



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ
KATEDRA AUTOMATYKI I INŻYNIERII BIOMEDYCZNEJ**

Praca dyplomowa inżynierska

**Wykorzystanie metody generacji kodu ST do sterowania
modelem laboratoryjnym kaskady zbiorników**

*Utilizing ST code generation method for control a laboratory
model of cascade tanks system*

Autor:	<i>Michał Kiniorski</i>
Kierunek studiów:	Automatyka i Robotyka
Opiekun pracy:	<i>dr inż. Maciej Rosół</i>

Kraków, 2016 rok

Uprzedzony o odpowiedzialności karnej na podstawie art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2006 r. Nr 90, poz. 631z późn. zm.): „ Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystyczne wykonanie albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, wideogram lub nadanie.”, a także uprzedzony o odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (t.j. Dz. U. z 2012 r. poz. 572, z późn. zm.) „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności studenta student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej „sądem koleżeńskim”, oświadczam, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i że nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.

.....

*Serdeczne podziękowania
dla Pana dr inż. Macieja
Rosoła za cierpliwość oraz
pomoc przy realizacji
niniejszej pracy*

Spis treści

1	WSTĘP.....	5
2	OPIS NARZĘDZI PROGRAMOWYCH I SPRZĘTOWYCH.....	7
2.1	ŚRODOWISKO MATLAB/SIMULINK.	7
2.2	ŚRODOWISKO PC WORX DLA STEROWNIKÓW PHOENIX CONTACT.....	7
2.3	PLC CODER.	8
2.4	STANOWISKO LABORATORYJNE.	8
2.4.1	<i>Połączenie obiektu z komputerem.</i>	<i>9</i>
2.4.2	<i>Opis sterownika PLC.</i>	<i>10</i>
3	IDENTYFIKACJA.	13
3.1	MODEL MATEMATYCZNY.	13
3.2	SKALOWANIE SYGNAŁÓW Z CZUJNIKÓW CIŚNIENIA.....	16
3.3	CHARAKTERYSTYKA PRZETWARZANIA POMPY.	18
3.4	IDENTYFIKACJA WSPÓŁCZYNNIKÓW WYPŁYWU.	21
4	SYNTEZA REGULATORA LQ.	25
4.1	LINEARYZACJA MODELU.	25
4.2	REGULATOR LQ.....	28
4.3	REALIZACJA REGULATORA LQ W ŚRODOWISKU MATLAB SIMULINK.	29
4.4	TESTY SYMULACYJNE.	31
5	GENERACJA KODU ST ORAZ TESTY NA STEROWNIKU	36
5.1	OBSŁUGA MODUŁÓW STEROWNIKA.....	36
5.2	OPIS PROGRAMU URUCHAMIANEGO NA STEROWNIKU.	39
5.3	TESTY NA OBIEKCIE RZECZYWISTYM.	40
6	PODSUMOWANIE	47
	BIBLIOGRAFIA	48
	SPIS RYSUNKÓW	50
	SPIS TABEL.....	52

1 Wstęp

Automatyzacja procesów technologicznych jest bardzo powszechna w dzisiejszych czasach. Przy budowie systemów automatyki przemysłowej nieodłącznym elementem są sterowniki PLC. Pozwalają one na wdrożenie rozbudowanego algorytmu sterowania, który umożliwia zaplanowane działanie danego procesu. Dodatkowo, moduły komunikacyjne wraz z panelami operatorskimi stwarzają możliwości do projektowania systemów SCADA (Supervisory Control And Data Acquisition), nadzorujących cykl technologiczny bądź produkcyjny i dokonujących wizualizacji istotnych parametrów dostępnych dla operatora.

Wykonanie algorytmu sterowania i regulacji na sterownikach PLC jest możliwe przy użyciu języka drabinkowego LD, strukturalnego ST, grafów sekwencji SFC oraz innych języków zgodnych z normą IEC 61131-3. Projektowanie złożonych systemów technologicznych, używając wyżej wymienionych narzędzi, należy do czynności czasochłonnych. Innym problem jest to, że nie wszystkie środowiska programistyczne posiadają możliwość przeprowadzenia symulacji stworzonej aplikacji przed wprowadzeniem jej do rzeczywistości. W takich przypadkach dobrą alternatywą jest skorzystanie z pakietu MATLAB/Simulink, który dzięki graficznej nakładce, pozwala na tworzenie schematów przy wykorzystaniu biblioteki z dużą liczbą wbudowanych bloków. Projektowanie w tym oprogramowaniu zalicza się do wysoce intuicyjnych. Ogromnym atutem w kontekście powiązania pomiędzy MATLABem, a programowalnymi sterownikami logicznymi jest dostępność przybornika PLC Coder. Jest to narzędzie generujące kod Structured Text (ST), zrozumiały dla sterownika. Podział na część symulacyjną i projektową udostępnia większe możliwości do odpowiedniego testowania, a następnie weryfikowania stworzonego systemu sterowania.

Celem niniejszej pracy jest projekt i wykonanie układu sterowania modelem laboratoryjnym kaskady zbiorników z użyciem sterownika PLC firmy Phoenix Contact. Do realizacji oprogramowania zostaną wykorzystane środowiska PC Worx i MATLAB/Simulink z możliwością automatycznej generacji kodu ST. Autor przeprowadzi identyfikację parametrów modelu matematycznego obiektu i na jego podstawie dokona syntezy algorytmu sterowania. Zadaniem tego algorytmu jest stabilizacja poziomów cieczy w zbiornikach na zadanych wartościach. Przewiduje się wykonanie graficznego interfejsu użytkownika. Końcowy etap to weryfikacja uzyskanych wyników na stanowisku laboratoryjnym z wykorzystaniem sterownika PLC typu ILC350PN. Jako kanał komunikacyjny zastosowano standard przemysłowy Profinet.

2 Opis narzędzi programowych i sprzętowych.

Każda praca naukowo-techniczna wymaga wiedzy na temat znajomości oprogramowania i sprzętu, które zostaną użyte. W tym rozdziale przedstawiono wybrane środowiska programowe, najbardziej przyjazne do symulacji zachowania systemów dynamicznych, a następnie weryfikacji algorytmu sterowania na docelowej platformie sprzętowej. W kolejnych podrozdziałach opisano programy wykorzystane w pracy oraz przedstawiono zwięzły opis stanowiska laboratoryjnego.

2.1 Środowisko MATLAB/Simulink.

Środowisko MATLAB/Simulink to narzędzie do tworzenia i analizy działania modeli symulacyjnych systemów dynamicznych. Graficzny interfejs użytkownika dostępny w Simulinku wspiera konstruowanie modeli w postaci schematów blokowych. Biblioteka zawiera ogromną ilość zasobów niezbędnych do zaprojektowania systemu automatyki. Zastosowanie podsystemów pozwala ustalić hierarchię w modelu, sprawiając, że zarządzanie stworzonym programem jest o wiele łatwiejsze. Istnieje inny wariant do budowania bardziej rozbudowanych systemów logicznych, który wykorzystuje przybornik Stateflow. Do głównych zalet programu zalicza się [1]:

- generację kodu C/HDL i innych na podstawie podsystemów diagramu Simulinka,
- hierarchizację komponentów,
- dostępność znacznej liczby tzw. *solverów* ODE,
- łatwe diagnozowanie błędów przy użyciu debugera.

2.2 Środowisko PC Worx dla sterowników Phoenix Contact.

Oprogramowanie PC Worx to złożony pakiet pozwalający na skonfigurowanie i zaprogramowanie wszystkich sterowników wyprodukowanych przez firmę Phoenix Contact. Należy ono do narzędzi rozbudowanych, cechujących się integracją poszczególnych komponentów. Zawiera ono całkowite wsparcie dla protokołów magistrali danych, takich jak: Interbus, Profinet, Profibus, Modbus-TCP oraz DeviceNet. Obsługa tych protokołów umożliwia tworzenie wielu sieci PLC w obrębie jednego projektu. Dodatkowo w programie dostępny jest widok prezentujący sieć w postaci drzewa. Twórcy oprogramowania ułatwili programistą tworzenie kodu poprzez wprowadzenie adresowania względnego oraz automatycznego uzupełniania tekstu w trakcie edytowania. Narzędzie pozwala na wprowadzanie zmian w trybie online. W czasie działania programu wgranego na sterownik,

można dołączyć nowe zmienne zarówno lokalne, jak i globalne oraz tworzyć nowe funkcje i bloki funkcyjne. Okna kontrolne oraz lista zmiennych, których wartości mogą być wymuszane, sprawiają, że łatwiej o prawidłową diagnozę stworzonego programu. Wbudowane narzędzia diagnostyczne zapewniają szeroki wachlarz możliwości, począwszy od tworzenia punktów przerwań, a kończąc na sprawdzaniu całych instancji. Wspieranymi przez PC Worx językami programowania są [2]:

- lista instrukcji (IL),
- język bloków funkcyjnych (FBD),
- grafy sekwencji (SFC),
- tekst strukturalny (ST),
- język drabinkowy (LD).

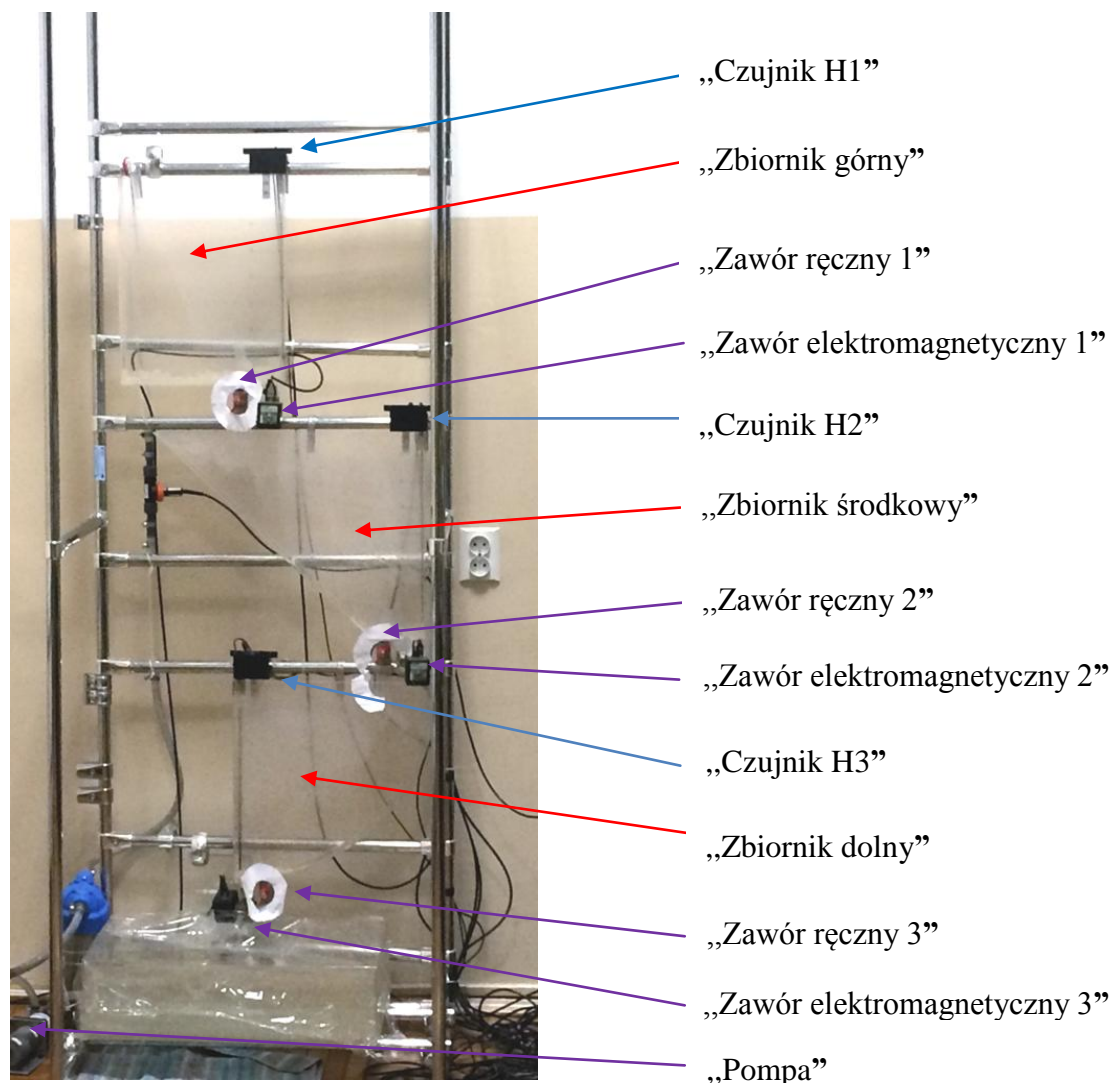
2.3 PLC Coder.

PLC Coder jest narzędziem dostępnym w Simulinku. Umożliwia on wytwarzanie kodu C/C++, HDL oraz ST [3]. Biorąc pod uwagę zakres pracy najistotniejsza jest ostatnia opcja, która umożliwia tworzenie kodu z użyciem przybornika PLC Coder zgodnie z normą IEC 61131-3. Narzędzie to wspomaga ponad 180 bloków Simulinka oraz dużą liczbę funkcji dostępnych w MATLABie. Korzystając z PLC Coder można dokonać kompleksowej implementacji systemu sterowania opartego na stanach czy zaawansowanych matematycznie algorytmach. Zapewnia on optymalizację kodu wynikowego, która redukuje pamięć potrzebną do działania aplikacji, a przy okazji przyspiesza czas przeprowadzenia generacji tego kodu. Wbudowany identyfikator kontroli nazewnictwa pozwala na zachowanie nazw obiektów i sygnałów stworzonych w modelu. Tworzony jest raport wygenerowanego kodu, zawierający opis interfejsu, raport śledzenia oraz prezentację generowanych plików źródłowych. Dodatkowo PLC Coder udostępnia wgląd do listy składającej się z informacji o ilości linii kodu, zmiennych i stałych globalnych. Na końcu można skorzystać z kontrolera testu tzw. *Test harness*. Zezwala on na weryfikację uzyskanego kodu ST przez porównanie zachowania modelu symulacyjnego i aplikacji w języku ST. Analizę wyników przeprowadza się, używając środowiska programistycznego zintegrowanego z PLC.

2.4 Stanowisko laboratoryjne.

Realizacja pracy była możliwa, dzięki skorzystaniu ze stanowiska laboratoryjnego kaskady zbiorników, znajdującego się w Katedrze Automatyki i Inżynierii Biomedycznej AGH. Składało się ono z komputera PC, sterownika PLC wraz z niezbędnymi modułami I/O

oraz obiektu sterowania. Obiekt sterowania stanowi kaskadowe połączenie trzech zbiorników (górnego, środkowego i dolnego) o różnym kształcie i wymiarach geometrycznych. Każdy zbiornik wyposażono w piezorezystancyjny czujnik ciśnienia, służący do pomiaru poziomu cieczy oraz dwa typy zaworów (elektromagnetyczny i ręczny). Za pomocą pompy, napędzanej przez silnik prądu stałego, dostarczano wodę do górnego zbiornika. Źródłem cieczy był prostopadłościenny zbiornik umiejscowiony poniżej dolnego zbiornika. Na rysunku 1 przedstawiono zdjęcie obiektu.



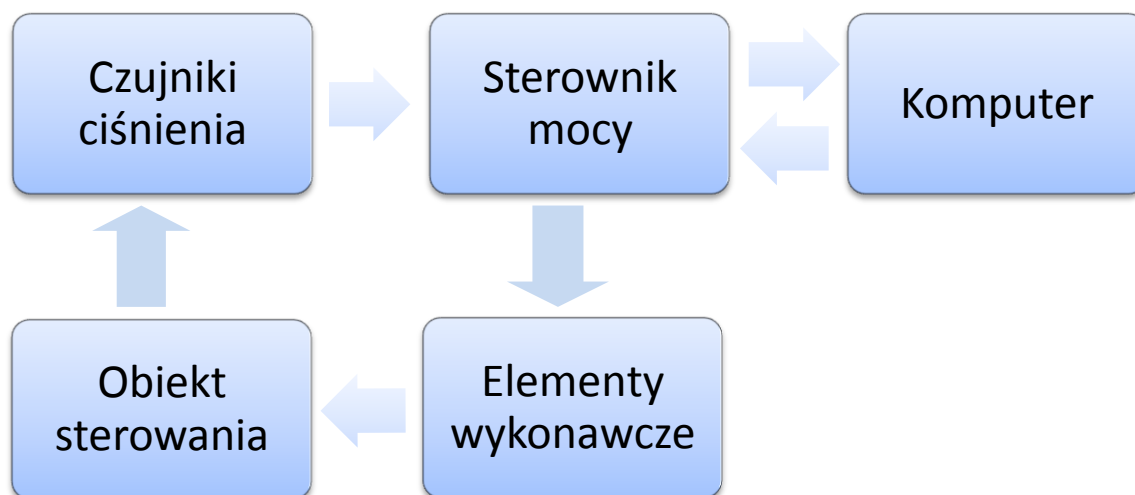
Rys. 1 Obiekt sterowania.

2.4.1 Połączenie obiektu z komputerem.

Na etapie identyfikacji wykorzystano komputer z zainstalowaną kartą wejść/wyjść RT-DAC/PCI oraz środowiskiem MATLAB/Simulink. Karta RT-DAC/PCI stanowi interfejs pomiędzy sterowanym obiektem, a komputerem. Czujniki ciśnienia i elementy wykonawcze (pompa DC i zawory proporcjonalne) podłączono korzystając z wejść/wyjść cyfrowych

i analogowych karty. Dostęp do danych pomiarowych oraz sygnałów sterujących z poziomu programu Simulink uzyskano, wykorzystując dostarczone przez producenta sterowniki programowe I/O. Na podstawie stworzonych w Simulinku diagramów, wygenerowano i skompilowano kod w języku C/C++, za pomocą przyborników RTW i RTWT.

Na rysunku 2 przedstawiono schemat blokowy połączeń pomiędzy obiektem sterowania, sterownikiem mocy oraz komputerem. Blok elementy wykonawcze oznacza pompę prądu stałego oraz zawory proporcjonalne. Sterownik mocy odpowiada za dostarczenie wymaganego prądu do elementów wykonawczych (sterowanie jest typu PWM) oraz stanowi wstępny układ kondycjonowania sygnałów z czujników pomiarowych. Komputer podłączony jest do sterownika mocy przez opisaną wcześniej kartę RT-DAC4/PCI.



Rys. 2 Schemat blokowy połączeń pomiędzy obiektem sterowania, sterownikiem mocy oraz komputerem.

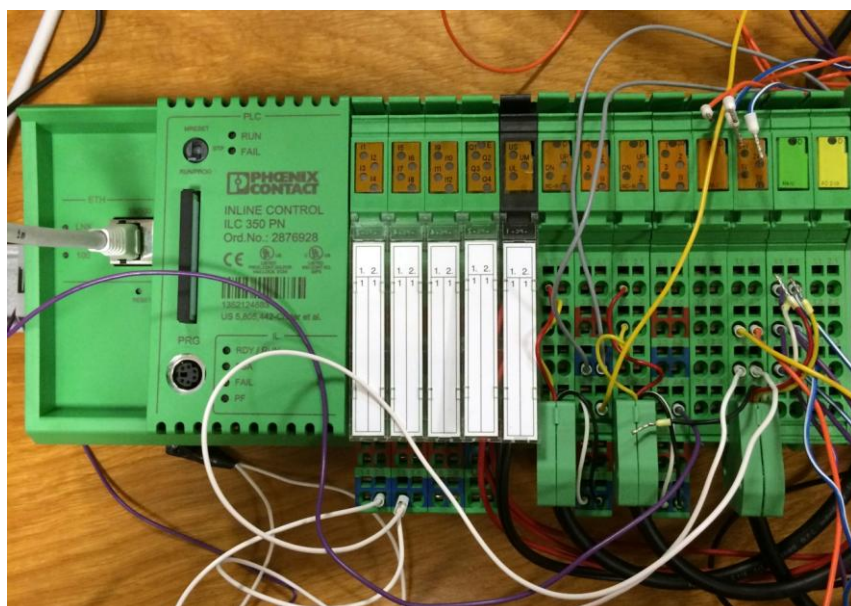
2.4.2 Opis sterownika PLC.

Sterownik PLC podłączono do obiektu na etapie badań eksperymentalnych algorytmów sterowania. Użyty sterownik ILC 350 PN zamontowano na szynie DIN 35 wraz z modułami I/O. Należy on do rodziny Inline, wspierającej standard Profinet. W tabeli 1 zaprezentowano moduły dołączone do sterownika wraz z opisem ich parametrów.

Tab. 1 Wykaz modułów sterownika ILC 350 INLINE.

Nazwa modułu	Typ	Parametry
IB IL INC-IN- PAC	Obsługa enkodera inkrementalnego (nieużywany)	<ul style="list-style-type: none"> dopuszczalna temperatura pracy w zakresie od -25°C do 55°C trzy wejścia dyskretne o nominalnym napięciu 24V DC jedno wyjście dyskretne (tranzystorowe) typu NPN o nominalnym napięciu wyjściowym 0,25 V DC jedno wejście enkoderowe z możliwością wyboru pomiędzy symetrycznym i asymetrycznym typem enkodera(może być zasilane z 5V DC oraz z 24V DC)
IB IL PWM/2- PAC	Wyjścia PWM	<ul style="list-style-type: none"> dopuszczalna temperatura pracy w zakresie od -25°C do 55 dwa wyjścia PWM o napięciu wyjściowym 24 V DC, nominalnym prądzie 500mA i rezystancji wewnętrznej 200MΩ dwa wyjścia PWM o napięciu wyjściowym 5 V DC, nominalnym prądzie 10mA i rezystancji wewnętrznej 50Ω
IB IL AI4/U- PAC	Wejścia analogowe	<ul style="list-style-type: none"> dopuszczalna temperatura pracy w zakresie od -25°C do 55°C cztery wejścia różnicowe (napięciowe) przetwornik A/D o rozdzielczości 12 bitów wejściowy sygnał napięciowy 0V..10V lub -10V...10V
IB IL AO 2/UI- PAC	Wyjścia analogowe (nieużywany)	<ul style="list-style-type: none"> dwa wyjścia analogowe przetwornik D/A o rozdzielczości 12 bitów wyjściowy sygnał: <ul style="list-style-type: none"> a)prądowy(0mA...20mA, 4mA..20mA i -20mA..20mA) b)napięciowy(0V...10V i -10V..10V)

Przy połączeniu pomiędzy zbiornikami, a sterownikiem PLC zastosowano zaprojektowaną przez firmę INTECO płytkę łączeniową. Na rysunku 3 umieszczono zdjęcie sterownika.



Rys. 3 Zdjęcie sterownika ILC 350 PN firmy Phoenix Contact.

3 Identyfikacja.

Pierwszym etapem tworzenia układu sterowania jest identyfikacja parametrów modelu matematycznego kaskadowo połączonych zbiorników oraz wyznaczenie niezbędnych do realizacji modelu charakterystyk. W tym celu przeprowadzono pomiary na stanowisku laboratoryjnym. Proces identyfikacji składał się z następujących kroków:

- Konwersja sygnałów napięciowych z czujników piezorezystancyjnych ciśnienia na poziomy cieczy w poszczególnych zbiornikach wyrażonych w [cm],
- Wyznaczeniu zależności pomiędzy współczynnikiem wypełnienia sygnału PWM pompy, a jej wydajnością wyrażoną w [cm³/s],
- Identyfikacja współczynników wypływu zaworów ręcznych.

W kolejnych podrozdziałach przedstawiono opis modelu matematycznego oraz kolejne ww. kroki procesu identyfikacji.

3.1 Model matematyczny.

Model matematyczny kaskady trzech zbiorników, opisany równaniami różniczkowymi, opracowano na podstawie pozycji [4]. Równanie (1) dotyczy górnego zbiornika, (2) środkowego, a (3) dolnego.

$$\frac{dH_1}{dt} = \frac{1}{S(H_1)} \cdot q - \frac{1}{S(H_2)} \cdot C_1 \sqrt{H_1} \quad (1)$$

$$\frac{dH_2}{dt} = \frac{1}{S(H_2)} \cdot C_1 \sqrt{H_1} - \frac{1}{S(H_2)} \cdot C_2 \sqrt{H_2} \quad (2)$$

$$\frac{dH_3}{dt} = \frac{1}{S(H_3)} \cdot C_2 \sqrt{H_2} - \frac{1}{S(H_3)} \cdot C_3 \sqrt{H_3} \quad (3)$$

gdzie:

H_1, H_2, H_3 - poziomy wody w zbiorniku [m] (górnym , środkowym, dolnym),

q - sterowanie, dopływ wody do górnego zbiornika [m³/s],

C_1, C_2, C_3 - współczynniku wypływu (górnym, środkowym, dolnym).

Wzory na powierzchnie swobodne cieczy w poszczególnych zbiornikach opisano równaniami (4) – (6).

$$S(H_1) = a \cdot w \quad (4)$$

$$S(H_2) = c \cdot w + \frac{H_2}{H_{2max}} b \cdot w \quad (5)$$

$$S(H_3) = w \cdot \sqrt{R^2 - (R - H_3)^2} \quad (6)$$

gdzie:

a - szerokość górnego zbiornika,

w - głębokość wszystkich zbiorników,

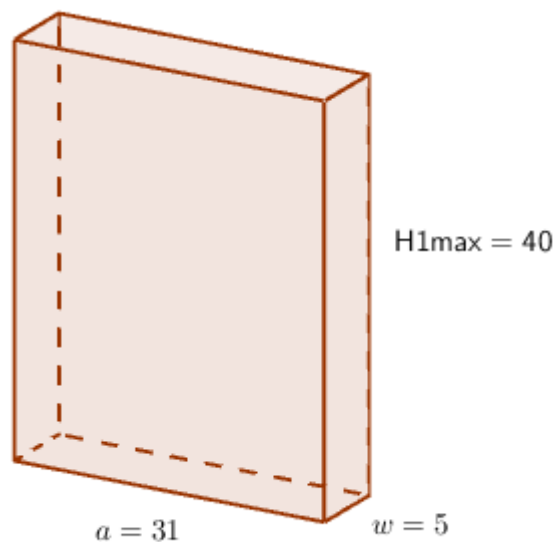
H_{2max} - wysokość górnego zbiornika,

b - szerokość części trójkątnej środkowego zbiornika,

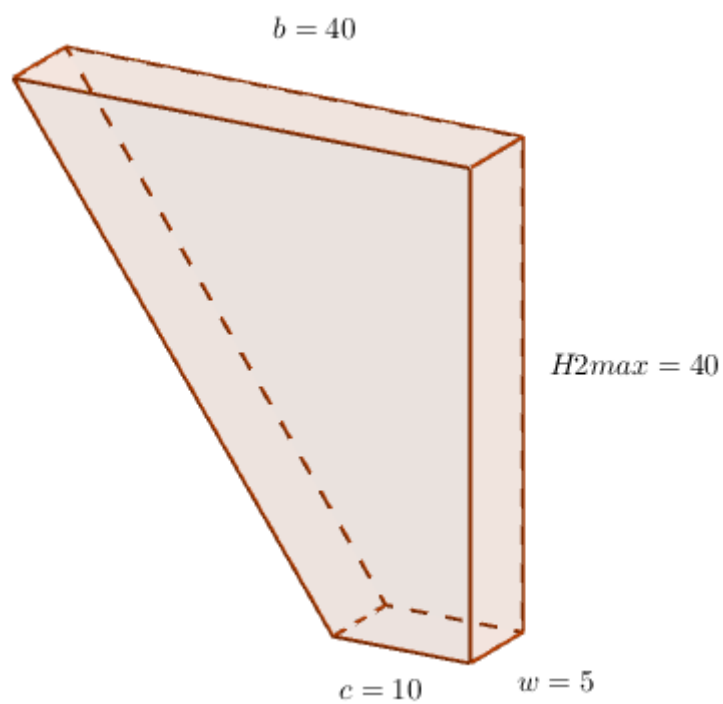
c - szerokość części prostopadłościowej środkowego zbiornika,

R - promień ćwiartki walca w dolnym zbiorniku.

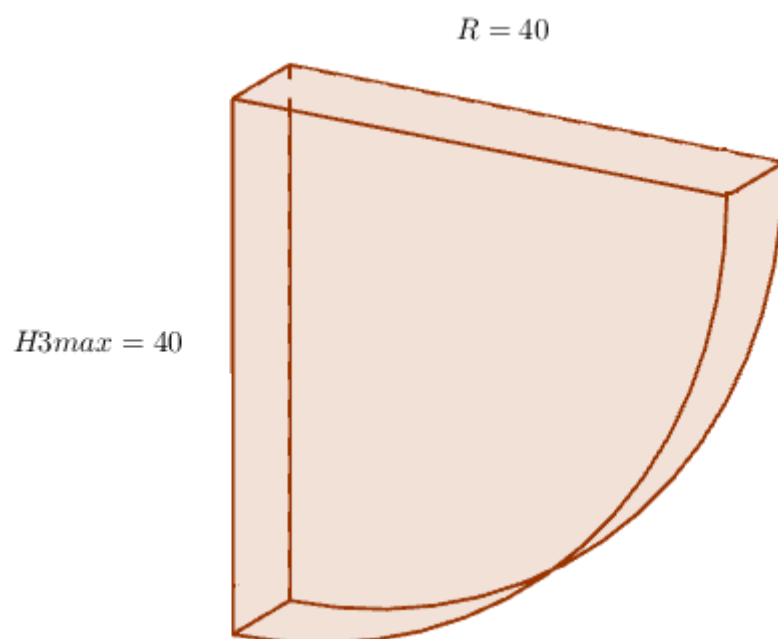
Na rysunkach od 4 do 6 przedstawiono wymiary wszystkich trzech zbiorników w celu zobrazowania rozpatrywanej w pracy kaskady.



Rys. 4 Wymiary geometryczne górnego zbiornika.



Rys. 5 Wymiary geometryczne środkowego zbiornika.



Rys. 6 Wymiary geometryczne dolnego zbiornika.

3.2 Skalowanie sygnałów z czujników ciśnienia.

Pierwszym etapem identyfikacji było odpowiednie przeskalowanie sygnału napięciowego generowanego przez czujnik ciśnienia na poziom cieczy w zbiorniku [5]. W tym celu skorzystano z połączenia opisanego w rozdziale 2.4.1. W programie Simulink wykonano model umożliwiający odczytanie wartości napięć z poszczególnych czujników poziomu. Metoda identyfikacji polegała na ustaleniu zadanego poziomu cieczy w zbiorniku i odczycie wartości napięcia z czujnika, która odpowiadała temu poziomowi. Charakterystykę przetwarzania czujników uzyskano przez aproksymację pięciu punktów pomiarowych wielomianem I stopnia.

W tabeli 2 umieszczono współczynniki wielomianu I stopnia (7) dla każdego z czujników. Aproksymacji dokonano z wykorzystaniem wbudowanej funkcji *polyfit*, dostępnej w MATLABie.

$$H_i = a_i U_i + b_i \quad (7)$$

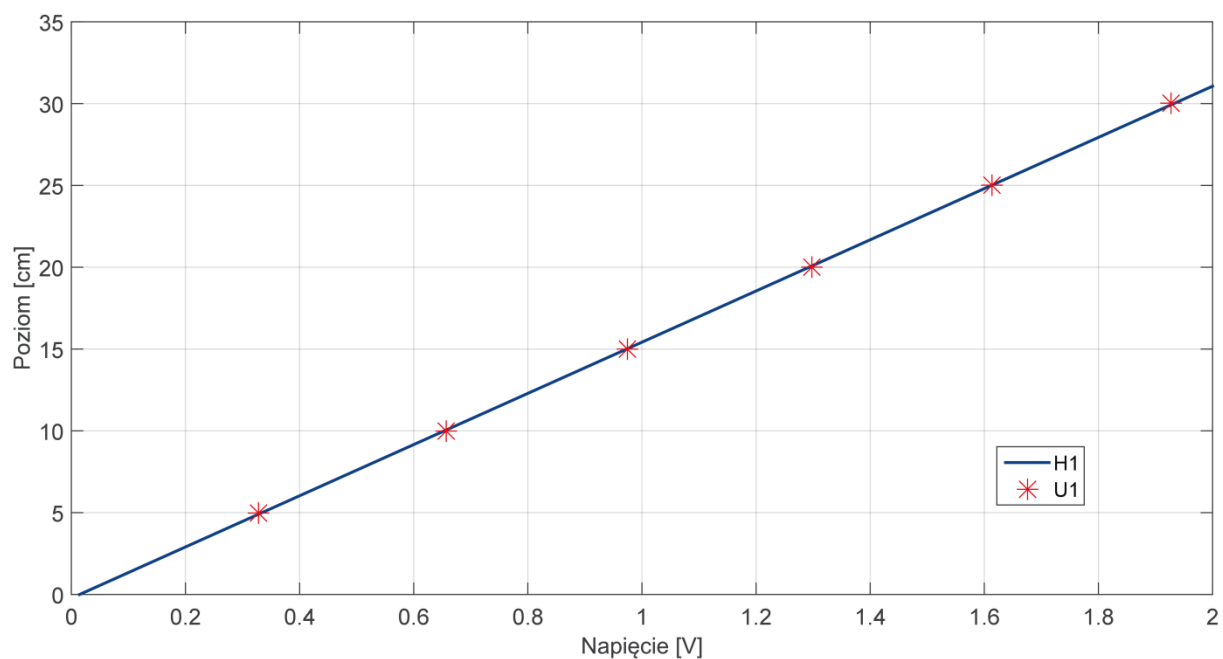
gdzie: i – indeks (numer) kolejnych zbiorników, $i = 1, 2, 3$.

a_i, b_i – parametry wielomianu I stopnia.

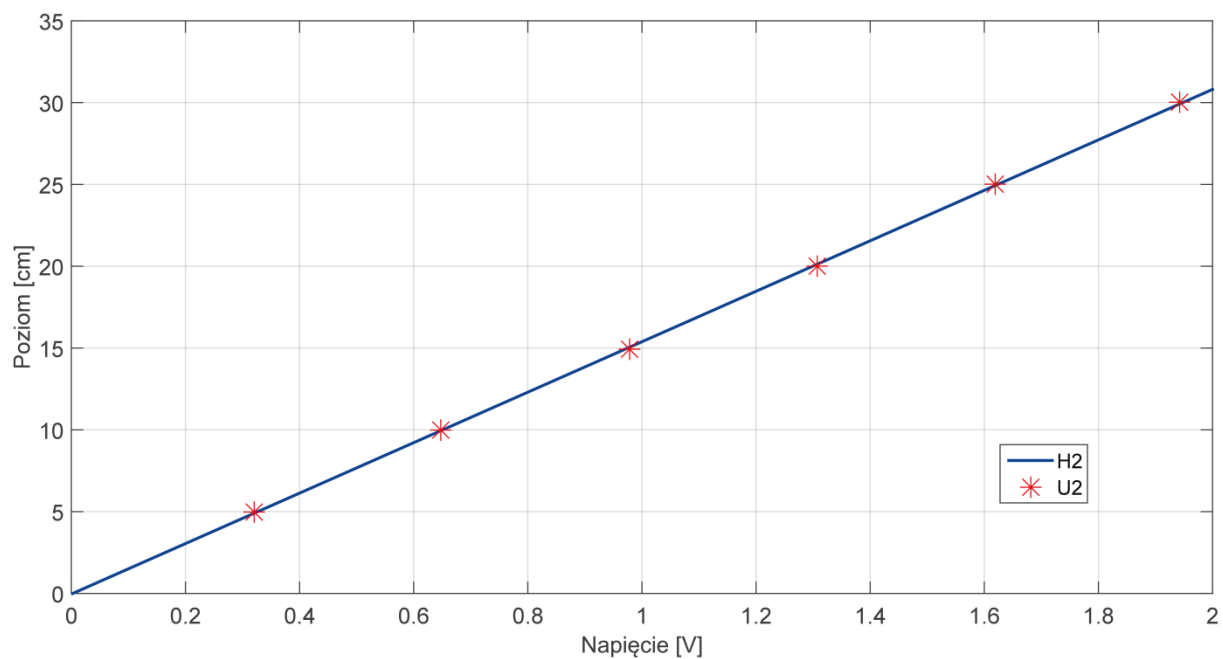
Tab. 2 Współczynniki wielomianu charakterystyk przetwarzania dla poszczególnych czujników.

Czujnik	a_i	b_i
Górny	15,6445	-0,2253
Środkowy	15,4212	-0,0352
Dolny	24,7059	-24,4920

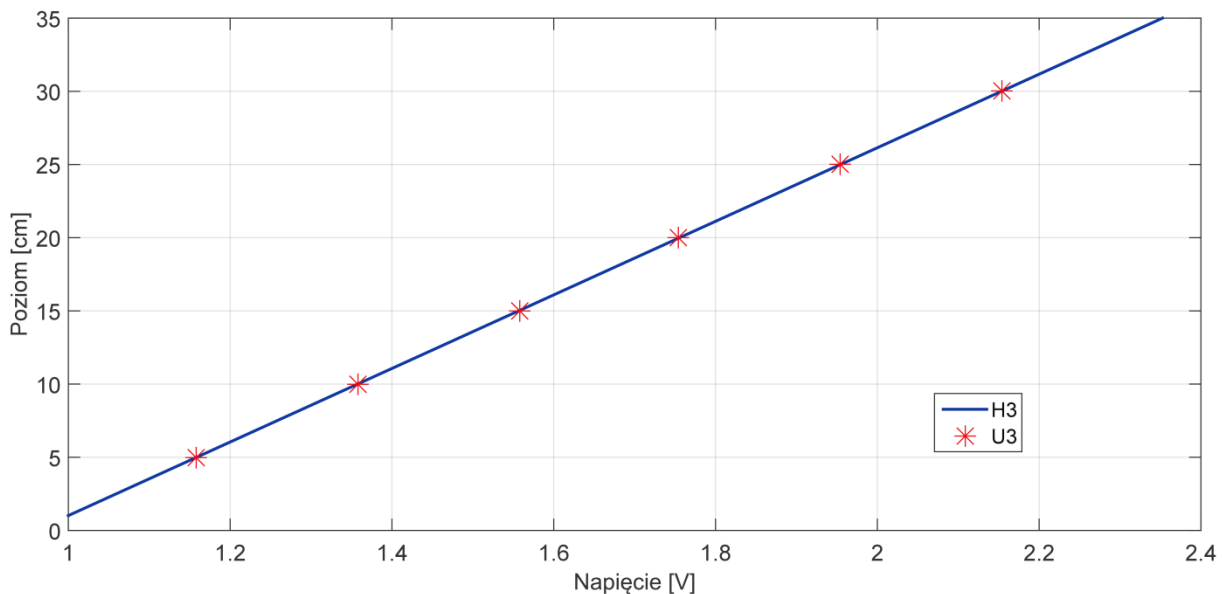
Na rysunkach od 7 do 9 zaprezentowano zależności pomiędzy generowanym przez czujnik napięciem, a wysokością wody, odpowiednio w zbiornikach: górnym, środkowym i dolnym.



Rys. 7 Charakterystyka przetwarzania dla czujnika znajdującego się w górnym zbiorniku.



Rys. 8 Charakterystyka przetwarzania dla czujnika znajdującego się w środkowym zbiorniku.



Rys. 9 Charakterystyka przetwarzania dla czujnika znajdującego się w dolnym zbiorniku.

Wykresy z rysunków od 7 do 9 pokazują, małe różnice pomiędzy uzyskanymi charakterystykami, a danymi pomiarowymi. Zastosowanie wielomianu I stopnia do aproksymacji w zupełności wystarczyło do przetwarzania sygnału z piezorezystancyjnych czujników ciśnienia.

3.3 Charakterystyka przetwarzania pompy.

Pompą sterowano przy użyciu współczynnika wypełnienia sygnału PWM w zakresie od 0 do 1. Wartość ta przekładała się na prędkość obrotową silnika prądu stałego, który napędzał pompę. Im wyższa wartość sterowania tym większy przypływ cieczy do górnego zbiornika. Charakterystyka przetwarzania pompy opisuje zależność pomiędzy wartością współczynnika wypełnienia PWM a wydajnością pompy. Wyznaczenie tej charakterystyki polegało na napełnianiu górnego zbiornika wodą, przy ustalonej wartości współczynnika wypełnienia i zamkniętych zaworach: ręcznym oraz proporcjonalnym. Aby wyznaczyć wydajność pompy odczytano, z charakterystyki napełniania, czas po którym poziom wody w zbiorniku uzyskał wartość $H_I=20$ cm. Przy obliczeniach posłużono się zależnością (8).

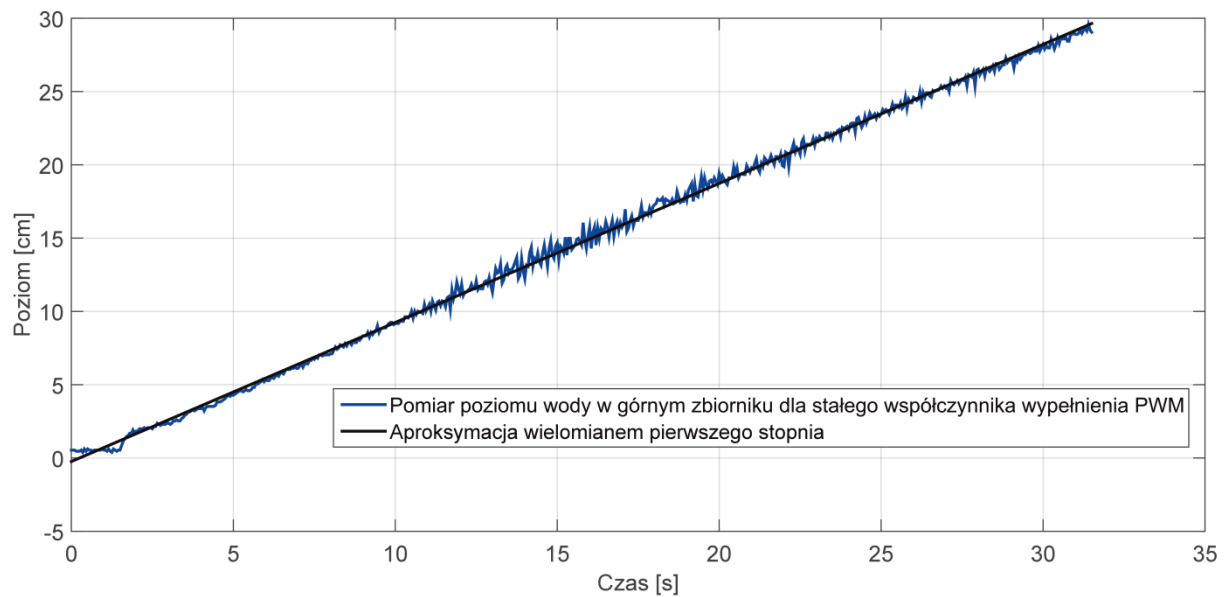
$$q = \frac{20aw}{t_i} \quad (8)$$

gdzie:

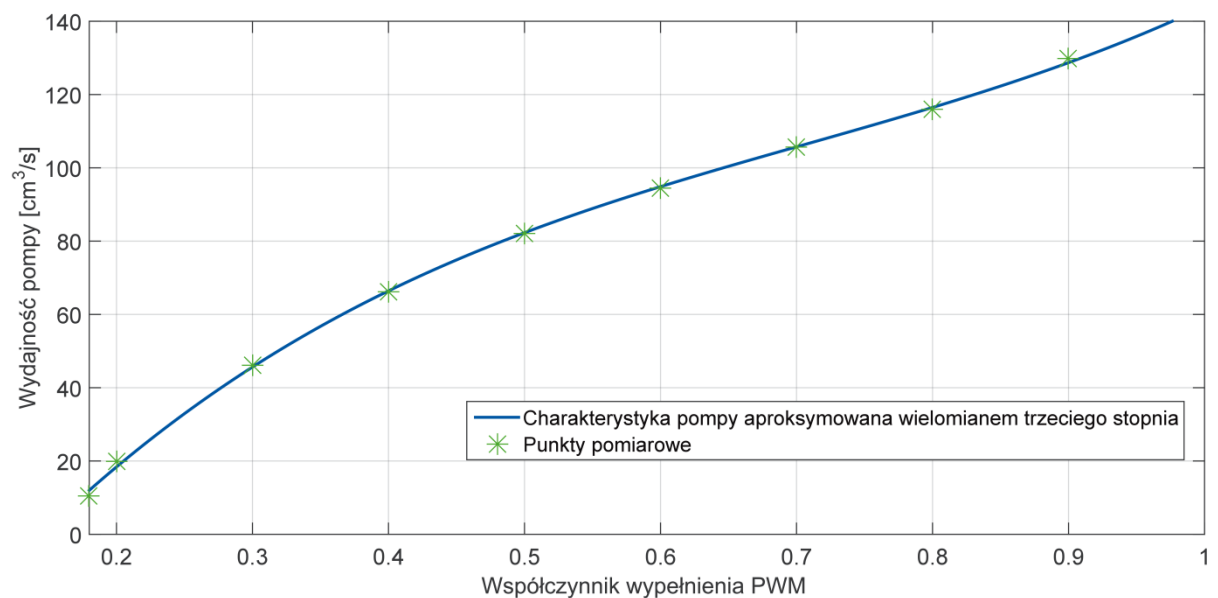
t_i – odczytany czas napełniania zbiornika dla ustalonego współczynnika wypełnienia.

a, w – wymiary geometryczne górnego zbiornika.

Przykładową charakterystykę napełniania, uzyskaną dla współczynnika wypełnienia równego 1 (maksymalna wydajność), pokazano na rysunku 10. Na rysunku 11 zamieszczono charakterystykę przetwarzania pompy.



Rys. 10 Charakterystyka napełniania dla współczynnika wypełnienia równego 1.



Rys. 11 Charakterystyka wydajności pompy

Do wyznaczenia wielomianu najbardziej zgodnego z punktami pomiarowymi, zastosowano wbudowaną w MATLABa funkcję *polyfit*. W trakcie testów na sterowniku okazało się, że identyfikację charakterystyki przetwarzania pompy należało powtórzyć. Powodem była inna częstotliwość sygnału PWM generowanego przez sterownik PLC (10 kHz) w porównaniu z kartą RT-DAC/PCI (5 kHz). W efekcie zarówno zakres działania, jak i wydajność pompy przy danym współczynniku PWM były różne. Na rysunku 12 umieszczono ostateczną charakterystykę przetwarzania pompy wyznaczoną dla częstotliwości 10 kHz. W tabeli 3 przedstawiono współczynniki otrzymanej charakterystyki. W zależności (9) zaprezentowano wzór na wielomian III stopnia.

$$q = aw^3 + bw^2 + cw + d \quad (9)$$

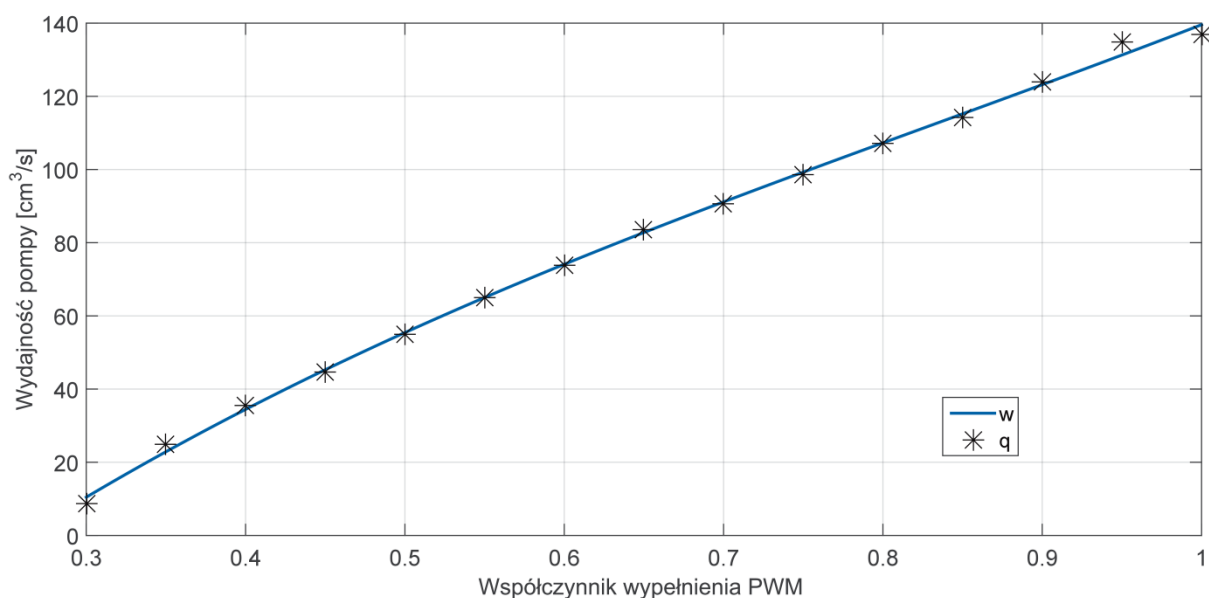
gdzie:

q - wartość współczynnika wypełnienia PWM

w - wydajność pompy [cm^3/s]

Tab. 3 Współczynniki trójmianu dla ostatecznej charakterystyki wydajności pompy.

Współczynnik	A	b	c	D
Wartość	116,4105	-290,4543	400,2483	-86,6295

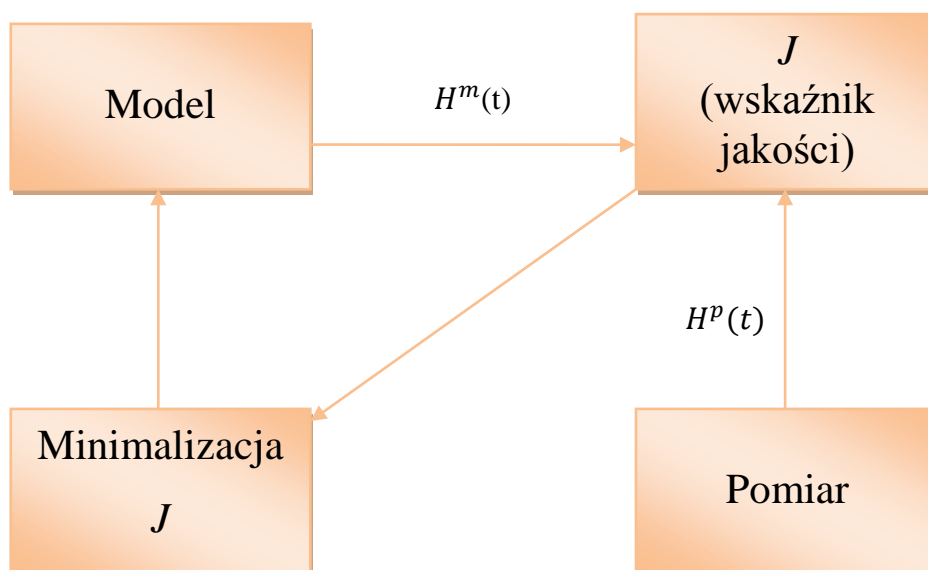


Rys. 12 Ostateczna charakterystyka wydajności pompy.

Porównując rysunki 11 i 12 można zauważyć, że pompa zaczynała swoje działanie przy współczynniku wypełnienia równym 0,3, a nie jak poprzednio przy wartości 0,18. Przybliżenie dla skrajnych wartości generowało błędy, jednak w przedziale pracy pompy w stanie ustalony od 0,5 do 0,8 wartości wypełnienia PWM otrzymano zadowalające wyniki aproksymacji.

3.4 Identyfikacja współczynników wypływu.

Końcowy etap procesu identyfikacji to wyznaczenie parametrów modelu matematycznego. Analiza równań różniczkowych zamieszczonych w podrozdziale 3.1 wskazała na konieczność wyznaczenia współczynników wypływu (C_1 , C_2 , C_3). W tym celu skorzystano z metody strojonego modelu, zgodnie z pozycją [6], polegającej na minimalizacji wskaźnika jakości uwzględniającego różnicę pomiędzy odpowiedziami modelu oraz obiektu rzeczywistego. Schemat działania tej metody przedstawiono na rysunku 13.



Rys. 13 Metoda strojonego modelu.

Przyjęty wskaźnik jakości wyrażono przez sumę z kwadratu różnicy pomiędzy odpowiedziami, w formie przebiegów czasowych poziomów cieczy w poszczególnych zbiornikach, modelu i obiektu (10).

$$J = \sum_{i=1}^N \left(H_j^p(i) - H_j^m(i) \right)^2 \quad (10)$$

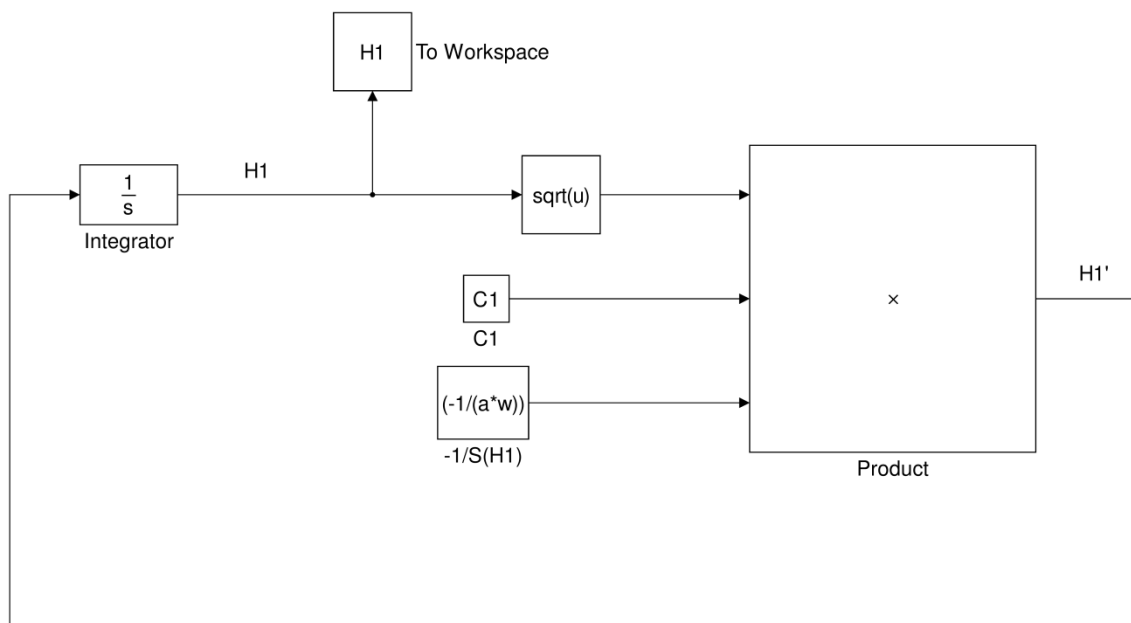
gdzie:

H_j^p - pomiar poziomu cieczy dla poszczególnych zbiorników, $j= 1, 2, 3$.

H_j^m - symulacyjny poziom cieczy dla poszczególnych zbiorników, $j= 1, 2, 3$.

Dobry wskaźnik surowo karze za duże różnice pomiędzy eksperymentalnymi i symulacyjnymi przebiegami czasowymi poziomów wody. Takie podejście pozwala na dokonanie aproksymacji i wyznaczenie przybliżonych wartości poszukiwanych parametrów.

Stworzony w Simulinku model dla górnego zbiornika, wykorzystany w identyfikacji pokazano na rysunku 14. Modele pozostałych zbiorników wyglądają podobnie. Przy realizacji modelu symulacyjnego wykorzystano równania stanu opisane w rozdziale 3.1. W trakcie przeprowadzania identyfikacji parametrów skorzystano z funkcji *fminsearch*. Znajduje się ona w przyborniku *Optimization Toolbox*. Funkcja ta bazuje na bezgradientowych metodach znajdowania minimum funkcji wielu zmiennych. Pozwoliła na optymalne wyznaczenie poszukiwanych współczynników wypływu oraz wartości początkowych poziomów $X0_i$ w poszczególnych zbiornikach. W tabeli 4 przedstawiono zidentyfikowane wartości parametrów C_1 , C_2 i C_3 .

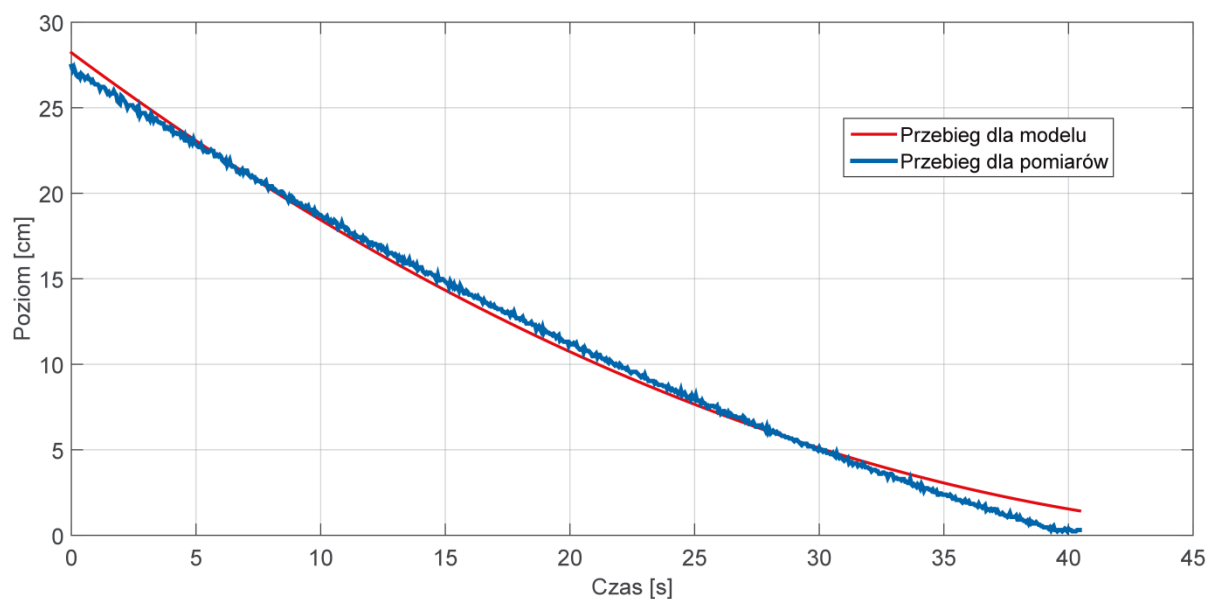


Rys. 14 Model górnego zbiornika wykonany w Simulinku

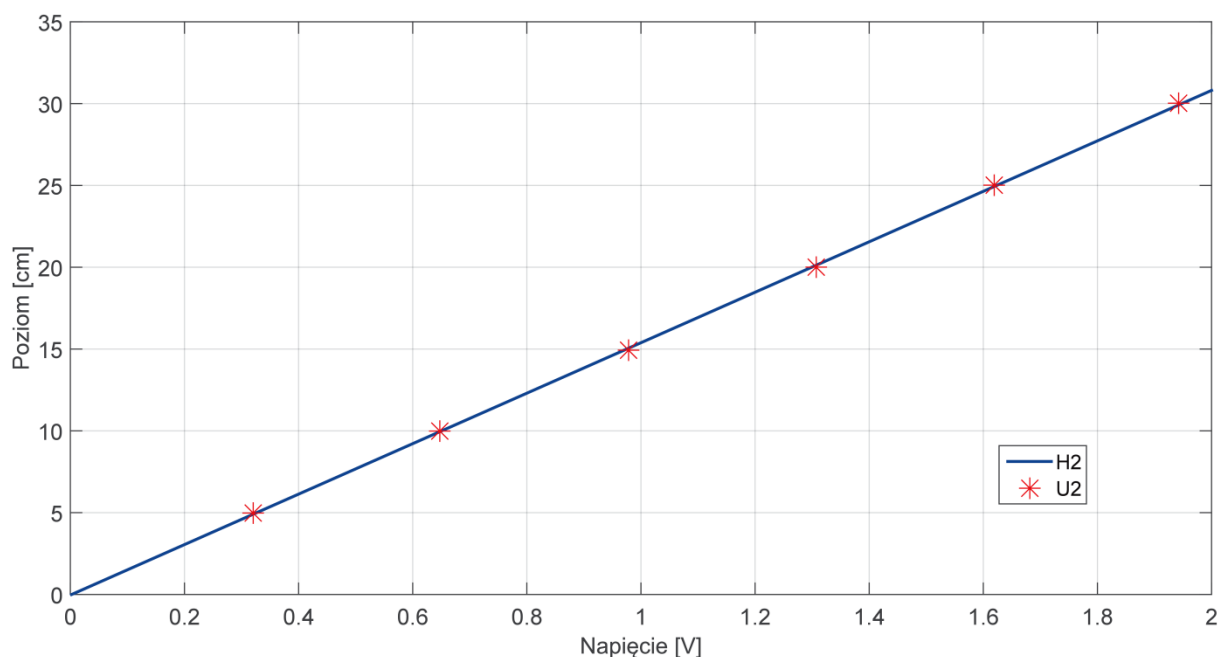
Tab. 4 Zidentyfikowane wartości współczynników wypływu i początkowych poziomów cieczy.

C1	C2	C3
31,5575	21,5016	27,1402
$X0_1$	$X0_2$	$X0_3$
28,2209	30,2045	30,8106

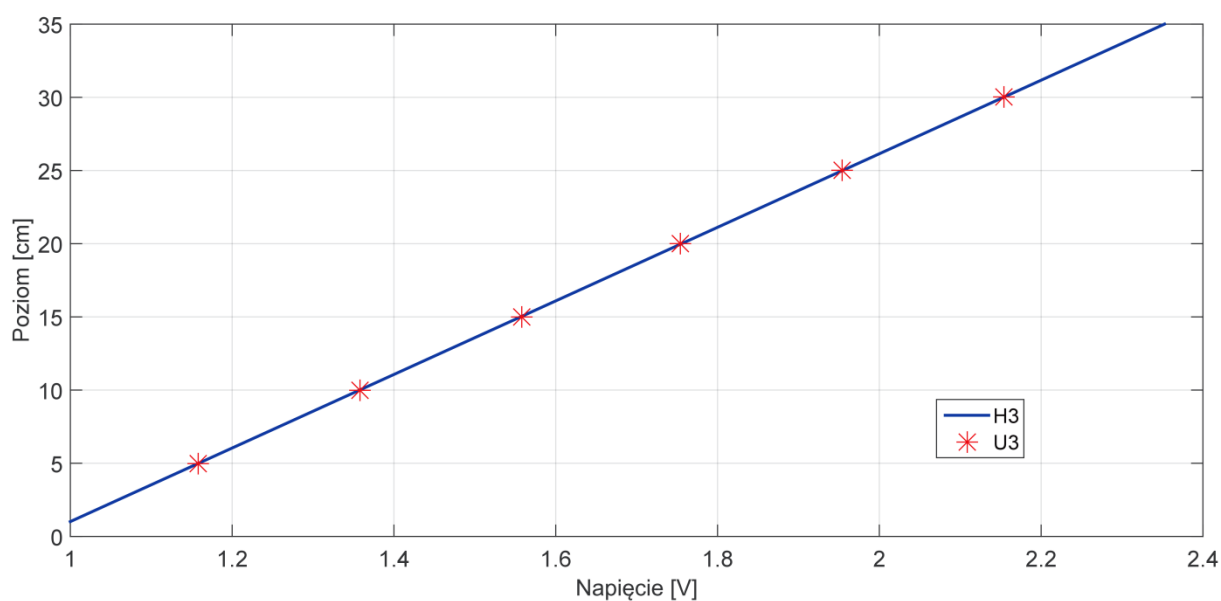
Po wielu testach wyniki zaprezentowane w tabeli 4 wybrano jako ostateczne. Zmiany opcji funkcji *fminsearch*, takich jak: maksymalna ilość iteracji, czy wybór metody poszukiwania minimum, nie powodowały otrzymania innych wartości od tych umieszczonych powyżej. Drugi z wyznaczonych parametrów, czyli początkowy poziom cieczy w zbiorniku, nie wpływa na wartości współczynników wypływu. Na rysunkach od 15 do 17 zaprezentowano charakterystyki porównawcze danych pomiarowych i symulacyjnych dla wszystkich zbiorników



Rys. 15 Porównanie charakterystyk wypływu w górnym zbiorniku.



Rys. 16 Porównanie charakterystyk wypływu w środkowym zbiorniku.



Rys. 17 Porównanie charakterystyk wypływu w dolnym zbiorniku.

Jak można zauważyć na rysunkach od 15 do 17 odpowiedź modelu w zadowalającym stopniu przybliża odpowiedź obiektu rzeczywistego. Największe różnice są widoczne na początku opróżniania zbiorników oraz gdy poziom spada poniżej 5 centymetrów. Otrzymane wyniki, wskazują, że przyjęta procedura optymalizacji sprawdza się, jako narzędzie umożliwiające wyznaczenie parametrów modelu matematycznego. Zdecydowano, że zbyt czasochłonna identyfikacja za pomocą strojonego modelu zostanie zastąpiona linearyzacją w otoczeniu punktów pracy w stanie ustalonym, co opisano w rozdziale 4.

4 Synteza regulatora LQ.

W rozdziale zawarto kolejne kroki, wymagane do syntezy regulatora LQ dla kaskady trzech zbiorników. Pierwsze dwa podrozdziały dotyczą matematycznego opisu procesu linearyzacji oraz regulatora. Następnie zaprezentowano realizację stworzonego systemu sterowania z wykorzystaniem Simulinka oraz przeprowadzone testy symulacyjne.

4.1 Linearyzacja modelu.

Zaprojektowanie regulatora LQ wymaga istnienia modelu liniowego obiektu dynamicznego. W przypadku, gdy dynamika procesu opisana jest modelem nieliniowym, wymagana jest jego linearyzacja w otoczeniu punktów pracy. Takim procesem jest badana w pracy kaskada zbiorników.

Początkowo w metodzie dynamicznej linearyzacji należy wyznaczyć punkty równowagi $P_0(x_{10}, x_{20}, x_{30})$ [7]. Są to takie punkty dla których wartości funkcji f_1, f_2 i f_3 opisanych w równaniu (11) wynoszą 0.

$$\begin{cases} \frac{dx_1}{dt} = f_1(x_1, x_2, x_3) \\ \frac{dx_2}{dt} = f_2(x_1, x_2, x_3) \\ \frac{dx_3}{dt} = f_3(x_1, x_2, x_3) \end{cases} \quad (11)$$

Następnym krokiem jest rozwinięcie funkcji w szereg Taylor'a [8] w wybranym punkcie równowagi, uwzględniając pominięcie pochodnych wyższych rzędów niż pierwszy (12).

$$\begin{aligned} f_i(x_1, x_2, x_3) \approx & f_i(x_{10}, x_{20}, x_{30}) + \frac{\partial f_i}{\partial x_1}(x_{10}, x_{20}, x_{30})(x_1 - x_{10}) + \\ & + \frac{\partial f_i}{\partial x_2}(x_{10}, x_{20}, x_{30})(x_2 - x_{20}) + \frac{\partial f_i}{\partial x_3}(x_{10}, x_{20}, x_{30})(x_3 - x_{30}) \end{aligned} \quad (12)$$

W przypadku, gdy $x_i \approx x_{i0}$, to $f_i(x_{10}, x_{20}, x_{30}) = 0$ dla $i \in \{1, 2, 3\}$.

Ostatnim etapem dynamicznej linearyzacji jest stworzenie Jakobianu otrzymanych funkcji po rozwinięciu w szereg Taylor'a. Dla rozważanego przypadku ogólnego (11) Jakobian opisano macierzą (13).

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_{10}, x_{20}, x_{30}) & \frac{\partial f_1}{\partial x_2}(x_{10}, x_{20}, x_{30}) & \frac{\partial f_1}{\partial x_3}(x_{10}, x_{20}, x_{30}) \\ \frac{\partial f_2}{\partial x_1}(x_{10}, x_{20}, x_{30}) & \frac{\partial f_2}{\partial x_2}(x_{10}, x_{20}, x_{30}) & \frac{\partial f_2}{\partial x_3}(x_{10}, x_{20}, x_{30}) \\ \frac{\partial f_3}{\partial x_1}(x_{10}, x_{20}, x_{30}) & \frac{\partial f_3}{\partial x_2}(x_{10}, x_{20}, x_{30}) & \frac{\partial f_3}{\partial x_3}(x_{10}, x_{20}, x_{30}) \end{bmatrix} \quad (13)$$

Wyznaczenie punktów równowagi wykorzystano do identyfikacji współczynników wpływu. Linearyzacja posłużyła do wyznaczenia macierzy stanu oraz macierzy sterowania, używanych przy implementacji regulatora LQ. Metoda stanu ustalonego okazała się skuteczna i bardziej dokładna od metody stojonego modelu w przypadku stabilizacji poziomów cieczy na kilku wybranych poziomach.

Punkty równowagi odpowiadające stanowi ustalonemu dla rozważanej w pracy kaskady zbiorników wyznaczano na podstawie zależności (14).

$$\begin{cases} f_1(H_{10}, H_{20}, H_{30}, q_0) = \frac{dH_1}{dt} = 0 \\ f_2(H_{10}, H_{20}, H_{30}, q_0) = \frac{dH_2}{dt} = 0 \\ f_3(H_{10}, H_{20}, H_{30}, q_0) = \frac{dH_3}{dt} = 0 \end{cases} \quad (14)$$

gdzie:

$H_{10}, H_{20}, H_{30}, q_0$ - punkty równowagi, w których zlinearyzowano system.

Korzystając z równań (4)-(6) oraz zależności (14) wyznaczono matematyczne wzory współczynników wpływu C_1, C_2, C_3 , które przedstawiono za pomocą układu równań (15).

$$\begin{cases} C_1 = \frac{q_0}{\sqrt{H_{10}},} \\ C_2 = \frac{q_0}{\sqrt{H_{20}},} \\ C_3 = \frac{q_0}{\sqrt{H_{30}},} \end{cases} \quad (15)$$

Wzorując się na opisie linearyzacji nieliniowych układów dynamicznych zamieszczonym na stronie 25, wyznaczono Jakobiany funkcji dla kaskady trzech zbiorników.

Ze względu na duże skomplikowanie wzorów na pochodne cząstkowe w zależności od poziomów cieczy, przedstawiono je za pomocą układu równań (16). Pochodne cząstkowe w zależności od sterowania zaprezentowano w równaniu (17).

$$\left\{ \begin{array}{l} \frac{\partial f_1(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_1} = \frac{-C_1}{2\sqrt{H_{10}}aw} \\ \frac{\partial f_1(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_2} = 0 \\ \frac{\partial f_1(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_3} = 0 \\ \frac{\partial f_2(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_1} = \frac{C_1}{2\sqrt{H_{10}}X} \\ \frac{\partial f_2(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_2} = \frac{-C_1\sqrt{H_{10}}bw}{H_{2max}X^2} + \frac{C_2\sqrt{H_{20}}bw}{H_{2max}X^2} - \frac{C_2}{2\sqrt{H_{20}}X} \\ \frac{\partial f_2(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_3} = 0 \\ \frac{\partial f_3(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_1} = 0 \\ \frac{\partial f_3(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_2} = \frac{C_2}{2\sqrt{H_{20}}w\sqrt{R^2 - (H_{30} - R)^2}} \\ \frac{\partial f_3(H_{10}, H_{20}, H_{30}, q_0)}{\partial H_3} = \frac{(2H_{30} - 2R)(C_2\sqrt{H_{20}} - C_3\sqrt{H_{30}})}{Y} - \frac{C_3}{2\sqrt{H_{30}}w\sqrt{R^2 - (H_{30} - R)^2}} \end{array} \right. \quad (16)$$

gdzie:

$$X = cw + \frac{H_{20}bw}{H_{2max}}$$

$$Y = 2w(R^2 - (H_{30} - R)^2)^{1.5}$$

$$B = \left[\begin{array}{l} \frac{\partial f_1(H_{10}, H_{20}, H_{30}, q_0)}{\partial q} = \frac{1}{aw} \\ \frac{\partial f_2(H_{10}, H_{20}, H_{30}, q_0)}{\partial q} = 0 \\ \frac{\partial f_3(H_{10}, H_{20}, H_{30}, q_0)}{\partial q} = 0 \end{array} \right] \quad (17)$$

gdzie:

B - macierz sterowania systemu

Wszystkie użyte w równaniach (16) i (17) symbole opisano w podrozdziale 3.1.

4.2 Regulator LQ.

Parametry regulatora LQ wyznacza się na podstawie minimalizacji wskaźnika jakości [9], opisanego funkcjonalem kwadratowym (18).

$$J(u) = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (18)$$

Wartość sterowania w przypadku regulatora LQ wyliczana jest z zależności (19).

$$u = -Kx \quad (19)$$

gdzie:

x - stan, u - sterowanie, K - macierz wzmocnień,

Q - macierz kosztów od stanu, R - macierz kosztów od sterowania.

Macierze Q i R należy tak dobrać, aby wskaźnik J osiągał jak najniższą wartość. Duże wartości dla wagi Q w porównaniu z wagą R powodują nakładanie większych kar na zmienne stanu. Postępując odwrotnie istotniejszy jest płynny sygnał sterujący bez przeregulowań. Dopasowanie macierzy kosztów jest najtrudniejszym etapem przy projektowaniu regulatora LQ. Czasami potrzeba wielu testów do otrzymania zadowalających wyników. Problemem może być również sposób wyselekcjonowania początkowych wartości wag i zachowania odpowiednich proporcji pomiędzy nimi. W efekcie nie dla każdego systemu ten typ regulatora jest odpowiedni. W celu otrzymania macierzy wzmocnień, należy rozwiązać algebraiczne równanie Riccatiego [10], a następnie wyznaczyć macierz K ze wzoru (20) po wcześniejszym doborze macierzy Q i R .

$$K = R^{-1} B^T S \quad (20)$$

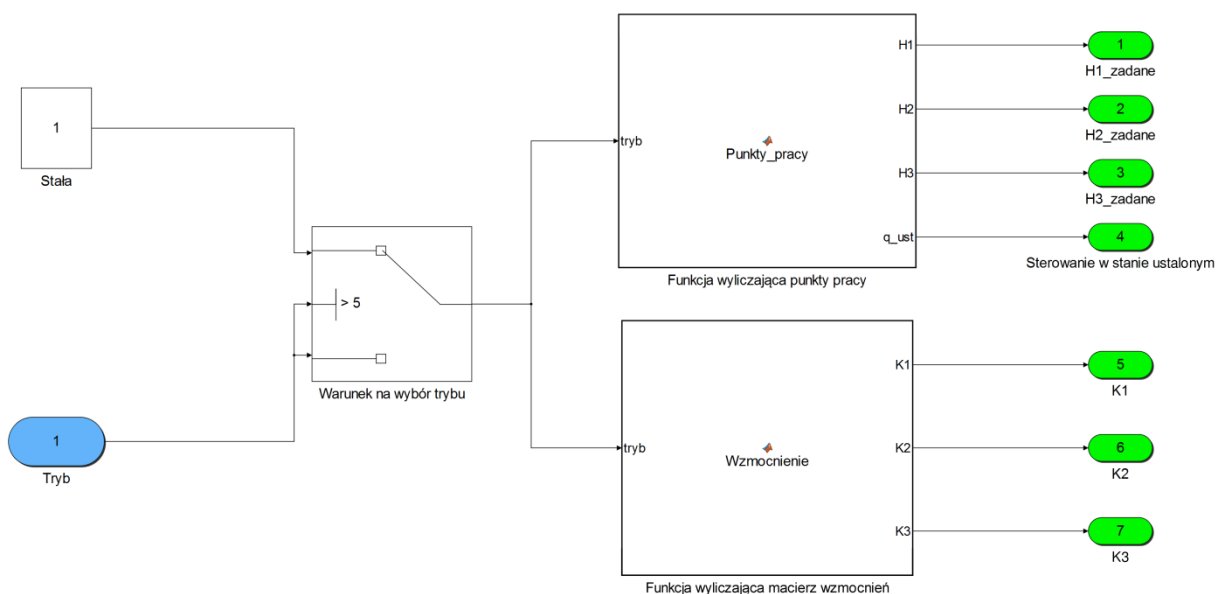
gdzie:

R - macierz wag, B - macierz wejść w równaniu stanu opisującym dynamikę systemu

S - rozwiązanie równania Riccatiego.

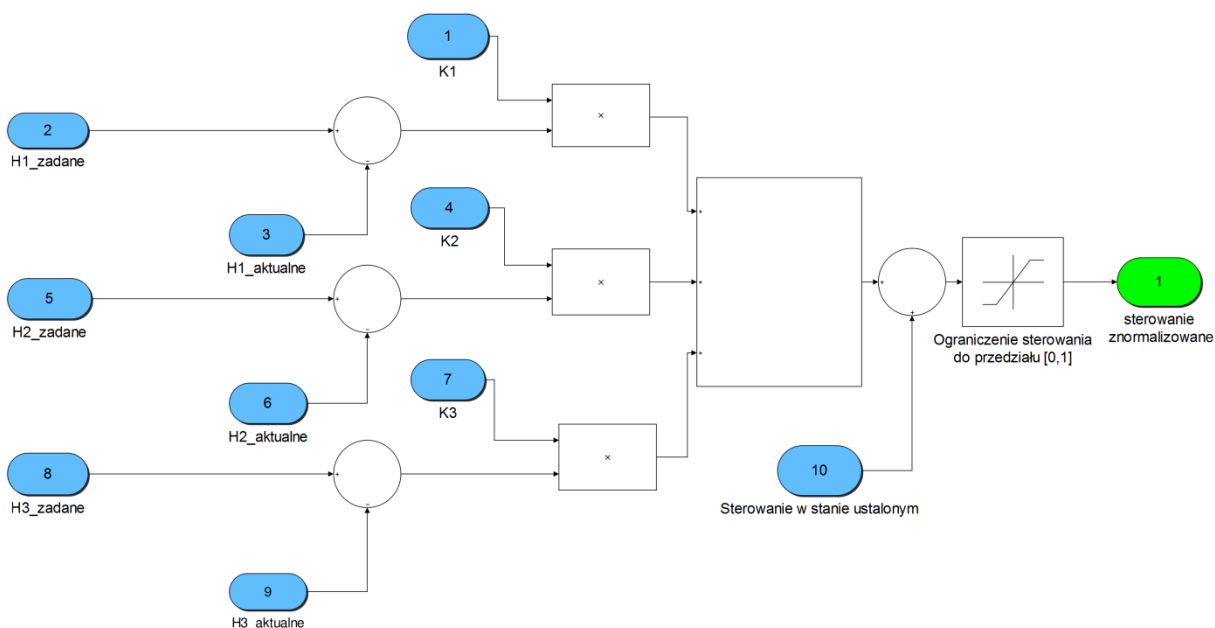
4.3 Realizacja regulatora LQ w środowisku MATLAB Simulink.

Przeprowadzenie identyfikacji oraz wiedza teoretyczna na temat regulatora LQ pozwoliły na zaprojektowanie systemu sterowania dla kaskady zbiorników. Zadanie wykonano, korzystając z oprogramowania Simulink. Wybrano 5 zestawów poziomów wody w poszczególnych zbiornikach, dla których osiągnięto stan ustalony w trakcie procesu identyfikacji. Dla wybranych wartości zadanych wyznaczono parametry regulatora LQ. Ustalono, że dla prawidłowej pracy pompy, wartości współczynnika wypełnienia sygnału PWM zawierają się w przedziale od 0,6 do 0,8. Poziomy dobrano tak, aby w stanie ustalonym sterowanie znajdowało się w powyższych zakresach. Na rysunkach od 18 do 20 pokazano kolejne podsystemy stworzonego modelu regulatora.



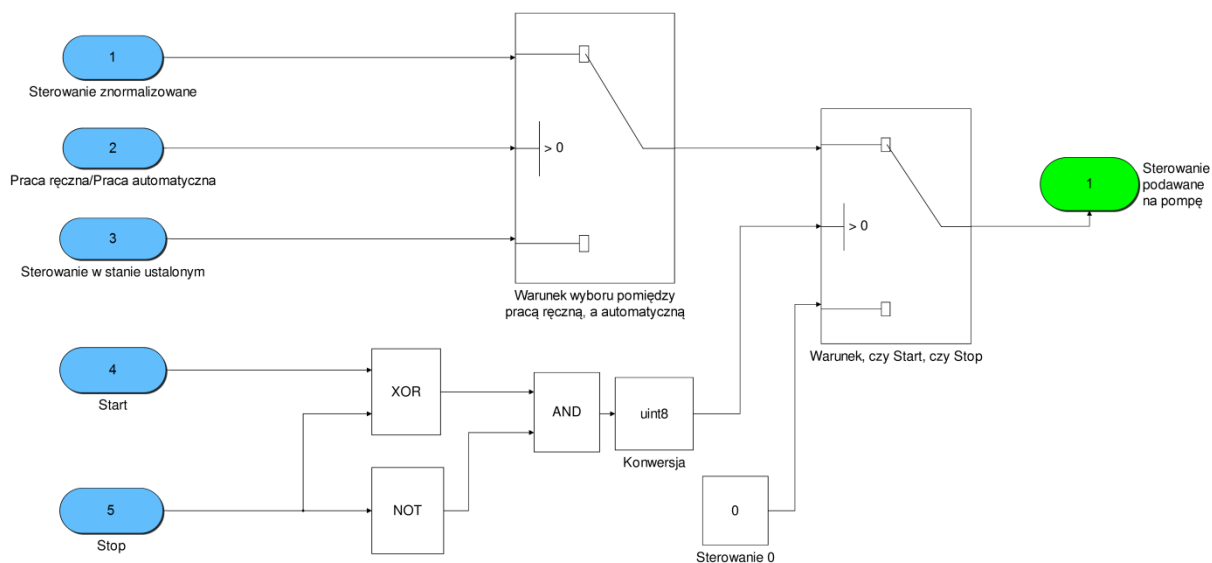
Rys. 18 Podsystem odpowiedzialny za wybór odpowiednich wartości zadanych oraz macierzy wzmocnień w zależności od wybranego trybu.

Na rysunku 18 przedstawiono podsystem, umożliwiający wybór wartości zadanych dla każdego ze zbiorników oraz odpowiadających im macierzy wzmocnień regulatora LQ. W przypadku nieprawidłowej wartości zmiennej wejściowej (zaznaczonej kolorem niebieskim na rysunku 18) tryb ustawiano domyślnie na pierwszy. Wyjścia z podsystemu zaznaczono kolorem zielonym.



Rys. 19 Podsystem realizujący regulator LQ.

Za pomocą podsystemu widocznego na rysunku 19 zrealizowano wyliczanie sygnału sterującego podawanego na pompę. Na początku obliczano dla każdego ze zbiorników różnicę pomiędzy zadanymi poziomami cieczy, a ich aktualnymi wartościami (błąd regulacji). Następnie mnożono każdy z uchybów z odpowiadającą wartością wzmocnienia, wyznaczoną za pomocą wbudowanej w MATLABie funkcji *dlqr*. W modelu regulatora uwzględniono wartość sterowania w stanie ustalonym.



Rys. 20 Podsystem dokonujący wybór pomiędzy pracą ręczną i automatyczną.

Ostatni podsystem regulatora zawiera dwie funkcjonalności. Pierwsza z nich pozwala na wybór pomiędzy pracą automatyczną (wartość sterowania wyznaczana na podstawie stworzonego regulatora LQ), a pracą ręczną (stała wartość sterowania w stanie ustalonym). Druga umożliwia rozpoczęcie/zakończenie działania stworzonego systemu.

4.4 Testy symulacyjne.

Na podstawie modelu regulatora opisanego w podrozdziale 4.3 przeprowadzono testy symulacyjne. Na rysunkach od 21 do 25 zaprezentowano przebiegi poziomów wody w zbiornikach dla wszystkich trybów rozważanych w pracy. W tabeli 5 przedstawiono wyznaczone dla każdego z trybów: macierze wzmocnień, wartości poziomów cieczy oraz sterowanie w stanie ustalonym.

Tab. 5 Poziomy cieczy, sterowanie w stanie ustalonym oraz macierz K dla każdego trybu.

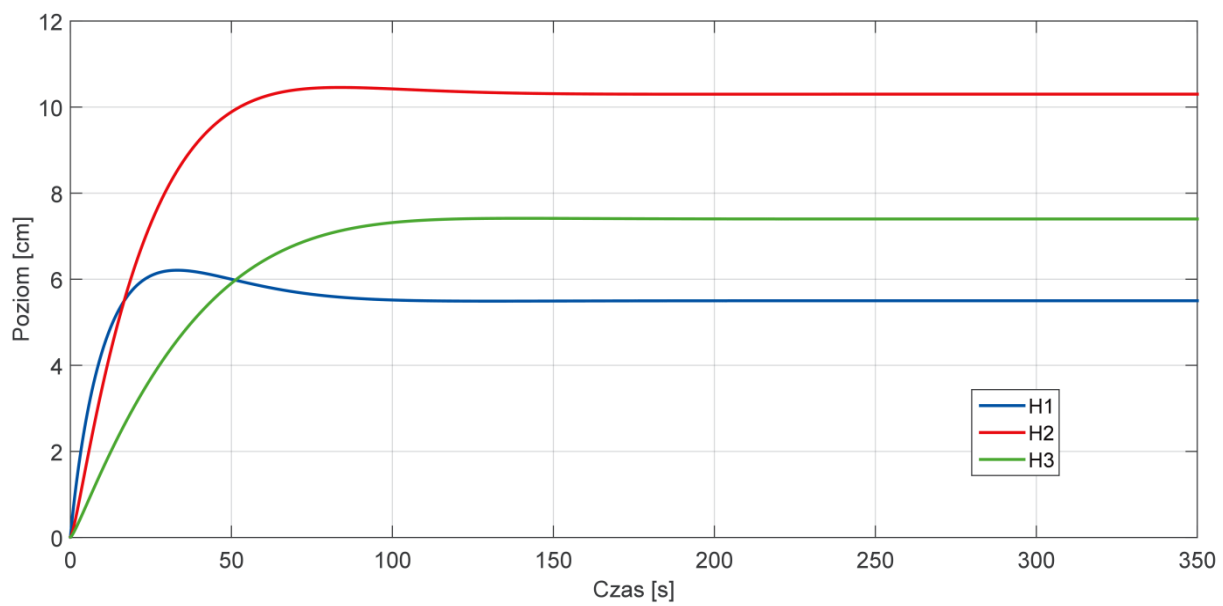
Tryb	$H1$	$H2$	$H3$	q_{ust}	K
1	5,5	10,3	7,4	0,6	[0,0296 0,0133 0,0084]
2	7,4	14,2	9,6	0,65	[0,0319 0,0162 0,0095]
3	9,6	18	11,6	0,7	[0,0333 0,0183 0,0098]
4	11,8	22,5	14,5	0,75	[0,0358 0,0221 0,0113]
5	14	25,5	16,5	0,8	[0,0369 0,0238 0,0117]

gdzie:

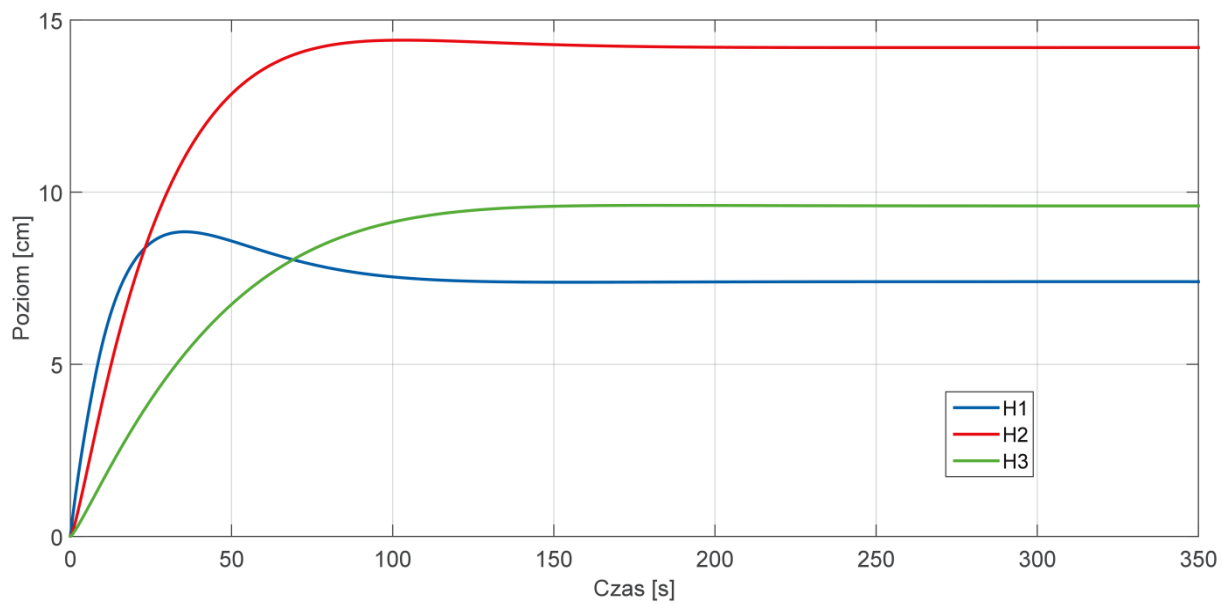
$H1, H2, H3$ - poziomy w zbiornikach w kolejności: górny, środkowy, dolny.

q_{ust} - sterowanie w stanie ustalonym.

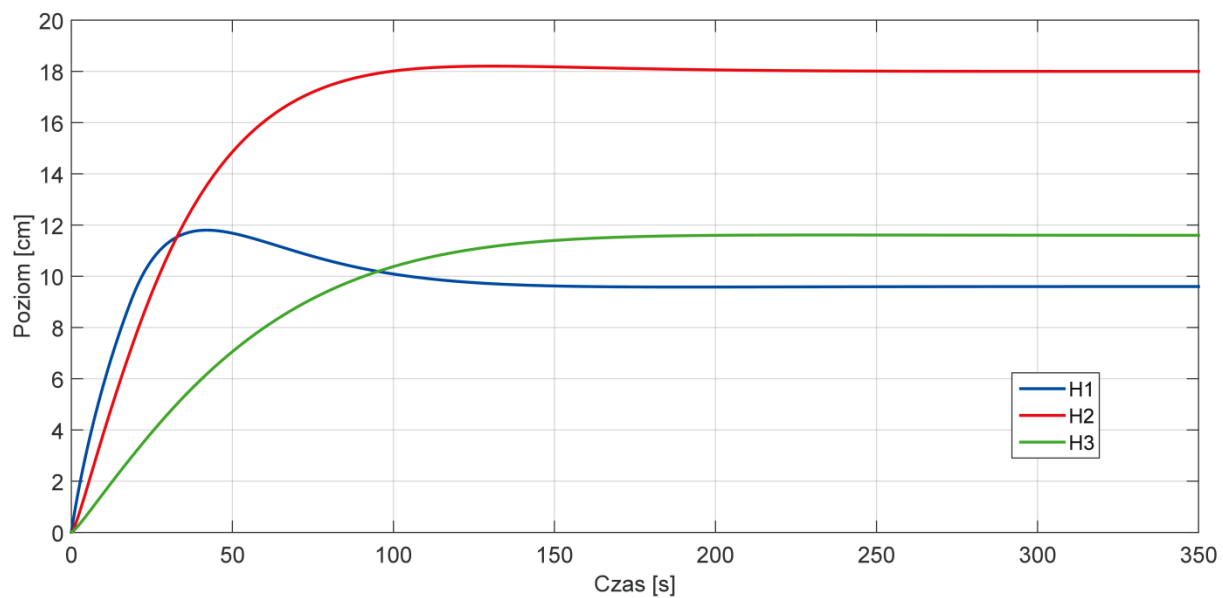
K - macierz wzmocnień dla regulatora LQ



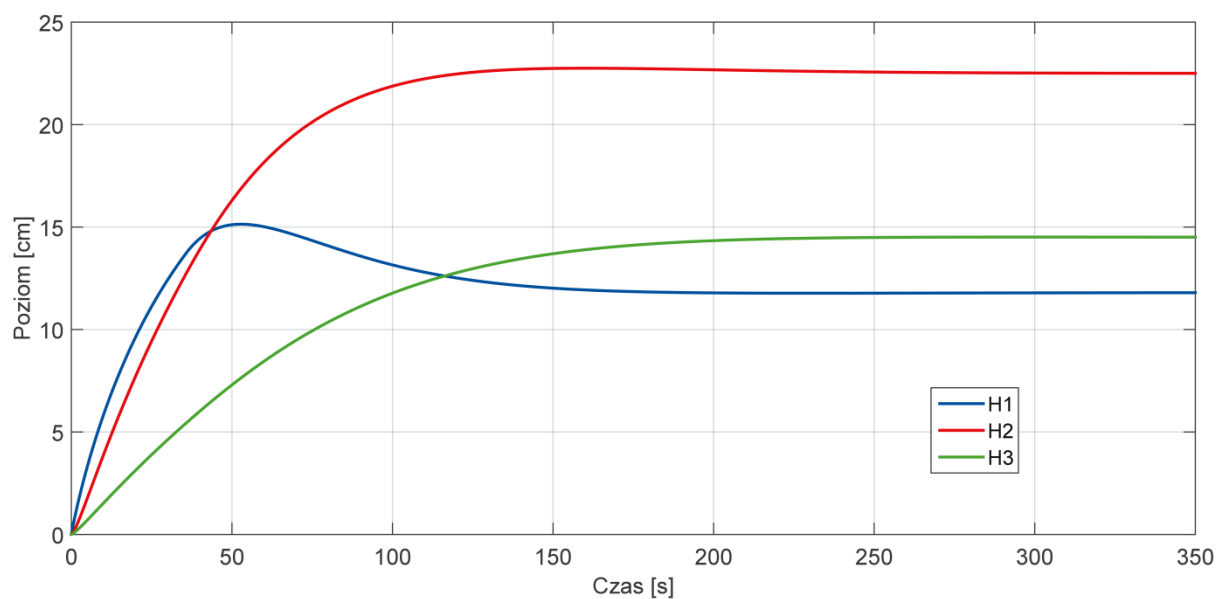
Rys. 21 Przebiegi symulacyjne poziomów wody w zbiornikach dla trybu 1.



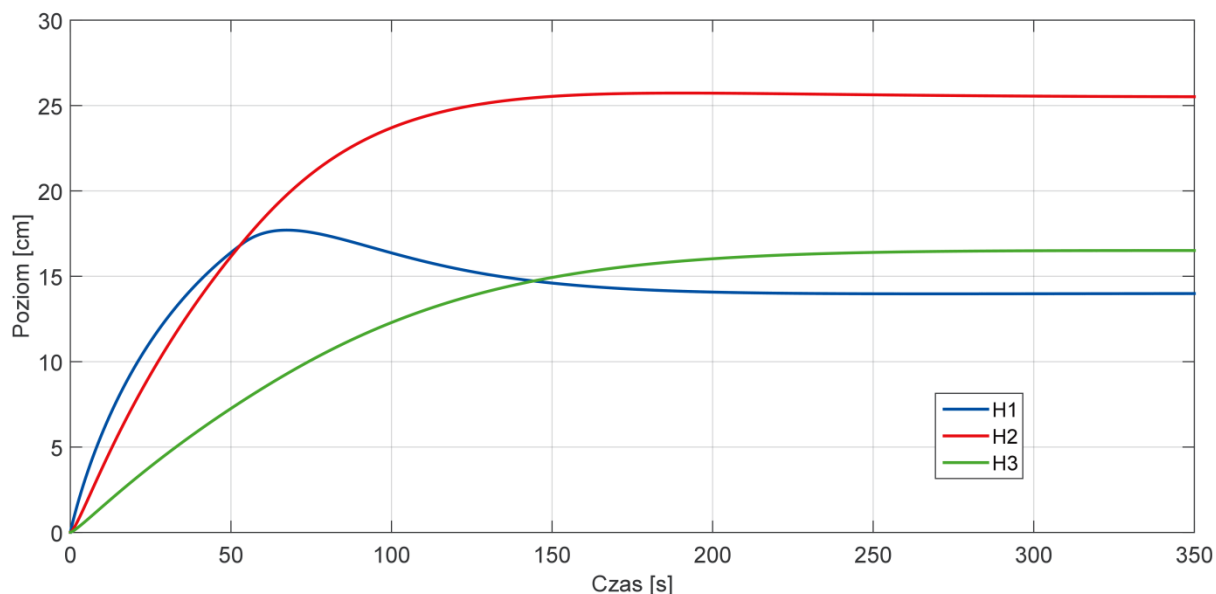
Rys. 22 Przebiegi symulacyjne poziomów wody w zbiornikach dla trybu 2.



Rys. 23 Przebiegi symulacyjne poziomów wody w zbiornikach dla trybu 3.

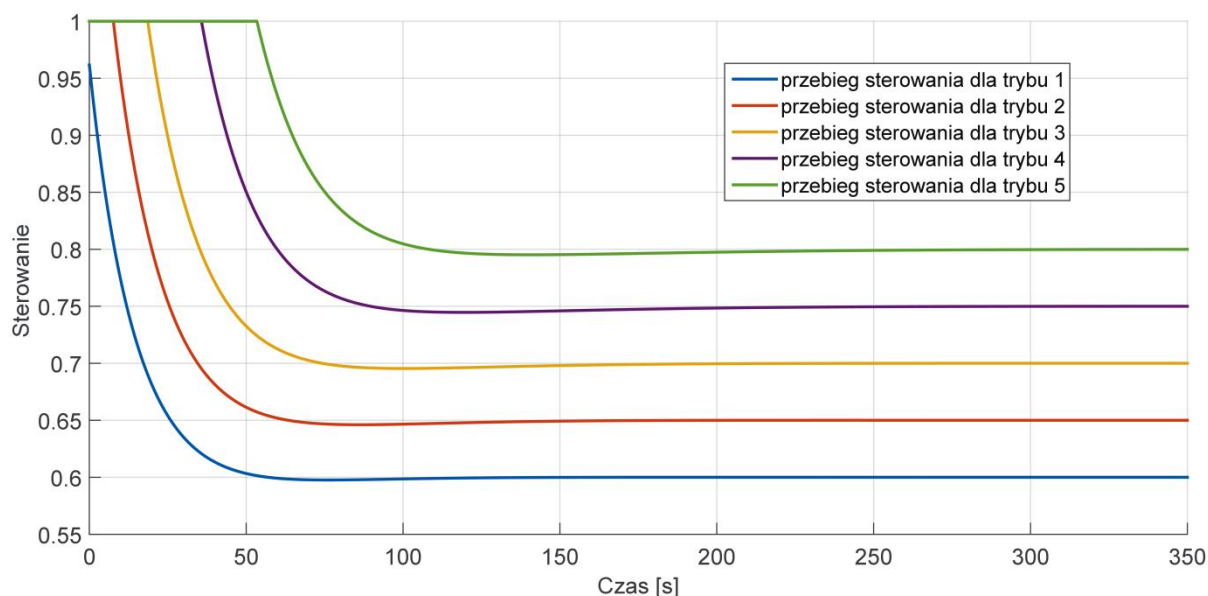


Rys. 24 Przebiegi symulacyjne poziomów wody w zbiornikach dla trybu 4.



Rys. 25 Przebiegi symulacyjne poziomów wody w zbiornikach dla trybu 5.

Analiza rysunków 21-25 wskazują, że im wyższe wartości zadane poziomów cieczy tym dłuższy czas regulacji. Czas ten, dla każdego z prezentowanych przypadków, mieścił się w przedziale od 100 do 150 sekund. Zamieszczone parametry w tabeli 5 wyznaczono dla identycznych macierzy Q i R . Większe kary nakładano na sterowanie w celu uniknięcia przeregulowań sygnału sterującego. Przeregulowania wpływałyby negatywnie na prawidłową pracę pompy. Na rysunku 26 pokazano przebiegi sterowania dla każdego z trybów.



Rys. 26 Przebiegi sygnału sterującego dla każdego z trybów rozpatrywanych w pracy.

Wykres z rysunku 26 wskazuje, że wyższe wartości sterowania (poziomów cieczy) w stanie ustalonym powodowały nasycenie sterowania przez początkowy okres działania regulatora. Analiza przebiegów pozwala postawić tezę, że górna granica wartości sterowania w stanie ustalonym, dla wybranych macierzy Q i R regulatora LQ, wyniosła 0,8.

5 Generacja kodu ST oraz testy na sterowniku

Zgodnie z tematem pracy większość podprogramów aplikacji uruchamianej na sterowniku, zrealizowano, korzystając z oprogramowania Simulink. Z tego powodu oprócz generacji kodu regulatora, stworzono odpowiednie podsystemy umożliwiające obsługę wykorzystywanych modułów I/O sterownika PLC. Opis obsługi modułów oraz generacji kodu ST zamieszczono w podrozdziale 5.1. Następnie dokonano zwięzłego opisu programu uruchamianego na sterowniku PLC. Na końcu rozdziału opisano przeprowadzone testy na obiekcie rzeczywistym.

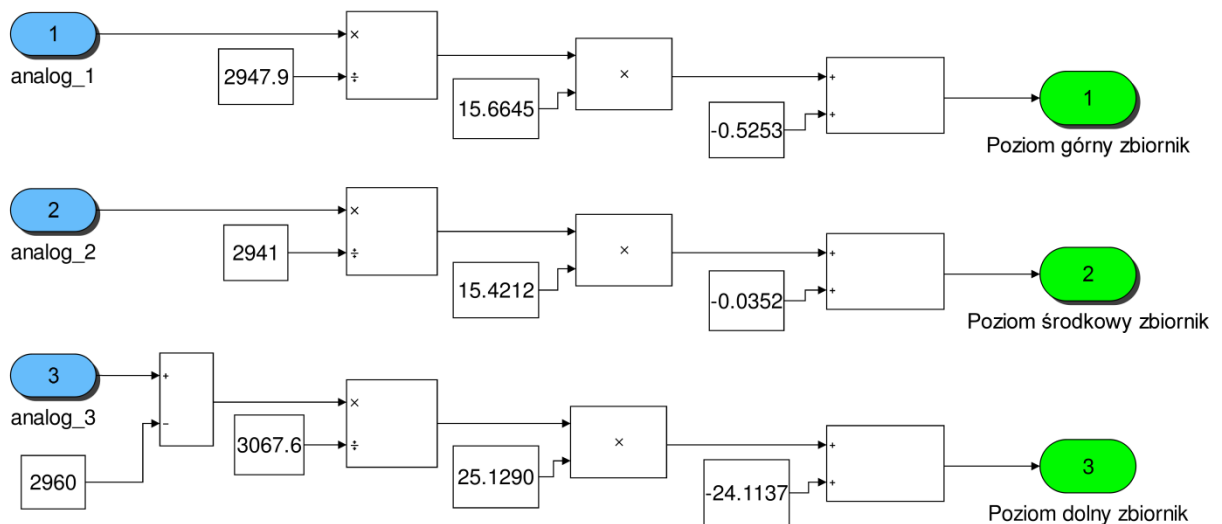
5.1 Obsługa modułów sterownika.

Łącząc ze sobą podsystemy z rysunków od 18 do 20 stworzono kod ST regulatora LQ. Przed generacją kodu ustalono następujące opcje PLC Codera:

- minimalizację pętli algebraicznych,
- tworzenie raportu wygenerowanego kodu,
- długość nazw sygnałów nieprzekraczającą 30 liter,
- docelowe środowisko generowanego kodu, jako PC Worx,
- usuwanie konwersji danych zmiennoprzecinkowych na całkowitoliczbowe.

Po zatwierdzenie powyższych ustawień, wygenerowano kod regulatora LQ. Raport wygenerowanego kodu nie wykazał błędów oraz ostrzeżeń. Formatem stworzonego kodu był XML.

Zaprojektowano trzy podsystemy pozwalające na obsługę modułów sterownika firmy Phoenix Contact. Pierwszy z nich dotyczył wejść analogowych. Za pomocą tego modułu odczytywano sygnały z czujników ciśnienia, dostarczających do sterownika informację w postaci napięcia. Z tego powodu zgodnie z informacją zawartą w podrozdziale 3.2 konieczne było odpowiednie przeskalowanie odczytywanych sygnałów zgodnie z tabelą 2. Przy pomiarze napięcia wykorzystano metodę dwuprzewodową ze względu na niedużą odległość pomiędzy obiektem sterowania, a modułem sterownika. Wadą takiego rodzaju pomiaru jest słabe tłumienie zakłóceń. Zgodnie z [11], wejścia analogowe posiadały słowa skojarzone z modułem , pozwalające na: odczytanie sygnału oraz ustawienie zakresu napięć na 0-10V(używany w pracy) lub $\pm 10V$. Na rysunku 27 pokazano realizację podsystemu obsługi wejść analogowych.



Rys. 27 Podsystem realizujący obsługę wejść analogowych sterownika.

Na podstawie rysunku 27 wygenerowano kod, którego fragment zamieszczono w tabeli 6.

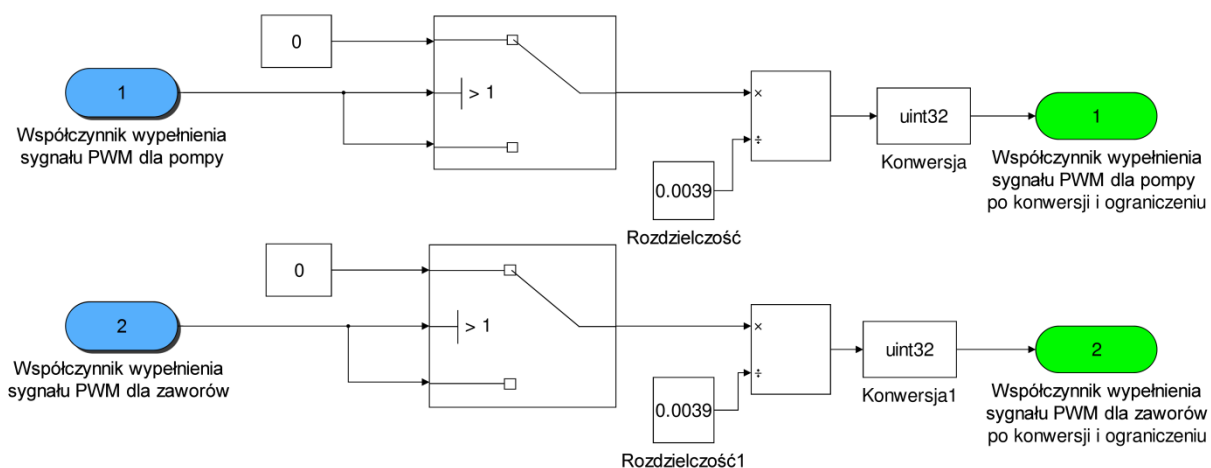
Tab. 6 Fragment wygenerowanego kodu realizujący obsługę wejść analogowych.

```
<inputVars>
<variable name="analog_3">
  <type>
    <REAL/>
  </type>
</variable>
<variable name="analog_2">
  <type>
    <REAL/>
  </type>
</variable>
<variable name="analog_1">
  <type>
    <REAL/>
  </type>
</variable>
</inputVars>
</interface>
<body>
  <ST>
    <html xmlns="http://www.w3.org/1999/xhtml">
      <div id="MWTDESCRIPTION" wsName="obsługa_analogow0" xml:space="preserve"><br/>
      Poziom_gorny_zbiornik := ((analog_1 / 2947.9) * 15.6645) + -0.5253;<br/>
      Poziom_srodkowy_zbiornik := ((analog_2 / 2941.0) * 15.4212) + -0.0352;<br/>
      Poziom_dolny_zbiornik := (((analog_3 - 2960.0) / 3067.6) * 25.129) + -24.1137;<br/></div></html>
```

Na początku wygenerowanego kodu występuję deklaracja zmiennych. W zamieszczonym fragmencie w tabeli 6 widoczna jest deklaracja zmiennych wejściowych.

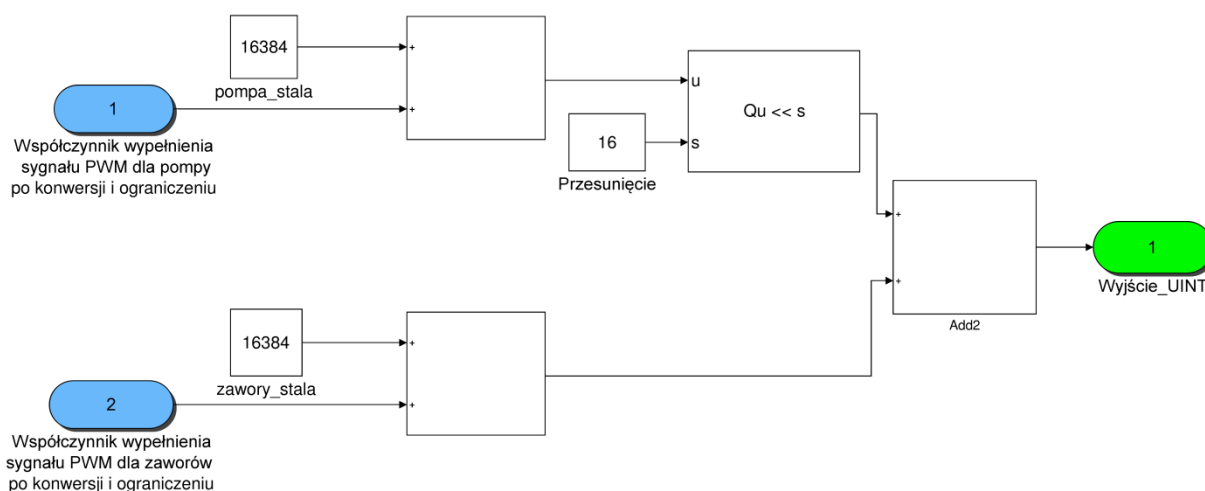
Nazwy zmiennych są zgodne z nazewnictwem zastosowanym w podsystemie zaprezentowanym na rysunku 27. Ostatnie trzy linijki umieszczonego w tabeli 6 kodu realizują przeliczanie sygnału z czujników ciśnienia na poziomy wody dla każdego ze zbiorników.

Obsługa modułu wyjść PWM sterownika była kolejnym podsystemem stworzonym w oprogramowaniu Simulink na podstawie którego przeprowadzono generację kodu. Ze względu na fakt posiadania przez moduł tylko dwóch wyjść, zdecydowano, że jedno z nich zostało podłączone do pompy, a drugie wspólnie do wszystkich trzech zaworów elektromagnetycznych. Przy takim sposobie podłączenie zawory elektromagnetyczne posłużyły, jako zadajniki zakłóceń. Na rysunku 28 umieszczono podsystem umożliwiający ograniczenie, a następnie konwersję współczynnika wypełnienia sygnału PWM podawanego na pompę i zawory proporcjonalne. Na rysunku 29 przedstawiono schemat pozwalający na umieszczenie wypełnienia PWM w skojarzonym z modułem PWM słowie wyjściowym.



Rys. 28 Podsystem realizujący ograniczenie oraz konwersję współczynnika wypełnienia sygnału PWM.

Za pomocą schematu z rysunku 28 wygenerowano kod pozwalający na podanie współczynnika wypełnienia, jako liczby zmiennoprzecinkowej. Przy użyciu przełączników sprawdzano, czy dane wejściowe (kolor niebieski) zawierają się w przedziale od 0 do 1. Liczba 0,0039 widoczna na schemacie to rozdzielczość sygnału PWM. Na końcu podsystemu realizowano konwersję współczynników wypełnienia ze zmiennych typu *REAL* na typ całkowitoliczbowy.



Rys. 29 Podsystem realizujący umieszczenie w słowie wyjściowym modułu współczynników wypełnienia sygnałów PWM dla pompy i zaworów.

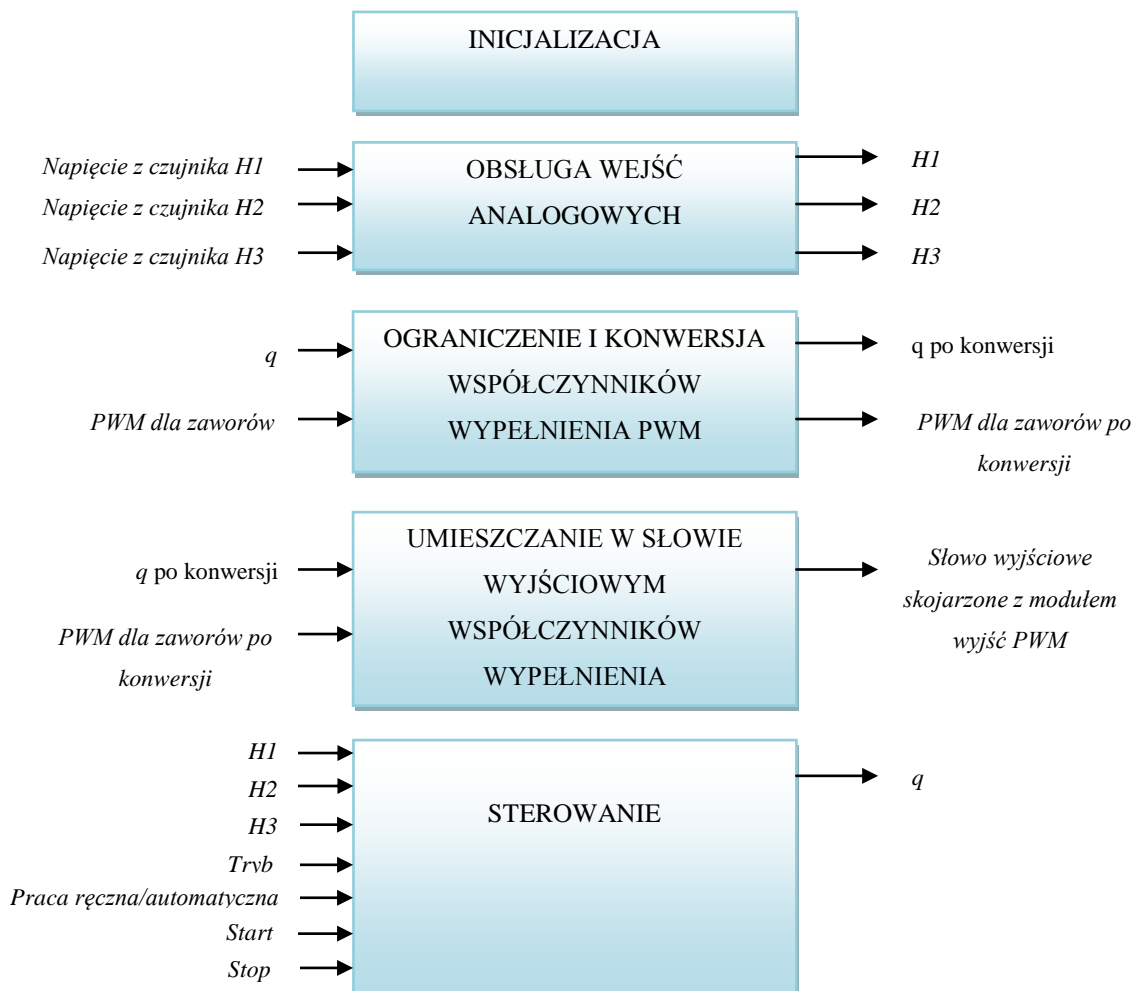
Na podstawie podsystemu z rysunku 29 umożliwiono wpisywanie współczynnika wypełnienia elementów wykonawczych do słowa wyjściowego skojarzonego z modułem PWM. Liczba 16384 zaprezentowana na schemacie wynikała z ustawienia częstotliwości sygnału PWM na 10 kHz, zgodnie z notą aplikacyjną producenta [12]. Za pomocą przesunięcia bitowego zapewniono odpowiednie umieszczenie wartości współczynników wypełnienia w wyjściowym słowie skojarzonym z modułem.

5.2 Opis programu uruchamianego na sterowniku.

Realizacja podsystemów opisanych w podrozdziale 5.1 umożliwiła stworzenie aplikacji z użyciem oprogramowania PC Worx. Odpowiednia konfiguracja modułów sterownika oraz ustawień sieciowych pozwoliły na komunikację pomiędzy sterownikiem PLC, a komputerem klasy PC. Wygenerowane kody ST realizujące poprawne działanie regulatora oraz wykorzystanych modułów sterownika ograniczyły do minimum tworzenie podprogramów z wykorzystaniem środowiska PC Worx. Import pliku XML generowanego przez przybornik PLC Coder do środowiska PC Worx zrealizowano poprzez wykorzystanie biblioteki umożliwiającej parsowanie plików XML oraz narzędzia do importu plików o nazwie *"Import PLCopen xml file"* (dostępnego w oprogramowaniu PC Worx).

W jedynym podprogramie stworzonym bez użycia Simulinka, zawarto inicjalizację współczynników wypełnienia PWM podawanych na elementy wykonawcze oraz konfigurację zakresu wejść analogowych. Nazwy podprogramów dobrano zgodnie z podsystemami

opisanymi w podrozdziale 5.1. Jedynym wyjątkiem jest podprogramu "sterowanie" odpowiedzialny za realizację regulatora LQ. Na rysunku 30 schemat blokowy zależności pomiędzy podprogramami oraz zmiennymi globalnymi w głównym programie.



Rys. 30 Schemat blokowy głównego program uruchamianego na sterownik PLC.

5.3 Testy na obiekcie rzeczywistym.

Ostatnim etapem pracy była weryfikacja stworzonego regulatora na obiekcie rzeczywistym. W celu lepszego zobrazowania otrzymanych wyników postanowiono użyć oprogramowania umożliwiającego wykonanie graficznego interfejsu użytkownika (GUI). Skorzystano z program Visu+, który współpracuje ze sterownikami firmy Phoenix Contact. Serwer OPC zrealizowano, wykorzystując wbudowane narzędzia środowiska PC Worx takie jak: *OPC Test Client* oraz *OPC Configurator*. Po zapoznaniu się z dokumentacją oprogramowania Visu +[13], stworzono panel operatorski składający się z dwóch ekranów. Na pierwszym ekranie (przedstawiony na rysunku 31) zawarto ustawienia konfiguracyjne dla

regulatora . Na drugim (przedstawiony na rysunku 32) umieszczono przebiegi czasowe sterowania oraz poziomów cieczy w poszczególnych zbiornikach.

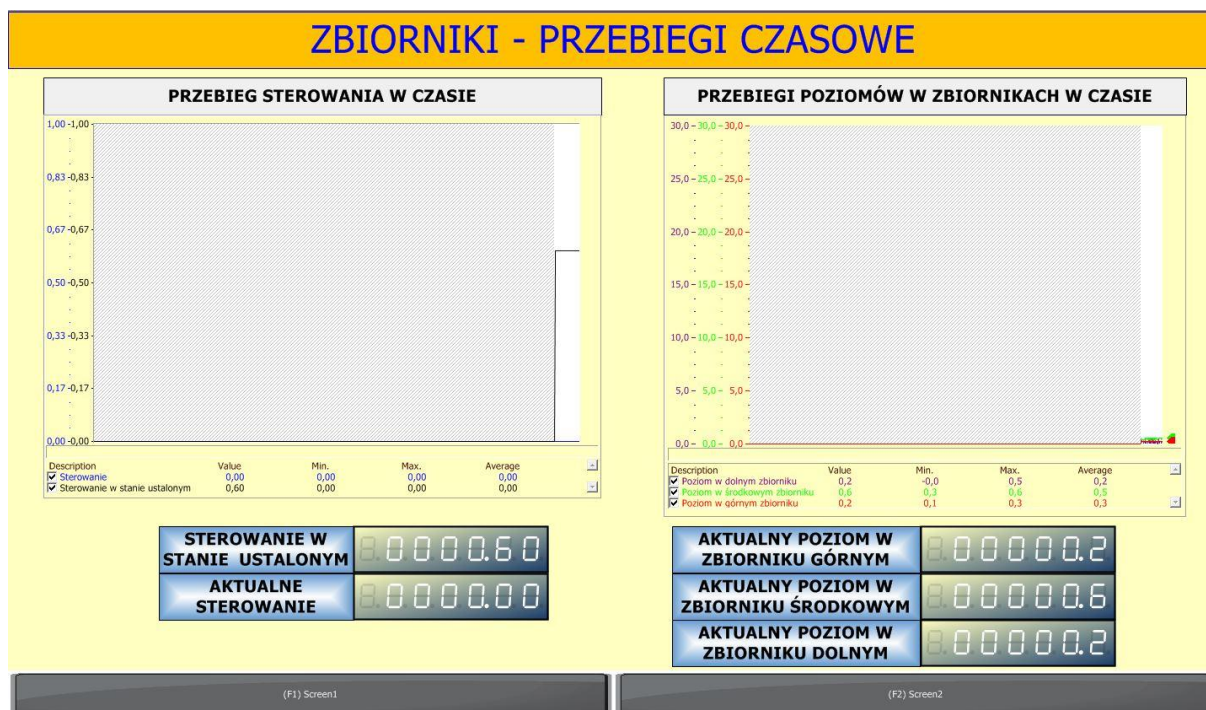


Rys. 31 Widok pierwszego ekranu do wizualizacji.

Za pomocą ekranu z rysunku 31 umożliwiono:

- wybór trybu pracy pomiędzy ręczną i automatyczną,
- ustawienie współczynnika wypełnienia sygnału PWM dla zaworów,
- włączenie lub wyłączenie działania regulatora, korzystając z przycisków "START" i "STOP",
- wybór jednego z pięciu trybów (zestawu zadanych poziomów dla każdego zbiornika)

Dodatkowo na ekranie wyświetlano wartości elementów macierzy K oraz zadane poziomy w poszczególnych zbiornikach. W celu informowania użytkownika o aktualnym stanie wykonywania programu zastosowano diody sygnalizacyjne.



Rys. 32 Widok drugiego ekranu do wizualizacji.

Zdecydowano się na umieszczenie przebiegów sterowania oraz poziomów w zbiornikach na dwóch różnych wykresach. Za pomocą drugiego ekranu umożliwiono podgląd:

- wartości sterowania w stanie ustalonym,
- aktualnego sterowania,
- aktualnych poziomów w zbiornikach.

Początkowo testy przeprowadzono bez zadawania zakłóceń. W tabeli 7 zamieszczono otrzymane wartości poziomów cieczy po wykonaniu testów na obiekcie rzeczywistym, a w tabeli 8 różnice pomiędzy wynikami otrzymanymi symulacyjnie i eksperymentalnie.

Tab. 7 Otrzymane poziomy wody w poszczególnych zbiornikach po wykonaniu testów na obiekcie rzeczywistym.

Tryb	Poziom w górnym zbiorniku w stanie ustalonym [cm]	Poziom w środkowym zbiorniku w stanie ustalonym [cm]	Poziom w dolnym zbiorniku w stanie ustalonym [cm]
1	5,2	10,4	7,4
2	7,1	14,3	9,4
3	9,2	18,1	11,6
4	11,5	23,1	14,3
5	13,8	26,0	16,6

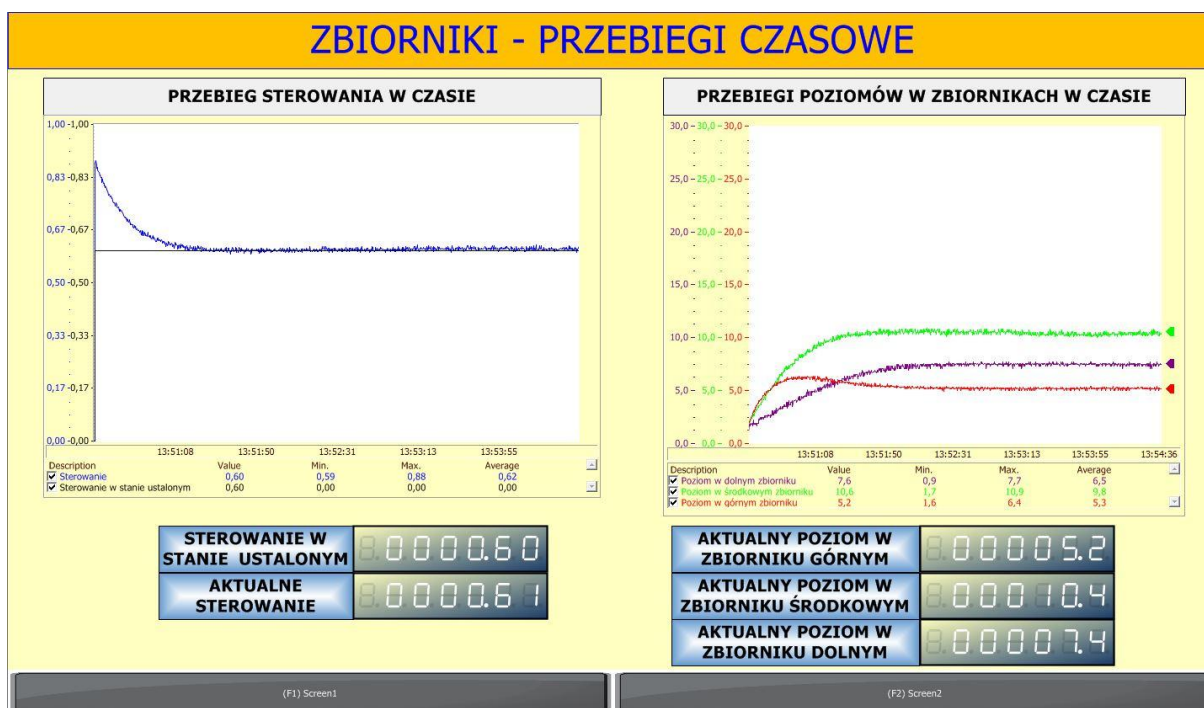
Tab. 8 Różnice pomiędzy poziomami wody w poszczególnych zbiornikach dla wyników symulacyjnych i eksperymentalnych.

Tryb	Różnica dla górnego zbiornika [cm]	Różnica dla środkowego zbiornika [cm]	Różnica dla dolnego zbiornika [cm]
1	0,3	0,1	0
2	0,3	0,1	0,2
3	0,4	0,1	0
4	0,3	0,6	0,2
5	0,2	0,5	0,1

Na podstawie tabeli 7 wykazano stabilizację cieczy na zadanych poziomach. Różnice poziomów po testach na obiekcie rzeczywistym w stosunku do wartości otrzymanych symulacyjnie nie przekroczyły 0,6 centymetra. Prawdopodobnymi przyczynami niewielkich różnic były:

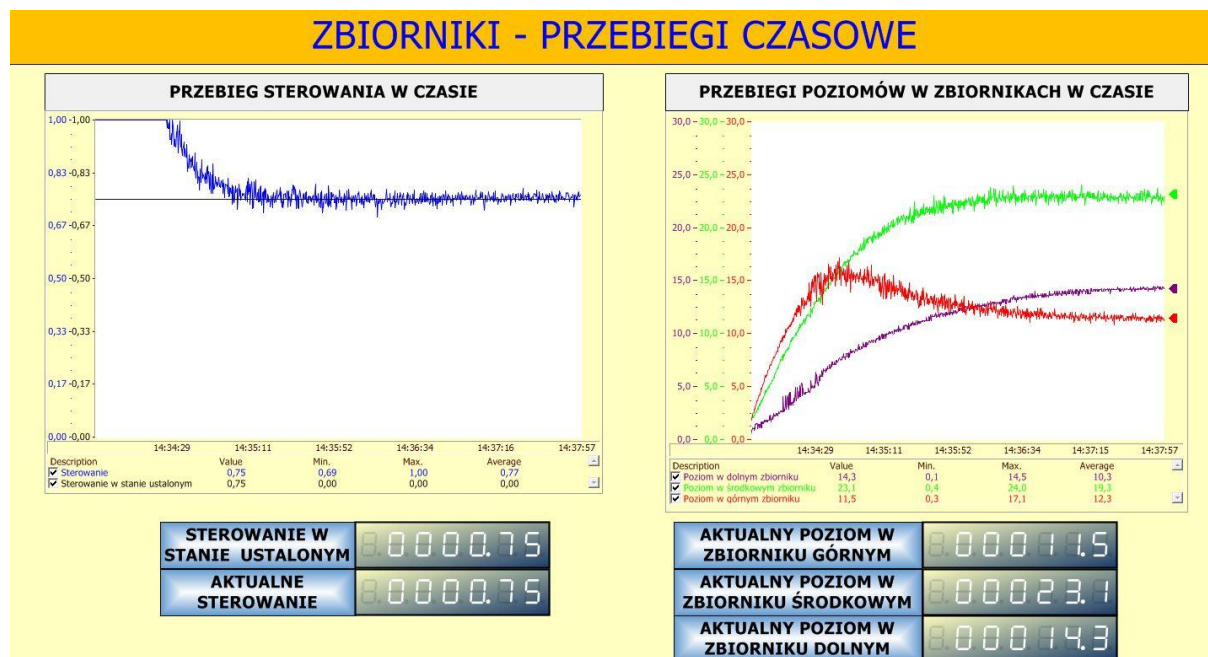
- brak części całkującej w regulatorze (uchyb ustalony),
- niedokładność pomiaru czujników ciśnienia,

Na rysunkach 33 i 34 przedstawiono widok drugiego ekranu wizualizacji dla dwóch wybranych trybów.



Rys. 33 Widok drugiego ekranu wizualizacji dla trybu 1.

ZBIORNIKI - PRZEBIEGI CZASOWE



Rys. 34 Widok drugiego ekranu wizualizacji dla trybu 4.

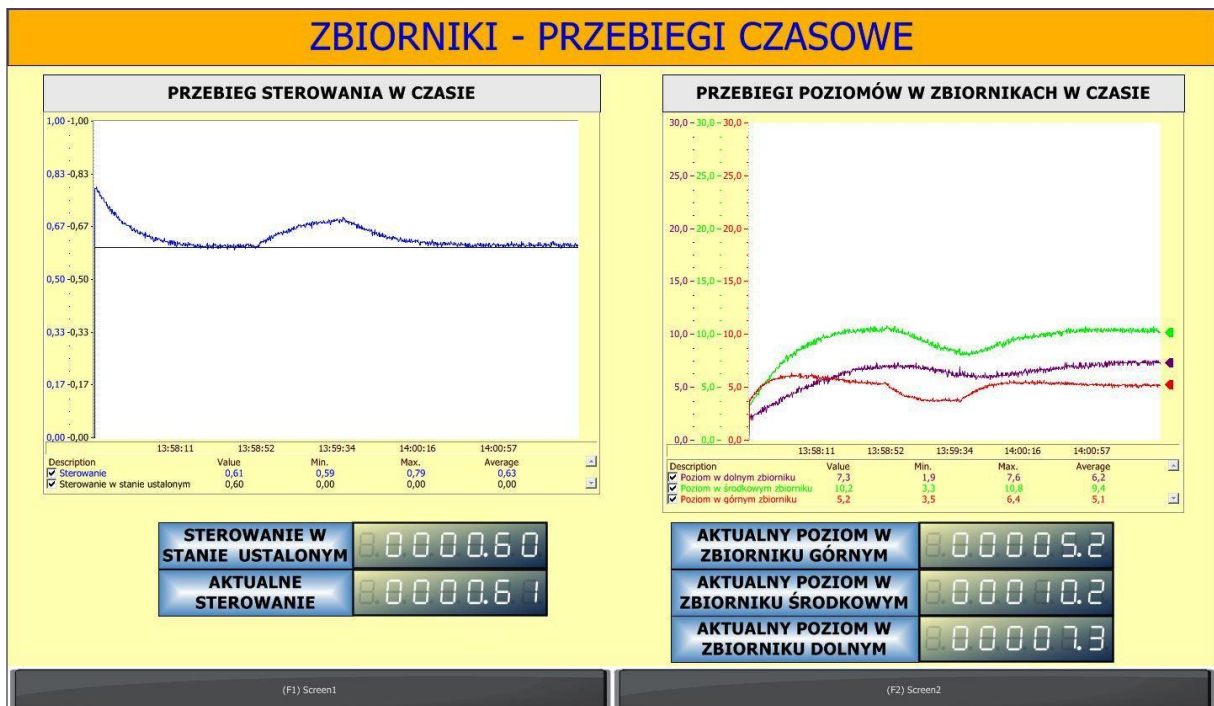
Analiza przebiegów z rysunków 33-34 pozwala stwierdzić, że dla wyższej wartości sterowania w stanie ustalonym widoczne są szумы w przebiegach poziomów oraz sterowania.

Końcowy etap testów polegał na zadawaniu zakłóceń w postaci otwarcia zaworów elektromagnetycznych. W tabeli 9 przedstawiono otrzymane wartości poziomów w poszczególnych zbiornikach po zadaniu zakłóceń.

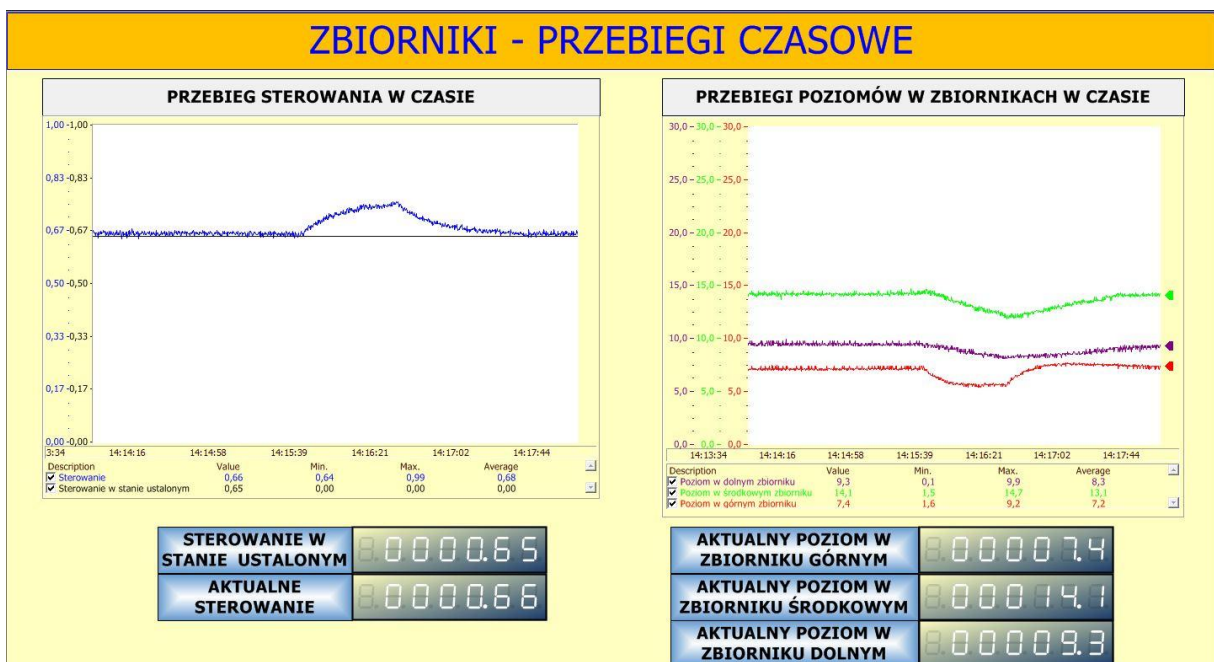
Tab. 9 Otrzymane poziomy w poszczególnych zbiornikach dla każdego z trybów po zadaniu zakłóceń.

Tryb	Poziom w górnym zbiorniku w stanie ustalonym [cm]	Poziom w środkowym zbiorniku w stanie ustalonym [cm]	Poziom w dolnym zbiorniku w stanie ustalonym [cm]
1	5,2	10,2	7,3
2	7,4	14,1	9,3
3	9,4	18,4	11,6
4	11,6	22,5	14,1
5	14,0	25,9	16,1

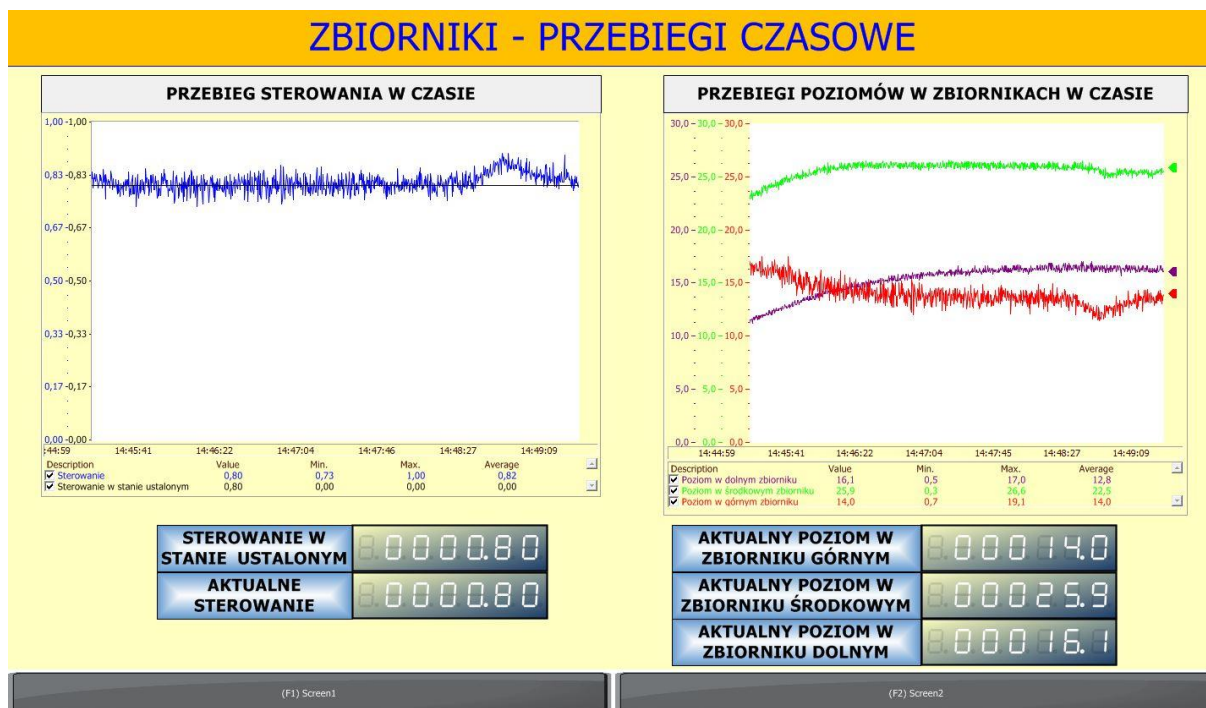
Porównanie wyników z tabel 7 i 9 wykazują, że regulator prawidłowo reagował w przypadku wystąpienia zakłóceń. Poprawność tego wniosku przedstawiają widoki drugiego ekranu wizualizacji zamieszczone na rysunkach od 35 do 37.



Rys. 35 Widok drugiego ekranu wizualizacji dla trybu 1 po podaniu zakłóceń.



ZBIORNIKI - PRZEBIEGI CZASOWE



Rys. 37 Widok drugiego ekranu wizualizacji dla trybu 4 po podaniu zakłóceń.

Podobnie, jak to miało miejsce dla testów bez podawania zakłóceń, im wyższa wartość sterowania w stanie ustalonym tym przebiegi były bardziej zaszumione. Powodem tej sytuacji był fakt, że przy wyższym sterowaniu, poziomy w zbiornikach szybko zwiększały swoją wartość przez co czujniki ciśnienia nie były w stanie poprawnie odczytać wartości poziomów.

6 Podsumowanie

Celem pracy inżynierskiej było wykorzystanie metody generacji kodu ST z programu Simulink, do sterowania kaskadą trzech zbiorników. Istotna część kodu systemu sterowania uruchamianego na sterowniku PLC powstała w formie diagramów graficznych Simulinka. Kod ten został następnie użyty w środowisku PC Worx. Zdecydowano się na takie postępowanie, aby testy przeprowadzić w pierwszej kolejności symulacyjnie, a następnie bezpośrednio na obiekcie rzeczywistym. Zastosowanie przybornika PLC Coder pozwoliło na zapoznanie się z wadami oraz zaletami automatycznej generacji kodu ST. Przeprowadzone testy oraz wykonanie pracy prowadzą do następujących wniosków:

- tworzenie schematów blokowych z wykorzystaniem Simulinka jest prostsze od konstruowania programów opierających się na językach zgodnych z normą IEC 61131-3 (zaleta),
- mała ilość kroków potrzebnych do prawidłowego dokonania generacji kodu ST (zaleta),
- wykorzystanie jednego programu do identyfikacji, zaprojektowania regulatora oraz testowania stworzonego algorytmu (zaleta),
- możliwość przeprowadzenia generacji kodu dla większości znanych firm produkujących sterowniki PLC (zaleta),
- wsparcie przez PLC Coder dla ograniczonej ilości bloków dostępnych w Simulinku (wada),

Powyższe stwierdzenia pozwalają postawić tezę, że PLC Coder to dobra alternatywa dla środowisk zintegrowanych ze sterownikami, które umożliwiają pisanie kodu za pomocą tekstu strukturalnego. Jedyną zauważalną wadą powoduje, że ogromne możliwości Simulinka nie są w pełni wykorzystywane, co jednak nie implikuje odrzucenia tego sposobu do realizacji programów wgrywanych na PLC.

Stworzony projekt inżynierski podyktowany był chęcią wykonania, jak największej ilości elementów koniecznych do realizacji regulatora LQ za pomocą oprogramowania Simulink. Ostatecznie poziomy w zbiornikach stabilizowały się w pobliżu ustalonych wartości zadanych, zgodnie z początkowymi założeniami pracy. Dalsze możliwości rozwoju projektu inżynierskiego to:

- synteza regulatora opartego na sieciach neuronowych,
- jednoczesne sterowanie pompą oraz zaworami elektromagnetycznymi,
- wykorzystanie panelu operatorskiego do sterowania kaskadą zbiorników.

Bibliografia

- [1] Oprogramowanie Naukowo-Techniczne, *Simulink*, Strona Internetowa, Polska, Zasoby Sieci Internet
<http://www.ont.com.pl/produkty/lista-produktow/simulink/>
- [2] Phoenix Contact, PC Worx User's Manual, Dokumentacja techniczna, wersja 7127_en_03, Niemcy, 27.07.2011, Zasoby sieci Internet
http://media.digikey.com/pdf/Data%20Sheets/Phoenix%20Contact%20PDFs/PC_WorX_QSG.pdf
- [3] The Mathworks Inc, PLC Coder User's Guide, Dokumentacja techniczna, USA, 2010, Zasoby Sieci Internet
<http://www.manualslib.com/manual/392933/MATLAB-Simulink-Plc-Coder-1.html#manual>
- [4] *Równania laboratoryjnego modelu zbiorników*, Materiały do laboratorium problemowego, AGH, Polska, Zasoby sieci Internet
http://aq.ia.agh.edu.pl/Aquarium/Dydaktyk/Laborat/LP/model_zbiornikow.pdf
- [5] Inteco Inc, *Multitank System User's Manual*, Dokumentacja techniczna, Polska, Zasoby sieci Internet
<http://www.diee.unica.it/~pisano/3TankUserManual.pdf>
- [6] Twardowski T. *Dokładność pomiaru i błąd w przetwarzaniu sygnału pomiarowego*, Wykład z komputerowej analizy systemów pomiarowych, AGH, Katedra Metrologii, Kraków 2004, Zasoby sieci Internet
<http://webcache.googleusercontent.com/search?q=cache:Zf7t9B7aSyAJ:home.agh.edu.pl/~ttward/model/w9.doc+&cd=1&hl=pl&ct=clnk&gl=pl>
- [7] Baranowski J. , Mitkowski W. , Tutaj A. *Teoria Sterowania - Materiały pomocnicze do ćwiczeń laboratoryjnych*, Wydawnictwo AGH, Kraków 2007
- [8] Duzinkiewicz K. Grochowski M. Piotrowski R. *Linearyzacja statyczna i dynamiczna*, Materiały pomocnicze do ćwiczeń z podstaw automatyki, Politechnika Gdańska, Wydział Elektrotechniki i Automatyki, Gdańsk 2009, Zasoby sieci Internet
http://eia.pg.edu.pl/documents/184139/28394332/PA_2009_2010_CW_T7_Material_pomocniczy.pdf
- [9] Mitkowski W. *Stabilizacja systemów dynamicznych*, Wydawnictwo AGH, Kraków 1991
- [10] Czajczyk Ł. Chornikowski Ł. Kania T. *Problem liniowo kwadratowy*

z nieskończonym czasem sterowania, Notatki w Internecie, Polska, Zasoby sieci Internet

http://zasoby1.open.agh.edu.pl/dydaktyka/automatyka/c_3_bad_operac_elektrotec hnika_fizyka/teoria_sterowania.php?subpage=wyklad&id=39

- [11] Phoenix Contact, *IB IL AI 4/U-PAC Inline analog input terminal*, Dokumentacja techniczna, wersja 8081_en_01_C01, Niemcy, 14.04.2015
- [12] Phoenix Contact, *IB IL PWM/2 (-PAC) Inline function terminal for pulse width modulation and frequency modulation*, Dokumentacja techniczna, wersja 6920_en_01, Niemcy, 03.2007.
- [13] Phoenix Contact, *Visualization software for Windows systems version 2.20 or later*, Dokumentacja techniczna, wersja 8150_en_01, Niemcy, 25.02.2014

Spis rysunków

RYS. 1 OBIEKT STEROWANIA.	9
RYS. 2 SCHEMAT BLOKOWY POŁĄCZEŃ POMIĘDZY OBIEKTEM STEROWANIA, STEROWNIKIEM MOCY ORAZ KOMPUTEREM.	10
RYS. 3 ZDJĘCIE STEROWNIKA ILC 350 PN FIRMY PHOENIX CONTACT.	12
RYS. 4 WYMIARY GEOMETRYCZNE GÓRNEGO ZBIORNIKA.	14
RYS. 5 WYMIARY GEOMETRYCZNE ŚRODKOWEGO ZBIORNIKA.	15
RYS. 6 WYMIARY GEOMETRYCZNE DOLNEGO ZBIORNIKA.	15
RYS. 7 CHARAKTERYSTYKA PRZETWARZANIA DLA CZUJNIKA ZNAJDUJĄCEGO SIĘ W GÓRNYM ZBIORNIKU.	17
RYS. 8 CHARAKTERYSTYKA PRZETWARZANIA DLA CZUJNIKA ZNAJDUJĄCEGO SIĘ W ŚRODKOWYM ZBIORNIKU.	17
RYS. 9 CHARAKTERYSTYKA PRZETWARZANIA DLA CZUJNIKA ZNAJDUJĄCEGO SIĘ W DOLNYM ZBIORNIKU.	18
RYS. 10 CHARAKTERYSTYKA NAPEŁNIANIA DLA WSPÓŁCZYNNIKA WYPEŁNIENIA RÓWNEGO 1.	19
RYS. 11 CHARAKTERYSTYKA WYDAJNOŚCI POMPY.	19
RYS. 12 OSTATECZNA CHARAKTERYSTYKA WYDAJNOŚCI POMPY.	20
RYS. 13 METODA STROJONEGO MODELU.	21
RYS. 14 MODEL GÓRNEGO ZBIORNIKA WYKONANY W SIMULINKU.	22
RYS. 15 PORÓWNANIE CHARAKTERYSTYK WYPŁYWU W GÓRNYM ZBIORNIKU.	23
RYS. 16 PORÓWNANIE CHARAKTERYSTYK WYPŁYWU W ŚRODKOWYM ZBIORNIKU.	24
RYS. 17 PORÓWNANIE CHARAKTERYSTYK WYPŁYWU W DOLNYM ZBIORNIKU.	24
RYS. 18 PODSYSTEM ODPOWIEDZIALNY ZA WYBÓR ODPOWIEDNICH WARTOŚCI ZADANYCH ORAZ MACIERZY WZMOCNIEŃ W ZALEŻNOŚCI OD WYBRANEGO TRYBU.	29
RYS. 19 PODSYSTEM REALIZUJĄCY REGULATOR LQ.	30
RYS. 20 PODSYSTEM DOKONUJĄCY WYBÓR POMIĘDZY PRACĄ RĘCZNĄ I AUTOMATYCZNĄ.	30
RYS. 21 PRZEBIEGI SYMULACYJNE POZIOMÓW WODY W ZBIORNIKACH DLA TRYBU 1.	32
RYS. 22 PRZEBIEGI SYMULACYJNE POZIOMÓW WODY W ZBIORNIKACH DLA TRYBU 2.	32
RYS. 23 PRZEBIEGI SYMULACYJNE POZIOMÓW WODY W ZBIORNIKACH DLA TRYBU 3.	33
RYS. 24 PRZEBIEGI SYMULACYJNE POZIOMÓW WODY W ZBIORNIKACH DLA TRYBU 4.	33
RYS. 25 PRZEBIEGI SYMULACYJNE POZIOMÓW WODY W ZBIORNIKACH DLA TRYBU 5.	34
RYS. 26 PRZEBIEGI SYGNAŁU STERUJĄCEGO DLA KAŻDEGO Z TRYBÓW ROZPATRYWANYCH W PRACY.	34
RYS. 27 PODSYSTEM REALIZUJĄCY OBSŁUGĘ WEJŚĆ ANALOGOWYCH STEROWNIKA.	37

RYS. 28 PODSYSTEM REALIZUJĄCY OGRANICZENIE ORAZ KONWERSJĘ WSPÓŁCZYNNIKA WYPEŁNIENIA SYGNAŁU PWM	38
RYS. 29 PODSYSTEM REALIZUJĄCY UMIESZCZENIE W SŁOWIE WYJŚCIOWYM MODUŁU WSPÓŁCZYNNIKÓW WYPEŁNIENIA SYGNAŁÓW PWM DLA POMPY I ZAWORÓW.	39
RYS. 30 SCHEMAT BLOKOWY GŁÓWNEGO PROGRAM URUCHAMIANEGO NA STEROWNIK PLC...	40
RYS. 31 WIDOK PIERWSZEGO EKRANU DO WIZUALIZACJI.	41
RYS. 32 WIDOK DRUGIEGO EKRANU DO WIZUALIZACJI.....	42
RYS. 33 WIDOK DRUGIEGO EKRANU WIZUALIZACJI DLA TRYBU 1.	43
RYS. 34 WIDOK DRUGIEGO EKRANU WIZUALIZACJI DLA TRYBU 4.	44
RYS. 35 WIDOK DRUGIEGO EKRANU WIZUALIZACJI DLA TRYBU 1 PO PODANIU ZAKŁÓCEŃ.....	45
RYS. 36 WIDOK DRUGIEGO EKRANU WIZUALIZACJI DLA TRYBU 2 PO PODANIU ZAKŁÓCEŃ.....	45
RYS. 37 WIDOK DRUGIEGO EKRANU WIZUALIZACJI DLA TRYBU 4 PO PODANIU ZAKŁÓCEŃ.....	46

Spis tabel

TAB. 1 WYKAZ MODUŁÓW STEROWNIKA ILC 350 INLINE.	11
TAB. 2 WSPÓLCZYNNIKI WIELOMIANU CHARAKTERYSTYK PRZETWARZANIA DLA POSZCZEGÓLNYCH CZUJNIKÓW.	16
TAB. 3 WSPÓLCZYNNIKI TRÓJMIANU DLA OSTATECZNEJ CHARAKTERYSTYKI WYDAJNOŚCI POMPY.	20
TAB. 4 ZIDENTYFIKOWANE WARTOŚCI WSPÓLCZYNNIKÓW WYPŁYWU I POCZĄTKOWYCH POZIOMÓW CIECZY.	23
TAB. 5 POZIOMY CIECZY, STEROWANIE W STANIE USTALONYM ORAZ MACIERZ K DLA KAŻDEGO TRYBU.	31
TAB. 6 FRAGMENT WYGENEROWANEGO KODU REALIZUJĄCY OBSŁUGĘ WEJŚĆ ANALOGOWYCH.	37
TAB. 7 OTRZYMANE POZIOMY WODY W POSZCZEGÓLNYCH ZBIORNIKACH PO WYKONANIU TESTÓW NA OBIEKCIE RZECZYWISTYM.	42
TAB. 8 RÓŻNICE POMIĘDZY POZIOMAMI WODY W POSZCZEGÓLNYCH ZBIORNIKACH DLA WYNIKÓW SYMULACYJNYCH I EKSPERYMENTALNYCH.	43
TAB. 9 OTRZYMANE POZIOMY W POSZCZEGÓLNYCH ZBIORNIKACH DLA KAŻDEGO Z TRYBÓW PO ZADANIU ZAKŁÓCEŃ.	44