pulsu, którego wypełnienie powinno mieścić się w zakresie od 0,5 ms do 2,5ms (Rys. 4.14) [18].



Rys. 4.14. Zależność wypełnienia sygnału sterującego serwomechanizmem od wychylenia wału

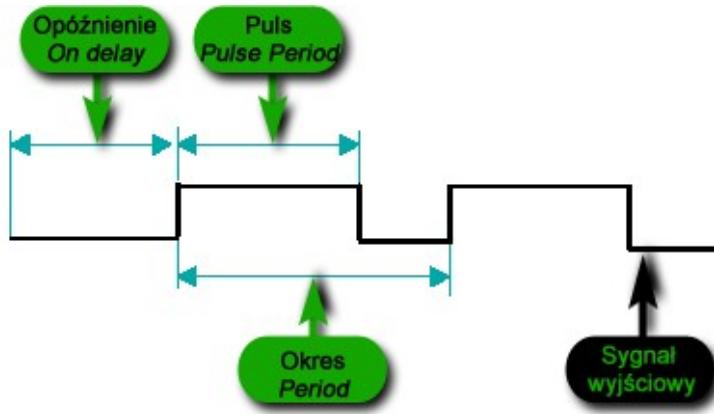
Z punktu widzenia programowego, aby generować sygnał PWM należy wpierw zdefiniować wyjście szybkiego licznika sterownika PLC. Odbywa się to we właściwościach sterownika PLC: *HW Config/CPU Count Properties* (Rys. 2.19/ Rys 4.15) Fizycznymi wyjściami sygnału PWM mogą być pierwsze trzy wyjścia zintegrowanego modułu cyfrowego sterownika DO. Wyjścia te rezerwuje się we właściwościach szybkiego licznika w polu wyboru *Channel*. W polu wyboru *Operating mode* wybierany jest sposób wykorzystania szybkiego licznika, w tym przypadku jest to *Pulse-width modulation* (PWM) [4,6].



Rys. 4.15. Parametry PWM - *HW Config/CPU Count Properties/Pulse-width modulation*.

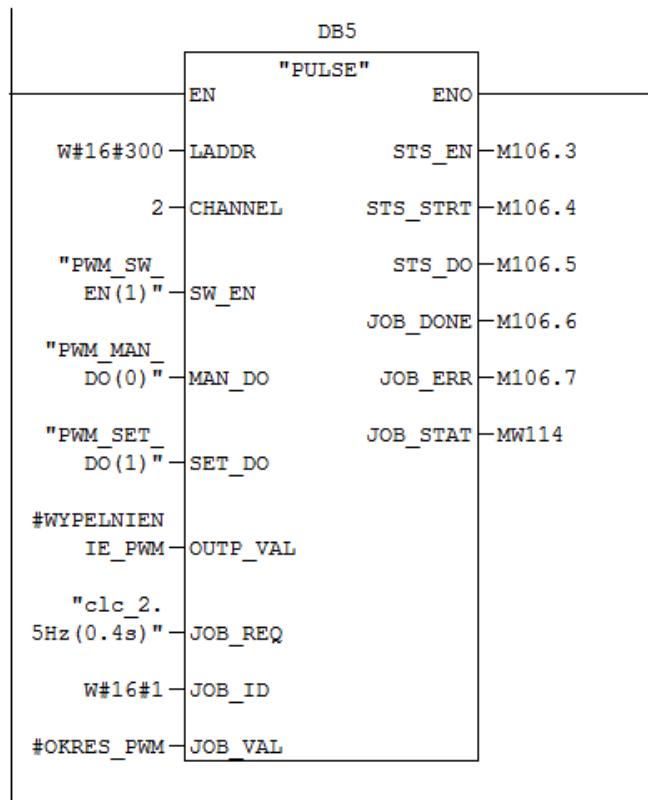
Parametry PWM (Rys. 4.15.):

- **Output format** (Format wyjściowy) – zakres wartości wyjściowych *Per mil* lub *S7 analog value*:
 - *Per mil* – wartość wyjściowa PWM w zakresie od 0.000 do 1.000,
 - *S7 analog value* – wartość wyjściowa PWM w zakresie od 0 do 27648 (analogowa),
- **Time base** – określa rozdzielcość podstawy czasu *1ms* lub *0.1ms*. W układzie sterowania kulką na pochylni *Time base* określa mnożnik okresu sygnału PWM. *Time base = 0.1ms*, co znaczy że wartość okresu powinna być zdefiniowana następująco: **L#200 * 0.1ms = 20ms (50Hz)**,
- **On delay** – opóźnienie czasowe między początkiem generowania PWM a pojawiennia się pulsu.
- **Period** – okres generowanego sygnału PWM. W przypadku układu sterowania kulką na pochylni parametr ten w konfiguracji sprzętowej *HW Config* nie ma znaczenia ponieważ zostaje nadpisany przez systemową funkcję sterującą sygnałem PWM,
- **Minimum pulse duration** – minimalny czas trwania pulsu; tłumienie za krótkich implusów. Minimalny czas trwania pulsu dla układu sterowania jest nie mniejszy jak 0.2.



Rys. 4.16. Sygnał szybkiego licznika PWM.

Do sterowania wypełnieniem pulsu służy funkcja systemowa *SFB 49 – PULSE* (Rys. 4.17) umieszczona w *Standard library/System Function Blocks/SFB49 PULSE TEC_FUNC*.



Rys. 4.17. Funkcja systemowa *SFB49 – PULSE*.

Ogólny opis parametrów funkcji SFB 49 umieszczono w tabeli 9, należy natomiast zwrócić szczególną uwagę na następujące parametry:

- **LADDR** – należy odczytać adres szybkiego licznika w *HW Config/CPU Counter Properties/Adresses*. Adres w *HW Config* wyrażony jest w postaci dziesiętnej, natomiast dla funkcji SFB 49 należy podać adres w postaci heksadecymalnej, więc dec(768)=hex(300),

- **CHANNEL** – numer kanału musi być zgodny z kanałem sparametryzowanym w *HW Config* (Rys. 4.15 – *Channel*),
- **OUTP_VAL** - dla *JOB_ID=W#16#1 OUTP_VAL* steruje wypełnieniem z zakresu od 0 do 26748 (według parametru szybkiego licznika w *HW Config* - *Output format* ustawionego na *S7 analog value*), gdzie 0 = 0% a 26748=100% wypełnienia pulsu.
- **JOB_REQ** – w celach badawczych na ten parametr podano *Clock Memory* aby sprawdzić zachowanie serwomechanizmu dla różnych częstotliwości (okresu) nadpisywanych w parametrze *JOB_VAL*.

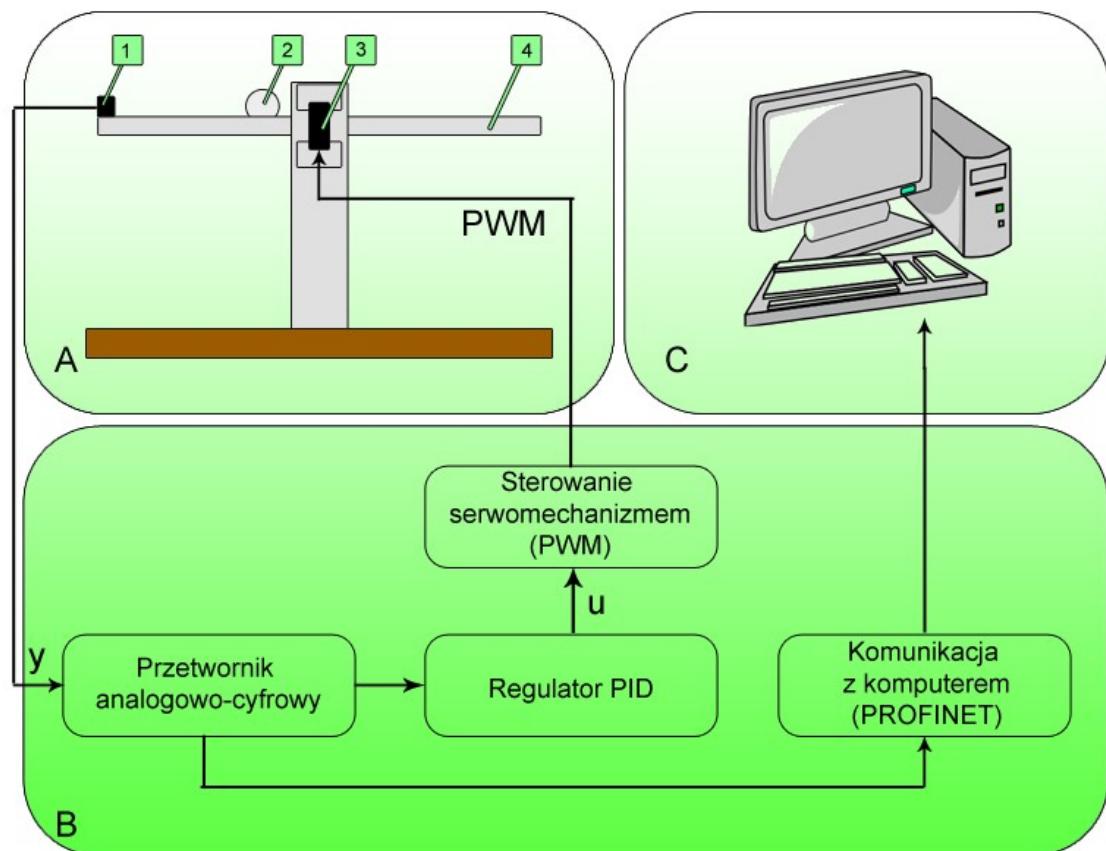
Funkcję SFB49 umieszczono w bloku OB35 wraz z regulatorem PID obie funkcje były wywoływanie z tym samym cyklem (podrozdział 5.2).

Tabela 9. Parametry funkcji *SFB 49 - PULSE*

SFB 49 - PULSE		
Parametr	Typ danych	Opis
Parametry wejściowe		
LADDR	WORD	Adres szybkiego licznika PLC odczytany z <i>HW Config</i>
CHANNEL	INT	Zarezerwowane wyjście dla szybkiego licznika w sterowniku PLC
SW_EN	BOOL	START/STOP wyjścia <i>PULSE</i>
MAN_DO	BOOL	Ustawianie ręcznej kontroli nad wyjściem
SET_DO	BOOL	Kontrola wyjścia
OUTP_VAL	INT	Ustawienie wartości wyjścia w zależności od pracy funkcji <i>JOB_ID</i> oraz formatu wyjścia ustawionego w <i>HW Config</i> – <i>Output format</i>
JOB_REQ	BOOL	Inicjacja <i>JOB_ID</i>
JOB_ID	WORD	Określa tryb pracy funkcji <i>PULSE</i> , 0-praca bez funkcji, 1 – sterowanie okresem i wypełnieniem , 4 – sterowanie minimalnym czasem trwania pulsu, 81 – odczyt okresu i wypełnienia, 84 – odczyt minimalnego czasu trwania pulsu
JOB_VAL	DINT	Wartość dla pracy <i>JOB_ID</i>
Parametry wyjściowe		
STS_EN	BOOL	Status wykonanej pracy
STS_STRT	BOOL	Status startu pracy
STS_DO	BOOL	Status wyjścia
JOB_DONE	BOOL	Informuje (1) gdy nowa praca może zostać wykonana
JOB_ERR	BOOL	Błąd pracy
JOB_STAT	WORD	Numer błędu pracy

5. PROGRAMOWANIE ALGORYTMÓW STEROWANIA CYFROWEGO

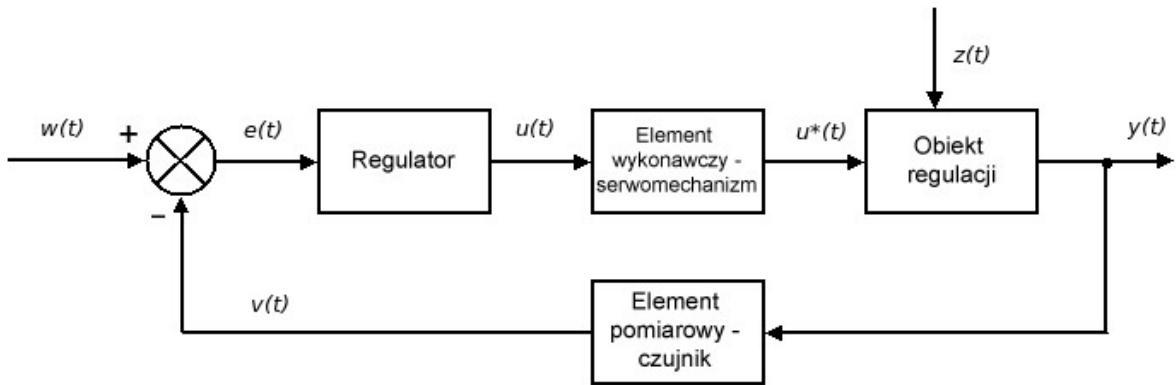
Układ sterowania położeniem kulki na pochylni podzielony został na trzy obszary działania (Rys. 45.). Pierwszym obszarem jest część mechaniczna zawierająca pochylnię umocowaną na wale osadzonym w łożyskach, czujnik optyczny mierzący położenie kulki oraz serwomechanizm jako element wykonawczy odpowiedzialny za przechylanie pochylni. Drugi obszar zawiera elementy przetwarzające sygnały oraz sterujące przez sterownik PLC. W obszarze trzecim znajduje się stacja robocza PC odpowiedzialna za diagnostykę i parametryzowanie układu sterowania.



Rys. 5.1. Ideowy schemat układu sterowania, gdzie:

A – kulka na pochylni; B – układ przetwarzania sygnałów, wyspa we/wy, sterownik PLC; C – komputer PC; 1 – optyczny czujnik odległości; 2 – kulka; 3 – serwomechanizm; 4 – pochylnia, y – wielkość regulowana, u – sygnał sterujący.

W schemacie układu ideowego (Rys. 5.1) wyraźnie widać pętlę ze sprzężeniem zwrotnym. Układ ten można przedstawić za pomocą schematu funkcjonalnego (Rys. 46.).



Rys. 5.2. Schemat funkcjonalny układu automatycznej regulacji kulki na pochylni, gdzie:

$w(t)$ – wartość sygnału zadanego, $e(t)$ - uchyb, $u(t)$ – wartość sygnału sterowania, $u^*(t)$ – wymuszenie, $y(t)$ wartość sygnału wyjściowego, $v(t)$ – wartość sygnału zwrotnego, $z(t)$ – zakłócenia.

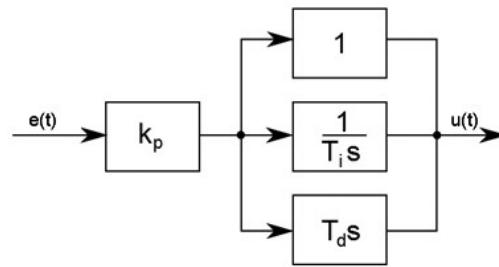
Układ automatycznej regulacji (Rys. 5.2) składa się z elementu porównującego (sumator), regulatora, elementu wykonawczego, obiektu sterowania oraz układu pomiarowego. Sumator oblicza różnice pomiędzy sygnałem $w(t)$ a sygnałem $v(t)$ otrzymaną poprzez układ sprzężenia zwrotnego z sygnały $y(t)$ otrzymaną z układu pomiarowego w sterowanym obiekcie. Otrzymany w układzie sumującym sygnał $e(t)$ jest przekazywany do regulatora, który przekształcając go w sygnał sterowania $u(t)$ do elementu wykonawczego, który oddziałuje na obiekt podając sygnał na jego wejście $u^*(t)$. Na regelowany obiekt mogą działać zakłócenia $z(t)$ (np. nagłe zmiany oświetlenia).

[12,14]

5.1. Algorytmy różnicowe regulatora PID dla potrzeb sterowania PLC

Regulator użyty w projekcie jest regulatorem PID (ang. *Proportional Integral Derivative*), który składa się z następujących członów (Rys. 5.3) [13,14]:

- proporcjonalnego P o wzmacnieniu k_p ,
- całkującego I o czasie zdwojenia T_i (stała całkowania),
- różniczkującego D o czasie wyprzedzenia T_d (stała różniczkowania) oraz stałej czasowej inercji T (opóźnienie czasu członu różniczkującego).



Rys. 5.3. Schemat regulatora PID.

Regulator PID realizuje algorytm:

$$u(t) = k_p \left[e(t) + \frac{1}{T_i} * \int_0^t e(\tau) d\tau + T_d * \frac{d}{dt} e(t) \right] + u_0$$

Transmitancja operatorowa idealnego regulatora PID wygląda następująco:

$$G_{PID}(s) = k_p \left[1 + \frac{1}{T_i s} + s * T_d \right]$$

Idealne różniczkowanie jest nierealizowalne fizycznie. Transmitancja operatorowa rzeczywistego regulatora PID wygląda następująco:

$$G_{PID}(s) = k_p \left[1 + \frac{1}{T_i s} + \frac{T_d s}{Ts + 1} \right]$$

Porównując parametry regulatora w programie sterowania do algorytmu regulatora PID:

- $k_p = GAIN$ – wzmacnianie,
- $T_i = TI$ – stała całkowania,
- $T_d = TD$ – stała różniczkowania,
- $T = TM_LAG$ – stała czasowa inercji
- $e(t) = SP - PV$ – uchyb (SP – wartość zadana, PV – procesowy sygnał wejściowy),
- $u(t) = LMN$ – sygnał sterujący,

otrzymujemy algorytm regulacji systemowego regulatora PID w programie:

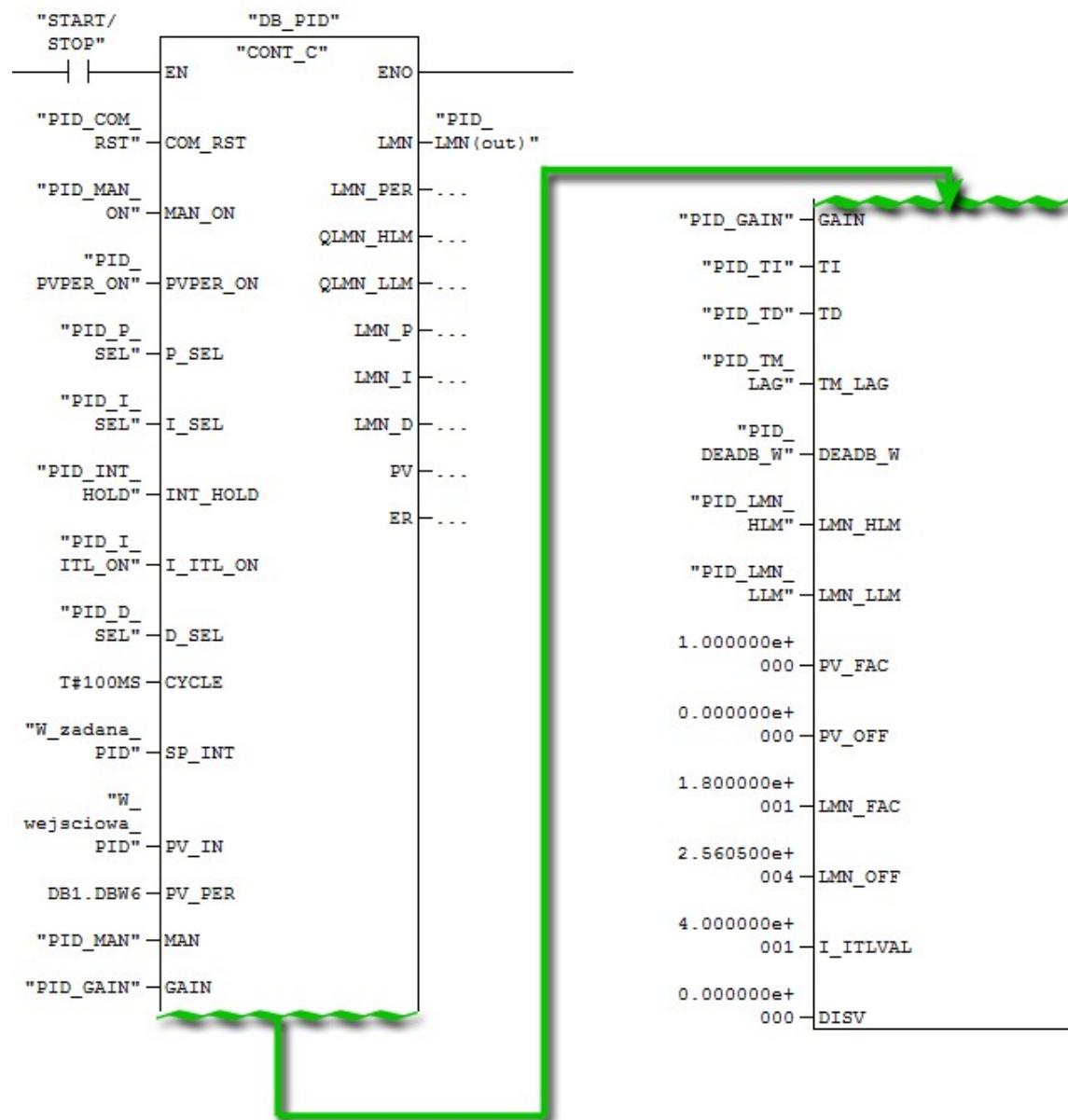
$$LMN = GAIN \left[(SP - PV) + \frac{1}{TI} * \int_0^t (SP - PV) d\tau + TD * \frac{d}{dt} (SP - PV) \right] + LMN_0$$

Natomiast transmitancja operatorowa systemowego regulatora PID w programie przyjmuje postać:

$$G_{PID}(s) = GAIN \left[1 + \frac{1}{TI * s} + TD * \frac{TD s}{TM_LAG s + 1} \right]$$

5.2. Funkcje do obsługi PID w PLC

Do realizacji algorytmu regulatora PID w układzie sterowania wykorzystano systemowy blok funkcyjny *FB 41 – CONT_C* (Rys. 5.4). Jest on używany do sterowania kulką na pochylni dla znanej wielkości zadanej (*SP_INT*) i procesowej wartości wejściowej (sygnał sprzężenia zwrotnego *PV_IN* pochodzący z przetwornika mierzącego wartość sygnału wyjściowego czujnika po linearyzacji). Dla dużych zakłóceń jest on przystosowany, wykorzystując wejście *DISV*, do sprzężenia kompensacyjnego (do przodu) od mierzzonego zakłócenia (nieużywany w projekcie). Ponadto ma on opcje ręcznego sterowania wyjściem *MAN_ON* [4,6].



Rys. 5.4. Funkcja *FB 41 - CONT_C*

Dla prawidłowej pracy bloku FB41, należy go koniecznie umieścić w bloku przerwań cyklicznych OB35. Przerwania cykliczne (*ang. Cyclic Interrupts*) wykorzystywane są do przetwarzania pewnych fragmentów programu co określony, stały czas. W niektórych przypadkach cykl pracy głównego bloku OB1 będzie się zmieniać, co spowodowałoby że działanie regulatora PID będzie się wachać, a w najgorszym wypadku kontrolowałby zmienną wyjściową w niestabilny sposób. Należy zwrócić szczególną uwagę aby parametr bloku OB35 określający cykl wywoływania (Rys. 2.18) był zgodny z cyklem wywoływania bloku *FB41 – CYCLE*, w przeciwnym wypadku cykl OB35 i cykl FB41 nie będą zsynchronizowane co doprowadziłoby do niestabilnej pracy regulatora.

https://www.automation.siemens.com/mcms/sce/de/fortbildung/ausbildungsunterlagen/download_ausbildungsunterlagen/b_weiterfuehrende_funktionen_step7_programmierung/Documents/EN_b03_controlengineering.pdf?HTTPS=REDIR , page 46]

Tabela 10. Parametry funkcji *FB 41 - CONT_C*

FB 41 CONT_C		Opis
Parametr	Typ danych	
Parametry wejściowe		
COM_RST	BOOL	sygnał odpowiedzialny za restart regulatora
MAN_ON	BOOL	aktywacja sygnału ręcznego sterowania
PVPER_ON	BOOL	dla TRUE aktywacja zmiennoprzecinkowa procesowych urządzeń peryferyjnych; jeżeli zmienna procesu jest odczytywana z urządzenia I/O, wejście PV_PER musi być podłączone do I/O. Sygnał sprzężenia zwrotnego będzie wtedy skalowany. Dla FALSE sygnał od razu w postaci liczby zmiennoprzecinkowej
P_SEL	BOOL	aktywacja członu proporcjonalnego
I_SEL	BOOL	aktywacja członu całkującego
INT_HOLD	BOOL	wstrzymanie członu całkującego; dla TRUE wyjście tego członu będzie nieaktywne
I_ITL_ON	BOOL	inicjacja członu całkującego
D_SEL	BOOL	aktywacja członu różniczkującego
CYCLE	TIME	cykl wywołania bloku FB41
SP_INT	REAL	zmienna, która jest wartością zadaną
PV_IN	REAL	zmiennoprzecinkowa wartość wejściowa
MAN	REAL	wprowadzana wartość wejścia podczas sterowania ręcznego
GAIN	REAL	wartość wzmacniania
TI	TIME	stała całkowania
TD	TIME	stała różniczkowania
TM_LAG	TIME	opóźnienie członu różniczkującego
DEADB_W	REAL	szerokość strefy martwej
LMN_HLM	REAL	górnna granica wartości sterującej (wartości wyjścia)

LMN_LLM	REAL	dolna granica wartości sterującej (wartości wyjścia)
PV_FAC	REAL	tutaj znajduje się mnożnik funkcji skalującej; opcjonalnie 1. Parametr jest stosowany do dopasowania zakresu zmienności sygnału z przetwornika pomiarowego do zakresu zmienności wartości zadanej SP_INT
PV_OFF	REAL	wejście przesuwające funkcję skalującą w górę i w dół; opcjonalnie 0
LMN_FAC, LMN_OFF	REAL	analitycznie jak PV, stosuje się je do dopasowania sygnału wyjściowego z regulatora do sygnału potrzebnego w obiekcie
I_ITL_VAL	REAL	wartość inicjacji działania członu całkującego
DISV	REAL	wejście dla sygnału zakłócającego
Parametry wyjściowe		
LMN	REAL	zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora
LMN_PER	REAL	sygnał wyjściowy regulatora na wejścia peryferyjne
QLMN_HLM	BOOL	górna granica wartości skalującej
QLMN_LLM	BOOL	dolna granica wartości skalującej
LMN_P	REAL	komponent proporcjonalny
LMN_I	REAL	komponent całkujący
LMN_D	REAL	komponent różniczkujący
PV	REAL	zmienna procesowa
ER	REAL	sygnalizacja o błędzie

Zmienną procesową może być wejście z urządzenia peryferyjnego I/O (przetwornika pomiarowego w torze sprzężenia zwrotnego) – wtedy parametr PVPER_ON jest TRUE lub jeżeli jest od razu w postaci zmiennoprzecinkowej to PVPER_ON jest FALSE. Funkcja CRP_IN (Rys. 50.) konwertuje wartość z urządzenia peryferyjnego PV_PER na postać liczby zmiennoprzecinkowej z przedziału -100...100% zgodnie ze wzorem:

$$\text{Wyjście } CRP_{IN} = PV_{PER} * \frac{100}{27648}$$

Funkcja PV_NORM normalizuje (skaluje) wartości z CRP_IN zgodnie ze wzorem:

$$\text{Wyjście } PV_{NORM} = \text{Wyjście } CRP_{IN} * PV_{FAC} + PV_{OFF}$$

gdzie:

PV_FAC – ten parametr zmienia nachylenie prostej skalującej, domyślnie ma ustawioną wartość 1, czyli nie zmienia skali.

PV_OFF – powoduje przesunięcie skalowania w górę i w dół, domyślnie ma ustawioną wartość 0, czyli nie przesuwa skali.

W projekcie wartość PV jest w postaci zmiennoprzecinkowej (sygnał z czujnika po linearyzacji).

Do regulatora jest wprowadzany sygnał błędu ER, określony w węźle sumacyjnym, gdzie wyznaczana jest różnica między wartością zadaną SP_INT i zmienną procesową po normalizacji (skalowaniu) PV. W celu wyeliminowania małych oscylacji wynikających np. wahania się kulki na pochylni w okolicy wartości zadanej, wprowadza się człon DEADBAND określający martwą strefę.

Regulator PID w bloku funkcji FB41 CONT_C korzysta z trójczłonowego algorytmu pozycyjnego. Każdy z członów może być aktywowany i dezaktywowany indywidualnie. To pozwala na tworzenie regulatorów P, I, D, PI, PD i PID.

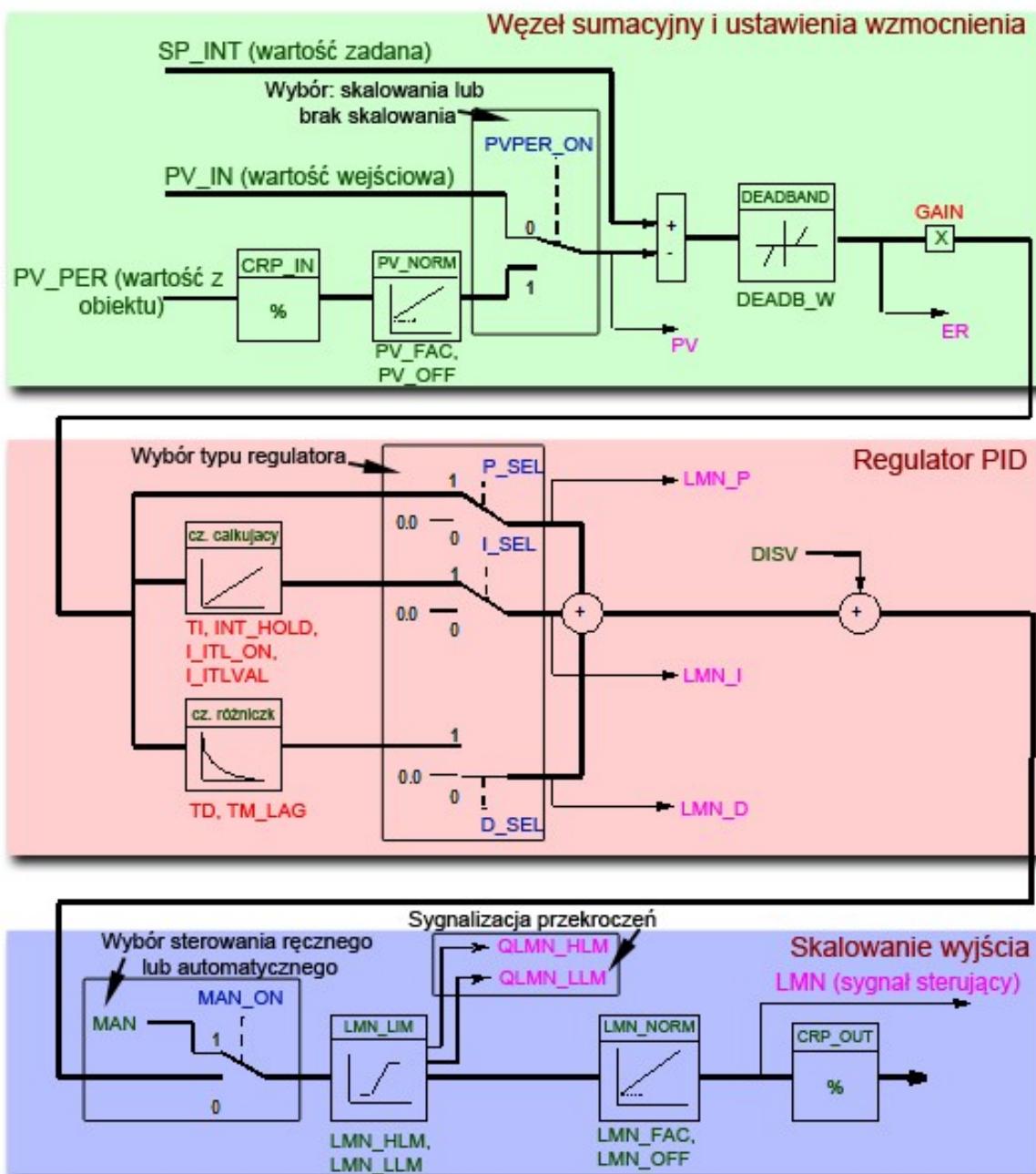
W trybie ręcznym , zmienna sterująca wprowadzana jest ręcznie przez wejście MAN. Wewnętrznie człon całkujący jest ustawiony na LMN – LMN_P – DISV, natomiast człon różniczkujący na 0, zaletą jest łagodne przełączanie do trybu automatycznego. **W projekcie nie używa się trybu ręcznego.**

Wartość sterująca (wyjściowa z regulatora) może być ograniczona przy użyciu funkcji LMN_LIM. Sygnalizujące bity QLMN_HLM i QLMN_LLM wskazują, kiedy zmienna wejściowa wyszła poza zakres. Funkcja LMN_NORM skaluje wyjście LMN_LIM zgodnie ze wzorem:

$$LMN = LMNLIMIT * LMN_FAC + LMN_ON$$

Blok FB41 umożliwia wykonanie kompletnego restartu, gdy parametr wejścia COM_RST jest ustawiony. Podczas restartu regulatora człon całkujący jest ustawiony wewnątrz do wartości inicjacji I_ITVAL. Wszystkie inne wyjścia są ustawione zgodnie z nastawieniami domyślnymi.

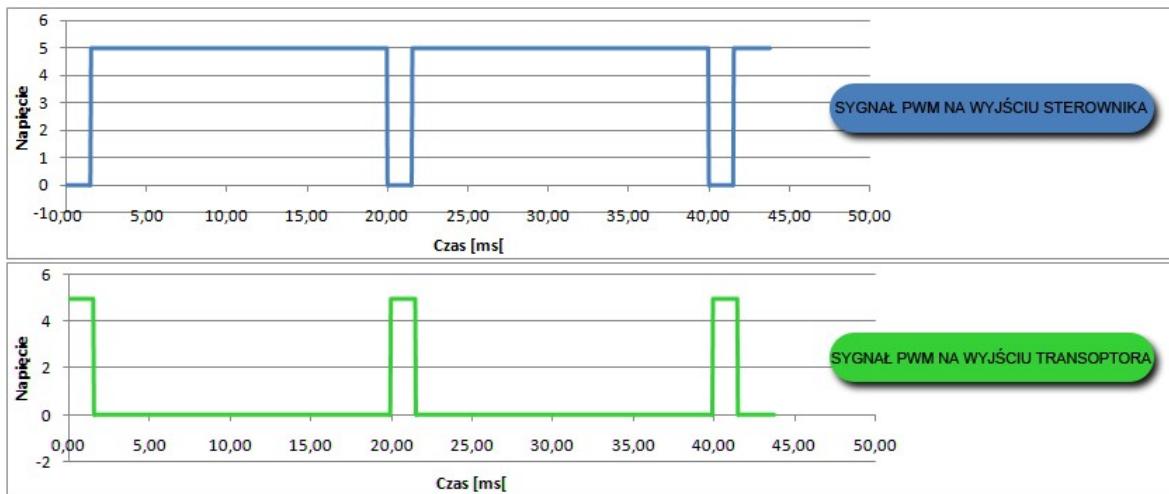
Szczegółowy schemat obsługi bloku FB41 umieszczono na rysunku 50.



Rys. 5.5. Szczegółowy schemat obsługi bloku FB41 - CONT_C.

Ograniczenie wartości wyjściowej regulatora:

W układzie sterowania kulką na pochylni puls sygnału PWM sterującego serwomechanizmem wychodząc z wyjścia szybkiego licznika PLC osiąga napięcie 24V, dla serwomechanizmu natomiast sygnał sterujący musi osiągać wartość ok 5V. Za przełożenie napięcia z 24V na 5V odpowiedzialny jest transoptor umieszczony w konsoli sterującej. W wyniku przełożenia napięcia sygnał PWM zostaje odwrócony, co się wiąże z generowaniem sygnału odwrotnie do wymaganego sygnału sterującego serwomechanizmem. Rozwiążanie problemu objaśnia rysunek 5.6.



Rys. 5.6. Porównanie sygnału PWM na wyjściu sterownika i wyjściu transoptora

W związku z parametrem *Output format* (Rys. 4.15) ustawionym na *S7 analog value* sterowanie wypełnieniem pulsu sygnału PWM odbywa się następująco:

- wartość wejściowa *SFB49 (OUTP_VAL)* 0 = 0% wypełnienia pulsu PWM,
- wartość wejściowa *SFB49 (OUTP_VAL)* 27648 = 100% wypełnienia pulsu PWM.

W układzie sterowania kulką na pochylni dla łagodnej regulacji wychylenia powinny mieścić się w zakresie :

- maximum + **10°** dla poziomego położenia lewego pochylni = ok. **25785** wartości wejściowej *SFB49 (OUTP_VAL)*,
- **0°** dla położenia poziomego pochylni – ok. **25605** wartości wejściowej *SFB49 (OUTP_VAL)*,
- minimum - **10°** dla poziomego położenia prawego pochylni = ok. **25425** wartości wejściowej *SFB49 (OUTP_VAL)*.

Za ograniczenie wartości sterującej w regulatorze PID (FB41) odpowiedzialne są parametry *LMN_HLM* oraz *LMN_LLM*. Aby ograniczać wychylenie pochylni w stopniach, użyto skalowania za pomocą parametrów *LMN_FAC* oraz *LMN_OFF*.

Skoro **-10°** wychylenia pochylni to **25425** wartości wejściowej generatora PWM a **0° = 25605** to mnożnik funkcji skalującej *LMN_FAC* = 18 ($25605 - 25425 = 18 * 10^\circ$).

Z kolei parametr *LMN_OFF* przesuwający funkcję w górę i w dół wynosi 25605, przykładowo:

$$LMN = LMNLIMIT * LMN_FAC + LMN_ON$$

$$25425 = -10^\circ * 18 + 25605$$

Dlatego dzięki funkcji skalującej LMN_NORM parametry ograniczające wartość wyjściową regulatora tj. LMN_HLM i LMN_LLM wynoszą +/- **10°**.

Restart regulatora :

Ten sam blok regulatora PID umieszczono również w bloku OB100 w celu jego restartu. Blok OB100 wykonuje się przed innymi blokami OB podczas uruchamiania się sterownika (przejście w tryb pracy RUN). W bloku regulatora PID umieszczonego w OB100 wykorzystuje się tylko jeden parametr: COM_RST odpowiedzialny za restart PID, jest on ustawiany na TRUE w pierwszej linii programu OB100 a w następnej na FALSE. Taka operacja zapewnia restart regulatora podczas uruchamiania się sterownika i ma na celu wyzerowanie zapamiętanych wyników członu różniczkującego z regulacji przed ponownym uruchomieniem sterownika.

W tabeli 11 umieszczono opis parametryzacji bloku FB41 – *CONT_C* z perspektywy układu sterowania, gdzie pominięto nie używane parametry.

Tabela 11. Parametryzacja używanych w układzie sterowania funkcji bloku FB41 - *CONT_C*.

FB 41 CONT_C		Opis
Parametr	Wartość	Parametry wejściowe
COM_RST	TRUE / FALSE	w bloku OB100 wykonuje się sekwencja: TRUE a następnie FALSE w celu restartu regulatora po ponownym restarcie regulatora
PVPER_ON	FALSE	ponieważ wejście sygnału procesowego jest wyrażona w postaci zmiennoprzecinkowej
P_SEL	TRUE	ponieważ wykorzystuje się człon proporcjonalny
I_SEL	TRUE	ponieważ wykorzystuje się człon całkujący
D_SEL	TRUE	ponieważ wykorzystuje się człon różniczkujący
CYCLE	100 ms	cykl wywołania bloku FB41 musi być taki sam jak cykl wywołania bloku OB35 w którym zawarty jest FB41 (musi zachodzić synchronizacja)
SP_INT	10.0...70.0	w zależności od wartości zadanej (pożądana pozycja kulki na pochylni). Zadawanie wartości SP odbywa się za pomocą przycisków konsoli sterującej lub w czasie rzeczywistym w tablicy VAT (rozdział 6)
PV_IN	8.0...72.0	zmienna wartość z procesu (położenie kulki odczytane z czujników po linearyzacji)
MAN	0.0	wartość MAN nie ma znaczenia ponieważ w projekcie nie używa się sterowania ręcznego
GAIN	0.01...10	wartość wzmacnienia zależna jest od pozostałych parametrów regulatora. Należy dążyć do jak najmniejszego wzmacnienia aby nie uszkodzić serwomechanizmu. Zbyt duże wzmacnienie powoduje gwałtowne zmiany położenia pochylni.

TI	0...∞ s	im większa stała czasu całkowania tym mniejszy wpływ członu całkującego
TD	0...30 s	im mniejszy czas stałej różniczkowania tym mniejszy wpływ członu różniczkującego na obiekt. Przy zwiększaniu należy stopniowa zmniejszać wzmocnienie ponieważ zbyt duża wartość stałej różniczkowania powoduje dynamiczne oscylacje pochylni.
TM_LAG	0...500 ms	im większe opóźnienie to odstępy między działaniami członu różniczkującego większe. Zwiększenie w efekcie powoduje delikatne wychylenia równi
DEADB_W	1.0..1.5	szerokość strefy martwej powinna mieścić się w tym zakresie. Poniżej 1.0 regulator będzie próbował ciągle stabilizować kulkę mimo bardzo bliskiego położenia kulki w pożądanej pozycji, powyżej 1.5 powstaje uchyb statyczny.
LMN_HLM	< 10	górną granicą wartości sterującej nie powinna przekraczać 10 stopni ! zwiększenie spowoduje zbyt duży kąt wychylenia równi w prawą stronę.
LMN_LLM	> -10	analogicznie jak wyżej
LMN_FAC	18	$10 * \mathbf{18} = 180$
LMN_OFF	25605	$180 + \mathbf{25605} = 25805$ (10 stopni)
Parametry wyjściowe		
LMN	25425...25805	wyjście regulatora określa szerokość pulsu dla generatora PWM. -10 stopni wychylenia pochylni dla 25425, +10 dla 25805

6. TESTOWANIE ALGORYTMÓW STEROWANIA

W celu wyznaczenia charakterystyk czasowych w odpowiedzi układu sterowania na wymuszenia oraz dobór parametrów regulatora PID wykorzystano trzy aplikacje w tym dwa z pakietu narzędziowego *Step 7* [4,6]:

- ***Step7 Variable Table*** – umożliwia monitorowanie i modyfikowanie zmiennych,
- ***Step7 PID Control*** – umożliwia parametryzację regulatora PID oraz graficzny podgląd do 4 parametrów regulatora,
- ***AutoSpy (DEMO)*** – umożliwia graficzny i tabelaryczny podgląd dowolnych zmiennych zimportowanych bezpośrednio ze sterownika oraz eksport tych danych do pliku .csv (*Excel*). Wersja *DEMO* ogranicza możliwości programu do dwóch zmiennych.

Step7 Variable Table:

	Address	Symbol	Display format	Status value	Modify value			
1	// ODCZYT Z CZUJNIKOW							
2	MD 56	"X_REAL1"	FLOATING_POL...	54.55516				
3	MD 60	"X_REAL2"	FLOATING_POL...	55.21329				
4	MD 210		FLOATING_POL...	54.88422				
5	MD 64	"W_wejsciowa_PID"	FLOATING_POL...	54.55516				
6	MD 22	"analog1_VAL"	FLOATING_POL...	1.132813				
7	MD 28	"analog2_VAL"	FLOATING_POL...	2.340133				
8								
9	// PARAMETRY REGULATORA PID							
10	MD 118	"W_zadana_PID"	FLOATING_POL...	55.0				
11	MD 126	"PID_GAIN"	FLOATING_POL...	4.5	4.5			
12	MD 130	"PID_TI"	TIME	T#5s	T#5s			
13	MD 134	"PID_TD"	TIME	T#800ms	T#800ms			
14	MD 138	"PID_TM_LAG"	TIME	T#200ms	T#200ms			
15	MD 142	"PID_DEADB_W"	FLOATING_POL...	1.0	1.0			
16	MD 146	"PID_LMN_HLM"	FLOATING_POL...	10.0	10.0			
17	MD 150	"PID_LMN_LLM"	FLOATING_POL...	-10.0	-10.0			
18	MD 154	"PID_LMN(out)"	FLOATING_POL...	25617.95				
19	M 116.0	"PID_COM_RST"	BOOL	false				
20								

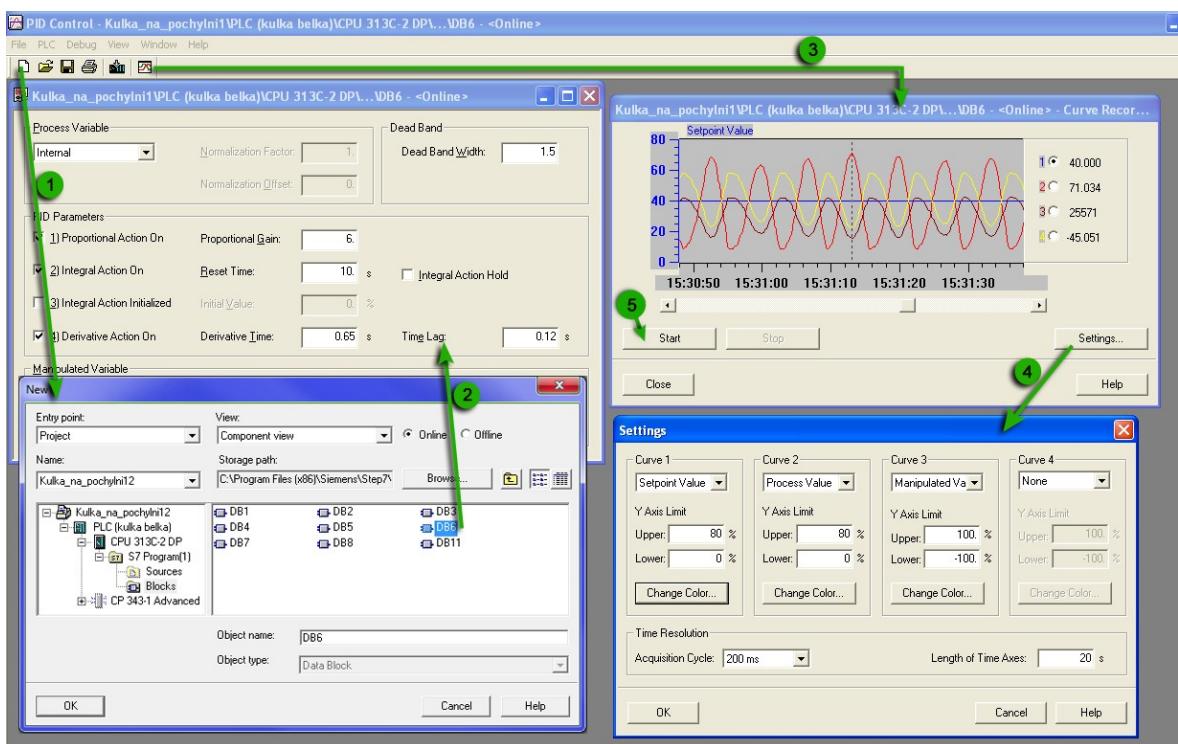
Rys. 6.1. Tablica VAT - monitorowanie i modyfikowanie zmiennych, gdzie:

1 – komentarz, 2 - wyłączenie podglądu ,3 - włączanie modyfikacji zmiennych, 4 - kolumna z adresami pamięci, 5 - kolumna z symbolami zmiennych, 6 - kolumna z typem zmiennych, 7 - kolumna - podgląd online zmiennych, 8 - kolumna z wartością modyfikującą, 9 - podgląd online trybu pracy sterownika.

Monitorowanie i modyfikowanie zmiennych (Rys. 6.1) jest narzędziem, które umożliwia obserwowanie i/lub wysterowanie zmiennych z określonych obszarów pamięci systemowej CPU, w tym również zmiennych występujących w programie użytkowym, w wybranych formach. Monitorowanie zmiennych odbywa się w trybie *online* po aktywacji opcji *Monitor variable* (Rys. 6.1 ozn. 2) natomiast modyfikowanie przebiega w dwóch krokach: należy wprowadzić wartość w wierszu modyfikowanej zmiennej w kolumnie *Modify value* (Rys. 6.1 ozn. 8) a następnie aktywować opcję *Modify variable* (Rys. 6.1 ozn. 3).

Narzędzie *Variable Table* wykorzystano do zmiany parametrów regulatora w trybie *online* co znacznie przyśpieszyło dobór nastaw PID.

Step7 PID Control:

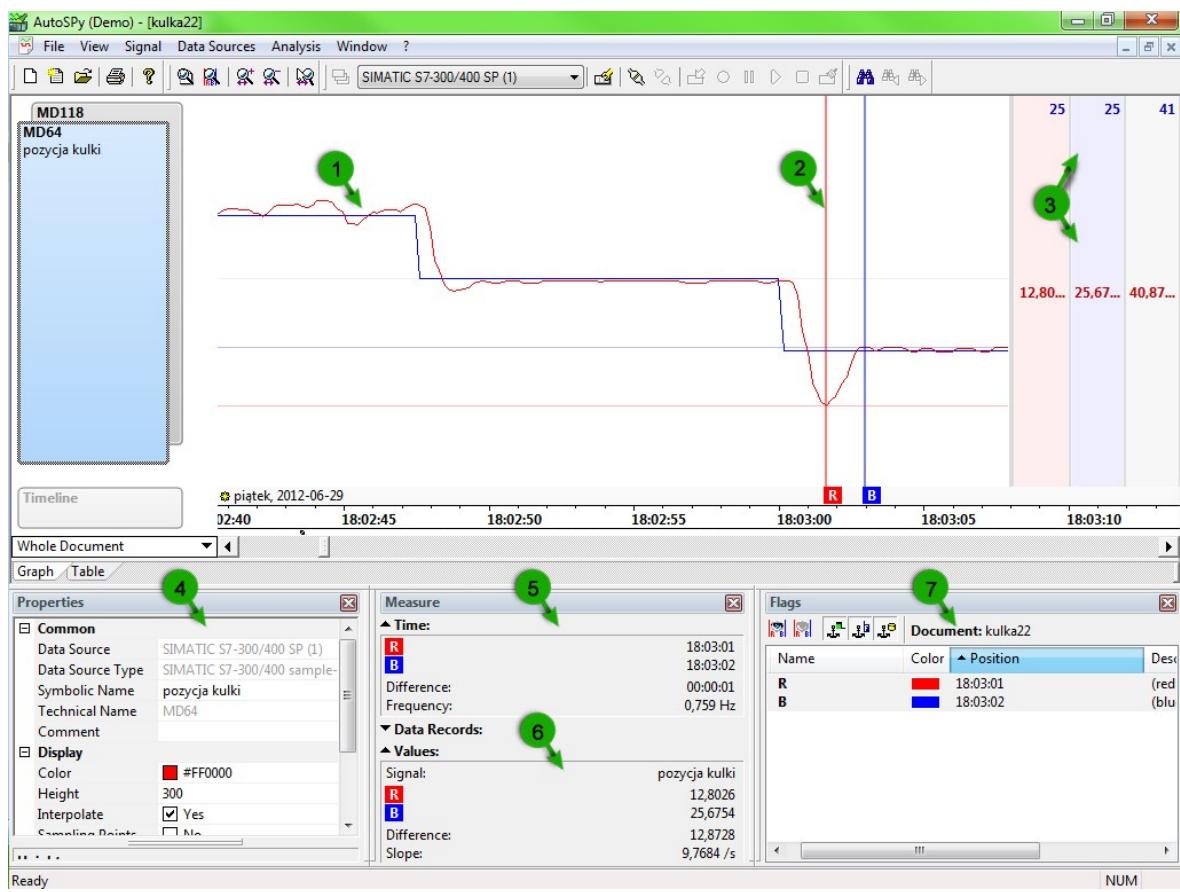


Rys. 6.2. Konfiguracja *PID Control*.

Aby uruchomić aplikację *PID Control* należy znaleźć ścieżkę do pliku *S7wriapx.exe*, przy standardowych ustawieniach Step7 powinna wyglądać następująco: *C:\Program Files SIEMENS\Step7\S7WRI\S7wriapx.exe*. Po uruchomieniu aplikacji należy wybrać zakładkę *File/New* w celu utworzenia nowego podglądu parametrów PID. W oknie tworzenia podglądu (Rys. 6.2 ozn. 1) należy w pierwszej kolejności wybrać odpowiednią nazwę aktualnie używanego projektu *Simatic Manager* w polu wyboru *Name*, następnie zaznaczyć opcję *Online* i odszukać w folderze blok danych z którego korzysta PID (DB6). Po zatwierdzeniu pojawi się okno do parametryzacji regulatora PID (Rys. 6.2 ozn. 2),

zmiana parametrów nastąpi po wybraniu opcji *PLC/Download*. Następnie należy wybrać otworzyć okno podglądu graficznego poprzez wybranie opcji *Debug/Curve* (Rys. 6.2 ozn. 4), wejść w ustawienia *Settings* (Rys. 6.2 ozn. 4). W ustawieniach można zdefiniować ile zmiennych ma generować okno podglądu (maksymalnie 4), zmienne te to: wartość zadana, wartość sygnału procesowego, wartość sterująca, wyjście członów P, I lub D oraz uchyb. Dla każdego sygnału należy zdefiniować ograniczenie osi Y w polu *Y Axis Limit* w zakresie zmian tego sygnału. Po zatwierdzeniu (OK) oraz aktywacji przycisku *Start* (Rys. 6.2 ozn. 5) podgląd zacznie generować wykresy czasowe wybranych parametrów regulatora PID w trybie *online*.

AutoSpy (DEMO):



Rys. 6.3. Aplikacja *AutoSpy (DEMO)*, gdzie:

- 1 – graficzny podgląd zmiennych, 2 – flaga, 3 – początkowe, flagowe i końcowe wartości sygnałów, 4 – parametry wyświetlania sygnałów, 5 – dane czasowe flagi, 6 – dane amplitudowe flagi, 7 – lista flag.

Aplikację *AutoSpy (DEMO)* wykorzystano w projekcie do archiwizacji danych w celu utworzenia charakterystyk czasowych za pomocą programu *Excel*. Oprócz tego *AutoSpy (DEMO)* posiada wiele innych bardzo przydatnych funkcji, np. wykorzystanie flag (Rys. 6.3 ozn. 2) do odczytania amplitudy i czasu (Rys. 6.3 ozn. 3,5,6) w wybranym punkcie sygnału lub eksport danych sygnałów do pliku .csv tylko z zakresu między flagami co. Właściwości te znacznie ułatwiały określenie jakości układu regulacji.

Ze względu na wysoki poziom skomplikowania konfiguracji aplikacji *AutoSpy (DEMO)* nie umieszczono w pracy dokładnego opisu konfiguracji. Dokładny opis w języku angielskim można znaleźć na stronie producenta programu [19].

6.1. Dobór nastaw regulatora

W celu doboru nastaw regulatora PID posłużono się metodą *Zieglera Nicholsa*, która oparta jest o pomiar parametrów oscylacji. Polega ona na [12,14]:

1. Wyłączeniu członu całkującego i różniczkującego – należy Ti ustawić na jak największą wartość natomiast Td i T na zero (TI = 10h, TD=0, TM_LAG=0).
2. Zwiększeniu wzmacnienia Kp (GAIN) do czasu, aż osiągnie się ostateczne wzmacnienie krytyczne Kkr, przy którym sygnał wyjściowy pętli zacznie oscylować niegasnąco.
3. Wzmocnienie Kkr i okres oscylacji Tkr wykorzystuje się następnie, by ustawić parametry zgodnie z tabelą 12.
4. Korygowanie parametrów poprzez strojenie ręczne.

Tabela 12. Metoda *Zieglera Nicholsa*.

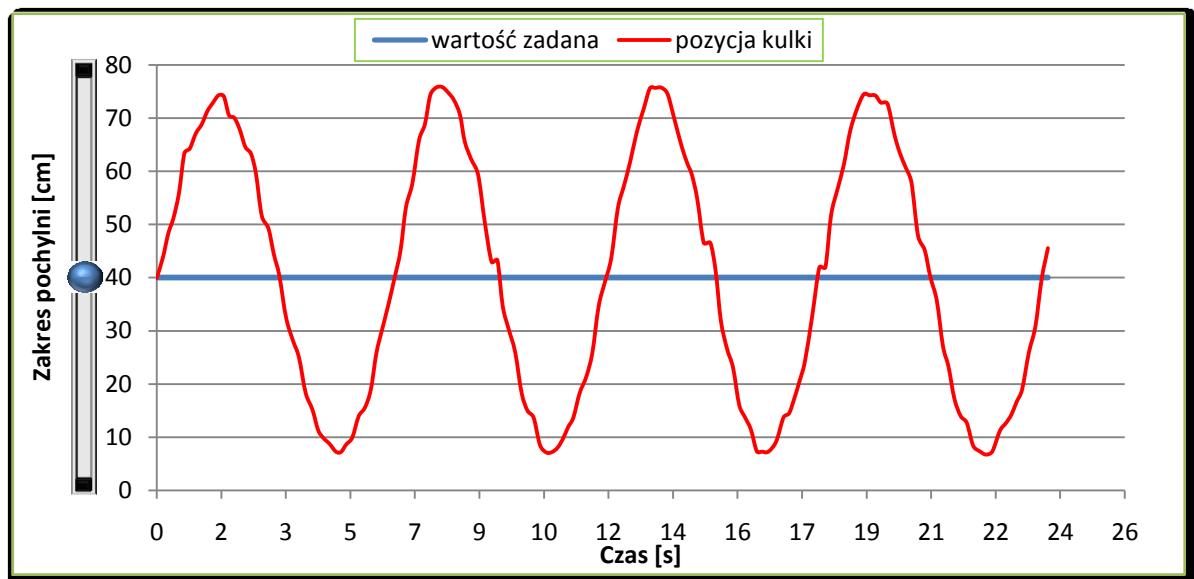
Typ regulacji	Kp (GAIN)	Ti (TI)	Td (TD)
P	0.5 Kkr	-	-
PI	0.45 Kkr	1.2 Tkr	
PID	0.6 Kkr	Tkr/2	Tkr/8

Zalety tej metody to brak wymogu identyfikacji dynamiki obiektu oraz to, że wyznaczone nastawy gwarantują (prawie zawsze) stabilność układu regulacji, choć nie gwarantują dobrych wskaźników jakości regulacji. W celu poprawy tych wskaźników można skorygować wyznaczone nastawy poprzez strojenie ręczne (metodą „prób i błędów”) [wiki].

Poniżej przedstawiono proces doboru nastaw parametrów regulatora PID metodą *Zieglera Nicholsa*:

Doprowadzenie do niegasnących oscylacji układu:

Do wyznaczenia ostatecznego wzmocnienia krytycznego K_{kr} zwiększano wzmocnienie członu proporcjonalnego K_p (GAIN) od 0.05 aż do osiągnięcia niegasących oscylacji sygnału wyjściowego regulatora, co osiągnięto przy $K_{kr} = 0.17$. Charakterystykę czasową przemieszczenia kulki na pochylni z niegasącymi oscylacjami przedstawia poniższy rysunek:



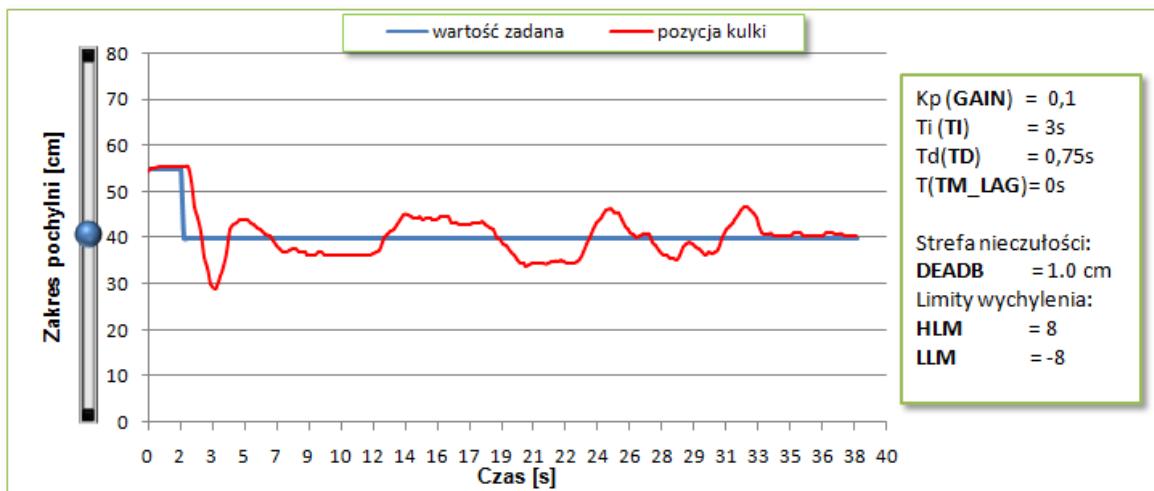
Rys. 6.4. Charakterystyka czasowo odległościowa przemieszczenia kulki na pochylni - wyznaczanie okresu oscylacji T_{kr} .

Z powyższego rysunku odczytując okres oscylacji $T_{kr} = \text{ok. } 6\text{s}$ oraz biorąc pod uwagę wzmocnienie krytyczne $K_{kr}=0.17$, wyznaczono parametry regulatora PID metodą *Zieglera Nicholsa*, które przedstawiono w poniższej tabeli:

Tabela 13. Dobór nastaw regulatora PID metodą Zieglera Nicholsa.

$K_{kr} = 0.1$ $T_{kr} = 6.4\text{s}$	K_p (GAIN)	T_i (T_l) [s]	T_d (T_d) [s]
Wzór	$0.6 K_{kr}$	$T_{kr}/2$	$T_{kr}/8$
Parametry PID	0.1	3	0.75

Odpowiedź układu zamkniętego na wymuszenie skokowe dla wyznaczonych parametrów metodą Zieglera Nicholsa regulatora przedstawia poniższy rysunek.

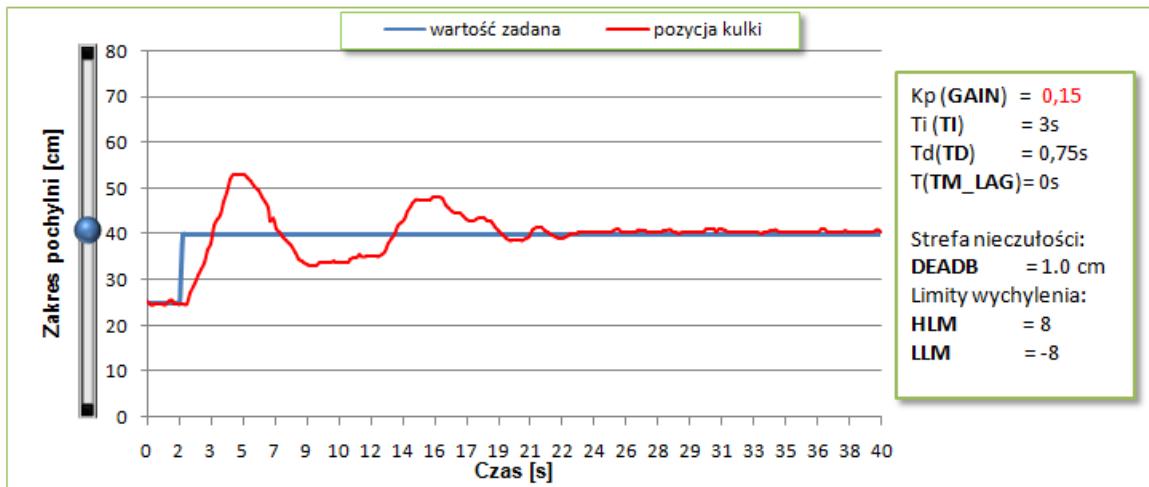


Rys. 6.5. Odpowiedź układu zamkniętego na wymuszenie skokowe.

Z rysunku 6.5 zauważać można że czas regulacji jest bardzo długi (ok. 32s).

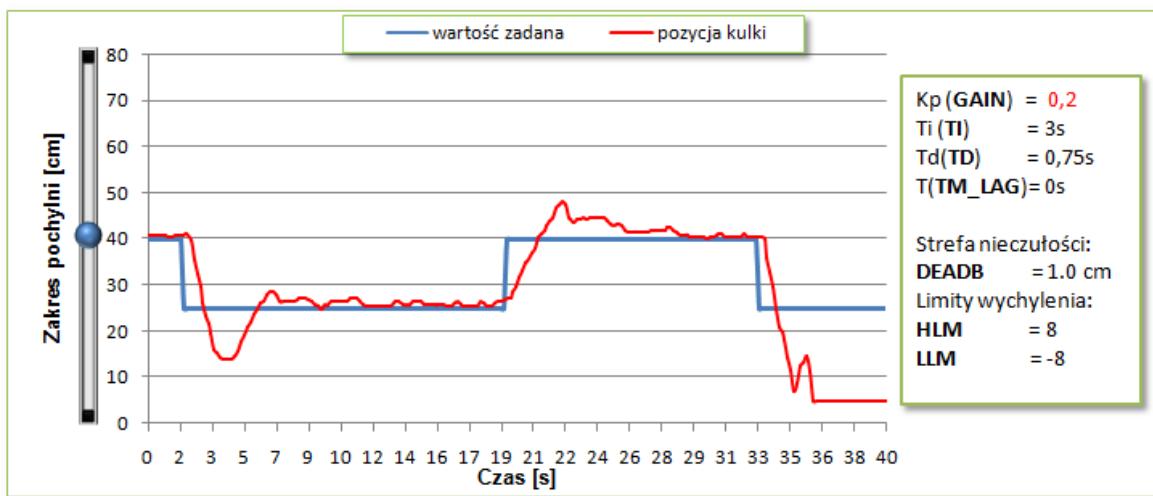
W następnych rysunkach przedstawione zostanie korygowanie parametrów PID wyznaczonych metodą Zieglera Nicholsa. Na rysunkach tych, w legendzie nastaw regulatora (po prawej stronie) kolorem czerwonym wyróżniono zmianę parametrów w stosunku do poprzedniego rysunku.

Odpowiedź układu zamkniętego na wymuszenie skokowe po zwiększeniu Kp, przedstawia poniższy rysunek.



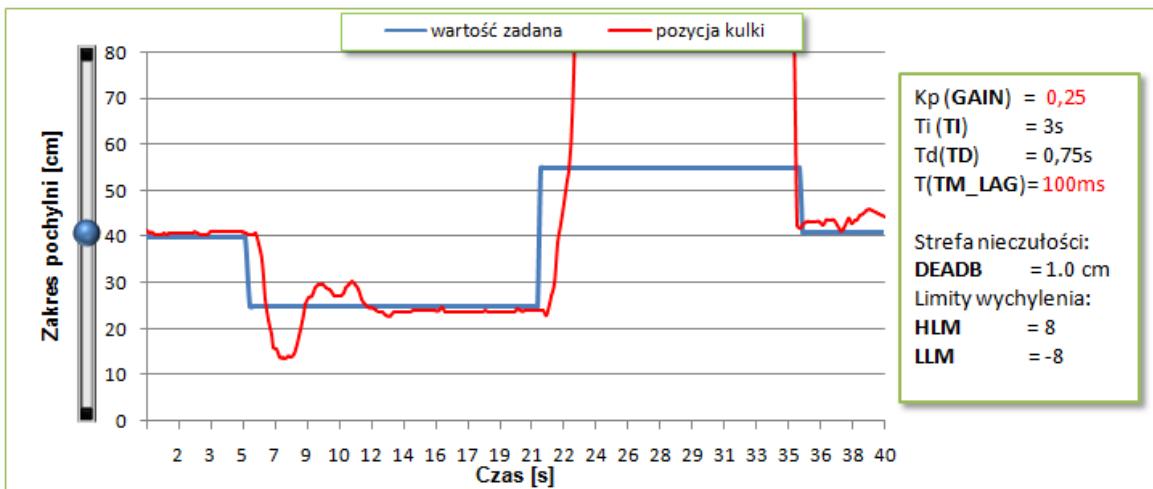
Rys. 6.6. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie Kp=0,15.

Jak widać z rysunku 6.6, po zwiększeniu Kp czas regulacji się skrócił do ok. 18s. Następna próba korygowania odbywa się poprzez kolejne zwiększenie wzmacnienia, co przedstawia rysunek 6.7.



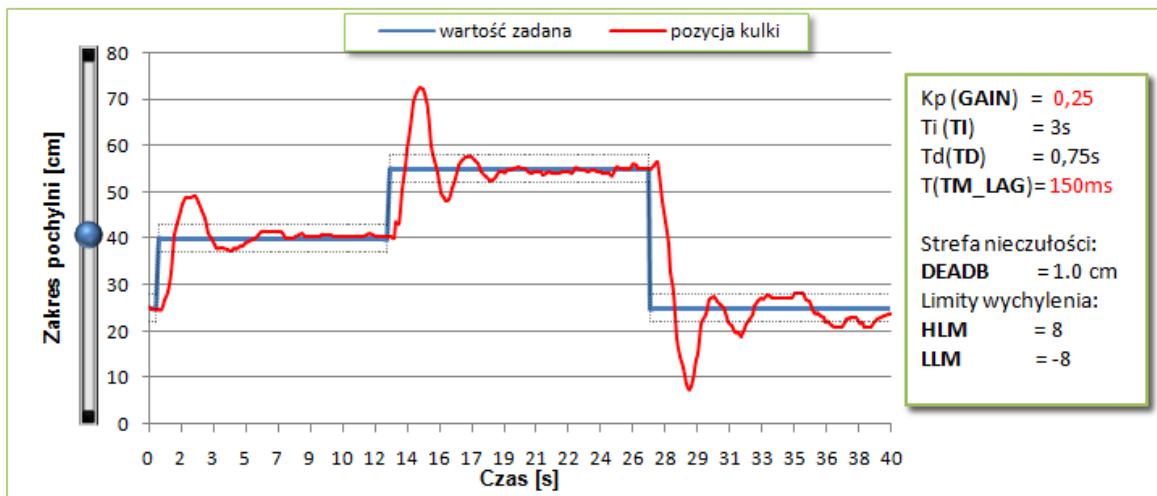
Rys. 6.7. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie $K_p=0,2$.

Z powyższego rysunku widać że czas regulacji znacznie się skrócił (do ok. 3s), lecz podczas ostatniego wymuszenia skokowego kulka została wytrącona z pochylni. Kolejna próba korygowania (Rys. 6.8) polega na dodaniu inercji T w celu opóźnienia członu różniczkującego, który wpływa na dynamikę zachowania pochylni co mogło być powodem wytrącenia kulki z pochylni.



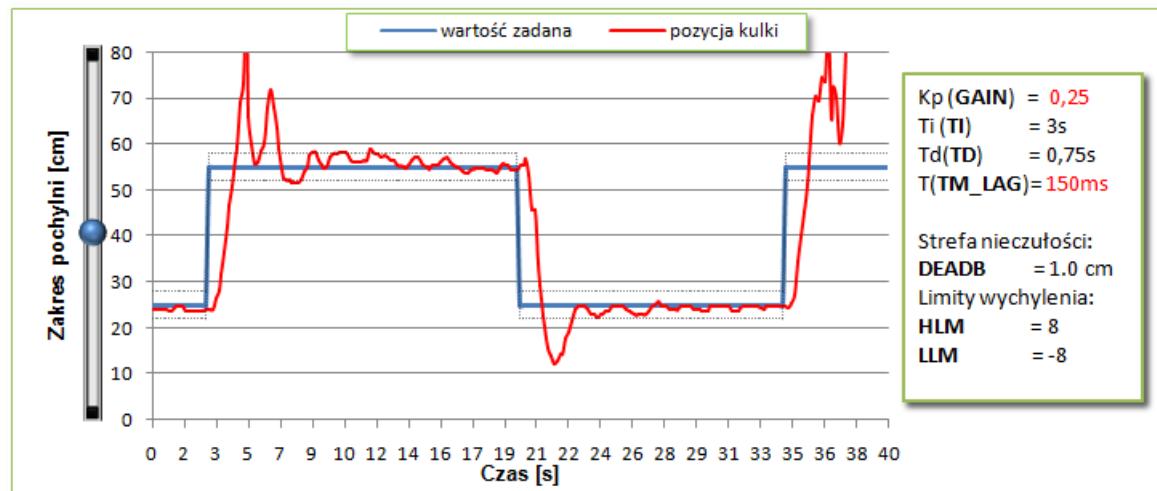
Rys. 6.8. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie $K_p=0,25$
oraz dodanie inercji $T=100ms$.

Jak przedstawia powyższy rysunek, mimo dodania inercji T kulka nadal zostaje wytrącona z pochylni. Kolejna próba korygowania polega na ponownym zwiększeniu inercji T, co przedstawia poniższy rysunek.



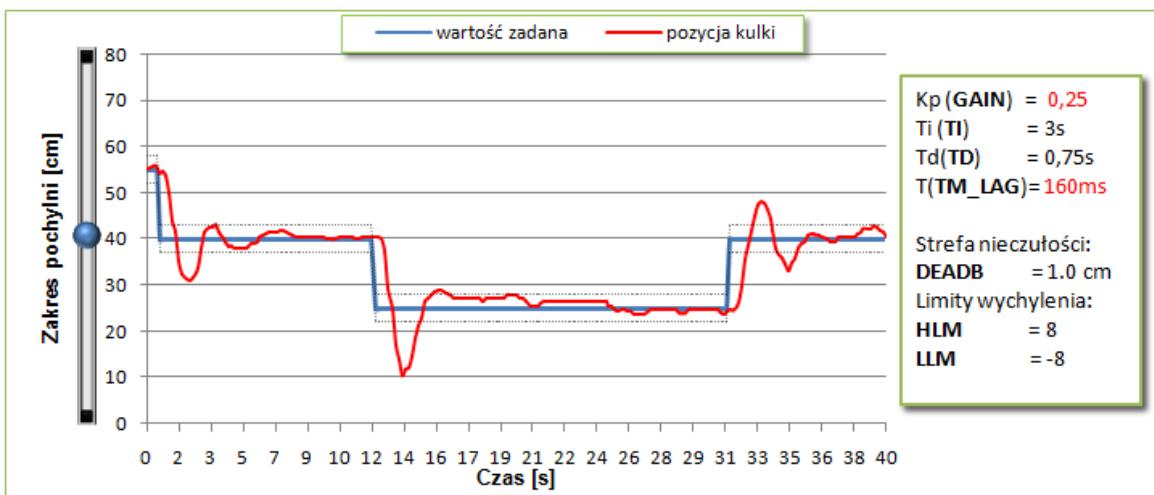
Rys. 6.9. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie $K_p=0,25$ oraz zwiększenie $T=150\text{ms}$; (1/2).

Po ponownym zwiększeniu inercji T układ regulacji jest w stanie ustabilizować kulkę na pochylni w żądanym miejscu. Jednak przy dużym wymuszeniu skokowym (o 30 cm), kulka zaczyna podskakiwać, czego nie da się zauważyć na wykresie ponieważ przy małych skokach kulki czujniki wciąż odczytują realne położenie kulki. W celu upewnienia się czy kulka nie zostanie wytrącona ponownie przy tych samych nastawach regulatora, przeprowadzono kolejne badanie z którego wyznaczoną odpowiedź układu zamkniętego na wymuszenie skokowe przedstawia poniższy rysunek.



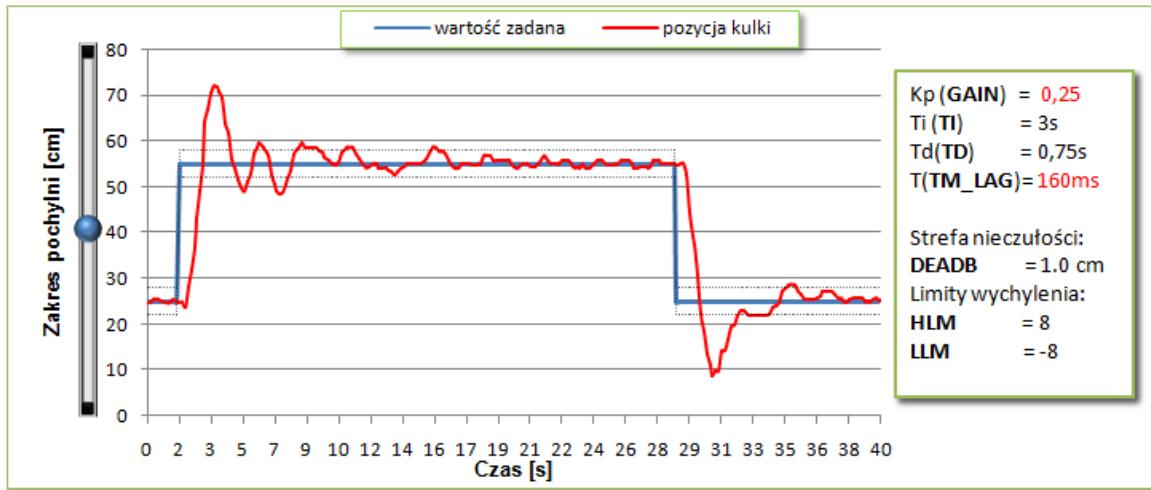
Rys. 6.10. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie $K_p=0,25$ oraz zwiększenie $T=150\text{ms}$; (2/2).

Jak widać z rysunku 6.10. pochylnia jest w stanie wytrącić kulkę z pochylni. Kolejna próba korygowania polega na wyznaczeniu granicznej inercji T (zbyt duża powoduje zwiększenie czasu regulacji lub wprowadza układ w niegasnące oscylacje), co przedstawia poniższy rysunek.



Rys. 6.11. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie Kp=0,25 oraz zwiększenie T=160ms (1/2).

Z rysunku 6.11 widać że dla małych wymuszeń skokowych (o 15 cm) układ regulacji jest w stanie ustabilizować kulę w żądanym miejscu w dość krótkim czasie. W celu sprawdzenia czy pochylnia nie wytrąci kulki przy dużych wymuszeniach skokowych (o 30 cm) przeprowadzono badanie ponownie przy tych samych nastawach regulatora, co przedstawia poniższy rysunek.

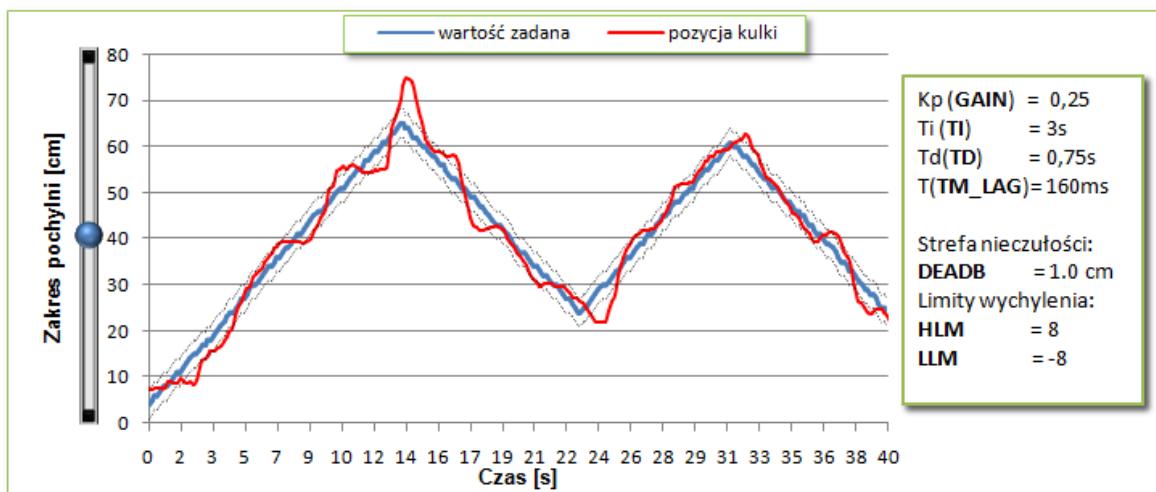


Rys. 6.12. Odpowiedź układu zamkniętego na wymuszenie skokowe - zwiększenie Kp=0,25 oraz zwiększenie T=160ms (2/2).

Badanie odpowiedzi układu regulacji na duże wymuszenia skokowe przeprowadzono kilkukrotnie w celu upewnienia się czy pochylnia przy tak skorygowanych parametrach nie wytrąci kulki. Okazuje się że przy dużych skokach układ jest w stanie wyregulować kulę w żądanym miejscu pochylni co udowadnia rysunek 6.12.

Jako eksperyment przeprowadzono kolejne badania dla tak dobranych parametrów wyznaczając odpowiedź układu zamkniętego dla innych wymuszeń oraz na zakłócenie przemieszczeniowe (wytrącanie kulki z równowagi).

Eksperyment 1: odpowiedź układu zamkniętego na wymuszenie jako sygnał trójkątny przedstawia poniższy rysunek.



Rys. 6.13. Odpowiedź układu zamkniętego na wymuszenie trójkątne – parametry regulatora PID po skorygowaniu metody Zieglera Nicholsa.

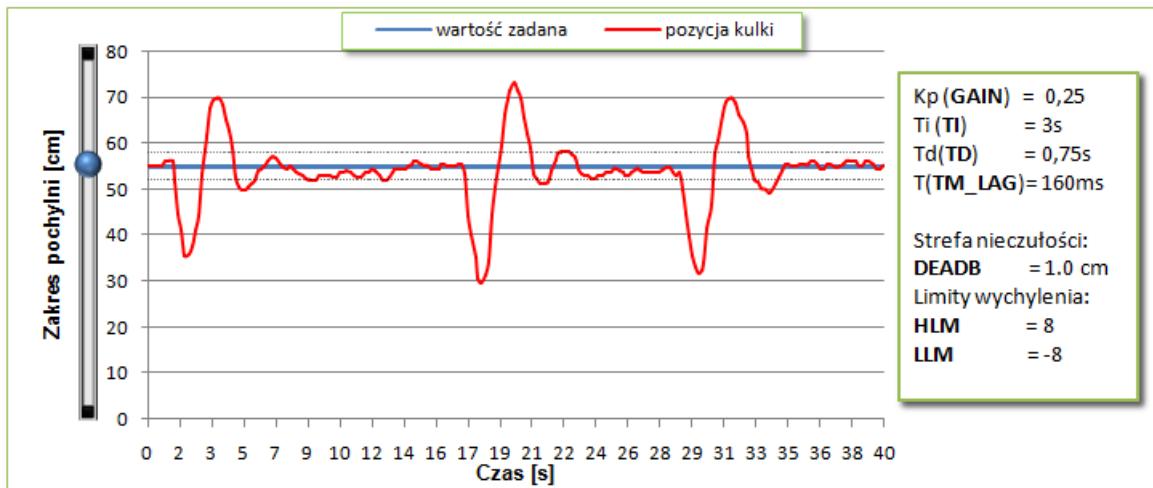
Jak widać z rysunku 6.13 układ regulacji próbuje nadążyć za zmianą wartości zadanej w okolicach której kulka oscyluje. Przy wierzchołkach trójkąta występuje przeregulowanie ponieważ następuje odwrócenie kierunku wartości zadanej.

Eksperyment 2: Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na środku pochylni (40cm) przedstawia poniższy rysunek.



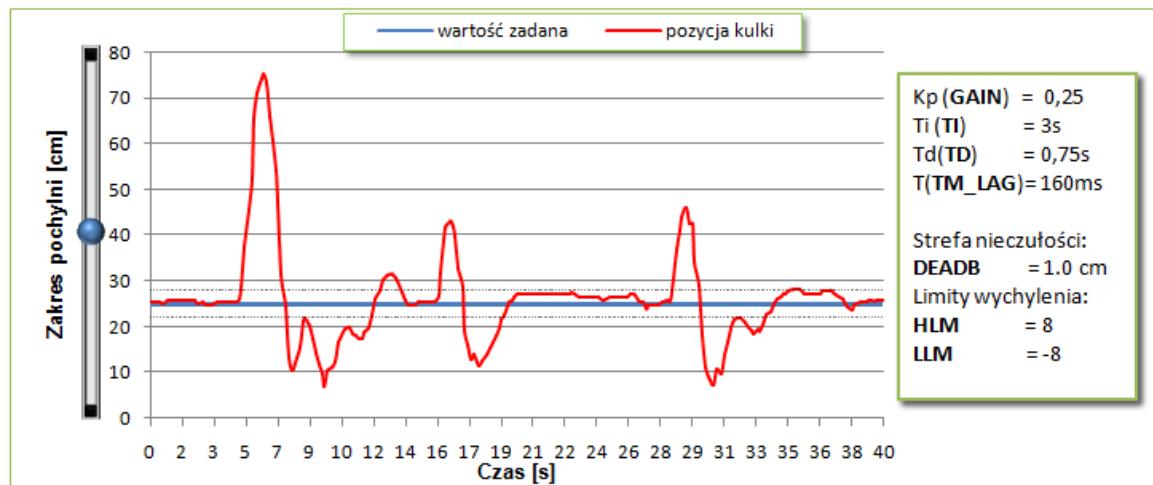
Rys. 6.14. Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na środku pochylni (40cm) – parametry regulatora PID po skorygowaniu metody Zieglera Nicholsa

Eksperyment 3: Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na prawej stronie pochylni (55cm) przedstawia poniższy rysunek.



Rys. 6.15. Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na prawej stronie pochylni (55cm) – parametry regulatora PID po skorygowaniu metody Zieglera Nicholsa.

Eksperyment 4: Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na lewej stronie pochylni (25cm) przedstawia poniższy rysunek.



Rys. 6.16. Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na lewej stronie pochylni (25cm) – parametry regulatora PID po skorygowaniu metody Zieglera Nicholsa.

Jak przedstawiają rysunki 6.14, 6.15, 6.17 układ regulacji po wytrąceniu kulki równowagi zarówno w punkcie 25, 40 i 55 jest ustabilizować kulkę z powrotem w żądanym miejscu pochylni.

Początkowe korygowanie parametrów dobranych metodą Zieglera Nicholsa zostały zakończone. Dalsze próby korygowania na podstawie powyższych parametrów przedstawione zostały w następnym podrozdziale wraz z wyznaczeniem wskaźników jakości dynamicznej układu regulacji.

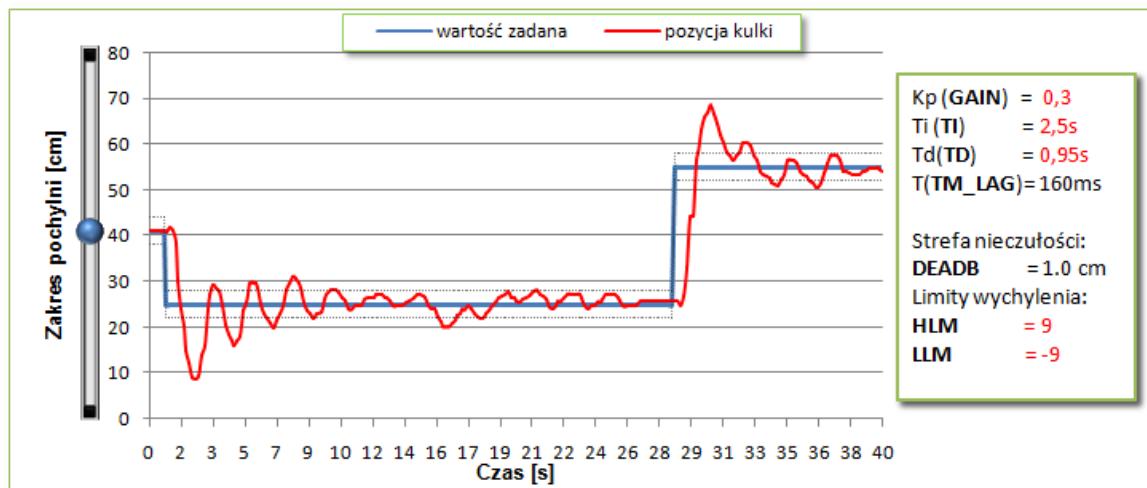
6.2. Jakość dynamiczna układu regulacji PLC

Dobierając nastawy regulatora PID w celu osiągnięcia jak najlepszych wskaźników jakości układu regulacji należy wziąć pod uwagę następujące właściwości obiektu:

- kula może czasami ślizgać się na pochylni,
- pochylnia ma określony zakres kąta, poza który nie może się wychylić,
- istnieją niewielkie luzy w układzie mechanicznym serwomechanizmu w skutek czego pochylnia w niektórych przypadkach może się zawahać,
- kula nie jest idealnie okrągła i jednorodna, składa się z dwóch połączonych półkul w skutek czego, mimo osiągnięcia wartości zadanej kula bezwładnie może zmienić pozycję,
- kula nie zawsze ma kontakt z pochylnią, przy nieodpowiednim doborze nastaw regulatora PID kula może podskakiwać (im dalej od środka pochylni tym częściej) lub spaść z pochylni.

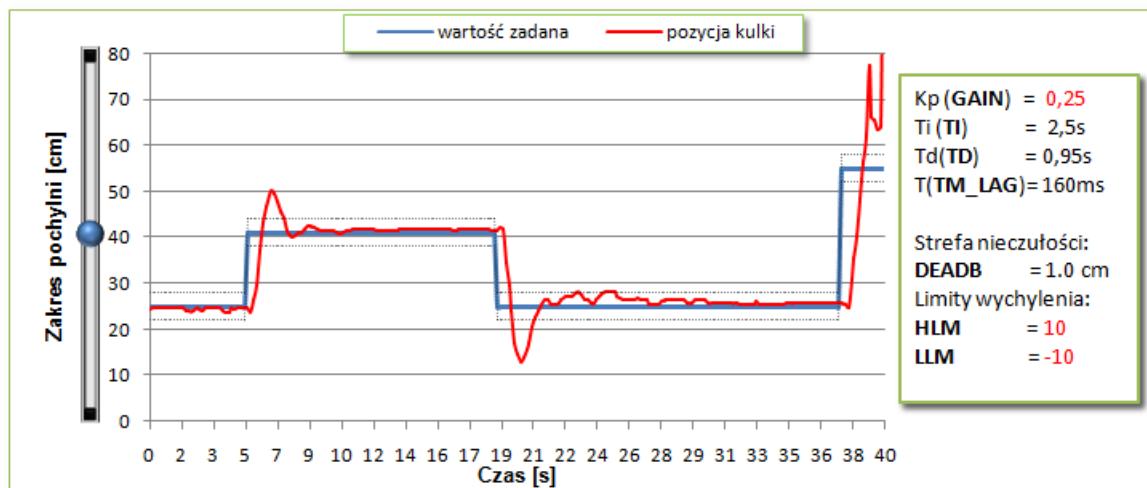
Wyznaczanie parametrów regulatora PID w celu uzyskania jak najlepszych wskaźników jakości dynamicznej układu regulacji, odbywa się w oparciu o skorygowane nastawy PID uzyskane metodą *Zieglera Nicholsa*. Wyniki badań przedstawiono w postaci odpowiedzi układu zamkniętego na wymuszenia skokowe o skoku +15, -15, +30, -30 oraz wyznaczonych i obliczonych wskaźników jakości dynamicznej układu regulacji, połączone i oznaczone jako rysunek. Po wynikach w tabeli 14 umieszczone podsumowanie wyników. Na rysunkach po prawej stronie umieszczone parametry regulatora odpowiadające charakterystyce po lewej stronie, kolorem czerwonym wyróżnione zmianę parametrów w stosunku do poprzedniego rysunku. W tabelach zawartych w każdym poniższym rysunku wartości oddzielone ukośnikiem „/” oznaczają wskaźnik jakości dla kolejnego wymuszenia skokowego, przykładowo gdy na charakterystyce są trzy wymuszenia skokowe dla danego wskaźnika wartość zostaje oddzielona ukośnikiem X1 / X2 / X3 gdzie X oznacza wartość wskaźnika dla danego wymuszenia skokowego Charakterystyki posegregowane według wzmacnienia Kp zaczynając od największego Kp=0,3.

Rysunek 6.17 przedstawia odpowiedź układu regulacji na wymuszenia skokowe o skoku -15 i +30



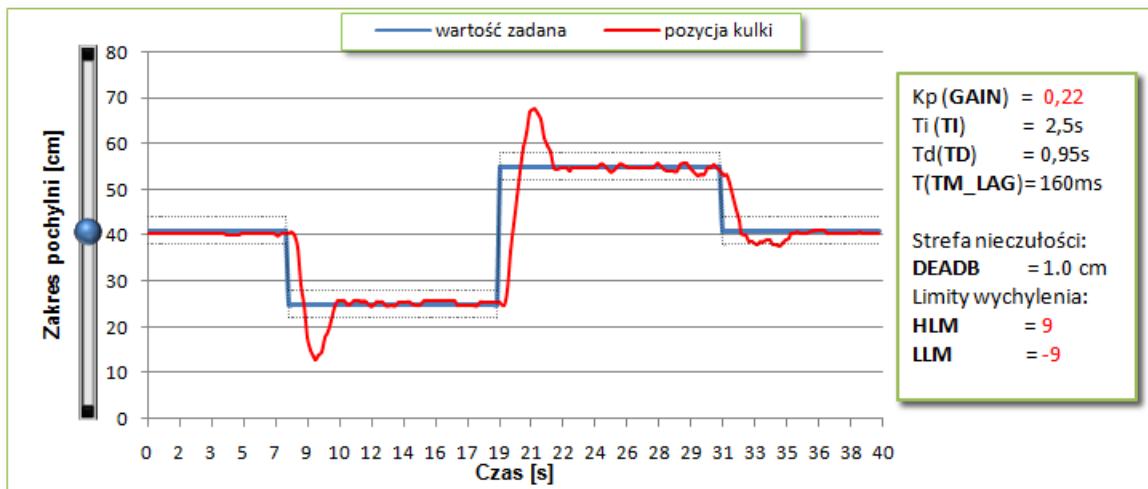
Rys. 6.17. Odpowiedź układu zamkniętego na wymuszenia skokowe - zwiększenie Kp=0,3, zmniejszenie Ti=2,5s, zwiększenie Td = 0,95s, zwiększenie limitu wychylenia +/-9.

Jak przedstawia powyższy rysunek kula po wymuszeniu skokowym długi czas oscyluje w okolicy wartości zadanej. Może to być spowodowane zbyt dużym wzmacnieniem w stosunku dla stałej czasu całkowania. Poniższy rysunek przedstawia odpowiedź układu zamkniętego na wymuszenia skokowe po zmniejszeniu wzmacnienia Kp.



Rys. 6.18. Odpowiedź układu zamkniętego na wymuszenia skokowe - zmniejszenie Kp=0,25, zwiększenie limitu wychylenia +/-10.

Odpowiedź układu zamkniętego na wymuszenie skokowe po zmniejszeniu Kp i kątów wychylenia przedstawia poniższy rysunek.



Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o - 15 / +30 / -15			
0,8 / 1,1 / 1	2,3 / 2,4 / 1,8	8 / 2 / 3	73 / 43 / 18

Rys. 6.19. Odpowiedź układu zamkniętego na wymuszenia skokowe - zmniejszenie Kp=0,22, zmniejszenie limitu wychylenia +/-9.

Z powyższego rysunku widać że dobrane parametry są bliskie osiągnięcia uzyskania pożądanych wskaźników jakości układu regulacji. Jedynie zbyt duże przeregulowanie bezwzględne odbiega od uzyskania celu. Kolejna próba doboru nastaw regulatora polega na zmniejszeniu czasu opóźnienia członu różniczkującego (inercja T), co przedstawia poniższy rysunek.



Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o - 15 / +15			
1,0 / 1,1	5,6 / 6	43 / 55	55 / 3

Rys. 6.20. Odpowiedź układu zamkniętego na wymuszenia skokowe - zmniejszenie inercji T=130ms.

Z rysunku 6.20 widać że po zmniejszeniu parametru T przeregulowanie bezwzględne zmalało, jednak pogorszyły się inne wskaźniki, czas regulacji i przeregulowanie względne. Poniższy rysunek przedstawia odpowiedź układu zamkniętego na wymuszenia skokowe o skoku -15, +15, +15, -15, po zmniejszeniu stałej całkowania i zwiększeniu inercji T.

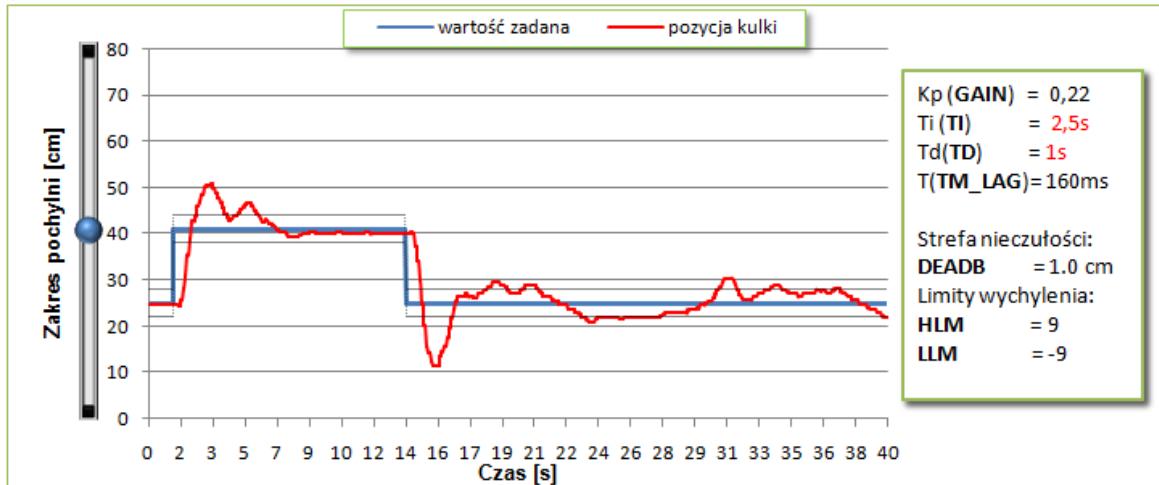


Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o -15 / +15 / +15 / -15			
1,1 / 0,8 / 1,0 / 1,1	3,3 / 2,9 / 4,7 / 6,9	65 / 36 / 65 / 1	48 / 31 / 66 / 56

Rys. 6.21. Odpowiedź układu zamkniętego na wymuszenia skokowe - zmniejszenie stałej całkowania

Ti=2,2s, zwiększenie inercji T=160ms.

Poniższy rysunek przedstawia odpowiedź układu zamkniętego na wymuszenia skokowe o skoku +15 / -15 po zwiększeniu stałej całkowania Ti i stałej różniczkowania Td.

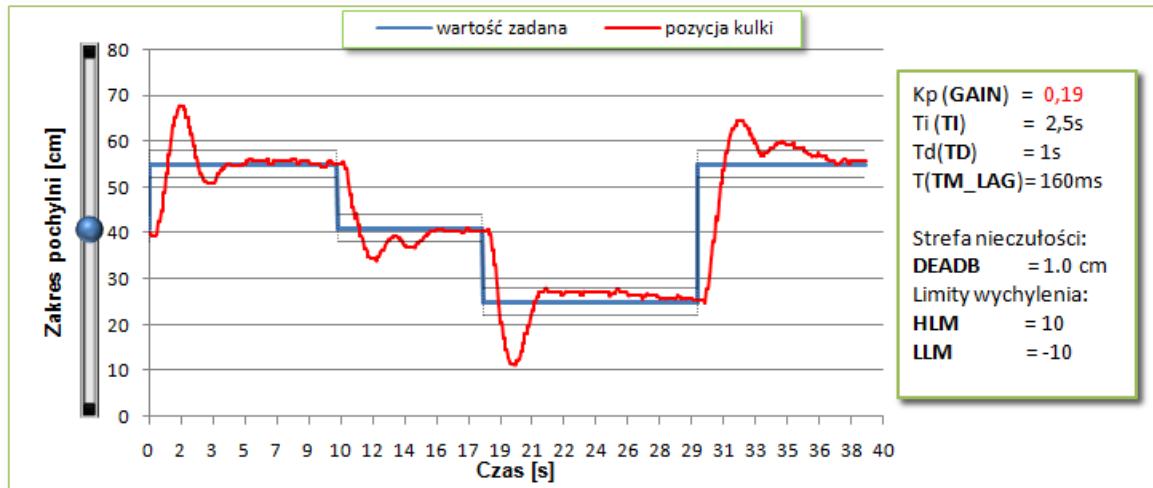


Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o +15 / -15			
1,0 / 0,9	4,7 / nie wyregulowano	17 / 29	52 / 93

Rys. 6.22. Odpowiedź układu zamkniętego na wymuszenia skokowe - zwiększenie stałej całkowania

Ti=2,5s, zwiększenie stałej różniczkowania Td=1s.

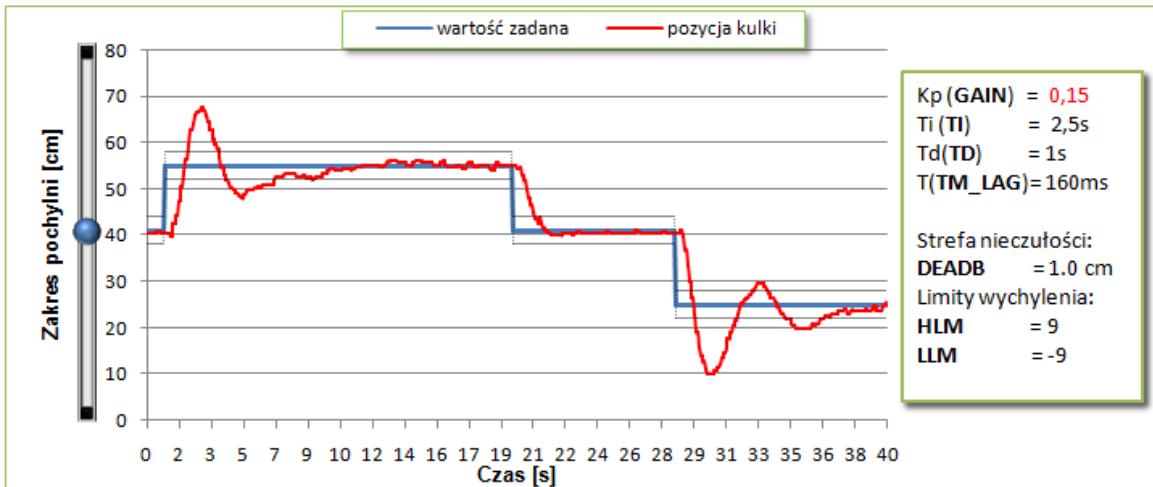
Poniższy rysunek przedstawia odpowiedź układu zamkniętego na wymuszenia skokowe o skoku +15 / -15 / -15 / +30 po zmniejszeniu wzmocnienia Kp.



Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o +15 / -15 / -15 / +30			
1,1 / 1,3 / 1,0 / 1,5	3,8 / 4,7 / 3,5 / 5,7	29 / 0 / 11 / 0	85 / 46 / 94 / 31

Rys. 6.23. Odpowiedź układu zamkniętego na wymuszenia skokowe - zmniejszenie wzmocnienia $K_p=0,19$.

Poniższy rysunek przedstawia odpowiedź układu zamkniętego na wymuszenia skokowe o skoku +15 / -15 / -15 po ponownym zmniejszeniu wzmocnienia Kp.



Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o +15 / -15 / -15			
1,4 / 2,0 / 1,0	5,1 / 1,4 / 8,1	63 / 0 / 34	73 / 4 / 96

Rys. 6.24. Odpowiedź układu zamkniętego na wymuszenia skokowe - zmniejszenie wzmocnienia $K_p=0,15$.

Rysunek 6.25 przedstawia odpowiedź układu zamkniętego na wymuszenia skokowe o skoku +15 / -15 / -15 po zwiększeniu wzmocnienia Kp i znacznym zmniejszeniu limitu wychylenia pochylni.



Czas odpowiedzi [s]	Czas regulacji [s]	Przeregulowanie względne [%]	Przeregulowanie bezwzględne [%]
skok wartości zadanej o -15 / +15 / +15 / -15			
1,2 / 1,1 / 1,2 / 0,9	6,2 / 4 / 6,3 / 6,0	55 / 44 / 45 / 47	106 / 89 / 92 / 60

Rys. 6.25. Odpowiedź układu zamkniętego na wymuszenia skokowe - zwiększenie wzmocnienia

$K_p=0,22$, zmniejszenie limitów wychylenia pochylni +/- 6,5.

Z rysunku 6.25 widać że przy małych kątach pracy pochylni kulka przemieszcza się płynnie i powoli czego skutkiem jest długi czas regulacji

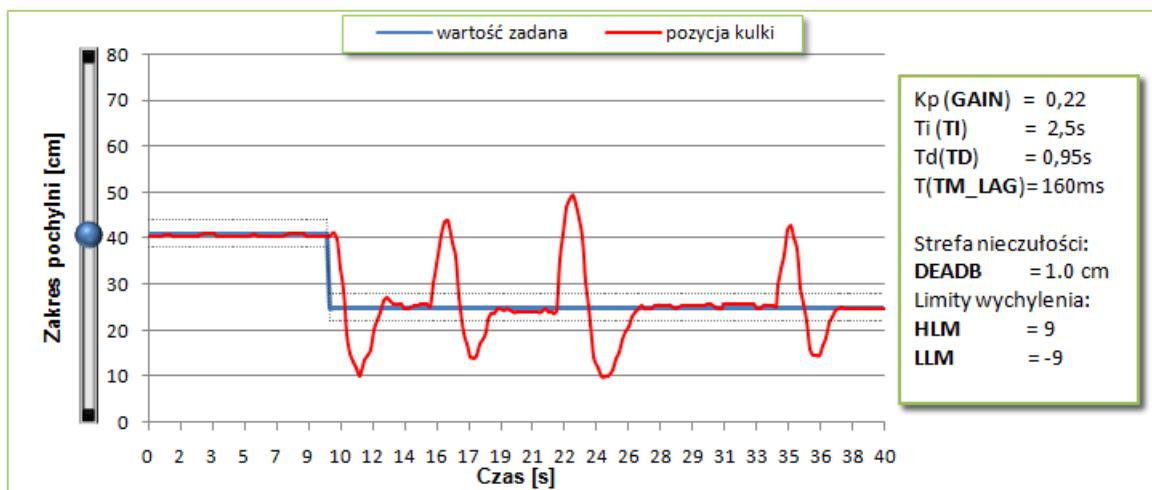
Tabela 14 przedstawia podsumowanie wyników badania wpływu nastaw regulatora na wskaźniki jakości układu regulacji. W kolumnach parametrów regulatora PID kolorem czerwonym wyróżniono zmianę nastaw analogicznie do zmieniających się nastaw na rysunkach. Komórka tabeli wypełniona kolorem czerwonym oznacza, że przy takim doborze nastaw regulatora wystąpiło wytrącenie kulki z pochylni. Wiersz tabeli wypełniony kolorem żółtym wyróżnia parametry regulatora PID które osiągnęły stosunkowo najkorzystniejsze dla obiektu wskaźniki jakości układu regulacji.

Tabela 14. Podsumowanie wyników badania wpływu nastaw regulatora PID na wskaźniki jakości układu regulacji.

Parametry regulatora PID						Czas odpowiedzi	Czas regulacji	Przer. Względne	Przer. Bezwzgl.
Kp	Ti	Td	T	DB	LIM				
0,3	2,5s	0,95s	160ms	1.0	+/- 9	0,9 s	15,0 s	25 %	106 %
0,25	2,5s	0,95s	160ms	1.0	+/- 10	0,9 s	2,2 s	5 %	70 %
0,22	2,5s	0,95s	160ms	1.0	+/- 9	0,9 s	2,3 s	5 %	43 %
0,22	2,5s	0,95s	130ms	1.0	+/- 9	1,0 s	5,8 s	48 %	29 %
0,22	2,2s	0,95s	160ms	1.0	+/- 9	1,0 s	4,5 s	42%	51 %
0,22	2,5s	1s	160ms	1,0	+/- 9	0,9 s	brak	21 %	70 %
0,19	2,5s	1s	160ms	1,0	+/- 9	1,2 s	4,0 s	10 %	64 %
0,15	2,5s	1s	160ms	1,0	+/- 9	1,3 s	4,9 s	49 %	60 %
0,22	2,5s	1s	160ms	1,0	+/- 6,5	1,1 s	5,7 s	48 %	87 %

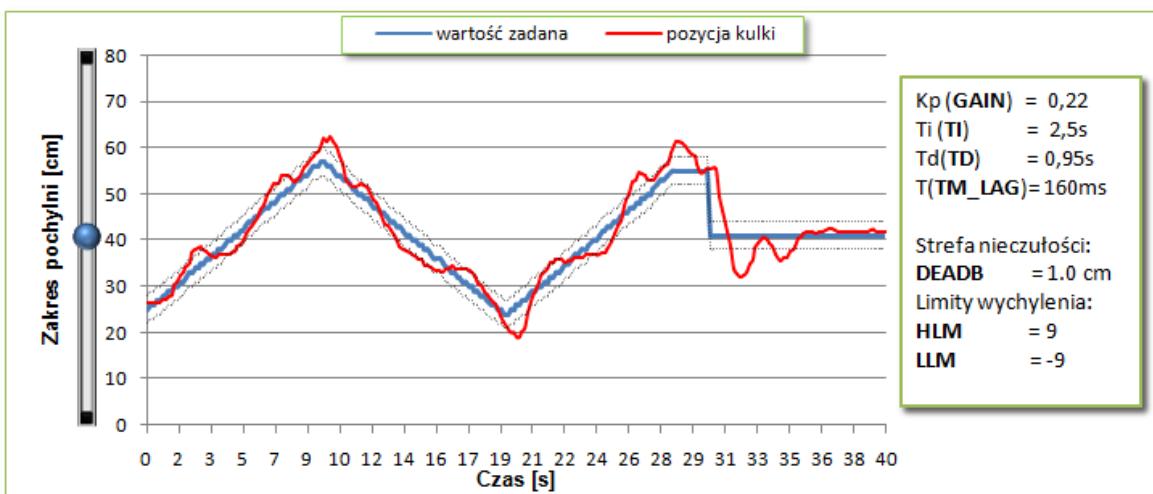
Dla parametrów regulatora wyróżnionych kolorem żółtym w tabeli 14. Wykonano następujące badania eksperymentalne:

Eksperiment 1: Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na lewej stronie pochylni (25cm) przedstawia poniższy rysunek.



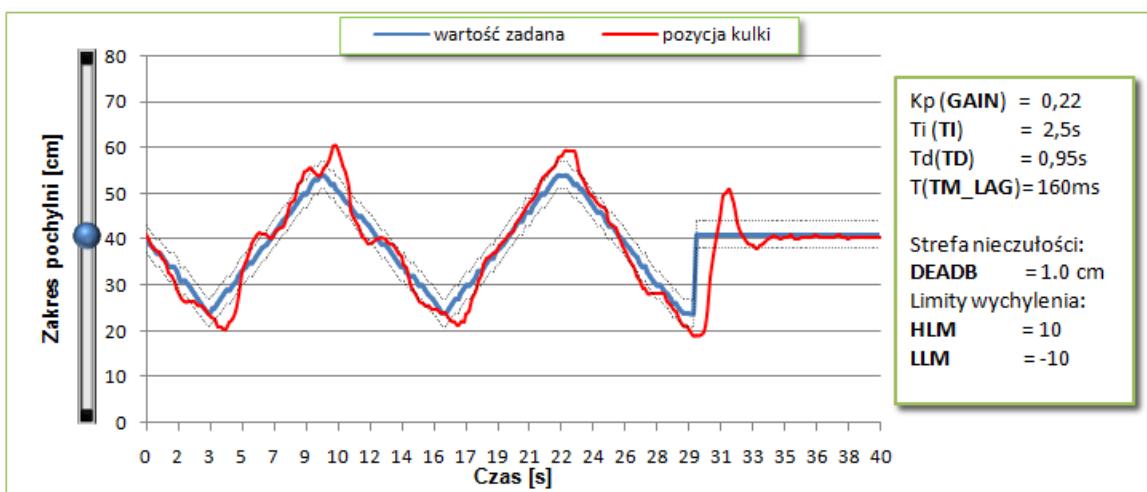
Rys. 6.26. Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na lewej stronie pochylni (25cm)

Eksperiment 2: odpowiedź układu zamkniętego na wymuszenie jako sygnał trójkątny o okresie 18s przedstawia poniższy rysunek.



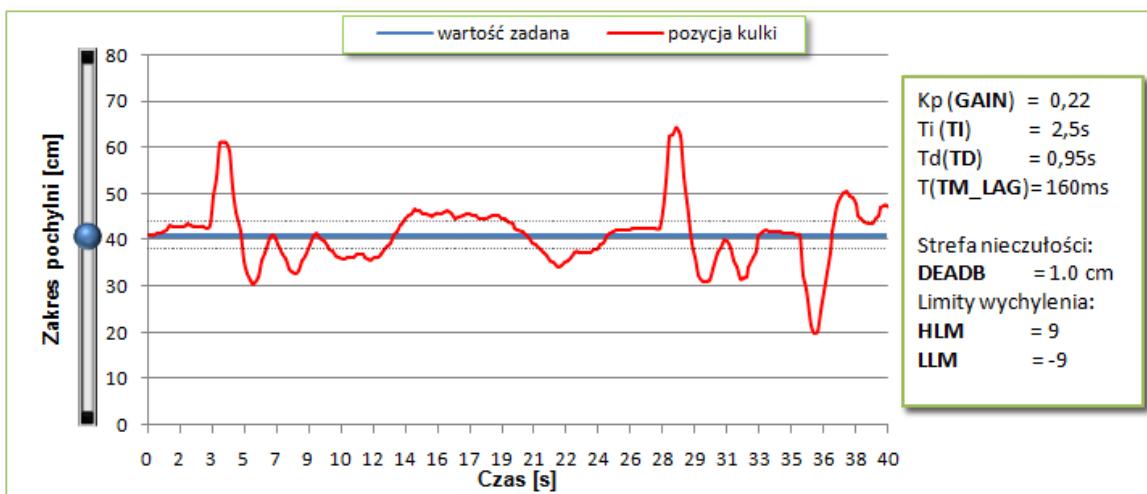
Rys. 6.27. odpowiedź układu zamkniętego na wymuszenie jako sygnał trójkątny.

Eksperiment 3: odpowiedź układu zamkniętego na wymuszenie jako sygnał trójkątny o okresie 12s przedstawia poniższy rysunek.

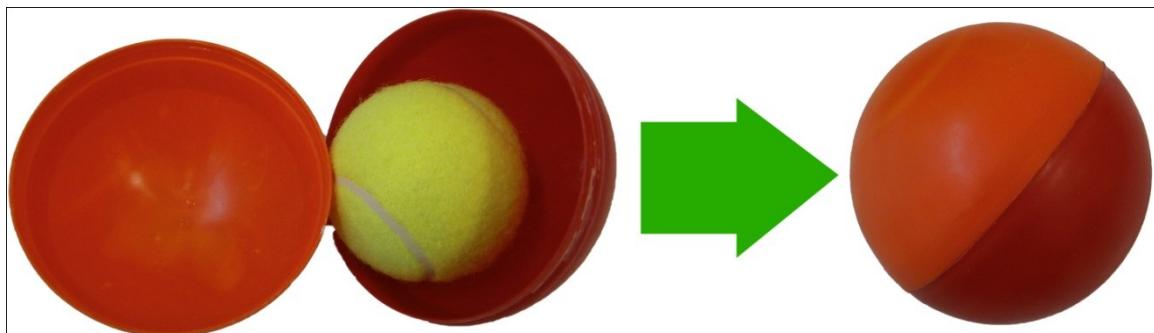


Rys. 6.28. odpowiedź układu zamkniętego na wymuszenie jako sygnał trójkątny.

Eksperyment 4: Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na środku pochylni (40cm) dla niestandardowej kulki (Rys. 6.30) przedstawia poniższy rysunek.



Rys. 6.29. Odpowiedź układu zamkniętego na zakłócenie przemieszczeniowe na środku pochylni (40cm) dla niestandardowej kulki.



Rys. 6.30. Niestandardowa kulka – *Pilka w Kulce*

PODSUMOWANIE

Projekt układu sterowania czasu rzeczywistego położeniem kulki na pochylni z wykorzystaniem sieci PROFINET został opracowany z myślą o możliwości przeprowadzania na nim późniejszych badań w celach dydaktycznych. Wkładano wszelkie starania aby opisy poszczególnych etapów projektowania były jak najbardziej czytelne i zrozumiałe dla użytkownika, niemającego wcześniej do czynienia z takim układem sterowania.

Biorąc pod uwagę ogólny projektu, wydaje się on stosunkowo łatwy do zrealizowania. Jednak w drodze projektowania pojawiało się wiele problemów związanych z poszczególnymi elementami układu sterowania, co wynikało z niewielkiej lub braku wiedzy na ich temat. Po przestudiowaniu powszechnie dostępnej literatury, osiągnięto cele projektu krok po kroku. Jako pierwszy etap zapoznania się z elementami układu sterowania wybrano sterowanie serwomechanizmem. Na początku zbadano jak wartości sygnału sterującego silnikiem ze sterownika powodują wychylenie wału silnika. Przeanalizowano i zbadano wszystkie możliwe sposoby sterowania serwomechanizmem na udostępnionym sprzęcie i wybrano metodę uznawaną za najlepszą. Kolejnym problemem było przeliczanie napięcia z wyjścia czujników optycznych na wartość określającą realne położenie kulki na pochylni. Pierwszą koncepcją było stworzenie tablicy z zawartością odległości od 0 do 80 cm co 0,5 cm i warunków określających która wartość z tablicy zostanie podstawiona dla danego napięcia czujnika. Prostszą metodą okazało się zlinearyzowanie charakterystyki odległościowo napięciowej czujnika metodą aproksymacji. Początkowo próby przeprowadzano na jednym czujniku, jednak szybko okazało się że jeden czujnik znacznie ograniczy zakres pochylni, w którym kula będzie mogła się stabilizować. Rozwiążanie problemu zapewniło zastosowanie dwóch czujników i stworzenie warunków określających przełączający punkt odczytów czujników. Konfiguracja i parametryzacja stanowiska PLC przebiegła bezproblemowo, ponieważ związany z tym zakres projektu, pokrywał się z zakresem nauczania studiów. Wcześniejszego zainteresowanie programowaniem sterowników PLC znacznie ułatwiło stworzenie algorytmu sterowania dla układu regulacji.

Po utworzeniu układu sterowania i upewnieniu się że założenia projektowe zgadzają się z układem regulacji, przystąpiono do badań polegających na doborze nastaw regulatora PID oraz ich wpływu na jakość regulacji. Jako pierwszą i zarazem trafną metodę doboru nastaw zastosowano metodę *Zieglera-Nicholsea*. Co prawda oryginalne nastawy z tej

metody nie zapewniły dobrych wskaźników jakości regulacji, lecz dały nastawy początkowe które stabilizowały obiekt. Po stopniowym korygowaniu nastaw osiągnięto parametry regulatora PID, które zapewniały wskaźniki jakości regulacji ocenione jako akceptowalne przez układ.

Podsumowując, stworzenie stanowiska dydaktycznego praktycznie od podstaw, zmusza do szczegółowego zapoznania się z każdym elementem układu sterowania. Zdobыта wiedza związana z projektem pozwoliła na zapoznanie się z układem regulacji oraz wpływu nastaw regulatora PID na jakość regulacji.

LITERATURA

1. A. Burghardt, J. Giergel, *Modelowanie inżynierskie - Modelowanie i sterowanie układu kulka – belka*, Katedra Mechaniki Stosowanej i Robotyki, Politechnika Rzeszowska Gliwice 2009.
2. M. Ganobis, *Distributed control of „ball on beam” system*, University of Science and Technology AGH October 2009.
3. Fu Zhang, *Feedback linearization control of system with singularitirs: a ball-beam revisit*, The Mathworks October 2010.
4. J. Kwaśniewski, *Programowalny sterownik SIMATIC S7-300 w praktyce inżynierskiej*, BTC, Legionowo 2009.
5. Siemens, *SIMATIC STEP7 – Podstawy programowania w STEP7*, Podręcznik – wydanie 3, Siemens Warszawa 2011 .
6. Siemens, *Pomoc programu STEP7*, Simatic Menager.
7. Siemens, *Sterowniki PLC SIMATIC – Innowacyjne rozwiązania w automatyce*, Siemens Warszawa 2011.
8. Siemens, *SIMATIC ET 200 – Rozproszone systemy wejść/wyjść*, Siemens Warszawa 2011.
9. Siemens, *PROFINET Podręcznik – wydanie 1*, Siemens Warszawa 2010.
10. Inżynieria & Utrzymanie Ruchu Zakładów Przemysłowych, Marzec 2012 Nr 2 (80), Warszawa 2012.
11. Elektronika Praktyczna plus, Międzynarodowy magazyn elektroników konstruktorów, Wydanie specjalne 3/2010
12. F. Siemieniako, Z. Gosiewski, *Automatyka – Modelowanie i analiza układów*, Tom I, Politechnika Białostocka Białystok 2006.
13. F. Siemieniako, Z. Gosiewski, *Automatyka – Modelowanie i analiza układów*, Tom II, Politechnika Białostocka Białystok 2006.
14. T. Kaczorek, *Teoria układów regulacji automatycznej*, WNT Warszawa 1977.
15. http://www.automatyka.siemens.pl/docs/docs_ia/mFAQ12CP3431ET200S.pdf
16. <http://www.acroname.com/robotics/info/articles/irlinear/irlinear.html>
17. http://botland.com.pl/attachment.php?id_attachment=13
18. <http://www.henryk.mbapp.com/servo/servo.html>
19. <http://www.autopsy.de/en/index.html>

ZAŁĄCZNIK 1 – LISTING PROGRAMU STEROWANIA

Załącznik 1 zawiera cały listing programu wraz z komentarzami i odnośnikami do rysunków i tabel projektu. Załącznik został podzielony na podrozdziały:

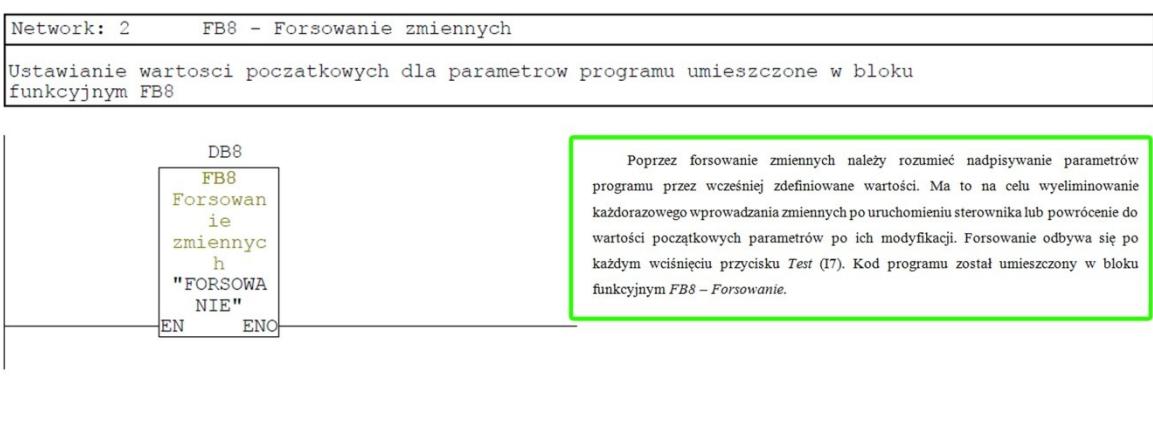
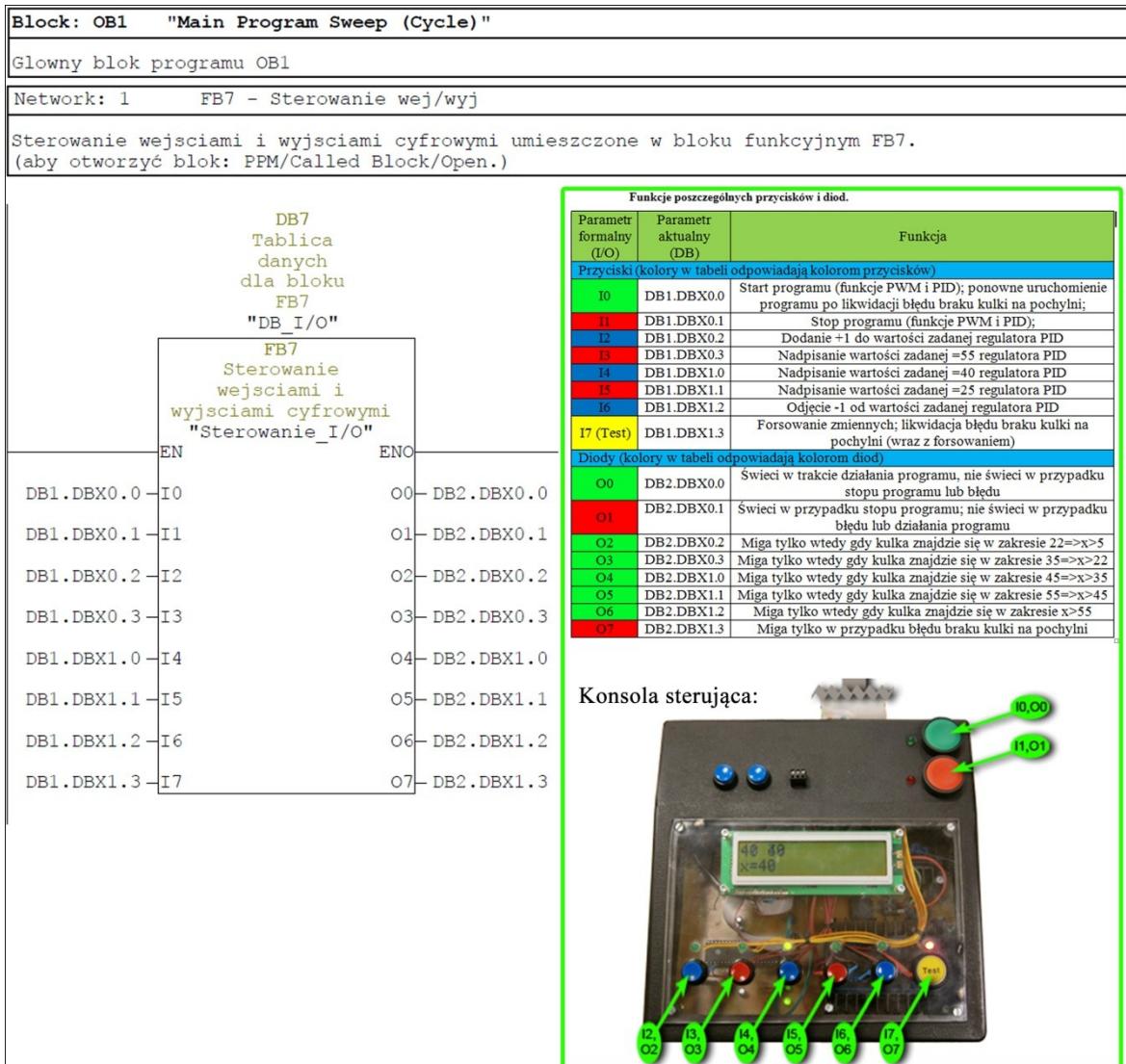
Z 1.1. Blok OB1 – zawiera główny program.	str. 91
Z 1.2. Blok FB7 – sterowanie przyciskami i diodami	str. 95
Z 1.3. Blok FB8 – Forsowanie zmiennych	str. 99
Z 1.4. Blok FB3 – Linearyzacja i sterowanie odczytem czujników	str. 103
Z 1.5. Blok OB35 – Cykliczne przerwania –zawiera blok PID oraz FB4 w którym jest zawarty generator PWM	str. 106
Z 1.6. Blok FB4 – Generowanie sygnału PWM	str. 109
Z 1.7. Blok OB100 – restart regulatora PID (STL)	str. 110

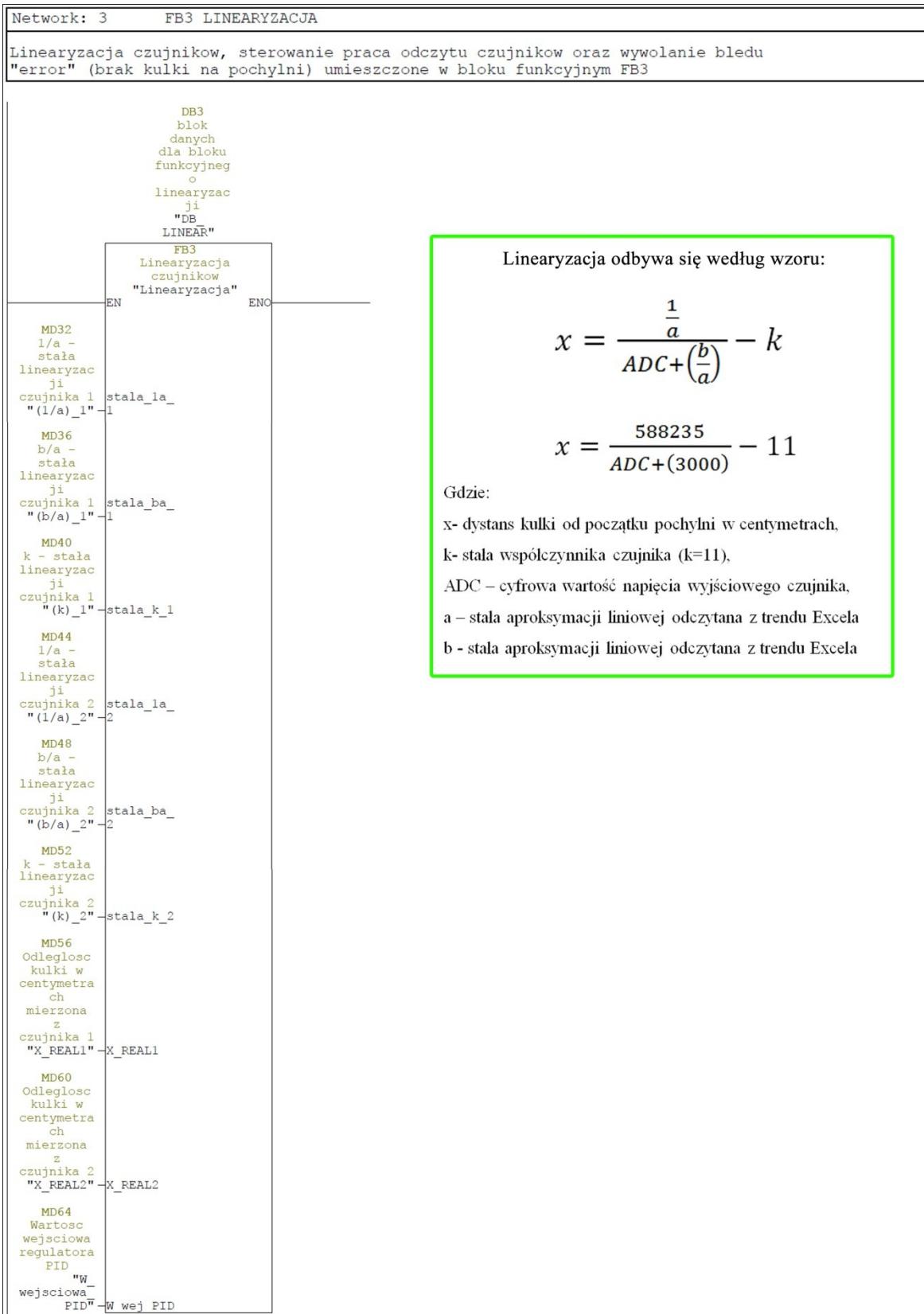
Poniższa tabela zawiera spis symboli używanych w programie wraz z komentarzami.

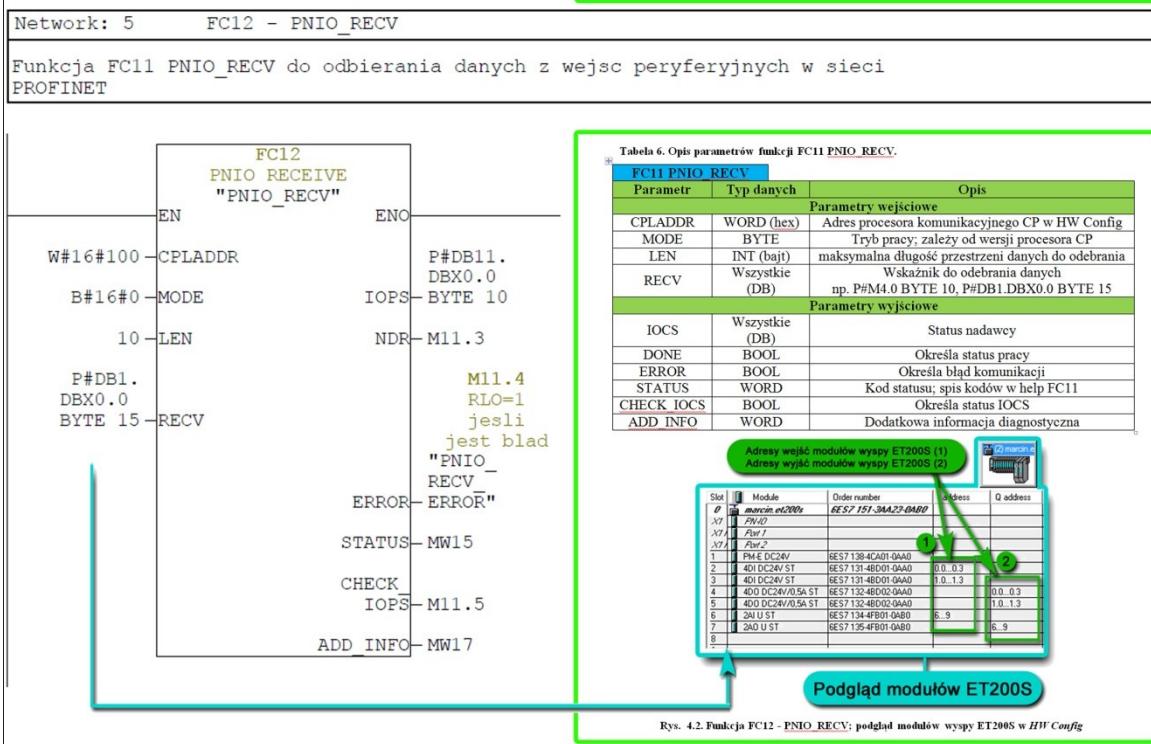
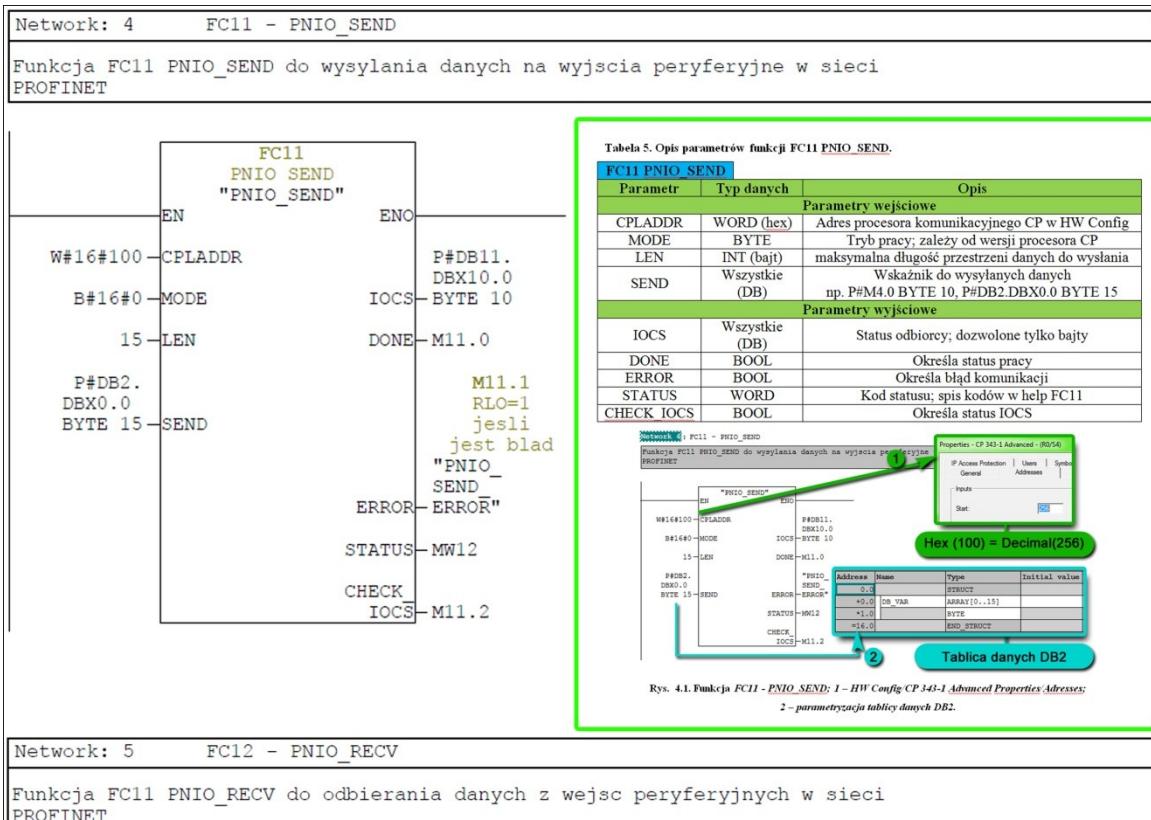
Tablica symboli:

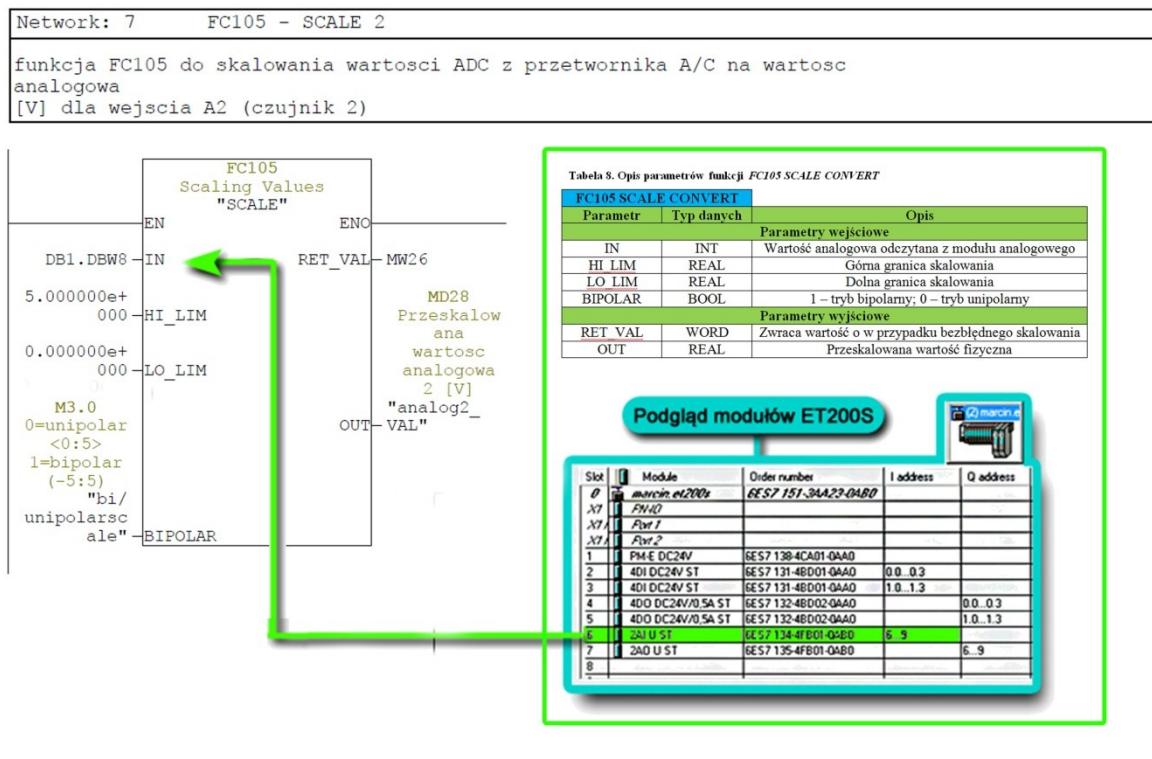
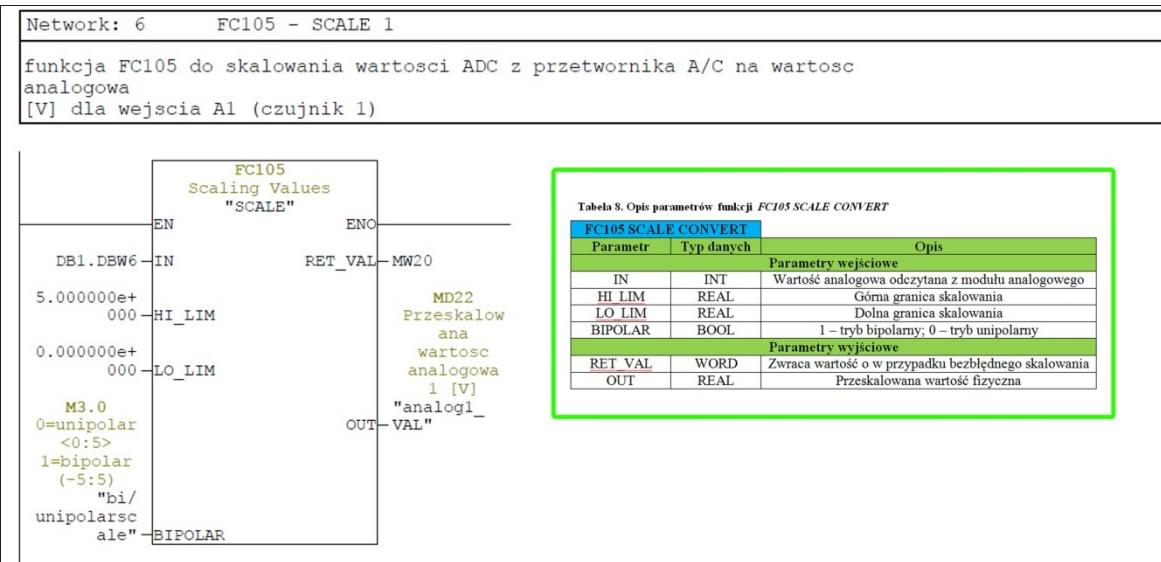
	Symbol /	Address	Data type	Comment
1	(1/a)_1	MD 32	REAL	1/a - stała linearyzacji czujnika 1
2	(1/a)_2	MD 44	REAL	1/a - stała linearyzacji czujnika 2
3	(b/a)_1	MD 36	REAL	b/a - stała linearyzacji czujnika 1
4	(b/a)_2	MD 48	REAL	b/a - stała linearyzacji czujnika 2
5	(k)_1	MD 40	REAL	k - stała linearyzacji czujnika 1
6	(k)_2	MD 52	REAL	k - stała linearyzacji czujnika 2
7	analog1_VAL	MD 22	REAL	Przeskalowana wartosc analogowa 1 [V]
8	analog2_VAL	MD 28	REAL	Przeskalowana wartosc analogowa 2 [V]
9	bi/unipolarscale	M 3.0	BOOL	0=unipolar <0:5> 1=bipolar (-5:5)
10	clc_0.5Hz(2s)	M 100.7	BOOL	wywołanie bitu co 2s
11	clc_0.625Hz(1.6s)	M 100.6	BOOL	wywołanie bitu co 1.6s
12	clc_1.25Hz(0.8s)	M 100.4	BOOL	wywołanie bitu co 0.8s
13	clc_10Hz(0.1s)	M 100.0	BOOL	wywołanie bitu co 0.1s
14	clc_1Hz(1s)	M 100.5	BOOL	wywołanie bitu co 1s
15	clc_2.5Hz(0.4s)	M 100.2	BOOL	wywołanie bitu co 0.4s
16	clc_2Hz(0.5s)	M 100.3	BOOL	wywołanie bitu co 0.5s
17	clc_5Hz(0.2s)	M 100.1	BOOL	wywołanie bitu co 0.2s
18	CONT_C	FB 41	FB 41	Ciągły regulator PID
19	DB_I/O	DB 7	FB 7	Tablica danych dla bloku FB7
20	DB_LINEAR	DB 3	FB 3	blok danych dla bloku funkcyjnego linearyzacji
21	DB_PID	DB 6	FB 41	Tablica danych dla regulatora PID
22	DB_PNIO_RECV	DB 1	DB 1	Blok odbierania danych z sieci PROFINET
23	DB_PNIO_SEND	DB 2	DB 2	Blok wysyłania danych do sieci PROFINET
24	error	M 5.1	BOOL	blad - brak kulki na pochylni
25	FORS(I7)	M 4.2	BOOL	Znacznik forsowania
26	FORSOWANIE	FB 8	FB 8	Forsowanie zmiennych
27	Linearyzacja	FB 3	FB 3	Linearyzacja czujników
28	OB1	OB 1	OB 1	Główny blok programu
29	OB100(RESTART)	OB 100	OB 100	Kompletny restart regulatora PID
30	OB35(100ms)	OB 35	OB 35	blok wywoływany cyklicznie co 100ms
31	PID_COM_RST	M 116.0	BOOL	Reset regulatora PID
32	PID_D_SEL	M 116.7	BOOL	aktywacja członu różniczkującego PID
33	PID_DEADB_W	MD 142	REAL	szerokość strefy nieczulosci
34	PID_GAIN	MD 126	REAL	wzmocnienie członu proporcjonalnego PID
35	PID_I_ITL_ON	M 116.6	BOOL	inicjacja członu całkującego
36	PID_I_SEL	M 116.4	BOOL	aktywacja członu całkującego PID
37	PID_INT_HOLD	M 116.5	BOOL	wstrzymanie członu całkującego PID
38	PID_LMN(out)	MD 154	REAL	zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora
39	PID_LMN_HLM	MD 146	REAL	góra granica wartości sterującej
40	PID_LMN_LLM	MD 150	REAL	dolina granica wartości sterującej
41	PID_MAN	MD 122	REAL	Wartość wyjściowa regulatora dla sterowania ręcznego
42	PID_MAN_ON	M 116.1	BOOL	Przelaczanie na tryb sterowania ręcznego
43	PID_P_SEL	M 116.3	BOOL	aktywacja członu proporcjonalnego PID
44	PID_PVPER_ON	M 116.2	BOOL	Przelaczanie typu wartości procesowej (real/analog)
45	PID_TD	MD 134	TIME	stała czasowa różniczkowania PID
46	PID_TI	MD 130	TIME	stała czasowa członu całkującego PID
47	PID_TM_LAG	MD 138	TIME	stała czasowa inercji
48	PNIO_RECV	FC 12	FC 12	PNIO RECEIVE
49	PNIO_RECV_ERR...	M 11.4	BOOL	RLO=1 jeśli jest blad
50	PNIO_SEND	FC 11	FC 11	PNIO SEND
51	PNIO_SEND_ERR...	M 11.1	BOOL	RLO=1 jeśli jest blad
52	PPP	VAT 1		Podgląd Parametrow Programu
53	PULSE	SFB 49	SFB 49	Pulse output module
54	PWM_GEN	FB 4	FB 4	Generowanie sygnału PWM
55	PWM_MAN_DO(0)	M 106.1	BOOL	Reczna kontrola PWM
56	PWM_okres	MD 110	DINT	okres dla przebiegu pwm (zawsze 200ms = 50Hz)
57	PWM_SET_DO(1)	M 106.2	BOOL	kontrolowanie wyjsc PWM
58	PWM_SW_EN(1)	M 106.0	BOOL	start/stop wyjście PWM
59	SCALE	FC 105	FC 105	Scaling Values
60	SET_C1	M 4.4	BOOL	Nadpisanie wyjścia licznika przez PV
61	START/STOP	M 4.1	BOOL	Start/Stop programu (PID,PWM)
62	Sterowanie_I/O	FB 7	FB 7	Sterowanie wejściami i wyjściami cyfrowymi
63	W_wejsciowa_PID	MD 64	REAL	Wartość wejściowa regulatora PID
64	W_zadana_PID	MD 118	REAL	Wartość zadana (cm)
65	WYPEŁNIENIE	MW 108	INT	Wypełnienie sygnału PWM - sterowanie serwem
66	X_Odleglosc	MD 68	REAL	zmienna procesowa (odleglosc kulki od poczatku rowni)
67	X_REAL1	MD 56	REAL	Odleglosc kulki w centymetrach mierzona z czujnika 1
68	X_REAL2	MD 60	REAL	Odleglosc kulki w centymetrach mierzona z czujnika 2

Z 1.1. Blok OB1









Z 1.2. Blok FB7 – sterowanie przyciskami i diodami

Parametry wejścia/wyjścia bloku FB7:

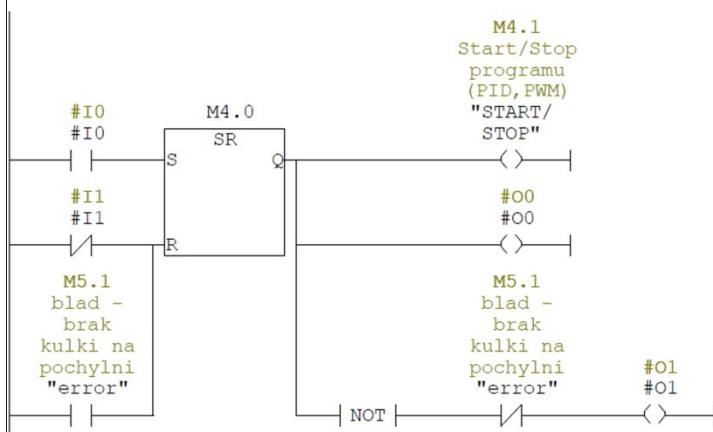
Name	Data Type	Address	Initial Value	Comment
IN		0.0		
I0	Bool	0.0	FALSE	
I1	Bool	0.1	FALSE	
I2	Bool	0.2	FALSE	
I3	Bool	0.3	FALSE	
I4	Bool	0.4	FALSE	
I5	Bool	0.5	FALSE	
I6	Bool	0.6	FALSE	
I7	Bool	0.7	FALSE	
OUT		0.0		
O0	Bool	2.0	FALSE	
O1	Bool	2.1	FALSE	
O2	Bool	2.2	FALSE	
O3	Bool	2.3	FALSE	
O4	Bool	2.4	FALSE	
O5	Bool	2.5	FALSE	
O6	Bool	2.6	FALSE	
O7	Bool	2.7	FALSE	
IN_OUT		0.0		
STAT		0.0		
TEMP		0.0		

Block: FB7 Sterowanie DI/DO

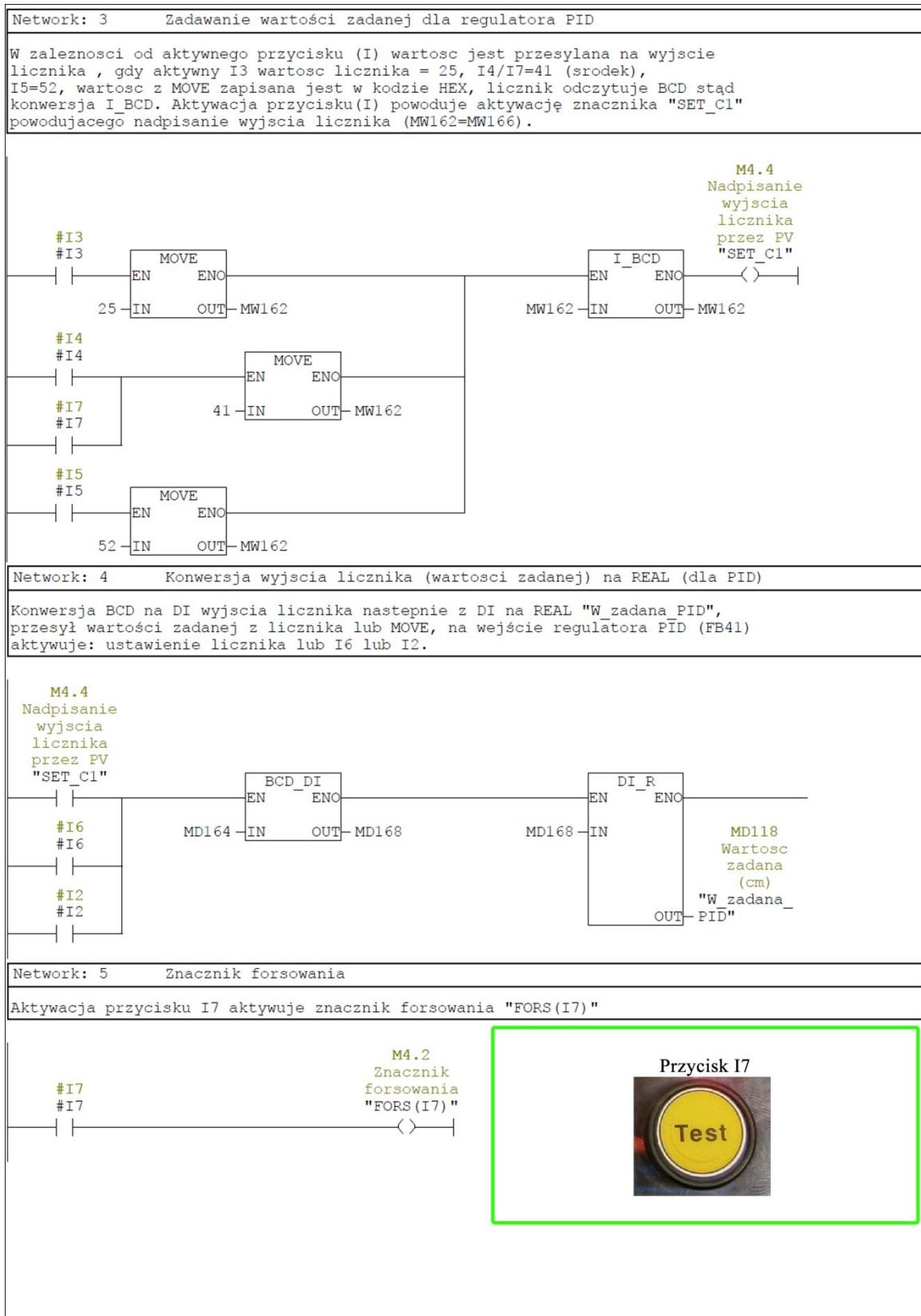
Network: 1 Sterowanie START/STOP Programu

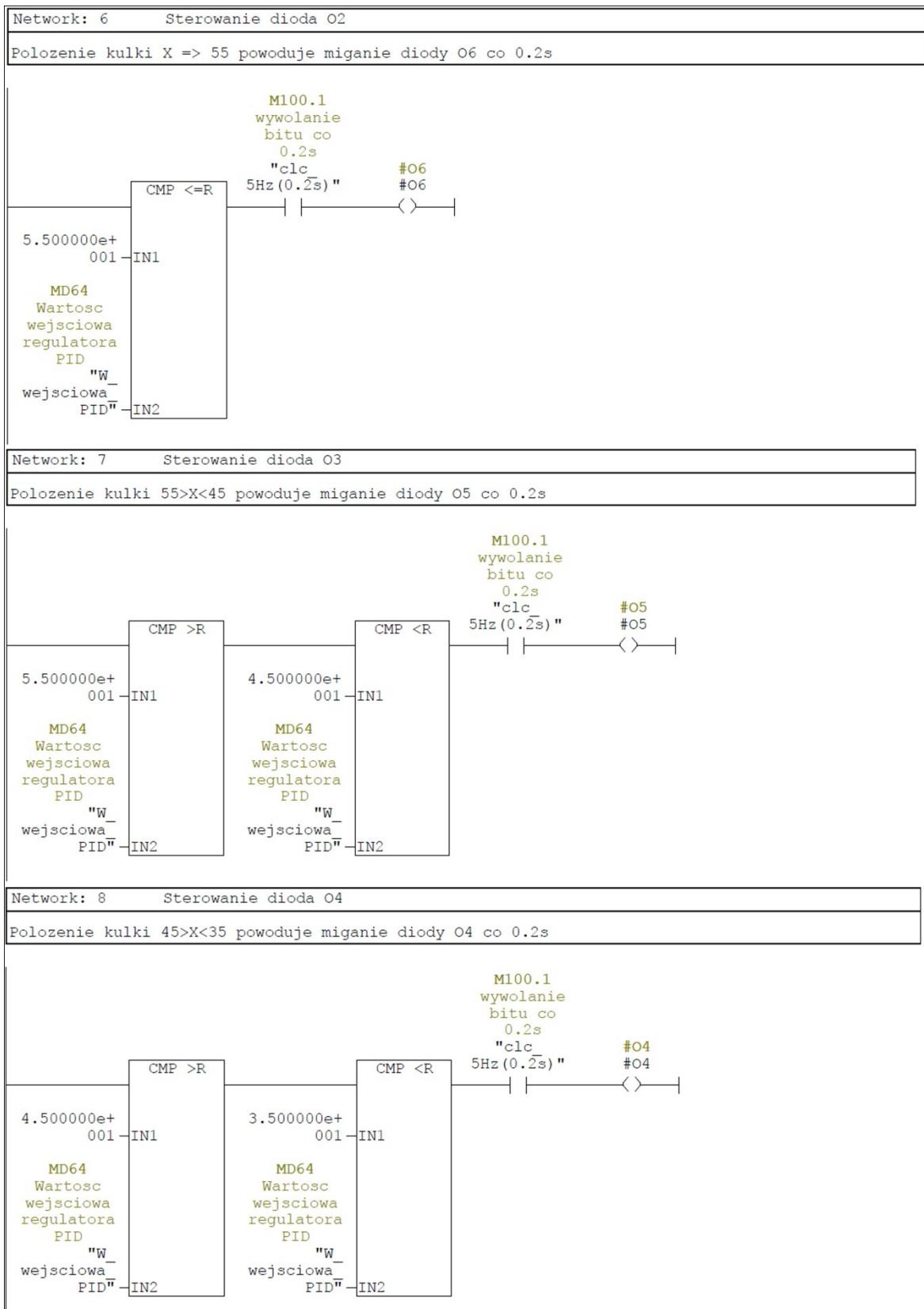
Wcisnąć przycisk I0 aktywowany zostaje znacznik START/STOP odpowiedzialny za aktywację głównych funkcji programu (PWM, PID, SEND, RECV), przycisk I1 (normalnie zamknięty) lub wywołanie błędu (brak kulki na pochylni) dezaktywuje znacznik START/STOP.

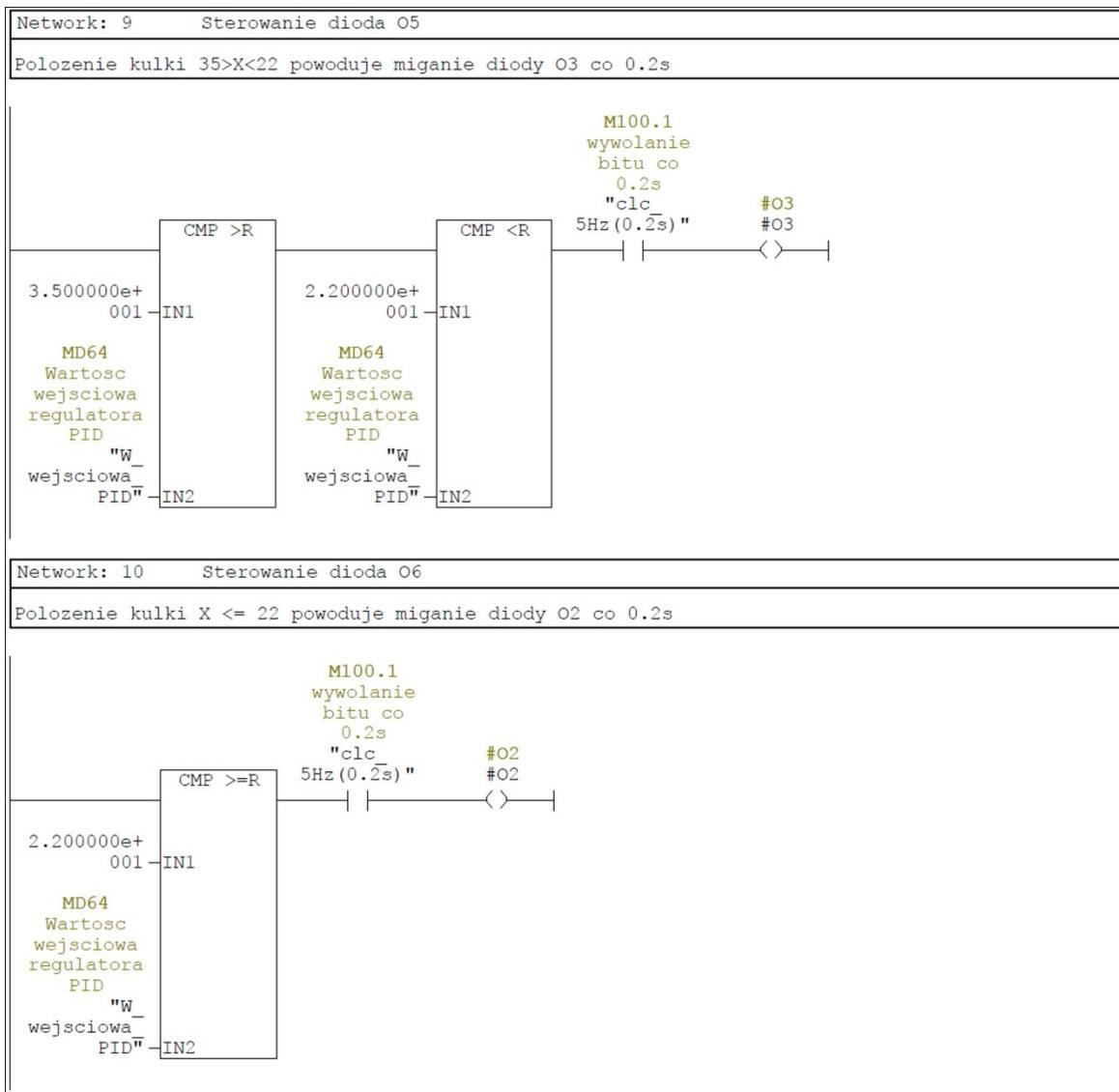
Gdy START/STOP aktywny zapala się dioda O0 a gaśnie O1, gdy START/STOP nieaktywny i error nieaktywny zapala się dioda O1 a gasnie O0.



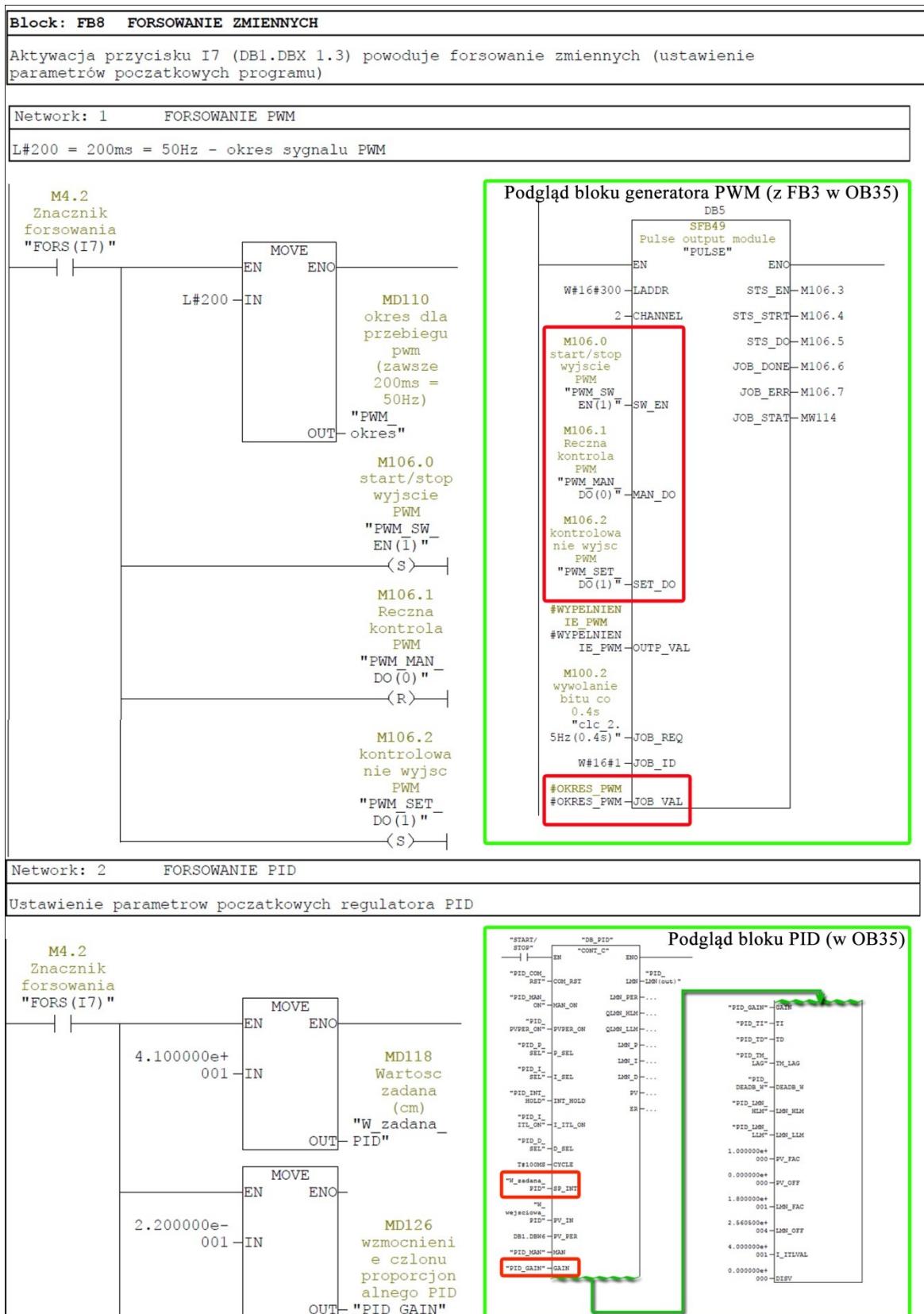
Układ sterowania czasu rzeczywistego położeniem kulki na pochylni z wykorzystaniem sieci PROFINET

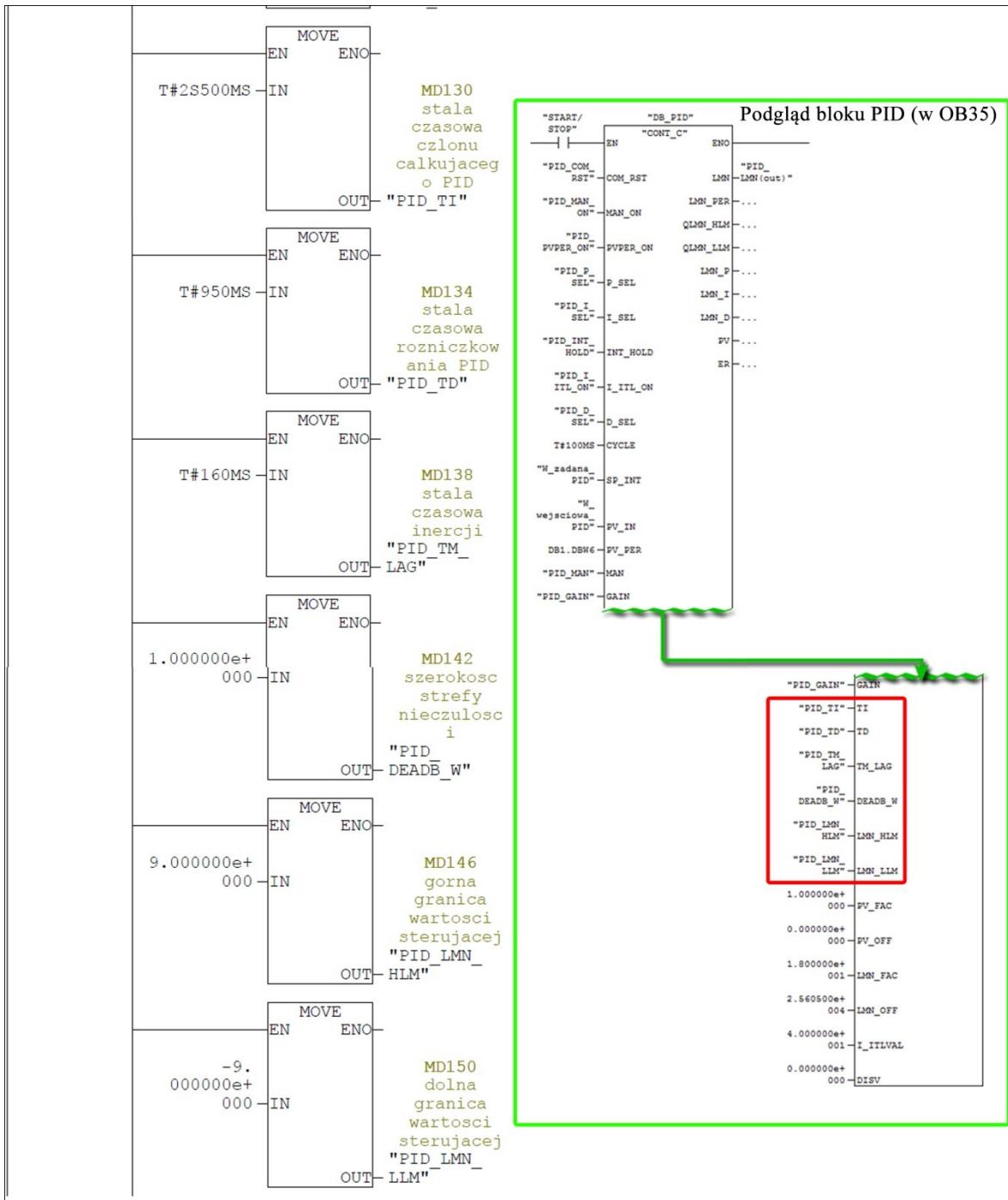


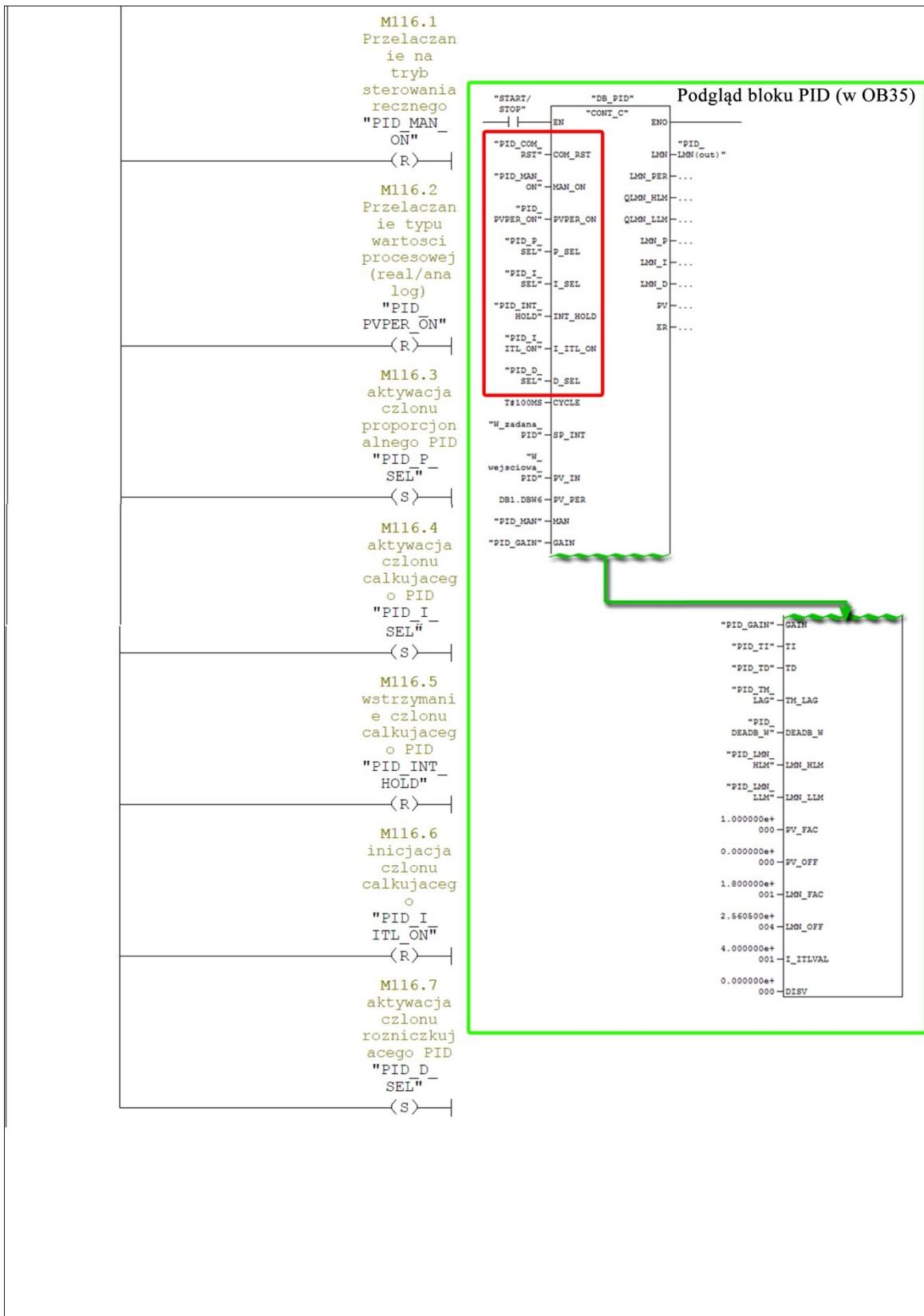




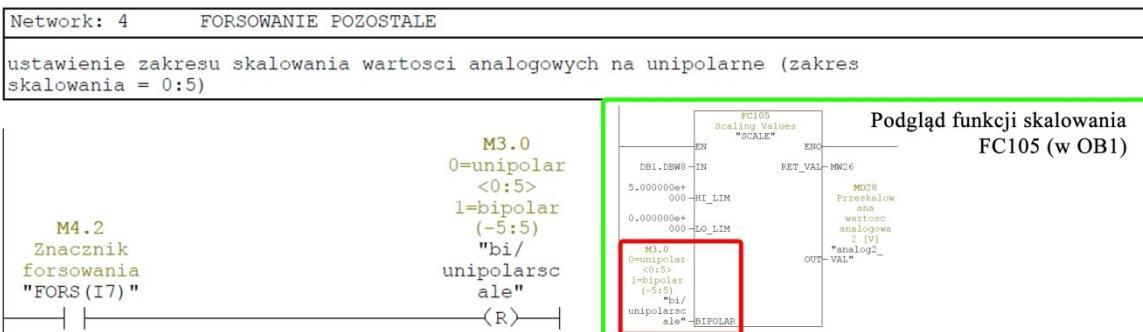
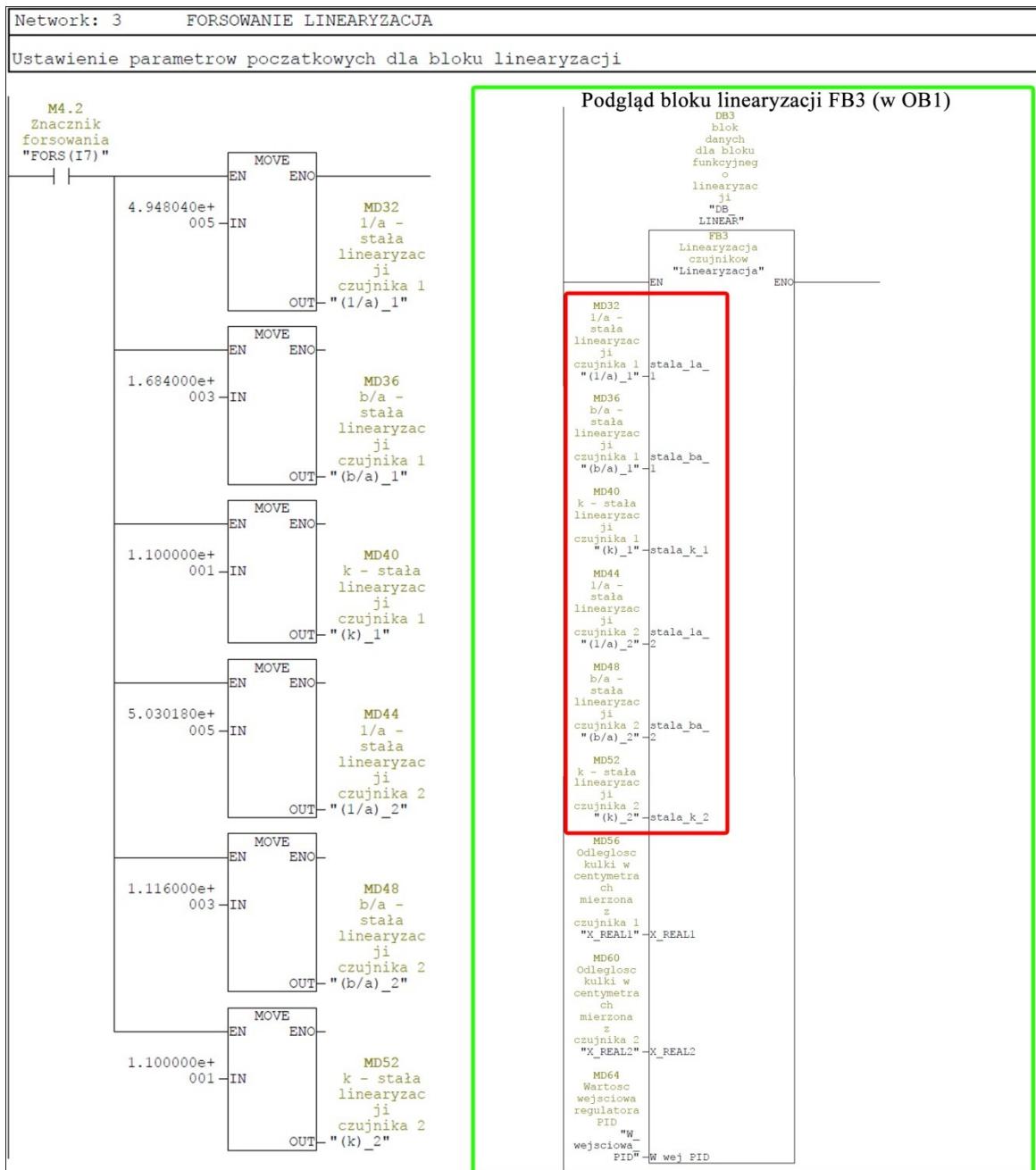
Z 1.3. Blok FB8 – Forsowanie zmiennych







Układ sterowania czasu rzeczywistego położeniem kulki na pochylni z wykorzystaniem sieci PROFINET



Z 1.4. Blok FB3 – Linearyzacja i sterowanie odczytem czujników

Parametry wejścia/wyjścia bloku FB3

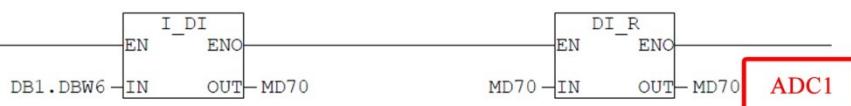
Name	Data Type	Address	Initial Value	Comment
IN		0.0		
stala_la_1	Real	0.0	5.882350e+005	
stala_ba_1	Real	4.0	3.000000e+003	
stala_k_1	Real	8.0	1.100000e+001	
stala_la_2	Real	12.0	5.882350e+005	
stala_ba_2	Real	16.0	3.000000e+003	
stala_k_2	Real	20.0	1.100000e+001	
OUT		0.0		
IN_OUT		0.0		
X_REAL1	Real	24.0	0.000000e+000	
X_REAL2	Real	28.0	0.000000e+000	
W_wej_PID	Real	32.0	0.000000e+000	
STAT		0.0		
TEMP		0.0		

Block: FB3 LINEARYZACJA CZUJNIKOW

Linearyzacja czujników metodą aproksymacji

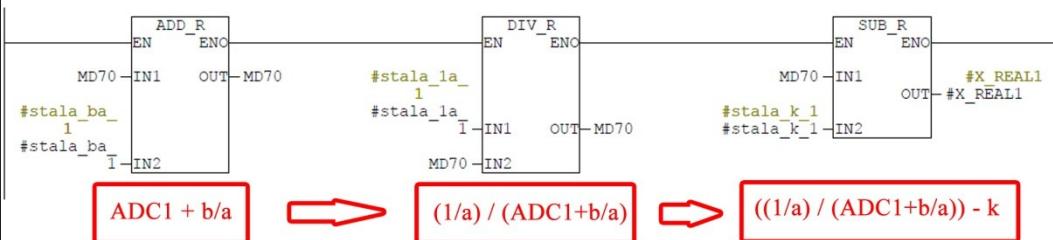
Network: 1 Linearyzacja czujnika 1

Konwersja sygnału ADC 1 z przetwornika cyfrowego z kodu hex na real



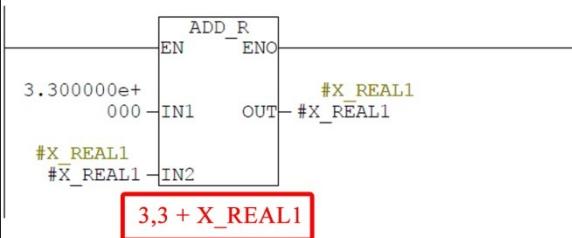
Network: 2 Linearyzacja czujnika 1

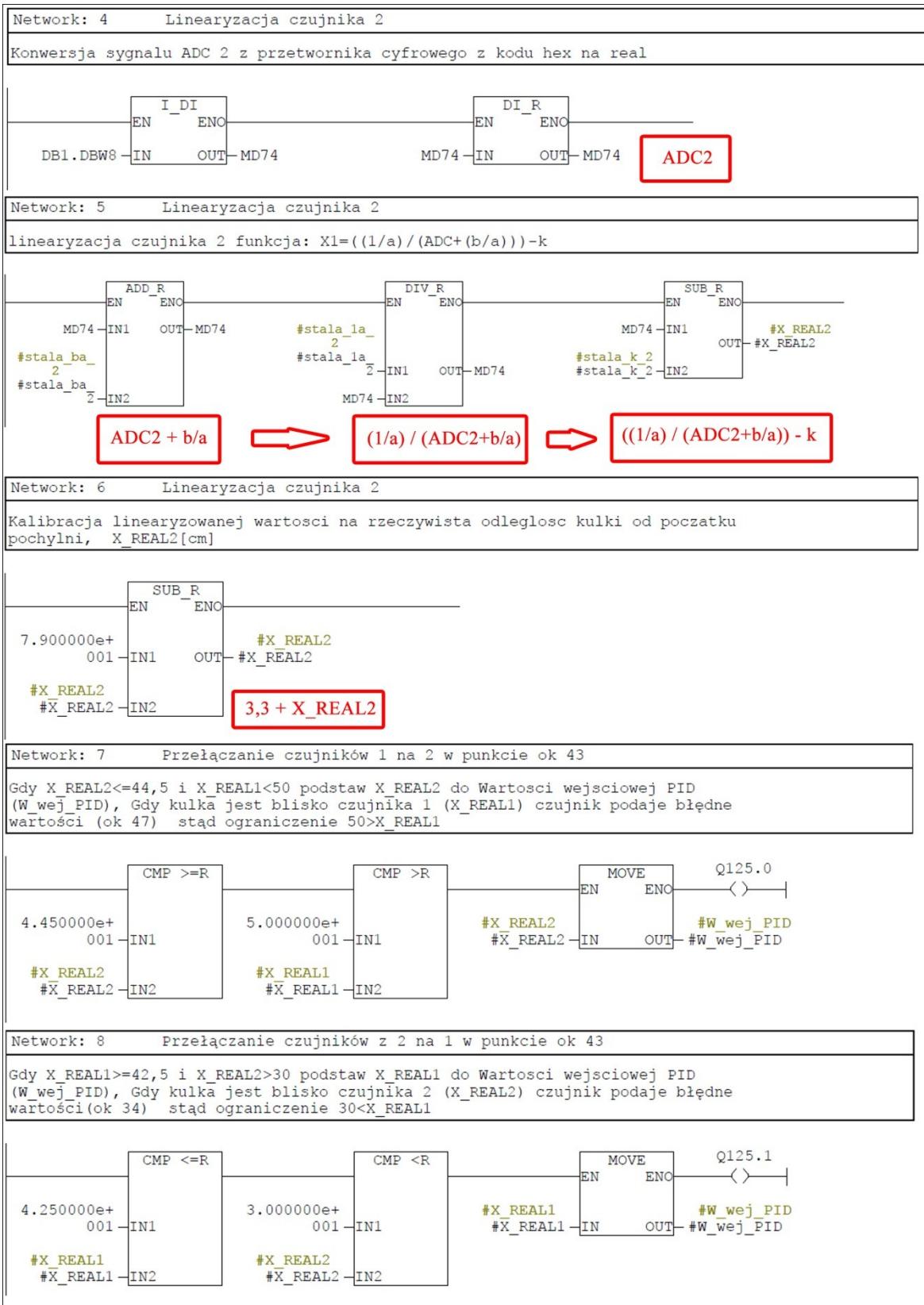
linearyzacja czujnika 1 funkcja: $X1=((1/a) / (ADC1+b/a)) - k$

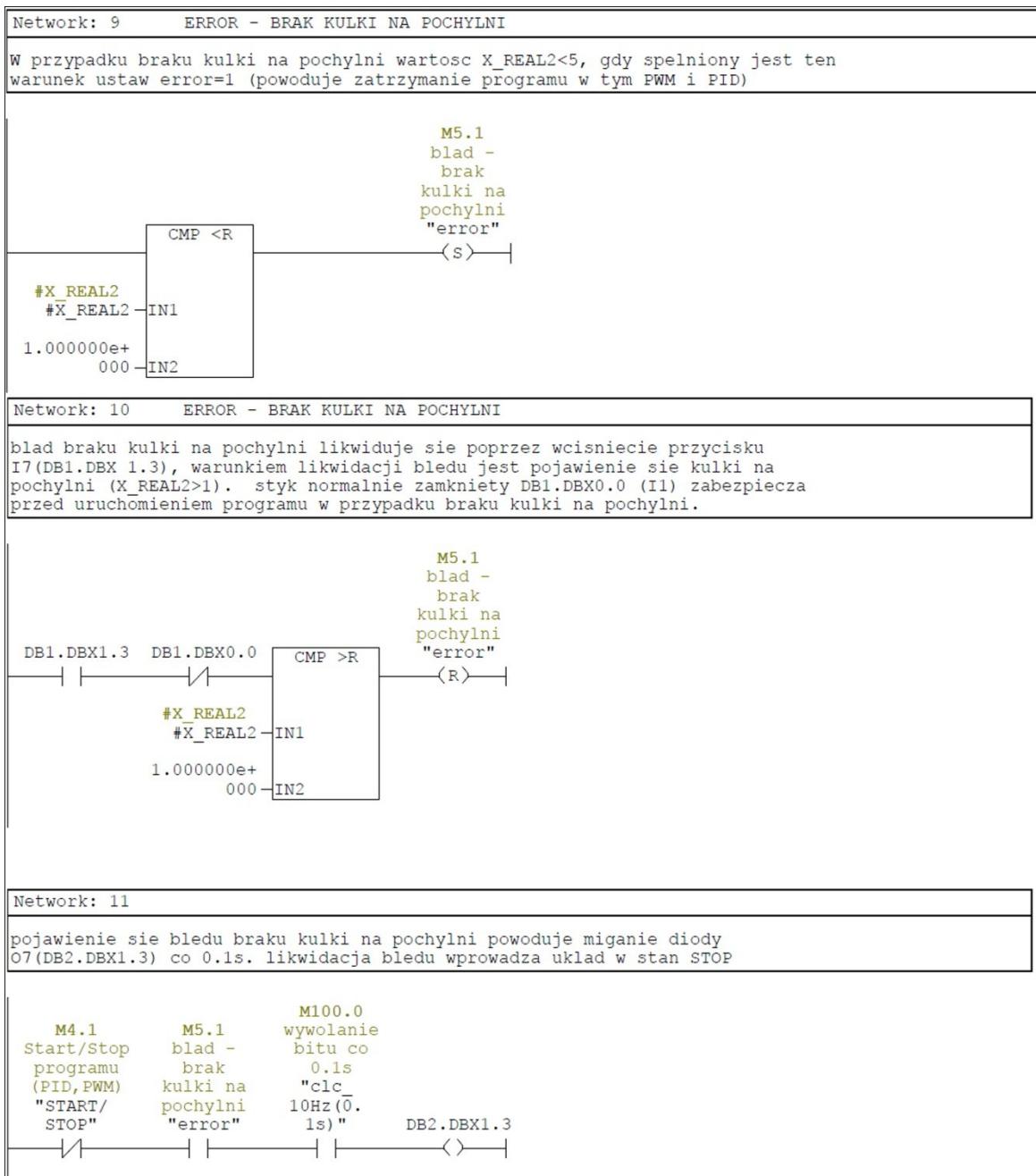


Network: 3

Kalibracja linearyzowanej wartości na rzeczywista odległość kulki od początku pochylni, $X_REAL1[\text{cm}]$



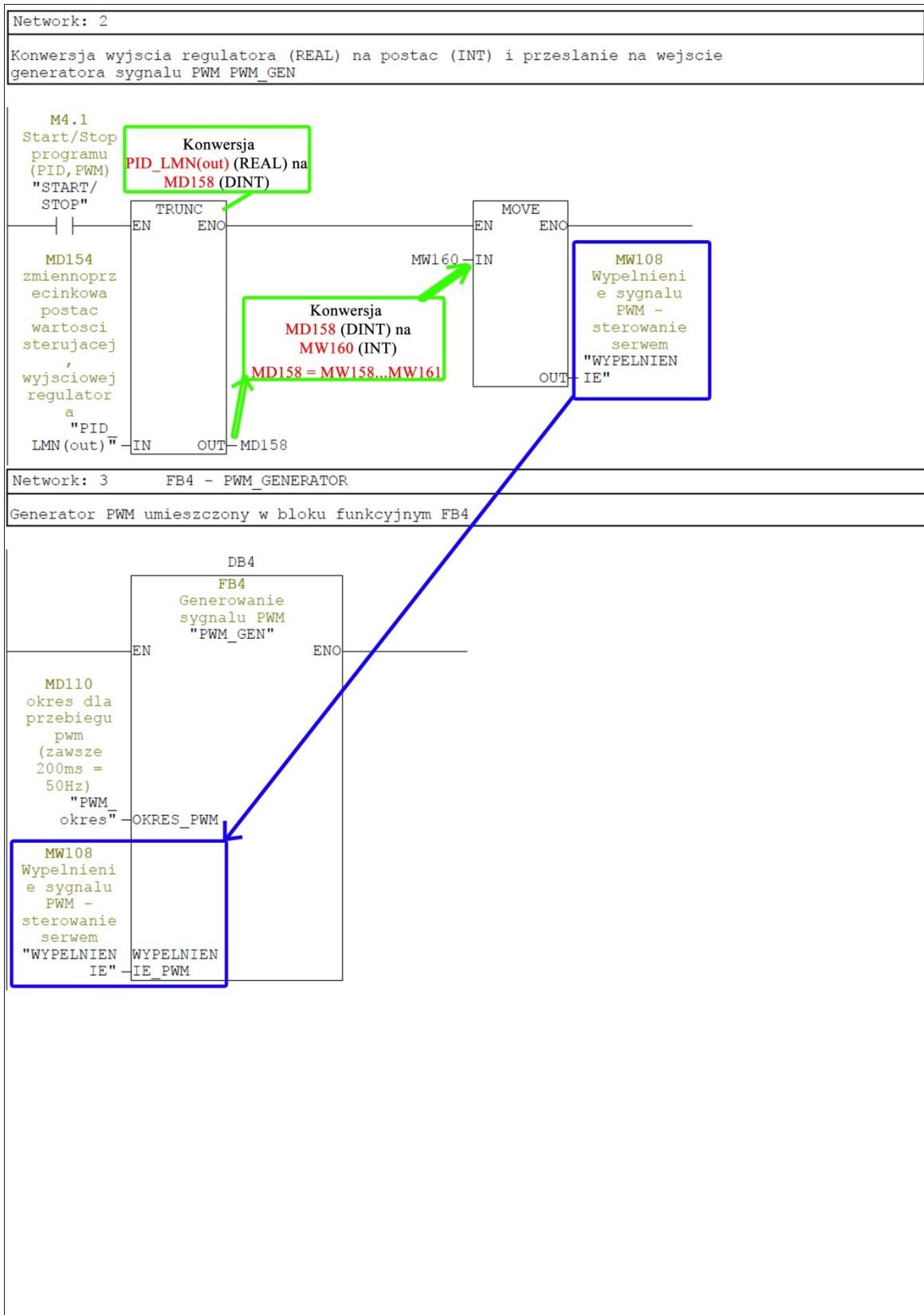




Z 1.5. Blok OB35 – Cykliczne przerwania

Block: OB35 "Cyclic Interrupt"																																																																																																														
Cykliczne wywołanie bloku OB35, co 100ms																																																																																																														
Network: 1 FB41 - CONT_C																																																																																																														
<p>Ciągły regulator PID, czas wywołania bloku PID musi być zgodny z wywołaniem bloku OB35 (CYCLE PID=100ms), gorna granica wartości sterującej (wychylenie pochylni) "PID_LMN_HLM" nie może przekraczać 15 stopni natomiast dolna "PID_LMN_LLM" nie mniej jak -15!</p>																																																																																																														
	<p>DB6 Tablica danych dla regulatora PID "DB_PID" FB41 Ciągły regulator PID "CONT_C"</p> <p>M4.1 Start/Stop programu (PID, FWM) "START/STOP" EN ENO</p> <p>M116.0 Reset regulatora PID "PID_COM_RST" COM_RST</p> <p>M116.1 Przelaczanie na tryb sterowania ręcznego "PID_MAN_ON" MAN_ON</p> <p>M116.2 Przelaczanie typu wartości procesowej (real/ana log) "PID_PVPER_ON" PVPER_ON</p> <p>M116.3 aktywacja członu proporcjonalnego PID "PID_P_SEL" SEL</p> <p>M116.4 aktywacja członu całkującego PID "PID_I_SEL" SEL</p> <p>M116.5 wstrzymanie członu całkującego PID "PID_INT_HOLD" INT_HOLD</p> <p>M116.6 inicjacja członu całkującego "PID_I_ITL_ON" ITL_ON</p> <p>M116.7 aktywacja członu różniczkującego PID "PID_D_SEL" SEL</p> <p>T#100MS CYCLE</p> <p>MD118 Wartość zadana (cm) "W_zadana_PID" SP_INT</p> <p>MD64 Wartość wejściowa regulatora PID "W_wejsciowa_PID" PV_IN</p>	<p>MD154 zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora "PID_LMN(out)"</p> <p>LMN PER</p> <p>QLMN_HLM</p> <p>QLMN_LLM</p> <p>LMN_P</p> <p>LMN_I</p> <p>LMN_D</p> <p>PV</p> <p>ER</p>																																																																																																												
<p>Tabela 10. Parametry funkcji FB 41 - CONT_C</p> <table border="1"> <thead> <tr> <th>Parametr</th> <th>Typ danych</th> <th>Opis</th> </tr> </thead> <tbody> <tr> <td>Parametry wejściowe</td> <td></td> <td></td> </tr> <tr> <td>COM_RST</td> <td>BOOL</td> <td>sygnał odpowiadający za restart regulatora</td> </tr> <tr> <td>MAN_ON</td> <td>BOOL</td> <td>aktywacja sygnału ręcznego sterowania dla TRUE aktywacja zmiennoprzecinkowa procesowych urządzeń peryferyjnych; jeżeli zmenna procesu jest odczytywana z urządzenia I/O, wejście PV_PER musi być podłączone do I/O. Sygnał sprawżenia zwrotnego będzie wtedy skalowany. Dla FALSE sygnał od razu w postaci liczby zmiennoprzecinkowej</td> </tr> <tr> <td>PVPER_ON</td> <td>BOOL</td> <td></td> </tr> <tr> <td>P_SEL</td> <td>BOOL</td> <td>aktywacja członu proporcjonalnego</td> </tr> <tr> <td>I_SEL</td> <td>BOOL</td> <td>aktywacja członu całkującego</td> </tr> <tr> <td>INT_HOLD</td> <td>BOOL</td> <td>wstrzymanie członu całkującego; dla TRUE wyjście tego członu będzie nieaktywne</td> </tr> <tr> <td>I_ITL_ON</td> <td>BOOL</td> <td>inicjacja członu całkującego</td> </tr> <tr> <td>D_SEL</td> <td>BOOL</td> <td>aktywacja członu różniczkującego</td> </tr> <tr> <td>CYCLE</td> <td>TIME</td> <td>cykl wywołania bloku FB41</td> </tr> <tr> <td>SP_INT</td> <td>REAL</td> <td>zmienna, która jest wartością zadaną</td> </tr> <tr> <td>PV_IN</td> <td>REAL</td> <td>zmiennoprzecinkowa wartość wejściowa</td> </tr> <tr> <td>MAN</td> <td>REAL</td> <td>wprowadzana wartość wejścia podczas sterowania ręcznego</td> </tr> <tr> <td>GAIN</td> <td>REAL</td> <td>wartość wzmacniania</td> </tr> <tr> <td>TI</td> <td>TIME</td> <td>stała całkowania</td> </tr> <tr> <td>TD</td> <td>TIME</td> <td>stała różniczkowania</td> </tr> <tr> <td>TM_LAG</td> <td>TIME</td> <td>opóźnienie członu różniczkującego</td> </tr> <tr> <td>DEADB_W</td> <td>REAL</td> <td>szerokość strefy martwej</td> </tr> <tr> <td>LMN_HLM</td> <td>REAL</td> <td>górną granicą wartości sterującej (wartości wyjścia)</td> </tr> <tr> <td>LMN_LLM</td> <td>REAL</td> <td>dolną granicą wartości sterującej (wartości wyjścia)</td> </tr> <tr> <td>PV_FAC</td> <td>REAL</td> <td>tutaj znajduje się mnożnik funkcji skalującej; opcjonalnie 1. Parametr jest stosowany do dopasowania zakresu zmienności sygnału z przetwornika pomiarowego do zakresu zmienności wartości zadanej SP_INT</td> </tr> <tr> <td>PV_OFF</td> <td>REAL</td> <td>wejście przesuwające funkcję skalującą w górę i w dół; opcjonalnie 0</td> </tr> <tr> <td>LMN_FAC, LMN_OFF</td> <td>REAL</td> <td>analitycznie jak PV, stosuje się je do dopasowania sygnału wyjściowego z regulatora do sygnału potrzebnego w obiekcie</td> </tr> <tr> <td>I_ITL_VAL</td> <td>REAL</td> <td>wartość inicjacji działania członu całkującego</td> </tr> <tr> <td>DISV</td> <td>REAL</td> <td>wejście dla sygnału zakłócającego</td> </tr> <tr> <td colspan="3">Parametry wyjściowe</td></tr> <tr> <td>LMN</td> <td>REAL</td> <td>zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora</td> </tr> <tr> <td>LMN_PER</td> <td>REAL</td> <td>sygnał wyjściowy regulatora na wejścia peryferyjne</td> </tr> <tr> <td>QLMN_HLM</td> <td>BOOL</td> <td>górną granicą wartości skalującej</td> </tr> <tr> <td>QLMN_LLM</td> <td>BOOL</td> <td>dolną granicą wartości skalującej</td> </tr> <tr> <td>LMN_P</td> <td>REAL</td> <td>komponent proporcjonalny</td> </tr> <tr> <td>LMN_I</td> <td>REAL</td> <td>komponent całkujący</td> </tr> <tr> <td>LMN_D</td> <td>REAL</td> <td>komponent różniczkujący</td> </tr> <tr> <td>PV</td> <td>REAL</td> <td>zmienna procesowa</td> </tr> <tr> <td>ER</td> <td>REAL</td> <td>sygnalizacja o błędzie</td> </tr> </tbody> </table>	Parametr	Typ danych	Opis	Parametry wejściowe			COM_RST	BOOL	sygnał odpowiadający za restart regulatora	MAN_ON	BOOL	aktywacja sygnału ręcznego sterowania dla TRUE aktywacja zmiennoprzecinkowa procesowych urządzeń peryferyjnych; jeżeli zmenna procesu jest odczytywana z urządzenia I/O, wejście PV_PER musi być podłączone do I/O. Sygnał sprawżenia zwrotnego będzie wtedy skalowany. Dla FALSE sygnał od razu w postaci liczby zmiennoprzecinkowej	PVPER_ON	BOOL		P_SEL	BOOL	aktywacja członu proporcjonalnego	I_SEL	BOOL	aktywacja członu całkującego	INT_HOLD	BOOL	wstrzymanie członu całkującego; dla TRUE wyjście tego członu będzie nieaktywne	I_ITL_ON	BOOL	inicjacja członu całkującego	D_SEL	BOOL	aktywacja członu różniczkującego	CYCLE	TIME	cykl wywołania bloku FB41	SP_INT	REAL	zmienna, która jest wartością zadaną	PV_IN	REAL	zmiennoprzecinkowa wartość wejściowa	MAN	REAL	wprowadzana wartość wejścia podczas sterowania ręcznego	GAIN	REAL	wartość wzmacniania	TI	TIME	stała całkowania	TD	TIME	stała różniczkowania	TM_LAG	TIME	opóźnienie członu różniczkującego	DEADB_W	REAL	szerokość strefy martwej	LMN_HLM	REAL	górną granicą wartości sterującej (wartości wyjścia)	LMN_LLM	REAL	dolną granicą wartości sterującej (wartości wyjścia)	PV_FAC	REAL	tutaj znajduje się mnożnik funkcji skalującej; opcjonalnie 1. Parametr jest stosowany do dopasowania zakresu zmienności sygnału z przetwornika pomiarowego do zakresu zmienności wartości zadanej SP_INT	PV_OFF	REAL	wejście przesuwające funkcję skalującą w górę i w dół; opcjonalnie 0	LMN_FAC, LMN_OFF	REAL	analitycznie jak PV, stosuje się je do dopasowania sygnału wyjściowego z regulatora do sygnału potrzebnego w obiekcie	I_ITL_VAL	REAL	wartość inicjacji działania członu całkującego	DISV	REAL	wejście dla sygnału zakłócającego	Parametry wyjściowe			LMN	REAL	zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora	LMN_PER	REAL	sygnał wyjściowy regulatora na wejścia peryferyjne	QLMN_HLM	BOOL	górną granicą wartości skalującej	QLMN_LLM	BOOL	dolną granicą wartości skalującej	LMN_P	REAL	komponent proporcjonalny	LMN_I	REAL	komponent całkujący	LMN_D	REAL	komponent różniczkujący	PV	REAL	zmienna procesowa	ER	REAL	sygnalizacja o błędzie		
Parametr	Typ danych	Opis																																																																																																												
Parametry wejściowe																																																																																																														
COM_RST	BOOL	sygnał odpowiadający za restart regulatora																																																																																																												
MAN_ON	BOOL	aktywacja sygnału ręcznego sterowania dla TRUE aktywacja zmiennoprzecinkowa procesowych urządzeń peryferyjnych; jeżeli zmenna procesu jest odczytywana z urządzenia I/O, wejście PV_PER musi być podłączone do I/O. Sygnał sprawżenia zwrotnego będzie wtedy skalowany. Dla FALSE sygnał od razu w postaci liczby zmiennoprzecinkowej																																																																																																												
PVPER_ON	BOOL																																																																																																													
P_SEL	BOOL	aktywacja członu proporcjonalnego																																																																																																												
I_SEL	BOOL	aktywacja członu całkującego																																																																																																												
INT_HOLD	BOOL	wstrzymanie członu całkującego; dla TRUE wyjście tego członu będzie nieaktywne																																																																																																												
I_ITL_ON	BOOL	inicjacja członu całkującego																																																																																																												
D_SEL	BOOL	aktywacja członu różniczkującego																																																																																																												
CYCLE	TIME	cykl wywołania bloku FB41																																																																																																												
SP_INT	REAL	zmienna, która jest wartością zadaną																																																																																																												
PV_IN	REAL	zmiennoprzecinkowa wartość wejściowa																																																																																																												
MAN	REAL	wprowadzana wartość wejścia podczas sterowania ręcznego																																																																																																												
GAIN	REAL	wartość wzmacniania																																																																																																												
TI	TIME	stała całkowania																																																																																																												
TD	TIME	stała różniczkowania																																																																																																												
TM_LAG	TIME	opóźnienie członu różniczkującego																																																																																																												
DEADB_W	REAL	szerokość strefy martwej																																																																																																												
LMN_HLM	REAL	górną granicą wartości sterującej (wartości wyjścia)																																																																																																												
LMN_LLM	REAL	dolną granicą wartości sterującej (wartości wyjścia)																																																																																																												
PV_FAC	REAL	tutaj znajduje się mnożnik funkcji skalującej; opcjonalnie 1. Parametr jest stosowany do dopasowania zakresu zmienności sygnału z przetwornika pomiarowego do zakresu zmienności wartości zadanej SP_INT																																																																																																												
PV_OFF	REAL	wejście przesuwające funkcję skalującą w górę i w dół; opcjonalnie 0																																																																																																												
LMN_FAC, LMN_OFF	REAL	analitycznie jak PV, stosuje się je do dopasowania sygnału wyjściowego z regulatora do sygnału potrzebnego w obiekcie																																																																																																												
I_ITL_VAL	REAL	wartość inicjacji działania członu całkującego																																																																																																												
DISV	REAL	wejście dla sygnału zakłócającego																																																																																																												
Parametry wyjściowe																																																																																																														
LMN	REAL	zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora																																																																																																												
LMN_PER	REAL	sygnał wyjściowy regulatora na wejścia peryferyjne																																																																																																												
QLMN_HLM	BOOL	górną granicą wartości skalującej																																																																																																												
QLMN_LLM	BOOL	dolną granicą wartości skalującej																																																																																																												
LMN_P	REAL	komponent proporcjonalny																																																																																																												
LMN_I	REAL	komponent całkujący																																																																																																												
LMN_D	REAL	komponent różniczkujący																																																																																																												
PV	REAL	zmienna procesowa																																																																																																												
ER	REAL	sygnalizacja o błędzie																																																																																																												

Tabela 10. Parametry funkcji FB 41 - CONT_C		
Parametr	Typ danych	Opis
Parametry wejściowe		
COM_RST	BOOL	sygnał odpowiedzialny za restart regulatora
MAN_ON	BOOL	aktywacja sygnału ręcznego sterowania
PVPER_ON	BOOL	dla TRUE aktywacja zmiennoprzecinkowa procesowych urządzeń peryferyjnych; jeżeli zmienna procesu jest odczytywana z urządzenia I/O, wejście PV_PER musi być podłączone do I/O. Sygnał sprzężenia zwrotnego będzie wtedy skalowany. Dla FALSE sygnał od razu w postaci liczby zmiennoprzecinkowej
P_SEL	BOOL	aktywacja członu proporcjonalnego
I_SEL	BOOL	aktywacja członu całkującego
INT_HOLD	BOOL	wstrzymanie członu całkującego; dla TRUE wyjście tego członu będzie nieaktywne
I_ITL_ON	BOOL	inicjacja członu całkującego
D_SEL	BOOL	aktywacja członu różniczkującego
CYCLE	TIME	cykl wywołania bloku FB41
SP_INT	REAL	zmienna, która jest wartością zadaną
PV_IN	REAL	zmiennoprzecinkowa wartość wejściowa
MAN	REAL	wprowadzana wartość wejścia podczas sterowania ręcznego
GAIN	REAL	wartość wzmacniania
TI	TIME	stała całkowania
TD	TIME	stała różniczkowania
TM_LAG	TIME	opóźnienie członu różniczkującego
DEADB_W	REAL	szerokość strefy martwej
LMN_HLM	REAL	górną granicą wartości sterującej (wartości wyjścia)
LMN_LLM	REAL	dolną granicą wartości sterującej (wartości wyjścia)
PV_FAC	REAL	tutaj znajduje się mnożnik funkcji skalującej; opcjonalnie 1. Parametr jest stosowany do dopasowania zakresu zmienności sygnału z przetwornika pomiarowego do zakresu zmienności wartości zadanej SP_INT
PV_OFF	REAL	wejście przesuwające funkcję skalującą w górę i w dół; opcjonalnie 0
LMN_FAC,	REAL	analitycznie jak PV, stosuje się je do dopasowania sygnału wyjściowego z regulatora do sygnału potrzebnego w obiekcie
LMN_OFF		
I_ITL_VAL	REAL	wartość inicjacji działania członu całkującego
DISV	REAL	wejście dla sygnału zakłócającego
Parametry wyjściowe		
LMN	REAL	zmiennoprzecinkowa postać wartości sterującej, wyjściowej regulatora
LMN_PER	REAL	sygnał wyjściowy regulatora na wejścia peryferyjne
QLMN_HLM	BOOL	górną granicą wartości skalującej
QLMN_LLM	BOOL	dolną granicą wartości skalującej
LMN_P	REAL	komponent proporcjonalny
LMN_I	REAL	komponent całkujący
LMN_D	REAL	komponent różniczkujący
PV	REAL	zmienna procesowa
ER	REAL	sygnalizacja o błędzie



Z 1.6. Blok FB4 – Generowanie sygnału PWM

Parametry wejścia/wyjścia bloku FB4:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
OKRES_PWM	DIInt	0.0	L#200	
OUT		0.0		
IN_OUT		0.0		
WYPELNIENIE_PWM	Int	4.0	0	
STAT		0.0		
TEMP		0.0		

Block: FB4 GENERATOR PWM - PULSE SFB49

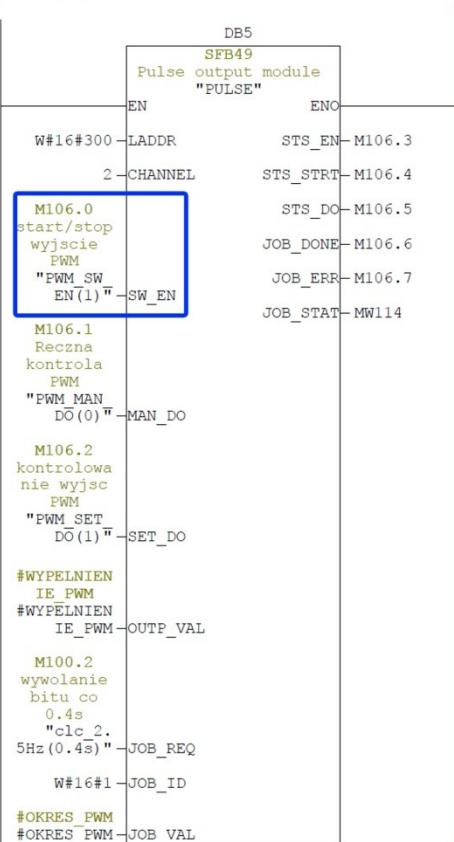
Network: 1 start/stop wyjście PWM

załączanie PWM



Network: 2 SFB 49 - GENEROWANIE SYGNALU PWM

Blok SFB49 odpowiedzialny za generowanie sygnału PWM, szerokość pulsu sygnału PWM dyktuje wyjście regulatora PID - LMN, JOB_VAL (okres PWM) zawsze = 200 (200*0,1ms=20ms=50Hz)



Rys. 4.15. Parametry PWM - HW Config/CPU Count Properties/Pulse-width modulation.

Tabela 9. Parametry funkcji SFB 49 - PULSE/

SFB 49 - PULSE		
Parametr	Typ danych	Opis
Parametry wejściowe		
LADDR	WORD	Adres szybkiego licznika PLC odczytany z HW Config
CHANNEL	INT	Zarezerwowane wyjście dla szybkiego licznika w sterowniku PLC
SW_EN	BOOL	START/STOP wyjścia PULSE
MAN_DO	BOOL	Ustawianie ręcznej kontroli nad wyjściem
SET_DO	BOOL	Kontrola wyjścia
OUTP_VAL	INT	Ustawienie wartości wyjścia w zależności od pracy funkcji <u>JOB_ID</u> oraz formatu wyjścia ustawionego w HW Config - Output format
JOB_REQ	BOOL	Inicjacja <u>JOB_ID</u>
JOB_ID	WORD	O określa tryb pracy funkcji PULSE, 0-praca bez funkcji, 1 - sterowanie okresem i wypełnieniem, 4 - sterowanie minimalnym czasem trwania pulsu, 81 - odczyt okresu i wypełnienia, 84 - odczyt minimalnego czasu trwania pulsu
JOB_VAL	DINT	Wartość dla pracy <u>JOB_ID</u>
Parametry wyjściowe		
STS_EN	BOOL	Status wykonanej pracy
STS_STRT	BOOL	Status startu pracy
STS_DO	BOOL	Status wyjścia
JOB_DONE	BOOL	Informuje (1) gdy nowa praca może zostać wykonana
JOB_ERR	BOOL	Błąd pracy
JOB_STAT	WORD	Numer błędu pracy

Z 1.7. Blok OB100 – restart regulatora PID (STL)

Block: OB100 "Complete Restart"
restart regulatora PID w przypadku ponownego uruchomienia sterownika w celu wykasowania starych danych.
Network: 1 RESTART REGULATORA PID
Restart regulatora PID - blok OB100 realizuje się tylko raz przy uruchamianiu sterownika ale przed uruchomieniem bloku OB1,
<pre>CALL "CONT_C" , "DB_PID" FB41 / DB6 -- Ciagly regulator PID / Tablica danych dla regulatora PID COM_RST :=TRUE MAN_ON := PVPER_ON := P_SEL := I_SEL := INT_HOLD := I_ITL_ON := D_SEL := CYCLE :=T#100MS SP_INT := PV_IN := PV_PER := MAN := GAIN := TI := TD := TM_LAG := DEADB_W := LMN_HLM := LMN_LLM := PV_FAC := PV_OFF := LMN_FAC := LMN_OFF := I_ITLVAL := DTSV := LMN := LMN_PER := QLMN_HLM:= QLMN_LLM:= LMN_F := LMN_I := LMN_D := PV := ER :=</pre>
Network: 2 complete restart
przywrocenie pracy PID
<pre>CLR = "DB_PID".COM_RST DB6.DBX0.0 -- complete restart</pre>