



POLITECHNIKA ŚLĄSKA

WYDZIAŁ AUTOMATYKI, ELEKTRONIKI I INFORMATYKI

Praca dyplomowa magisterska

Sterownik urządzeń pomiarowych zrealizowany na komputerze PDA

Autor: Roman Gajdzik

Kierujący pracą: dr inż. Józef Wiora

Gliwice, wrzesień 2014

Spis treści

1. Wstęp	3
2. Cel i zakres pracy	4
3. Struktura obiektu rzeczywistego w postaci kotłowni wraz z instalacją centralnego ogrzewania	6
4. Część sprzętowa sterownika urządzeń pomiarowych	8
4.1. Tablet PC	9
4.2. Konwerter Bluetooth/RS-232	11
5. Część programowa sterownika urządzeń pomiarowych	13
5.1 Środowisko programistyczne	14
5.2 Struktura aplikacji sterownika urządzeń pomiarowych	15
5.3 Protokół komunikacyjny pomiędzy sterownikiem urządzeń pomiarowych a modulem regulatora kotłowni	20
5.4 Graficzny interfejs użytkownika	22
6. Symulator obiektu rzeczywistego	30
7. Podstawy teoretyczne	32
7.1 Symulacja komputerowa obiektów dynamicznych	32
7.2 Metoda Eulera	33
7.3 Metoda wykorzystująca transformatę "Z"	37
7.4 Regulator krokowy	38
7.5 Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	41
7.6 Nastawy regulatorów	43
7.7 Jakość regulacji	45
8. Opis testów	46
8.1 Testy jakości regulacji za pomocą regulatora krokowego	49
8.2 Testy jakości regulacji za pomocą algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	67
8.3 Porównanie wyników testów	86

8.4 Test algorytmu sterowania układem solarnym oraz algorytmu sterowania pracą pompy w zbiorniku wody użytkowej	96
9. Podsumowanie.....	97
10. Kierunki dalszego rozwoju projektu.....	99
Bibliografia	101
Spis załączników	103

1. Wstęp

Od początku swego istnienia ludzkość dążyła do poprawy jakości własnego życia poprzez tworzenie narzędzi oraz urządzeń, za pomocą których była w stanie stawiać czoła niedogodnościom codziennego życia. Z upływem czasu problemy z jakimi borykał się człowiek stawały się coraz bardziej skomplikowane oraz złożone. Prowadziło to do powstawania innowacyjnych rozwiązań, których przykładem jest pojawienie się układów regulacji automatycznej. Już w starożytności konstruowano urządzenia wykorzystujące sprzężenie zwrotne, które stanowi podstawę automatyki. Twórcy tych urządzeń nie byli jednak świadomi rewolucyjnej myśli jaką zawarli w swych wynalazkach. Za kamień milowy w rozwoju urządzeń automatyki przyjmuje się wynalezienie przez Korneliusza Drebbel'a, około 1620 roku regulatora temperatury paleniska. Prawdziwy przełom nastąpił jednak wraz z rozpoczęciem rewolucji przemysłowej w Europie pod koniec XVIII wieku. Szybki rozwój techniki oraz matematycznej teorii sterowania sprawił, iż powstawały coraz to dokładniejsze oraz sprawniejsze analogowe układy regulacji. Usprawnienie analogowych układów regulacji pozwoliło na zastąpienie omylnego oraz niedokładnego czynnika ludzkiego maszynami, których dokładność oraz wydajność pozostawała zawsze stała. Doprowadziło to do jeszcze szybszego rozwoju zarówno technicznego, jak i gospodarczego [1].

Na naszych oczach rozgrywa się kolejna rewolucja w dziedzinie automatyki. Wraz z rozwojem elektroniki, który przypadł na drugą połowę XX wieku, automatyka stała się prężnie rozwijającą dziedziną nauki. Coraz dokładniejsze urządzenia pomiarowe, nastawcze oraz regulatory powodują, iż cyfrowe układy regulacji wypierają swych analogowych przodków, poprawiając tym samym dokładność oraz wydajność wielu procesów przemysłowych. Postęp w dziedzinie elektroniki spowodował, iż urządzenia elektroniczne, które mogą zostać wykorzystane do syntezy układów regulacji, stały się powszechne. Doskonałym przykładem są urządzenia klasy PDA (ang. Personal Digital Assistant). Są to małych rozmiarów przenośne komputery osobiste, które charakteryzują się tym, iż są programowalne. Ich moc obliczeniowa jest na tyle duża, iż mogą zostać

wykorzystane do zaimplementowania dowolnego algorytmu sterowania. Ponadto, urządzenia takie posiadają wiele interfejsów (Bluetooth, Wi-Fi, itp.) pozwalających na komunikację z innymi urządzeniami elektronicznymi. Jednym z popularniejszych urządzeń klasy PDA jest tablet z systemem operacyjnym Android. Urządzenie takie oferuje wiele możliwości, jest stosunkowo tanie, a wiedza na temat sposobu programowania jest ogólnie dostępna w sieci internetowej.

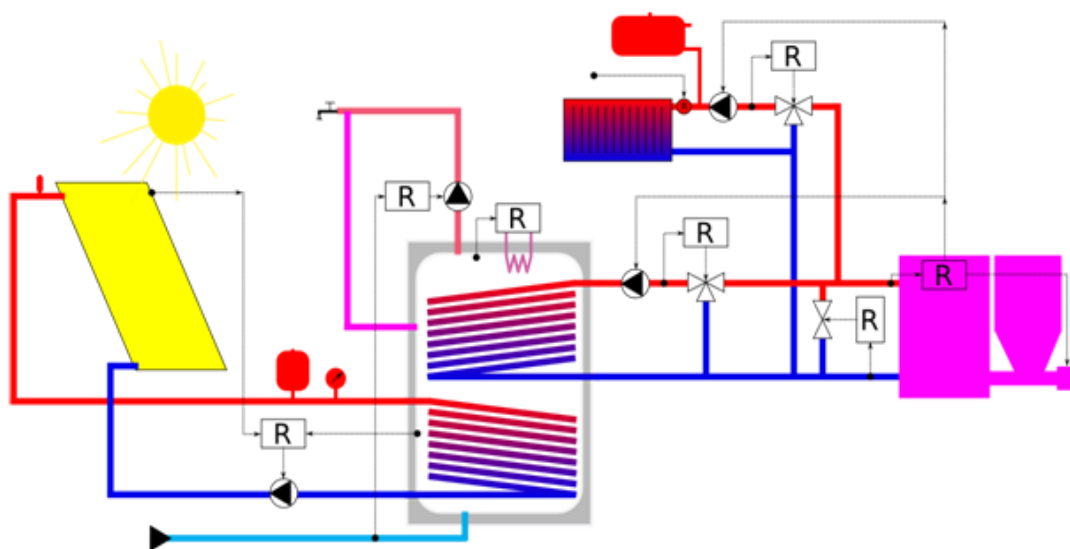
Niniejsza praca poświęcona jest próbie ukazania możliwości, jakie dają współczesne urządzenia elektroniczne w kontekście syntezy układu regulacji. Ponadto, celem pracy jest określenie wydajności stosowania urządzeń klasy PDA w układach regulacji. Niniejsze opracowanie porusza zagadnienia dotyczące doboru urządzeń elektronicznych, ich oprogramowania oraz teorii sterowania.

2. Cel i zakres pracy

Zasadniczym celem pracy dyplomowej było stworzenie sterownika urządzeń pomiarowych bazując na urządzeniu klasy PDA¹. Niniejsza praca jest kontynuacją projektu inżynierskiego pt. „Moduł regulatora kotłowni” autorstwa Wojciecha Grajdka [3]. Projekt ten został zwieńczony zbudowaniem modułu, który pełni rolę urządzenia pośredniczącego pomiędzy urządzeniami wykonawczymi w postaci pomp i elektrozaworów oraz przyrządami pomiarowymi, a sterownikiem urządzeń pomiarowych. Głównym zadaniem sterownika urządzeń pomiarowych jest obliczenie sygnałów sterujących dla poszczególnych urządzeń wykonawczych, na podstawie zaimplementowanych algorytmów sterowania oraz sygnałów pochodzących z czujników temperatury i przepływu. Wartości obliczonych sygnałów sterujących zostają przesłane za pośrednictwem przyjętego protokołu oraz interfejsu do modułu regulatora kotłowni zbudowanego przez Wojciecha Grajdka. Założono, iż sterownik musi być mobilny tzn. nie musi

¹ Urządzenie klasy PDA – (ang. Personal Digital Assistant) jest to przenośne urządzenie, które służy jako osobisty menadżer danych. Urządzenia PDA wykorzystywane są do korzystania z internetu, aplikacji biurowych, aplikacji multimedialnych. Posiada możliwość łączenia się z innymi urządzeniami poprzez interfejsy, takie jak Bluetooth lub Wi-Fi [2].

znajdować się w pobliżu modułu regulatora kotła. Rysunek 2.1. przedstawia schemat kotłowni wraz z instalacją centralnego ogrzewania, dla której zbudowany został omawiany sterownik. Natomiast struktura obiektu zostanie omówiona w następnym rozdziale.



Rysunek 2.1. Schemat kotłowni wraz z instalacją centralnego ogrzewania [4].

W celu określenia jakości regulacji obiektu za pomocą zbudowanego sterownika należało przeprowadzić szereg badań, które dostarczyły obiektywnych informacji na temat procesu regulacji. Badania te zostały przeprowadzone z wykorzystaniem zaprogramowanego na komputerze klasy PC symulatora obiektu rzeczywistego. Brak dostępu do obiektu rzeczywistego wymusił użycie wyżej wymienionego symulatora.

Zakres pracy obejmuje:

- omówienie struktury obiektu rzeczywistego oraz wydzielenie pętli regulacyjnych występujących w obiekcie,
- omówienie struktury sterownika urządzeń pomiarowych,
- dobór urządzeń oraz elementów elektronicznych umożliwiających zbudowanie sterownika urządzeń pomiarowych,
- omówienie struktury oprogramowania sterownika urządzeń pomiarowych,

- dobór oprogramowania umożliwiającego programowanie urządzenia klasy PDA,
- omówienie zagadnień teoretycznych związanych z układami regulacji, doborem nastaw regulatorów oraz symulacją obiektów rzeczywistych,
- zaprogramowania sterownika urządzeń pomiarowych,
- zaprogramowanie symulatora obiektu rzeczywistego,
- przeprowadzenie badań jakości regulacji w układzie regulacji obiektu ze sterownikiem urządzeń pomiarowych.

3. Struktura obiektu rzeczywistego w postaci kotłowni wraz z instalacją centralnego ogrzewania

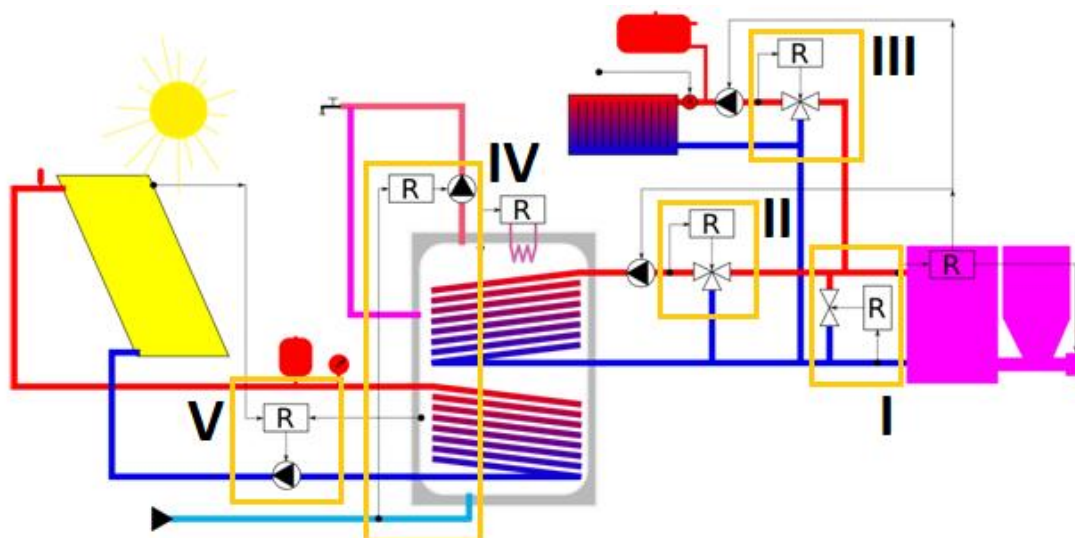
Układ kotłowni składa się z następujących elementów:

- kocioł C.O.,
- grzejniki C.O.,
- zbiornik wody użytkowej,
- panel solarny,
- rurociągi,
- elektrozapory,
- pompy C.O.

Omówienie struktury kotłowni wraz z instalacją centralnego ogrzewania należy zacząć od jej głównego elementu czyli kotła C.O. Kocioł zasilany jest paliwem w postaci węgla, który magazynowany jest w specjalnym zasobniku. Zadaniem kotła C.O. jest podgrzewanie wody, która następnie siecią rurociągów jest rozprowadzana po całym układzie. Ilość węgla podawanego przez podajnik w czasie procesu spalania regulowana jest przez wbudowany sterownik. Dodatkowym zadaniem wbudowanego sterownika jest sterowanie pracą pomp C.O. umieszczonych na dopływie podgrzanej wody do zbiornika wody użytkowej oraz układu grzejników. Praca tych pomp powoduje szybsze rozprowadzanie podgrzanej wody po całym układzie.

Kolejnym elementem, omawianego układu, jest zbiornik wody użytkowej wraz z panelem solarnym. Jest to zbiornik, w którym podgrzewana jest woda przeznaczona do wykorzystania przez użytkowników budynku, w którym znajduje się instalacja. Energia niezbędna do podniesienia temperatury w zbiorniku wody użytkowej jest dostarczana trzema niezależnymi drogami. W okresie grzewczym woda ogrzewana jest energią pochodzącą z wody podgrzanej w kotle C.O. Kolejnym źródłem energii jest energia pochodząca z panelu solarnego. Ponadto, układ zasilany jest ciepłem uzyskiwanym dzięki pracy elektrycznego podgrzewacza wody. Podgrzewacz ten działa w pętli regulacyjnej, co pozwala na utrzymanie temperatury wody w zbiorniku na zadanym poziomie. Pobieranie energii z panelu solarnego oraz wody podgrzanej w kotle C.O. pozwala na zmniejszenie kosztów podgrzewania wody użytkowej.

W układzie kotłowni wraz z instalacją centralnego ogrzewania można wyróżnić kilka pętli regulacyjnych, w których rolę regulatora pełni sterownik urządzeń pomiarowych. Rysunek 3.1. przedstawia schemat kotłowni wraz z instalacją centralnego ogrzewania z zaznaczonymi pętlami regulacyjnymi.



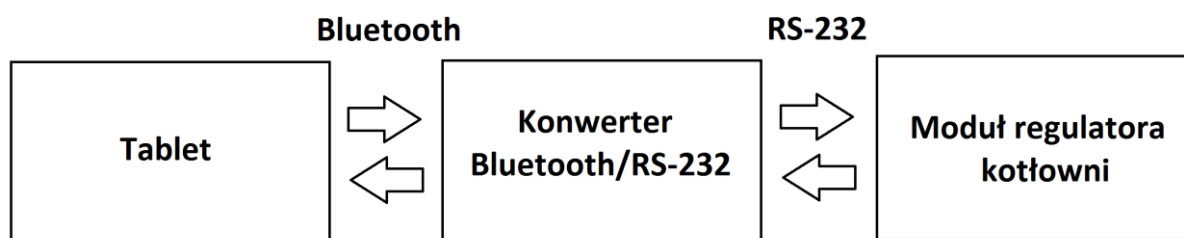
Rysunek 3.1. Schemat kotłowni wraz z instalacją centralnego ogrzewania z zaznaczonymi pętlami regulacyjnymi [4].

Pętla regulacyjna oznaczona na Rysunku 3.1. jako I ma za zadanie utrzymywać temperaturę wody wejściowej kotła na zadanym poziomie. Wymaganie utrzymywania wody wejściowej kotła na zadanym poziomie wynika z tego, iż przy pewnej temperaturze wody wejściowej zużycie kotła jest znacznie mniejsze. W celu regulacji temperatury wejściowej kotła zainstalowano urządzenie pomiarowe w postaci czujnika temperatury oraz elektrozawór, który stanowi urządzenie wykonawcze. Zadaniem pętli II jest regulacja temperatury wody, która znajduje się w zbiorniku użytkowym. W tym przypadku urządzeniem wykonawczym jest elektrozawór trójdrożny, a rolę urządzenia pomiarowego, podobnie jak w przypadku pętli I, pełni czujnik temperatury. Celem pętli regulacyjnej III jest utrzymywanie zadanej temperatury wody wchodzącej do grzejników znajdujących się w pomieszczeniach mieszkalnych. Tak jak w przypadku pętli regulacyjnej II, rolę urządzenia wykonawczego pełni elektrozawór trójdrożny, a urządzenie pomiarowe stanowi czujnik temperatury. Pętla IV ma za zadanie sterować pracą urządzenia wykonawczego w postaci pompy C.O. Dzięki urządzeniu pomiarowemu jakim jest czujnik przepływu umieszczony na przyłączy, z którego uzupełniania jest woda w zbiorniku wody użytkowej, można uzyskać informację, kiedy pobierana jest woda ze zbiornika wody użytkowej. Informacja ta pozawala na sterowanie pracą pompy C.W.U w sposób, który w znacznym stopniu przyspiesza proces dostarczenia ciepłej wody do miejsc jej poboru w całym budynku. Pętla regulacji V ma za zadanie uruchamiać urządzenie wykonawcze, którym w tym przypadku jest pompa obiegu, tylko wtedy gdy, temperatura wody w panelu solarnym jest wyższa o zadaną temperaturę od temperatury wody w zbiorniku wody użytkowej. Ponadto włączanie oraz wyłączanie pracy pompy następuje z pewną określoną histerezą. W pętli tej mamy do czynienia z dwoma urządzeniami pomiarowymi w postaci czujników temperatury.

4. Część sprzętowa sterownika urządzeń pomiarowych

Część sprzętowa sterownika urządzeń pomiarowych składa się z dwóch elementów (Rysunek 4.1.). Pierwszym z nich jest tablet PC, na którym znajduje się oprogramowanie zawierające algorytmy sterowania poszczególnych pętli

regulacyjnych omówionych w poprzednim rozdziale. Drugim elementem jest konwerter umożliwiający konwersję danych przesyłanych interfejsem RS-232 na dane obsługiwane poprzez interfejs Bluetooth.



Rysunek 4.1. Schemat połączenia modułu regulatora kotłowni ze sterownikiem urządzeń pomiarowych.

4.1. Tablet PC

Opis urządzenia, na którym zdecydowano się zaimplementować sterownik urządzeń pomiarowych należy zacząć od krótkiego zarysu historycznego urządzeń PDA. Pierwsze urządzenia PDA pojawiły się w latach dziewięćdziesiątych XX wieku [5]. Charakteryzowały się one ciekawym interfejsem pośredniczącym, pomiędzy użytkownikiem, a samym urządzeniem, w postaci rysika, który umożliwiał interakcję [6]. Wraz z upływem czasu ich możliwości zwiększały się poprzez wyposażenie w coraz szybsze procesory oraz większe moduły pamięci. Po dziś dzień spełniają one rolę nawigacji, podręcznych notatników, czytników e-booków, bądź miniaturowych, budżetowych komputerów wykorzystywanych przez programistów na własne potrzeby. Wadą tradycyjnych urządzeń PDA jest mały dostęp do informacji dotyczących ich programowania oraz słaba integracja urządzeń różnych producentów. Obecnie tradycyjne urządzenia PDA wypierane są przez swych bliskich krewnych, zwanych tabletami PC [7]. W praktyce są one potomkami tradycyjnych urządzeń PDA i idealnie wpisują się w przytoczoną definicję tych urządzeń. Różnią się natomiast, interfejsem pośredniczącym,

pomiędzy urządzeniem, a użytkownikiem. W przeciwieństwie do tradycyjnych urządzeń PDA jest to interfejs dotykowy nie wymagający żadnych wskaźników w formie rysików. Procesory oraz moduły pamięci RAM wbudowane w dzisiejsze tablety pozwalają na obsługiwanie zaawansowanych aplikacji. Zainstalowane na nich systemy operacyjne (najpopularniejsze to system operacyjny Android oraz iOS [8]) umożliwiają programistom łatwy dostęp do wszystkich zasobów fizycznych urządzenia (np. modułu GPS). Kolejnym argumentem przemawiającym za tabletami PC jest łatwość dostępu do informacji na temat ich programowania. Istnieje wiele publikacji oraz stron internetowych poświęconych tematyce programowania tabletów z systemami operacyjnymi Android lub iOS [9, 10]. Tablety są urządzeniami interaktywnymi. Posiadają możliwość komunikacji z innymi urządzeniami za pośrednictwem wielu interfejsów. Najpopularniejsze z nich to interfejs Wi-Fi oraz Bluetooth. Stopień integracji, pomiędzy urządzeniami różnych producentów stoi na wysokim poziomie. Wynika to z uniwersalności systemów operacyjnych znajdujących się na współczesnych tabletach. Ponadto, zaletą tabletów PC jest ich powszechna dostępność oraz niska cena.

Wszystkie wyżej wymienione zalety w połączeniu z wymaganiem możliwości implementacji kilku algorytmów sterowania, wynikającym z charakteru rozpatrywanego problemu, oraz wymaganiem możliwości komunikacji za pomocą interfejsu Bluetooth z modułem regulatora kotłowni sprawiły, iż ostateczny wybór urządzenia, na którym zaimplementowano sterownik urządzeń pomiarowych padł na urządzenie typu tablet PC. W niniejszej pracy został wykorzystany tablet firmy ASUS model MeMO Pad HD 7.

Najważniejsze cechy tabletu ASUS MeMO PAD HD 7 [11]:

- system operacyjny Android 4.2,
- 7'' wyświetlacz podświetlany diodami LED WXGA,
- procesor Quad-Core 1.2 GHz,
- pamięć RAM 1 GB,
- bezprzewodowa sieć WLAN802.11 b/g/n,
- bezprzewodowa sieć Bluetooth V4.0,
- interfejs micro USB,

- bateria Litowo – polimerowa 15Wh,
- wymiary: 196.8 x 120.6 x 10.8 mm (szer. x gł. x wys.),
- waga : 302 g.

4.2. Konwerter Bluetooth/RS–232

Potrzeba zastosowania konwetera danych przesyłanych za pomocą interfejsu RS-232 na dane przesyłane interfejsem Bluetooth wynika z dwóch przyczyn. Pierwszą z nich jest to, iż moduł regulatora kotłowni zbudowany przez Wojciecha Grajka komunikuje się z innymi urządzeniami wyłącznie za pośrednictwem standardu RS-232. Jest to standard szeregowej transmisji danych opracowany w latach 60 XX wieku. Moduł regulatora kotłowni posiada wyprowadzone złącze RxD (odpowiedzialne za odbiór danych), TxD (odpowiedzialne za wysyłanie danych), złącze GND (masa) oraz złącze Vcc (zasilanie). Złącza te podłączone są do interfejsu UART odpowiedzialnego za komunikację szeregową i znajdującego się w mikrokontrolerze sterującym modulem regulatora kotłowni. Drugim powodem użycia konwertera danych jest chęć redukcji zbędnego okablowania. Istnieje możliwość połączenia urządzenia typu tablet PC z modulem regulatora kotła za pośrednictwem zwykłego kabla RS-232 jednak wymagałoby to umieszczenia na stałe tabletu PC w pobliżu modułu regulatora kotłowni. Zakłada się jednak mobilność sterownika urządzeń pomiarowych.

Idealnym rozwiązaniem powyższych problemów wynikających z założeń konstrukcyjnych oraz funkcjonalnych okazał się moduł AVT 1635 [12]. Jest to łatwy w użyciu układ umożliwiający zastosowanie łączności Bluetooth w każdym systemie z interfejsem UART [12]. Układ ten oparty jest na module transmisyjnym Bluetooth BTM–222. Moduł ten, w zależności od warunków propagacji, zapewnia zasięg do 100 m. Zaletą urządzenia jest to, iż po podłączeniu go do jakiegokolwiek układu z interfejsem UART staje się „przeźroczysty” dla protokołu komunikacyjnego [12]. Oznacza to, iż konwersja danych przesyłanych standardem RS–232 na dane w standardzie Bluetooth zachodzi automatycznie, bez ingerencji użytkownika. Dodatkowym atutem modułu jest jego dostępność. Wiele sklepów oferuje zestawy złożeniowe, bądź już gotowe moduły AVT 1635. Rysunek 4.2.

[illegible]

W celu dopasowania parametrów transmisji modułu AVT 1635 do parametrów transmisji modułu regulatora kotłowni należało przeprowadzić konfigurację urządzenia AVT 1635. Konfiguracja modułu polegała na podłączeniu modułu AVT 1635 za pomocą konwertera EM 212, który dokonuje konwersji danych przesyłanych standardem RS-232 na dane zgodne ze standardem USB, do komputera PC. Następnie, za pomocą programu PuTTY, będącego konsolą zdolną do przesyłania oraz odbierania danych za pośrednictwem szeregowych portów COM, należało ustawić wymagane parametry urządzenia. W tym celu posłużono się komendami AT, które umożliwiają ustawienie odpowiednich rejestrów modułu BTM-222 odpowiedzialnych za konfigurację urządzenia. Opis wszystkich komend AT znajduje się na stronie internetowej www.kamami.pl [13].

Aby możliwa była komunikacja pomiędzy modulem regulatora kotłowni, a modulem AVT 1635 należało ustawić następujące parametry transmisji:

- szybkość transmisji : 2400 b/s,
- 8 bitów danych,
- brak bitu parzystości,
- 1 bit stopu,
- brak kontroli przepływu.

Parametry te wynikają z konstrukcji modułu regulatora kotłowni. W celu ustawienia powyższych parametrów należało użyć następujących komend AT:

- ATL* - ustawienie szybkości transmisji na 2400 b/s,
- ATC0 - brak kontroli przepływu,
- ATK0 - 1 bit stopu,
- ATM0 - brak bitu parzystości.

Domyślna liczba bitów danych była ustawiona na 8 bitów.

5. Część programowa sterownika urządzeń pomiarowych

Część programowa sterownika jest niezwykle ważna, gdyż to w niej zawierają się algorytmy sterujące poszczególnymi urządzeniami wykonawczymi. Zaimplementowano dwie wersje aplikacji sterownika urządzeń pomiarowych. W jednej z nich algorytmy regulacji oparte są o regulator krokowy, natomiast w drugiej o regulator PID. Zgodnie z filozofią aplikacji dedykowanych dla urządzeń z systemem operacyjnym Android część logiczna aplikacji sterownika urządzeń pomiarowych została napisana w języku obiektowym Java, z kolei elementy graficznego interfejsu użytkownika w języku skryptowym XML. Informacje na temat podstaw programowania w języku obiektowym Java zostały zaczerpnięte z książki autorstwa K. Sierra i B. Bates [14], a informacje na temat języka XML znalezione zostały na portalu www.developer.android.com [9].

5.1 Środowisko programistyczne

Omówienie części programowej należy rozpocząć od środowiska programistycznego, w którym programowano sterownik. Wybór zintegrowanego środowiska programistycznego Eclipse podyktowany był tym, iż środowisko to jest bezpłatne, a wsparcie oferowane przez producenta stoi na wysokim poziomie. Istnieją dwie możliwości pozyskania oprogramowania Eclipse zdolnego do programowania urządzeń z systemem operacyjnym Android. Pierwszym z nich jest pobranie tego środowiska ze strony producenta [15]. Jest to jednak środowisko programistyczne zorientowane na języki, takie jak Java, C, oraz C++. W celu uzyskania możliwości programowania urządzeń z systemem Android należy, zainstalować dodatek ADT (Android Development Tools), który rozszerza możliwości programu Eclipse o zintegrowane środowisko programistyczne dedykowane dla programistów systemu Android. Po zainstalowaniu dodatku ADT w programie Eclipse pojawia się możliwość pobrania narzędzi programowych, narzędzi służących do debugowania kodu (Android SDK Tools) oraz bibliotek, które zawierają wiele metod oraz obiektów pozwalających programiście na efektywne programowanie. Drugim sposobem pozyskania środowiska Eclipse umożliwiającego programowanie urządzeń z systemem operacyjnym Android jest pobranie paczki ze strony www.developer.android.com [16], w której znajduje się skonfigurowane środowisko programistyczne Eclipse, dodatek ADT oraz narzędzia Android SDK Tools. Podejście to jest mniej czasochłonne oraz pozwala uniknąć błędów podczas konfiguracji.

5.2 Struktura aplikacji sterownika urządzeń pomiarowych

Aplikacja sterownika urządzeń pomiarowych charakteryzuje się następującymi funkcjonalnościami:

- umożliwia połączenie z modułem regulatora kotłowni za pomocą interfejsu Bluetooth,
- implementuje algorytmy sterowania odpowiedzialne za regulację w pętlach regulacyjnych przedstawionych w rozdziale 3,
- posiada możliwość pracy w tle tzn. po przejściu do innej aplikacji algorytmy sterowania zaimplementowane w sterowniku kontynuują swoją pracę,
- umożliwia podgląd poszczególnych zmiennych procesowych,
- umożliwia podgląd pracy poszczególnych urządzeń wykonawczych w postaci pomp oraz elektrozaworów,
- umożliwia zmianę parametrów poszczególnych algorytmów sterowania,
- zapamiętuje wpisane przez użytkownika parametry poszczególnych algorytmów sterowania,
- wyświetla aktualne parametry poszczególnych algorytmów sterowania,
- uniemożliwia wpisanie błędnych parametrów poszczególnych algorytmów sterowania.

Część logiczna aplikacji sterownika urządzeń pomiarowych została napisana w języku obiektowym Java. Oznacza to, iż zgodnie z filozofią programowania obiektowego, aplikacja została podzielona na kilka jednostek zwanych klasami, które odpowiedzialne są za poszczególne funkcjonalności aplikacji. Poniżej przedstawiono opis najważniejszych klas:

- Klasa "MainActivity" – jest to klasa, której jednym z głównych zadań jest pośredniczenie w komunikacji pomiędzy, klasą odpowiedzialną za obsługę danych odbieranych oraz wysyłanych poprzez interfejs Bluetooth (klasa "ObslugaStanuPolaczenia"), a klasami służącymi do prezentacji graficznej zmiennych procesowych oraz parametrów poszczególnych algorytmów sterowania (np. klasa "PanelRegulatorPID"). Ponadto, w klasie tej

zaimplementowano moduł, umożliwiający włączenie fizycznego urządzenia Bluetooth w przypadku, gdy to urządzenie znajduje się w stanie wyłączenia. Dodatkowo zachodzi tutaj proces wyszukiwania urządzeń zewnętrznych, z którymi można połączyć sterownik urządzeń pomiarowych. Kolejnym ważnym zadaniem klasy "MainActivity" jest uruchomienie instancji klasy ServiceBluetooth, w przypadku wybrania przez użytkownika urządzenia zewnętrznego, z którym współpracować ma sterownik urządzeń pomiarowych. Klasa ServiceBluetooth umożliwia pracę aplikacji w tle i zostanie opisana w kolejnym podpunkcie. Kolejnymi funkcjami opisywanej klasy jest obsługa zdarzeń przełączania pomiędzy, kolejnymi panelami graficznymi sterownika urządzeń pomiarowych oraz zapamiętywanie w pamięci urządzenia parametrów poszczególnych algorytmów sterowania. W przypadku wyłączenia przez użytkownika aplikacji w klasie "MainActivity" zachodzi proces zamknięcia kanału komunikacyjnego Bluetooth, usunięcie poszczególnych zmiennych oraz obiektów wykorzystywanych w procesie komunikacji bezprzewodowej Bluetooth.

- Klasa "ServiceBluetooth" – klasa rozszerzająca klasę "Service". Pozwala na wykonywanie w tle bardziej złożonych operacji nie wymagających interakcji z użytkownikiem [17]. Głównym zadaniem klasy "ServiceBluetooth" jest utworzenie instancji klasy "ObslugaStanuLaczenia", której celem jest konfiguracja kanału Bluetooth służącego do przesyłania danych, a następnie stworzenie instancji klasy "ObslugaStanuPolaczenia", odpowiedzialnej za proces komunikacji poprzez kanał Bluetooth. Ponadto klasa "ServiceBluetooth" pośredniczy w wymianie danych pomiędzy, klasą "MainActivity", a klasą "ObslugaStanuPolaczenia". Poprzez rozszerzenie klasy "Service", klasa "ServiceBluetooth" zapewnia utrzymanie połączenia kanału Bluetooth między, sterownikiem urządzeń pomiarowych, a modulem regulatora kotłowni oraz bieżące wykonywanie algorytmów sterowania w przypadku przejścia aplikacji w stan działania w tle.
- Klasa "ObslugaStanuLaczenia" – klasa odpowiedzialna za konfigurację kanału Bluetooth, którym odbywać się będzie komunikacja.
- Klasa "ObslugaStanuPolaczenia" – klasa odpowiedzialna za obsługę komunikacji pomiędzy sterownikiem i modulem regulatora kotłowni poprzez

beprzewodowy interfejs Bluetooth. W klasie tej odbierane są poszczególne zmienne procesowe, wysyłane przez moduł regulatora kotłowni, a następnie na ich podstawie zostają obliczane wartości sterowania i wysyłane z powtorem do sterownika urządzeń pomiarowych. W klasie tej znajdują się instancje klas "PID" lub "RegulatorKrokowy" oraz "RegulatorSolarZbiornik", które implementują poszczególne regulatory.

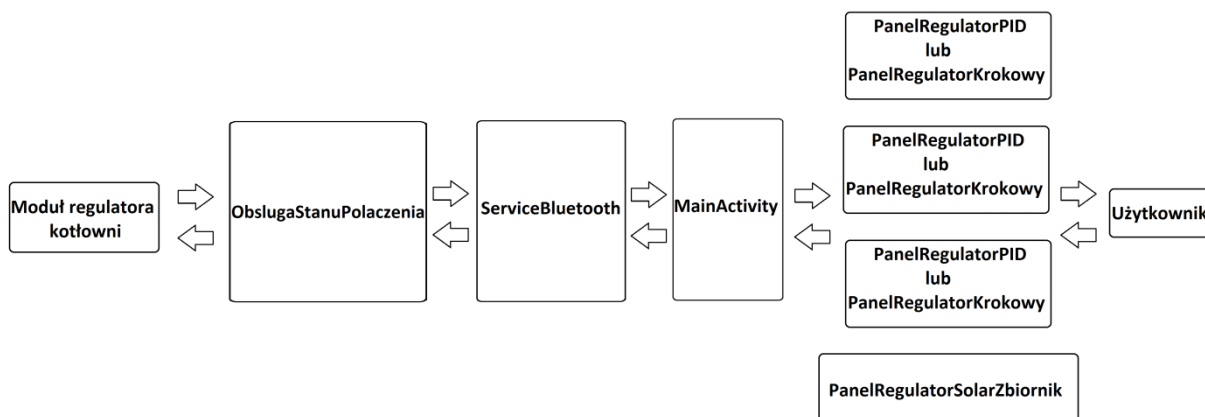
- Klasa "PanelRegulatorKrokowy" – klasa odpowiedzialna za obsługę zdarzeń wprowadzania przez użytkownika parametrów regulatora krokowego i przesyłanie wprowadzonych danych do klasy "MainActivity". Dodatkowo, zadaniem opisywanej klasy jest wyświetlanie aktualnych parametrów, zmiennych procesowych oraz stanu urządzeń wykonawczych.
- Klasa "PanelRegulatorPID" – klasa, której celem jest obsługa zdarzeń wprowadzania przez użytkownika parametrów algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym, a następnie przesyłanie wprowadzonych danych do klasy "MainActivity". Ponadto zadaniem opisywanej klasy jest wyświetlanie aktualnych parametrów, zmiennych procesowych oraz stanu urządzeń wykonawczych.
- Klasa "PanelRegulatorSolarZbiornik" – klasa służąca do obsługi zdarzeń wprowadzania parametrów histerezy występującej w algorytmie sterowania pompą wymuszającą przepływ wody przez panel solarny. Ponadto, zadaniem tej klasy jest wyświetlanie aktualnych wartości temperatur wody na zbiorniku wody użytkowej oraz na panelu solarnym. Dodatkową funkcją jest wyświetlanie stanu pracy pompy znajdującej się na zbiorniku wody użytkowej oraz pompy wymuszającej przepływ wody przez panel solarny.
- Klasa "RegulatorSolarZbiornik" – klasa, której celem jest obsługa zdarzeń wprowadzania parametrów regulatora sterującego pompą wymuszającą przepływ wody przez panel solarny oraz regulatora sterującego pracą pompy znajdującej się na zbiorniku wody użytkowej. Co więcej, zadaniem opisywanej klasy jest wyświetlanie stanów pracy obu pomp.
- Klasa "RegulatorKrokowy" – klasa implementująca regulator krokowy. Klasa ta posiada metodę "obliczSterowanieRegulatorKrokowy", która jako parametr wejściowy przyjmuje aktualną wartość regulowanej wielkości, następnie na tej podstawie oblicza aktualne sterowanie podawane na

elektrozawór. Instancje tej klasy znajdują się w klasie "ObslugaStanuPolaczenia", w której na bieżąco obliczane są sterowania przesyłane na poszczególne urządzenia wykonawcze w postaci elektrozaworów.

- Klasa "PID" – klasa implementująca algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym. Klasa ta posiada metodę "obliczPID", która jako parametr wejściowy przyjmuje aktualną wartość regulowanej wielkości i na tej podstawie oblicza aktualne sterowanie przesyłane na elektrozawór. Instancje tej klasy znajdują się w klasie "ObslugaStanuPolaczenia", w której na bieżąco obliczane są sterowania przesyłane na poszczególne urządzenia wykonawcze w postaci elektrozaworów.
- Klasa "RegulatorSolarZbiornik" - klasa, która implementuje algorytm sterowania pompą wymuszającą przepływ wody przez panel solarny oraz pompą znajdującą się na zbiorniku wody użytkowej.
- Klasa "ConnectPanel" – klasa odpowiedzialna za wyświetlenie dostępnych urządzeń zewnętrznych oraz obsługę zdarzenia wyboru przez użytkownika urządzenia, z którym sterownik ma nawiązać komunikację. Ponadto, po ustanowieniu połączenia, panel ten informuje o tym, iż sterownik nawiązał komunikację z urządzeniem zewnętrznym.
- Klasa "DialogBledneDane" – klasa umożliwiająca wyświetlanie okna z informacją o wprowadzeniu błędnych parametrów. Klasa "DialogBledneDane" używana jest w klasach odpowiedzialnych za obsługę zdarzeń wprowadzania parametrów algorytmów sterowania (np. klasa "PanelRegulatorPID"). Opisywana klasa rozszerza klasę "DialogFragment", która służy do wyświetlania okienek dialogowych [18].
- Klasa "AdapterFragments" – klasa pomocnicza, która umożliwia użytkownikowi łatwą nawigację pomiędzy poszczególnymi panelami graficznymi aplikacji.

Kolejnym zagadnieniem, które wymaga szczegółowego umówienia jest sposób przekazywania danych między poszczególnymi elementami aplikacji.

Rysunek 5.1. przedstawia schemat przepływu informacji pomiędzy modulem regulatora kotłowni, aplikacją sterownika urządzeń pomiarowych, a użytkownikiem.



Rysunek 5.1. Schemat przepływu informacji pomiędzy modulem regulatora kotłowni, aplikacją sterownika oraz użytkownikiem.

Dane przesyłane przez moduł regulatora kotłowni za pośrednictwem bezprzewodowego interfejsu Bluetooth odbierane są w klasie "ObsługaStanuPolaczenia", która zawiera instancje klas odpowiedzialnych za algorytmy sterowania poszczególnymi układami wykonawczymi. Po obliczeniu sterowań następuje przesłanie danych do modułu regulatora kotłowni. Dane na temat procesu oraz parametrów poszczególnych algorytmów sterowania przekazywane są do klasy "ServiceBluetooth", skąd trafiają do klasy "MainActivity". Komunikacja pomiędzy, klasą "ObsługaStanuPolaczenia", "ServiceBluetooth" oraz "MainActivity" zrealizowana została przy pomocy obiektu typu Handler. Obiekt ten pozwala na przekazywanie wiadomości w obrębie jednego wątku [19]. W przypadku przesłania referencji obiektu typu Handler z jednego wątku do drugiego, uzyskujemy możliwość komunikacji pomiędzy tymi wątkami. Klasa "MainActivity" za pośrednictwem odwołania się do referencji wskazujących na klasy "PanelRegulatorPID", "PanelRegulatorKrokowy" oraz "PanelRegulatorSolarZbiornik" ustawia aktualne dane na temat procesu oraz parametrów algorytmów sterowania w graficznym interfejsie użytkownika.

W przypadku zmiany parametrów algorytmów sterowania przez użytkownika, transmisja danych odbywa się analogicznie w drugą stronę, aż do momentu odebrania ich w klasie "ObsługaStanuPolaczenia".

5.3 Protokół komunikacyjny pomiędzy sterownikiem urządzeń pomiarowych a modułem regulatora kotłowni

Wykorzystywany protokół komunikacyjny został stworzony przez Wojciecha Grajdka podczas konstrukcji modułu regulatora kotłowni. Niniejszy rozdział poświęcony jest przedstawieniu składni protokołu za pośrednictwem, którego odbywa się komunikacja pomiędzy sterownikiem urządzeń pomiarowych, a modułem regulatora kotłowni.

Protokół komunikacyjny został podzielony na komendy żądań oraz komendy odpowiedzi. Komendy przyjmują postać ciągu znaków, które w celu przesłania interfejsem Bluetooth zamieniane są na ciąg bajtów. Komendy żądań wysyłane są przez sterownik urządzeń pomiarowych i mają na celu wymuszenie przesłania odpowiedzi przez moduł regulatora kotłowni. Komendy żądań mogą przyjąć jedną z następujących postaci:

- "t1" – żądanie przesłania wskazania czujnika temperatury odgrywającego rolę urządzenia pomiarowego w pętli I opisanej w rozdziale 3,
- "t2" – żądanie przesłania wskazania czujnika temperatury odgrywającego rolę urządzenia pomiarowego w pętli II opisanej w rozdziale 3,
- "t3" – żądanie przesłania wskazania czujnika temperatury odgrywającego rolę urządzenia pomiarowego w pętli III opisanej w rozdziale 4,
- "t4" – żądanie przesłania wskazania czujnika temperatury znajdującego się w panelu solarnym,

- "t5" – żądanie przesłania wskazania czujnika temperatury znajdującego się w zbiorniku wody użytkowej,
- "p1" – żądanie przesłania wskazania czujnika przepływu w układzie zbiornika wody użytkowej.

Żądania wysyłane są za pośrednictwem klasy "ObsługaStanuPolaczenia", w której zaimplementowano 5 obiektów klasy Timer oraz 5 obiektów rozszerzających klasę TimerTask. Obiekt typu Timer pozwala na sekwencyjne wykonywanie zadania określonego w obiekcie rozszerzającym klasę TimerTask z pewnym interwałem czasowym [20, 21]. Każdy z pięciu obiektów klasy Timer ma za zadanie wysłać jedno z wyżej wymienionych żądań z interwałem czasowym równym jednej sekundzie, przy czym jeden z tych obiektów wysyła żądanie "t4", a następnie "t5". Podejście takie związane jest z algorytmem sterowania pompą wymuszającą przepływ wody w panelu solarnym, który wymaga znajomości temperatury wody w panelu solarnym oraz temperatury wody w zbiorniku wody użytkowej. W celu zapobiegnięcia kolizji wysyłanych żądań, kolejne obiekty klasy Timer uruchamiane są z opóźnieniem 200 ms.

W odpowiedzi na odebranie żądania, moduł regulatora kotłowni dokonuje analizy przesłanych danych, a następnie wysyła odpowiedź. Odpowiedź może przyjąć jedną z następujących postaci:

- "\$1@xx.x#" – odpowiedź na żądanie "t1", zawierająca wskazanie czujnika temperatury znajdującego się w pętli regulacyjnej I opisanej w rozdziale 3. Część "xx.x" reprezentuje temperaturę zwróconą przez czujnik temperatury lub w przypadku awarii czujnika może przyjąć wartość "Bład".
- "\$2@xx.x#" – odpowiedź na żądanie "t2", zawierająca wskazanie czujnika temperatury znajdującego się w pętli regulacyjnej II opisanej w rozdziale 3. Część "xx.x" reprezentuje temperaturę zwróconą przez czujnik temperatury lub w przypadku awarii czujnika może przyjąć wartość "Bład".
- "\$3@xx.x#" – odpowiedź na żądanie "t3", zawierająca wskazanie czujnika temperatury znajdującego się w pętli regulacyjnej III opisanej w rozdziale 3. Część "xx.x" reprezentuje temperaturę

zwróconą przez czujnik temperatury lub w przypadku awarii czujnika może przyjąć wartość "Bład".

- "\$4@xx.x#" – odpowiedź na żądanie "t4", zawierająca wskazanie czujnika temperatury znajdującego się w panelu solarnym. Część "xx.x" reprezentuje temperaturę zwróconą przez czujnik temperatury lub w przypadku awarii czujnika może przyjąć wartość "Bład".
- "\$5@xx.x#" – odpowiedź na żądanie "t5", zawierająca wskazanie czujnika temperatury znajdującego się w zbiorniku wody użytkowej. Część "xx.x" reprezentuje temperaturę zwróconą przez czujnik temperatury lub w przypadku awarii czujnika może przyjąć wartość "Bład".
- "\$p@xxxx#" – odpowiedź na żądanie "p1". W przypadku wykrycia przepływu wody przez czujnik przepływu (w układzie zbiornika wody użytkowej), część "xxxx" przyjmuje wartość "1111". W przypadku braku przepływu część "xxxx" przyjmuje wartość "2222". W razie wystąpienia awarii czujnika przepływu część "xxxx" przyjmuje wartość "Bład".

Odebrane przez sterownik urządzeń pomiarowych komendy odpowiedzi są analizowane pod kątem rozpoznania, na które z żądań zawierają odpowiedź oraz czy niosą ze sobą wymagane dane czy informację o błędzie wynikającym z awarii poszczególnych czujników. Gdy dane zostają sklasyfikowane jako poprawne, zostają przekazane do poszczególnych algorytmów sterowania. W przypadku wykrycia awarii, stosowna informacja zostaje wyświetlona w graficznym interfejsie użytkownika.

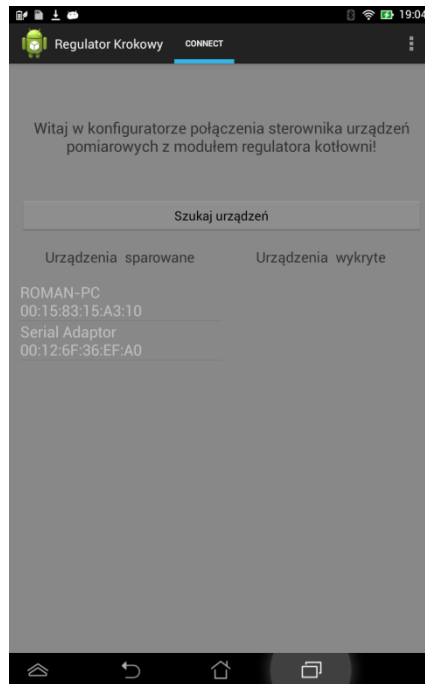
5.4 Graficzny interfejs użytkownika

Jednym z głównych założeń projektowania aplikacji na urządzenia z systemem operacyjnym Android jest oddzielenie graficznego interfejsu użytkownika od części kodu odpowiedzialnej za zachowanie aplikacji. Podejście takie pozwala na łatwą modyfikację interfejsu graficznego bez ingerencji w kod

źródłowy odpowiedzialny za logikę aplikacji. Podział aplikacji na część graficzną oraz część logiczną realizowana jest przez zastosowanie języka XML, za pomocą którego deklarowane są elementy graficzne składające się na interfejs graficzny (np. przyciski, wyświetlacze itp.) [22]. Język XML pozwala na łatwiejszą wizualizację struktury interfejsu graficznego, co prowadzi do przyspieszenia procesu poszukiwania ewentualnych błędów.

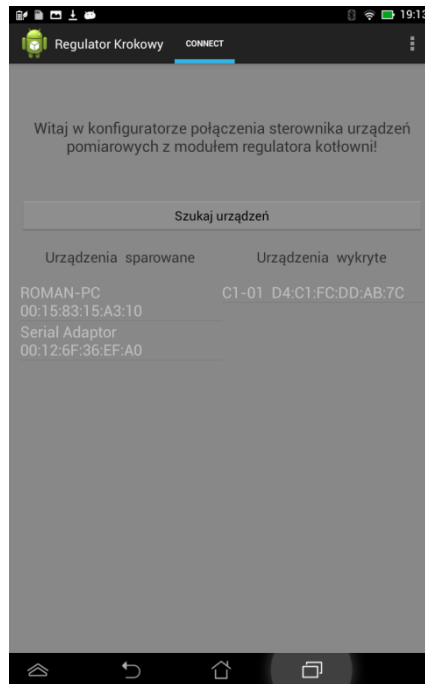
W aplikacji sterownika urządzeń pomiarowych stworzono 4 rodzaje paneli graficznych składających się na interfejs graficzny. Panele graficzne pełnią rolę pośrednika pomiędzy, użytkownikiem, a częścią aplikacji odpowiedzialnej za logikę. W części logicznej, każdy z paneli graficznych zostaje przypisany do specjalnie utworzonej klasy, w której zaimplementowano metody obsługujące zdarzenia wynikające z interakcji użytkownika z panelem graficznym.

Pierwszym panelem graficznym, wyświetlany po załączeniu aplikacji, jest panel "panel_connect.xml". Opisany panel graficzny przypisany został do klasy "ConnectPanel", w której zaimplementowano metody obsługujące zdarzenia wynikające z interakcji użytkownika z panelem graficznym. Rysunek 5.2. przedstawia zrzut ekranu panelu "panel_connect.xml". W panelu tym użytkownik ma możliwość wybrania urządzenia zewnętrznego, z którym chce się połączyć, spośród listy urządzeń sparowanych oraz urządzeń wykrytych przez moduł Bluetooth będący częścią tabletu, na którym zaimplementowano sterownik. Lista urządzeń sparowanych zawiera urządzenia, z którymi sterownik urządzeń pomiarowych nawiązał kontakt w przeszłości, natomiast lista urządzeń wykrytych zawiera nowe urządzenia, z którymi sterownik urządzeń pomiarowych nie nawiązał wcześniej kontaktu. Lista urządzeń wykrytych wypełnia się w przypadku naciśnięcia przez użytkownika przycisku "Szukaj urządzeń" oraz występowania w zasięgu modułu Bluetooth nowych urządzeń.



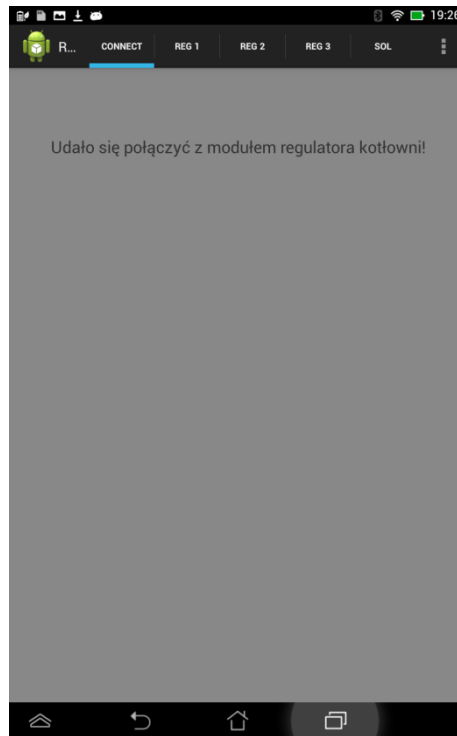
Rysunek 5.2. Zrzut ekranu panelu graficznego "panel_connect.xml".

Rysunek 5.3. przedstawia zrzut ekranu panelu "panel_connect.xml" wyświetlanego po naciśnięciu przycisku "Szukaj urządzeń". Można zauważyć, iż w zasięgu modułu Bluetooth znalazło się urządzenie, z którym sterownik urządzeń pomiarowych wcześniej nie nawiązał komunikacji.



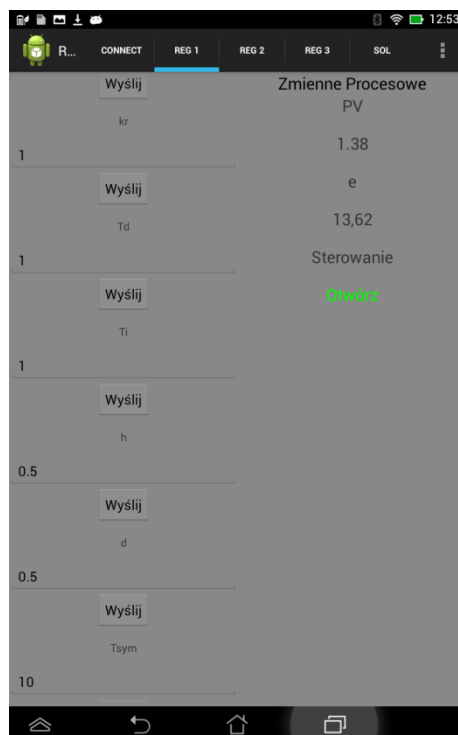
Rysunek 5.3. Zrzut ekranu panelu graficznego "panel_connect.xml" w przypadku znalezienia nowych urządzeń.

Po wybraniu przez użytkownika urządzenia, z którym ma połączyć się sterownik urządzeń pomiarowych panel graficzny "panel_connect.xml" zmienia swoją postać. Rysunek 5.4. przedstawia zrzut ekranu panelu po połączeniu z urządzeniem zewnętrznym. W panelu graficznym pojawił się napis informujący użytkownika o tym, iż udało się nawiązać połączenie. Ponadto w górnej części panelu pojawiły się dodatkowe zakładki.



Rysunek 5.4. Zrzut ekranu panelu graficznego "panel_connect.xml" po połączeniu z urządzeniem zewnętrznym.

Wybierając zakładkę "Reg 1", "Reg 2" lub "Reg 3" zostaje uruchomiony jeden z trzech paneli graficznych tego samego typu. W zależności od wersji aplikacji są to panele typu "panel_pid.xml" lub "panel_krokowy.xml". Rysunek 5.5. przedstawia zrzut ekranu panelu graficznego "panel_pid.xml", a Rysunek 5.6. zrzut panelu "panel_krokowy.xml". Panele zostały odpowiednio przypisane do klas "PanelRegulatorPID" oraz "PanelRegulatorKrokowy", które implementują metody obsługujące zdarzenia wynikające z interakcji użytkownika z panelemi graficznymi. W zakładkach "Reg 1", "Reg 2" oraz "Reg 3" użytkownik ma możliwość zmiany parametrów regulatorów występujących odpowiednio w pętlach regulacyjnych I, II, oraz III opisanych w rozdziale 3. W zależności od wersji aplikacji są to regulatory krokowe lub regulatory implementujące algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym [23].

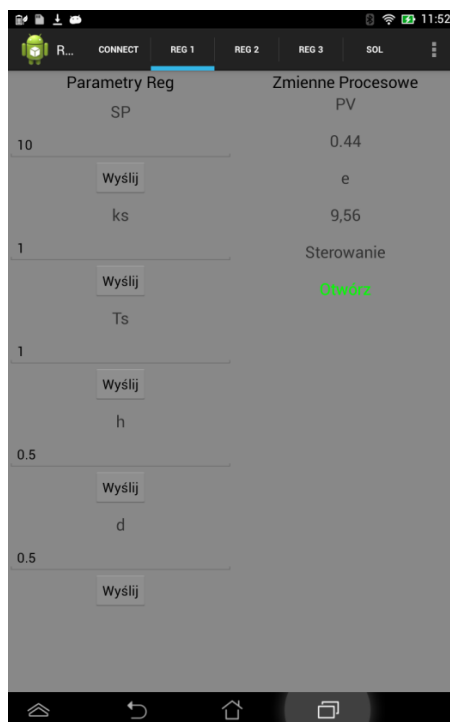


Rysunek 5.5. Zrzut ekranu panelu graficznego "panel_pid.xml".

Panel graficzny "panel_pid.xml" pozwala użytkownikowi na zmianę parametrów regulatora, implementującego algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym (algorytm wraz z jego parametrami został opisany w rozdziale poświęconym podstawom teoretycznym), poprzez wpisanie ich wartości w odpowiednie pola tekstowe, a następnie przyciśnięcie przycisku "Wyślij". Aktualne wartości parametrów regulatora wyświetlane są na bieżąco, w tych samych polach tekstowych, do których można wpisać ich nowe wartości. Ponadto, panel ten pozwala użytkownikowi na zmianę punktu pracy regulatora oraz na podgląd zmiennych procesowych, takich jak aktualna wartość wielkości regulowanej (pole "PV"), czy wartość uchybu regulacji (pole "e"). Dodatkowo, zaimplementowano pole "Sterowanie", które umożliwia podgląd aktualnego sterowania podawanego na urządzenie wykonawcze w postaci elektrozaworu. Lista parametrów regulatora jest na tyle długa, że nie mieści się w obrębie wyświetlacza. Aby rozwiązać ten problem zastosowano listę przewijaną, która poprzez gest przesunięcia palcem po ekranie w obrębie listy pozwala na podgląd parametrów, które nie są widoczne. Ważną funkcjonalnością omawianego panelu

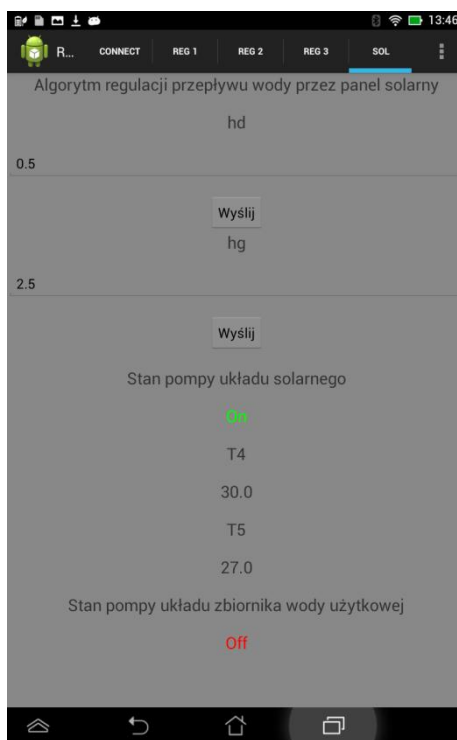
graficznego jest to, iż po wpisaniu nowych wartości parametrów regulatora zostają one zapisane w pamięci urządzenia, a następnie w przypadku ponownego uruchomienia aplikacji zostają automatycznie wyświetlone w odpowiadających im polach tekstowych.

Panel graficzny "panel_krokowy.xml" pozwala użytkownikowi na zmianę parametrów regulatora krokowego (regulator krokowy wraz z jego parametrami został opisany w rozdziale poświęconym podstawom teoretycznym), poprzez wpisanie ich wartości w odpowiednie pola tekstowe, a następnie przyciśnięcie przycisku "Wyślij". Aktualne wartości parametrów regulatora wyświetlane są na bieżąco, w tych samych polach tekstowych, do których można wpisać ich nowe wartości. Ponadto, panel ten pozwala użytkownikowi na zmianę punktu pracy regulatora oraz na podgląd zmiennych procesowych, takich jak aktualna wartość wielkości regulowanej (pole "PV"), czy wartość uchybu regulacji (pole "e"). Dodatkowo zaimplementowano pole "Sterowanie", które umożliwia podgląd aktualnego sterowania podawanego na urządzenie wykonawcze w postaci elektrozaworu. Podobnie jak w przypadku panelu "panel_pid.xml", wpisane parametry zostają zapamiętane w pamięci urządzenia i w przypadku ponownego uruchomienia aplikacji wyświetlane w odpowiadających im polach tekstowych.



Rysunek 5.6. Zrzut ekranu panelu graficznego "panel_krokowy.xml".

Ostatnią z zakładek występujących w graficznym interfejsie użytkownika jest zakładka "Sol". Wybranie tej zakładki powoduje wyświetlenie panelu graficznego "panel_solar_zbiornik.xml", który został przypisany do klasy "PanelRegulatorSolarZbiornik". Klasa ta implementuje metody służące do obsługi zdarzeń wynikających z interakcji użytkownika z panelem graficznym. Rysunek 5.7. przedstawia zrzut ekranu panelu graficznego "panel_solar_zbiornik.xml". Panel ten odpowiedzialny jest za zmianę parametrów algorytmu sterowania przepływem wody przez panel solarny. Ponadto, w panelu tym zostaje wyświetlona temperatura wody na zbiorniku wody użytkowej oraz temperatura wody w panelu solarnym. Dodatkowo, panel wyświetla informację na temat sterowania pracą pomp odgrywających rolę urządzeń wykonawczych w pętlach regulacyjnych **IV** oraz **V** opisanych w rozdziale 3.



Rysunek 5.7. Zrzut ekranu panelu graficznego "panel_solar_zbiornik.xml".

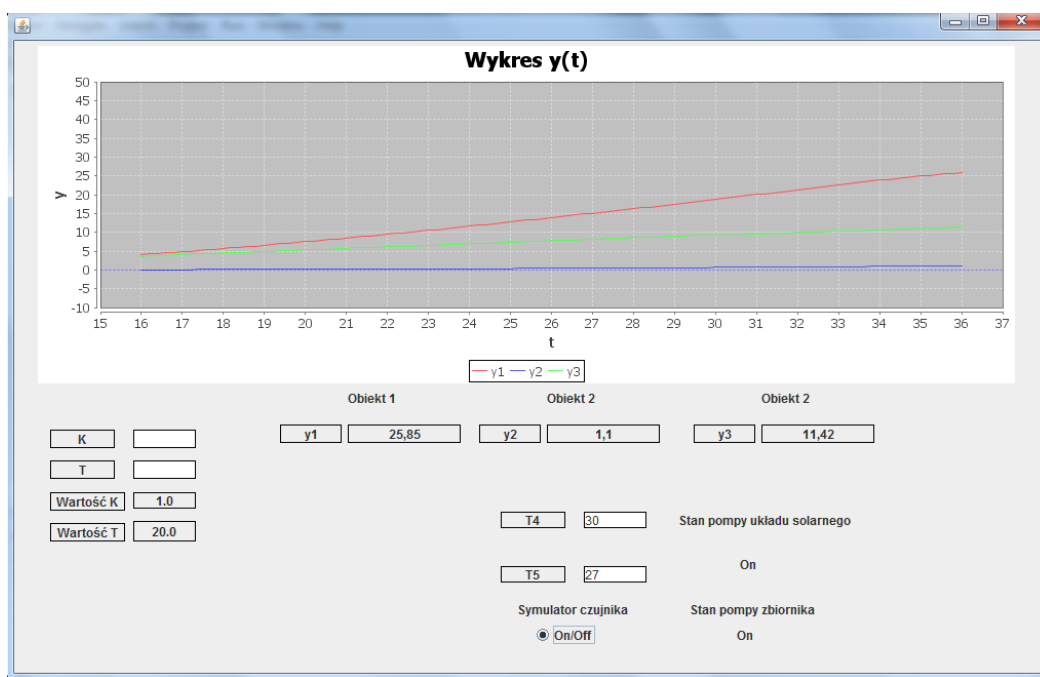
Wyświetlacz "T4" wyświetla temperaturę wody w panelu solarnym, natomiast wyświetlacz "T5" odpowiedzialny jest za wyświetlanie temperatury wody

znajdującej się w zbiorniku wody użytkowej. Na podstawie różnicy tych temperatur oraz histerezy (której dolną granicę określa wartość wpisana w pole tekstowe "hd", a granice górną wartość wpisana w pole tekstowe "hg") zostaje regulowana praca pompy C.O. znajdująca się w układzie panelu solarnego. Sterowanie pompą wyświetlane jest w wyświetlaczu "Stan pompy układu solarnego" i może przyjąć dwie wartości, a mianowicie wartość "On" (odpowiadającą załączeniu pompy) oraz wartość "Off" (odpowiadającą wyłączeniu pompy). Wyświetlacz "Stan pompy układu zbiornika wody użytkowej" informuje o stanie pracy pompy znajdującej się w układzie zbiornika wody użytkowej. Podobnie jak w przypadku wyświetlacza stanu pracy pompy układu solarnego, może przyjąć tylko dwie wartości. Wartości dolnej oraz górnej granicy histerezy zostają zapisane w pamięci urządzenia i w razie ponownego uruchomienia aplikacji wyświetlone są w odpowiadających im polach tekstowych.

6. Symulator obiektu rzeczywistego

W celu przetestowania działania sterownika urządzeń pomiarowych zaimplementowano aplikację, której celem jest symulacja obiektu rzeczywistego. Podejście takie wynika z braku dostępu do obiektu rzeczywistego. Symulator został napisany w języku obiektowym Java i znajduje się na komputerze PC. Rysunek 6.1. przedstawia zrzut ekranu symulatora obiektu rzeczywistego. Połączenie symulatora obiektu rzeczywistego ze sterownikiem urządzeń pomiarowych zrealizowano za pomocą adaptera Bluetooth "Bluetooth USB Dongle". Jest to urządzenie, które po wpięciu do portu USB komputera PC tworzy w systemie operacyjnym wirtualny port szeregowy COM umożliwiając programowe wysyłanie oraz odbieranie danych za pośrednictwem interfejsu Bluetooth. Użycie adaptera Bluetooth jest znacznie wygodniejsze, niż użycie modułu AVT 1635, który wymaga konwertera danych RS-232/USB oraz dodatkowego kabla USB. Kwestia przesyłania oraz odbierania danych w przypadku zastosowania adaptera "Bluetooth USB Dongle" oraz układu AVT 1635 w połączeniu z konwerterem RS-232/USB wygląda identycznie. Powyższe cechy adaptera Bluetooth zdecydowały o jego wykorzystaniu w procesie komunikacji sterownika urządzeń pomiarowych z symulatorem obiektu rzeczywistego.

Realizacja komunikacji od strony programowej została oparta o bibliotekę "RXTX JAVA Communication API" [24]. Jest to biblioteka odpowiedzialna za obsługę szeregowego portu COM.



Rysunek 6.1. Zrzut ekranu symulatora obiektu rzeczywistego.

Po uruchomieniu symulatora obiektu rzeczywistego, wyświetlana jest aplikacja okienkowa, jak na Rysunku 6.1. Jest ona podzielona na dwie części. W pierwszej z nich znajduje się wykres, odpowiedzi trzech symulowanych obiektów dynamicznych, które zastępują obiekty rzeczywiste z pętli I, II oraz III opisanych w rozdziale 3. Wybrane obiekty zostaną omówione w rozdziale poświęconym testom sterownika urządzeń pomiarowych. Druga część panelu symulatora zawiera dane dotyczące symulacji w formie liczbowej. Po lewej stronie znajdują się pola tekstowe "K" oraz "T" służące do wpisywania parametrów na podstawie, których symulowane są obiekty dynamiczne. Poniżej nich, znajdują się elementy wyświetlające aktualne wartości tych parametrów. Bezpośrednio pod wykresem wyświetlane są aktualne wartości wielkości regulowanych symulowanych obiektów dynamicznych. Pole tekstowe oznaczone jako "T4" służy do wpisywania wskazania czujnika temperatury wody znajdującej się w panelu

solarnym, znów pole tekstowe "T5" umożliwia wpisanie wskazania czujnika temperatury wody w zbiorniku wody użytkowej. Przycisk oznaczony jako "Symulator czujnika" odpowiada wskazaniu czujnika przepływu wody znajdującego się w układzie zbiornika wody użytkowej. Elementy oznaczone jako "Stan pompy układu solarnego" oraz "Stan pompy zbiornika" mają za zadanie wyświetlać aktualny stan pracy pompy regulującej przepływ wody przez panel solarny oraz pompy znajdującej się w układzie zbiornika wody użytkowej.

7. Podstawy teoretyczne

Rozdział ten poświęcony jest przybliżeniu zagadnień teoretycznych pozwalających na komputerową symulację obiektów dynamicznych. Dotyczy także, teorii regulacji układów hydraulicznych oraz doboru nastaw regulatorów.

7.1 Symulacja komputerowa obiektów dynamicznych

Aby dokonać symulacji komputerowej obiektów dynamicznych opisanych transmitancją ciągłą, należy dokonać przekształceń w efekcie, których powstaną równania pozwalające na numeryczne wyznaczanie odpowiedzi obiektów dynamicznych w aktualnej chwili czasu. W tym celu można zastosować dwa podejścia. Jedno z nich polega na zamianie transmitancji ciągłej na transmitancję dyskretną wykorzystując przekształcenie „Z”, a następnie przekształceniu otrzymanej transmitancji dyskretniej na równanie różnicowe. Drugie z podejść zakłada przekształcenie transmitancji ciągłej na równanie różniczkowe, a następnie rozwiązanie równania jedną z metod numerycznych dających przybliżone rozwiązanie równania różniczkowego. Rozdział 7.2 został poświęcony numerycznej metodzie Eulera w wyniku, której otrzymuje się przybliżone rozwiązanie równania różniczkowego. Rozdział 7.3 odnosi się do metody wykorzystującej przekształcenie „Z”. Stosując jedną z powyższych metod otrzymuje się równanie umożliwiające wyznaczanie odpowiedzi obiektu dynamicznego w danej chwili czasu na podstawie aktualnego sterowania oraz poprzednich odpowiedzi obiektu. Umożliwia to symulowanie obiektu dynamicznego w czasie rzeczywistym. Wystarczy napisać aplikację, która posłuży

do obliczania aktualnej odpowiedzi obiektu dynamicznego, co pewien znany interwał czasowy.

7.2 Metoda Eulera

Metoda Eulera pozwala na przybliżone rozwiązywanie równań różniczkowych zwyczajnych. W niniejszej pracy rozważa się dwa przypadki równań różniczkowych zwyczajnych, a mianowicie równanie różniczkowe zwyczajne pierwszego oraz drugiego rzędu.

Równanie różniczkowe zwyczajne pierwszego rzędu opisane jest następującą zależnością [25]:

$$F(x, y, y') = 0 \quad (1)$$

Z kolei równanie różniczkowe zwyczajne drugiego rzędu opisane jest zależnością [25]:

$$F(x, y, y', y'') = 0 \quad (2)$$

Za pomocą metody Eulera zostały zasymulowane obiekty o następujących transmitancjach ciągłych:

$$K(s) = \frac{k}{(1 + sT_1)(1 + sT_2)} \quad (3)$$

oraz

$$K(s) = \frac{k\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (4)$$

Obiekt opisany równaniem (3) jest obiektem typu inercja drugiego rzędu i odpowiada obiektowi regulacji w pętli regulacji II, natomiast obiekt opisany równaniem (4) odpowiada obiektowi w pętli regulacji III i jest obiektem o charakterze oscylacyjnym [26]. Pętle regulacji zostały opisane w rozdziale 3.

Przekształcając równanie (3) oraz (4) oraz stosując odwrotną transformatę Laplace'a otrzymano następujące równania:

$$T_1 T_2 \frac{d^2 y(t)}{dt^2} + (T_1 + T_2) \frac{dy(t)}{dt} + y(t) - ku(t) = 0 \quad (5)$$

$$\frac{d^2 y(t)}{dt^2} + 2\xi\omega_n \frac{dy(t)}{dt} + \omega_n^2 y(t) - k\omega_n^2 u(t) = 0 \quad (6)$$

Równania (5) oraz (6) przyjmują taką samą formę, jak równanie (2), a więc są to równania różniczkowe zwyczajne drugiego rzędu. Pozwala to na zastosowanie metody Eulera w celu znalezienia ich przybliżonego rozwiązania, które pozwoli na zasymulowanie tych obiektów.

Na podstawie opracowania wg J. Klamka i wsp. [26] dla równania różniczkowego zwyczajnego pierwszego rzędu opisanego równaniem (7), można sformułować równania (8) oraz (9). Parametr h można rozumieć jako okres próbkowania.

$$y^{(1)}(x) = f(x, y(x)) \quad (7)$$

$$\frac{y_{i+1} - y_i}{h} = y^{(1)}(x) \quad (8)$$

$$y_{i+1} = y_i + \Delta y_i \quad (9)$$

$$\Delta y_i = hf(x_i, y_i) \quad (10)$$

Równania (8), (9) oraz (10) stanowią istotę metody Eulera. Dzięki tym równaniom można zastąpić krzywą całkową $y(x)$ łamaną składającą się z odcinków prostych tworzonych przez punkty y_i oraz y_{i+1} [26]. Wymaga to jednak znajomości warunków początkowych. Ponadto, stosując metodę Eulera należy posiadać świadomość, iż dobór parametru h ma znaczenie na działanie metody. Dobór zbyt dużego parametru h powoduje wzrost błędu metody, znów dobór zbyt małego parametru h przyczyni się do wzrost błędu zaokrągleń [26].

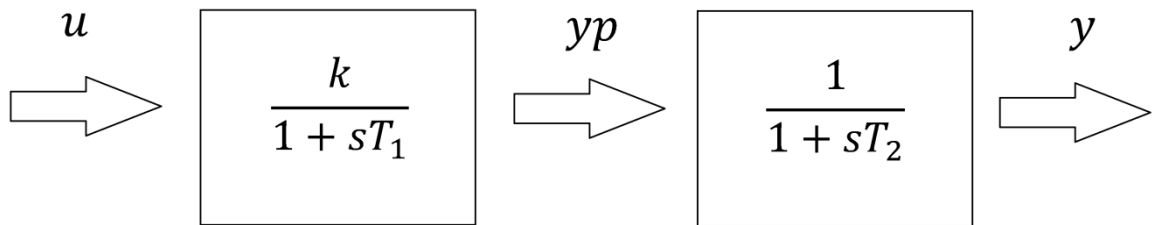
Wykorzystując równania (8), (9) oraz (10) oraz rozpatrując równanie (3) jako szeregowe połączenie dwóch członów w postaci inercji pierwszego rzędu przedstawione na Rysunku 7.1., można wyprowadzić następujące równania:

$$T_1 \frac{dyp(t)}{dt} + yp(t) = ku(t) \quad (11)$$

$$yp_{i+1} = yp_i + \frac{h}{T_1} (ku_i - yp_i) \quad (12)$$

$$T_2 \frac{dy(t)}{dt} + y(t) = yp(t) \quad (13)$$

$$y_{i+1} = y_i + \frac{h}{T_2} (yp_i - y_i) \quad (14)$$



Rysunek 7.1. Schemat blokowy obiektu opisanego równaniem (3).

Równanie (12) oraz (14) pozwalają na komputerową symulację obiektu dynamicznego opisanego równaniem (3). W aplikacji symulatora równania (12) oraz (14) są obliczane z okresem próbkowania h równym 100 ms.

W przypadku równania (6), które opisuje obiekt dynamiczny wyrażony transmitancją ciągłą (4), zastosowanie metody Eulera prowadzi do następujących równań:

$$\frac{d^2y(t)}{dt^2} = \frac{\frac{y_{i+2} - y_{i+1}}{h} - \frac{y_{i+1} - y_i}{h}}{h} \quad (15)$$

$$\frac{dy(t)}{dt} = \frac{y_{i+1} - y_i}{h} \quad (16)$$

$$y(t) = y_i \quad (17)$$

$$u(t) = u_i \quad (18)$$

Podstawiając równania (15), (16), (17) oraz (18) do równania (6) oraz dokonując przekształceń otrzymano następujące równanie:

$$y_{i+2} = (2 - 2\xi\omega_n h)y_{i+1} + (2\xi\omega_n h - \omega_n^2 h^2 - 1)y_i + kh^2\omega_n^2 u_i \quad (19)$$

Równanie (19) pozwala na komputerową symulację obiektu dynamicznego opisanego równaniem (4). W aplikacji symulatora równanie (19) jest obliczane z okresem próbkowania h równym 100 ms.

Metoda Eulera pozwala również na symulację dynamiki siłowników oraz zaworów występujących w pętlach regulacji opisanych w rozdziale 3. Dynamika siłowników opisana została następującym równaniem [27]:

$$K(s) = \frac{1}{T_s s} \quad (20)$$

Czas T_s jest równy czasowi niezbędnemu do zmiany położenia siłownika z pozycji otwarcia 0% do pozycji otwarcia 100%. Przekształcając równanie (20) oraz stosując odwrotną transformatę Laplace'a otrzymano następujące równanie:

$$\frac{dy(t)}{dt} = \frac{1}{T_s} u(t) \quad (21)$$

Równanie (21) przyjmuje taką samą formę jak równanie (1), a więc jest to równanie różniczkowe zwyczajne pierwszego rzędu. Pozwala to na zastosowanie metody Eulera w celu znalezienia przybliżonego rozwiązania równania (21) pozwalającego na zasymulowanie obiektu opisanego tym równaniem. Wykorzystując równania (8), (9) oraz (10) można wyprowadzić następujące równanie:

$$y_{i+1} = y_i + \frac{h}{T_s} u_i \quad (22)$$

Równanie (22) pozwala na komputerową symulację obiektu dynamicznego opisanego równaniem (20).

Dynamika zaworu została opisana następującym równaniem [27]:

$$K(s) = \frac{k_z}{1 + T_z s} \quad (23)$$

Przekształcając równanie (23) oraz stosując odwrotną transformatę Laplace'a otrzymano następujące równanie:

$$T_z \frac{dy(t)}{dt} = k_z u(t) - y(t) \quad (24)$$

Równanie (24) przyjmuje taką samą formę jak równanie (1), a więc jest to równanie różniczkowe zwyczajne pierwszego rzędu. Pozwala to na zastosowanie metody Eulera w celu znalezienia przybliżonego rozwiązania równania (24), które pozwoli na zasymulowanie obiektu opisanego tym równaniem. Wykorzystując równania (8), (9) oraz (10) można wyprowadzić następujące równanie:

$$y_{i+1} = y_i + (k_z u_i - y_i) \frac{h}{T_z} \quad (25)$$

Równanie (25) pozwala na komputerową symulację obiektu dynamicznego opisanego równaniem (23).

7.3 Metoda wykorzystująca transformatę "Z"

Opisywana metoda wykorzystuje transformatę "Z" (transformata Laurenta). Transformatą "Z" nazywamy przekształcenie dyskretnej funkcji czasu $Z[f(nT)] = F(z)$ [17]. Transformata "Z" opisana jest następującym równaniem [28]:

$$F(z) = \sum_{n=0}^{\infty} f(nT) z^{-n} \quad (26)$$

Za pomocą metody wykorzystującej transformatę "Z" zasymulowano obiekt opisany następującą transmitancją ciągłą:

$$K(s) = \frac{k}{1 + sT} e^{-sT_0} \quad (27)$$

Dodatkowo zakłada się, iż czas opóźnienia T_0 jest całkowitą wielokrotnością czasu próbkowania h .

$$T_0 = nh, \text{ gdzie } n \in N \quad (28)$$

Obiekt opisany równaniem (27) odpowiada obiektowi regulacji w pętli I opisanej w rozdziale 3.

Wykorzystując tablicę transformat Laplace'a oraz "Z" [29], a także twierdzenie o przesunięciu w dziedzinie czasu [28] można przekształcić transmitancję ciągłą opisaną równaniem (27) na następującą transmitancję dyskretną:

$$K(z) = \frac{k}{T} \frac{z}{z - e^{-\frac{h}{T}}} z^{-n} \quad (29)$$

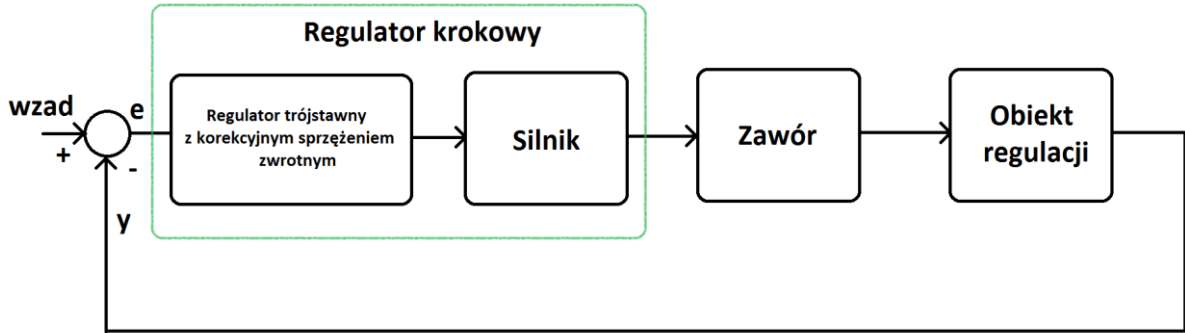
Transmitancję dyskretną opisaną równaniem (29) można przekształcić w równanie różnicowe opisane poniższym równaniem:

$$y_i = e^{-\frac{h}{T}} y_{i-1} + \frac{k}{T} u_{i-n} \quad (30)$$

Równanie (30) pozwala na komputerową symulację obiektu dynamicznego opisanego równaniem (27). W aplikacji symulatora równanie (30) jest obliczane z okresem próbkowania h równym 100 ms.

7.4 Regulator krokowy

Regulator krokowy swoje najczęstsze zastosowanie znajduje w układach, których schemat ideowy przedstawia Rysunek 7.2. Regulator krokowy na podstawie uchybu regulacji oblicza sterowanie, które podawane jest na urządzenie wykonawcze w postaci zaworu. Zawór bezpośrednio oddziałuje na obiekt regulacji, czyniąc go tym samym sterowalnym. Schemat układu sterowania z Rysunku 7.2. odpowiada pętlom regulacji I, II oraz III opisanym w rozdziale 3.



Rysunek 7.2. Schemat układu sterowania z regulatorem krokowym [30].

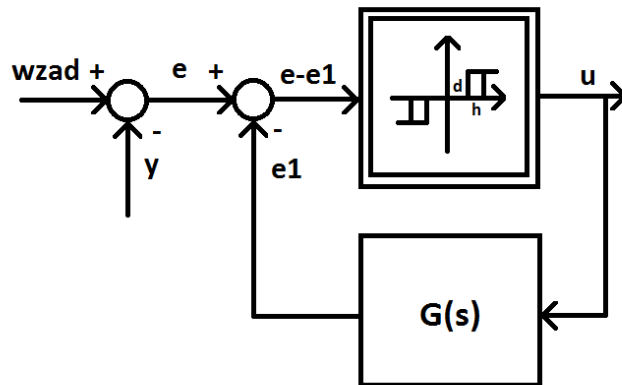
Regulator krokowy składa się z regulatora trójstawnego objętego korekcyjnym sprzężeniem zwrotnym oraz silnika (najczęściej elektrycznego) [30]. Schemat ideowy regulatora trójstawnego objętego korekcyjnym sprzężeniem zwrotnym przedstawia Rysunek 7.3. Jeśli korekcyjne sprzężenie zwrotne przyjmie postać $G(s) = \frac{k_s}{1+T_s}$ (inercja pierwszego rzędu) oraz element trójstawny w postaci przekaznika trójpołożeniowego potraktuje się jako element o bardzo dużym wzmocnieniu k_r , to transmitancja regulatora trójstawnego z korekcyjnym sprzężeniem zwrotnym, będzie wyrażona następującym równaniem [27]:

$$G_r(s) = \frac{1}{\frac{1}{k_r} + \frac{k_s}{1+T_s}} \quad (31)$$

Przyjmując $k_r \rightarrow \infty$ (możliwe tylko wtedy, gdy strefa histerezy przekaznika $h \rightarrow 0$) transmitancja $G_r(s)$ przyjmuje następującą postać [27]:

$$G_r(s) = \frac{1}{k_s} (1 + sT_s) \quad (32)$$

Regulator trójstawny z korekcyjnym sprzężeniem zwrotnym w postaci $G(s) = \frac{k_s}{1+T_s}$ można, więc traktować jako regulator typu PD.

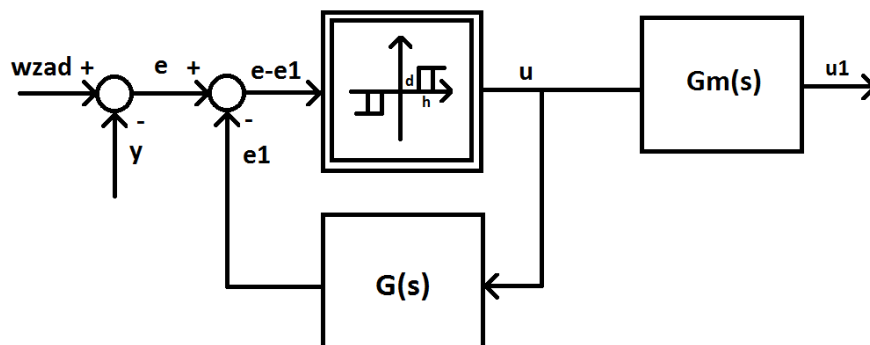


Rysunek 7.3. Schemat ideowy regulatora trójstawnego objętego korekcyjnym sprzężeniem zwrotnym [27], [30].

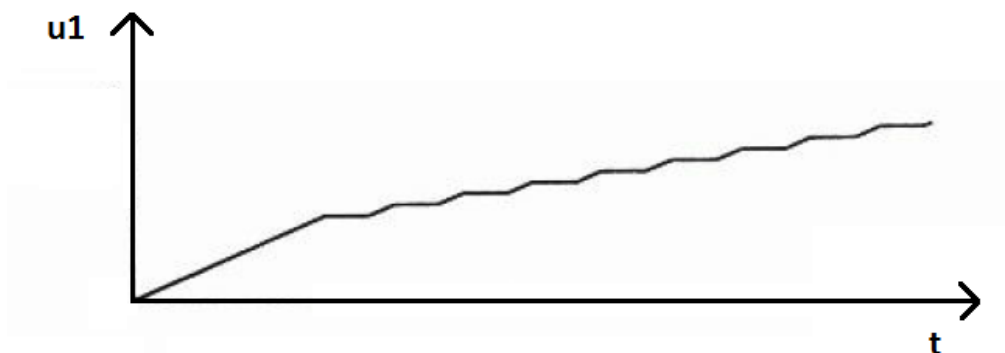
Schemat połączenia szeregowego regulatora trójstawnego objętego korekcyjnym sprzężeniem zwrotnym z silnikiem, które tworzy regulator krokowy przedstawia Rysunek 7.4. Silnik elektryczny modeluje się jako element typu całka, można więc przyjąć, iż opisuje go funkcja przejścia $G_m(s) = \frac{1}{T_m s}$ [31]. Jeśli funkcja przejścia regulatora trójstawnego objętego korekcyjnym sprzężeniem zwrotnym opisana jest równaniem (24) to transmitancja opisująca regulator krokowy przyjmuje postać równania (33) [31].

$$G_{rk}(s) = \frac{T_s}{T_m k_s} \left(1 + \frac{1}{T_s s} \right) \quad (33)$$

Regulator krokowy można, więc traktować jako regulator typu PI. Przykładowy przebieg wyjścia regulatora krokowego przedstawia Rysunek 7.5. Na wykresie można zaobserwować, iż przebieg jest zbliżony do skokowej odpowiedzi regulatora typu PI.



Rysunek 7.4. Schemat regulatora krokowego [29].

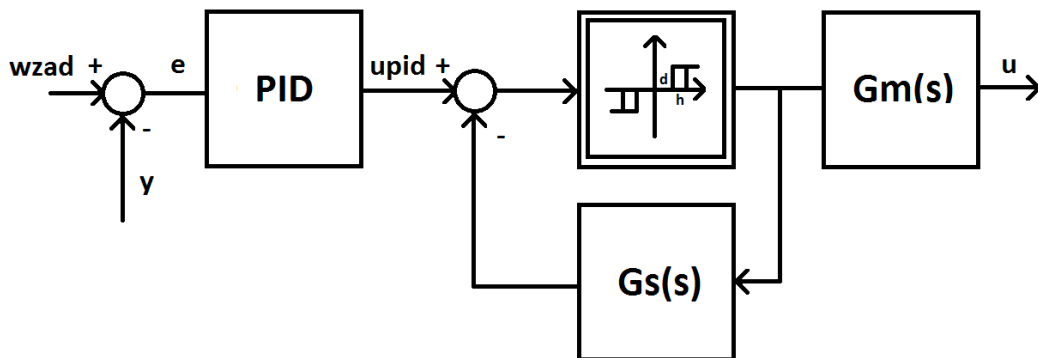


Rysunek 7.5. Przykładowa odpowiedź skokowa regulatora krokowego.

7.5 Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym

Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym stosuje się, gdy urządzenie wykonawcze jest stałoprędkościowym, rewersyjnym siłownikiem elektrycznym oraz nie ma możliwości dokonania pomiaru położenia tego siłownika [30]. Rysunek 7.6 przedstawia schemat ideowy regulatora krokowego z wewnętrznym sprzężeniem zwrotnym. Element $G_m(s)$ reprezentuje siłownik, natomiast element $G_s(s)$ stanowi model siłownika, dzięki któremu

uzyskuje się informację na temat jego położenia. Uchyb regulacji zostaje przekazywany na regulator PID, następnie sygnał wypracowany przez ten regulator jest porównywany z symulowanym położeniem siłownika. Sygnał powstały w wyniku porównania sygnału z regulatora PID z symulowanym położeniem siłownika jest przesyłany na przekaźnik trójpoleżeniowy, którego wyjście jest przekazywane na siłownik sterujący pracą obiektu.



Rysunek 7.6. Schemat ideowy regulatora krokowego z wewnętrznym sprzężeniem zwrotnym [30].

Wadą algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym jest to, iż jego działanie jest uzależnione od modelu siłownika. Czas całkowania modelu powinien być równy czasowi przestawiania siłownika [30]. Nie jest to jednak możliwe, ponieważ nie da się uniknąć drobnych różnic czasowych. Ponadto, występują opóźnienia wynikające z rozruchu oraz zatrzymania siłownika [30]. Dodatkowo siłownik może zostać zatrzymany przez wyłączniki krańcowe, a informacja o tym zdarzeniu nie jest przekazywana do modelu [30]. Wszystko to, może po pewnym czasie prowadzić do błędnego wskazania położenia siłownika przez jego model.

Przyjmując pewne, założenia dotyczące algorytmu można domniemywać, iż będzie on działał poprawnie nawet w dłuższym przedziale czasowym. Jeśli przyjąć, iż model siłownika opisany jest funkcją przejścia $G_s(s) = \frac{1}{T_s s}$ [30],

a siłownik funkcją przejścia $G_m(s) = \frac{1}{T_m s}$ [31] oraz to, iż element trójstawny w postaci przekaźnika trójpołożeniowego można traktować, jak element o bardzo dużym wzmocnieniu k_r ($k_r \rightarrow \infty$, gdy $h \rightarrow 0$), to pozwala to na wyprowadzenie następujących równania:

$$\frac{u_{pid}(s)}{e(s)} = k_p + k_d s + \frac{k_i}{s} \quad (34)$$

$$\frac{u(s)}{u_{pid}(s)} = \frac{\frac{T_s}{T_m}}{\frac{T_s}{k_r} + 1} \quad (35)$$

Przyjmując $k_r \rightarrow \infty$ (możliwe tylko wtedy, gdy strefa histerezy przekaźnika $h \rightarrow 0$) transmitancja $\frac{u(s)}{u_{pid}(s)}$ przyjmuje następującą postać:

$$\frac{u(s)}{u_{pid}(s)} = \frac{T_s}{T_m} \quad (36)$$

Można stwierdzić, iż transmitancja $\frac{u(s)}{u_{pid}(s)}$ jest elementem typu wzmocnienie, którego wartość zależy od wartości czasu całkowania modelu siłownika oraz czasu przestawiania siłownika. Funkcja przejścia dla algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym przyjmuje więc następującą postać:

$$\frac{u(s)}{e(s)} = \frac{T_s}{T_m} \left(k_p + k_d s + \frac{k_i}{s} \right) \quad (37)$$

Równanie (37) pozwala na stwierdzenie, iż przy spełnieniu pewnych założeń, algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym powinien zachowywać się jak regulator typu PID.

7.6 Nastawy regulatorów

Dobór nastaw algorytmów regulacji przedstawionych w rozdziale 7.5 oraz 7.6 przeprowadzony został według metody Zieglera-Nicholsa [32]. Polega ona na umieszczeniu obiektu regulacji w układzie regulacji zamkniętej z regulatorem typu

P, a następnie zwiększaniu wzmocnienia regulatora do momentu, aż układ znajdzie się na granicy stabilności. Następnie, na podstawie wzmocnienia regulatora typu P, dla którego układ znalazł się na granicy stabilności (wzmocnienie krytyczne k_{kr}) oraz okresu drgań krytycznych t_{kr} , wyznacza się parametry regulatora P, PI lub PID. Sposób obliczania parametrów poszczególnych regulatorów na podstawie k_{kr} oraz t_{kr} przedstawia tabela 7.1. Parametry uzyskane na podstawie danych zamieszczonych w tabeli 7.1. odnoszą się to transmitancji regulatora PID wyrażonej następującym równaniem:

$$G_r(s) = k_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (38)$$

Tabela 7.1. Dobór nastaw regulatora P, PI lub PID na podstawie metody Zieglera-Nicholsa [23].

Typ regulatora \ Człon	k_p	T_i	T_d
P	$0,5k_{kr}$	-	-
PI	$0,5k_{kr}$	$0,85t_{kr}$	-
PID	$0,5k_{kr}$	$0,85t_{kr}$	$0,12t_{kr}$

W celu wyznaczenia parametrów regulatora krokowego należało przeliczyć parametry uzyskane za pomocą metody Zieglera-Nicholsa zgodnie z równaniem (33), natomiast, aby wyznaczyć parametry algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym zastosowano równanie (37).

7.7 Jakość regulacji

W celu porównania jakości regulacji automatycznej za pomocą regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym posłużono się następującymi wskaźnikami jakości regulacji automatycznej [28], [32]:

- średni uchyb e_{sr} ,
- przeregulowanie opisane poniższym równaniem:

$$\sigma = \frac{e_{p1}}{e_{p0}} 100\%, \text{ gdzie} \quad (39)$$

e_{p1} – druga amplituda uchybu $e(t)$,

e_{p0} – pierwsza amplituda uchybu $e(t)$,

- czas regulacji t_r , który upłynął od momentu wystąpienia zakłócenia lub zmiany wartości zadanej do ustalenia się wahań uchybu $e(t)$ w przyjętych granicach,
- wskaźniki całkowite wyrażone poniższymi równaniami:

$$I_1 = \int_0^{\infty} |e(t)| dt, \quad (40)$$

$$I_2 = \int_0^{\infty} e^2(t) dt, \quad (41)$$

$$I_3 = \int_0^{\infty} t |e(t)| dt, \quad (42)$$

$$I_4 = \int_0^{\infty} t^2 |e(t)| dt. \quad (43)$$

8. Opis testów

Jednym z ważniejszych etapów realizacji pracy dyplomowej pt. „Sterownik urządzeń pomiarowych zrealizowany na komputerze PDA” był etap testowania sterownika. Testy dostarczyły danych na podstawie, których można wyciągnąć wnioski na temat właściwości sterownika oraz wykorzystanych algorytmów sterowania.

Proces testowania polegał na połączeniu sterownika urządzeń pomiarowych z symulatorem obiektu rzeczywistego, a następnie przebadaniu jakości regulacji w pętlach regulacyjnych I, II oraz III (patrz rozdział 3), za pomocą regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym. Ponadto przetestowano algorytmy regulacji występujące w pętlach regulacyjnych IV oraz V (patrz rozdział 3). Test jakości regulacji w pętlach regulacyjnych I, II oraz III został podzielony na trzy podgrupy, w których dokonano sprawdzenia wpływu zmiany wartości zadanej, wprowadzenia zakłócenia oraz zmiany wartości parametrów sterowanych obiektów na jakość regulacji poszczególnych algorytmów sterowania. Test wpływu zmiany wartości zadanej na jakość regulacji polegał na zmianie wartości zadanej z wartości równej 26 na wartość 40. Następnie po upływie około czterech stałych czasowych (dla wszystkich obiektów przyjęto uproszczenie zakładające, że stała czasowa to 600 sekund) rozpoczynał się test wpływu zakłóceń. Polegał on na dodaniu zakłócenia, które obniżało wartość wyjścia regulowanych obiektów o wartość równą 5. Po upływie czterech stałych czasowych rozpoczynał się test wpływu zmiany parametrów regulowanych obiektów na jakość regulacji. Polegał on na zwiększeniu wzmocnień wszystkich trzech obiektów o wartość 0.5. Po upływie czterech stałych czasowych test wpływu zmiany parametrów regulowanych obiektów dobiegał końca.

W celu wyciągnięcia obiektywnych wniosków na temat jakości procesu regulacji posłużono się wskaźnikami opisanymi w rozdziale 7.7. Dane dotyczące sterowania zostały przez symulator obiektu rzeczywistego zapisane w pliku

tekstowym, a następnie za pomocą programu Matlab zaprezentowane w postaci wykresów.

Testy jakości regulacji w pętlach regulacyjnych I, II oraz III zostały przeprowadzone dla obiektów opisanych w rozdziałach 7.2 oraz 7.3 za pomocą równań (3), (4) oraz (27). Dla obiektu opisanego równaniem (3) przyjęto następujące parametry:

$$k = 0.5$$

$$T_1 = 600 \text{ [s]}$$

$$T_2 = 300 \text{ [s]}$$

Z kolei dla obiektu opisanego równaniem (4) przyjęto następujące parametry:

$$k = 0.5$$

$$\omega_n = 0.0016 \left[\frac{1}{s} \right]$$

$$\xi = 0.5$$

Dla obiektu opisanego równaniem (27) przyjęto następujące parametry:

$$k = 0.5$$

$$T = 600 \text{ [s]}$$

$$T_0 = 30 \text{ [s]}$$

Za pomocą metody Zieglera-Nicholsa opisanej w rozdziale 7.6 oraz programu Matlab-Simulink dokonano doboru nastaw regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym. Tabela 8.1 zawiera nastawy regulatora krokowego, natomiast tabela 8.2 nastawy algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym dla poszczególnych obiektów regulacji. Ponadto, dla algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym przyjęto, iż symulowany czas przestawiania siłownika jest równy rzeczywistemu czasowi przestawiania siłownika i wynosi 10 [s].

Tabela 8.1. Nastawy regulatora krokowego dla poszczególnych obiektów regulacji.

Parametr Obiekt	k_s	T_s
$K(s) = \frac{k}{1+sT}e^{-sT_0}$	64,336	10149 [s]
$K(s) = \frac{k}{(1+sT_1)(1+sT_2)}$	0,6248	5635,5 [s]
$K(s) = \frac{k\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$	1,593	9953,5 [s]

Tabela 8.2. Nastawy algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym dla poszczególnych obiektów regulacji.

Parametr Obiekt	k_p	T_i	T_d
$K(s) = \frac{k}{1+sT}e^{-sT_0}$	15,78	10149 [s]	1432,8 [s]
$K(s) = \frac{k}{(1+sT_1)(1+sT_2)}$	902	5635,5 [s]	795,6 [s]
$K(s) = \frac{k\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$	625	9953,5 [s]	1405,2 [s]

W przypadku obu algorytmów sterowania przyjęto, iż sterfa martwa przekaźnika trójpołożeniowego d jest równa 0,2, natomiast histereza h jest równa 0,1. Ponadto, założono następujące parametry zaworu opisanego równaniem (23):

$$k_z = 2$$

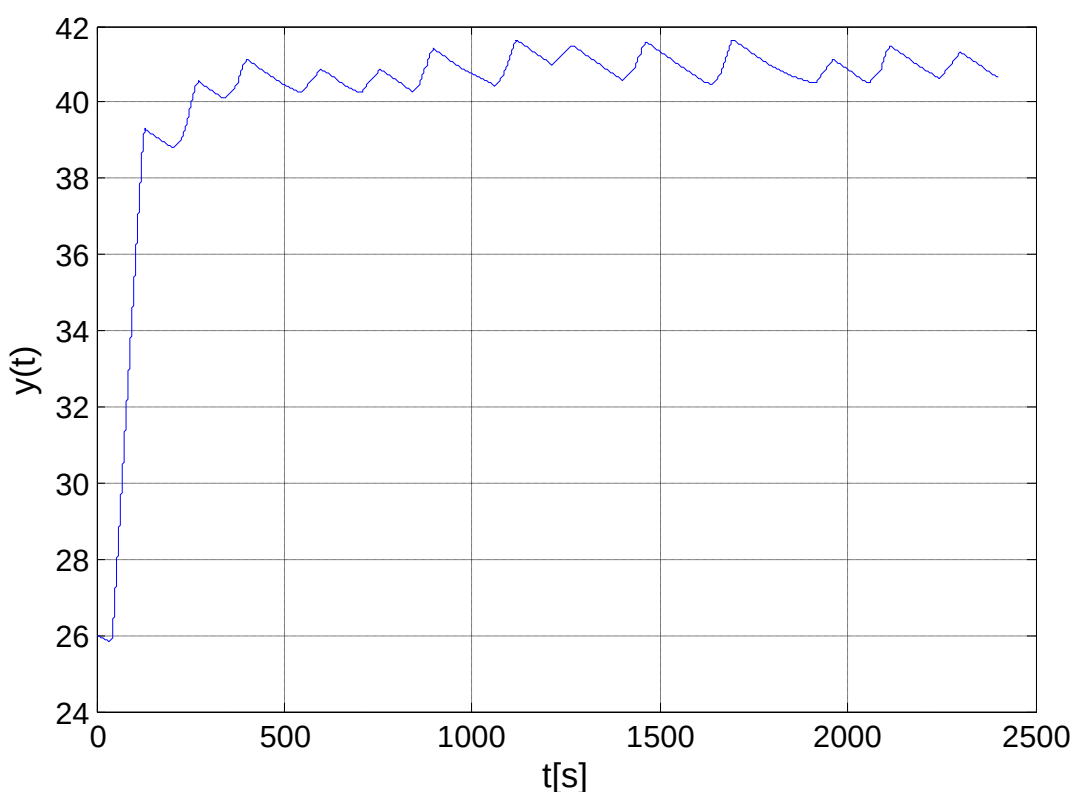
$$T_z = 0.5 \text{ [s]}$$

8.1 Testy jakości regulacji za pomocą regulatora krokowego

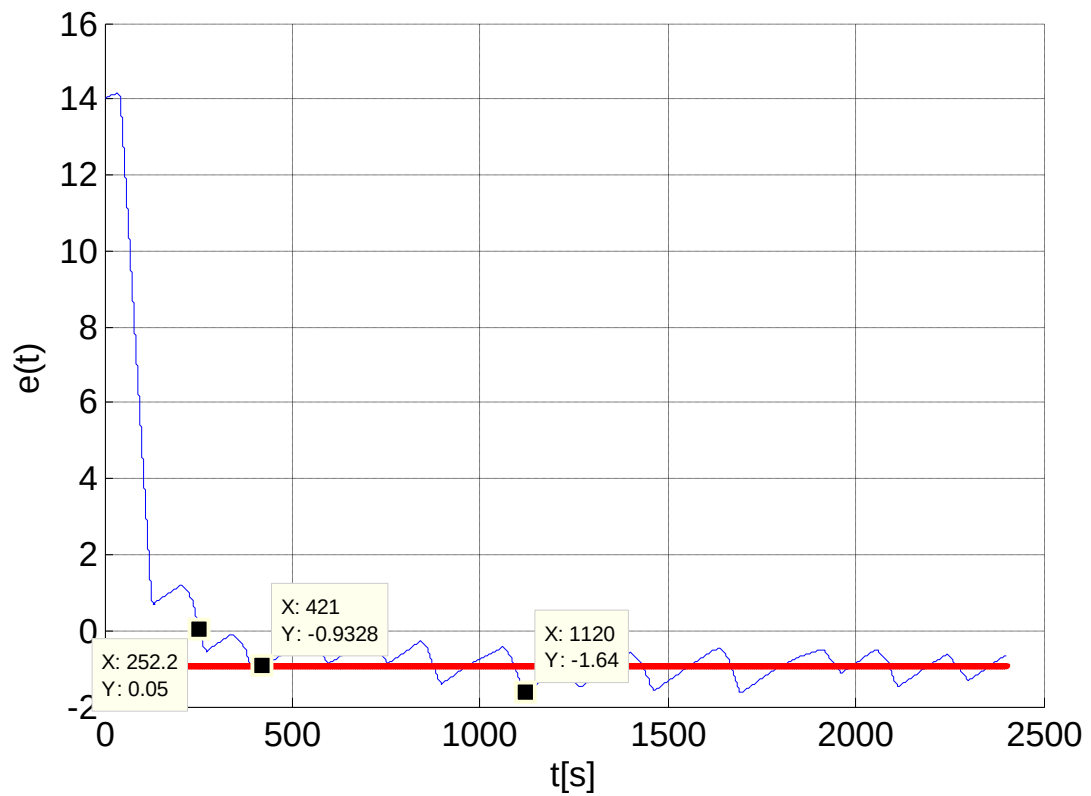
Niniejszy rozdział zawiera dane, zebrane podczas testów jakości regulacji za pomocą regulatora krokowego, przedstawione w formie wykresów oraz tabel zawierających wskaźniki jakości opisane w rozdziale 7.7. Poniżej zamieszczono wyniki uzyskane dla poszczególnych obiektów regulacji.

Obiekt typu inercja z opóźnieniem ($K(s) = \frac{k}{1+sT} e^{-sT_0}$)

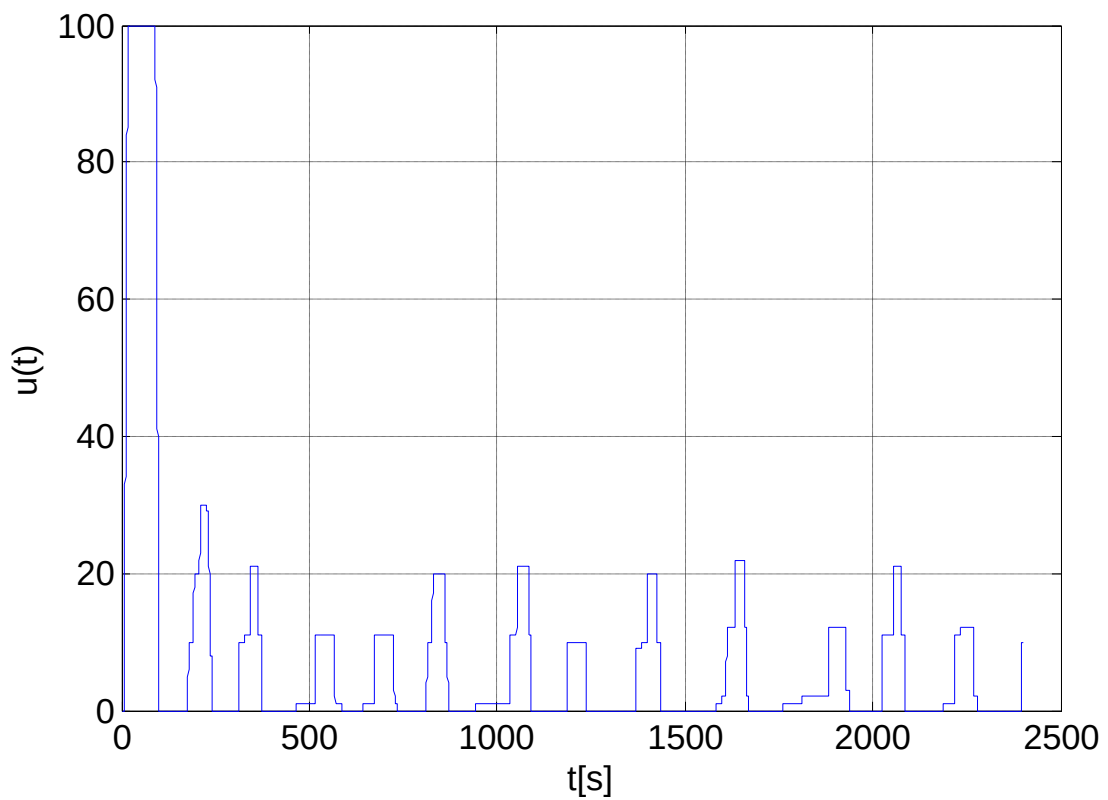
Rysunek 8.1 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.2 przebieg uchybu w procesie regulacji wynikającym ze zmiany wartości zadanej. Tabela 8.3 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.3 prezentuje przebieg wyjścia regulatora krokowego.



Rysunek 8.1. Odpowiedź układu na zmianę wartości zadanej.



Rysunek 8.2. Przebieg uchybu regulacji w procesie wynikającym ze zmiany wartości zadanej.

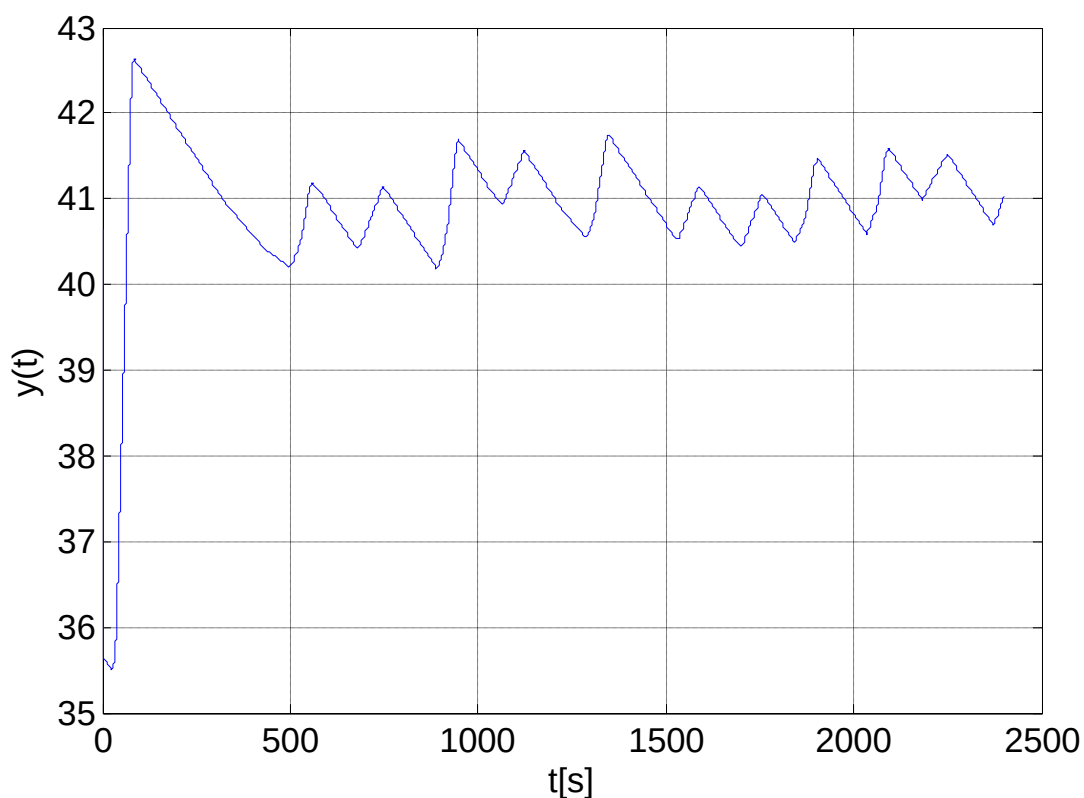


Rysunek 8.3. Przebieg wyjścia regulatora krokowego.

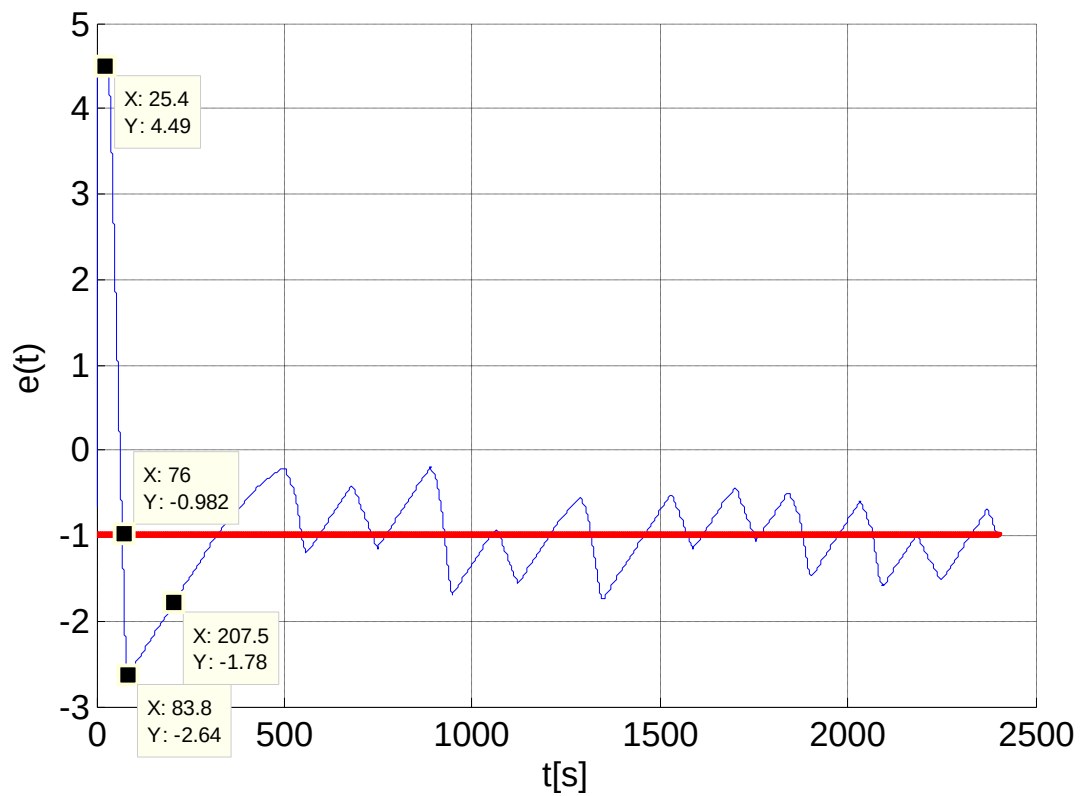
Tabela 8.3. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany wartości zadanej.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,9328	5,08 [%]	252.2 [s]	3191,6	16208	$2,7377 \cdot 10^6$	$4,4075 \cdot 10^9$

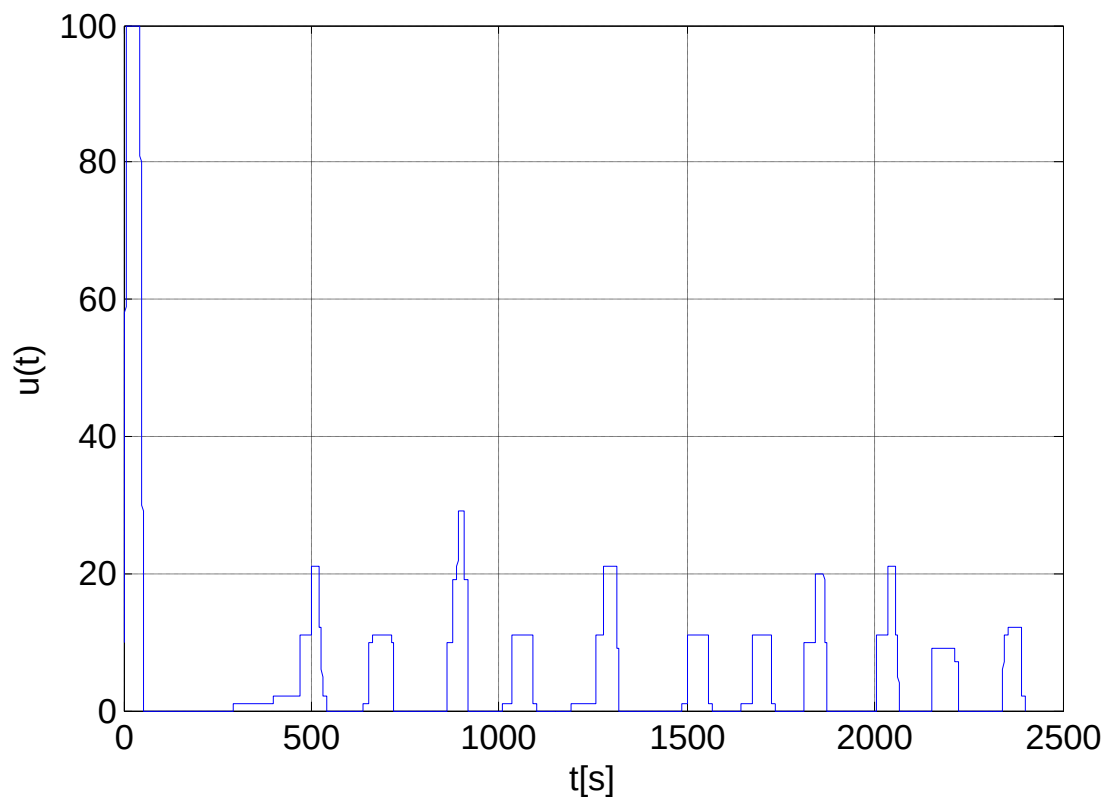
Rysunek 8.4 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.5 przebieg uchybu w procesie regulacji wynikającym z pojawienia się zakłócenia na wyjściu obiektu. Tabela 8.4 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.6 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.4. Odpowiedź układu na pojawienie się zakłócenia na wyjściu obiektu regulacji.



Rysunek 8.5. Przebieg uchybu regulacji w procesie wynikającym z pojawienia się zakłócenia na wyjściu obiektu regulacji.

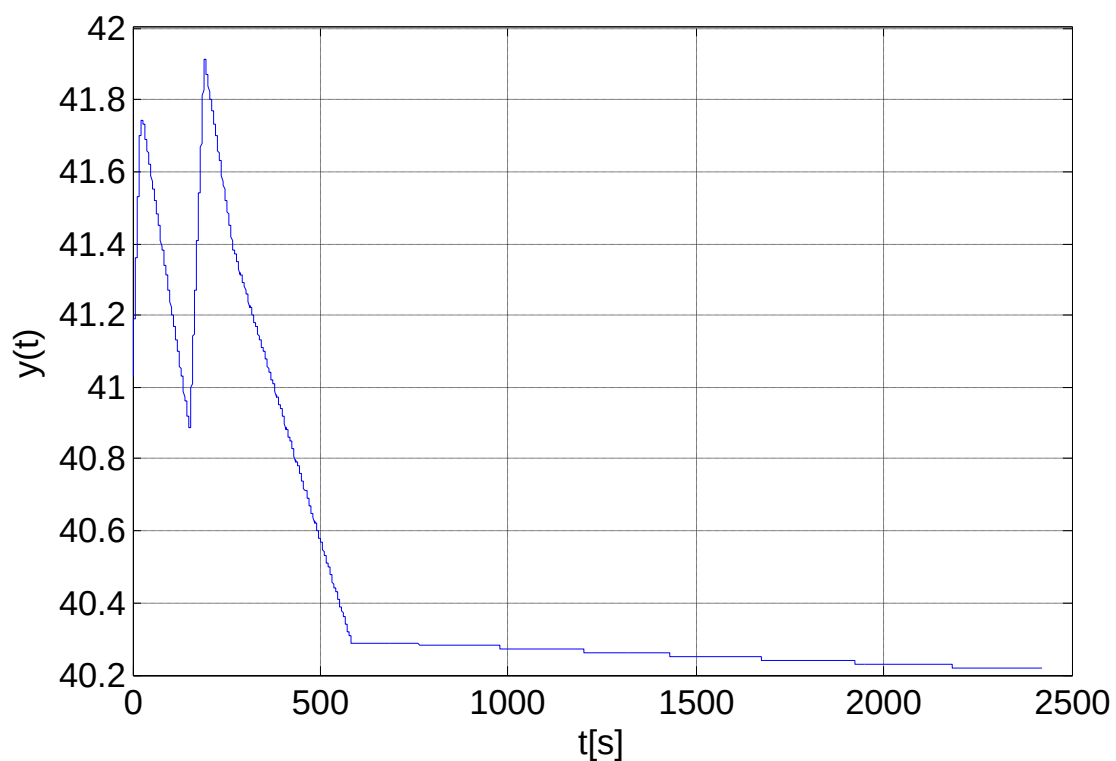


Rysunek 8.6. Przebieg wyjścia regulatora krokowego.

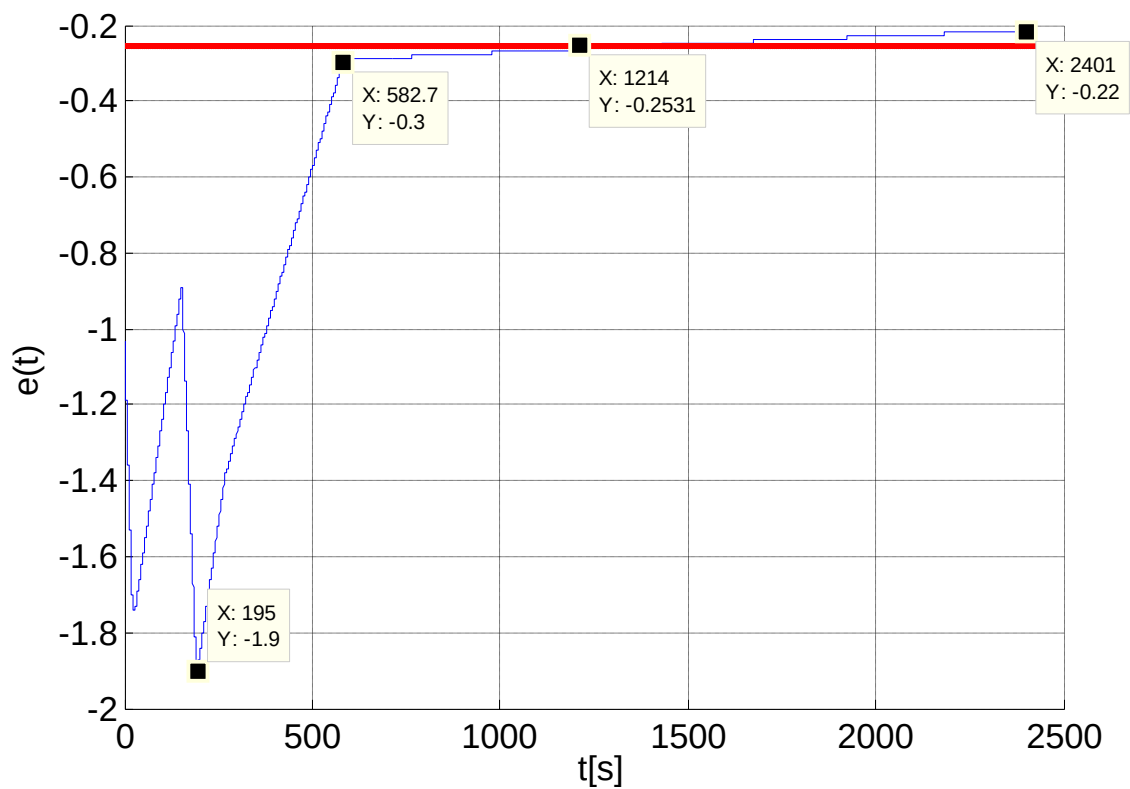
Tabela 8.4. Wskaźniki jakości opisujące proces regulacji wynikający z pojawienia się zakłócenia na wyjściu obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,982	30,30 [%]	207,5 [s]	2626,4	3841,4	$2,8848 \cdot 10^6$	$4,6880 \cdot 10^9$

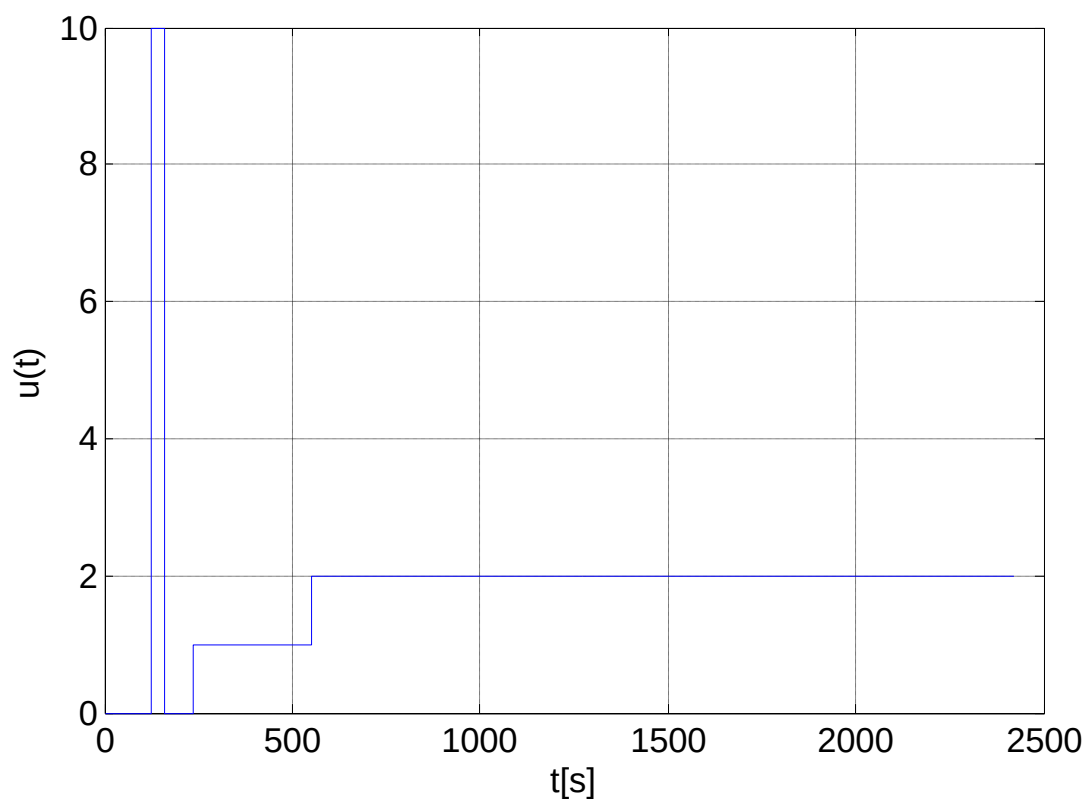
Rysunek 8.7 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.8 przebieg uchybu w procesie regulacji wynikającym ze zmiany parametrów obiektu regulacji. Tabela 8.5 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.9 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.7. Odpowiedź układu na zmianę parametrów obiektu regulacji.



Rysunek 8.8. Przebieg uchybu regulacji w procesie wynikającym ze zmiany parametrów obiektu regulacji.



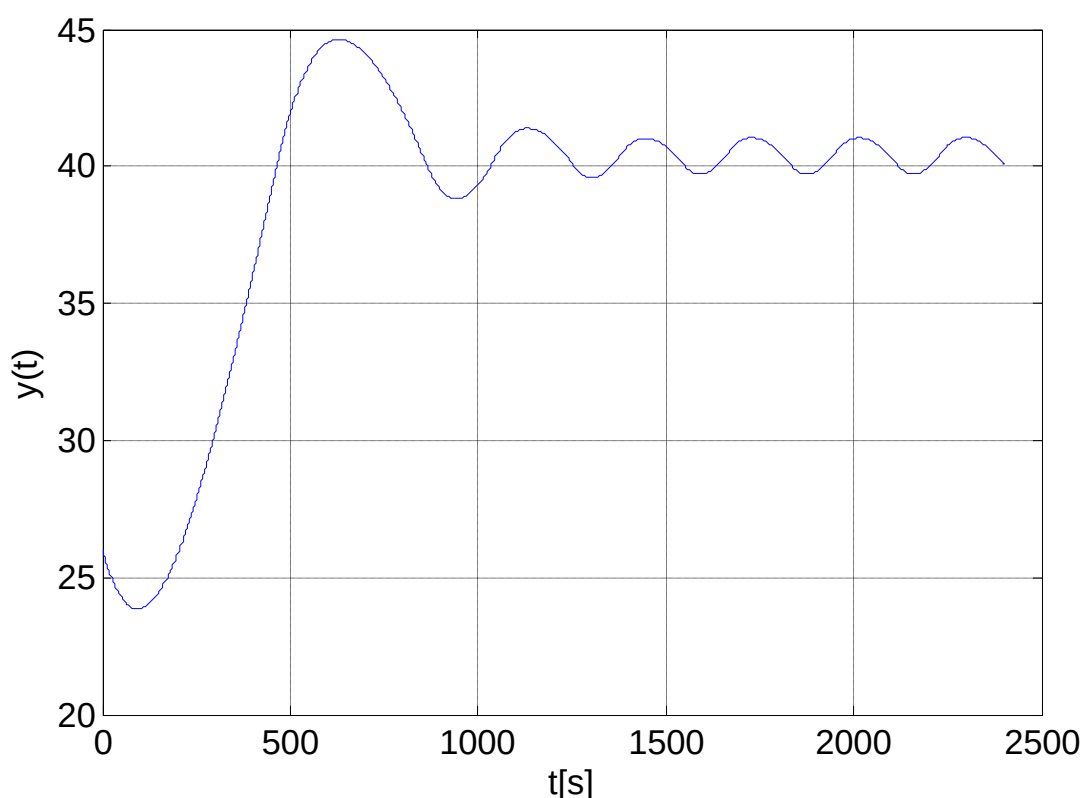
Rysunek 8.9. Przebieg wyjścia regulatora krokowego.

Tabela 8.5. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany parametrów obiektu regulacji.

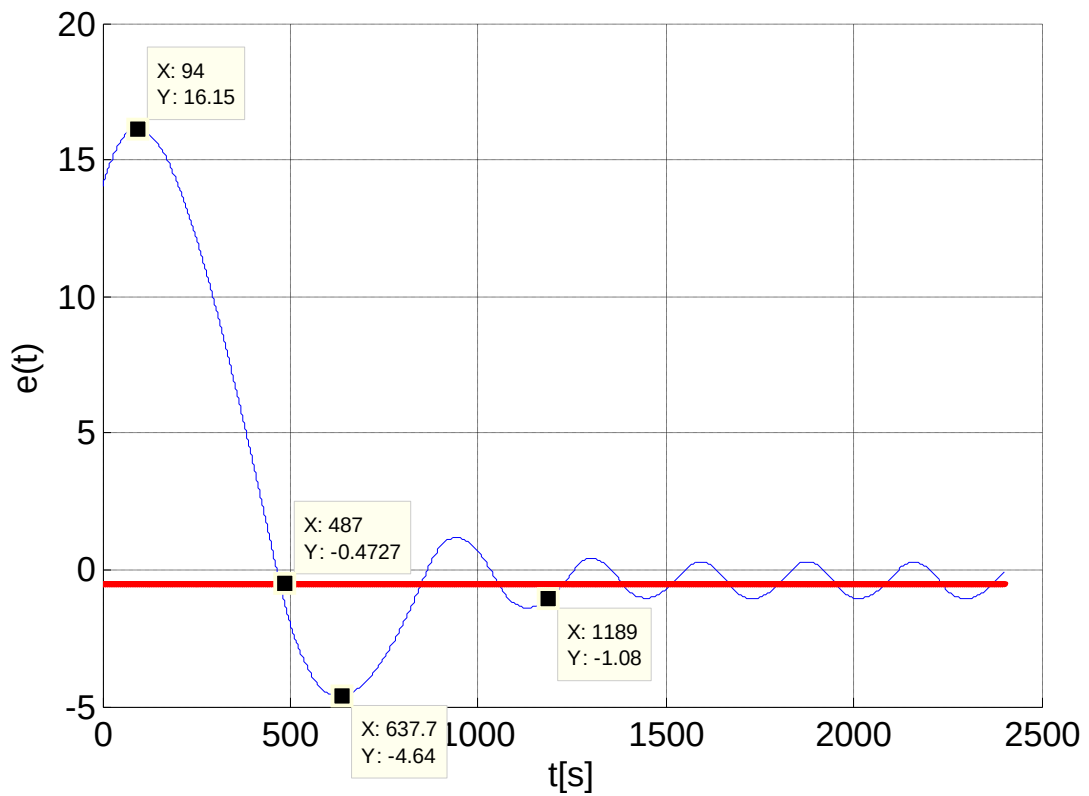
e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,253	2,01 [%]	583,7 [s]	1114,3	944,6	$8,3351 \cdot 10^5$	$1,1699 \cdot 10^9$

Obiekt typu inercja II rzędu
$$K(s) = \frac{k}{(1+sT_1)(1+sT_2)}$$

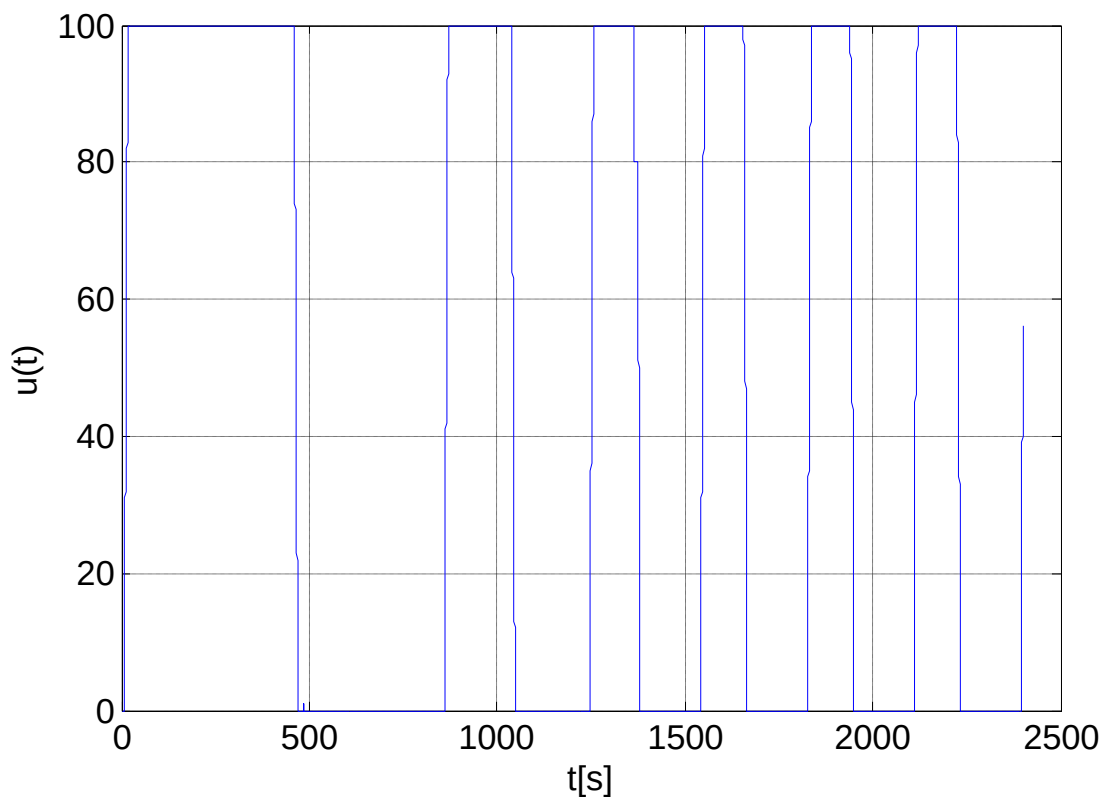
Rysunek 8.10 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.11 przebieg uchybu w procesie regulacji wynikającym ze zmiany wartości zadanej. Tabela 8.6 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.12 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.10. Odpowiedź układu na zmianę wartości zadanej.



Rysunek 8.11. Przebieg uchybu regulacji w procesie wynikającym ze zmiany wartości zadanej.

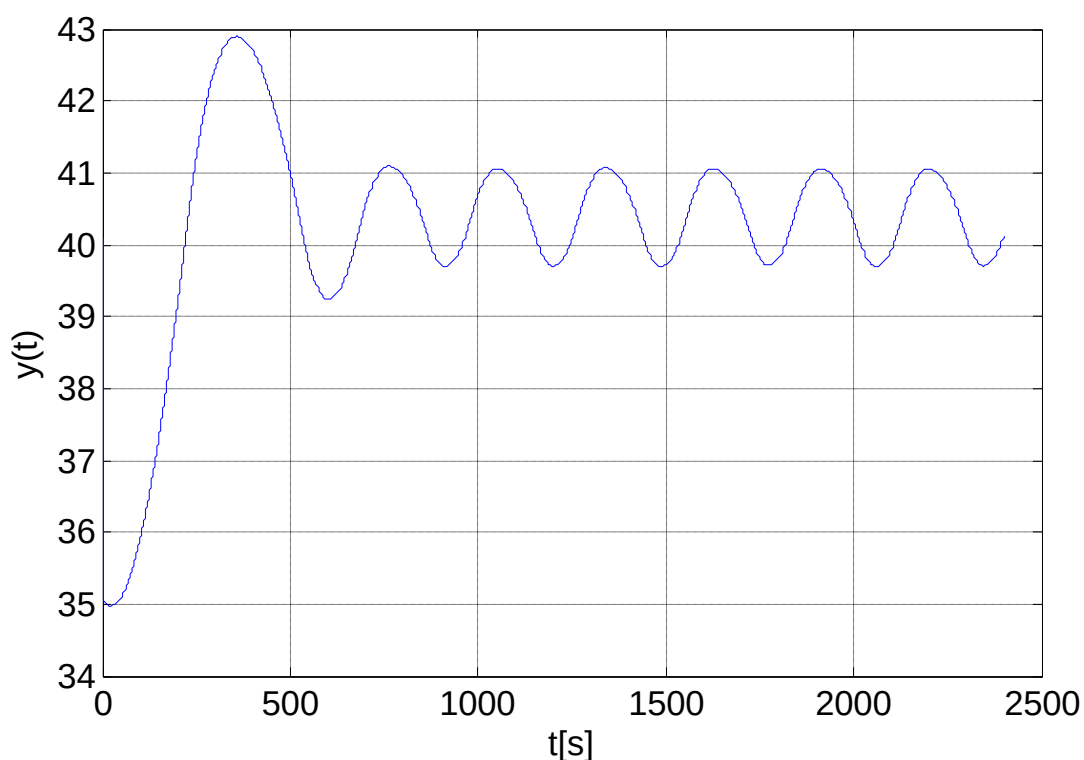


Rysunek 8.12. Przebieg wyjścia regulatora krokowego.

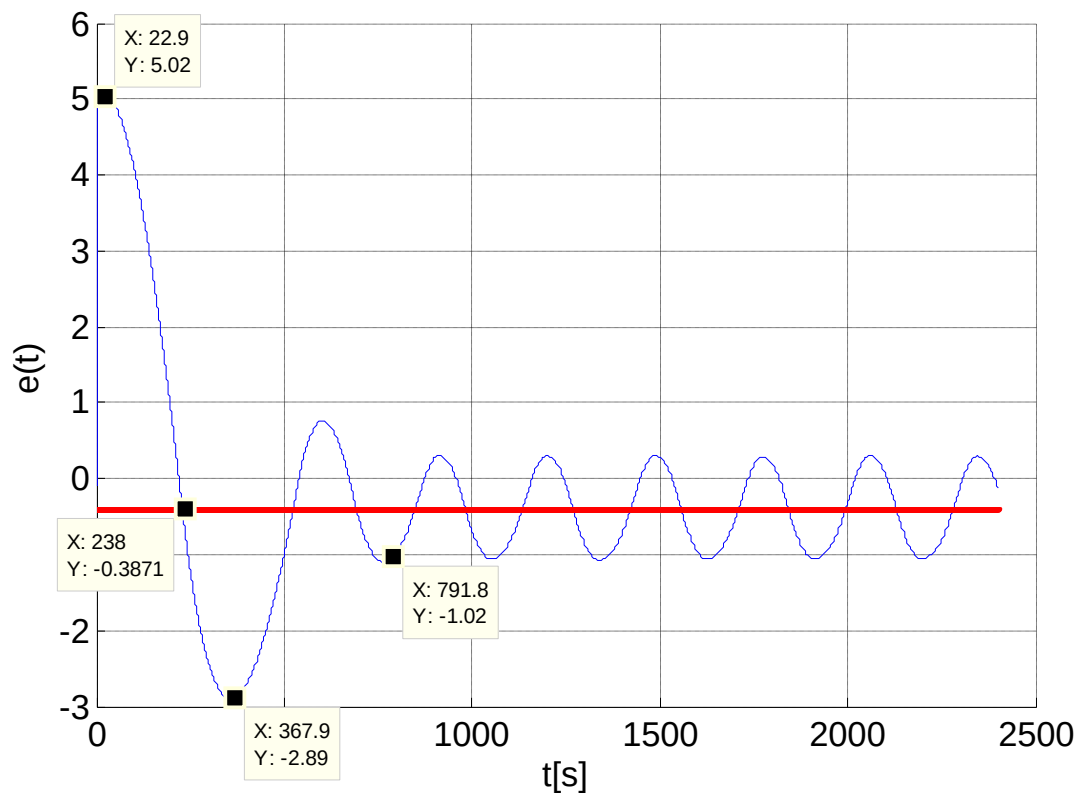
Tabela 8.6. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany wartości zadanej.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,473	26,58 [%]	1189 [s]	7290,6	73325	$3,1898 \cdot 10^6$	$3,3091 \cdot 10^9$

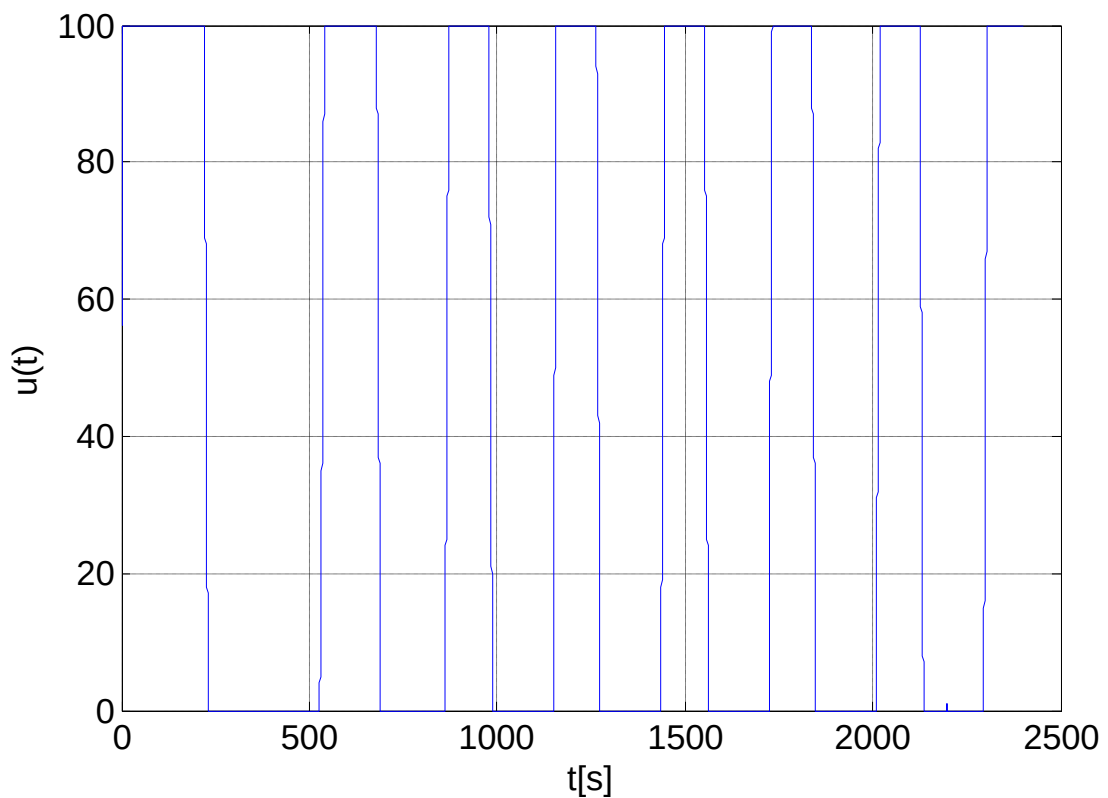
Rysunek 8.13 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.14 przebieg uchybu w procesie regulacji wynikającym z pojawienia się zakłócenia na wyjściu obiektu. Tabela 8.7 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.15 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.13. Odpowiedź układu na pojawienie się zakłócenia na wyjściu obiektu regulacji.



Rysunek 8.14. Przebieg uchybu regulacji w procesie wynikającym z pojawienia się zakłócenia na wyjściu obiektu regulacji.

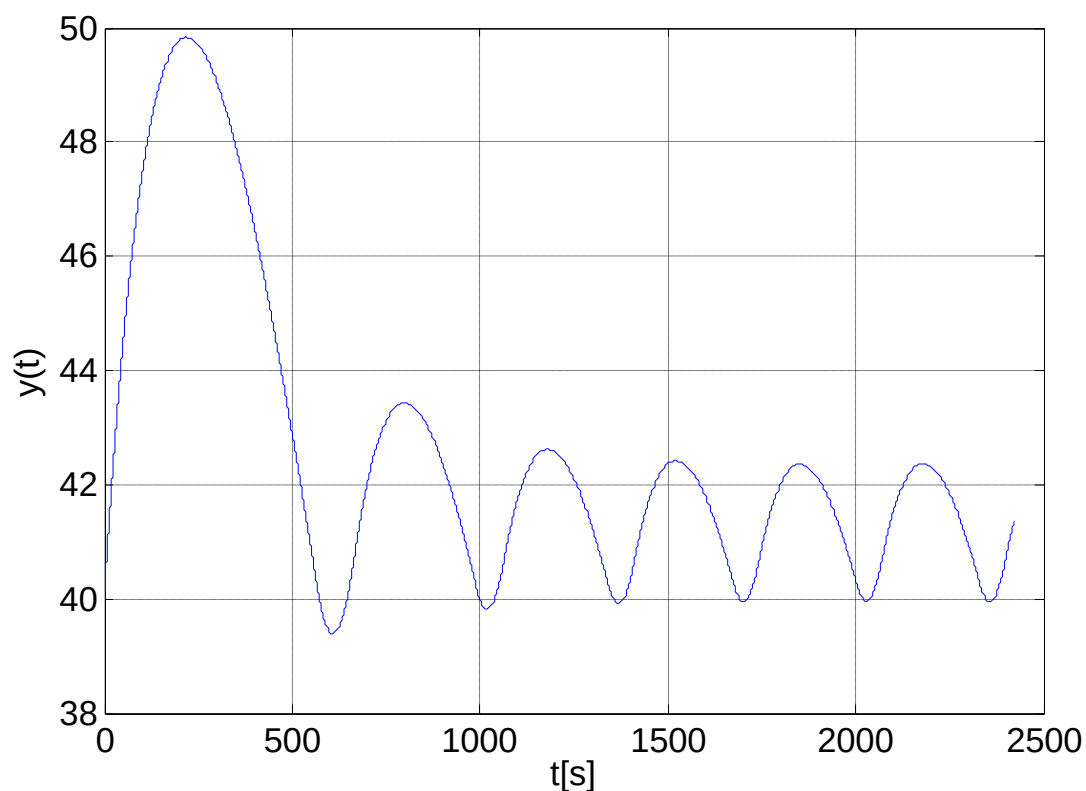


Rysunek 8.15. Przebieg wyjścia regulatora krokowego.

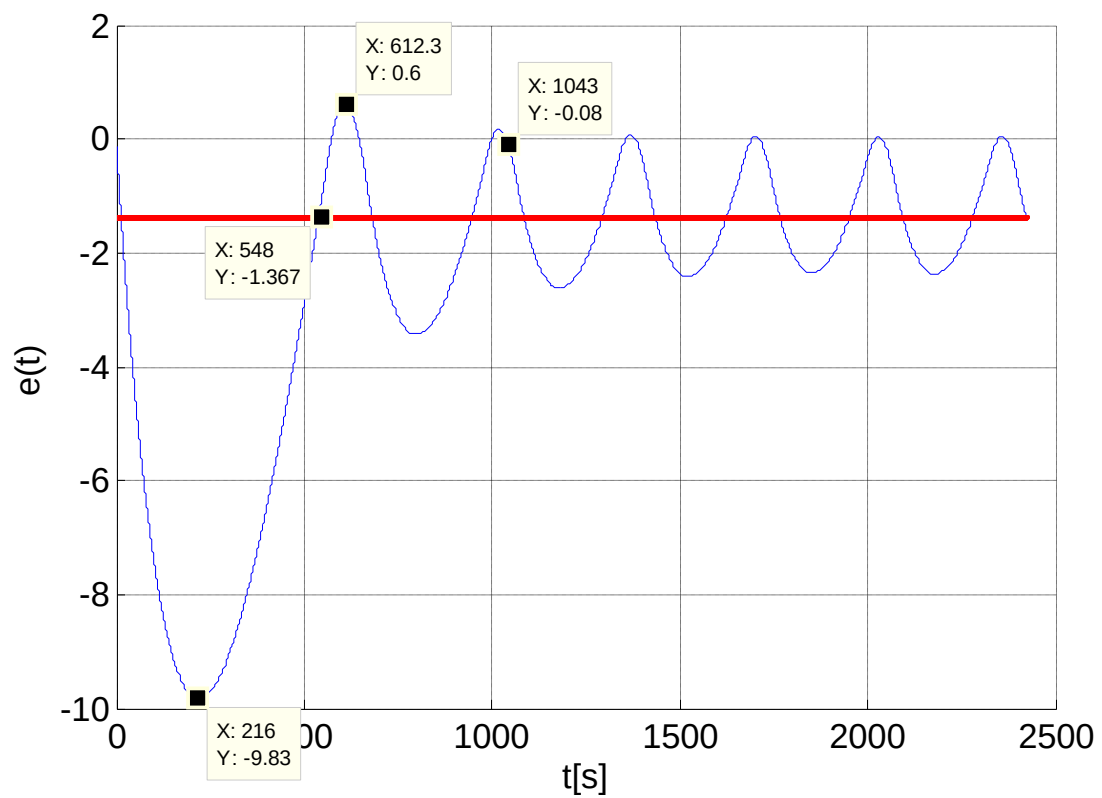
Tabela 8.7. Wskaźniki jakości opisujące proces regulacji wynikający z pojawienia się zakłócenia na wyjściu obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,387	54,02 [%]	791,8 [s]	2358,5	5242,9	$1,728 \cdot 10^6$	$2,4452 \cdot 10^9$

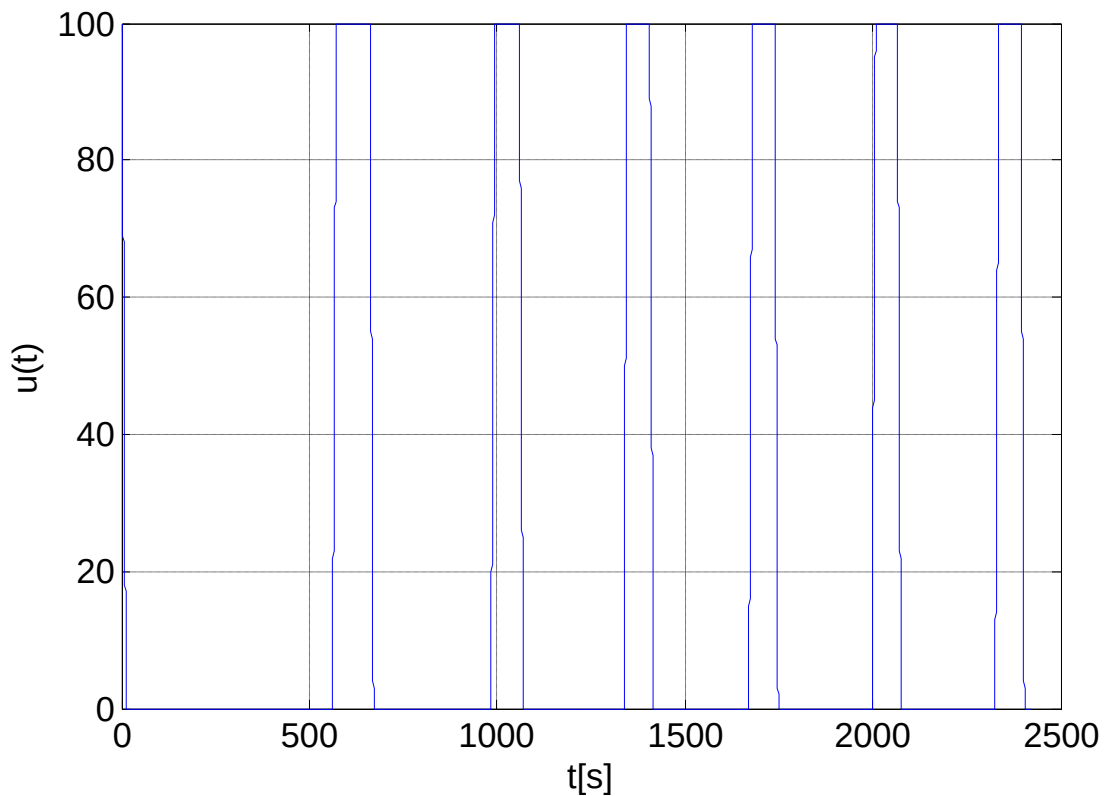
Rysunek 8.16 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.17 przebieg uchybu w procesie regulacji wynikającym ze zmiany parametrów obiektu regulacji. Tabela 8.8 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.18 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.16. Odpowiedź układu na zmianę parametrów obiektu regulacji.



Rysunek 8.17. Przebieg uchybu regulacji w procesie wynikającym ze zmiany parametrów obiektu regulacji.



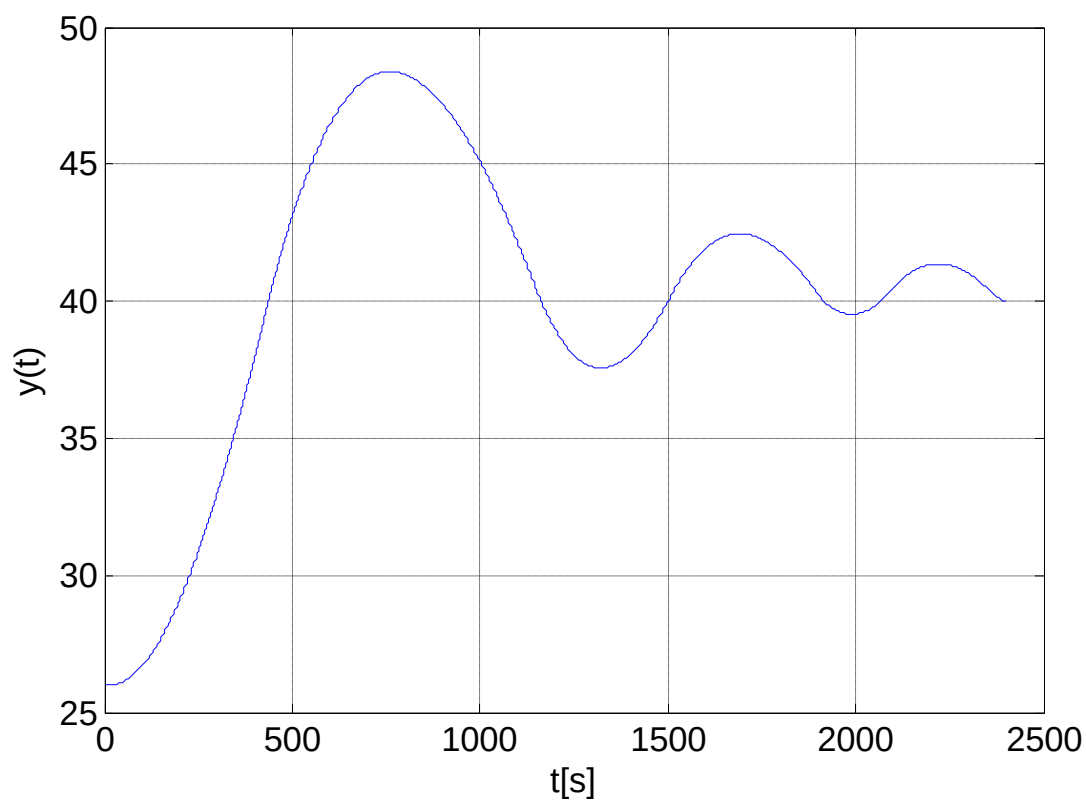
Rysunek 8.18. Przebieg wyjścia regulatora krokowego.

Tabela 8.8. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany parametrów obiektu regulacji.

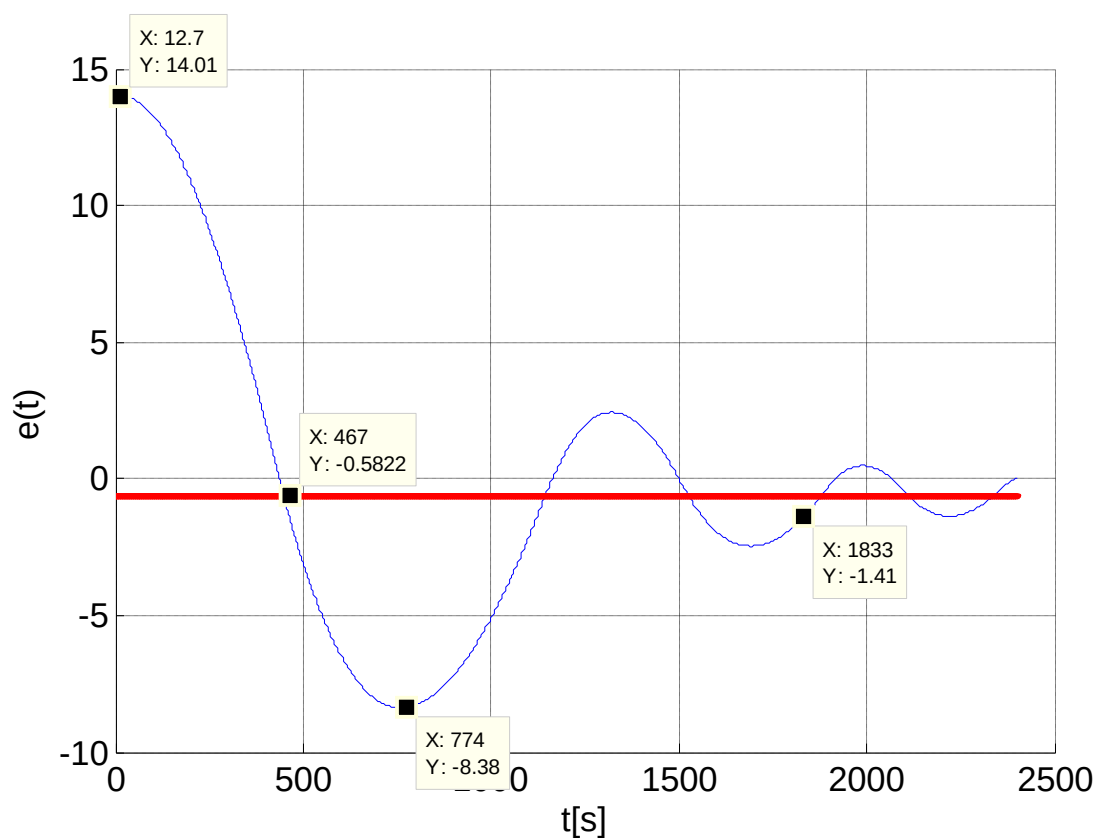
e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-1,367	23,34 [%]	1043 [s]	6410,1	34229	$4,8818 \cdot 10^6$	$6.6658 \cdot 10^9$

$$\text{Obiekt oscylacyjny } (K(s) = \frac{k\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2})$$

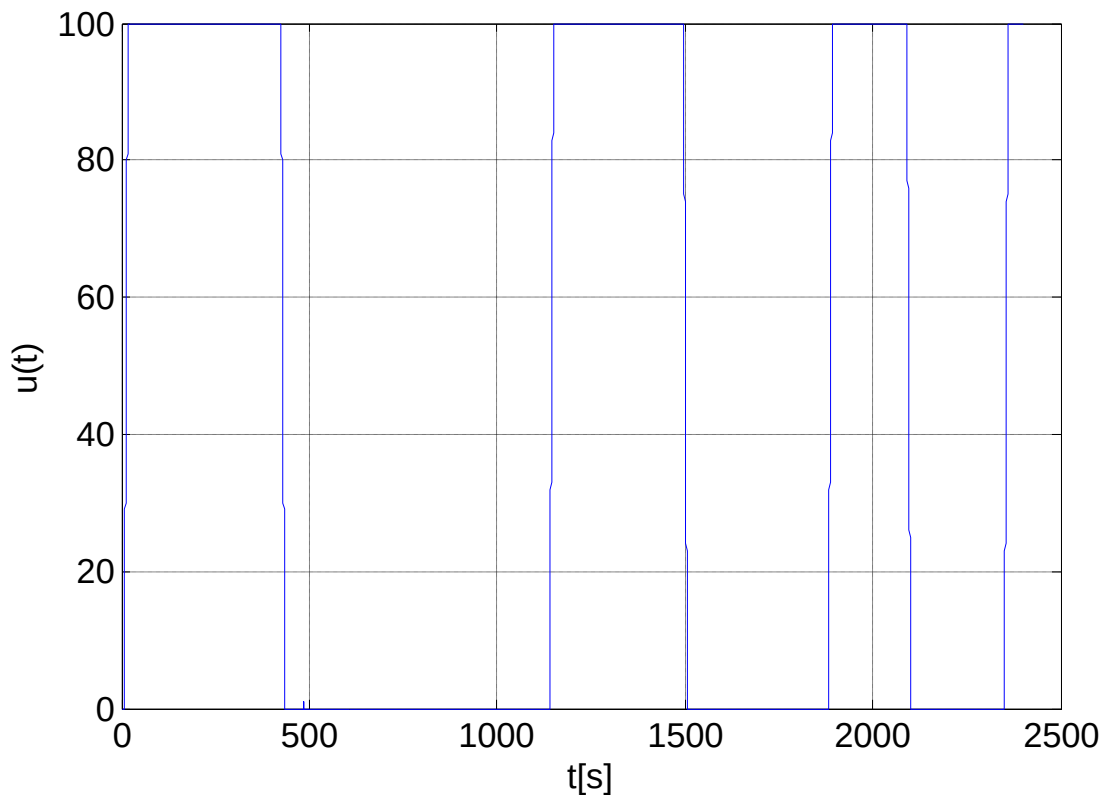
Rysunek 8.19 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.20 przebieg uchybu w procesie regulacji wynikającym ze zmiany wartości zadanej. Tabela 8.9 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.21 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.19. Odpowiedź układu na zmianę wartości zadanej.



Rysunek 8.20. Przebieg uchybu regulacji w procesie wynikającym ze zmiany wartości zadanej.

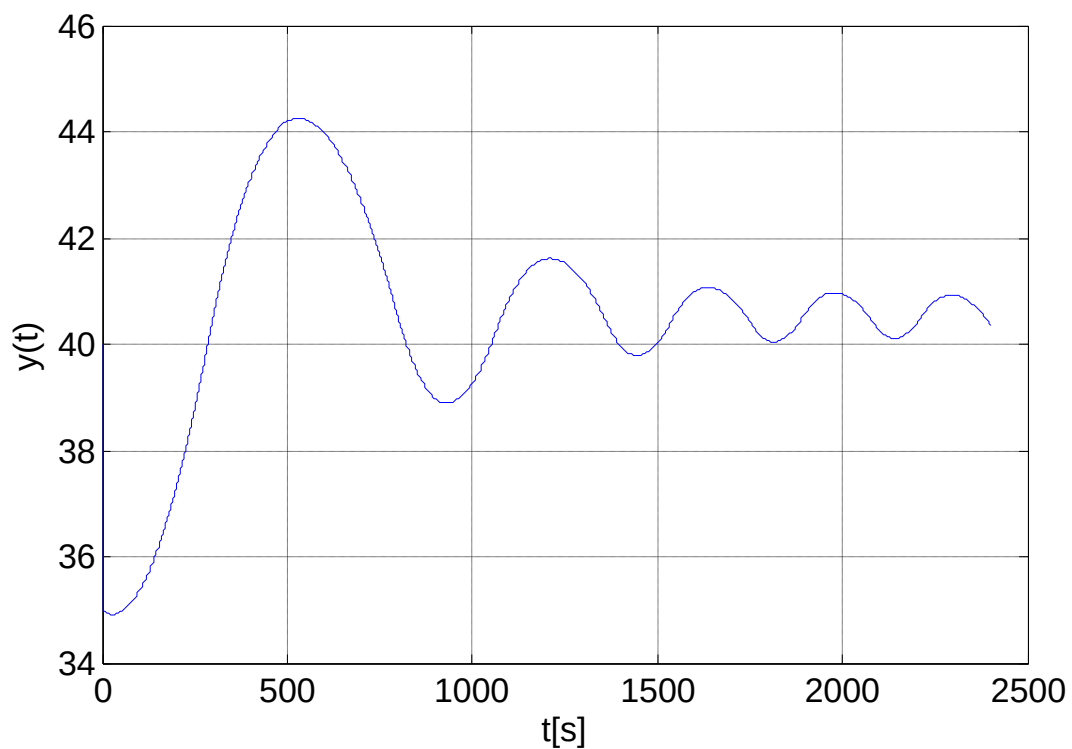


Rysunek 8.21. Przebieg wyjścia regulatora krokowego.

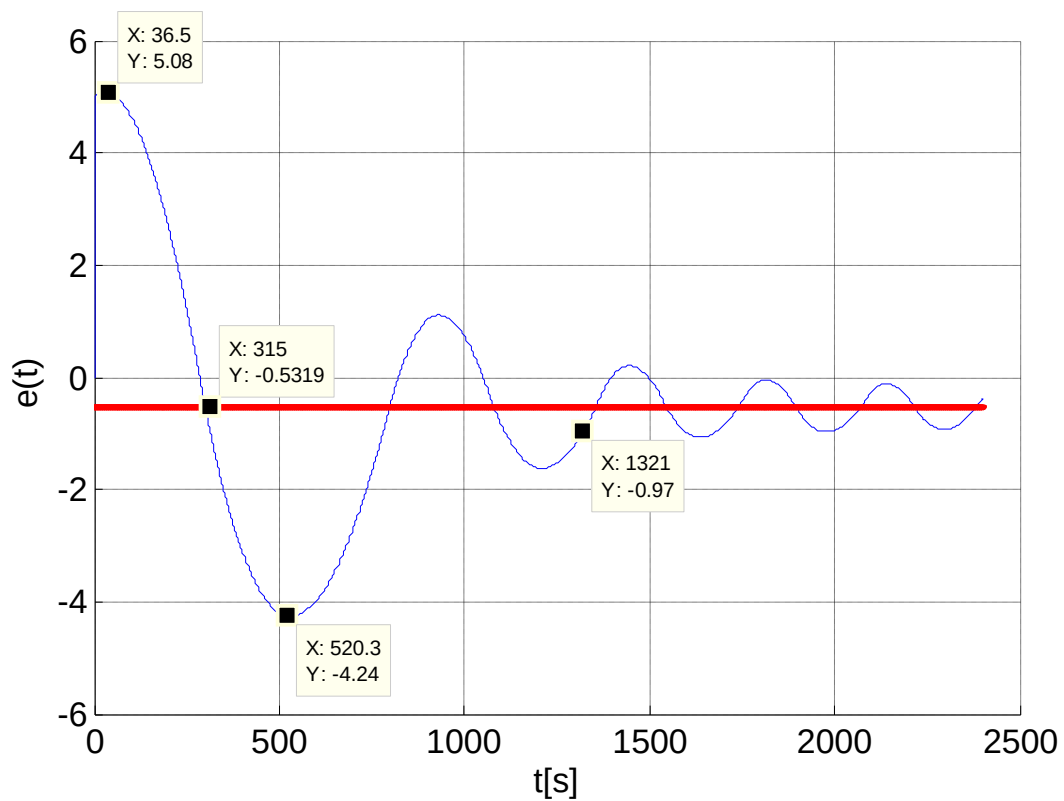
Tabela 8.9. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany wartości zadanej.

$e_{\dot{s}r}$	σ	t_r	I_1	I_2	I_3	I_4
-0,5822	53,47 [%]	1833 [s]	9576,5	74404	$6,3915 \cdot 10^6$	$7,242 \cdot 10^9$

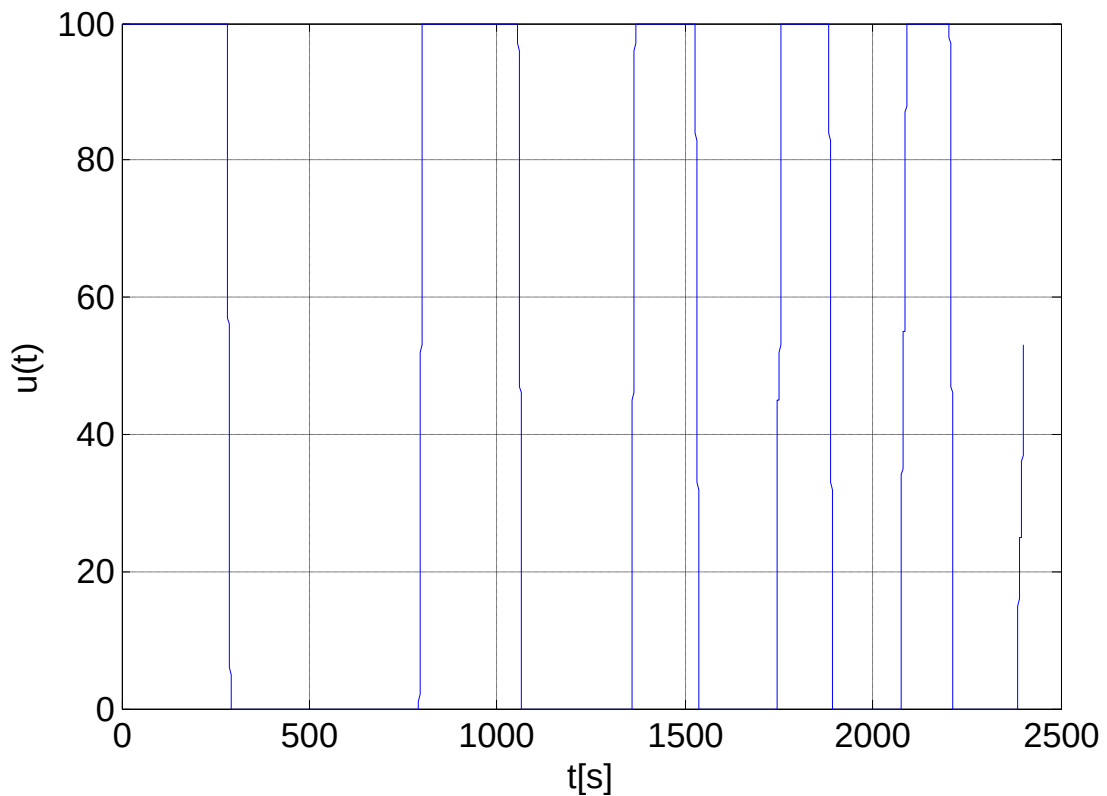
Rysunek 8.22 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.23 przebieg uchybu w procesie regulacji wynikającym z pojawienia się zakłócenia na wyjściu obiektu. Tabela 8.10 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.24 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.22. Odpowiedź układu na pojawienie się zakłócenia na wyjściu obiektu regulacji.



Rysunek 8.23. Przebieg uchybu regulacji w procesie wynikającym z pojawienia się zakłócenia na wyjściu obiektu regulacji.

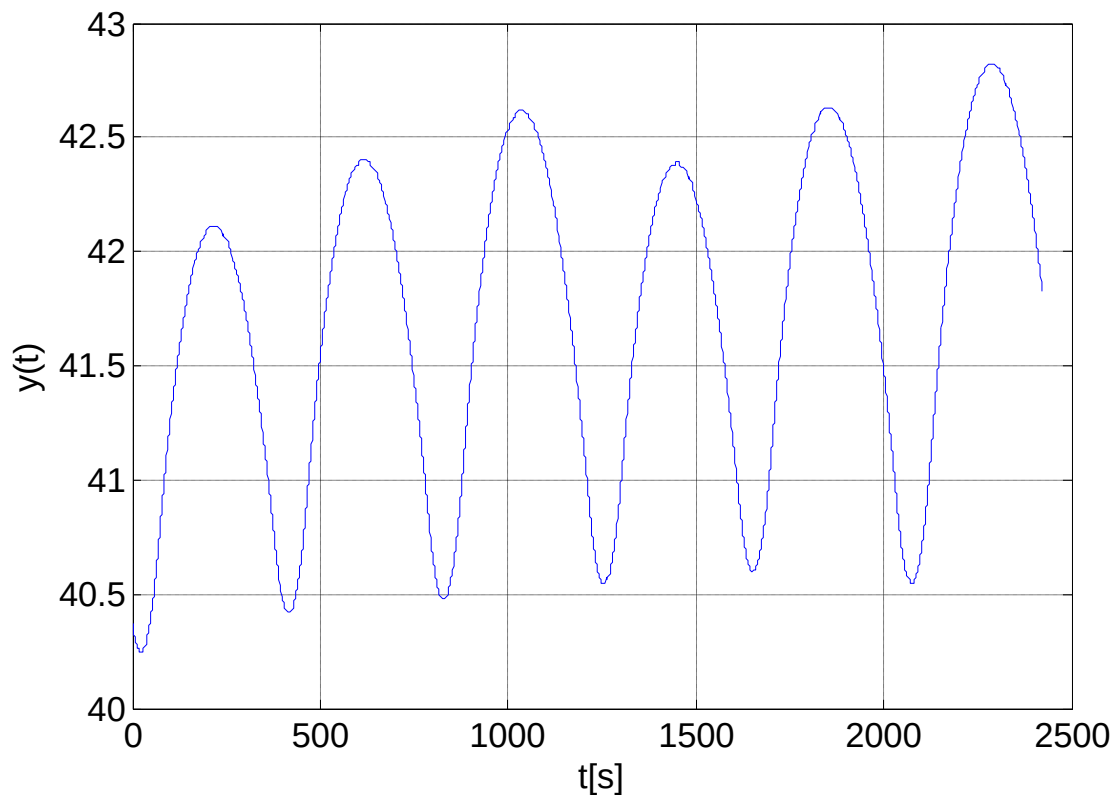


Rysunek 8.24. Przebieg wyjścia regulatora krokowego.

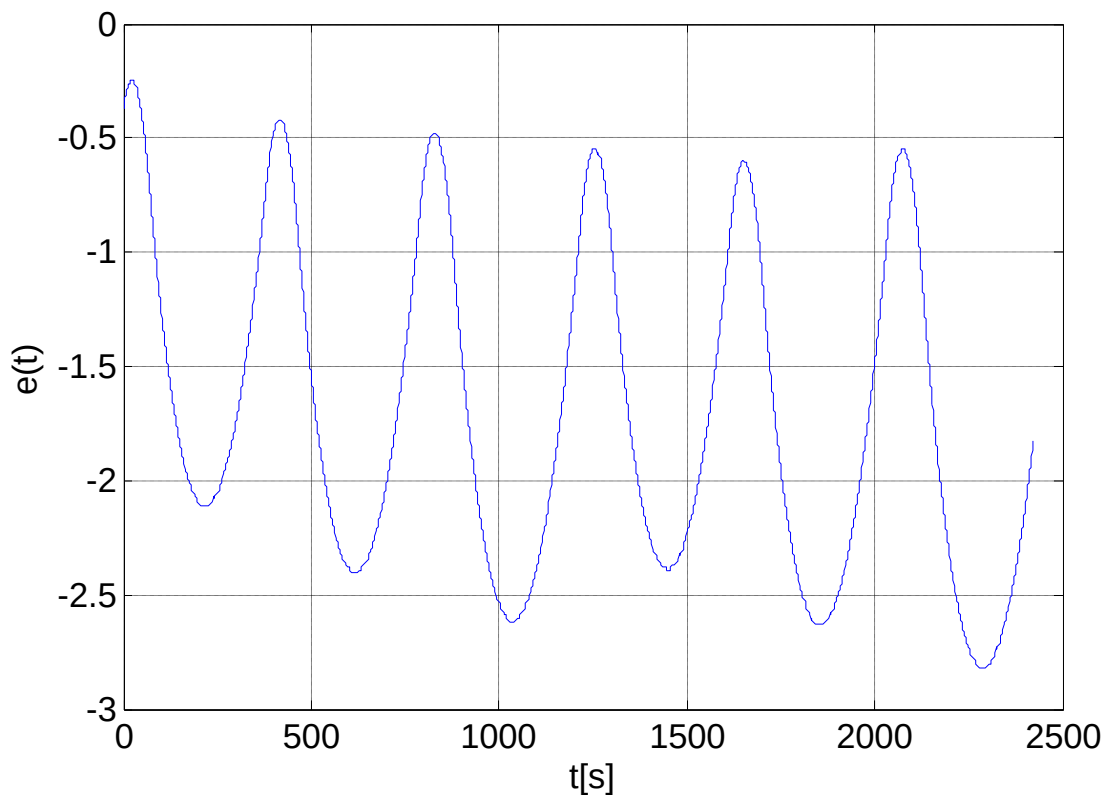
Tabela 8.10. Wskaźniki jakości opisujące proces regulacji wynikający z pojawienia się zakłócenia na wyjściu obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,5319	66,08 [%]	1321 [s]	3589,4	10318	$2,6001 \cdot 10^6$	$3,2929 \cdot 10^9$

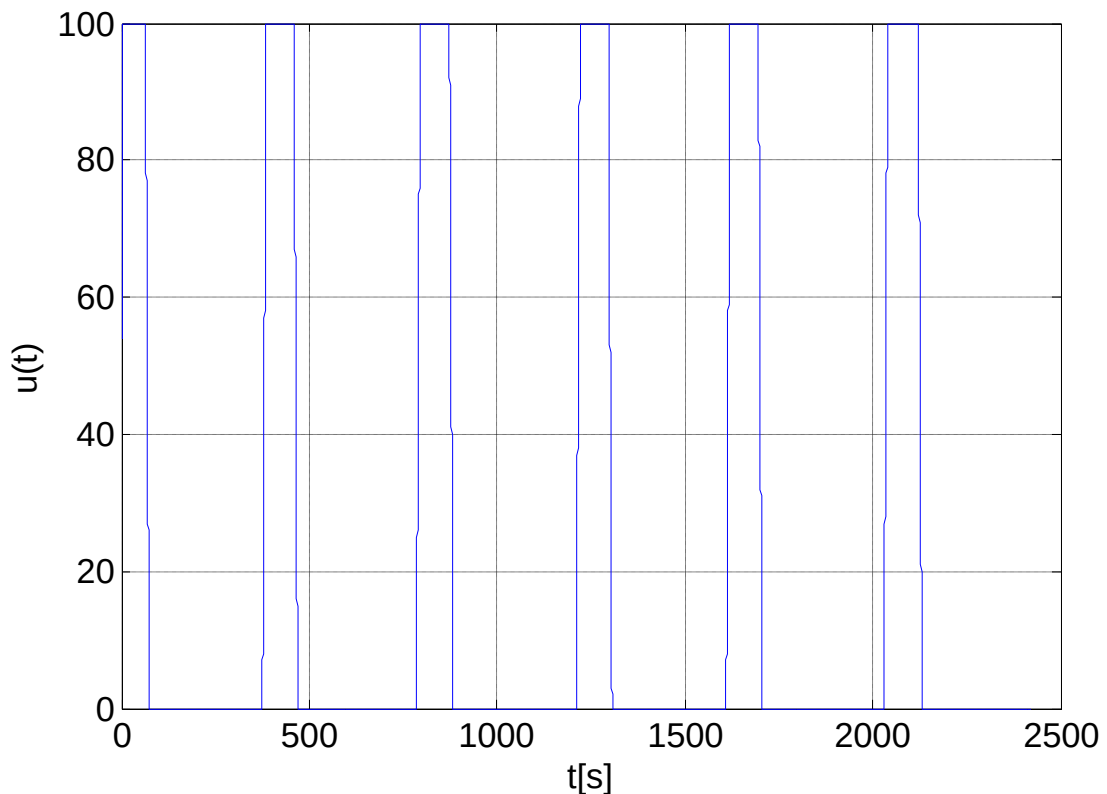
Rysunek 8.25 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.26 przebieg uchybu w procesie regulacji wynikającym ze zmiany parametrów obiektu regulacji. Tabela 8.11 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.27 przedstawia przebieg wyjścia regulatora krokowego.



Rysunek 8.25. Odpowiedź układu na zmianę parametrów obiektu regulacji.



Rysunek 8.26. Przebieg uchybu regulacji w procesie wynikającym ze zmiany parametrów obiektu regulacji.



Rysunek 8.27. Przebieg wyjścia regulatora krokowego.

Tabela 8.11. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany parametrów obiektu regulacji.

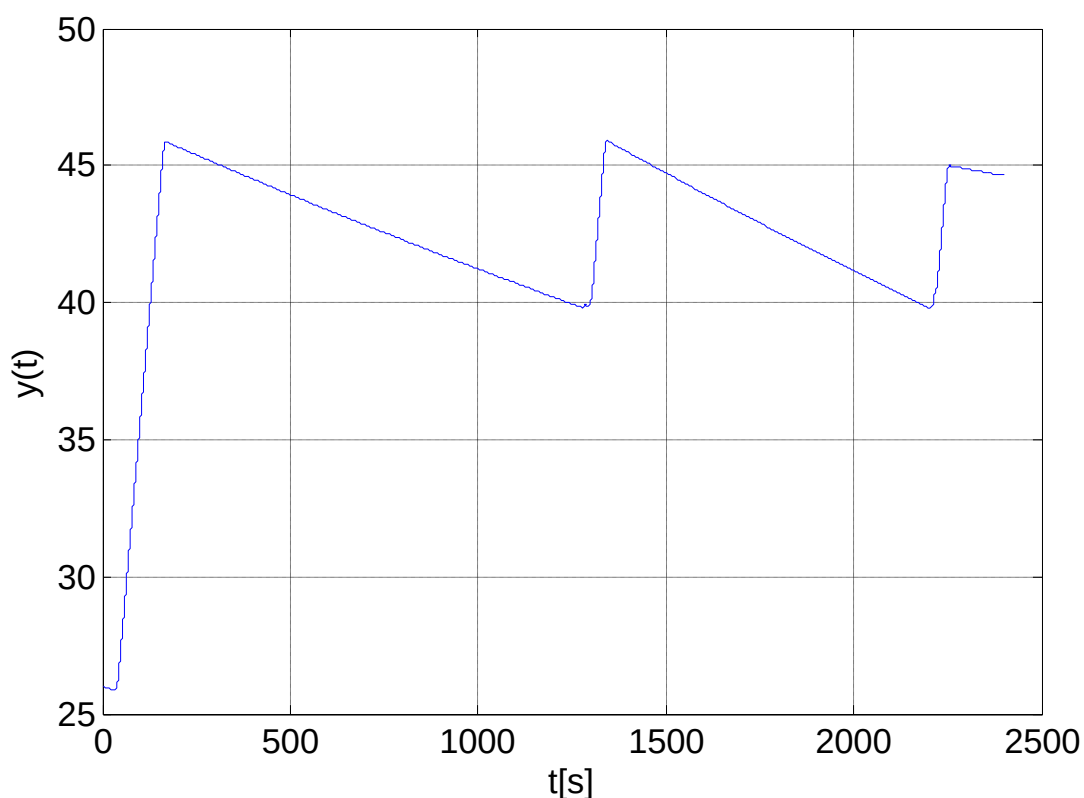
e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-	-	-	-	-	-	-

8.2 Testy jakości regulacji za pomocą algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym

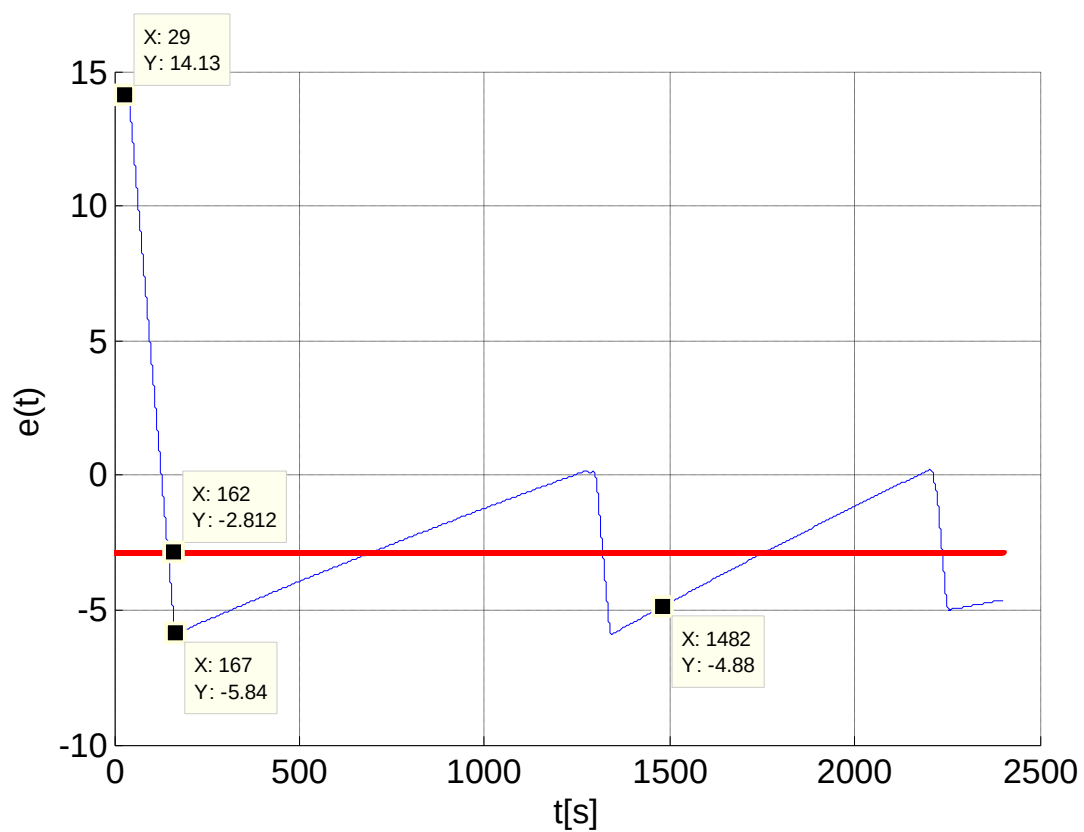
Niniejszy rozdział zawiera dane, zebrane podczas testów jakości regulacji za pomocą algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym, przedstawione w formie wykresów oraz tabel zawierających wskaźniki jakości opisane w rozdziale 7.7. Poniżej zamieszczono wyniki uzyskane dla poszczególnych obiektów regulacji.

Obiekt typu inercja z opóźnieniem ($K(s) = \frac{k}{1+sT} e^{-sT_0}$)

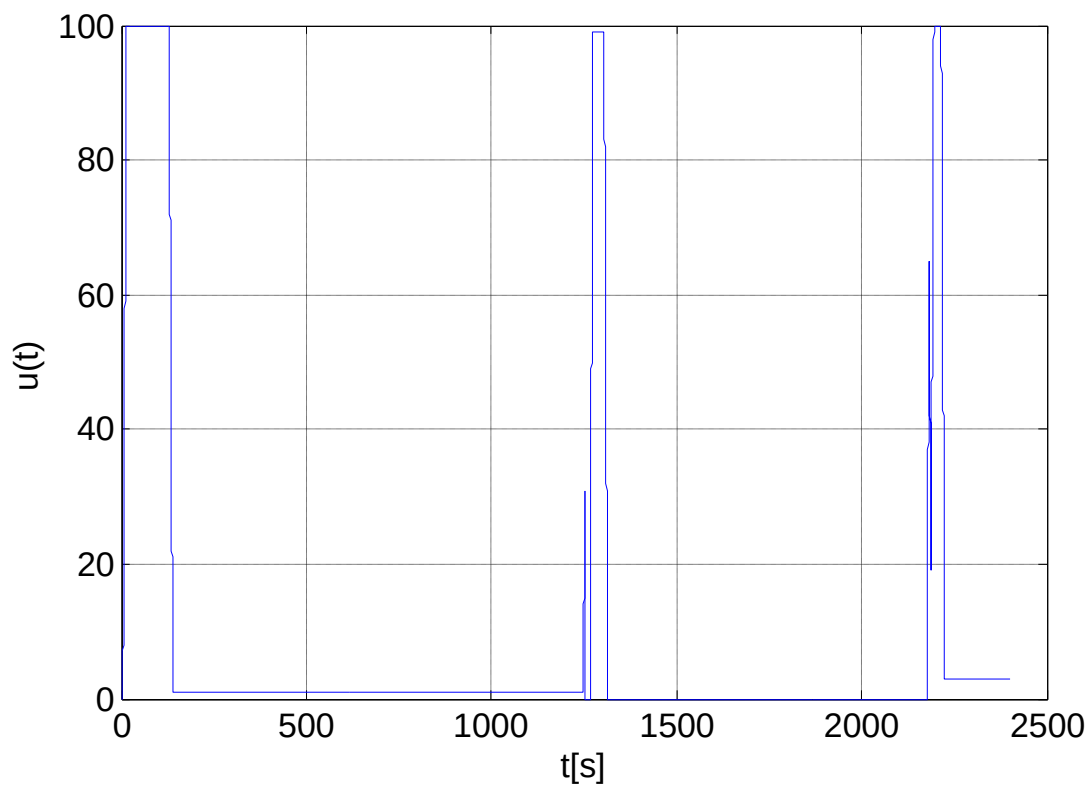
Rysunek 8.28 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.29 przebieg uchybu w procesie regulacji wynikającym ze zmiany wartości zadanej. Tabela 8.12 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.30 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.28. Odpowiedź układu na zmianę wartości zadanej



Rysunek 8.29. Przebieg uchybu regulacji w procesie wynikającym ze zmiany wartości zadanej.

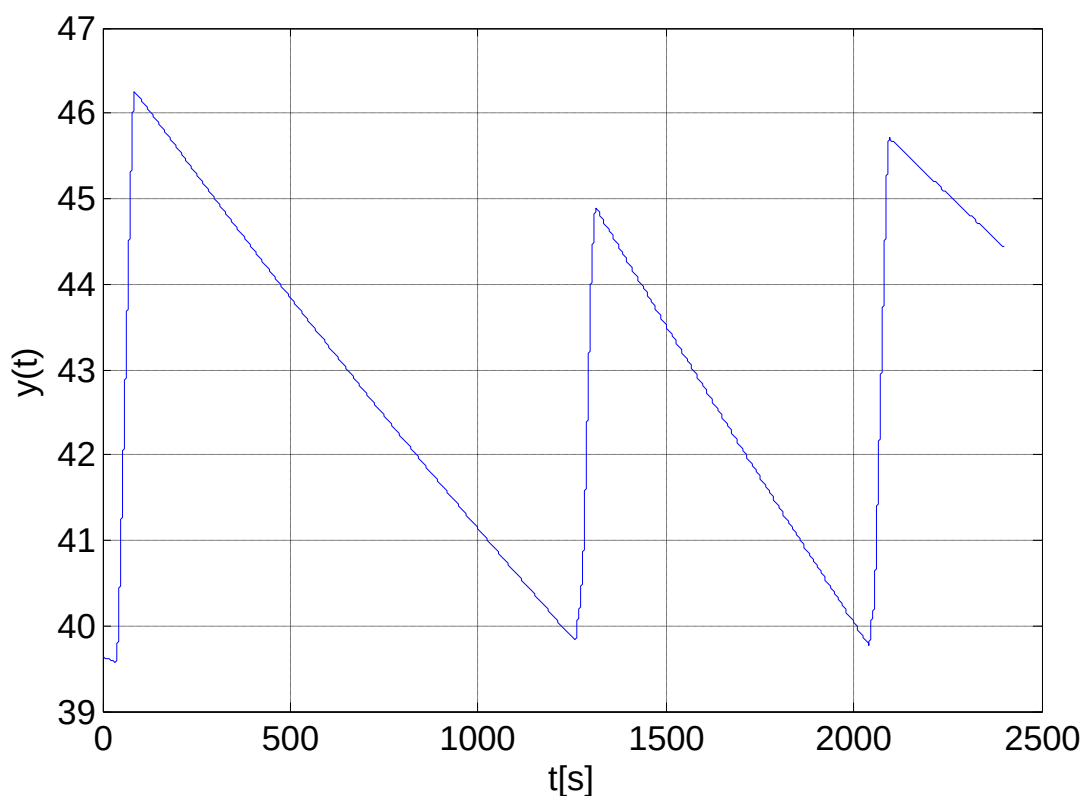


Rysunek 8.30. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

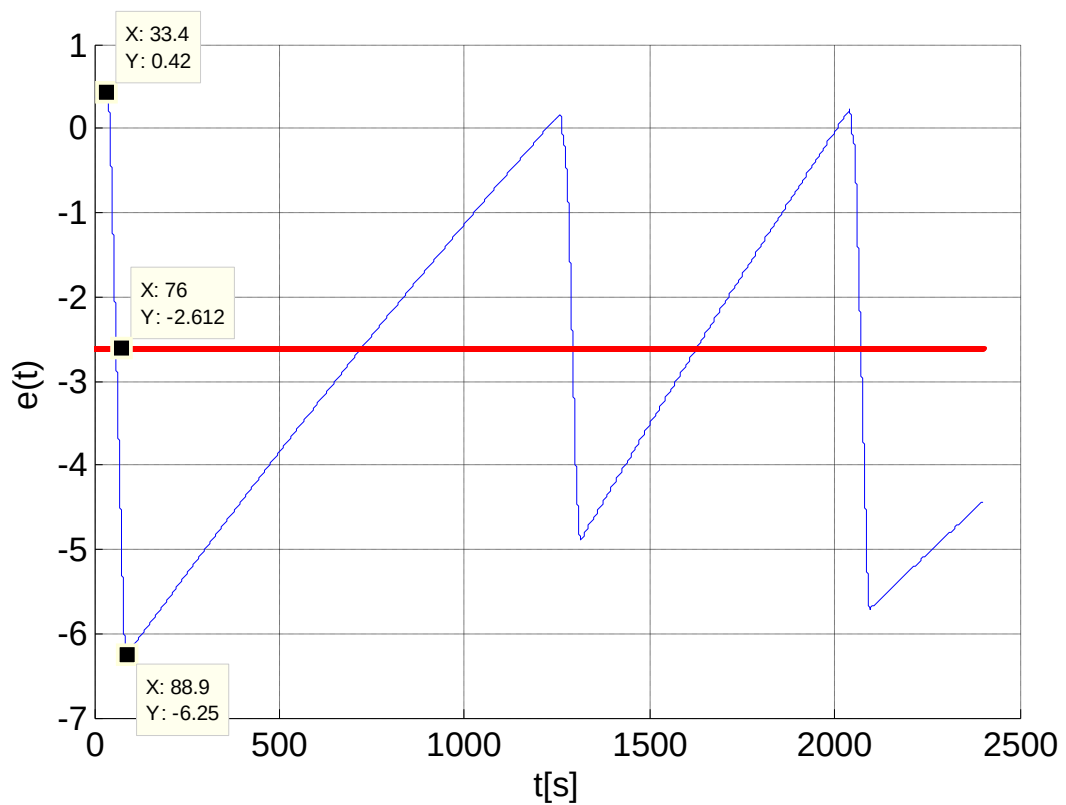
Tabela 8.12. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany wartości zadanej.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-2,812	17,87 [%]	1482 [s]	7728,2	39816	$7,6155 \cdot 10^6$	$1,1988 \cdot 10^{10}$

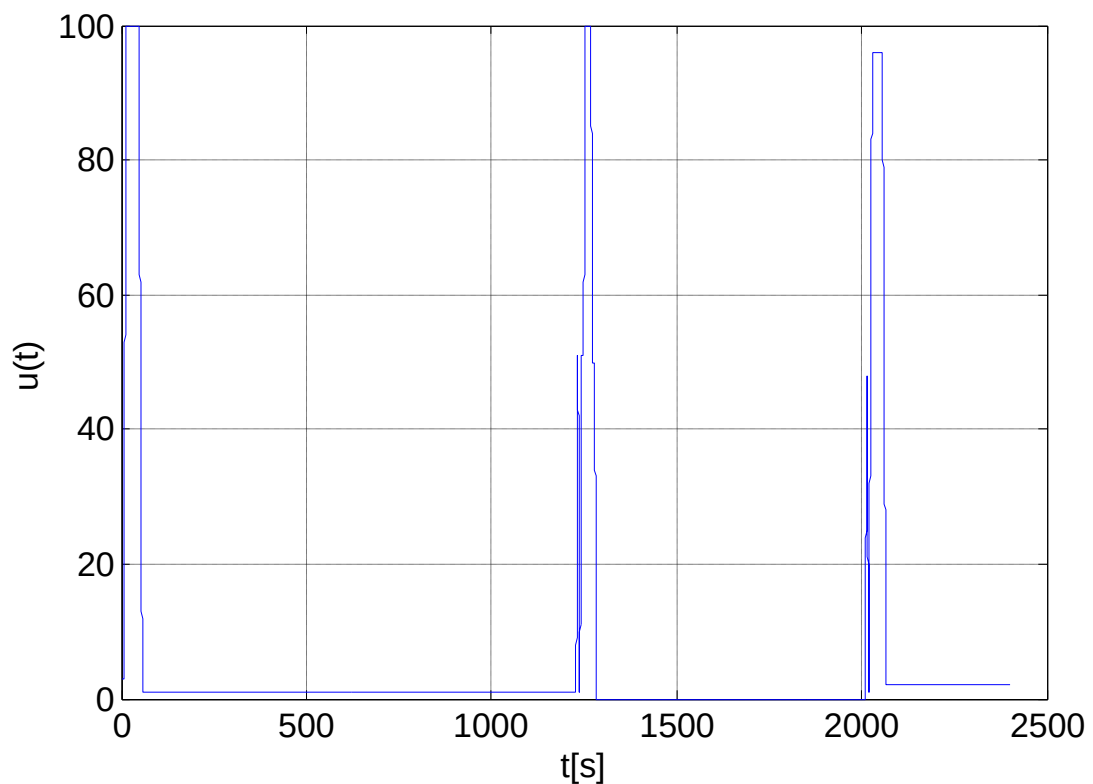
Rysunek 8.31 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.32 przebieg uchybu w procesie regulacji wynikającym z pojawienia się zakłócenia na wyjściu obiektu. Tabela 8.13 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.33 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.31. Odpowiedź układu na pojawienie się zakłócenia na wyjściu obiektu regulacji.



Rysunek 8.32. Przebieg uchybu regulacji w procesie wynikającym z pojawienia się zakłócenia na wyjściu obiektu regulacji.

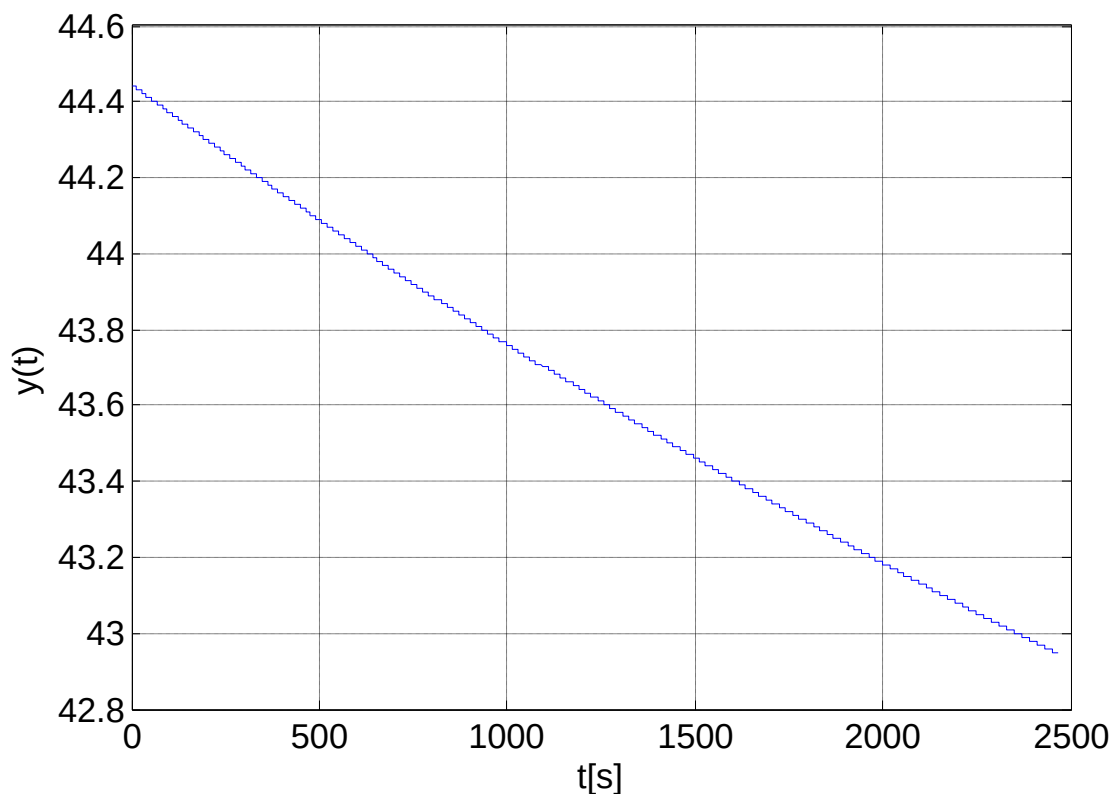


Rysunek 8.33 Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

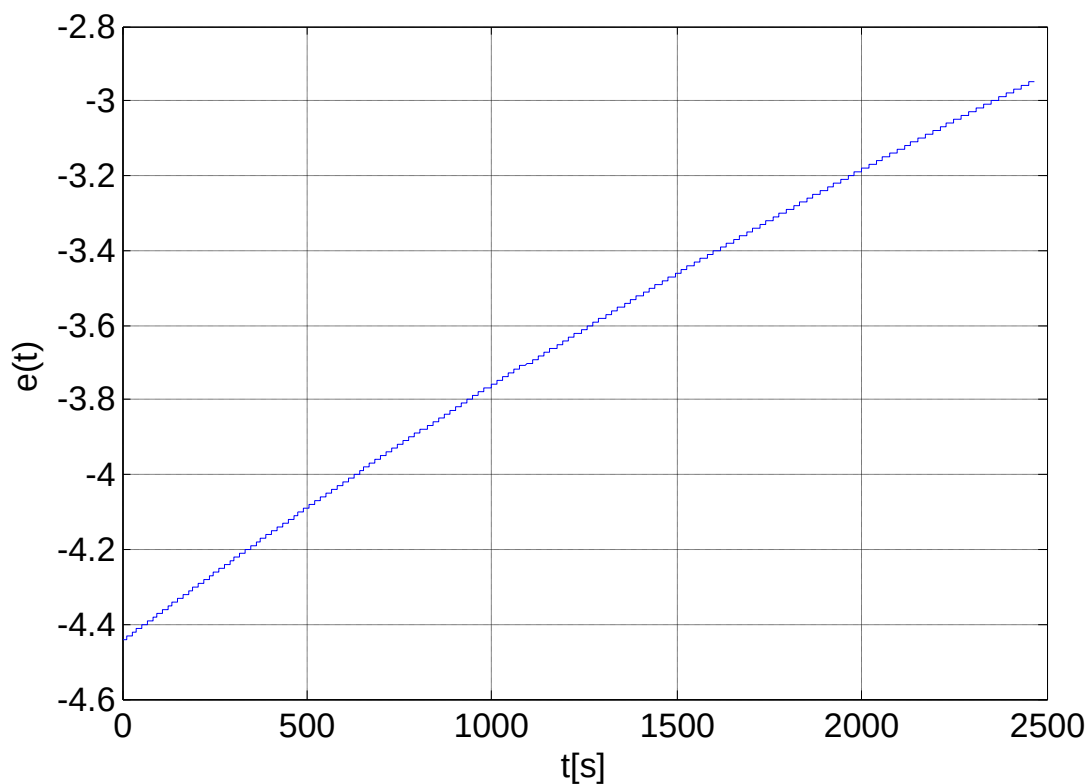
Tabela 8.13. Wskaźniki jakości opisujące proces regulacji wynikający z pojawienia się zakłócenia na wyjściu obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-2,612	166,9 [%]	-	7077,5	29072	$8,0611 \cdot 10^6$	$1,3538 \cdot 10^{10}$

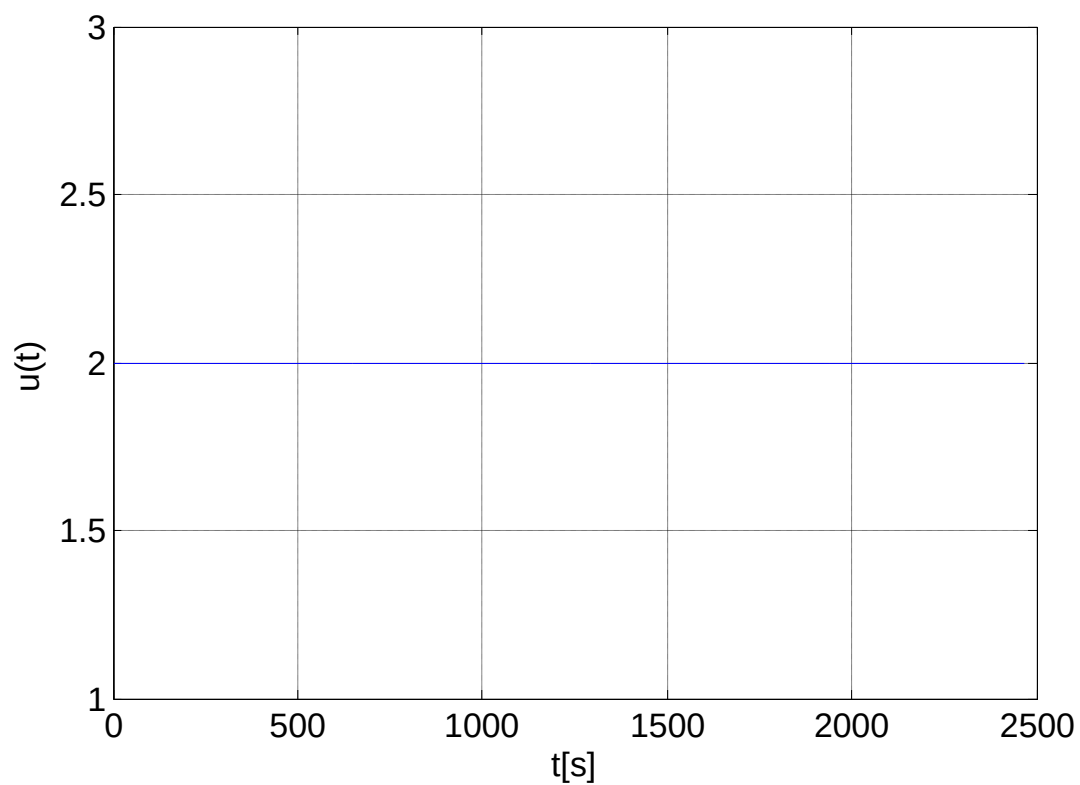
Rysunek 8.34 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.35 przebieg uchybu w procesie regulacji wynikającym ze zmiany parametrów obiektu regulacji. Tabela 8.14 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.36 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.34. Odpowiedź układu na zmianę parametrów obiektu regulacji.



Rysunek 8.35. Przebieg uchybu regulacji w procesie wynikającym ze zmiany parametrów obiektu regulacji.



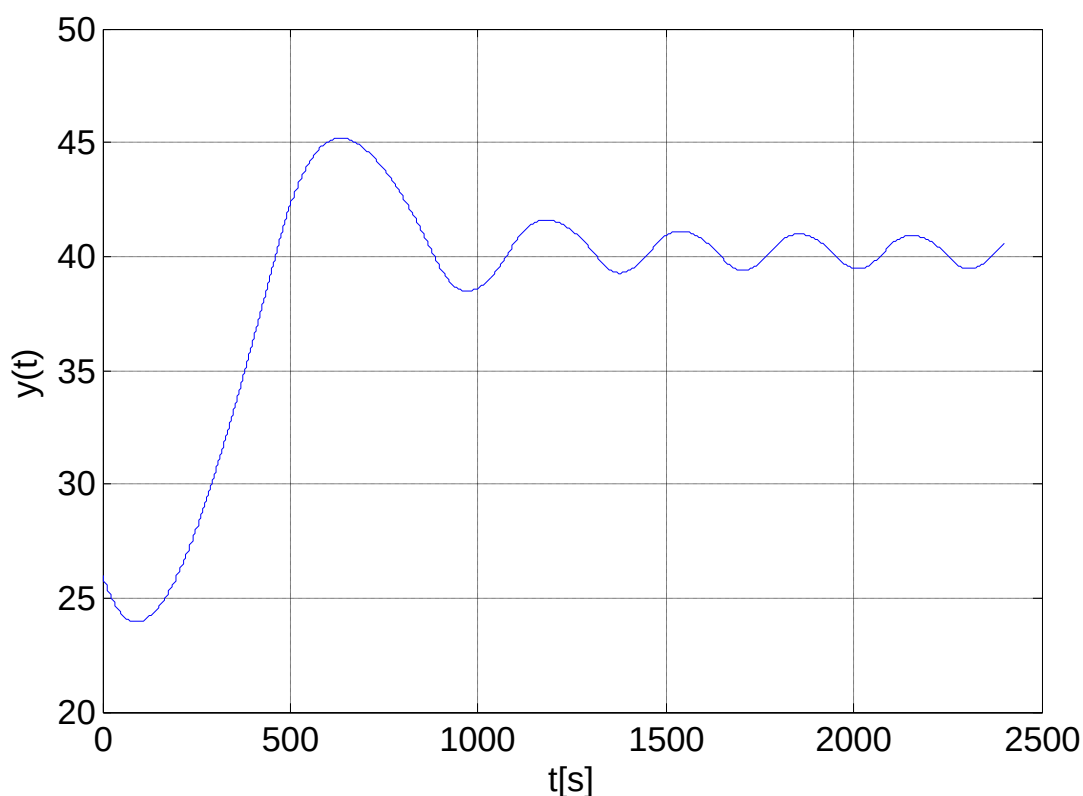
Rysunek 8.36. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

Tabela 8.14. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany parametrów obiektu regulacji.

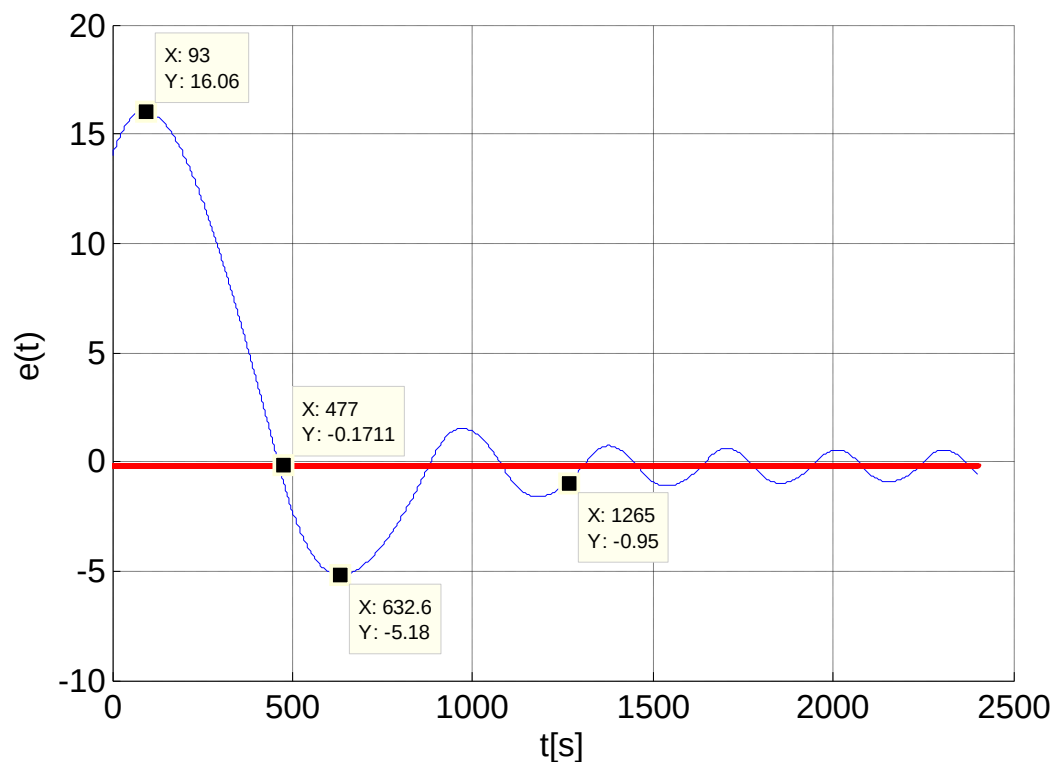
e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-	-	-	-	-	-	-

Obiekt typu inercja II rzędu
$$K(s) = \frac{k}{(1+sT_1)(1+sT_2)}$$

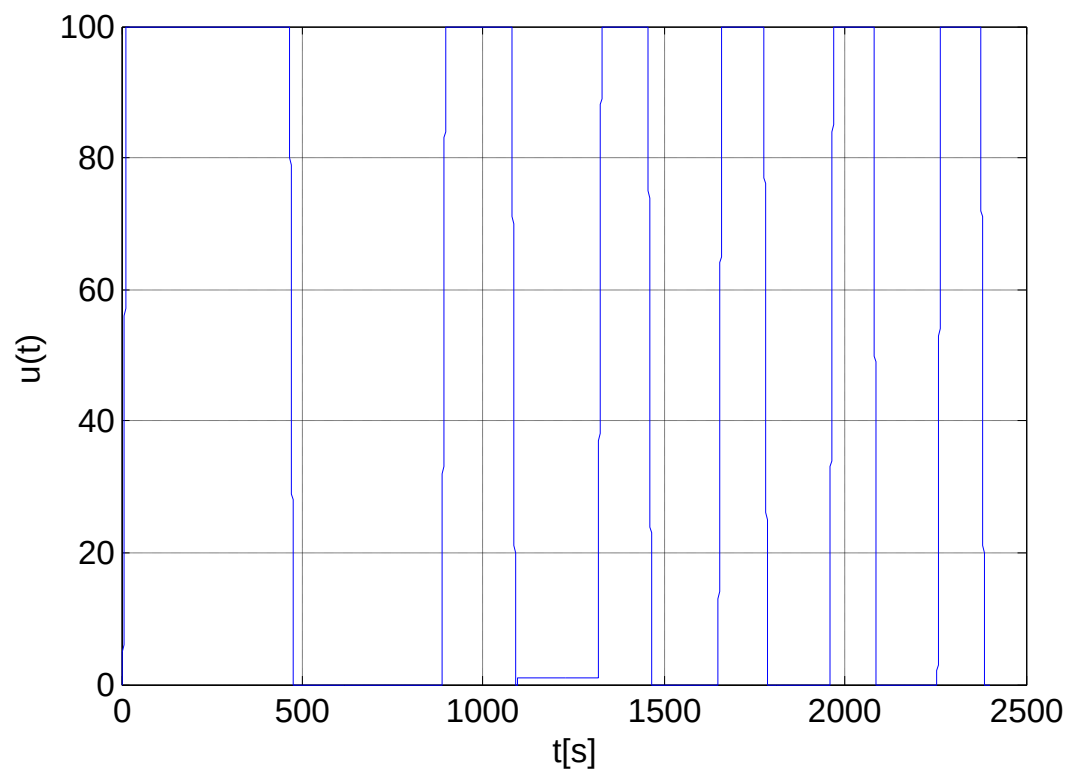
Rysunek 8.37 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.38 przebieg uchybu w procesie regulacji wynikającym ze zmiany wartości zadanej. Tabela 8.15 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.39 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.37. Odpowiedź układu na zmianę wartości zadanej.



Rysunek 8.38. Przebieg uchybu regulacji w procesie wynikającym ze zmiany wartości zadanej.

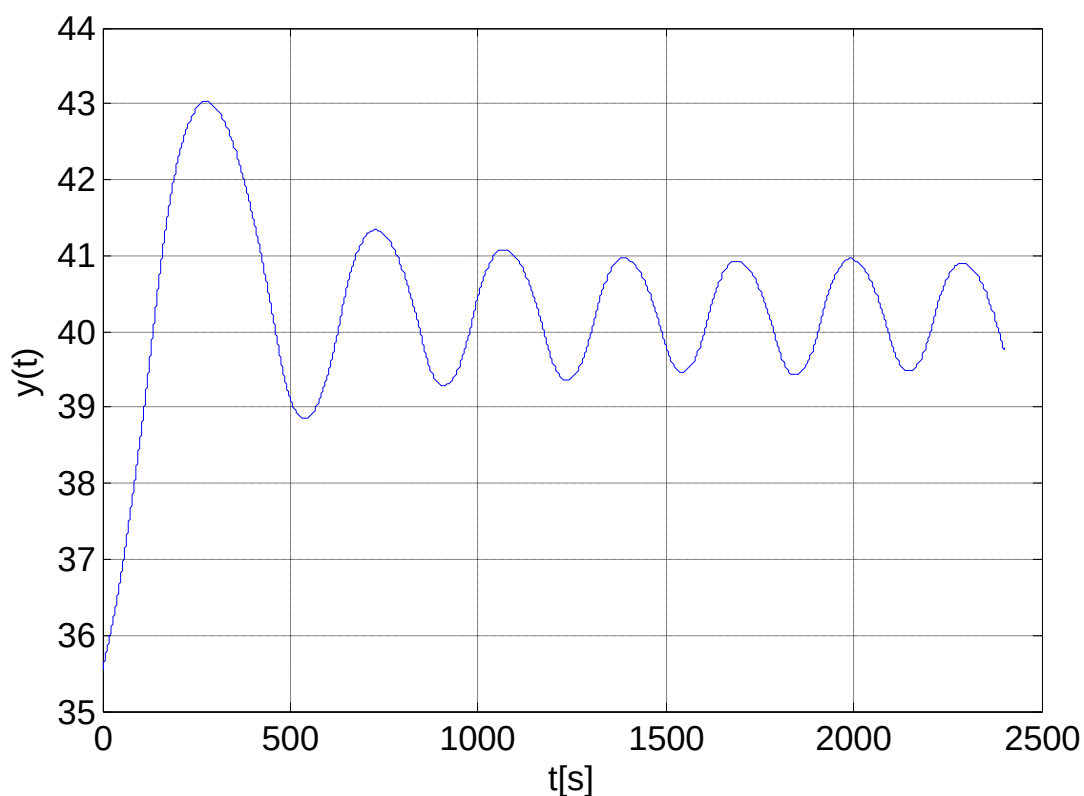


Rysunek 8.39. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

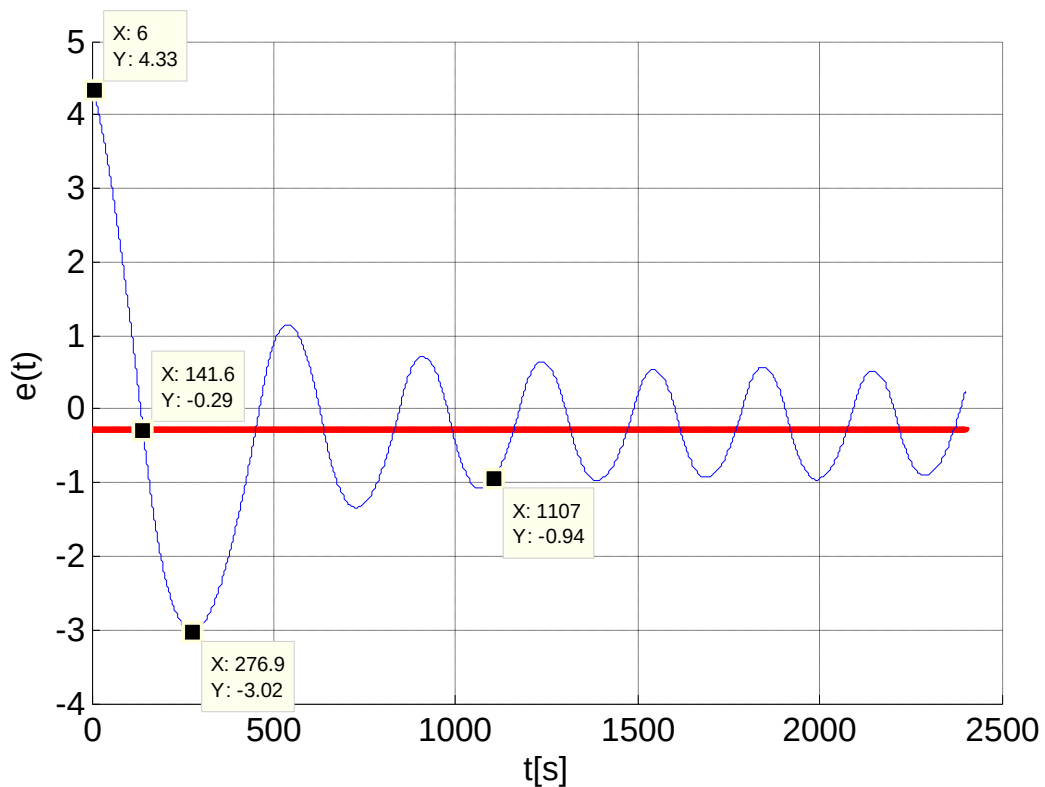
Tabela 8.15. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany wartości zadanej.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,1711	30,86 [%]	1265 [s]	7511,8	7354,4	$3,3706 \cdot 10^6$	$3,3707 \cdot 10^9$

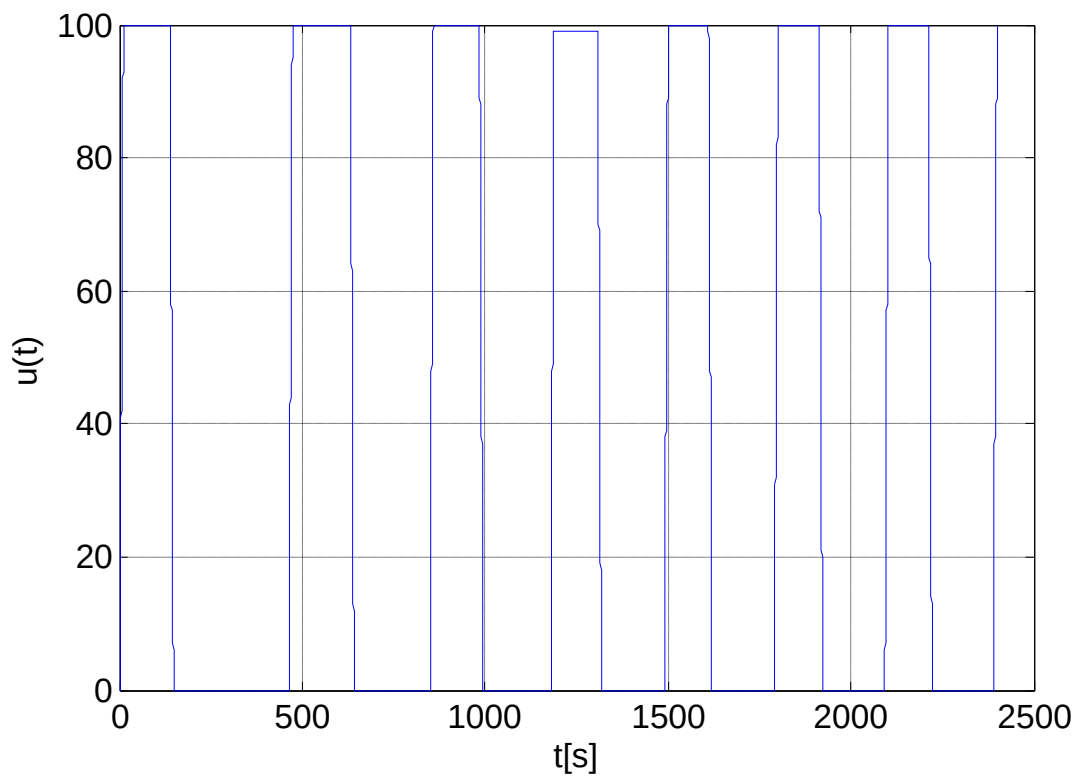
Rysunek 8.40 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.41 przebieg uchybu w procesie regulacji wynikającym z pojawienia się zakłócenia na wyjściu obiektu. Tabela 8.16 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.42 przedstawia przebieg algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.40. Odpowiedź układu na pojawienie się zakłócenia na wyjściu obiektu regulacji.



Rysunek 8.41. Przebieg uchybu regulacji w procesie wynikającym z pojawienia się zakłócenia na wyjściu obiektu regulacji.

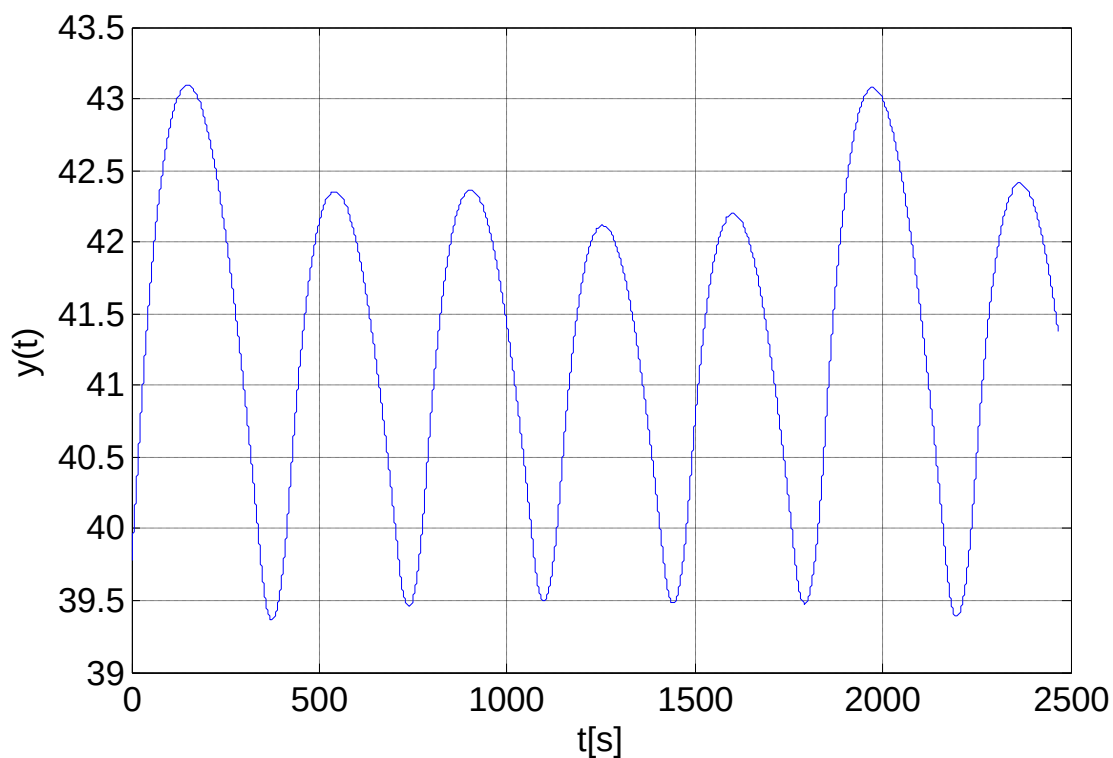


Rysunek 8.42. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

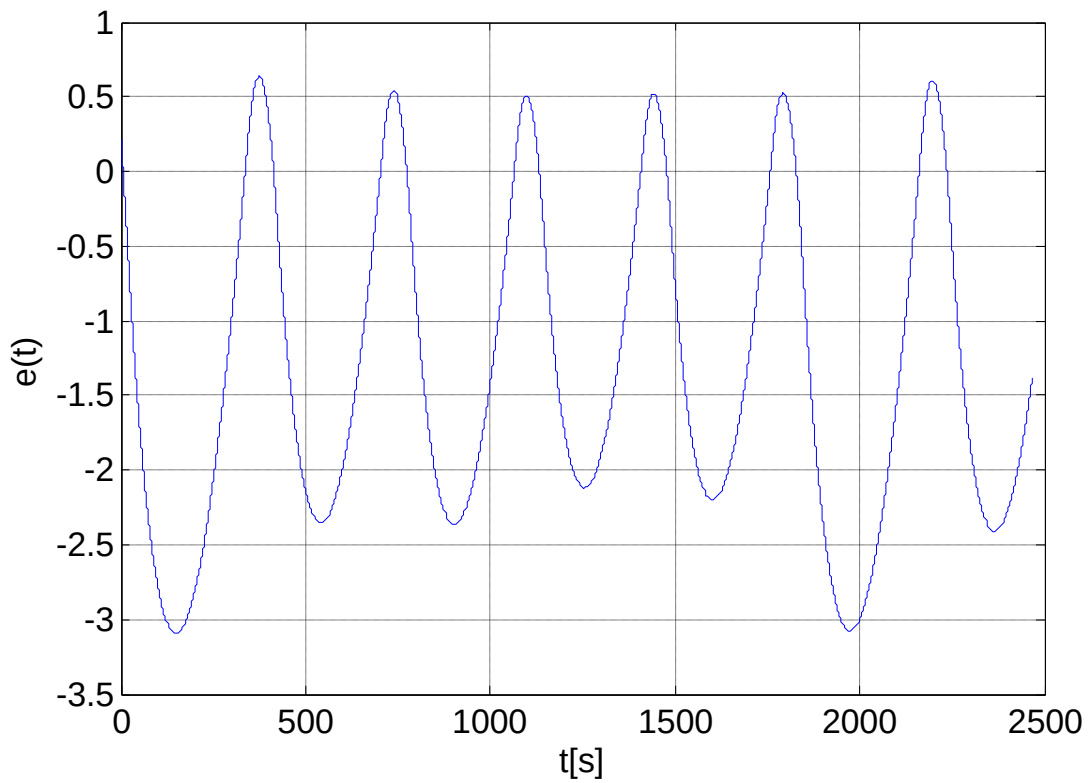
Tabela 8.16. Wskaźniki jakości opisujące proces regulacji wynikający z pojawienia się zakłócenia na wyjściu obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,29	59,09 [%]	1107 [s]	2150	3543,7	$1,7656 \cdot 10^6$	$2,5355 \cdot 10^9$

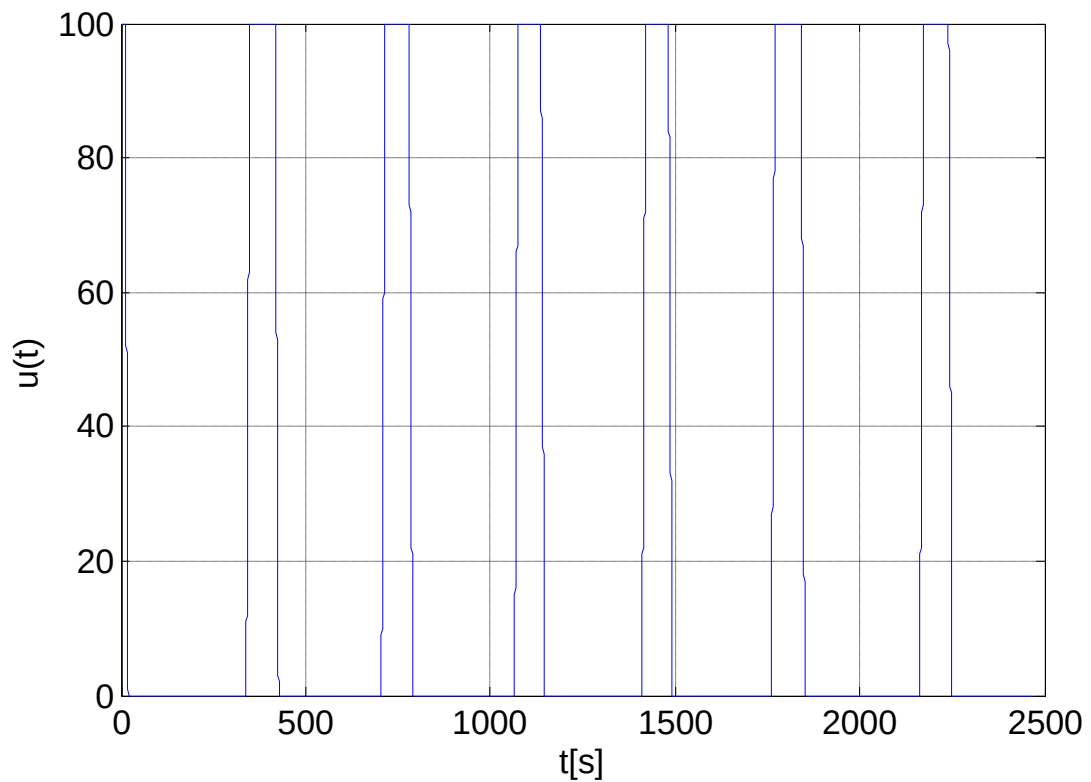
Rysunek 8.43 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.44 przebieg uchybu w procesie regulacji wynikającym ze zmiany parametrów obiektu regulacji. Tabela 8.17 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.45 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.43. Odpowiedź układu na zmianę parametrów obiektu regulacji.



Rysunek 8.44. Przebieg uchybu regulacji w procesie wynikającym ze zmiany parametrów obiektu regulacji.



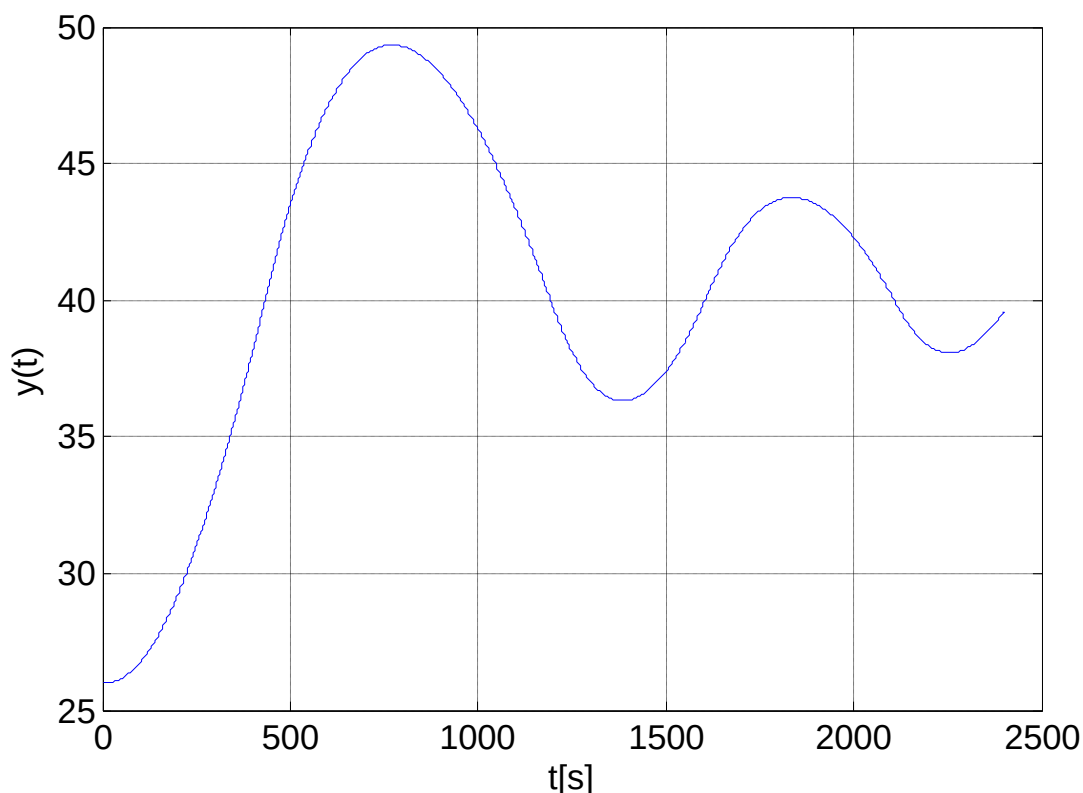
Rysunek 8.45. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

Tabela 8.17. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany parametrów obiektu regulacji.

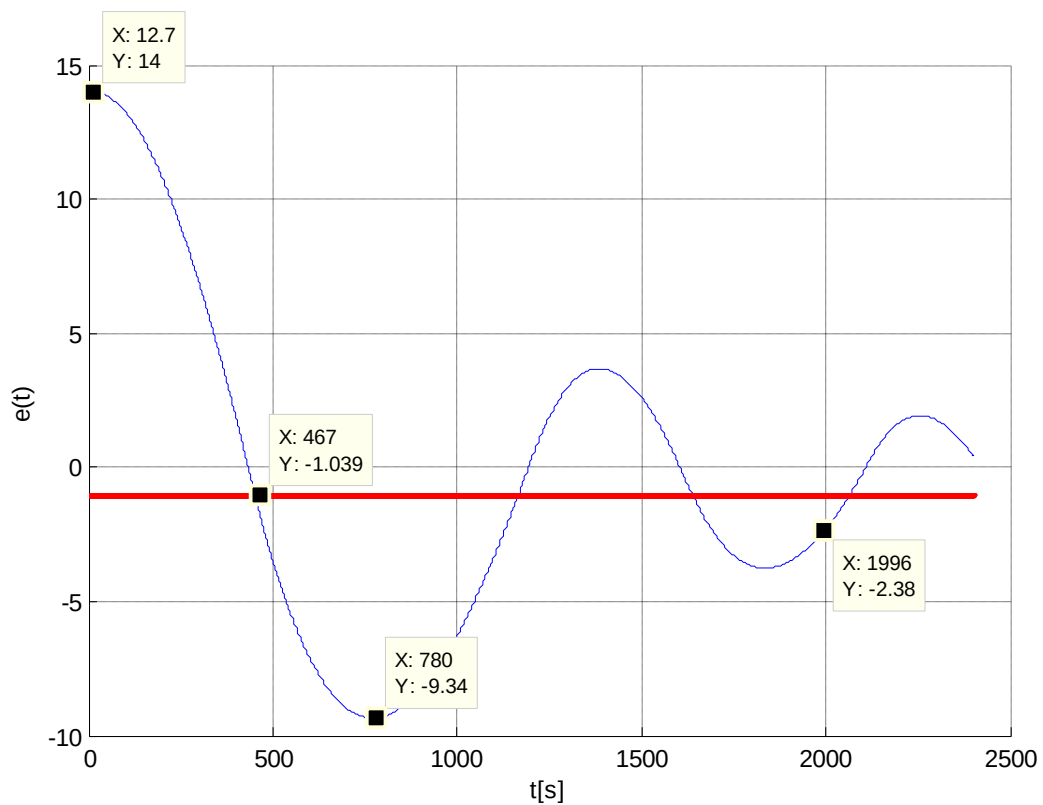
e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-	-	-	-	-	-	-

$$\text{Obiekt oscylacyjny } (K(s) = \frac{k\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2})$$

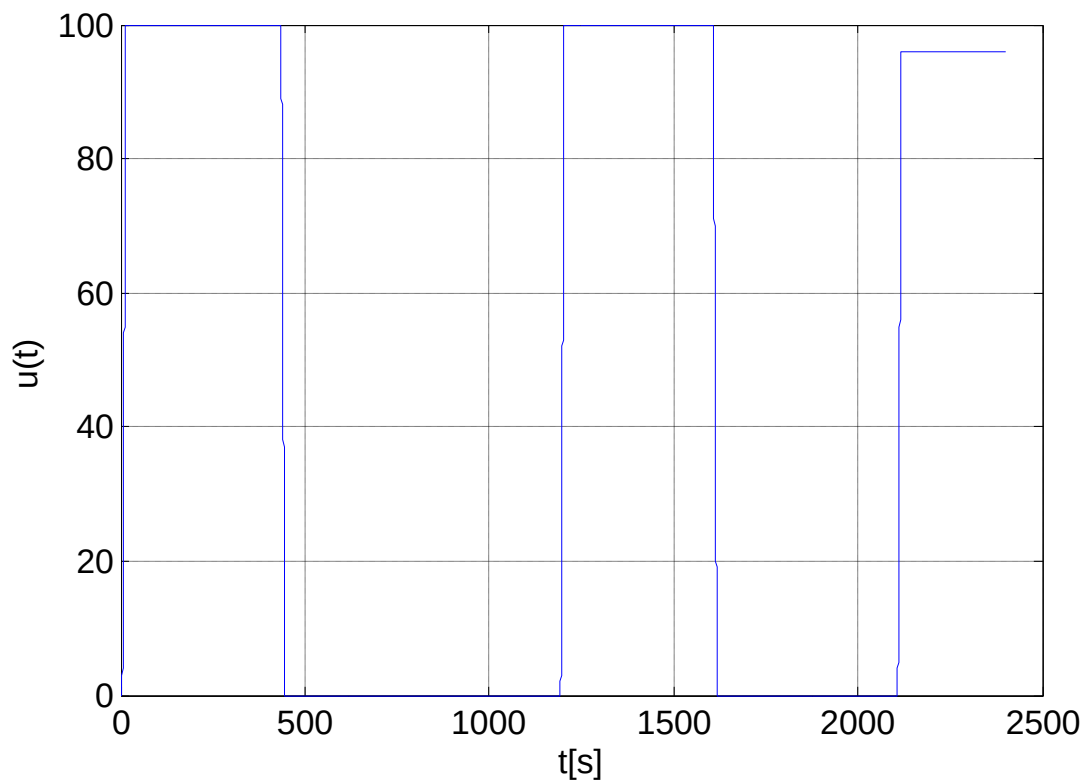
Rysunek 8.46 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.47 przebieg uchybu w procesie regulacji wynikającym ze zmiany wartości zadanej. Tabela 8.18 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.48 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.46. Odpowiedź układu na zmianę wartości zadanej.



Rysunek 8.47. Przebieg uchybu regulacji w procesie wynikającym ze zmiany wartości zadanej.

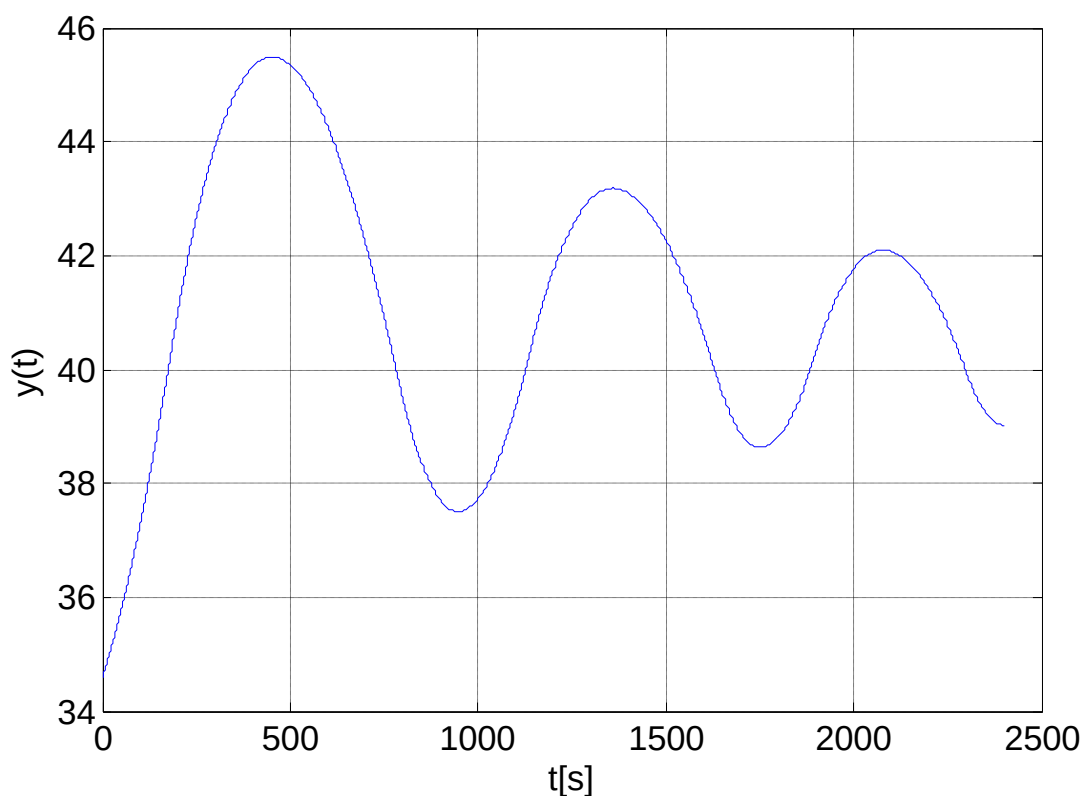


Rysunek 8.48. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

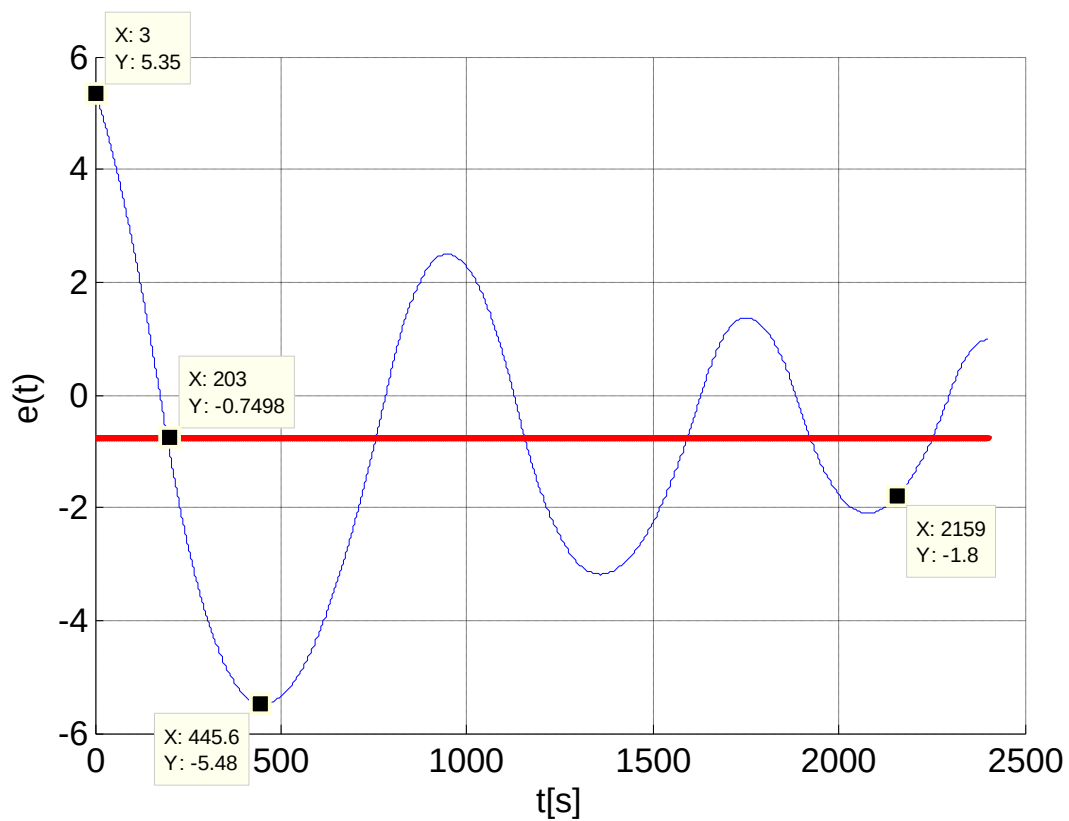
Tabela 8.18. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany wartości zadanej.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-1,039	55,2 [%]	1996 [s]	11298	86366	$8,9725 \cdot 10^6$	$1,1531 \cdot 10^{10}$

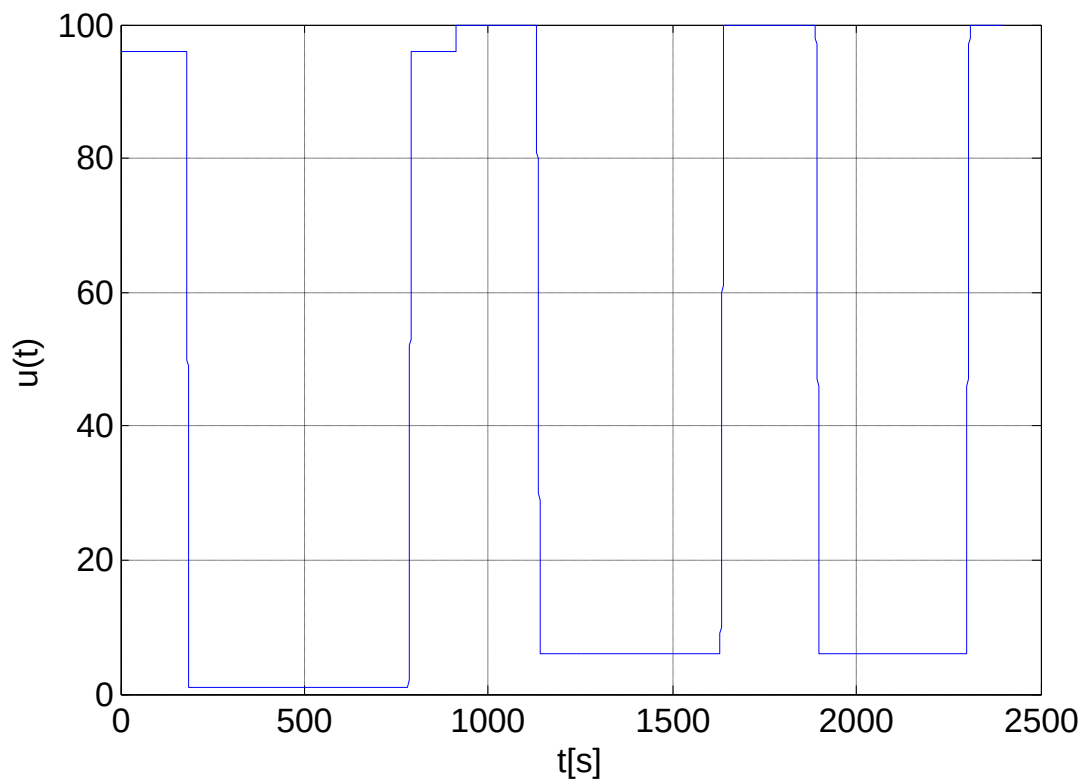
Rysunek 8.49 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.50 przebieg uchybu w procesie regulacji wynikającym z pojawienia się zakłócenia na wyjściu obiektu. Tabela 8.19 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.51 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.49. Odpowiedź układu na pojawienie się zakłócenia na wyjściu obiektu regulacji.



Rysunek 8.50. Przebieg uchybu regulacji w procesie wynikającym z pojawienia się zakłócenia na wyjściu obiektu regulacji.

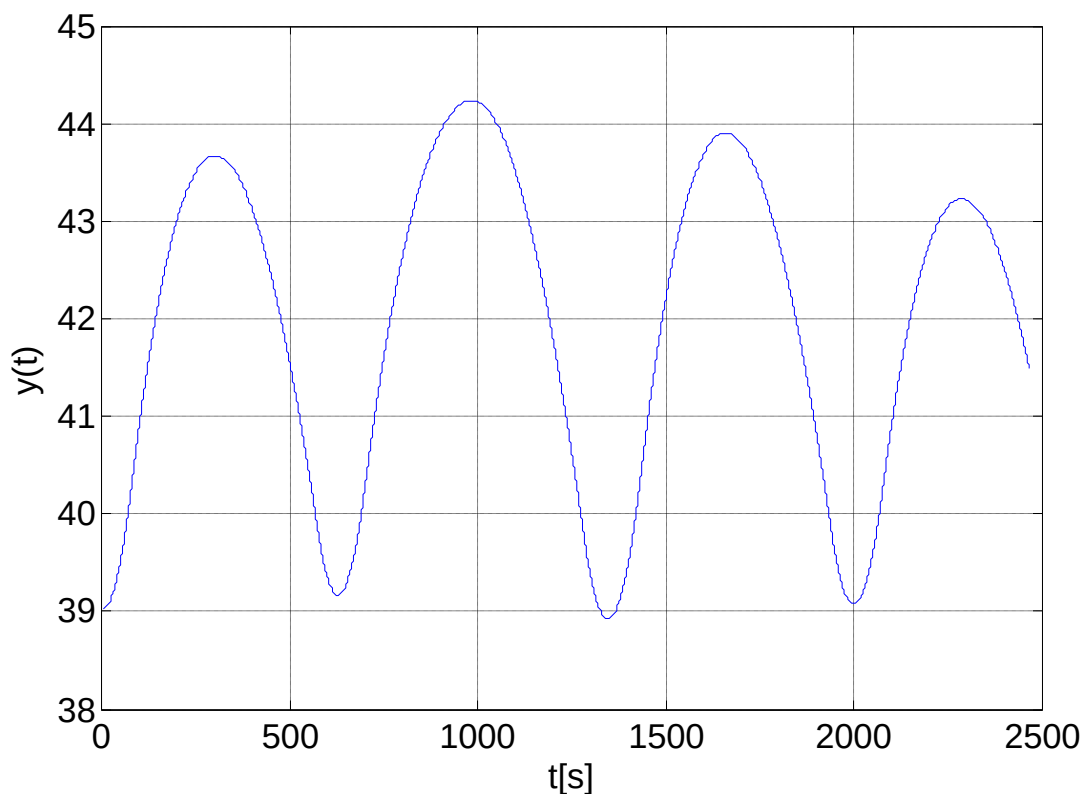


Rysunek 8.51. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

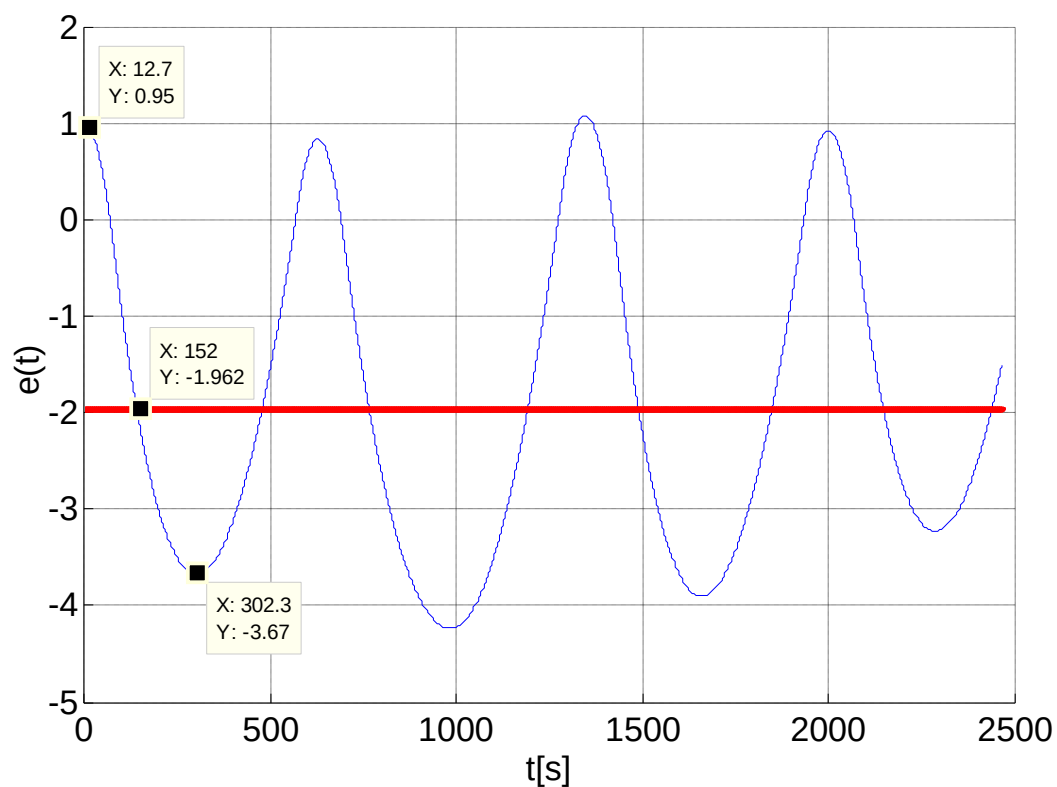
Tabela 8.19. Wskaźniki jakości opisujące proces regulacji wynikający z pojawienia się zakłócenia na wyjściu obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-0,7498	77,55 [%]	2159 [s]	5235,8	16765	$4,8282 \cdot 10^6$	$6,6512 \cdot 10^9$

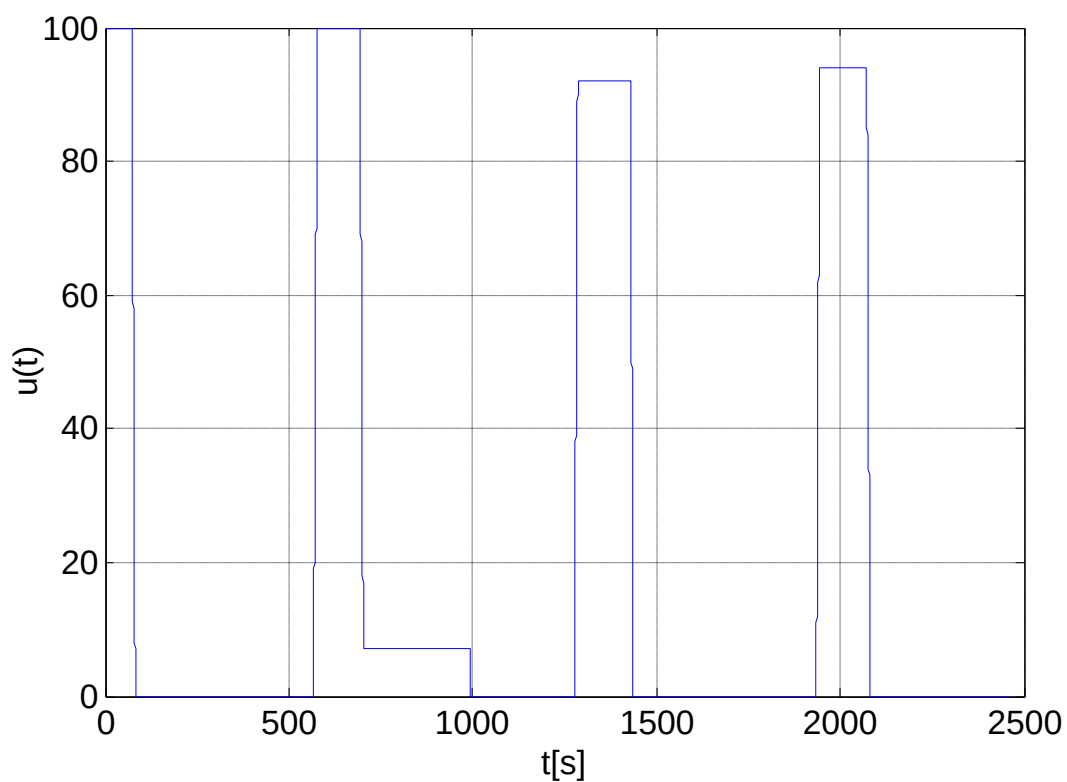
Rysunek 8.52 przedstawia odpowiedź obiektu regulacji, natomiast rysunek 8.53 przebieg uchybu w procesie regulacji wynikającym ze zmiany parametrów obiektu regulacji. Tabela 8.20 zawiera wskaźniki jakości opisujące omawiany proces regulacji. Rysunek 8.54 przedstawia przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.



Rysunek 8.52. Odpowiedź układu na zmianę parametrów obiektu regulacji.



Rysunek 8.53. Przebieg uchybu regulacji w procesie wynikającym ze zmiany parametrów obiektu regulacji.



Rysunek 8.54. Przebieg wyjścia algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

Tabela 8.20. Wskaźniki jakości opisujące proces regulacji wynikający ze zmiany parametrów obiektu regulacji.

e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
-1,962	10,97 [%]	-	5437,7	16046	$6,7304 \cdot 10^6$	$1,0986 \cdot 10^{10}$

8.3 Porównanie wyników testów

W celu określenia, który z zastosowanych układów sterowania charakteryzował się lepszym jakościowo procesem regulacji w układzie regulacji z wybranym obiektem oraz zakłóceniem dokonano porównania wyników uzyskanych w czasie testów.

Tabela 8.21 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu typu inercja z opóźnieniem z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na zmianę wartości zadanej.

Tabela 8.21. Porównanie wskaźników jakości dla procesów regulacji obiektu typu inercja z opóźnieniem w odpowiedzi na zmianę wartości zadanej.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	- 0,9328	5,08 [%]	252.2 [s]	3191,6	16208	$2,7377 \cdot 10^6$	$4,4075 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-2,812	17,87 [%]	1482 [s]	7728,2	39816	$7,6155 \cdot 10^6$	$1,1988 \cdot 10^{10}$

Porównując wskaźniki jakości zawarte w Tabeli 8.21 można stwierdzić, iż w przypadku układu regulacji z obiektem typu inercja z opóźnieniem oraz zakłóceniem w postaci zmiany wartości zadanej, regulator krokowy charakteryzuje się lepszym jakościowo procesem regulacji. Regulacja za pomocą regulatora krokowego charakteryzuje się mniejszym średnim uchybem, przeregulowaniem oraz czasem regulacji. Ponadto, wszystkie wskaźniki całkowite okazały się przyjmować mniejsze wartości dla układu regulacji z regulatorem krokowym.

Tabela 8.22 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu typu inercja z opóźnieniem z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na wprowadzenie zakłócenia na wyjściu obiektu. Porównując wskaźniki jakości zawarte w Tabeli 8.22 można stwierdzić, iż regulacja z wykorzystaniem regulatora krokowego ponownie okazała się lepsza. Wskazuje na to fakt, iż wszystkie z wybranych wskaźników jakości przyjmują mniejsze wartości niż w przypadku regulacji algorytmem regulacji krokowej z wewnętrznym sprzężeniem zwrotnym. Ponadto w przypadku wykorzystania algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym można zauważyć, iż w założonym czasie testu odpowiedź układu nie zdołała się ustabilizować.

Tabela 8.22. Porównanie wskaźników jakości dla procesów regulacji obiektu typu inercja z opóźnieniem w odpowiedzi na zakłócenie na wyjściu obiektu.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-0,982	30,30 [%]	207,5 [s]	2626,4	3841,4	$2,8848 \cdot 10^6$	$4,6880 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprężeniem zwrotnym	-2,612	166,9 [%]	-	7077,5	29072	$8,0611 \cdot 10^6$	$1,3538 \cdot 10^{10}$

Tabela 8.23 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu typu inercja z opóźnieniem z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprężeniem zwrotnym w odpowiedzi na zmianę wzmocnienia obiektu regulacji.

Tabela 8.23. Porównanie wskaźników jakości dla procesów regulacji obiektu typu inercja z opóźnieniem w odpowiedzi na zmianę wzmocnienia obiektu regulacji.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-0,253	2,01 [%]	583,7 [s]	1114,3	944,6	$8,3351 \cdot 10^5$	$1,1699 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-	-	-	-	-	-	-

Porównując wskaźniki jakości zawarte w Tabeli 8.23 oraz przebiegi sterowań oraz odpowiedzi układów zawarte w rozdziałach 8.1 oraz 8.2, można stwierdzić, iż układ regulacji z algorytmem regulacji krokowej z wewnętrznym sprzężeniem zwrotnym nie zareagował na zmianę wzmocnienia obiektu. Przyczyn tego zjawiska można upatrywać się w fakcie, iż symulowane otwarcie zaworu może po pewnym czasie regulacji przestać pokrywać się z rzeczywistym otwarciem zaworu. Ponadto, w przypadku regulatora krokowego, regulacja charakteryzuje się wskaźnikami jakości na wysokim poziomie. Małe wartości średniego uchybu, przeregulowania oraz krótki czas regulacji (na poziomie jednej stałej czasowej obiektu) pozwalają stwierdzić, iż regulator krokowy dobrze poradził sobie ze zmianą wzmocnienia obiektu regulacji.

Tabela 8.24 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu typu inercja drugiego rzędu z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na zmianę wartości zadanej.

Tabela 8.24. Porównanie wskaźników jakości dla procesów regulacji obiektu typu inercja drugiego rzędu w odpowiedzi na zmianę wartości zadanej.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-0,473	26,58 [%]	1189 [s]	7290,6	73325	$3,1898 \cdot 10^6$	$3,3091 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-0,171	30,86 [%]	1265 [s]	7511,8	73544	$3,3706 \cdot 10^6$	$3,3707 \cdot 10^9$

Porównując wskaźniki jakości zawarte w Tabeli 8.24 można stwierdzić, iż układ regulacji z regulatorem krokowym charakteryzuje się mniejszym przeregulowaniem, czasem regulacji oraz wskaźnikami całkowymi I_1 , I_2 , I_3 oraz I_4 , natomiast algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym charakteryzuje się mniejszym średnim uchybem.

Tabela 8.25 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu typu inercja drugiego rzędu z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na wprowadzenie zakłócenia na wyjściu obiektu.

Tabela 8.25. Porównanie wskaźników jakości dla procesów regulacji obiektu typu inercja drugiego rzędu w odpowiedzi na zakłócenie na wyjściu obiektu.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-0,387	54,02 [%]	791,8 [s]	2358,5	5242,9	$1,728 \cdot 10^6$	$2,4452 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-0,29	59,09 [%]	1107 [s]	2150	3543,7	$1,7656 \cdot 10^6$	$2,5355 \cdot 10^9$

Porównując wskaźniki jakości zawarte w Tabeli 8.25 można stwierdzić, iż regulacja z wykorzystaniem algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym charakteryzuje się mniejszym średnim uchybem oraz wskaźnikami całkowymi I_1 oraz I_2 . Ponadto, można stwierdzić, iż przeregulowania w obu przypadkach są na podobnym poziomie. Czas regulacji zdecydowanie przemawia za wyborem regulatora krokowego.

Tabela 8.26 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu typu inercja drugiego rzędu z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na zmianę wzmocnienia obiektu regulacji.

Tabela 8.26. Porównanie wskaźników jakości dla procesów regulacji obiektu typu inercja drugiego rzędu w odpowiedzi na zmianę wzmocnienia obiektu regulacji.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-1,367	23,34 [%]	1043 [s]	6410,1	34229	4,8818*10 ⁶	6.6658*10 ⁹
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-	-	-	-	-	-	-

Wskaźniki jakości zawarte w tabeli 8.26 zdecydowanie przemawiają za wyborem regulatora krokowego. Zmiana wzmocnienia obiektu w przypadku układu regulacji z algorytmem regulacji krokowej z wewnętrznym sprzężeniem zwrotnym spowodowała, iż układ znalazł się w stanie, który przypomina stan granicy stabilności lub wręcz destabilizacji.

Tabela 8.27 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu oscylacyjnego z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na zmianę wartości zadanej.

Tabela 8.27. Porównanie wskaźników jakości dla procesów regulacji obiektu oscylacyjnego w odpowiedzi na zmianę wartości zadanej.

Wskaźnik jakości Alorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-0,582	53,47 [%]	1833 [s]	9576,5	74404	$6,3915 \cdot 10^6$	$7,242 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprężeniem zwrotnym	-1,039	55,2 [%]	1996 [s]	11298	86366	$8,9725 \cdot 10^6$	$1,1531 \cdot 10^{10}$

Porównując wskaźniki jakości zawarte w Tabeli 8.27 można stwierdzić, iż wszystkie wskaźniki jakości przyjmują mniejsze wartości dla układu regulacji z wykorzystaniem regulatora krokowego. Jedynie wartość przeregulowania w obu przypadkach jest niemal taka sama. Wskazuje to na zdecydowanie lepszą jakość regulacji w układzie z regulatorem krokowym.

Tabela 8.28 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu oscylacyjnego z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na wprowadzenie zakłócenia na wyjściu obiektu. Porównując wskaźniki jakości zawarte w Tabeli 8.28 można stwierdzić, iż wszystkie wskaźniki jakości przyjmują mniejsze wartości dla układu regulacji z wykorzystaniem regulatora krokowego. Wskazuje to na zdecydowanie lepszą jakość regulacji w układzie z regulatorem krokowym.

Tabela 8.28. Porównanie wskaźników jakości dla procesów regulacji obiektu oscylacyjnego w odpowiedzi na zakłócenie na wyjściu obiektu.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-0,531	66,08 [%]	1321 [s]	3589,4	10318	$2,6001 \cdot 10^6$	$3,2929 \cdot 10^9$
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-0,749	77,55 [%]	2159 [s]	5235,8	16765	$4,8282 \cdot 10^6$	$6,6512 \cdot 10^9$

Tabela 8.29 zawiera porównanie wskaźników jakości charakteryzujących proces regulacji obiektu oscylacyjnego z wykorzystaniem regulatora krokowego oraz algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym w odpowiedzi na zmianę parametrów obiektu regulacji.

Tabela 8.29. Porównanie wskaźników jakości dla procesów regulacji obiektu oscylacyjnego w odpowiedzi na zmianę wzmocnienia obiektu regulacji.

Wskaźnik jakości Algorytm regulacji	e_{sr}	σ	t_r	I_1	I_2	I_3	I_4
Regulator krokowy	-	-	-	-	-	-	-
Algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym	-1,962	10,97 [%]	-	5437,7	16046	$6,7304 \cdot 10^6$	$1,0986 \cdot 10^{10}$

W przypadku układu sterowania z regulatorem krokowym zmiana wzmocnienia obiektu regulacji doprowadziła do destabilizacji układu. W przypadku układu regulacji z algorytmem regulacji krokowej odpowiedź układu nie ustaliła się w założonym czasie eksperymentu. Można stwierdzić, iż żaden z badanych algorytmów sterowania nie sprostął zakłóceniu w postaci zmiany wzmocnienia obiektu regulacji.

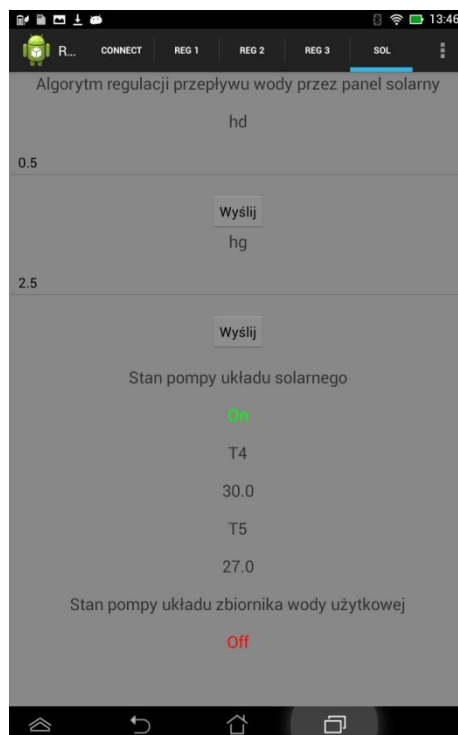
Na podstawie porównania wyników testów opisanych w rozdziałach 8.1 oraz 8.2 można wywnioskować, iż w zdecydowanej większości przypadków układ regulacji z regulatorem krokowym lepiej sterował obiektem regulacji, niż algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym. Należy jednak dodać, iż w zależności od przyjętego kryterium może wystąpić sytuacja, w której wybór algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym skutkować będzie lepszymi wynikami z punktu widzenia obranego kryterium. Ponadto, przeprowadzone testy wykazały jedną z wad algorytmu regulacji krokowej z wewnętrznym sprzężeniem zwrotnym, którą okazała się sytuacja rozbieżności symulowanego otwarcia zaworu z otwarciem rzeczywistym.

8.4 Test algorytmu sterowania układu solarnego oraz algorytmu sterowania pracą pompy w zbiorniku wody użytkowej

W celu przetestowania algorytmu sterowania układu solarnego połączono sterownik urządzeń pomiarowych z symulatorem obiektu rzeczywistego, a następnie ustalono granice histerezy różnicy temperatur na panelu słonecznym oraz zbiorniku wody użytkowej na wartość dolną 0,5 °C oraz wartość górną 2,5 °C (histereza załączania pompy w układzie V opisanym w rozdziale 3). Oznacza to, że jeśli różnica temperatury wody znajdującej się w panelu solarnym oraz temperatury wody w zbiorniku wody użytkowej przekroczy 2,5 °C pompa zostanie załączona, natomiast jeśli spadnie poniżej 0,5 °C zostanie wyłączona. W przypadku, gdy różnica temperatur znajduje się pomiędzy 0,5 °C a 2,5 °C stan pracy pompy zależy od poprzedniego stanu pracy, w jakim znajdowała się pompa. Rysunek 8.55 przedstawia konfigurację histerezy pracy pompy w sterowniku urządzeń pomiarowych. Następnie, w symulatorze obiektu rzeczywistego wpisywano wartości temperatury wody znajdującej się w panelu solarnym oraz w zbiorniku wody użytkowej i notowano stan pracy pompy wymuszany przez sterownik urządzeń pomiarowych. Tabela 8.30 przedstawia wyniki testu.

Tabela 8.30. Wyniki testu algorytmu sterowania układu panelu solarnego.

Temperatura wody w panelu solarnym	Temperatura wody w zbiorniku wody użytkowej	Różnica temperatur	Poprzedni stan pracy pompy	Aktualny stan pracy pompy
35 °C	30 °C	5 °C	-	ON
32 °C	30 °C	2 °C	ON	ON
30,2 °C	30 °C	0,2 °C	ON	OFF
30,2 °C	29 °C	1,2 °C	OFF	OFF



Rysunek 8.55. Konfiguracja histerezy pracy pompy znajdującej się w układzie panelu solarnego w sterowniku urządzeń pomiarowych.

Analizując wyniki testu algorytmu sterowania układu panelu solarnego zawarte w tabeli 8.30 można stwierdzić poprawność jego działania. Układ sterowany jest zgodnie z założoną histerezą różnicy temperatur.

W celu sprawdzenia poprawności działania algorytmu sterowania pracą pompy znajdującej się w zbiorniku wody użytkowej, w symulatorze obiektu rzeczywistego symulowano wskazania czujnika przepływu (czujnik przepływu w pętli regulacyjnej IV opisanej w rozdziale 3). Po zasymulowaniu wskazania przepływu algorytm sterowania załączał pompę znajdującą się w zbiorniku wody użytkowej, znów po zasymulowaniu wskazania braku przepływu algorytm wyłączał pompę. Można więc stwierdzić, iż algorytm działa poprawnie.

9. Podsumowanie

Niniejsza praca miała na celu ukazanie możliwości, jakie niosą ze sobą współczesne urządzenia mobilne w kontekście układów automatyki. Realizacja projektu przebiegła zgodnie z założonym planem. Spełniono tym samym wszystkie

zaplanowane funkcjonalności i wymagania. W wyniku przeprowadzonych prac powstał sterownik, który z powodzeniem może zostać wykorzystany w rzeczywistych układach automatyki. Sterownik ten jest idealnym przykładem tego, jak niewielkim kosztem można zbudować urządzenie mogące znaleźć zastosowanie w codziennym życiu.

Projekt sterownika urządzeń pomiarowych składał się z kilku etapów. Pierwszym z nich był dobór warstwy sprzętowej oraz skonfigurowanie jej elementów w celu poprawnej współpracy. Warstwa sprzętowa składała się z tabletu z systemem operacyjnym Android oraz konwertera Bluetooth/RS-232. Możliwości obliczeniowe tabletu wystarczyły w zupełności na implementację algorytmów sterowania oraz interfejsu graficznego użytkownika. Konwerter odpowiedzialny był za komunikację z modułem regulatora kotłowni, a jego wybór podyktowany był założeniem o bezprzewodowej komunikacji zapewniającej mobilność sterownika. Interfejs komunikacyjny Bluetooth okazał się rozwiązaniem zapewniającym stabilność połączenia na wysokim poziomie oraz szybkość transmisji wystarczającą dla wymagań sterownika urządzeń pomiarowych.

Kolejnym etapem projektu był dobór narzędzi programistycznych oraz zaprogramowanie warstwy programowej sterownika urządzeń pomiarowych. W czasie realizacji projektu korzystano z darmowego zintegrowanego narzędzia programistycznego "Eclipse", który pozwala między innymi na programowanie urządzeń mobilnych z systemem operacyjnym Android oraz aplikacji w języku obiektowym Java. Podczas tworzenia programu sterownika urządzeń pomiarowych kierowano się ideą modułowości oprogramowania. Zakłada ona, iż poszczególne zadania są wykonywane w osobnych jednostkach programowych, co w znaczący sposób ułatwia rozwijanie oraz utrzymanie kodu. Podejście to pozwoliło na łatwą detekcję błędów oraz dało możliwość szybkiego wprowadzania zmian wynikających z rozwijającego się nieustannie kodu. Wszelkie informacje na temat programowania urządzeń mobilnych zaczerpnięto z sieci internetowej, w której znajduje się wiele poradników poświęconych temu tematowi. Realizacja projektu pozwoliła na poszerzenie horyzontów w dziedzinie szeroko pojętego programowania oraz zaowocowała powstaniem dwóch wersji programu sterownika urządzeń pomiarowych, w których realizowane są dwa różne algorytmy

sterownia. Ponadto, w wyniku potrzeby przetestowania sterownika urządzeń pomiarowych powstał symulator obiektu rzeczywistego napisany w języku obiektowym Java.

Jednym z ważniejszych etapów projektu okazało się usystematyzowanie wiedzy teoretycznej na podstawie, której zaimplementowane zostały wybrane algorytmy sterowania oraz symulator obiektu rzeczywistego. Przegląd literatury pozwolił na wybór optymalnych algorytmów sterowania oraz umożliwił komputerową implementację obiektów dynamicznych zrealizowaną na komputerze klasy PC. Bez podstaw teoretycznych niemożliwe byłoby przeprowadzenie testów dostarczających danych na temat powstałego sterownika oraz jakości jego sterowania.

Ostatnim etapem projektu było przeprowadzenie testów w wyniku, których uzyskano dane na temat procesu regulacji za pomocą sterownika urządzeń pomiarowych. Testy dotyczyły trzech różnych typów obiektów regulacji, a także trzech różnych zjawisk, które powodowały powstanie uchybu regulacji. Dane zebrane w trakcie doświadczeń dały szeroki pogląd na jakość regulacji za pomocą wybranych algorytmów sterowania. Pozwoliły również na stwierdzenie, iż regulator krokowy przewyższa jakościowo algorytm regulacji krokowej z wewnętrznym sprzężeniem zwrotnym.

10. Kierunki dalszego rozwoju projektu

Jednym z głównych kierunków rozwoju, który warto rozważyć jest połączenie dwóch aktualnie funkcjonujących wersji oprogramowania sterownika w integralną całość umożliwiającą użytkownikowi wybór algorytmu sterowania na poszczególnych kanałach regulacyjnych. Dodatkowo zaimplementowanie kolejnych bardziej rozbudowanych algorytmów sterownia, pozwoliłoby na rozszerzenie możliwości sterownia. Ponadto, istnieje możliwość rozwoju w kierunku archiwizowania danych dotyczących procesu regulacji w pamięci sterownika oraz ich prezentacji w formie wykresów na jednym z dodatkowych paneli graficznych. Kolejnym kierunkiem zmian wartym przemyślenia jest przekazywanie danych za pośrednictwem sieci internetowej do zewnętrznego

serwera. Takie rozwiązanie umożliwiłoby podgląd stanu pracy regulowanego procesu, a także stworzyło możliwość zdalnego wpływania na proces poprzez zmianę jego parametrów, z każdego miejsca na ziemi. Dalszy rozwój projektu wymaga przeprowadzenia testów jakości sterowania obiektu rzeczywistego. Zastosowanie takiego podejścia umożliwi analizę pozwalającą na dopracowanie sterownika w kontekście pracy z układami rzeczywistymi.

Powyższe drogi rozwoju projektu stanowią tylko część pomysłów, jakie można zrealizować na już powstałym urządzeniu. Zastosowanie tabletu z systemem operacyjnym Android daje niskończone możliwości, co czyni opracowany sterownik urządzeń pomiarowych projektem silnie rozwojowym.

Bibliografia

- [1] http://pl.wikipedia.org/wiki/Historia_automatyki
- [2] www.techopedia.com/definition/4619/personal-digital-assistant-pda
- [3] W. Grajdek. *Moduł regulatora kotłowni*. Praca dyplomowa inżynierska, Gliwice, 2013.
- [4] J. Wiora. *Heat control in a detached house*. Institute of Automatic Control, Gliwice, 2010.
- [5] <http://www.britannica.com/EBchecked/topic/725975/PDA>
- [6] http://en.wikipedia.org/wiki/Personal_digital_assistant#Touch_screen
- [7] <http://www.webopedia.com/TERM/P/PDA.html>
- [8] <http://tabletoid.pl/poradnik/system-operacyjny-w-tablecie>
- [9] <http://www.developer.android.com>
- [10] <https://developer.apple.com>
- [11] www.asus.com/pl/Tablets_Mobile/ASUS_MeMO_Pad_HD_7/specifications/
- [12] <http://ep.com.pl/files/8904.pdf>
- [13] http://www.kamami.pl/dl/btm222_datasheet.pdf
- [14] K. Sierra, B. Bates. *SCJP Sun Certified Programmer for Java 6 Study Guide*. The McGraw-Hill Companies, USA, 2008.
- [15] <http://www.eclipse.org>
- [16] <http://developer.android.com/sdk/index.html>
- [17] <http://developer.android.com/reference/android/app/Service.html>
- [18] <http://developer.android.com/reference/android/app/DialogFragment.html>
- [19] <http://developer.android.com/reference/android/os/Handler.html>
- [20] <http://developer.android.com/reference/java/util/Timer.html>
- [21] <http://developer.android.com/reference/java/util/TimerTask.html>
- [22] <http://developer.android.com/guide/topics/ui/declaring-layout.html>
- [23] J. Kuźnik. *Regulatory i układy regulacji*. Wydawnictwo Politechniki Śląskiej, Gliwice, 2006.
- [24] http://www.jcontrol.org/download/rxtx_en.html

- [25] W. Kryszicki, L. Włodarski. *Analiza matematyczna w zadaniach*. Wydawnictwo Naukowe PWN, Warszawa, 2000.
- [26] J. Klamka, Z. Ogonowski, M. Jamicki, M. Stasik. *Metody Numeryczne*. Wydawnictwo Politechniki Śląskiej, Gliwice, 1998.
- [27] D. Horla. *Podstawy Automatyki Ćwiczenia Laboratoryjne*. Wydawnictwo Politechniki Poznańskiej, Poznań 2003.
- [28] K. Amborski, A. Marusak. *Teoria sterowania w ćwiczeniach*. Państwowe Wydawnictwo Naukowe, Warszawa, 1978.
- [29] <http://lpsa.swarthmore.edu/LaplaceZTable/LaplaceZFuncTable.html>
- [30] J. Kostro. *Elementy, urządzenia i układy automatyki*. Wydawnictwa Szkolne i Pedagogiczne, Warszawa, 1986.
- [31] M. Żelazny. *Podstawy automatyki*. Państwowe Wydawnictwo Naukowe, Warszawa, 1976.
- [32] A. Urbaniak. *Podstawy automatyki*. Wydawnictwo Politechniki Poznańskiej, Poznań, 2001.

Spis załączników

Załącznik 1. Płyta CD z dwiema wersjami aplikacji sterownika urządzeń pomiarowych, aplikacją symulatora obiektu rzeczywistego oraz pracą magisterską w formacie pdf.