



**EduAkademia.pl**

prace naukowe na zlecenie

Praca-licencjacka-chomikuj-25

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

AKADEMIA GÓRNICZO -HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE WYDZIAŁ ELEKTROTECHNIKI,  
AUTOMATYKI, INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ

KATEDRA AUTOMATYKI I INŻYNIERII BIOMEDYCZNEJ

PRACA DYPLOMOWA INŻYNIERSKA

BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO

MOBILNEGO ROBOTA RYSUJĄCEGO

BUILDING AND PROGRAMMING OF A TWO-WHEELED MOBILE

DRAWING ROBOT

AUTOR: KRZYSZTOF KAPUSTA

KIERUNEK STUDIÓW : AUTOMATYKA I ROBOTYKA

OPIEKUN PRACY: DR INŻ. DARIUSZ MARCHEWKA

KRAKÓW 2013 / 2014

PRACA INŻYNIERSKA AGH EAIIB 20141

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

#### OŚWIADCZENIE AUTORA PRACY

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ PRACĘ DYPLOMOWĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I ŻE NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMIENIONE W PRACY.

.....

PODPIS

2  
AGH EAIIB 2014  
PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

## PODZIĘKOWANIA

DZIĘKUJĘ MOJEMU PROMOTOROWI, DR INŻ. DARIUSZOWI MARCHEWCE, ZA WSPARCIE I OPIEKĘ MERYTORYCZNĄ NAD TĄ PRACĄ. DZIĘKUJĘ RÓWNIEŻ

MATEUSZOWI FURDYNIE ZA CENNE UWAGI DOTYCZĄCE KONSTRUKCJI ROBOTA.

Spis treści

Spis treści .....	
5	
Wstęp .....	
7	
Cel i zakres pracy .....	
8	
Rozdział 1.	
Wprowadzenie teoretyczne.....	
9	
1. 1.	
Roboty mobilne i ich napędy .....	
9	
1. 2.	
Roboty i urządzenia rysujące .....	
11	
Rozdział 2. Konstrukcja mechaniczna robota.....	
13	
2. 1.	
Wybór sposobu sterowania .....	
13	
2. 2.	
Części napędowe .....	
14	
2. 3.	
Mechanizm opuszczania elementu rysującego .....	
15	
2. 4.	
Złożenie konstrukcji .....	
17	
Rozdział 3.	
Kontroler robota .....	

19	
3. 1.	
Wybór podzespołów .....	
.....	20
3. 1. 1.	
Mikrokontroler .....	
20	
3. 1. 2.	
Sekcja zasilania .....	
21	
3. 1. 3.	
Układy kontrolujące silniki .....	
22	
3. 1. 4.	
Moduły komunikacyjne .....	
23	
3. 1. 5.	
Peryferia .....	
23	
3. 2.	
Schemat elektryczny .....	
24	
3. 3.	
Projekt PCB .....	
26	
3. 4.	
Wykonanie elektroniki .....	
27	
3. 5.	
Uruchomienie .....	
27	
Rozdział 4.	
Oprogramowanie .....	
29	
4. 1.	
System operacyjny czasu rzeczywistego .....	
29	
4. 2.	
Warstwa abstrakcji sprzętu .....	
30	
4. 3.	
Zadania i komunikacja między nimi .....	
31	
4. 3. 1.	
Zadanie kontroli silników .....	
32	
4. 3. 2.	
Zadanie generatora ruchu .....	
34	
4. 3. 3.	

Zadanie kontroli pisaka .....	
38	

PRACA INŻYNIERSKA AGH EAIIB 20145

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

4. 3. 4.	
Zadanie telemetrii .....	
39	
4. 3. 5.	
Zadania komunikacyjne .....	
40	
Rozdział 5.	
Testy .....	
42	
5. 1.	
Dokładność rysowania kształtów .....	
42	
5. 2.	
Weryfikacja działania regulatora dojeżdżania do punktu .....	
45	
5. 3.	
Poprawność rysowania wszystkich rodzajów linii .....	
46	
5. 4.	
Pomiary poprawności działania sekcji zasilania .....	
47	
5. 5.	
Weryfikacja poprawności sterowania silnikami .....	
50	
5. 6.	
Identyfikacja problemu tzw. „drgania styków” mikrop rzeźączników .....	
51	
Rozdział 6.	
Podsumowanie .....	
53	
Wykaz źródeł ilustracji .....	
55	
Dodatek A:	
Schematy elektryczne .....	
56	
Dodatek B:	
Schemat elektryczny i projekt PCB modułu bezprzewodowego .....	
61	





## Wstęp

Robotyka jest prężnie rozwijającą się dziedziną nauki. Jej popularyzacja w ostatnich latach spowodowała ogromne zainteresowanie konstruktorów -amatorów budowania robotów, głównie mobilnych, które uczestniczą w międzynarodowych zawodach i pokazach. Najciekawsze, najbardziej zaawansowane i innowacyjne konstrukcje wygrywają cenne nagrody, a ich konstruktorzy zdobywają doświadczenie procentujące w przyszłej pracy naukowej lub zawodowej. [1]

Głównym powodem budowy mobilnego robota rysującego była chęć stworzenia konstrukcji całkowicie innej od dotychczas istniejących. Co prawda istnieje wiele komercyjnie dostępnych urządzeń i robotów zdolnych rysować, jak drukarki, plotery, czy roboty przemysłowe, jednak żadne z nich nie zapewnia tak unikalnego zestawu cech jak prezentowana w tej pracy konstrukcja.

Potencjalne zastosowania mobilnego robota rysującego wykraczają daleko poza zadania rysowania. Stanowi on wyjątkową platformę badawczą, na której można testować rozmaite algorytmy będące przedmiotem badań ośrodków naukowych związanych z robotyką na całym świecie. Jego unikalna cecha – znaczenie rzeczywiście przejechanej trasy – może być wykorzystana w konstrukcji robotów edukacyjnych. Stworzony w roku 1967 edukacyjny język programowania Logo opiera się na „żółwiu” który poruszany kolejnymi poleceniami zostawia na monitorze komputera ślad przejechanej trasy, jak na Rys. 0.1. Przeniesienie takiej funkcjonalności do świata rzeczywistego stworzyłoby unikalne doświadczenie, które mogłoby pomóc dzieciom w zgłębianiu podstaw programowania.

Rys. 0.1: Przykładowy wynik programu napisanego w języku Logo

PRACA INŻYNIERSKA AGH EAIIB 20147

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Cel i zakres pracy

Celem tej pracy jest zaprojektowanie, wykonanie i oprogramowanie mobilnego robota zdolnego rysować po powierzchni po której się porusza.

Rozdział pierwszy wprowadza w tematykę robotów mobilnych. Zostały w nim przedstawione podstawowe rodzaje konstrukcji i napędów, wraz z ich zaletami i wadami. Omówiono szczegółowo budowę i zastosowanie istniejących urządzeń drukujących i rysujących oraz robotów najczęściej wykorzystywanych do tego typu zadań.

W rozdziale drugim opisano konstrukcję zbudowanego robota rysującego. Szczegółowo przedstawiono wszystkie wykorzystane elementy mechaniczne. Przedstawiono rysunek zło-

żeńiowy konstrukcji i wytłumaczono zasadę działania robota.

Rozdział trzeci zawiera opis wykonanego układu elektrycznego sterującego robotem.

Opisano w nim użyty mikrokontroler oraz wszystkie elementy elektroniczne służące do m.in.

kontroli silników i przesyłania danych do komputera .

W rozdziale czwartym omówiono oprogramowanie stworzone dla mikrokontrolera sterującego każdym elementem robota. Przedstawiono algorytmy pozycjonowania w przestrzeni i algorytmy kontrolujące jazdę po zadanych trajektoriach.

Ostatni rozdział zawiera opis testów wykonanych celem sprawdzenia poprawności działania konstrukcji mechanicznej, elementów elektrycznych i programu. Zawarto w nim zdjęcia rysunków wykonywanych przez robota oraz pomiary kluczowych dla działania robota sygnałów elektrycznych.

8 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Rozdział 1. Wprowadzenie teoretyczne

Robot to programowalna maszyna zdolna do wykonywania serii skomplikowanych czynności w sposób autonomiczny, w szczególności z wykorzystaniem komputera [1]. Pierwsze roboty powstawały już w czasach starożytnych, ale maszyny sterowane przez program komputerowy pojawiły się dopiero kilkadziesiąt lat temu. W ostatnich latach dokonał się zadziwiający postęp i zaprojektowano wiele różnych rodzajów konstrukcji do wykonywania rozmaitych zadań.

Roboty można podzielić pod względem budowy na dwie ogólne kategorie (Tabela 1.1):

R  
OBOTY

ROBO  
TY

STAC  
JONARNE

MOBIL  
NE

- M ONI PULATORY

- J EŹDŹĄCE

PRZEM YSŁOWE

- K ROCZĄCE

- F REZA RKI,

- P ŁYWAJ ĄCE

TOKAR KI CNC

- L ATAJĄCE

- I NNE

- I NNE

Tabela 1.1: Podział robotów ze wzgl ędu na budow ę

Roboty stacjonarne znajduj ą zastosowanie g łównie w fabrykach i zakładach przem y-słowych. Wykonuj ą tam precyzyjne i powtarzalne czynności takie jak ci ęcie, spawanie, ma-lowanie, przenoszenie lub pakowanie

zastępując przy tym ludzi. Do tej kategorii można również zaliczyć obecną w każdym gospodarstwie domowym pralkę automatyczną lub drukarkę komputerową. Wszystkie roboty tego typu posiadają części ruchome, jednak nie mogą zmienić swojego stanowiska pracy. Jeżeli robot ma możliwość dowolnej zmiany swojego położenia, zalicza się do robotów mobilnych.

### 1. 1. Roboty mobilne i ich napędy

Roboty mobilne to maszyny zdolne do automatycznego poruszania się w zadanym środowisku, czy to na lądzie, w wodzie, czy w powietrzu. Dzięki temu, że nie wymagają bezpośredniej obecności człowieka, mogą przyjmować rozmaite kształty i rozmiary oraz wykonywać zadania niemożliwe dla tradycyjnych konstrukcji. Wszystkie posiadają charakterystyczny

PRACA INŻYNIERSKA AGH EAIIB 20149

K.KAPUSTA - BUDOWA I PROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

dla siebie system napędowy pozwalający na poruszanie się. Ich przestrzeń robocza jest konstrukcyjnie nieograniczona.

Największą grupą robotów mobilnych są roboty jeżdżące. Wykorzystuje się je między innymi do zadań transportowych, inspekcyjnych, serwisowych i edukacyjnych. Najważniejszą cechą różniącą je jest sposób poruszania się na lądzie. Można wyróżnić następujące typy napędu:

- ▶ Napęd kołowy
- ▶ Napęd gąsienicowy
- ▶ Inne

Roboty mobilne o napędzie kołowym można podzielić ze względu na ilość kół i liczbę punktów podparcia z podłożem. Najczęściej spotykanymi konstrukcjami są roboty z napędem gąsienicowym, dwukołowe w dodatkowymi punktami podparcia oraz czteroślupowe ze skórną osią, jak w samochodzie. Przykłady takich konstrukcji przedstawiono na rysunkach poniżej.

Rys. 1.1: Wojskowy robot  
Rys. 1.2: Arduino Robot - dwuko-  
Rys. 1.3: BST AG V Tugger - cztero-  
HD2-S Tactical o napę-  
łowy robot z dodatkowymi punkta-  
kołowy robot transportowy  
dzie gąsienicowym  
mi podparcia z podłożem

Roboty z napędem gąsienicowym (Rys. 1.1) charakteryzują się znakomitymi właściwościami trakcyjnymi. Są w stanie pokonać nawet najtrudniejsze przeszkody terenowe takie jak uskoki, wzniesienia i dziury. Podobnie jak roboty dwukołowe, potrafią obracać się w miejscu. Napęd gąsienicowy sprawdza się wspaniale w trudnym terenie, lecz do jazdy wewnątrz pomieszczeń można rozważyć inne możliwości.

Napęd kołowy (Rys. 1.2 i Rys. 1.3) zapewnia dużą przyczepność i umożliwia bardzo szybką jazdę. Jego złożoność w porównaniu do gąsienic jest znacznie mniejsza. Wybór liczby kół istotnie wpływa na właściwości jezdne. Pojazdy dwukołowe wyróżniają się nieporównywalną zwrotnością i dokładnością jazdy. Położenie robota można opisać za pomocą prostych równań, gdzie jedynymi parametrami jest droga przebyta przez każde z kół – tzw. odometria.

[2] Roboty dwukołowe zazwyczaj posiadają dodatkowy punkt podparcia z podłożem (w formie swobodnie poruszającego się koła lub kulki), aby zachować równowagę, choć istnieją

konstrukcje balansujące bez niego. Z drugiej strony roboty z większą ilością kół są znacznie stabilniejsze. Jeżeli dodatkowe koła są napędzane, to uzyskuje się właściwości jezdne bardzo podobne do konstrukcji z gąsienicami.

10 AGH EAIIB 2014 PR ACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

## 1. 2. Roboty i urządzenia rysujące

Istnieje wiele urządzeń potrafiących rysować na rozmaitych powierzchniach. Najczęściej spotykanym jest ploter, który sterowany z komputera rysuje kształty i nanosi tekst na ogromnych arkuszach papieru. Tym co odróżnia ploter od drukarki jest sposób druku. Drukarki wszystkich typów zadrukowują kartkę papieru od góry do dołu nanosząc mikroskopijne krople atramentu lub cząsteczki tonera. Na wydruku można je wyraźnie zobaczyć, zwłaszcza w powiększeniu (Rys. 1.4). W wydrukowanie równej, skośnej linii w tych technologiach jest niemożliwe. Zupełnie inaczej działa ploter, który może poruszać elementem rysującym i papierem niezależnie, co stwarza możliwość wydruku ciągłych kształtów o dowolnej orientacji (Rys. 1.5).

Rys. 1.4: Przykładowy wydruk z drukarki atramentowej w powiększeniu      Rys. 1.5: Przykładowy wydruk z plotera w powiększeniu

Klasyczne plotery zbudowane są jako bębnowe (Rys. 1.6) lub płaskie (Rys. 1.7). Istnieje jednak wiele innych możliwości budowy robotów rysujących. Jednym z najczęściej spotykanych są roboty bazujące na konstrukcji Delta (Rys. 1.8) lub SCARA (Rys. 1.9). Mogą one mieć znacznie większe pole robocze i działać szybciej niż standardowe plotery. Poza tym umożliwiają kontrolę wysokości na której odbywa się rysowanie, a co za tym idzie mogą one rysować nie tylko na płaskim papierze, ale na jakichkolwiek przedmiotach trójwymiarowych. Niemniej jednak są to roboty stacjonarne i jak każdy taki robot mają o ograniczoną przestrzeń roboczą. Rozwiązaniem tego problemu jest zastosowanie robota mobilnego.

Rys. 1.6: Ploter bębnowy Canon iPF820      Rys. 1.7: Ploter płaski ProDigi NEW

PRACA INŻYNIERSKA      AGH EAIIB 2014/11

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO



Rys. 1.8: Robot Delta firmy Kawasaki      Rys. 1.9: Robot SCARA firmy Mitsubishi

Roboty mobilne, jak już wspomniano wcześniej, cechują się nieograniczoną przestrzenią roboczą. Cechę tę można wykorzystać do budowy mobilnego robota rysującego zarówno kilkucentymetrowe jak i kilkumetrowe rysunki. Budowa takiego robota jest tematem tej pracy.

## Rozdział 2. Konstrukcja mechaniczna robota

Przed przystąpieniem do prac projektowych należy stworzyć dokładną specyfikację techniczną, aby zapewnić, że gotowa konstrukcja będzie spełniać swoje zadania prawidłowo.

Podstawowym celem budowy jest robot mobilny zdolny rysować po podłożu po którym się porusza. Podłoże będzie płaskie, nienachylone, o różnorodnej fakturze powierzchni. Mogą na nim występować niewielkie nierówności.

Rysowanie powinno móc być przerwane. Do rysowania powinny być wykorzystywane standardowe i ogólnodostępne materiały piśmiennicze, tj. pisaki, markery i długopisy. Aby rysunki były dokładne, robot będzie musiał precyzyjnie podążać za zadaną trajektorią. Będzie musiał więc znać swoją bieżącą pozycję w odniesieniu do punktu początkowego. Nie narzuca się żadnych ograniczeń na złożoność rysunków. Aby zapewnić poprawne narysowanie każdego kształtu, robot będzie musiał być zwrotny, powinien poruszać się tak samo do przodu i w tył oraz skręcać w miejscu. Prędkość rysowania powinna być jak największa. Za minimalną wartość dopuszczalną można przyjąć 5 cm/s.

### 2. 1. Wybór sposobu sterowania

Jak wspomniano w rozdziale 1. 1, jeżdżące roboty mobilne można podzielić pod względem sposobu sterowania na kilka grup. Pojazdy na gąsienicach i czterokołowe cechują się dużą przyczepnością i stabilnością na nierównym terenie. Okupione jest to jednak występowaniem znacznych poślizgów kół. Poślizg ten jest wręcz konieczny do działania maszyn podobnych do czołgu. Dwukołowce (Rys. 2.1), z drugiej strony, wyróżniają prostota sterowania i zwrotność. Nie radzą sobie jednak z nierównościami terenu. Zjawisko poślizgu jest zminimalizowane przez to, że robot może obracać każdym z kół niezależnie bez wpływu na położenie drugiego. Koła zapewniają jedynie dwa punkty podparcia, więc niezbędne jest zastosowanie dodatkowych elementów podporowych (znak X na Rys. 2.1).

PRACA INŻYNIERSKA AGH EAIIB 201413

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTY RYSUJĄCEGO

Rys. 2 .1: Schemat robota dwukołowego

Na potrzeby realizacji tej pracy wybrano konstrukcję dwukołową, głównie ze względu na minimalizację poślizgów kół. Zerowy poślizg jest jednym z założeń odometrii, estymującej pozycję robota na podstawie drogi przebytej przez każde z kół. Ten sposób pozycjonowania będzie opisany w rozdziale 4.3.4.

## 2.2. Części napędowe

Jako główny napęd robota wybrano dwa silniki prądu stałego zintegrowane z przekładnią zębatą i połączone z magnetycznymi enkoderami inkrementalnymi (Rys. 2.2). Oznaczenie producenta to 75:1 Metal Gearm otor 25D54L mm with 48 CPR Encoder. Przy wyborze kierowano się jego parametrami i w wysokim stopniu integracji wszystkich elementów, co skróciło czas projektowania. Przekładnia jest niezbędna, aby ograniczyć prędkość obrotową wału silnika i zwiększyć jego moment. Enkodery posłużą do odczytu położenia wału silnika, a przez to odczytu położenia robota w przestrzeni.

Rys. 2.2: Silnik DC Pololu serii 25Dx54L

Tabela 2.1 zawiera zbiór najważniejszych parametrów silnika.

Napięcie zasilania  
3 - 9 V DC

Pobór prądu w zwarciu  
2.2A @ 6V

Przełożenie przekładni  
74.83 : 1

Prędkość obrotowa za przekładnią  
75 obr/min @ 6V

Typ enkodera  
Magnetyczny inkrementalny 48 CPR

Rozdzielczość enkodera  
3592 CPR za przekładnią

Tabela 2.1: Parametry silnika

Do silnika dopasowano aluminiowe koła o średnicy 70mm z gumową oponą o okrągłym profilu (Rys. 2.3). Profil opony jest bardzo ważny, gdyż powierzchnia styku kół z podłożem powinna być jak najmniejsza. Zmniejszy to błędy pozycjonowania.

Przy tak wybranych elementach układu napędowego, robot będzie poruszał się z maksymalną prędkością równą 0.345 m/s przy średnim napięciu zasilania 7.5V, natomiast jeden impuls z enkodera będzie występował co 0.06 mm przejechanej drogi. Prędkość maksymalna jest całkowicie wystarczająca do płynnego i szybkiego rysowania. Dokładność zapewni duża rozdzielczość pomiaru położenia za pomocą enkoderów.

### 2. 3. Mechanizm opuszczania elementu rysującego

Pisak lub marker stosowany do rysowania został umieszczony pionowo w centralnym punkcie robota. Aby umożliwić jego podnoszenie i opuszczanie zaprojektowano konstrukcję mechaniczną, której łańcuch kinematyczny przedstawiono na Rys. 2.4.

PRACA INŻYNIERSKA AGH EAIIB 201415

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

2

3

1

4

Rys. 2.4: Schemat kinematyczny mechanizmu

opuszczania elementu rysującego

Elementem napędowym jest serwomechanizm modelarski (1). Został wybrany ze względu na bezpośrednie sterowanie położeniem ramienia roboczego (2), a nie jego prędkością obrotową, jak w przypadku samego silnika. Zmniejszyło to złożoność konstrukcji i skróciło czas budowy. Ramię robocze jest połączone z prowadnicą elementu rysującego (4) poprzez parę kinematyczną klasy IV (3). Zamienia ona ruch obrotowy ramienia roboczego na pionowy ruch postępowy elementu rysującego.

Elementy mechaniczne zostały zaprojektowane w programie Autodesk Inventor i wykonane z aluminium o grubości 2mm. Wymiary elementów zostały tak dobrane, aby zapewnić odpowiedni zakres ruchów elementu rysującego. Gotowy mechanizm jest widoczny na Rys. 2.5.

Rys. 2.5: Mechanizm opuszczania pisaka

## 2. 4. Złożenie konstrukcji

Konstrukcja mechaniczna została zaprojektowana w całości w programie Autodesk Inventor. Pozwoliło to znaleźć wszystkie błędy i niedopasowania przed wykonaniem każdej części. Model 3D całego robota widoczny jest na Rys. 2.6, a zdjęcie wykonanego robota na Rys. 2.7.

8

9

7

6     10

11

5

4

3

2

1



12

13

Rys. 2.6: Model 3D konstrukcji robota rysującego

Podstawa robota (1), wykonana z półprzezroczystego poliwęglanu grubości 4mm, ma kształt koła o średnicy 202 mm. W jej centralnym punkcie znajduje się otwór wraz z prowadnicą (4) przez który przechodzi pisak (9). Serwomechanizm (5), połączony z mechanizmem opuszczania pisaka (6), zamocowany jest obejmą do prostokątnej ramy (7) wykonanej z aluminium. Rama jest przymocowana bezpośrednio do podstawy śrubami.

Dwa silniki (12) połączone są z podstawą przy użyciu zakupionej w zestawie płytki montażowej i śrub. Koła (13) łączą się z wałem napędowym silników za pomocą aluminiowych piast. Dodatkowymi elementami podporowymi robota są dwie plastikowe kulki toczne w łożyskach (1) umieszczone z przodu i z tyłu robota. Ich wysokość jest regulowana za pomocą podkładek dystansowych.

Na robocie znajduje się również bateria (3) oraz układ elektryczny w ramie mocującej (10) wraz z modułem komunikacyjnym (11). Na szczycie ramy (7) znajduje się układ elek-

PRACA INŻYNIERSKA AGH EAIIB 2014/17

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

tryczny z czujnikami inercyjnymi (8). Układ ten musi znajdować się w znacznej odległości od silników wytwarzających zakłócające pola elektromagnetyczne. Projekt i integracja wspomnianego układu wykracza poza zakres tej pracy. Nie jest on niezbędnym elementem konstrukcji.

Rys. 2.7: Zdjęcie złożonego robota rysującego

### Rozdział 3. Kontroler robota

Jedną z ważniejszych decyzji przy projektowaniu robota jest wybór urządzenia, które będzie go kontrolowało. Wybór ten zależy od całej funkcjonalności, potencjalnych możliwości i zadaniach jakie robot będzie w stanie wykonywać.

W praktyce można spotkać kilka rozwiązań, które dają robotowi coraz większy stopień autonomii. W skrajnym przypadku robot jest całkowicie zdalnie sterowany, a o każdym jego ruchu decyduje komputer lub człowiek z oddali. Nie jest konieczne posiadanie żadnej jednostki obliczeniowej na pokładzie. Jeżeli część zadań, jak utrzymanie zadanej prędkości silników lub wstępna obróbka danych pomiarowych, będą wykonywane na robocie, wtedy konieczne jest zainstalowanie prostego procesora zaprogramowanego do tego celu. Możliwe jest również, że robot sam będzie podejmował ważne decyzje kontrolując sam siebie. W takim przypadku niezbędne jest zainstalowanie wydajnej jednostki obliczeniowej. Projektowany robot rysujący zalicza się do tej ostatniej kategorii.

Rys. 3.1: Komputer Raspberry Pi Rys. 3.2: Podstawowa wersja Arduino Uno

Na rynku istnieje wiele gotowych rozwiązań dla małych i wydajnych jednostek obliczeniowych. Ich możliwości wahają się znacznie od prostych platform Arduino z procesorami Atmel ATmega (Rys. 3.2), do rozbudowanych pełnoprawnych komputerów typu Beagle Bone lub Raspberry Pi (Rys. 3.1). Żadna z nich nie oferuje jednak w swojej podstawowej wersji wszystkich podzespołów niezbędnych do kontroli robota mobilnego.

Szukany kontroler powinien posiadać:

► Wydajny procesor, obsługujący wszystkie elementy mechaniczne, sensory i komunikację z komputerem

PRACA INŻYNIERSKA AGH EAIIB 2014/19

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

► Sekcję zasilania, umożliwiającą bezpośrednie podłączenie baterii

► Kontrolery silników i serwomechanizmu

► Interfejsy komunikacji zarówno przewodowej jak i bezprzewodowej

Jest możliwe skompletowanie jednostki sterującej spełniającej, a nawet przekraczającej, wszystkie powyższe wymagania, lecz wymaga to połączenia ze sobą kilku różnych modułów. Ostateczne rozwiązanie byłoby duże, drogie, nieestetyczne i nad wyraz złożone. Nie umożliwiłoby szerokiej rozbudowy. Z tego względu zdecydowano się na wykonanie własnego kontrolera, który będzie zawierał odpowiednie, wyselekcjonowane elementy, a przez co będzie wydajniejszy, mniejszy i co najważniejsze tańszy.

### 3. 1. Wybór podzespołów

Przed rozpoczęciem prac nad obwodem drukowanym (PCB), należy wybrać wszystkie główne komponenty. Kluczowym elementem jest jednostka obliczeniowa, która powinna sprostać skomplikowanym zadaniom sterowania jak również obsługiwać wszystkie peryferia.

W projekcie należy uwzględnić moduły komunikacyjne oraz układy kontrolujące urządzenia mocy – silniki. Na PCB powinny się również znaleźć obwody odpowiedzialne za dostarczanie odpowiednich napięć zasilających do wszystkich układów.

#### 3. 1. 1. Mikrokontroler

Centralną jednostką obliczeniową robota jest mikrokontroler STM32F407VG, 32-bitowy układ zbudowany na bazie rdzenia ARM, przeznaczony do wymagających systemów wbudowanych wykonujących dużą ilość obliczeń w czasie rzeczywistym. Układ ten integruje w sobie wysoce wydajny rdzeń wraz z jednostką

obliczeń zmiennoprzecinkowych oraz wiele różnorodnych peryferiów. Może on pracować z częstotliwością dochodzącą do 168MHz, wykonując ponad 200 milionów rozkazów na sekundę. [3]

Szybkość działania procesora ma kluczowe znaczenie, gdyż kontroler robota musi sterować wieloma procesami w czasie rzeczywistym, wykonując przy tym złożone obliczenia matematyczne na liczbach zmiennoprzecinkowych. Zapas mocy obliczeniowej będzie mógł być wykorzystany w celu implementacji nie przewidzianych na etapie projektowania funkcjonalności.

Wiele ze zintegrowanych peryferiów procesora jest słabo wykorzystywana. Dwa 32-bitowe zegary wyposażone są w interfejs enkoderów inkrementalnych, co likwiduje potrzebę ich programowej obsługi. Kilkanaście różnych interfejsów komunikacyjnych, wspomaganych kontrolerem DMA, może działać jednocześnie. Dwa z spośród sześciu dostępnych portów szeregowych jest używanych do komunikacji z komputerem. [3] Ważnym powodem wybrania właśnie tego mikrokontrolera jest również jego gotowy port systemu operacyjnego FreeRTOS, który został wykorzystany przy programowaniu. [4]

20 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

### 3. 1. 2. Sekcja zasilania

Robot jest zasilany akumulatorem litowo-polimerowym (LiPol) o napięciu znamionowym 7,4V. Akumulatory tego typu stały się popularne w robotyce przez ich szerokie wykorzystanie w modelarstwie amatorskim. Cechują się wysoką pojemnością elektryczną i wydajnością prądową połączoną z małą masą samej baterii. Ponadto są bardzo tanie i łatwo dostępne w różnych opcjach pojemności i napięcia wyjściowego.

Napięcie znamionowe baterii – 7,4V – podane jest przez producenta jedynie w celach orientacyjnych. W pełni naładowany moduł ma napięcie równe 8,2V, natomiast w głębokim rozładowaniu spada nawet do 6V. Podczas prawidłowej pracy nie powinno spaść poniżej 6,6V. Taki zakres napięć jest odpowiedni do bezpośredniego zasilania silników robota. Jako że prędkością kół steruje osobny zaprogramowany regulator, spadek napięcia podczas rozładowania baterii nie ma zauważalnego wpływu na poprawność pracy. Elementy elektroniczne wymagają jednak mniejszego i bardziej stabilnego napięcia zasilającego.

Sekcja zasilania robota została zbudowana w oparciu o przetwornicę impulsową LM2596-5.0 oraz regulator liniowy SPX1117-3.3.

Przetwornica impulsowa (buck converter) to wysokowydajny układ obniżający napięcie. W tej aplikacji zastosowano ją do zamiany napięcia baterii na stabilne napięcie stałe 5V, niezbędne do zasilania kilku układów scalonych, serwo mechanizmu sterującego położeniem pisaka na robocie oraz drugiej części sekcji zasilania. Zaletami przetwornicy impulsowej są wspomniana wysoka wydajność zamiany napięcia (nawet ponad 95%), czyli małe straty energii i brak wydzielania dużych ilości ciepła. Gdyby w jej miejscu zastosowano regulator liniowy, musiałby on rozproszyć kilka watów mocy. Do wad tego układu należy duża ilość sporych rozmiarów zewnętrznych elementów pasywnych wymaganych do poprawnej pracy. Takie rozwiązanie jest też znacznie droższe. [5]

Zastosowany układ LM2596 wybrano ze względu na mniejszy rozmiar elementów pasywnych niż w klasycznych modelach regulatorów. Jest to spowodowane wysoką częstotliwością przełączania – 150kHz, która zapewnia odpowiednio wysoką impedancję elementów zewnętrznych przy ich mniejszych parametrach znamionowych i rozmiarze. Poza tym, napięcia wejściowe i wyjściowe dla tego układu zawierają się w dopuszczalnych granicach, a wydajność prądowa 3A jest wystarczająca z dużym marginesem bezpieczeństwa. [5]

Drugi stopień sekcji zasilania to stabilizator napięcia SPX1117-3.3 zamieniający stałe napięcie 5V na napięcie 3,3V DC, które jest wykorzystywane przez większość układów scalonych. Nie przewidywano dużego poboru prądu z tej domeny napięciowej, więc poświęcono wysoką wydajność przetwornicy impulsowej na rzecz prostego i małego liniowego regulatora napięcia. Układy tego typu działają jak dzielnik napięcia, gwarantując stabilne napięcie na wyjściu poprzez wytracanie części napięcia wejściowego na sobie. W tej aplikacji regulator wytraca 1,7V. [6]

PRACA INŻYNIERSKA AGH EAIIB 201421

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

### 3. 1. 3. Układy kontrolujące silniki

Standardowym sposobem sterowania silnikiem prądu stałego jest układ mostka H przedstawiony schematycznie na Rys. 3.3. Układ ten pozwala na obrót silnika w obie strony poprzez otwarcie odpowiedniej pary naprzeciwległych tranzystorów. Do kontroli silników wybrano układy scalone VNH3SP30, będące kompletnym mostkiem H wraz z układami sterującymi bramkami tranzystorów MOSFET, zabezpieczającymi oraz wspomagającymi sterowanie prędkością poprzez sygnał PWM.

Rys. 3.3: Schemat elektryczny układu mostka H

Układ VNH3SP30 steruje się pięcioma sygnałami logicznymi. Cztery z nich służą do wyboru kierunku ruchu i włączania lub wyłączenia silnika. Piątym sygnałem jest sygnał PWM do kontroli prędkości obrotowej ruchu. Jak podają dokumentacje techniczne, poziomy napięć logicznych układu VNH3DP30 oraz mikrokontrolera nie są ze sobą zgodne. [3][7] Do konwersji napięć wykorzystano układ ST2378E. Pozwala on na konwersję ośmiu sygnałów logicznych pomiędzy dwoma dowolnymi domenami napięciowymi. Do sterowania dwoma układami VNH3SP30 wymagane jest łącznie 10 sygnałów logicznych. Dwa sygnały, które nie będą mogły być konwertowane układem ST2378E będą konwertowane dwoma dyskretnymi tranzystorami MOSFET analogicznie do Rys. 3.6. [8]

Odczytywanie sygnałów z enkoderów inkrementalnych z integrowanych z silnikiem nie będzie wymagało żadnych układów interfejsowych. Większość wejść I/O mikrokontrolera toleruje napięcia do 5V. [3]

Do kontroli serwomechanizmu kontrolującego pozycję elementu rysującego nie są potrzebne żadne układy scalone. Sterowanie realizowane jest poprzez podanie odpowiedniego sygnału PWM bezpośrednio do serwomechanizmu. Aby dopasować domeny napięciowe zastosowano taki sam układ z tranzystorem MOSFET, jak w przypadku sygnału PWM podawanego na mostki H (Rys. 3.6).

22 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

#### 3. 1. 4. Moduły komunikacyjne

Niezbędnym elementem w robocie mobilnym jest prosty i uniwersalny bezprzewodowy interfejs komunikacyjny. Jest on wykorzystywany między innymi do zdalnego sterowania, wysyłania rozmaitych komend oraz przesyłania danych telemetrycznych. W konstrukcji tego robota wykorzystano moduł WizFi210, będący gotowym rozwiązaniem dla połączeń z siecią WiFi. Moduł ten automatycznie łączy się z wybranym punktem dostępowym sieci bezprzewodowej i otwiera port TCP do którego można się połączyć z dowolnego komputera przy użyciu routera. Wybór podyktowany był prostotą zastosowania gotowego modułu jak również z możliwością komunikacji z dowolnym komputerem obsługującym sieć WiFi bez potrzeby zakupu dodatkowych urządzeń odbiorczych oraz bez instalowania sterowników i programów. Wybrany układ oferuje bogate opcje konfiguracyjne, a producent dostarcza oprogramowanie do zarządzania nim. [9]

Dodatkowym sposobem komunikacji bezprzewodowej jest pilot podczerwieni pracujący

w standardzie RC5. Pilot będzie używany jako podręczne urządzenie do kontroli najważniejszych opcji robota, jego awaryjnego wyłączenia oraz zdalnego sterowania. Układ demodulujący i filtrujący sygnał z pilota to TSOP4836. Usuwa on składową nośną sygnału podczerwonego i wysyła zdemodulowany sygnał do dalszej obróbki w mikrokontrolerze. Dodatkową

zaletą wybranego układu jest filtracja zakłóceń pochodzących między innymi z lamp fluorescencyjnych. [10]

Poza interfejsami bezprzewodowymi zastosowano dodatkowy redundantny interfejs przewodowy do użytku w przypadku awarii sieci bezprzewodowej. W celu ułatwienia obsługi na komputerze, wybrano najpopularniejszy aktualnie interfejs USB. Pomimo tego, że procesor STM32F407 posiada wewnętrzny kontroler tej magistrali, zdecydowano się na użycie zewnętrznego układu FT232R, który działa jako most pomiędzy USB a standardowym i prostym w obsłudze portem szeregowym. Taka decyzja była spowodowana przyspieszeniem okresu prototypowania, gdyż nie trzeba było pisać własnych sterowników do obsługi dość złożonej magistrali USB. Producent układu FT232R dostarcza własne sterowniki, które instalują wirtualny port COM w komputerze PC. W ten sposób komunikacja z procesorem może być zrealizowana z użyciem jednego z wielu gotowych programów komputerowych, jak Putty lub Termite. [11]

### 3. 1. 5. Peryferia

Oprócz wspomnianych wyżej modułów i układów scalonych, do projektu PCB dołączono kilka dodatkowych elementów. Sześć kolorowych diod LED zostało dodanych w celu sygnalizowania pracy programu, wystąpienia określonych zdarzeń lub alarmów. Dodatkowo sześć mikroprzełączników służy do zmian parametrów pracy programu, m.in. wybór aktywnych interfejsów komunikacyjnych. W projekcie został również zawarty miniaturowy potencjometr montażowy, który służy do regulacji wysokości pisaka.

PRACA INŻYNIERSKA AGH EAIIB 201423

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Większość niewykorzystanych nóżek mikrokontrolera została wyprowadzona w formie złączy. Dostępne w ten sposób są między innymi interfejsy szeregowy oraz wyjścia liczników. Będą mogły zostać bezproblemowo użyte w przyszłości.

### 3. 2. Schemat elektryczny

Schemat elektryczny wykonywano w bezpłatnym programie KiCad dostępnym na licencji GNU GPL v2. Przy projektowaniu połączeń wzorowano się na dokumentacjach technicznych układów scalonych oraz na przykładzie projektu płytki ewaluacyjnej dla mikrokontrolera – STM32F4DISCOVERY. [12]

Układ WizFi210 do połączeń z siecią WiFi został już tym etapie odłączony od reszty elektroniki i wykonany w osobnym obwodzie drukowanym jako przyłączany moduł. Taka decyzja była podjęta chęcią wykorzystania relatywnie drogiego układu scalonego w innych projektach. Dodatkowo znacząco zredukowało to rozmiary głównej płytki drukowanej, a poprzez odłączenie go od PCB i podłączenie bezpośrednio do komputera PC możliwa jest jego prosta rekonfiguracja. Schemat elektryczny oraz projekt PCB modułu znajdują się w Dodatku B.

Na schemacie płytki głównej kontrolera robota zawarto wszystkie złącza niezbędne do podłączenia silników, serwo mechanizmu, baterii, modułu WiFi oraz złącze USB. Dołączono wyłącznik zasilania. Przy każdym układzie scalonym i odbiorniku mocy umieszczono kondensatory ceramiczne o wartościach 100nF i 10uF w



celu odseparowania zakłóceń na ścieżkach zasilających.

Wszystkie schematy elektryczne zawarte są w Dodatku A. Poniżej przedstawiono najciekawsze fragmenty związane z kluczowymi funkcjonalnościami.

Rys. 3.4: Schemat elektryczny pierwszego stopnia sekcji zasilania

Na Rys. 3.4 przedstawiony został pierwszy stopień sekcji zasilania elektroniki robota. Układ U1 to przetwornica impulsowa LM2596-5.0. Zgodnie z dokumentacją dobrano odpowiednie wartości cewki L1 i kondensatorów C1 oraz C2, aby pasowały do poziomów napięć wejściowych, wyjściowych oraz do akceptowalnego poziomu odchyłek od napięcia referencyjnego 5V. Cewka L2 oraz kondensator C3 zostały dodane w celu silniejszej filtracji zakłó-

24 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

ceń poziomu napięcia. Dioda Schottky'ego D1 jest konieczna do blokowania napięć ujemnych pojawiających się na cewce L1 podczas przełączania układu przetwornicy. [5]

Rys. 3.5: Schemat elektryczny drugiego stopnia sekcji zasilania

Rys. 3.5 zawiera schemat drugiego stopnia sekcji zasilania złożonego z liniowego regulatora napięcia. Dwa kondensatory tantalowe C4 i C5 zostały wybrane zgodnie z zaleceniami dokumentacji technicznej. Jak można zauważyć, ilość elementów w tej sekcji jest dużo mniejsza niż w sekcji pierwszej. Również powierzchnia obwodu drukowanego zajmowanego przez obie części jest znacząco różna. Drugi stopień zajmuje ponad pięciokrotnie mniej powierzchni niż pierwszy. [6]

Rys. 3.6: Układ kondycjonowania sygnału sterującego serwomechanizmem

Na Rys. 3.6 przedstawiony jest układ zamiany poziomu napięcia sterującego serwomechanizmem. Serwomechanizm zasilany jest napięciem 5V i powinien być sterowany sygnałem logicznym w tej samej domenie napięciowej. Mikrokontroler działa natomiast przy napięciu 3,3V. Aby podnieść poziom wykorzystano prosty jednotranzystorowy układ inwertera. Wysoki sygnał logiczny podany na bramkę G tranzystora Q1 powoduje otwarcie kanału D-S. Napięcie na tranzystorze spada do niskiego poziomu, który jest podawany na złącze K1 serwomechanizmu. Analogicznie poprzez podanie logicznego sygnału '0' na bramkę G, tranzystor przechodzi w stan zatkania powodując powstanie napięcia wysokiego na drenie D.

Identyczne układy zostały wykorzystane do podniesienia napięcia sygnałów PWM podawanych na układy VNH3SP30, o czym było wspomniane w rozdziale 3.1.3.

### 3. 3. Projekt PCB

Projekt obwodów drukowanych został wykonany w programie KiCad. Najpierw dobrano odpowiednie obudowy do każdego elementu elektronicznego. Następnie dokonano wstępnego rozmieszczenia elementów i wykonano procedurę automatycznego prowadzenia ścieżek (autorouting). Iteracyjnie poprawiano fizyczne ułożenie elementów tak, aby zmniejszyć sumaryczną długość ścieżek i ilość przelotek tworzonych przez autorouter.

Przy rozmieszczaniu elementów brano pod uwagę ograniczenia i podstawowe zasady projektowe. Kondensatory separujące muszą się znajdować jak najbliżej źródła zakłóceń. Niektóre elementy, jak złącze baterii, włącznik i złącze USB, powinny znajdować się przy krawędzi płytki w celu łatwego dostępu do nich. Sekcja zasilania powinna być oddalona od pozostałych elementów, gdyż przepływają przez nią duże prądy i wytwarza sporą ilość zakłóceń elektromagnetycznych. Na rogach płytki powinno się znaleźć miejsce na otwory montażowe.

Po zadowalającym rozmieszczeniu wszystkich elementów, ręcznie poprawiono prowadzenie ścieżek oraz sprawdzono poprawność reguł projektowych w programie KiCad. Finalna wersja jest widoczna na Rys. 3.7. Zaprojektowana płytka drukowana ma wymiary 82,6 mm x 69 mm.

Rys. 3.7: Widok projektu płytki drukowanej

Jak wspomniano w rozdziale 3. 2 moduł WiFi znajdował się na osobnym PCB. Jego projekt jest dość prosty i polegał na doprowadzeniu niezbędnych pinów do złącza na krawędzi

26 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

płytki oraz na podłączeniu trzech diod LED wskazujących status. Projekt PCB tego modułu został zawarty w Dodatku B.

### 3. 4. Wykonanie elektroniki

Wygenerowane w programie KiCad pliki projektowe Gerber zostały wysłane do firmy zajmującej się produkcją obwodów drukowanych w małych ilościach. Nie było możliwe samodzielne wykonanie tak złożonego układu bez użycia profesjonalnych technologii.

Wszystkie elementy elektroniczne zostały przylutowane przy użyciu grotowej stacji lutowniczej oraz lutownicy na gorące powietrze. Poprawność połączeń była na bieżąco sprawdzana w powiększeniu oraz testowana miernikiem uniwersalnym w poszukiwaniu zwarc i braku połączeń. Zdjęcie wykonanego obwodu wraz z przylutowanymi elementami znajduje się na Rys. 3.8.

Układ konwersji  
Sekcja zasilania  
poziomów napięć

Mostki H  
Złącze USB

Złącze

modułu

WiFi

Procesor

Odbiornik podczewieni

Rys. 3.8: Zdjęcie wykonanej płytki drukowanej z zaznaczonymi najważniejszymi elementami

### 3. 5. Uruchomienie

Sprawdzenie działania poszczególnych elementów wyko nano pisząc proste programy testowe na mikrokontroler. Rozpoczęto od nawiązania połączenia między mikrokontrolerem

aprogramatorem podłączonym do komputera. Jako programator używana była płytka proto-typowa STM32F4DISCOVERY z wbudowanym programatorem ST-Link V2. Początkowo nie udawało się nawiązać komunikacji. Było to spowodowane błędem przylutowania oporni-

PRACA INŻYNIERSKA AGH EAIIB 201427

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJ ĄCEGO

ka podciągającego (pull-up) R102 przy nó Źce resetu mikrokontrolera. Rezystor ten oznaczony był na schemacie jako DNP (Do Not Populate), czyli nie należało go umieszczać na PCB. Po jego usunięciu programowanie przebiegało pomyślnie.

W pierwszej kolejności uruchomiono połączenie szeregowe z komputerem z wykorzystaniem przewodowego interfejsu komunikacyjnego USB. Umożliwiło to komunikację z działającym programem z komputera. Następnie sprawdzono poprawność generacji sygnałów kontrolujących silniki i serwomechanizm, poprawne działanie diod LED i mikroprzełączników oraz uruchomiono bezprzewodowe interfejsy komunikacyjne.

Po sprawdzeniu poprawnej pracy wszystkich elementów przystąpiono do pisania właści-

wego oprogramowania.



## Rozdział 4. Oprogramowanie

Oprogramowanie robota rysującego zostało napisane w języku C w standardzie C99. Kod kompilowano z użyciem kompilacji skróconej kompilatorem gcc-arm-none-eabi1 w wersji 4.7. Wybrany zintegrowany środowiskiem programistycznym zostało CooCox CoIDE2 oferujące bezpośrednio wsparcie dla użytego mikrokontrolera oraz programatora ST-Link V2. Przy pisaniu programu wykorzystano bibliotekę obsługującą peryferia mikrokontrolera CMSIS-CORE (Standard Peripherals Drivers v1.1.0) oraz bibliotekę CMSIS-DSP zawierającą specjalnie zoptymalizowane algorytmy cyfrowego przetwarzania sygnałów. Obie biblioteki dostarczane są przez producenta układu scalonego<sup>3</sup>.

### 4. 1. System operacyjny czasu rzeczywistego

System zaprojektowany został na bazie systemu operacyjnego czasu rzeczywistego FreeRTOS. Użycie systemu operacyjnego zapewniło znaczne uproszczenie architektury programu, co skutkowało zmniejszeniem ilości błędów w kodzie i przyspieszeniem prac programistycznych. Systemy czasu rzeczywistego zapewniają spełnienie zadanych wymagań czasowych, co było pożądane przy implementacji regulatorów, układów akwizycji danych pomiarowych i procesów równoległych. Napisanie programu o podobnej funkcjonalności bez jego użycia byłoby bardzo trudne, jeżeli nie niemożliwe.

FreeRTOS jest profesjonalnym i popularnym systemem operacyjnym czasu rzeczywistego (RTOS) oficjalnie wspierającym 34 architektury procesorów (w tym ARM Cortex-M4F). Wyróżnia go ogromna ilość doskonale udokumentowanych funkcjonalności oraz mały rozmiar skompilowanego kodu i niskie użycie pamięci RAM (tzw. footprint). Dodatkową zaletą jest dostępność wielu przykładów zarówno na stronie producenta, jak i w całym Internecie. FreeRTOS udostępnia takie struktury programistyczne jak zadania (task), kolejki (queue), semafony (semaphore), blokady (mutex) i sekcje krytyczne. Wspiera stosowanie przerwań systemowych, a co ważniejsze umożliwia bezproblemowe korzystanie z jednostki zmiennoprzecinkowej, nawet wewnątrz procedur obsługi przerwań, na procesorze z serii STM32F4. [4]

<sup>1</sup>Strona producenta: <https://launchpad.net/gcc-arm-embedded/>

<sup>2</sup>Strona producenta: [http://www.coocox.org/CooCox\\_CoIDE.htm](http://www.coocox.org/CooCox_CoIDE.htm)

<sup>3</sup>Produkt STSW-STM32065 z [www.st.com/web/catalog/tools/FM147/CL1794/SC961/SS1743/PF257901](http://www.st.com/web/catalog/tools/FM147/CL1794/SC961/SS1743/PF257901)



#### 4. 2. Warstwa abstrakcji sprzętu

Programowanie rozpoczęto od napisania warstwy abstrakcji sprzętu, czyli zbioru funkcji do kontroli elementów mechanicznych, odczytu czujników i ustawiania wartości sygnałów logicznych. Było to konieczne, aby w reszcie oprogramowania używać tych standardowych funkcji zamiast niskopoziomowych odwołań. Zadania systemu operacyjnego mogą pracować całkowicie niezależnie od fizycznej budowy i organizacji peryferii procesora i elektroniki. Warstwa abstrakcji sprzętu obejmuje między innymi:

- ▶ Ustawianie współczynnika wypełnienia sygnałów PWM kontrolujących silniki i serwo-mechanizm
- ▶ Ustawienie sygnałów logicznych kontrolujących kierunek obrotu silników lub ich hamowanie
- ▶ Odczytywanie pozycji enkoderów
- ▶ Odczytywanie aktualnego napięcia baterii
- ▶ Zapalanie i gaszenie sześciu diod LED
- ▶ Odczyt stanu sześciu mikroprzełączników
- ▶ Ustawianie sygnałów logicznych kontrolujących moduł WiFi

Sygnał PWM kontrolujący silniki generowany jest przy użyciu sprzętowego licznika. Częstotliwość sygnału ustalono na standardowe 10kHz. Jest to maksymalna wartość jaką podaje producent układów VNH3SP30. [7] Sygnały PWM podawane na silniki są zsynchronizowane względem siebie aby zapewnić jak najmniejszy maksymalny pobór mocy z baterii. Zostało to przedstawione na Rys. 4.1. Chwile kiedy oba sygnały mają logiczną wartość '1', zaznaczone szarym kolorem, są minimalizowane, a przez to czas gdy silniki pobierają prąd również jest minimalizowany. Dzięki takiemu rozwiązaniu na baterii nie pojawiają się wysokie wahania napięcia związane z dużą zmianą pobieranego prądu. Poprawność generacji sygnałów PWM została sprawdzona w rozdziale 5. 5.

'1'

FALA A

'0'

'1'

FALA B

'0'

Rys. 4.1: Schemat synchronizacji sygnałów PWM podawanych na silniki

Sygnał PWM sterujący serwomechanizmem jest również generowany z wykorzystaniem sprzętowego licznika. Parametry tego sygnału muszą być zgodne ze specyfikacją mode-

30 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

larskich serwomechanizmów analogowych. Częstotliwość sygnału PWM wynosi 50Hz, a współczynnik wypełnienia waha się od 2.5% do 11.5%, co odpowiada dwóm skrajnym położeniom ramienia roboczego. Wartości współczynnika wypełnienia zostały sprawdzone na rzeczywistym serwomechanizmie i odpowiednio dostrojone. Wartości skrajne zostały wpisane do programu.

Enkodery inkrementalne obsługiwane są przez sprzętowy interfejs bazujący na dwóch 32-bitowych licznikach. Dzięki temu, że wybrany mikrokontroler posiada taki interfejs możliwe było znaczne odciążenie programu z obsługi kilkudziesięciu tysięcy przerwań procesora na sekundę. Wartości początkowe obu liczników zostały ustawione na liczbę 0x8000000 w zapisie heksadecymalnym, co odpowiada środkowi ich zakresów. Przepiętnie licznika mogłoby nastąpić po otrzymaniu 2 147 483 648 impulsów, co odpowiada przejechaniu 131 km. Ewentualność przepiętnienia jest nieobsługiwana w kodzie programu z uwagi na rzeczywistą niemożliwość jej wystąpienia.

Napięcie baterii odczytywane jest przez wbudowany w mikrokontroler przetwornik analogowo-cyfrowy. Odczytana wartość jest skalowana do jednostek napięcia i zapisywana do globalnej zmiennej. Dodatkowo wykorzystano funkcję generowania przerwania procesora w chwili gdy zmierzone napięcie spadnie poniżej zadanej wartości. Służy to zabezpieczeniu baterii przez rozładowaniem, co prowadziłoby do jej zniszczenia. W chwili wystąpienia takiego przerwania wyłączone zostają silniki i serwomechanizm, a program zatrzymuje się, aby uniemożliwić ich powtórne włączenie. Konieczna jest wymiana lub doładowanie baterii.

Odczyt stanu mikroprzełączników wykonywany jest w procedurach obsługi przerwań. Przerwania są generowane po wykryciu zmiany poziomu logicznego na odpowiednich nóżkach mikrokontrolera. W wyniku błędu przy projektowaniu płytki PCB nie dodano kondensatorów filtrujących przy każdym mikroprzełączniku. Spowodowało to występowanie zjawiska tzw. „drgania styków” i pojawianie się wielu przerwań przy jednokrotnym przełączeniu mikroprzełącznika. Identyfikacja tego problemu została przeprowadzona w rozdziale 5.6. Negatywny wpływ „drgania styków” wyeliminowano programowo poprzez wyłączenie obsługi przerwań pochodzących od przełączników na zadany czas po pojawieniu się pierwszego przerwania. Zaimplementowano to z wykorzystaniem wbudowanego licznika.

4.3. Zadania i komunikacja między nimi

Całe oprogramowanie zostało podzielone na zadania działające wspólnie pod kontrolą systemu

operacyjnego (OS). Każde z nich ma przydzielony jeden z pięciu priorytetów wykonania odpowiadający „ważności” zadania. Zadania o wyższym priorytecie mają pierwszeństwo przed innymi.

Zadania komunikują się ze sobą za pomocą semaforów i kolejek FIFO należących do zestawu standardowych funkcji OS. Struktury te są bezpieczne do użytku z różnych miejsc programu wielowątkowego.

PRACA INŻYNIERSKA AGH EAIIB 201431

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Stworzono następujące zadania:

- ▶ Zadanie kontroli silników, w którym zaimplementowano regulator PID prędkości obrotowej kół
- ▶ Zadanie generatora ruchu, które wykonuje komendy jazdy po prostej lub łuku
- ▶ Zadanie kontroli pisaka, kontrolujące ruch pisaka, w tym jego podnoszenie, opuszczanie i rysowanie różnych rodzajów linii
- ▶ Zadanie telemetrii, które integruje informacje o położeniu pochodzące z różnych źródeł
- ▶ Zadania komunikacyjne, w tym odbierające znaki z różnych interfejsów, wysyłające dane tekstowe na interfejsy
- ▶ Zadania obsługi komend systemowych, które zostały odczytane przez jeden z interfejsów komunikacyjnych lub z pilota podczerwieni
- ▶ Dodatkowe zadania pomocnicze, np. kontrolujące poprawną pracę systemu operacyjnego

Wszystkie powyższe zadania, ich struktura i wykorzystane algorytmy, zostaną opisane w kolejnych rozdziałach.

#### 4. 3. 1. Zadanie kontroli silników

Zadanie kontroli silników zawiera implementację dwóch regulatorów PI regulujących prędkość obrotową silników. Regulator PI (proporcjonalno-całkujący) jest jednym z wariantów pełnego regulatora proporcjonalno-całkującego różniczkującego PID. Tego typu regulator jest jednym z podstawowych stosowanych w technice i zapewnia dobre własności regulacji. Dyskretną wersję pozycyjną tego regulatora opisuje wzór

=

gdzie:

- wzmacnienie proporcjonalne – stała czasowa cz ęści całkującej
- stała czasowa cz ęści różniczkującej – okres próbkowania
- uchyb regulacji w czasie
- wypracowane sterowanie w czasie

+

+

( - )

32 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Wzór ten nie jest prosty w implementacji z uwagi na jego nierekursywny charakter (suma wszystkich poprzednich pomiarów). Wersję przyrostową regulatora można otrzymać po prostych przekształceniach:

= + + +

gdzie przy zachowaniu powyższych oznaczeń:

= 1 + = - 1 + 2 - =

Wzór ten jest efektywny w implementacji na procesorach. [14]

Regulator w tej formie, dostępny w bibliotece CMSIS-DSP, został wykorzystany do kontroli prędkości obrotowej silników. Jako wartość zadana do regulatora została podana żądana prędkość obrotowa w rad/s. Aktualny błąd prędkości był obliczany na podstawie różnicy odczytu pozycji enkoderów pomiędzy okresami próbkowania zgodnie ze wzorem:

$= (- )$   
 $2\pi$   
1000  
- z%

gdzie:

3592  
10

&  
– wartość zadana prędkości obrotowej silnika

– pozycja enkodera w  
'-tym okresie próbkowania odczytana w jednostkach impulsów

(  
– współczynnik przeliczający ilość impulsów podczas jednego obrotu silnika na radiany

)\*+

– współczynnik przeliczający okres próbkowania (10ms) na sekundy.

Regulator na swoim wyjściu wyznacza współczynnik z zakresu [-1.0: 1.0] będący

współczynnikiem wypełnienia sygnału PWM podawanego na silniki. Znak wyniku odpowiada-

da za kierunek obrotu silnika. Liczba ta jest bezpośrednio używana w funkcji ustalającej prędkość silnika należącej do warstwy abstrakcji sprzętu.

Okres próbkowania został ustalony na 10ms, a priorytet zadania kontroli silników jest najwyższy, aby

zapewnić stałość okresu. Nastawy regulatora zostały dobrane metodą Zieglera-Nicholsa [13] i zoptymalizowane metodą prób i błędów. Okres próbkowania został dobrany w wyniku kompromisu pomiędzy dużą szybkością działania regulatora a ilością impulsów enkodera w okresie próbkowania. Niska ich ilość ogranicza znacznie rozdzielczość obliczania aktualnej prędkości i prowadzi do powstania oscylacji.

Zadanie kontroli silników współpracuje z innymi zadaniami. Wysyła informacje służące do estymacji położenia do zadania telemetrii (4.3.4) oraz komunikaty tekstowe o aktualnej prędkości silników do zadania komunikacyjnego (4.3.5). Cztery zadania wysyłają żądania ustalenia nowej prędkości silników.

PRACA INŻYNIERSKA AGH EAIIB 201433

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

#### 4.3.2. Zadanie generatora ruchu

Generacja ruchu to kontrola prędkości silników tak, aby robot jechał po określonej, zadanej ścieżce. Zaimplementowano cztery rodzaje ruchów:

- ▶ Jazda do po linii prostej na określoną odległość
- ▶ Obrót w miejscu o określony kąt lub do osiągnięcia podanej orientacji
- ▶ Jazda po łuku
- ▶ Dojeżdżanie do punktu

Każdy z wymienionych ruchów może być wykonany z pisakiem opuszczonym lub podniesionym, a w połączeniu z zadaniem kontroli pisaka (4.3.3) mogą być rysowane różne rodzaje linii.

Komendy wykonania określonego typu ruchu wysyłane są z komputera w zstandaryzowanej postaci odpowiadającej formatowi ramki na Rys. 4.5 w rozdziale 4.3.5. Każda komenda ruchu zawiera typ ruchu, flagę czy pisak powinien być opuszczony czy nie, prędkość ruchu oraz dwa parametry, których znaczenie zmienia się w zależności od typu ruchu. Zadanie wykonujące komendy po napotkaniu komendy generatora ruchu wysyła ją dalej do zadania generatora ruchu, gdzie jest ostatecznie wykonywana. Komendy są odbierane przez kolejkę FIFO. Wykonanie kolejnej rozpoczyna się po skończeniu poprzedniej, ale można z komputera wysłać kilkanaście na raz.

Zadanie generatora ruchu, po sprawdzeniu rodzaju ruchu, wywołuje odpowiednią funkcję z regulatorem. Wykonanie tej funkcji kończy się, gdy robot zakończy jazdę. Napisane regulatory zostaną przedstawione poniżej.

Na Rys. 4.2 widać schemat blokowy algorytmu jazdy po linii. Ideą algorytmu jest ustalenie prędkości silników na podaną wartość i czekanie na przejechanie zadanego dystansu. Od odległości 10 cm od celu robot zwalnia, aby płynnie się zatrzymać na końcu trasy. Wzór na prędkość kół na końcowym odcinku trasy przedstawia się następująco:

$$0 = 01 \cdot 0.2 + 0.8 \cdot 100 \Delta 4$$

gdzie:

0 - prędkość kół

01 - maksymalna zadana prędkość kół w m/s

$\Delta 4$  - pozostały do przejechania dystans w milimetrach



Rys. 4.2: Schemat blokowy algorytmu jazdy po linii

Algorytm obrotu w miejscu o zadany kąt działa bardzo podobnie do algorytmu jazdy po linii. W tym wypadku jednak nie jest regulowana przejechana odległość, ale kąt orientacji robota. Algorytm pozwala na obrót o zadany kąt lub do zadanej orientacji (przy nieznajomości orientacji początkowej).

PRACA INŻYNIERSKA AGH EAIIB 201435

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Rys. 4.3: Schemat blokowy algorytmu obrotu

Schemat blokowy algorytmu znajduje się na Rys. 4.3. W pierwszej kolejności wykonywane są obliczenia kąta obrotu w przypadku, gdy komenda dotyczyła obrotu do podanej orientacji. Następnie liczone są maksymalne prędkości obu kół z uwzględnieniem kierunku obrotu. Jeżeli do przejechania pozostało więcej niż 45 stopni, to robot obraca się z pełną prędkością. Gdy zbliży się na mniej niż 45 stopni do zadanego kąta orientacji, zwalnia, aby zatrzymać się płynnie. Wzór na prędkość kół przy zwalnianiu przedstawia się następująco:

$$v = v_1 \cdot 0.1 + 0.9 \cdot \Delta\theta / 45$$

gdzie:

$v$  – prędkość koła

$v_1$  – maksymalna prędkość koła (obliczona na podstawie zadanej prędkości) w m/s  $\Delta\theta$  – pozostały kąt do obrotu w stopniach

36 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Algorytm jazdy po łuku bardzo mocno opiera się o algorytm obrotu w miejscu. W istocie oba regulatory zostały zaimplementowane w jednej funkcji. Jedyną istotną różnicą jest wzór na prędkości maksymalne obu kół. Prędkości te wyrażają się wzorami:

$$v_0 = v_1 - \frac{r}{R} \cdot \omega$$

$$v_1 = v_1 + \frac{r}{R} \cdot \omega$$

gdzie:

$v_0, v_1$  - prędkości maksymalne odpowiednio lewego i prawego koła

$v$  - zadana prędkość jazdy w m/s

$\omega$  - parametr równy 1 gdy kąt obrotu jest dodatni, -1 w przeciwnym przypadku ;  $r$  - średnica robota - odległość między kołami  $R$  - promień łuku po którym robot powinien się poruszać

Poza powyższą zmianą, algorytm jazdy po łuku nie różni się od algorytmu obrotu.

Algorytm dojeżdżania do punktu jest najbardziej złożony. Jego bardzo uproszczony schemat blokowy znajduje się na Rys. 4.4. Na początku robot obraca się w miejscu w kierunku celu, a następnie włącza się specjalny regulator doprowadzający go do celu. Przed końcem trasy zmniejszana jest prędkość, aby robot powoli się zatrzymał. Wspomniany regulator bazuje na algorytmie opisanym w pracy [14], który został dość znacznie zmodyfikowany na potrzeby zadania dojeżdżania do punktu. Zaletami algorytmu jest zerowanie błędu orientacji robota w stosunku do celu, co powoduje, że robot nie mija punktu docelowego przejeżdżając obok.

Regulator, już w formie pseudokodu C, przedstawia się następująco:

```

const float k = 0.0003;
// Wzmocnienie regulatora
float dx
=
zadaneX
-
aktualneX;
// Odległość
od celu
wzdłuż osi X
float dy
=
zadaneY
-
aktualneY;
// Odległość
od celu
wzdłuż osi Y
float cosfi = cos(aktualnaOrientacja);
//
Cosinus kąta orientacji robota
float sinfi = sin(aktualnaOrientacja);
//
Sinus kąta orientacji robota

float ey = -sinfi*dx + cosfi*dy;
// Odległość celu prostopadła do
osi kół
float v = zadanaPredkosc;
// Prędkość postępową środka
robota
float w
= k
* zadanaPredkosc * ey;
// Prędkość obrotowa robota – tym

//
większa im większe ey
*/

if (d <
50)
{
//
redukcja prędkości na
końcu trasy
v = v * (0.7 * d / 50 + 0.3);

```

```
}  
if (wartoscBezwzględna(w) > zadanaPredkosc / 2) {  
w = znak(w) * command->Speed / 2; // ograniczenie prędkości obrotowej  
  
}  
  
float predkoscLewegoKola = v - w * SREDNICA_ROBOTA / 2; float predkoscPrawegoKola = v + w *  
SREDNICA_ROBOTA / 2;
```

PRACA INŻYNIERSKA AGH EAIIB 201437

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Rys. 4.4: Schemat blokowy algorytmu dojeżdżania do punktu

Poprzez połączenie komend dojeżdżania do punktu oraz obrotu o kąt absolutny można uzyskać dowolną

pozycję w przestrzeni dwuwymiarowej.

Dodatkową funkcjonalnością jest możliwość ustawienia skali w jakiej rysowane będą kształty. Skala ta wpływa na wartości współrzędnych  $x$  i  $y$  roboty jakie podawane są do funkcji rysujących. Można ją zmienić za pomocą komendy wysłanej do robota i obsługiwanej przez zadania komunikacyjne (4.3.5).

#### 4.3.3. Zadanie kontroli pisaka

Kontrola pisaka została dopisana na późnym etapie rozwoju oprogramowania. Było to podyktowane potrzebą dodania dodatkowej funkcjonalności obejmującej rysowanie linii nie tylko ciągłych, ale również z kropkowanych, kreskowanych i innych.

Zadanie kontroli pisaka obejmuje kolejkę FIFO przez którą inne zadania wydają komendy opuszczenia (rozpoczęcia rysowania) lub podniesienia pisaka. Zaimplementowano dodatkowe funkcje pozwalające na zmianę stylu rysowanej linii w każdym momencie.

Algorytm rysowania linii o dowolnej charakterystyce wykonany został przy użyciu czterostanowej maszyny stanów. Program monitoruje pozycję robota i oblicza odległość przez niego przebytą. Jeżeli przekroczy ona predefiniowany dystans pisak jest podnoszony lub opuszczany w zależności od aktualnego stanu. Dla przykładu linia typu kropka-kreska opisana jest następującymi stanami:

1. 1cm jazdy z pisakiem opuszczonym

2. 25cm jazdy z pisakiem podniesionym

38 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

3. 30cm jazdy z pisakiem opuszczonym

4. 25cm jazdy z pisakiem podniesionym

Przebyta odległość obliczana jest poprzez sumowanie odległości przebytych pomiędzy kolejnymi okresami próbkowania zadania (10ms).

#### 4.3.4. Zadanie telemetrii

Zadanie telemetrii jest zadaniem integracyjnym. Jego głównym celem jest aktualizowanie zmiennych opisujących bieżącą pozycję robota na płaszczyźnie na podstawie informacji uzyskiwanych z innych zadań. W momencie pisania tej pracy jedynie zadanie kontroli silników wysyła takie informacje, jednak w przyszłości planowane jest dodanie większej liczby sposobów pozycjonowania (w tym czujniki inercyjne) zwiększających ogólną dokładność.

Zadanie kontroli silników wysyła aktualizacje pozycji w formie przyrostów, to znaczy że w każdym okresie

próbki wbudowanego regulatora wysyłana jest informacja o zmianie położenia (wzdłuż osi A i A@) oraz kąta orientacji. Dane te obliczane są na podstawie danych

z enkoderów, a cały proces nosi nazwę odometrii. Równania odometrii na podstawie [2] [16] przedstawiają się następująco:

$$C(D+1) = C(D) \cdot \cos(L7(D) \cdot \Delta D) + M(D+1) \cdot \Delta D \cdot \sin(L7(D) \cdot \Delta D)$$

$$G(D+1) = G(D) \cdot \sin(L7(D) \cdot \Delta D) + M(D+1) \cdot \Delta D \cdot \cos(L7(D) \cdot \Delta D)$$

$$\theta(t+1) = \theta(t) + \Delta \theta$$

$$\Delta \theta = \Delta D \cdot M(D+1)$$

gdzie:  $\theta(D)$  - kąt  
 $M(D)$  -

A ponadto:

$$\theta(D) = \theta(0) + \int_0^D \omega(t) dt$$

gdzie:

$\omega(D)$  - prędkość lewego koła w chwili czasu D

; - odległość między kołami (średnica robota) = prędkość prawego koła w chwili czasu

Dokładność tej metody pozycjonowania zależy bardzo od dokładności wymiarów fizycznych robota (średnica kół i rozstaw osi). Odometria jest wrażliwa na wszelkie błędy nie-systematyczne, jak nierówności terenu lub poślizg kół. Występowanie tych czynników powoduje powstanie akumulującego się błędu, którego nie można poprawić polegając jedynie na tym sposobie pozycjonowania. Jest to spowodowane brakiem stałego punktu odniesienia dla pozycji oraz całkowitym charakterem powyższych równań. Zniwelowanie błędów możliwe byłoby przy zastosowaniu dodatkowego bezwzględnie sposobu pozycjonowania. [2]

PRACA INŻYNIERSKA AGH EAIIB 201439

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

#### 4.3.5. Zadania komunikacyjne

Do sprawnego działania komunikacji stworzono kilka ściśle współpracujących ze sobą zadań. Odbieranie danych i ich wysyłanie są od siebie całkowicie niezależne i zaimplementowane w całkowicie różny sposób.

Odbieranie danych z komputera rozpoczyna się w procedurach obsługi przerwań procesora. Zaimplementowano dwa takie przerwania, dla interfejsu przewodowego USB oraz dla interfejsu bezprzewodowego WiFi. Każde z nich po odebraniu jednego znaku pakuje go do struktury zawierającej pole wskazujące na źródło, a następnie wysyła do kolejki FIFO do zadania odbierającego (TaskInputBuffer). Zadanie to pełni rolę buforów odebranych znaków oraz implementuje maszyny stanów do wyodrębniania ramek komunikacyjnych. Każda ramka ma strukturę pokazaną na Rys. 4.5. Dane odebrane pomiędzy ramkami są ignorowane.

Separator      Parametry specyficzne dla typu komendy

<C:1:6:c:6.45>

Znak początku ramki      Znak końca ramki

Typ komendy

Rys. 4.5: Format ramki danych odbieranych z komputera

Po odebraniu całej ramki danych, zadanie TaskInputBuffer wysyła komendę, zawartą pomiędzy znakami początku i końca ramki, do kolejki FIFO zadania TaskCommandHandler, które zajmuje się interpretacją i wykonaniem komend. To w nim sprawdzana jest dopiero poprawność przekazanych parametrów i wykonywane są zadania związane z określoną komendą.

Zakres możliwości komend obejmuje zmianę parametrów regulatorów, wyłączenie i włączenie różnych modułów, wykonywanie akcji czy wypisywanie aktualnego stanu robota.

Wysyłanie danych do komputera rozwiązane zostało w zupełnie inny sposób. Jako że kilka zadań mogłoby jednocześnie żądać wysyłania, konieczne było zaimplementowanie mechanizmów wzajemnego wykluczania. Stworzono zadanie TaskPrintfConsumer, które udostępnia funkcję safePrint. Funkcja ta bazuje na standardowej funkcji języka C – printf – wysyłającej łańcuchy znaków na ekran. Funkcja safePrint formatuje dane w taki sam sposób jak printf, a wynikowy łańcuch umieszcza w kolejce FIFO zadania TaskPrintfConsumer. W ten sposób kilka zadań może jednocześnie pisać do komputera.

Funkcja zadania TaskPrintfConsumer ogranicza się do jak najszybszego opróżnienia kolejki przesyłając dane poprzez fizyczne interfejsy procesora. Aby zwiększyć efektywność użyte zostały strumienie DMA eliminujące udział procesora w procesie przesyłania. Po wy-

śnaniu każdego łańcucha znaków kontroler DMA wyzwala przerwanie, w którym z kolei wystawiany jest semafor informujący zadanie TaskPrintfConsumer o zakończeniu transferu.



Do obsługi komend pochodzących z pilota stworzone zostało osobne zadanie z uwagi na znaczną odmiennosc charakteru przychodzących danych. Do dekodowania sygnału przycho-dzącego z odbiornika podczerwieni została wykorzystana gotowa biblioteka firmy STMicro-electronics. Gdy zostanie odebrany sygnał naciśnięcia przycisku na pilocie wystawiany jest semafor, który informuje zadanie odczytują ce TaskRC5 o nowej dostępnej komendzie. Już zdekodowana komenda jest interpretowana i wykonywana jest skojarzona z nią akcja. Naj-bardziej użytecznymi akcjami są komendy pozwalające na zdalne sterowanie robotem oraz żądanie resetu procesora.

PRACA INŻYNIERSKA AGH EAIIB 201441

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Rozdział 5. Testy

### 5. 1. Dokładność rysowania kształtów

Celem tego testu jest sprawdzenie poprawności wykonywania komend przesyłanych bez-przewodowo do robota oraz zweryfikowanie dokładności rysowania.

Przygotowano następującą sekwencję komend, które zostały wysłane do robota:

<W:5000> odczekaj 5000ms  
<C:l:1:0.1:200> jazda do przodu 200mm z prędkością 0.1m/s i pisakiem opuszczonym  
  
<C:d:1:0.1:0:90> obrót w miejscu o 90 stopni w lewo  
<C:l:1:0.1:200> do przodu 200mm  
<C:d:1:0.1:0:90> obrót o 90 stopni w lewo  
  
<C:l:1:0.1:200> do przodu 200mm  
<C:d:1:0.1:0:90> obrót o 90 stopni w lewo  
<C:l:1:0.1:200> do przodu 200mm  
  
<C:d:0:0.1:0:90> obrót o 90 stopni w lewo, bez pisaka

<C:l:0:0.1:100> do przodu 100mm, bez pisaka  
<C:a:1:0.08:100:360> jazda po łuku o promieniu 100mm, 360

stopni z prędkością 0.08m/s z  
pisakiem opuszczonym

<C:p:0:0.05:200:0> jazda do punktu X=200mm Y=0mm bez

pisaka

<C:d:0:0.05:1:45> obrót w miejscu aż do uzyskania  
orientacji 45 stopni, bez pisaka

<q> poczekaj na zakończenie jazdy

<d:2> zmień pisak na linię kreskowaną

<C:a:1:0.1:141.42:360> jazda po łuku o promieniu 141.42mm,

360 stopni z prędkością 0.1m/s

<q> poczekaj na zakończenie jazdy

<d:4> zmień pisak na linię ciągłą

<C:p:0:0.1:-100:-100> do punktu X=-100mm Y=-100mm bez pisaka

<C:d:0:0.1:1:45> obrót do orientacji 45 stopni

Powyższa sekwencja powinna spowodować narysowanie kwadratu z kołem wpisanym w niego oraz drugim kołem opisanym na nim i narysowaną linią przerywaną. Wynikiem działania programu był rysunek widoczny na Rys. 5.1.

Rys. 5.1: Rysunek powstały po wykonaniu testowej sekwencji poleceń

Program wykonał się poprawnie i wszystkie kształty zostały dokładnie narysowane. Sekwencja testowa sprawdziła wszystkie regulatory zaimplementowane w zadaniu generatora ruchu (rozdział 4. 3. 2). Zadanie kontroli pisaka poprawnie zinterpretowało komendy zmiany stylu rysowanej linii i powstała równa linia przerywana. Zadania komunikacyjne poprawnie obsłużyły przychodzące komendy. Podczas działania programu na ekranie komputera wyświetlane były opisy wykonywanych akcji, widoczne poniżej:

Pausing

Done waiting

Driving 200mm

Waiting for driving to finish

Turning by 90.00 deg relative

Driving 200mm

Turning by 90.00 deg relative

Driving 200mm

Turning by 90.00 deg relative

Driving 200mm

Turning by 90.00 deg relative

Driving 100mm

Turning with radius 100.00mm and 360.0 degrees length

Driving to point X:200.0mm Y:0.0mm

Turning to 45.00 deg absolute

Turning with radius 141.42mm and 360.0 degrees length

Waiting for driving to finish

PRACA INŻYNIERSKA AGH EAIIB 201443

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Driving to point X:-100.0mm Y:-100.0mm

Turning to 45.00 deg absolute

Aby ujawnić niewidoczne na tak małym rysunku (Rys. 5.1) błędy, wykonano program testowy jeszcze raz, z ustawioną skalą rysowania na 2, komendą <m:2>. Ta sama sekwencja poleceń powinna dać w efekcie dwukrotnie powiększony rysunek. Wynik wykonania programu można zaobserwować na Rys. 5.2.

Rys. 5.2: Rysunek powstały po wykonaniu sekwencji poleceń rysowania z ustawionym dwukrotnym powiększeniem rysowania

Na Rys. 5.2 widać wyraźnie niedoskonałości rysunku zaznaczone strzałkami. Koła nie są współśrodkowe z kwadratem. Poza tym, jakość rysunków nie zmieniła się. Poprawnie został narysowany ten sam wzór co na Rys. 5.1, lecz w dwukrotnym powiększeniu. Linia przerywana nie została powiększona, gdyż parametr skali nie wpływa na sposób jej rysowania.

Aby sprawdzić, czy powstałe błędy są spowodowane niedoskonałościami algorytmów generowania ruchu, czy pozycjonowania, naniesiono na jeden wykres punktowy (Rys. 5.3) dane telemetryczne pozycji robota z obu przejazdów. Dane te zostały zebrane z wykorzystaniem zaimplementowanego mechanizmu logowania danych.

Rys. 5.3: Dane telemetryczne uzyskane w trakcie obu przejazdów testowych

Z wykresu wyraźnie widać, że trajektorie robota podczas obu przejazdów testowych nakładają się na siebie z dużą dokładnością. Na Rys. 5.1 oraz Rys. 5.2 można zaobserwować takie same błędy jak na Rys. 5.3. Świadczy to o tym, że niedoskonałości rysunku nie są spowodowane błędami pozycjonowania (odometrii), lecz błędami regulatorów generatorów ruchu. W rzeczywistości taki wynik nie jest niczym dziwnym, jeżeli wziąć pod uwagę ich prostotę. Żaden z zaimplementowanych algorytmów nie zapewnia podążania po zadanej trajektorii.

Zadanie podążania po trajektorii jest zadaniem trudnym i wykracza poza zakres tej pracy. Algorytmy takie są tematem pracy [15], która bazuje na tym samym robocie mobilnym i jego oprogramowaniu.

## 5. 2. Weryfikacja działania regulatora dojeżdżania do punktu

Regulator dojeżdżania do punktu wyróżnia się wśród innych tym, że reguluje prędkość silników w sposób ciągły. Pozostałe regulatory tylko ustawiają prędkość początkową silników i czekają na zakończenie ruchu. W celu sprawdzenia poprawności jego działania stworzono trajektorię testową w kształcie elipsy widoczną na Rys. 5.4. Wygenerowano sekwencję komend dojeżdżania do kolejnych punktów po średnich trajektorii zaznaczonych na wykresie.

Rys. 5.4: Wykres trajektorii w kształcie elipsy

Sekwencję wykonano uzyskując rezultat widoczny na Rys. 5.5.



Rys. 5.5 : Wynik sekwencji rysowania elipsy

Jak widać na Rys. 5.5, elipsa została narysowana poprawnie. Odcinki między punktami pośrednimi zostały przybliżone prostymi, a ich połączenia delikatnie zaokrąglone. Jest to spodziewany rezultat i regulator dojeżdżania do punktu był projektowany właśnie w celu uzyskania takich efektów.

Test dowodzi, że regulator dojeżdżania do punktu nadaje się do rysowania złożonych kształtów przybliżonych odcinkami prostymi.

### 5. 3. Poprawność rysowania wszystkich rodzajów linii

Celem tego testu jest prezentacja i sprawdzenie poprawności rysowania wszystkich zaimplementowanych rodzajów linii. Sprawdzone zostanie też, czy w trakcie rysowania złożonych rysunków, możliwa jest zmiana stylu aktualnie rysowanej linii.

46 AGH EAIIB 2014 PR ACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Stworzono sekwencję poleceń rysującą 5 kół, każde innym typem linii. Spodziewanym rezultatem jest narysowanie dużego koła linią ciągłą, w którym będą znajdowały się 4 mniejsze koła, styczne do dużego i przecinające się w jego środku. Każde z 4 małych kół powinno zostać narysowane innym rodzajem linii spośród linii kropkowanej, dwóch kreskowanych oraz linii kropka-kreska. Wynik działania widoczny jest na Rys. 5.6.

Rys. 5.6: Wynik sekwencji testowej sprawdzającej poprawność rysowania różnych linii

Jak widać z Rys. 5.6, wszystkie linie zostały narysowane poprawnym stylem. Koła przecinają się w jednym miejscu, co jeszcze raz pokazuje dokładność algorytmów rysujących.

#### 5. 4. Pomiary poprawności działania sekcji zasilania

Sprawdzenie poprawności działania sekcji zasilania, a w szczególności przetwornicy impulsowej, jest bardzo ważne z uwagi na złożoność tego układu. Konieczne jest zweryfikowanie, czy silniki i serwo mechanizm nie zakłócają swoją pracą działania wrażliwych elementów cyfrowych.

Pomiary rozpoczęto od zarejestrowania przebiegów napięcia akumulatora oraz napięcia na zaciskach jednego z silników gdy ten jest włączony. Test miał na celu wykazanie w jaki sposób praca silników wpływa na stabilność napięcia akumulatora. Wyniki pomiarów widoczne są na Rys. 5.7.

PRACA INŻYNIERSKA AGH EAIIB 201447

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

przesunięcie -3V

Rys. 5.7: Przebieg napięcia akumulatora podczas pracy silnika

Na Rys. 5.7 wyraźnie widoczne są oscylacje napięcia baterii w chwilach zmian napięcia podawanego na silniki przez mostki H. Oscylacje te mają maksymalną amplitudę równą około 0.6V, co stanowi 7.7% napięcia akumulatora, oraz częstotliwość równą 10kHz – równą częstotliwości sygnału PWM sterującego prędkością silników. Zakłócenia te mogłyby być jeszcze większe, gdyby zostały włączone z dużą prędkością oba silniki.

Jeżeli wahania napięcia akumulatora związane z pracą silników przeniosłyby się do innych domen napięciowych, mogłyby zakłócić pracę wrażliwych układów cyfrowych. Aby sprawdzić stabilność napięcia zasilającego elektronikę wykonano pomiar napięcia 5V regulowanego przez przetwornicę impulsową wraz z napięciem na zaciskach silnika, jak wyżej.

przesunięcie -6V

Rys. 5.8: Przebieg napięcia regulowanego 5V podczas pracy silnika

Na Rys. 5.8 widać, że praca silnika nie powoduje wahań napięcia 5V regulowanego przez przetwornicę impulsową. Przetwornica spełnia swoje zadanie separując silniki od wrażliwej elektroniki. Wartość napięcia 5V mogłaby ulec zakłóceniom pochodzącym z odbiorników mocy zasilanych bezpośrednio z przetwornicy. Jedynym takim elementem w robocie jest serwo-mechanizm. Należy więc sprawdzić, czy jego praca wywołuje istotne wahania napięcia zasilającego 5V.

## K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Przeprowadzono pomiar napięcia regulowanego 5V przy pracy serwomechanizmu. Aby wywołać warunki ciągłego obciążenia, przyłożono dodatkowe, zewnętrzne obciążenie do jego ramienia. Wyniki pomiarów znajdują się na Rys. 5.9 oraz Rys. 5.10.

Rys. 5.9: Przebieg napięcia regulowanego 5V przy pracy serwomechanizmu

Na Rys. 5.9 można zaobserwować małe wahania o częstotliwości 50Hz. Jest to częstotliwość sygnału PWM, jakim serwomechanizm swoim steruje silnikiem. Wahania te na pewno więc są spowodowane jego pracą. Na Rys. 5.10 zarejestrowano te same zakłócenia w powiększeniu.

Rys. 5.10: Przebieg napięcia regulowanego 5V przy pracy serwomechanizmu (powiększenie)

Widoczne na Rys. 5.10 oscylacje mają maksymalną amplitudę 260mV (widoczne piki napięcia). Zarejestrowane oscylacje mają negatywny wpływ na pracę układów bezpośrednio zasilanych z domeny napięciowej 5V. Do tych układów należą: moduł komunikacji bezprzewodowej WiFi, układ kontrolerski poziomów logicznych, mostki H oraz układ odbiorczy podczerwieni. Wszystkie wymienione układy zostały zabezpieczone za pomocą kondensatorów odprężających o wartościach pojemności 10uF i 100nF umieszczonych blisko wejść zasilania. Zmniejszają one wpływ wahań napięcia linii zasilającej. Nie zaobserwowano żadnych negatywnych efektów podczas pracy robota.

W skład sekcji zasilania wchodzi również regulator napięcia 3,3 V, który zasilą najbardziej krytyczny element – mikrokontroler. Jego niezawodna praca jest warunkiem koniecznym poprawnej pracy robota.

Istotne wahania napięcia zasilającego mogłyby doprowadzić do spontanicznego resetu mikrokontrolera i przerwania aktualnie wykonywanego przez robota zadania.

Aby zweryfikować stałość napięcia zasilającego 3,3V, wykonano jego pomiar podczas pracy serwomechanizmu pod znacznym obciążeniem. Wykazano powyżej, że takie warunki powodują powstanie zauważalnych zakłóceń napięcia zasilającego 5V, które jest bezpośrednio

PRACA INŻYNIERSKA AGH EAIIB 201449

K.KAPUSTA - BUDOWA I PROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

nim źródłem dla przetwornicy na napięcia 3,3V. Przebieg napięcia 5V również zmierzono w tym samym czasie. Wynik pomiaru w rzeczywistości jest na Rys. 5.11.

Rys. 5.11: Przebieg napięcia regulowanego 3,3V i 5V podczas pracy serwo mechanizmu (w powiększeniu)

Z Rys. 5.11 widać, że występujące w domenie napięciowej 5V zakłócenia spowodowane pracą serwomechanizmu nie są przenoszone na linię zasilającą 3,3V. Przetwornica napięcia spełnia swoją funkcję i zabezpiecza najbardziej wrażliwe układy scalone przed wahaniami napięcia zapewniając stabilne napięcie 3,3V.

Na podstawie wykonanych testów sekcji zasilania można stwierdzić, że jej wykonanie jest prawidłowe. Praca silników nie zakłóca pracy żadnych układów scalonych, natomiast praca serwomechanizmu powoduje nieistotnie małe zakłócenia. Układ elektryczny z regulatorami napięcia został zaprojektowany i wykonany poprawnie.

## 5. 5. Weryfikacja poprawności sterowania silnikami

Celem tego testu jest sprawdzenie parametrów sygnałów sterujących silnikami prądu stałego.

Zweryfikowano także poprawność synchronizacji sygnałów jak opisano w rozdziale 4. 2.

Silniki prądu stałego włączono do jazdy do przodu z małą prędkością, używając pilota. Zmierzono, za pomocą oscyloskopu cyfrowego, sygnały PWM jakie są podawane na mostki H (z mikrokontrolera poprzez układ podnoszenia napięcia). Spodziewano się otrzymać przebiegi dwóch zsynchronizowanych sygnałów PWM o częstotliwości 10kHz. Sygnały powinny być tak zsynchronizowane, aby minimalizować czas gdy oba mają logiczny poziom wysoki.

Zarejestrowano przebiegi widoczne na Rys. 5.12.

50 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

przesunięcie +6V

Rys. 5.12: Przebiegi sygnału PWM sterującego prędkością silników

Rezultatem testu są przebiegi pasujące do oczekiwanych. Są podobne do tych przedstawionych na Rys. 4.1. Wyraźnie widać synchronizację sygnałów. Częstotliwość sygnału jest odpowiednia i wynosi 10kHz. Wynik testu potwierdza całkowitą poprawność generacji sygnałów PWM sterujących silnikami.

5. 6. Identyfikacja problemu tzw. „drgania styków” mikroprzekaźników

„Drgania styków” to występowanie dodatkowych przełączeń podczas jednokrotnego przełączania elementów mechanicznych, takich jak przycisk i. Zjawisko to wpływa negatywnie na działanie systemów elektronicznych, gdy powoduje powstawanie dodatkowych zbroczy sygnału i wykrycie większej niż żądana liczby przełączeń.

„Drgania styków” były prawdopodobną przyczyną błędnego działania mikroprzełączników na płycie PCB. Ten test powinien zweryfikować tę tezę. Wykonano pomiary napięcia na nóżkach mikrokontrolera połączonych bezpośrednio z mikroprzełącznikiem. Dokonano jednokrotnego przełączenia.

Wynikiem zmiany stanu mikroprzełącznika z włączonego na wyłączony był przebieg widoczny na Rys. 5.13. Poziomą prostą zaznaczono poziom maksymalnego napięcia interpretowanego jako stan niski przez mikrokontroler – 0.8V. Jednokrotne przełączenie ze stanu wysokiego na niski spowodowało powstanie dodatkowych 4 par zbroczy narastających i opadających. Jest to ekwiwalentem pięciokrotnego przełączenia mikroprzełącznika w stan niski. [3]

Rys. 5.13: Przebieg napięcia na mikroprzełączniku podczas jego wyłączenia

Zmiana stanu mikroprzełącznika z wyłączonego na włączony wygenerowała przebieg widoczny na Rys. 5.14. Poziomą prostą zaznaczono poziom minimalnego napięcia odczytywanego jako stan wysoki – 2V. Włączenie mikroprzełącznika spowodowało powstanie jednej dodatkowej pary zboczy będącej ekwiwalentem dodatkowego przełączenia p rzełącznika. [3]

Rys. 5.14: Przebieg napięcia na mikroprzełączniku podczas jego włączenia

Błędne działanie mikroprzełączników zaobserwowane na etapie programowania było istotnie efektem „drganí styków”. Na podstawie analizy przebiegów zarejestrowanych na Rys. 5.13 oraz Rys. 5.14 możliwe było oszacowanie maksymalnego czasu po jakim drgania zanikają. Po około 80 mikrosekundach przy wyłączeniu mikroprzełącznika drgania pozostają już w bezpiecznym zakresie zaznaczonym na wykresie poziomą prostą.

Aby wyeliminować negatywne efekty „drganí styków” należało zaprogramować odpowiedni czas nieczułości mikrokontrolera na pojawiające się po sobie przełączenia. Wartość tego czasu wybrano, zachowując duży margines bezpieczeństwa, na 1ms. Jest to czas odpowiednio długi, aby „drgania styków” zanikły, a jednocześnie krótki na tyle, że nie utrudnia korzystania z mikroprzełączników poprzez ignorowanie rzeczywistych przełączeń.



## Rozdział 6. Podsumowanie

Cel niniejszej pracy, którym jest stworzenie i oprogramowanie mobilnego robota rysującego, udało się zrealizować. Skonstruowano solidną konstrukcję mechaniczną zapewniającą wymaganą dokładność rysowania. Zaprojektowano i wykonano układ elektryczny będący wysoce zintegrowanym mikroprocesorowym kontrolerem robota. Napisano oprogramowanie opierające się na systemie operacyjnym czasu rzeczywistego, które steruje procesami jazdy, rysowania i komunikacji z komputerem.

Wykonane w rozdziale 5 testy potwierdzają, że udało się wykonać konstrukcję zdolną do rysowania złożonych rysunków z dużą dokładnością. W pracy [17] opracowano dodatkowe regulatory sterujące silnikami i ruchem robota, które jeszcze bardziej zwiększają dokładność rysowania. Dowodzi to, że zbudowany mobilny robot rysujący stanowi interesującą platformę badawczą. Funkcja rysowania może być wykorzystana do rzeczywistego znaczenia przejechanej trasy.

Potencjalnymi kierunkami rozwoju projektu są między innymi dokładniejsze algorytmy sterowania i pozycjonowania robota. Prowadzone są prace nad zintegrowaniem zestawu czujników inercyjnych, zawierających akcelerometr, magnetometr i żyroskop, które pomogą zniwelować akumulujący się błąd pozycjonowania za pomocą odometrii. Planowane jest dodanie systemu wizyjnego wspomagającego proces wyznaczania pozycji robota względem nieruchomych elementów w przestrzeni. Dzięki wyjątkowej elastyczności wykonanego układu elektrycznego kontrolera, proces rozbudowy nie będzie wymagał jego wymiany.

Poprzez przystosowanie konstrukcji mechanicznej do nowych zadań lub środowisk pracy będzie możliwe zwiększenie potencjalnych zastosowań robota rysującego. Mógłby on zostać wykorzystany do automatycznego i bezobsługowego rysowania pasów drogowych lub linii na boiskach sportowych. Artyści mogliby wykorzystać przyciągając uwagę zdolności robota do tworzenia ogromnych i złożonych grafik na placach lub rynkach w centrach miast.

We wstępie wspomniano o edukacyjnym języku programowania Logo opierającym się na wirtualnym "zółwiu" znaczącym przejechaną przez siebie trasę. Po zmniejszeniu robota i kosztów jego budowy możliwe byłoby zaimplementowanie interpretera tego języka wprost na robocie i wykorzystanie go jako jedyne w swoim rodzaju narzędzia edukacyjnego pozwalającego wprowadzać dzieci w fascynujący świat informatyki i robotyki.

### Bibliografia

[1]T. Bräunl, „Research Relevance of Mobile Robot Competitions,” IEEE Robotics and Automation Magazine, tom 6, nr 4, pp. 32-37, Dec 1999.

[2]Oxford Dictionaries, „robot,” [Online]. Available: <http://www.oxforddictionaries.com/definition/english/robot>. [Data uzyskania dostępu: 22 Listopad 2013].

[3]D. M. Bradley, „Odometry: Calibration and Error Modeling,” Robotics Institute Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

[4]STMicroelectronics, „STM32F407 - dokumentacja techniczna,” [Online]. Available: [www.st.com](http://www.st.com).

[5]Real Time Engineers Ltd., „Strona główna systemu FreeRTOS,” [Online]. Available: [www.freertos.org](http://www.freertos.org). [Data uzyskania dostępu: 22 Listopad 2013].

[6]Texas Instruments, „LM2596 - dokumentacja techniczna,” [Online]. Available: [www.ti.com](http://www.ti.com).

[7]Sipex, „SPX1117 - dokumentacja techniczna,” [Online]. Available: [www.chipdocs.com](http://www.chipdocs.com).

[8]STMicroelectronics, „VNH3SP30 - dokumentacja techniczna,” [Online]. Available: [www.st.com](http://www.st.com).

[9]STMicroelectronics, „ST2378E - dokumentacja techniczna,” [Online]. Available: [www.st.com](http://www.st.com).

[10]WizNet, „WizFi210 - dokumentacja techniczna,” [Online]. Available: [www.wiznet.co.kr](http://www.wiznet.co.kr).

[11]Vishay, „TSOP4836 - dokumentacja techniczna,” [Online]. Available: [www.vishay.com](http://www.vishay.com).

[12]FTDI Chip, „FT232R - dokumentacja techniczna,” [Online]. Available: [www.ftdichip.com](http://www.ftdichip.com).

[13]STMicroelectronics, „STM32F4DISCOVERY schematics,” [Online]. Available: [www.st.com/stm32f4-discovery](http://www.st.com/stm32f4-discovery). [Data uzyskania dostępu: 22 Listopad 2013].

[14]W. Grega, „Metody i algorytmy sterowania cyfrowego w układach scentralizowanych i rozproszonych,” Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH w Krakowie, Kraków, 2004, pp. 170-180.

[15]S. Blažič, „A novel trajectory-tracking control law for wheeled mobile robots,” Robotics

54 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

and Autonomous Systems, tom 59, nr 11, pp. 1001-1007, listopad 2011.

[16]D. Marchewka, M. Piątek, „Wheeled mobile robot modeling aspects,” AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, Computer Science and Electronics, Department of Automatics, Kraków.

[17]M. Furdyna, Prototyp robota rysującego - algorytmy sterujące, Kraków: AGH, EAIIB, praca inżynierska, 2013.

#### Wykaz źródeł ilustracji

Wyszczególniono jedynie ilustracje niebędące utworami własnymi autora pracy.

► Rys. 0.1: Przykładowy wynik programu napisanego w języku Logo - <http://www.paulinegrant.com/wp-content/uploads/2013/07/square.png>

► Rys. 1.1: Wojskowy robot HD2-S Tactical o napędzie gąsienicowym - <http://www.superdroidrobots.com/shop/item.aspx/hd2-s-tactical-swat-eod-robot-with-multi-axis-arm/1090/>

► Rys. 1.2: Arduino Robot - dwukołowy robot z dodatkowymi punktami podparcia z podłożem - <http://www.robotmesh.com/arduino/arduino-robot>

► Rys. 1.3: BST AGV Tugger - czterokołowy robot transportowy - [http://www.dcvelocity.com/products/Mobile\\_Transportation\\_Equipment/titespace\\_bst\\_agv\\_tugger/](http://www.dcvelocity.com/products/Mobile_Transportation_Equipment/titespace_bst_agv_tugger/)

► Rys. 1.4: Przykładowy wydruk z drukarki atramentowej w powiększeniu - <http://cool.conservation-us.org/coolaic/sg/bpg/annual/v15/bp15-08.html>

► Rys. 1.5: Przykładowy wydruk z plotera w powiększeniu - <http://adrienlucca.wordpress.com/>

► Rys. 1.6: Ploter bębnowy Canon iPF820 -

[http://www.canon.pl/For\\_Work/Products/Professional\\_Print/Large\\_Format/iPF820/](http://www.canon.pl/For_Work/Products/Professional_Print/Large_Format/iPF820/)

► Rys. 1.7: Ploter płaski ProDigi NEW -

<http://www.friedheim.co.uk/Systemolutionsforpackaginganddiemaking/tabid/180/Default.aspx>

► Rys. 1.8: Robot Delta firmy Kawasaki - <http://www.kawasakirobot.co.uk/products>

► Rys. 1.9: Robot SCARA firmy Mitsubishi - [http://www.paa-](http://www.paa-automation.com/labauto/products/hardware/robots/p-robot-model.asp?r=Mitsubishi&m=RH-12SDH)

[automation.com/labauto/products/hardware/robots/p-robot-model.asp?r=Mitsubishi&m=RH-12SDH](http://www.paa-automation.com/labauto/products/hardware/robots/p-robot-model.asp?r=Mitsubishi&m=RH-12SDH)

► Rys. 2.2: Silnik DC Pololu serii 25Dx54L - <http://www.pololu.com/product/2286>

► Rys. 2.3: Koła ARW-01 ALUM SES ROBOT WHEEL - <http://www.robotshop.com/ca/en/lynxmotion-aluminum-ses-robot-wheel-pair.html>

► Rys. 3.1: Komputer Raspberry Pi - <http://pl.wikipedia.org/wiki/Plik:RaspberryPi.jpg>

► Rys. 3.2: Podstawowa wersja Arduino Uno - <http://arduino.cc/en/Main/ArduinoBoardUno>

PRACA INŻYNIERSKA AGH EAIIB 201455

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Dodatek A: Schematy elektryczne





PRACA INŻYNIERSKA AGH EAIIB 201457

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO





58 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

PRACA INŻYNIERSKA AGH EAIIB 201459

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO



60 AGH EAIIB 2014 PRACA INŻYNIERSKA

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO

Dodatek B: Schemat elektryczny i projekt PCB modułu bezprzewodowego

Projekt płytki PCB modułu bezprzewodowego.

Zdjęcia strony górnej i dolnej wykonanego modułu komunikacji bezprzewodowej.

PRACA INŻYNIERSKA AGH EAIIB 201461

K.KAPUSTA - BUDOWA I OPROGRAMOWANIE DWUKOŁOWEGO MOBILNEGO ROBOTA RYSUJĄCEGO



Schemat elektryczny modułu komunikacji bezprzewodowej.