

PROTOTYP MANIPULATORA DO ZADAŃ SPECJALNYCH

Łukasz ŚWISTAK¹, Damian SZROM², Anna SZLACHTA³

1. Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. W. Pola 2, 35-959 Rzeszów, student
tel.: 516310444 e-mail: swistak.lukasz@gmail.com
2. Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. W. Pola 2, 35-959 Rzeszów, student
tel.: 509605408 e-mail: szrom.damian@gmail.com
3. Politechnika Rzeszowska, Katedra Metrologii i Systemów Diagnostycznych, ul. W. Pola 2, 35-959 Rzeszów
tel.: 177432462 e-mail: annasz@prz.edu.pl

Streszczenie: W pracy opisano prototyp manipulatora przeznaczonego do pracy w warunkach szkodliwych. Przedstawiono strukturę oraz funkcjonalność manipulatora. Przedstawiany w artykule prototyp manipulator pozwala studentom zapoznać się z zagadnieniami kinematyki odwrotnej oraz konfiguracji osobliwej. Prototyp manipulatora został wykonany w skali 1:5. Opracowany system składa się z konstrukcji manipulatora, wraz ze sterownikiem i modulem Bluetooth oraz aplikacją do obsługi robota.

Słowa kluczowe: manipulator, modelowanie, kinematyka odwrotna

1. WSTĘP

Przedstawiony w pracy projekt powstał na podstawie manipulatorów przemysłowych do zastosowania w systemie rozproszonym, a także do pracy w trudnych warunkach.

Manipulatory w przemyśle mają wiele zastosowań, głównie wykonywanie zadań precyzyjnych, praca w warunkach szkodliwych oraz przenoszenie ciężkich elementów. Jednym z takich obiektów jest robot SCORBOT-ER 9Pro (rys. 1).



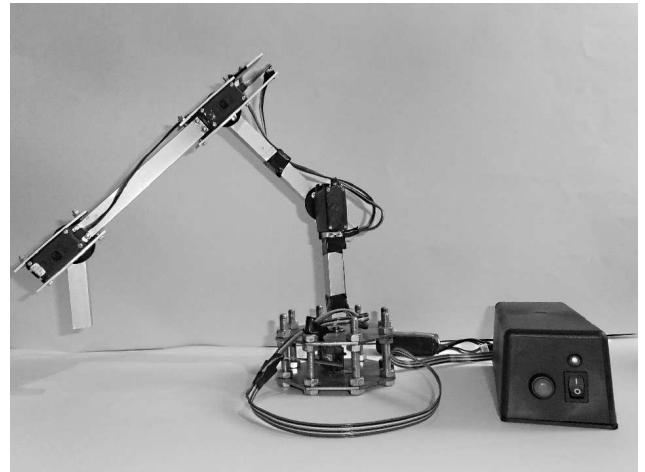
Rys. 1. Robot SCORBOT-ER 9Pro

Manipulator demonstrujący działanie urządzeń wykorzystywanych w przemyśle został zaprojektowany

i wykonany zgodnie z zasadami budowy robotów przemysłowych [1].

Przedstawiony manipulator pozwala studentom zapoznać się z zagadnieniami kinematyki odwrotnej oraz konfiguracji osobliwej, ponadto wykonanie niniejszego manipulatora pozwala na redukcję kosztów sprzętu laboratoryjnego.

Opracowany system składa się z konstrukcji manipulatora, wraz ze sterownikiem i modulem Bluetooth oraz dedykowaną aplikacją do obsługi robota (rys. 2). Jest to obiekt sterowania, który ma wykonywać ściśle określone zadania zgodnie z rozkazami otrzymanymi od urządzenia sterującego [2].



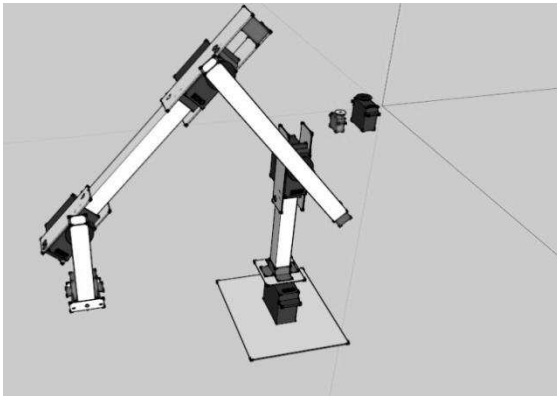
Rys. 2. Widok opracowanego manipulatora

Obiekt składa się ze stabilnej podstawy, czterech członów ramienia, czterech serwo mechanizmów, sterownika serw, stabilizatora napięcia oraz modułu do komunikacji. Do manipulatora można zamontować efektor: elektromagnes lub chwytak. Obiekt można sterować z dedykowanej aplikacji uruchomionej na komputerze PC poprzez zadawanie wartości poszczególnych kątów dla przegubów (JOINT), istnieje również możliwość zadania trajektorii przez poszczególne punkty (tryb AUTO), a także wskazanie docelowego punktu efektoru za pomocą współrzędnych

kartezjańskich. W trakcie pracy możliwy jest pomiar napięcia zasilania oraz pomiar wartości temperatury otoczenia i na tranzystorze mocy. Dane pomiarowe są zapisywane do bazy Oracle [3].

2. PROJEKT GRAFICZNY MANIPULATORA

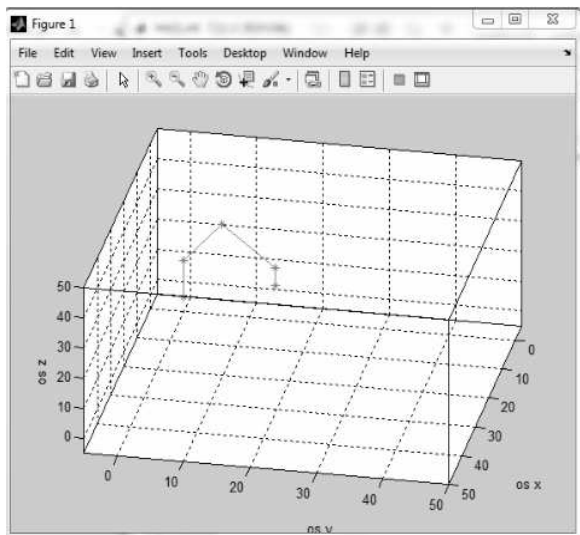
Opracowany manipulator jest prototypem prawdziwego obiektu i został zbudowany w skali 1:5. Do jego budowy wykorzystane zostały serwomechanizmy modelarskie. Konstrukcja została wykonana z aluminiowych profili. Konstrukcję zaprojektowano przy pomocy oprogramowania Google SketchUp w celu ustalenia rozmiarów poszczególnych elementów oraz sposobów połączeń części manipulatora (rys.3.).



Rys. 3. Widok projektu manipulatora w środowisku Google SketchUp

3. KINEMATYKA ODWROTNA

Po zamodelowaniu konstrukcji zostały wykonane obliczenia związane z kinematyką odwrotną. Do jej opracowania wykonano aplikację w środowisku Matlab. Argumentami funkcji są współrzędne początkowe oraz końcowe położenia efektora w przestrzeni zapisane w układzie współrzędnych kartezjańskich. Środowisko Matlab podczas wykonywania funkcji zwraca zestawy wartości kątów w dyskretnych chwilach czasu przejścia z punktu do punktu i na tej podstawie wyświetla animację przejścia efektora.



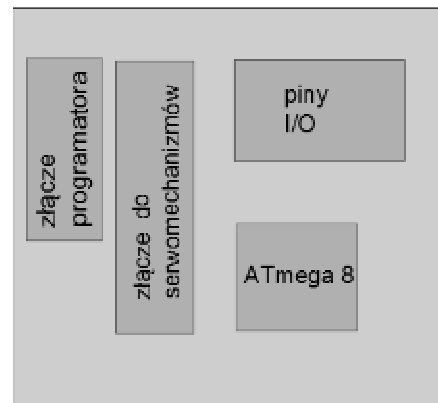
Rys. 4. Widok z symulacji przemieszczenia efektora manipulatora w środowisku Matlab

Główną informacją uzyskaną z opracowanej aplikacji kinematyki odwrotnej jest konfiguracja kątów na przegubach w chwili startu oraz na końcu symulacji. Uzyskane wartości kątów zostają przesyłane do sterownika manipulatora, który realizuje zmianę położenia punktu efektora. Widok okna opracowanej w Matlabie aplikacji przedstawia rysunek 4.

4. STRUKTURA SPRZĘTOWA

Schemat rozmieszczenia poszczególnych elementów na zaprojektowanej i wykonanej płytce PCB przedstawia rysunek 5.

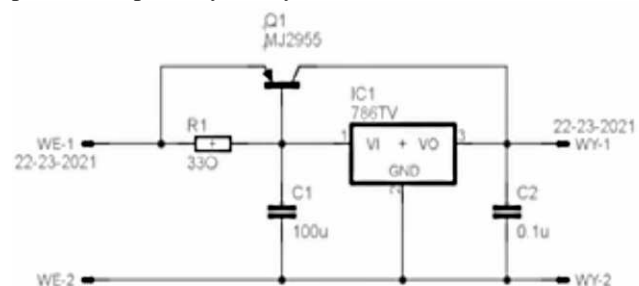
Komunikacja manipulatora z komputerem została zrealizowana za pomocą interfejsu bezprzewodowego Bluetooth. Takie rozwiązanie pozwala na zdalne sterowanie manipulatorem, co jest istotne w pracy w warunkach szkodliwych dla człowieka. Po stronie mikrokontrolera sterującego manipulatorem jest to moduł HC-05.



Rys. 5. Schemat blokowy układu sterownika manipulatora

Do budowy sterownika manipulatora wykorzystano procesor ATmega8, taktowany zewnętrznym zegarem kwarcowym o częstotliwości 16 MHz. Zastosowanie tak wysokiej rozdzielczości było niezbędne do precyzyjnego wysterowania serwomechanizmów, pomiaru warunków atmosferycznych oraz komunikacji z urządzeniem nadrzędnym.

Wartość natężenia prądu podczas pracy manipulatora ze względu na zastosowane serwomechanizmy może przekraczać 5 A [4], dlatego zaprojektowano i wykonano odpowiedni układ zasilania. Na rysunku 6 przedstawiono schemat elektryczny sterownika prądowego. Na radiatorze zmniejszającym rezystancję termiczną tranzystora umieszczony jest specjalny termistor, który odpowiada za pomiar temperatury tranzystora.



Rys. 6. Schemat elektryczny sterownika prądowego

Pomiar temperatury jest istotnym elementem właściwej pracy układu. W układzie zastosowano termistor NTC 10k. Niezbędne było określenie charakterystyki tego elementu.

W celu optymalizacji i poprawy dokładności wyznaczenia równania dla punktów pracy pomiaru temperatury otoczenia oraz temperatury obudowy tranzystora wykonano dwie aproksymacje. Takie rozwiązanie pozwala na oszczędzenie czasu procesora przy obliczaniu temperatury dla danego punktu pracy, ponieważ znacznie upraszcza postać wielomianu aproksymującego.

Obliczenia zostały wykonane metodą najmniejszych sumy kwadratów przy użyciu darmowego programu *Aproksymacja v1.5.7.2* [5, 6].

Wielomian dla pomiaru temperatury na tranzystorze ma postać:

$$f_{tr}(x) = -0,025 \cdot x^3 + 6,55 \cdot x^2 - 605,8 \cdot x + 2649,3. \quad (1)$$

Wielomian dla pomiaru temperatury otoczenia ma postać:

$$f_o(x) = 0,259 \cdot x^3 + 30,9 \cdot x^2 - 1502 \cdot x + 32294,9. \quad (2)$$

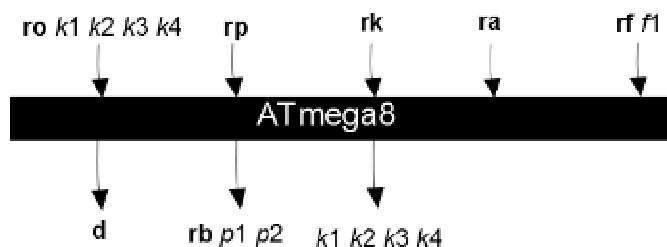
5. OPROGRAMOWANIE MIKROKONTROLERA

Program zaimplementowany na sterowniku posiada funkcję pomiarów stanu baterii oraz wartości temperatur, możliwość pseudo-równoległego generowania sygnałów sterujących serwomechanizmami, a także obsługę komunikacji z urządzeniem nadrzędnym. Program został napisany w środowisku Eclipse z pluginem do mikrokontrolerów AVR. Sterownik po załączeniu zasilania ustawia domyślną wartość wszystkich kątów, a następnie oczekuje na wciśnięcie przycisku na panelu.

Na rysunku 7 przedstawiono możliwy zestaw komend wysyłanych i odbieranych z mikrokontrolera ATmega8.

Każdy rozkaz wysyłany do mikrokontrolera rozpoczyna się znakiem „r”, następnie w zależności od typu rozkazu podawane są kolejne znaki, które oznaczają typ rozkazu. Rozkazy możemy podzielić na pięć grup:

- rozkaz „o” zadaje wartości kątów odpowiednio k_1, k_2, k_3, k_4 ,
- rozkaz „p” odpowiada za pomiary temperatury p_1, p_2 ,
- rozkaz „k” pobiera wartości zadanych kątów,
- rozkaz „a” zatrzymuje tryb auto,
- rozkaz „f” reguluje prędkość kątową serwomechanizmów.



Rys. 7. Schemat komunikacji

Odpowiedzi z mikrokontrolera zwracają odpowiednie wartości nastawionych kątów lub wykonanych pomiarów.

6. APLIKACJA

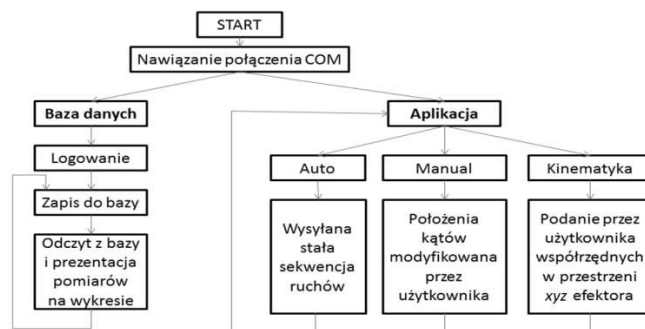
Do komunikacji z manipulatorem służy dedykowana aplikacja okienkowa, która została napisana w języku C# [7]. Tworzenie aplikacji to nie tylko tworzenie kodu, ale również *Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki PG, ISSN 2353-1290, Nr 38/2014*

tworzenie interfejsu użytkownika. Środowisko Microsoft Visual Studio.NET 2010 wspiera proces tworzenia aplikacji okienkowych w pełnym zakresie [8]. Zintegrowane narzędzia do projektowania interfejsu pozwalają na wygodne i szybkie budowanie aplikacji okienkowych.

Opracowana aplikacja składa się z dwóch części:

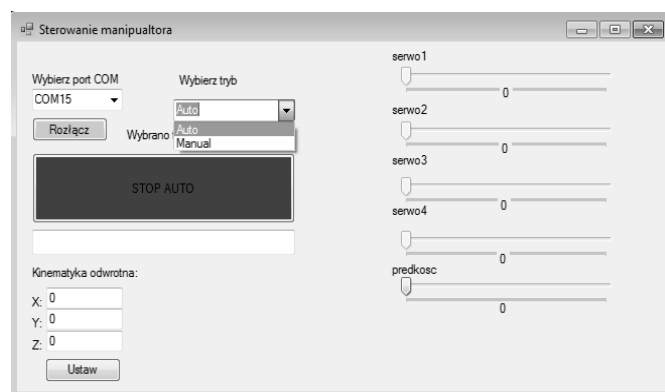
- okna odpowiedzialnego za sterowanie manipulatorem,
- okna odpowiedzialnego za zapisywanie położeń manipulatora do bazy oraz graficzne przedstawienie na wykresie.

Schemat blokowy opracowanej w języku C# aplikacji do sterowania manipulatorem przedstawiono na rysunku 8.



Rys. 8. Schemat blokowy programu

Sterowanie manipulatorem odbywa się przy użyciu opracowanego programu, na rysunku 9 przedstawiono interfejs użytkownika – tj. widok okna sterującego.



Rys. 9. Okno aplikacji do sterowania pracą manipulatora

Aplikacja pozwala na sterowanie manipulatorem w trzech trybach:

- tryb automatyczny,
- tryb ręczny,
- tryb kinematyki odwrotnej.

Włączenie trybu Auto powoduje wysłanie do manipulatora ramki z komendą uruchomienia sekwencji standardowej, która została zapisana w pamięci sterownika manipulatora.

Przełączenie się w tryb Manual pozwala na sterowanie oddzielnie każdym z przegubów manipulatora (poprzez przesuwanie suwaków znajdujących się po prawej stronie okna z rysunku 9). Przesunięcie jednego z suwaków powoduje zmianę przesyłanej ramki, która zawiera informację o wartościach kątów. W rezultacie powoduje to zmianę pozycji manipulatora.

Kolejnym trybem pracy manipulatora jest sterowanie punktem efektora przy użyciu kinematyki odwrotnej. Po

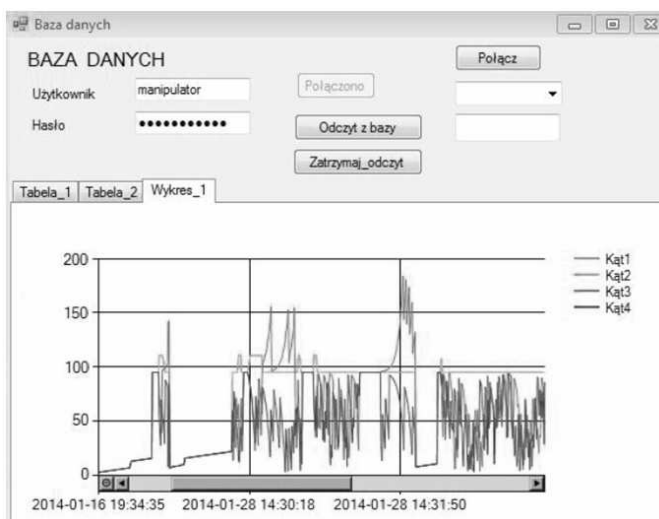
podaniu przez użytkownika współrzędnych efektora x , y , z i wciśnięciu przycisku Ustaw opracowana w środowisku Matlab aplikacja przelicza kinematykę odwrotną. Wyznacza wartości kątów dla wszystkich przegubów w zależności od pozycji i orientacji końcówki roboczej, a następnie wysyła odpowiednią ramkę z kątami do sterownika. Dzięki czemu efektor manipulatora przemieszcza się w podane miejsce w przestrzeni. Wysyłane ramki zawierają wartości kątów dla każdego z czterech serwomechanizmów (rys. 10).

Nazwa – typ rozkazu	dane			
ro	k_1	k_2	k_3	k_4

Rys. 10. Ramka danych z kątami

Aplikacja ma również możliwość ustawienia programowo prędkości kątowej serwomechanizmów.

Druga część aplikacji umożliwia zapis do bazy danych oraz odczyt informacji / danych i przedstawienie ich na prostym wykresie lub tabeli.



Rys. 11. Wykres położenia każdego z kątów

Na komputerze roboczym zainstalowano serwer bazy danych Oracle oraz stworzono konto wraz z bazą danych. Aplikacja służy do obsługi komunikacji z serwerem bazy danych. Po zalogowaniu się do bazy danych Oracle aplikacja pobiera dane ze sterownika, a następnie umieszcza je w bazie. Poszczególne dane w dziedzinie czasu prezentowane są na

wykresach. Dane gromadzone w bazie są archiwizowane, dzięki czemu jest wgląd do poszczególnych położenia manipulatora przedstawionych na wykresie lub w formie tabeli. Ponadto za pomocą zgromadzonych danych można sprawdzić czasy pracy manipulatora oraz temperaturę mierzoną przez termistory.

7. PODSUMOWANIE

W artykule przedstawiono prototyp manipulatora o konfiguracji 4R. Prezentowany manipulator umożliwia przenoszenie przedmiotów o niewielkiej masie, do około 15 dag. Wynika to z zastosowanych serwomechanizmów oraz masy własnej ramion manipulatora. Zaletą przedstawionego manipulatora jest niski koszt realizacji, co w przypadku zastosowań dydaktycznych jest atutem. Studenci podczas pracy z prototypem manipulatora mogą zapoznać się z zagadnieniami kinematyki odwrotnej, konfiguracji osłowej oraz działaniem robota, jako teleoperatora.

8. BIBLIOGRAFIA

1. Tomaszewski K.: Roboty Przemysłowe. Projektowanie układów mechanicznych. WNT, Warszawa, 1993.
2. Fedus A., Osypiuk R.: Architektura sterowania równoległym manipulatorem Hexa na bazie mikrokontrolerów rodziny SAM7S, Pomiary Automatyka Kontrola vol. 56, nr 6/2010, s. 581 -584.
3. Roger, U.: Oracle. Przewodnik projektowania baz danych. WNT, Warszawa 1995.
4. www.datasheet-pdf.com/datasheet/TowerPro_/633270/SG-5010.pdf.html, strona z dokumentacjami, dokumentacja serwomechanizmu SG-5010, 28.11.2013r.
5. Korgul A.: Analiza danych pomiarowych – Materiały pomocnicze dla studentów Wydziału Chemii UW. Wersja trzecia, uzupełniona.
6. <http://www.eureka-pile.republika.pl/bcc.html> Źródło programu Aproksymacja, programy autora Michała Bąbika, 10.01.2014r.
7. Lis M.: C#. Poznaj tajniki programowania w C#, Wydawnictwo Helion, Gliwice 2007.
8. Farbaniec D.: Microsoft Visual Studio 2012. Programowanie w C#, Wydawnictwo Helion, Gliwice 2013.

A PROTOTYPE OF MANIPULATOR FOR SPECIAL TASKS

The work describes a prototype manipulator designed to work under harmful conditions. The prototype of manipulator enables students to familiarize with problems of inverse kinematics and singular configuration was presented. Developed manipulator has been built at a scale of 1:5. The article presents the structure and functionality of the manipulator. The developed system consists of wireless communication by Bluetooth and control system. The application window, which was written in C #, is dedicated to communicate with the manipulator. Control of manipulator is carried out in three modes of operation.

Keywords: manipulator, modeling, inverse kinematics.