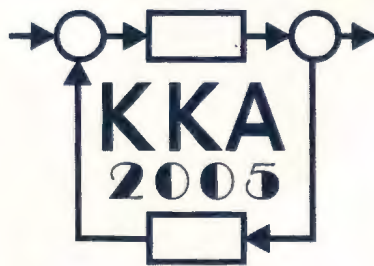


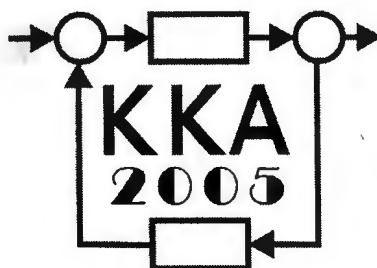
XV Krajowa Konferencja Automatyki

Tom II



**Redaktorzy:
Zdzisław Bubnicki
Roman Kulikowski
Janusz Kacprzyk**

XV Krajowa Konferencja Automatyki Tom II



Redaktorzy:
Zdzisław BUBNICKI
Roman KULIKOWSKI
Janusz KACPRZYK

ORGANIZATOR

Komitet Automatyki i Robotyki Polskiej Akademii Nauk
Instytut Badań Systemowych Polskiej Akademii Nauk

WSPÓLORGANIZATORZY

Politechnika Warszawska

Przemysłowy Instytut Automatyki i Pomiarów

Polskie Stowarzyszenie Pomiarów, Automatyki i Robotyki

ORGANIZATOR

Komitet Automatyki i Robotyki Polskiej Akademii Nauk
Instytut Badań Systemowych Polskiej Akademii Nauk

WSPÓLORGANIZATORZY

Politechnika Warszawska
Przemysłowy Instytut Automatyki i Pomiarów
Polskie Stowarzyszenie Pomiarów, Automatyki i Robotyki

KOMITET PROGRAMOWY

Przewodniczący	Zdzisław BUBNICKI
Zastępca Przewodniczącego	Roman KULIKOWSKI

CZŁONKOWIE

Stanisław BAŃKA	Michał BIAŁKO
Mikołaj BUSŁOWICZ	Władysław FINDEISEN
Ryszard GESSING	Henryk GÓRECKI
Jakub GUTENBAUM	Jerzy JÓZEFczyk
Stanisław KACZANOWSKI	Tadeusz KACZOREK
Janusz KACPRZYK	Jerzy KLAMKA
Józef KORBICZ	Zbigniew KOWALSKI
Krzysztof KOZŁOWSKI	Juliusz L. KULIKOWSKI
Krzysztof KUŹMIŃSKI	Kazimierz MALANOWSKI
Krzysztof MALINOWSKI	Wojciech MITKOWSKI
Antoni NIEDERLIŃSKI	Władysław PEŁCZEWSKI
Tadeusz PUCHAŁKA	Leszek RUTKOWSKI
Stanisław SKOCZOWSKI	Roman SŁOWIŃSKI
Jerzy ŚWIĄTEK	Andrzej ŚWIERNIAK
Ryszard TADEUSIEWICZ	Piotr TATJEWSKI
Krzysztof TCHOŃ	Leszek TRYBUS
Jan WĘGLARZ	Andrzej P. WIERZBICKI

KOMITET ORGANIZACYJNY

Przewodniczący	Roman KULIKOWSKI
Zastępcy Przewodniczącego	Janusz KACPRZYK
	Stanisław KACZANOWSKI
	Tadeusz KACZOREK
	Krzysztof MALINOWSKI
Członkowie	Roman OSTROWSKI
	Tadeusz PUCHAŁKA
	Dariusz WAGNER
Sekretarze naukowci	Jan STUDZIŃSKI
	Jan W. OWSIŃSKI

ISBN 83-89475-01-4

Copyright © Instytut Badań Systemowych Polskiej Akademii Nauk
All rights reserved

Druk: ARGRAF, Warszawa

APARATURA AUTOMATYKI

STEROWANIE MAGNETYCZNYM ZAWIESZENIEM Z WYKORZYSTANIEM SZYBKICH STEROWNIKÓW OPARTYCH NA TECHNOLOGII FPGA*

Paweł PIĄTEK

Akademia Górniczo-Hutnicza,
Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Katedra Automatyki,
Al. Mickiewicza 30, 30-059 Kraków, e-mail: ppi@ia.agh.edu.pl

Streszczenie: W niniejszej pracy przedstawiono proces tworzenia regulatora pracującego w układzie FPGA. Budowa takiego regulatora wymaga uwzględnienia specyfiki platformy sprzętowej, a w szczególności jej ograniczeń. Zbudowany regulator został zastosowany do sterowania laboratoryjnym układem magnetycznego zawieszenia, który jest obiektem wymagającym wysokich częstotliwości próbkowania, większych niż 1kHz. Artykuł przedstawia proces tworzenia i testowania regulatora dla tego obiektu. Przeprowadzono szereg eksperymentów symulacyjnych i eksperymentów na rzeczywistym obiekcie, a ich wyniki zostały zamieszczone w tej pracy.

Słowa kluczowe: Sterowanie w czasie rzeczywistym, układy FPGA, układy magnetycznego zawieszenia.

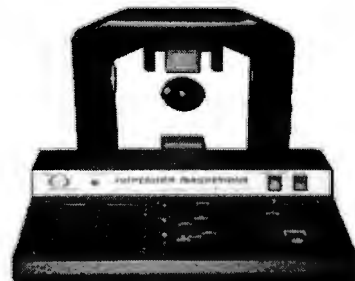
1. WSTĘP

W ostatnich latach obserwuje się dynamiczny rozwój techniki układów programowalnych FPGA. Rozwój ten jest widoczny zarówno w ciągłym powiększaniu pojemności układów, jak również w poprawianiu ich szybkości. Równolegle rozwijane jest oprogramowanie do tworzenia aplikacji. Powoduje to ciągły wzrost popularności tej techniki i tym samym spadek cen układów. Obecnie możliwe jest budowanie tanich układów sterowania opartych o stosunkowo pojemne struktury FPGA [4]. Układ taki powinien zapewniać jakość sterowania, co najmniej taką samą jak tradycyjne sterowniki oparte na systemach komputerowych przy znacznie niższej cenie. Celem niniejszej pracy jest przedstawienie sterownika układu magnetycznego zawieszenia opartego o układy FPGA. Specyfika tworzenia projektów dla układów programowalnych wymaga odpowiedniego dostosowania algorytmu regulatora. Praca przedstawia proces projektowania, budowania oraz testowania regulatora. W punktach 2 i 3 artykułu przedstawiono wykorzystany obiekt magnetycznej lewitacji oraz stosowany system sterowania. Zastosowany regulator i proces tworzenia został opisany w punkcie 4. Punkty 5 i 6 przedstawiają wykonane eksperymenty oraz ich wyniki.

2. OBIEKT REGULACJI

*Praca finansowana ze środków Komitetu Badań Naukowych w latach 2003 do 2005 jako projekt badawczy nr 4 T11A 021 25.

Obiektem regulacji wybranym do eksperymentów ze sterownikami FPGA jest układ magnetycznego zawieszenia przedstawiony na rys. 1. Urządzenie składa się ze sfery ferromagnetycznej, układu wytwarzania siły elektromagnetycznej, oraz układu pomiaru położenia sfery. Układ pomiaru położenia jest zbudowany w oparciu o element fotoelektryczny. Mierzy on ilość światła docierającego z reflektora oświetlającego sferę. Zależność położenia od ilości światła jest nieliniowa i należy to uwzględnić w budowie sterownika. Układ wykonawczy jest zbudowany z cewki oraz układu stabilizacji prądu przepływającego przez cewkę. Zmiany sterowania dokonuje się poprzez zmianę napięcia doprowadzonego do układu stabilizacji prądu.



Rys. 1. Laboratoryjny model magnetycznego zawieszenia.

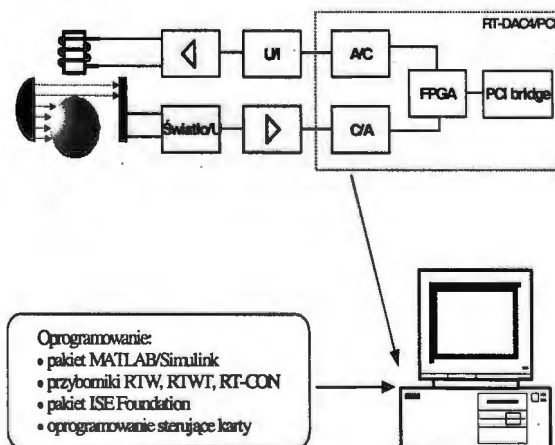
3. UKŁAD STEROWANIA

Przez klasyczny układ sterowania laboratoryjnym modelem magnetycznego zawieszenia, dla celów tego artykułu, rozumie się układ oparty na komputerze klasy PC wyposażonym w uniwersalną kartę pomiarowo-sterującą. W układzie takim sterownik jest budowany przy pomocy pakietu MATLAB/Simulink®. Zastosowanie przybownika RTW pozwala na przetworzenie algorytmu sterownika do postaci wykonywalnego programu. Wykorzystanie przybownika RTWT pozwala na uruchomienie wytworzonego wcześniej zadania sterującego z zachowaniem rygorów czasu rzeczywistego.

gramu. Wykorzystanie przybornika RTWT pozwala na uruchomienie wytworzonego wcześniej zadania sterującego z zachowaniem rygorów czasu rzeczywistego. W tym przypadku pętla sprzężenia zwrotnego jest zamknięta przez regulator obliczany w komputerze. Wadami takiego rozwiązania są: stosunkowo długie czasy wyliczania sterowania, wysokie koszty związane ze sprzętem i oprogramowaniem. Zaletami natomiast są: bardzo duża elastyczność systemu, prostota tworzenia nowych regulatorów oraz możliwość automatycznej generacji kodu.

Drugim przypadkiem rozważanym w tej pracy jest sytuacja, w której regulator jest zbudowany w strukturze układu FPGA. Zasadnicza różnica pomiędzy tymi konfiguracjami polega na tym, że w drugim przypadku pętla sprzężenia zwrotnego jest zamknięta w układzie programowalnym. A zatem przepływ danych odbywa się z pominięciem komputera. Zalety tego rozwiązania to: bardzo krótkie czasy wykonywania obliczeń i wysoka punktualność zadania sterującego. Wadą jest trudny proces tworzenia i testowania regulatora. W obu przypadkach został wykorzystany ten sam system sterowania, lecz w innych konfiguracjach. rys. 2 przedstawia schematycznie układ sterowania. Składa się on z:

- komputera PC,
- systemu inagnetycznego zawieszenia,
- karty pomiarowo-sterującej z układem FPGA,
- pakietu MATLAB/Simulink,
- przyborników RTW, RTWT, RT-CON,
- pakietu ISE Foundation,
- oprogramowania sterującego i konfiguracyjnego do karty.



W obu opisywanych przypadkach korzystano z karty pomiarowo-sterującej RT-DAC4/PCI. Karta ta zawiera układ FPGA rodziny Spartan II XC2S150 firmy Xilinx [7]. Odpowiednie mechanizmy sprzętowe oraz oprogramowanie pozwalają na zmianę konfiguracji układu FPGA bez konieczności wyłączania komputera. Dzięki temu można wielokrotnie modyfikować działanie karty, bez modyfikacji pozostałych sprzętowych elementów systemu sterującego. W szczególności nie trzeba two-

rzyć osobnych układów dopasowujących dla torów pomiarowego i sterującego. Zatem zmiana klasycznego systemu sterowania na system z regulatorem w układzie FPGA dokonuje się wyłącznie w sposób programowy i polega na zmianie konfiguracji karty. W czasie eksperymentów z klasycznym układem sterowania karta pełniła rolę zwykłej karty pomiarowo-sterującej. Podczas eksperymentów z regulatorem pracującym w strukturze FPGA komputer był wykorzystywany wyłącznie do zmian wartości zadanej oraz do zbierania danych pomiarowych.

4. REGULATOR

Proces budowania projektu dla układów FPGA jest odmienny od programowania klasycznych układów mikroprocesorowych. Dostępne języki opisu sprzętu np. wykorzystywany przez autora VHDL [3], również znacznie różnią się od typowych języków programowania komputerów. Mikroprocesory wykonują instrukcje w sposób sekwencyjny. Projekt napisany w języku opisu sprzętu jest przekształcany przez oprogramowanie do postaci automatów logicznych realizujących zaprogramowane zadanie. Wykonywanie operacji arytmetycznych, przepływ danych oraz kontrola wykonania zadanego algorytmu jest inna niż w układach mikroprocesorowych. Możliwe jest równoległe wykonywanie zadań.

Pojemność współczesnych układów FPGA cały czas rośnie, ale tanie układy nie mogą pomieścić dużych struktur logicznych. Projektując układ sterowania oparty o struktury FPGA należy uwzględnić następujące fakty:

- regulator powinien być możliwie najprostszy, ponieważ operacje arytmetyczne zajmują dużo zasobów,
- operacje arytmetyczne powinny być realizowane na liczbach stałoprzecinkowych, ponieważ operacje zmiennoprzecinkowe można realizować tylko w dużych strukturach.

Po uwzględnieniu ograniczeń sprzętu zdecydowano się na wykorzystanie w czasie eksperymentów regulatora PID. Regulator ten jest stosunkowo prosty w realizacji, a jego złożoność obliczeniowa jest nieduża. Jednocześnie ten typ regulatora jest dobrze opisany [5] w literaturze i przetestowany w praktyce.

Punktem wyjścia do rozważań był regulator opisany i przetestowany w [6].

$$u(t) = K \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (1)$$

W celu zakodowania algorytmu regulatora w języku VHDL należy w pierwszej kolejności dokonać dyskretyzacji regulatora [2], [1]. Postać dyskretnego algorytmu przyrostowego regulatora PID przedstawiono poniżej.

$$u(kT_0) = u((k-1)T_0) + q_0 \cdot e(kT_0) + q_1 \cdot e((k-1)T_0) + q_2 \cdot e((k-2)T_0) \quad (2)$$

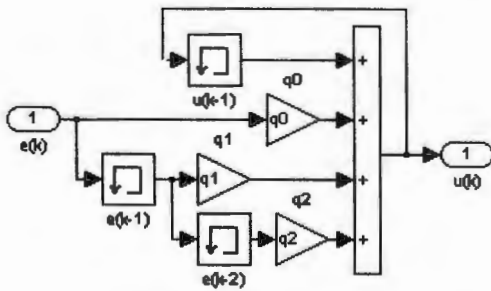
gdzie:

$$q_0 = K \left(1 + \frac{T_d}{T_0} \right),$$

$$q_1 = -K \left(1 + 2 \frac{T_d}{T_0} - \frac{T_0}{T_i} \right),$$

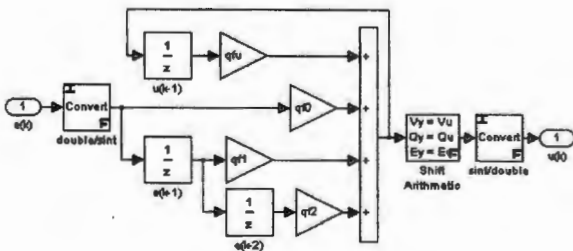
$$q_2 = K \frac{T_d}{T_0}$$

Regulator dyskretny w tej postaci nie nadaje się do bezpośredniego zastosowania w projekcie układu FPGA. Jak wcześniej wspomniano przy budowaniu projektów logiki układów programowalnych, należy stosować obliczenia na liczbach stałoprzecinkowych ze względu na ograniczoną pojemność układów. Dalszym uproszczeniem obliczeń jest rezygnacja z części ułamkowej liczb i takie przekształcenie formuły, aby możliwe było prowadzenie obliczeń na liczbach całkowitych. Wymaga to wprowadzenia odpowiedniego skalowania współczynników regulatora, sygnału wyjścia obiektu oraz sygnału sterowania. Ingerencji wymaga również sam algorytm. Należy go przekształcić tak, aby uniknąć operacji, które są trudne do zrealizowania w logice układów FPGA.



Rys. 2. Regulator dyskretny PID (przyrostowy).

Rys. 3 przedstawia dyskretny regulator PID (2) zrealizowany przy pomocy pakietu MATLAB/Simulink.



Rys. 3. Regulator PID dyskretny zbudowany z bloków „Fixed-Point”.

Rys. 4 przedstawia dyskretny regulator PID zbudowany z wykorzystaniem elementów przybornika Fixed-Point Blockset nazwany dalej zmodyfikowanym regulatorem dyskretnym. Główną różnicą pomiędzy tym regulatorem, a zwykłym regulatorem dyskretnym (2) jest sposób realizacji poszczególnych operacji arytmetycznych.

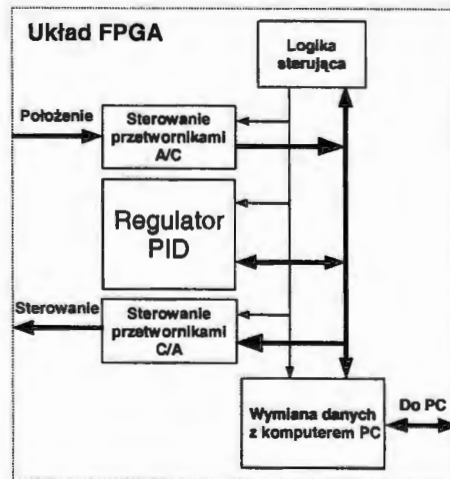
W przypadku regulatora z rys. 4 operacje te są realizowane przy pomocy liczb stałoprzecinkowych o zadanej precyzji. W przypadku regulatora wykorzystanego pod-

czas eksperymentów precyzja liczb wynosiła odpowiednio:

- dla obliczeń – 19 bitów ze znakiem,
- dla sygnałów położenia sfery oraz sygnałów sterowania – 12 bitów bez znaku.

Współczynniki regulatora zostały przeskalowane tak, aby zapewnić najlepszą dokładność obliczeń. W celu uniknięcia operacji dzielenia, do przeskalowania sygnału sterowania została wykorzystana operacja przesunięcia bitowego w prawo. Odpowiada ona dzieleniu przez odpowiednią potęgę liczby 2. Operację tą jest łatwo zrealizować w strukturze FPGA, w przeciwieństwie do operacji dzielenia.

Regulator przedstawiony na rys. 4 został wykorzystany do realizacji eksperymentów symulacyjnych mających potwierdzić jego przydatność do sterowania układem magnetycznego zawieszenia. Ta sama struktura regulatora została zapisana przy pomocy języka VHDL. Po dodaniu odpowiednich bloków sterujących pracą przetworników analogowo-cyfrowych i cyfrowo-analogowych oraz dodaniu odpowiedniej logiki sterującej, regulator ten został zaprogramowany w układzie FPGA.



Rys. 4. System sterowania zrealizowany w układzie FPGA.

Rys. 5 przedstawia strukturę logiki układu FPGA wykorzystywaną podczas badań. Logika została tak zaprojektowana, aby można było przy jej pomocy przeprowadzić eksperymenty na rzeczywistym obiekcie oraz eksperymenty typu Hardware In The Loop.

5. EKSPERYMENTY

Eksperymenty przeprowadzone w czasie badań można podzielić na trzy grupy:

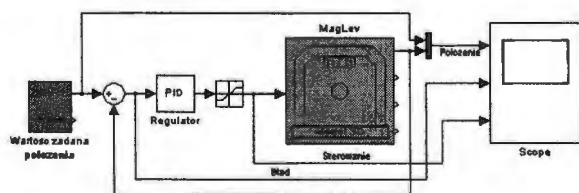
1. Eksperymenty symulacyjne.
2. Eksperymenty typu Hardware In The Loop.
3. Eksperymenty przeprowadzone na rzeczywistym obiekcie.

Eksperymenty symulacyjne przeprowadzono na każdym etapie budowy regulatora FPGA. Służyły one do wstępnej i częściowej weryfikacji pracy tworzonego regulatora-

ra. Do tego celu został wykorzystany ciągły, nieliniowy model laboratoryjnego systemu magnetycznego zawieszenia [6]. Zostały przeprowadzone następujące symulacje:

- sterowanie modelem przy pomocy regulatora ciągłego,
- sterowanie modelem przy pomocy dyskretnego regulatora,
- sterowanie modelem przy pomocy dyskretnego regulatora zbudowany za pomocą elementów przyborka Fixed-Point Blockset.

Wszystkie symulacje były realizowane w układzie sterowania przedstawionym na rys. 6.



Rys. 6. Układy sterowania.

Symulacje pozwoliły na bieżąco weryfikować proces tworzenia regulatora. Zastosowany przyborek Fixed-Point Blockset udostępnia bloki podstawowych operacji arytmetycznych i logicznych. Użytkownik może definiować typy zmiennych, na których będą przeprowadzane obliczenia. Dzięki tej właściwości możliwe jest zbudowanie algorytmu sterującego, który będzie pracował tak jak algorytm zapisany w układzie FPGA. Pozwoliło to na przetestowanie zmodyfikowanej formuły regulatora przed stworzeniem projektu dla układu FPGA i wstępną oceną jego przydatności do sterowania system magnetycznego zawieszenia.

Drugą grupą eksperymentów były eksperymenty z wykorzystaniem techniki Hardware In The Loop. Zostały przeprowadzone w celu potwierdzenia zgodności algorytmów stworzonych przy pomocy elementów Fixed-Point Blockset z algorytmami zapisanymi w języku VHDL. Eksperymenty te polegały na uruchomieniu regulatora PID w układzie FPGA i wirtualnemu połączeniu go z matematycznym modelem układu magnetycznego zawieszania. Eksperymenty tego typu pozwalają na weryfikację działania regulatora bez konieczności sterowania rzeczywistym obiektem. Zwykle stosuje się je w przypadkach, gdy nie można pozwolić sobie na błędy wprowadzone przez źle zaprojektowany regulator, np. sytuacje niebezpieczne. W przypadku sterowania rzeczywistym systemem magnetycznego zawieszania nie można w pełni przetestować działania regulatora z uwagi na niestabilność systemu.

W celu realizacji tej części badań zmodyfikowano logikę dla karty z układem FPGA tak, aby pozwalała na:

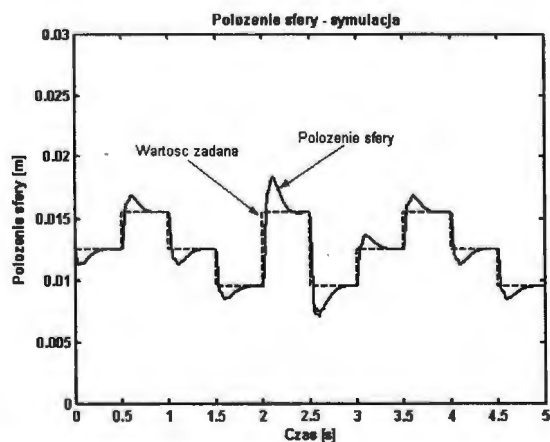
- wyzwalaną pracą regulatora – obliczenia wyłącznie na żądanie z komputera,
- możliwość przesyłania do komputera pośrednich wyników obliczeń,
- możliwość zmiany wszystkich parametrów układu regulacji.

Głównym zadaniem eksperymentów Hardware In The Loop było przetestowanie regulatora. Na ich podstawie można ocenić również przydatność wybranej przez autora ścieżki przenoszenia algorytmów sterowania na platformę FPGA.

Trzecią grupą przeprowadzonych eksperymentów były eksperymenty z wykorzystaniem rzeczywistego układu magnetycznego zawieszenia. Zostały przeprowadzone dla dwóch przypadków. W klasycznym układzie sterowania, w którym regulator jest wyliczany przez komputer oraz w układzie sterowania z układem FPGA, w którym pętla sprzężenia zwrotnego jest zamknięta przez układ FPGA. W pierwszym przypadku regulator został zbudowany przy pomocy pakietu MATLAB/Simulink, a następnie przez odpowiednie narzędzia przetworzony i uruchomiony z zachowaniem rygorów czasu rzeczywistego. W drugim przypadku regulator był umieszczony w układzie FPGA, a komputer aktualizował wartość zadaną oraz zbierał dane pomiarowe dla wizualizacji. Oba eksperymenty były przeprowadzone w takich samych warunkach, co pozwala na bezpośrednie porównanie wyników eksperymentów.

6. WYNIKI EKSPERYMENTÓW

W tej części artykułu zamieszczono wybrane wyniki eksperymentów przeprowadzonych w badanym układzie sterowania magnetycznym zawieszaniem.



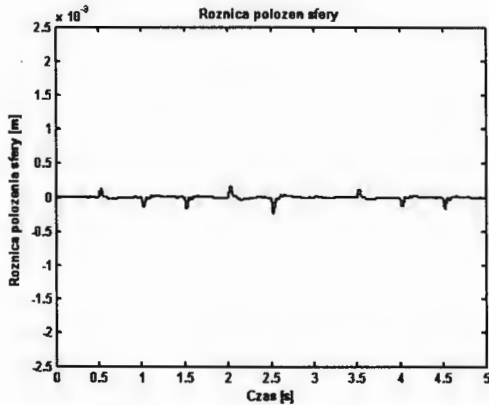
Rys. 7. Eksperymenty symulacyjne - regulator ciągły.

Przebieg z rys. 7 przedstawia pozycję lewitującej sfery oraz wartość zadaną. Wyniki zebrano podczas eksperymentu symulacyjnego z wykorzystaniem regulatora dyskretnego zbudowanego przy pomocy Fixed-Point Blockset. Przebiegi położenia sfery w pozostałych eksperymentach symulacyjnych niewiele odbiegały od przebiegu przedstawionego na rys. 7 i dlatego ich wykresy zostały pominięte.

Wszystkie symulacje oraz eksperymenty Hardware In The Loop były przeprowadzone dla takiego samego przebiegu wartości zadanej przedstawionego na rys. 7.

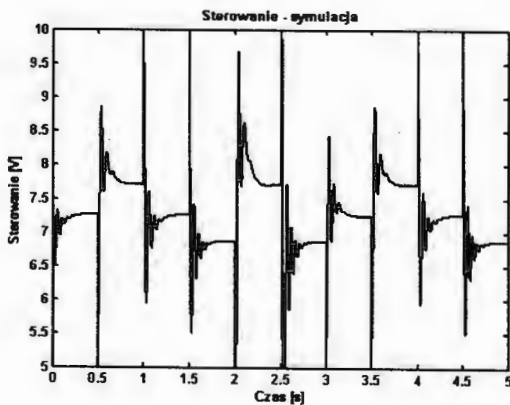
Rys. 8 przedstawia różnicę pomiędzy położeniami sfery odczytanymi podczas sterowania modelem przy pomocy regulatora ciągłego, a położeniami odczytanymi podczas sterowania modelem przy pomocy regulatora dys-

kretnego Fixed-Point. Różnica ta nie przekracza 0.25 mm. Należy zaznaczyć, że wyniki eksperymentów polegające na sterowaniu modelem za pomocą zmodyfikowanego regulatora dyskretnego były identyczne, bez względu na to czy regulator był obliczany w komputerze, czy układzie FPGA (eksperyment Hardware In The Loop).

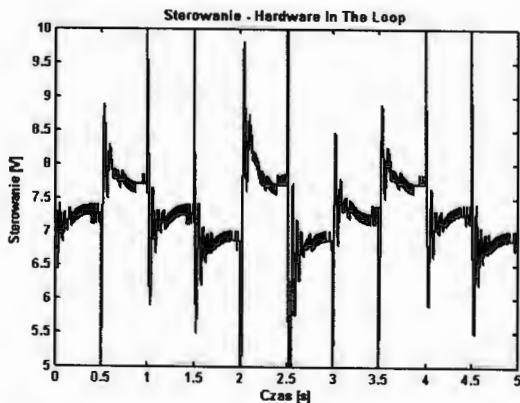


Rys. 8. Różnica położenia sfery – symulacja, eksperyment Hardware In The Loop.

Rys. 9 przedstawia sygnał sterowania uzyskany podczas eksperymentu symulacyjnego z wykorzystaniem zwykłego regulatora PID.



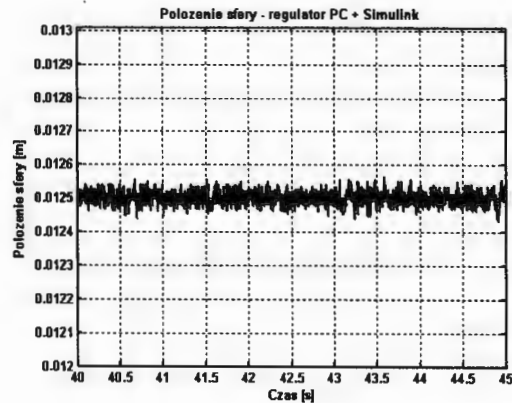
Rys. 9. Sterowanie - symulacja, regulator ciągły.



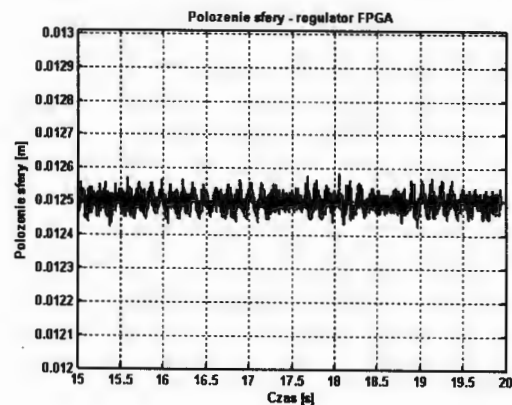
Rys. 10. Sterowanie - zmodyfikowany regulator dyskretny.

Rys. 10 przedstawia przebieg sygnału sterowania uzyskany w eksperymentach symulacyjnych typu Hardware In The Loop ze zmodyfikowanym regulatorem dyskretnym uruchomionym w układzie FPGA. Sygnały z rys. 9 i rys. 10 są do siebie zbliżone, ale nie identyczne. Przebieg sterowania dla zmodyfikowanego regulatora zbudowanego w FPGA jest bardziej nierównomierny. Przyczyną tego faktu jest przeprowadzenie obliczeń na liczbach stałoprecyzyjnych.

Ostatnimi eksperymentami były eksperymenty przeprowadzone na obiekcie rzeczywistym. Przeprowadzono dwa eksperymenty.



Rys. 11. Położenie sfery - regulator w PC.



Rys. 12. Położenie sfery - regulator w FPGA.

Pierwszy, w którym układ magnetycznego zawieszenia był sterowany za pomocą klasycznego regulatora PID opisanego w [6]. Drugi, w którym obiekt był sterowany za pomocą zmodyfikowanego, dyskretnego regulatora PID działającego w układzie FPGA.

W obu przypadkach wartość zadana była ustalona na stałym poziomie 12,5mm. Wyniki tych eksperymentów przedstawiają odpowiednio rys. 11 i rys. 12. Na wykresach zamieszczono przebiegi położenia lewitującej sfery.

7. WNIOSKI

Przeprowadzone eksperymenty symulacyjne oraz eksperymenty na rzeczywistym obiekcie pokazały, że możliwe jest sterowanie układem magnetycznego zawieszenia przy pomocy regulatora działającego w układzie

FPGA. Na rys. 11 i rys. 12 widoczne jest, że zmienność sygnałów położenia lewitującej sfery jest na podobnym poziomie dla systemu sterowania z układem FPGA oraz dla systemu z klasycznym układem sterowaniem. Z rys. 9 i rys. 10 wynika, że przebieg sygnału sterowania dla regulatora klasycznego i zmodyfikowanego regulatora dyskretnego nie jest taki sam. Strukturę regulatora przeznaczonego do pracy w układzie FPGA należy dodatkowo zmodyfikować w celu zmniejszenia błędów obliczeń stałoprzecinkowych.

Na podstawie przeprowadzonych eksperymentów można wnioskować, że zastosowanie specjalnych narzędzi (Fixed-Point Blokset) może być bardzo pomocne podczas tworzenia algorytmów sterujących dla struktur FPGA. Regulatory zbudowane przy pomocy tego narzędzia i zakodowane później za pomocą języka VHDL działały identycznie. Można zatem stosunkowo szybko zbudować regulator przy pomocy efektywnych narzędzi i sprawdzić jego działanie symulacyjnie. Dopiero po uzyskaniu odpowiedniej struktury algorytmu zakodować go w mało wygodnym języku opisu sprzętu. Zastosowanie regulatorów działających w strukturach układu FPGA niesie za sobą następujące korzyści:

- obniżenie kosztów układu sterującego,
- integrację wielu funkcji w jednym układzie scalonym,
- uzyskanie znacznie krótszych okresów sterowania,
- poprawienie punktualności zadania sterującego.

Niekorzystną cechą takiego układu sterowania jest natomiast wnoszenie dodatkowych błędów spowodowanych mniejszą precyzją obliczeń. Wadą jest również skomplikowany proces tworzenia projektu dla układu programowalnego.

MAGNETIC LEVITATION SYSTEM CONTROL WITH FAST CONTROLLER BASED ON FPGA TECHNOLOGY

Abstract: This paper presents the design procedure of the controller working in the FPGA circuit. The hardware platform constraints must be met during the development of the controller. The designed controller has been used with the laboratory magnetic suspension system. This system is unstable and required high sampling frequency. The paper presents designing and testing of the controller for this system. Results of several simulations and testing experiments on the plant are included.

Literatura

- [1] Duda J. (2003) *Modele matematyczne, struktury i algorytmy nadrzędnego sterowania komputerowego*. Wyd. AGH, Kraków.
- [2] Grega W. (1999) *Sterowanie cyfrowe w czasie rzeczywistym*. Wyd. Wydz. EAIiE AGH, Kraków.
- [3] Kalisz J. (2002) *Język VHDL w praktyce*. Wyd. Komunikacji i Łączności, Warszawa.
- [4] Kołek K. (2003) Realizacja sterownika PID w technologii FPGA. *IV Krajowa Konferencja Metody i Systemy Komputerowe w Badaniach Naukowych i Projektowaniu Inżynierskim*, Kraków.
- [5] Larminat P., Thomas Y. (1983) *Automatyka - układy liniowe*. WNT, Warszawa.
- [6] Piłat A. (2002) *Sterowanie układami magnetycznej lewitacji. Praca doktorska*. AGH Katedra Automatyki, Kraków.
- [7] Xilinx Inc. (2005) *Dokumentacja układów FPGA dostępna w Internecie pod adresem www.xilinx.com*.



Instytut Badań Systemowych
Polskiej Akademii Nauk

ISBN 83-89475-01-4