

Politechnika Warszawska

W Y D Z I A Ł M E C H A N I C Z N Y
E N E R G E T Y K I I L O T N I C T W A



Zakład Teorii Maszyn i Robotów

Praca dyplomowa inżynierska

na kierunku Robotyka i Automatyka
w specjalności Robotyka

Robot klasy minisumo z systemem
podnoszenia przeciwnika

Jakub Kuziemski

Numer albumu 313569

promotor
dr inż. Andrzej Chmielniak

WARSZAWA 2025

Robot klasy minisumo z systemem podnoszenia przeciwnika

Streszczenie.

Praca przedstawia proces projektowania, budowy oraz testowania robota klasy minisumo wyposażonego w system podnoszenia przeciwnika. Głównym celem projektu było wykonanie konstrukcji spełniającej wymagania turniejowe, która poza standardowymi funkcjonalnościami, takimi jak wykrywanie przeciwnika, wykrywanie krawędzi ringu i autonomiczne poruszanie się po ringu, posiada mechanizm umożliwiający uniesienie przeciwnika w celu ułatwienia wypchnięcia go z pola walki. Robot został wyposażony w rozkładające się na początku walki ramiona, które wykorzystując silniki oraz zamontowane na nich walce realizują ten cel.

W pracy omówiono szczegółowo proces projektowania mechanicznego, w tym wykorzystanie technologii druku 3D oraz projektowania obwodów PCB. Dobór komponentów takich jak mikrokontroler, czujniki i układy zasilania został przedstawiony z uwzględnieniem ich funkcji w konstrukcji robota. Zaprezentowano również algorytmy sterowania, które umożliwiają efektywne zachowanie na ringu, uwzględniając różne scenariusze rywalizacji.

Przeprowadzone testy wykazały, że zaproponowany system podnoszenia przeciwnika istotnie zwiększa skuteczność robota w starciach z tradycyjnymi konstrukcjami. W podsumowaniu wskazano potencjalne kierunki dalszej optymalizacji, takie jak ulepszenie algorytmów sterowania oraz zastosowanie bardziej zaawansowanych materiałów konstrukcyjnych.

Słowa kluczowe: robotyka, robot mobilny, minisumo

Minisumo robot with an opponent-lifting system

Abstract. The work presents the process of designing, building, and testing a minisumo robot equipped with an opponent-lifting system. The main goal of the project was to create a design that meets tournament requirements, which, in addition to standard functionalities such as detecting opponents, detecting the ring edges, and autonomous movement on the ring, includes a mechanism that allows lifting the opponent to facilitate pushing them out of the competition area. The robot is equipped with arms that unfold at the beginning of the fight, which, using motors and rollers mounted on them, achieve this goal.

The work discusses in detail the mechanical design process, including the use of 3D printing technology and PCB circuit design. The selection of components such as the microcontroller, sensors, and power supply circuits is presented, taking into account their function in the robot's design. It also presents control algorithms that enable efficient behavior on the ring, considering various competition scenarios.

The conducted tests showed that the proposed opponent-lifting system significantly increases the robot's effectiveness in battles with traditional designs. The summary highlights potential directions for further optimization, such as improving control algorithms and using more advanced construction materials.

Keywords: robotics, mobile robot, minisumo



.....
miejscowość i data

.....
imię i nazwisko studenta

.....
numer albumu

.....
kierunek studiów

OŚWIADCZENIE

Świadomy/-a odpowiedzialności karnej za składanie fałszywych zeznań oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie, pod opieką kierującego pracą dyplomową.

Jednocześnie oświadczam, że:

- niniejsza praca dyplomowa nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.) oraz dóbr osobistych chronionych prawem cywilnym,
- niniejsza praca dyplomowa nie zawiera danych i informacji, które uzyskałem/-am w sposób niedozwolony,
- niniejsza praca dyplomowa nie była wcześniej podstawą żadnej innej urzędowej procedury związanego z nadawaniem dyplomów lub tytułów zawodowych,
- wszystkie informacje umieszczone w niniejszej pracy, uzyskane ze źródeł pisanych i elektronicznych, zostały udokumentowane w wykazie literatury odpowiednimi odnośnikami,
- znam regulacje prawne Politechniki Warszawskiej w sprawie zarządzania prawami autorskimi i prawami pokrewnymi, prawami własności przemysłowej oraz zasadami komercjalizacji.

Oświadczam, że treść pracy dyplomowej w wersji drukowanej, treść pracy dyplomowej zawartej na nośniku elektronicznym (płycie kompaktowej) oraz treść pracy dyplomowej w module APD systemu USOS są identyczne.

.....
czytelny podpis studenta

Spis treści

1. Wstęp	9
1.1. Wprowadzenie	9
1.2. Cel pracy	10
1.3. Przegląd wybranych robotów minisumo	12
2. Projektowanie	16
2.1. Wstępna koncepcja	16
2.2. Konstrukcja mechaniczna	18
2.2.1. Pług	18
2.2.2. Ramiona robota	19
2.2.3. Mocowanie silników i baterii	20
2.2.4. Koła	20
2.2.5. Złożenie	21
2.3. Wybór komponentów	22
2.3.1. Silniki dla kół	22
2.3.2. Silniki dla ramion	23
2.3.3. Wykrywanie przeciwnika – czujnik odległości	23
2.3.4. Wykrywanie linii – czujnik zmian jasności podłoża	24
2.3.5. Główny układ sterujący	25
2.3.6. Akumulator	26
2.4. Projekt PCB	26
2.4.1. Układ zasilania	26
2.4.2. Układ sterowania	27
2.4.3. Układ sterowania silnikami	28
2.4.4. Układ wykrywania krawędzi	29
2.4.5. Wyjścia do układów zewnętrznych	29
2.4.6. Grubość ścieżek	30
3. Wykonanie robota	31
3.1. Druk 3D	31
3.2. Wykonanie PCB	31
3.3. Montaż konstrukcji	32
3.4. Dodatkowe obciążenie	35
4. Oprogramowanie	37
4.1. Architektura systemu	37
4.2. Algorytmy sterowania	38
4.3. Wstępna kalibracja czujników	40
4.4. Test nośności	41
5. Podsumowanie	43

5.1. Samokrytyka	43
5.2. Potencjalne usprawnienia lub rozbudowy	43
5.3. Wnioski	44
Bibliografia	45
Spis rysunków	48
Spis tabel	48
Spis załączników	48

1. Wstęp

1.1. Wprowadzenie

Roboty mobilne odgrywają istotną rolę we współczesnej robotyce, integrując różnorodne dziedziny wiedzy, takie jak mechanika, elektronika oraz programowanie. Zawody walk robotów, na różnych płaszczyznach, są popularną formą zarówno edukacji w dziedzinie robotyki mobilnej, jak i rozrywki samych konstruktorów. Zawody sumo jako jedne z nich, zyskały sobie znaczną popularność. Konkurencja ta polega na autonomicznej rywalizacji robotów, nawiązując do tradycyjnych japońskich zapasów sumo. Celem rywalizacji jest wypchnięcie przeciwnika poza okrągłą arenę zwaną dohyo, przy zachowaniu pełnej autonomii działania. Walki sumo, dzięki swojej widowiskowości, zdobyły dużą popularność. Zostały one podzielone na różne kategorie, w których ograniczają je wymiary i masa [1].

Tabela 1.1. Specyfikacja wymiarów i wag dla różnych klas robotów sumo ze strony robogames.net

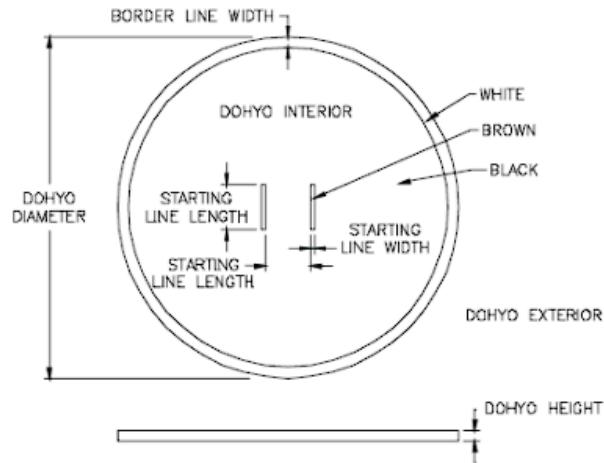
Klasa	Wysokość	Szerokość	Długość	Waga
Mega Sumo - Auton	brak ograniczeń	20.0 cm	20.0 cm	3,000 g
Mega Sumo - R/C	brak ograniczeń	20.0 cm	20.0 cm	3,000 g
Mega Sumo - Network	brak ograniczeń	20.0 cm	20.0 cm	3,000 g
Humanoid - R/C	50 cm	20.0 cm	20.0 cm	4,000 g
Kit Sumo	brak ograniczeń	15.0 cm	15.0 cm	1,000 g
Mini Sumo	brak ograniczeń	10.0 cm	10.0 cm	500 g
Micro Sumo	5 cm	5.0 cm	5.0 cm	100 g
Nano Sumo	2.5 cm	2.5 cm	2.5 cm	25 g

Jak wynika z tabeli 1.1 roboty klasy minisumo nie mogą przekroczyć w chwili startu, rozmiarów 100x100 mm, ani masy 500 g. Nie mają za to określonej maksymalnej wysokości, co zostanie wykorzystane w niniejszym projekcie. Konstrukcje są wyposażone w różnorodne czujniki, takie jak czujniki odległości, czujniki wykrywające linie na podłożu czy niekiedy nawet akcelerometry. Przy pomocy danych z owych czujników, za pomocą napisanego wcześniej programu robot musi namierzyć przeciwnika, a następnie, nie opuszczając samemu ringu, wypchnąć go. Wartymi nadmienienia ograniczeniami są także zakazy: intencjonalnego zakłócania odczytów przeciwnika, uszkadzania dohyo, strzelania, rzucania, intencjonalnego uszkadzania przeciwnika, a także przytwierdzania się do podłoża.

Rozmiar dohyo, na którym toczone są pojedynek, również jest uzależniony od kategorii [2].

Dość powszechnie [3][4] stosuje się także elektroniczne moduły startowe, których specyfikacja jest dostępna w regulaminach zawodów. Ich zadaniem jest zdalne uruchamianie dwóch robotów jednocześnie przez sędziego, co w efekcie jest nie tylko wygodne, ale również zwiększa uczciwość rywalizacji.

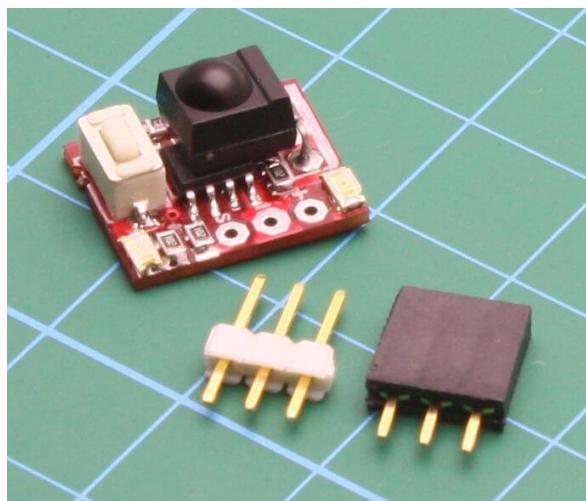
1. Wstęp



Rysunek 1.1. Dohyo z oznaczonymi wielkościami charakterystycznymi

Tabela 1.2. Wymiary dohyo dla różnych kategorii robotów na zawodach "Robothon" w USA [2]

Kategoria	Średnica	Wysokość	Szerokość obrzeża	Długość linii startowej	Szerokość linii startowej
Mini	77 cm	2.5 cm	2.5 cm	10 cm	1 cm
Micro	38.5 cm	1.25 cm	1.25 cm	5 cm	0.5 cm
Nano	19.25 cm	0.6 cm	0.63 cm	2.5 cm	0.25 cm



Rysunek 1.2. Komercyjnie sprzedawany moduł startowy firmy JSumo

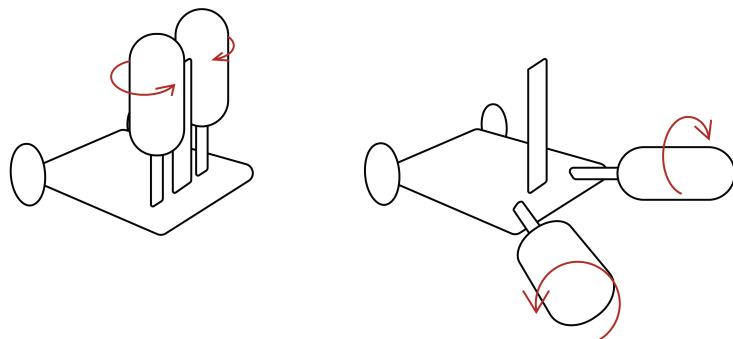
1.2. Cel pracy

Celem niniejszej pracy było wykonanie robota klasy minisumo, zgodnego ze wszystkimi wymaganiami turniejowymi, który poza standardowymi funkcjonalnościami robotów minisumo, czyli autonomiczną jazdą z czujnikami przeciwnika i linii, jest wyposażony w ramiona do podnoszenia przeciwnika.

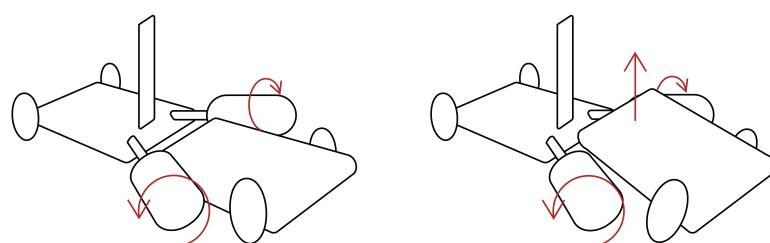
Jak większość robotów minisumo niniejszy robot jest wyposażony w transoptory słu-

żące do wykrywania linii krawędzi ringu. Trzy czujniki zbliżeniowe odpowiadają za namierzanie przeciwnika. Konstrukcja robota zakłada dwukołowy układ z frontem ślizgającym się po powierzchni maty, a sterowanie kierunkiem jazdy realizowane jest poprzez niezależne zasilanie obu silników, co tworzy typowy napęd różnicowy. Sterowanie całością odbywa się przy użyciu mikrokontrolera Atmega328P, a zasilanie zapewnia dwuogniowy akumulator litowo-polimerowy. Wszystkie komponenty są połączone za pomocą specjalnie zaprojektowanej płytki PCB oraz elementów wydrukowanych na drukarce 3D.

Charakterystycznym elementem robota są dwa ramiona zamontowane w przedniej części po bokach urządzenia. Szkic rozwiązania przedstawiono na rys. 1.3. Na początku walki ramiona są ustalone pionowo, aby nie przekraczać dopuszczalnych wymiarów robota. W pierwszych sekundach starcia robot rozkłada ramiona, uruchamiając silniki na ich końcach. Ramiona, kontynuując ruch, unoszą przeciwnika znajdującego się pomiędzy nimi. Zmniejszenie nacisku przeciwnika na podłożu ułatwia wypchnięcie go z ringu, co zwiększa szansę na zwycięstwo.



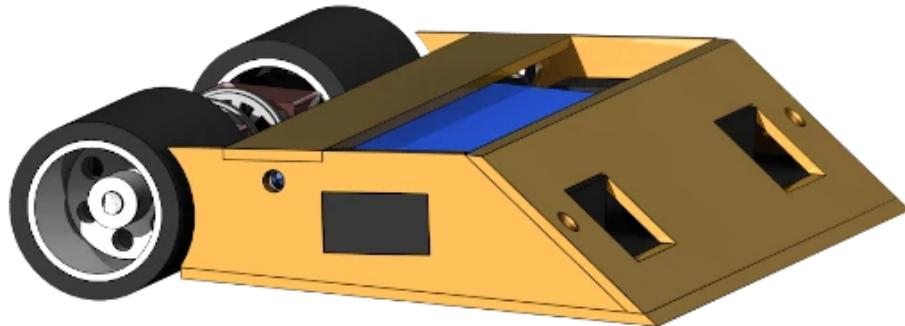
Rysunek 1.3. Szkic przedstawiający koncepcję rozkładania się robota



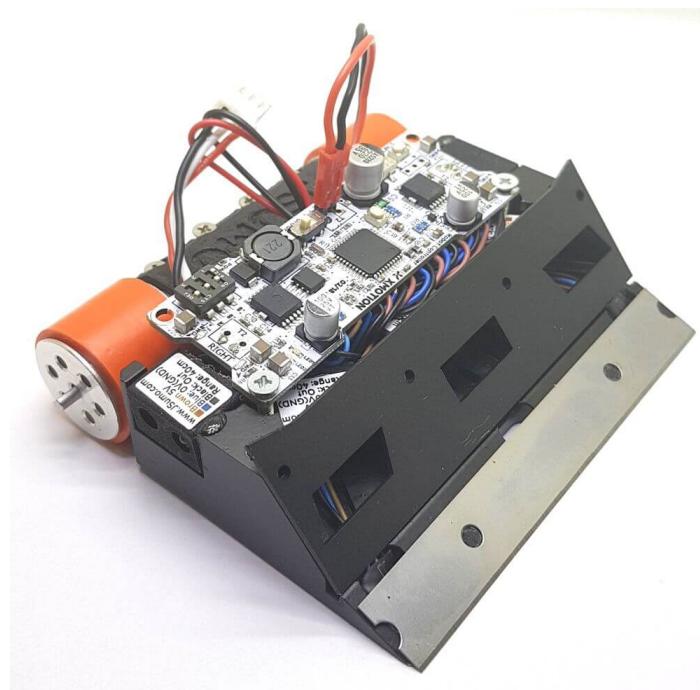
Rysunek 1.4. Szkic przedstawiający koncepcję podnoszenia przeciwnika ramionami

1.3. Przegląd wybranych robotów minisumo

Sformułowane wyżej wymagania, skumulowane z dość długą historią zawodów, stworzyły warunki do powstania preferowanych strategii i rozwiązań konstrukcyjnych. Roboty najczęściej mają masę dokładnie 500 g, nisko osadzony środek ciężkości, kilka czujników zbliżeniowych do detekcji przeciwnika, dwa silniki z tyłu oraz niekiedy transoptory do wykrycia linii na podłoże (nie jest to nawet konieczne przy niektórych strategiach).



Rysunek 1.5. Projekt robota, opisanego na forum internetowym [5]



Rysunek 1.6. Komercyjnie sprzedawany robot firmy JSumo [6]

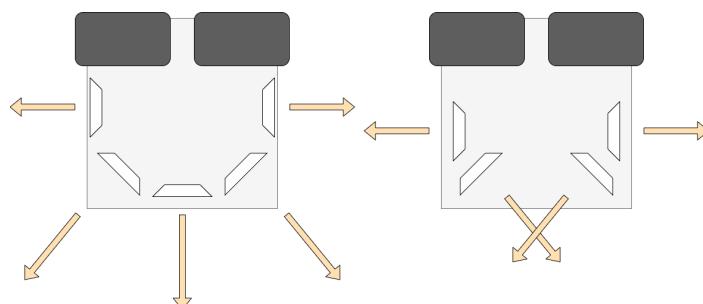
Jak można zauważyć na rys. 1.5 i rys. 1.6 roboty obu niezwiązanych ze sobą podmiotów mają wiele podobieństw. Zgadza się to również z prywatnymi obserwacjami autora, jako obserwatora lub uczestnika walk minisumo.

Pierwszym podobieństwem, które nie wynika bezpośrednio z wymagań turniejowych, jest wysokość robotów. Dla robota minisumo obrócenie go kołami do góry lub przewróce-

nie na któryś z boków jest w znacznej większości przypadków równoznaczne z porażką. Dominującym trendem są zatem roboty o jak najniżej położonym środku ciężkości oraz masie niewiele mniejszej od regulaminowego limitu. Ta druga cecha związana jest także ze zwiększoną siłą tarcia. W niekomercyjnym rozwiązaniu rolę dodatkowego obciążenia spełniały ołowiane ciężarki umieszczone wewnątrz konstrukcji [5]. W komercyjnym rozwiązaniu masa została dokładnie zaplanowana już na etapie produkcji i jak czytamy gotowy złożony zestaw waży 490 g.

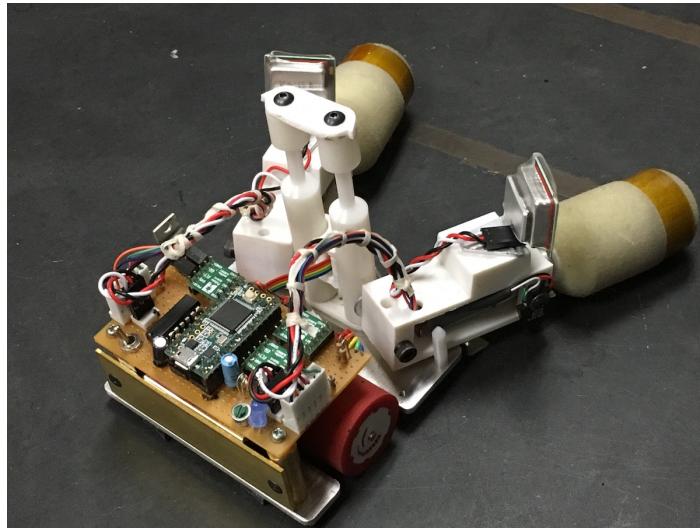
Uwagę może również zwrócić jednakowa konfiguracja kół. Intuicja mogłaby wskazywać, że konstrukcje czterokołowe z większą mocą powinny przeważyć. Obserwacje zawodów nasuwają jednak nieco inne wnioski. Czterokołowe konstrukcje często w wyniku zderzeń lub uniesienia przez przeciwnika i tak wykorzystują moc jedynie dwóch silników. Przez to, że roboty takiej konstrukcji nie opierają się frontem o matę, do takiego podniesienia dochodzi na tyle często, że konstrukcje czterokołowe są raczej wypierane. Ważnym szczegółem w dwukołowych rozwiązaniach jest umiejscowienie kół tak, aby nawet po lekkim uniesieniu przez przeciwnika koła nie traciły przyczepności.

Robot widoczny na rys. 1.5 jest wyposażony w dwa optyczne czujniki odległości SHARP GP2Y0D340K do wykrywania przeciwnika, podczas gdy jego komercyjny rywal w aż 5, również optycznych czujników odległości JS40F. Własne obserwacje wskazują, że liczba czujników zbliżeniowych robota waha się właśnie w tym przedziale. Ustawienie czujników również różni się w zależności od robota. Obserwacje zawodów pozwalają podzielić ustawienia na dwie grupy widoczne na rys. 1.7. Lewa charakteryzuje się nieco większym polem widzenia robota, podczas gdy prawa daje nieco dokładniejszą informację na temat odległości od przeciwnika. Każda z opcji ma więc swoje zalety, które konstruktory wykorzystują na potrzeby różnych strategii walki.



Rysunek 1.7. Dwie koncepcje ustawiania czujników zbliżeniowych

Na rys. 1.8 widzimy robota minisumo z 2016 roku autorstwa Dale'a A. Heatherington'a. Jest to jedyny znaleziony w literaturze (głównie strony internetowe) robot realizujący koncepcję zaproponowaną w niniejszej pracy. Jego ramiona rozkładają się wraz ze startem rundy, aby następnie unosiły przeciwnika, albo chociaż dostatecznie zmniejszały jego przyczepność. Jak czytamy w opracowaniu [7], robot wygrywa pojedynek z "klasycznym" robotem tego samego konstruktora. Na nagraniach, udostępnionych przez autora, z zawa-



Rysunek 1.8. Robot "Huggy bear", jedyny znaleziony robot realizujący koncepcję walki z ramionami do podnoszenia przeciwnika [7]

dów AHRC Robot Rally w 2016 widzimy, że robot wygrywa swoje starcia. Robot "Huggy Bear" jest zatem nie tylko dowodem na to, że koncepcja takiego robota może zadziałać, ale także dowodem, że może wygrywać zawody. Warto jednak wspomnieć, że na opisanych zawodach brały udział jedynie 4 roboty [8].

Opisując robota "Huggy Bear" warto jednak zwrócić również uwagę na jego wady. Robot ma kilka elementów, które mogłyby być nieco poprawione. Już przy niewielkim podniesieniu frontu robota przez przeciwnika "Huggy Bear" może tracić przyczepność, przez tylną część konstrukcji. Takie rozwiązania stosuje się w robotach dla zwiększenia ochrony robota przed atakiem z tyłu, jednak na nagraniach z zawodów nie widzimy takich sytuacji. Dla robota z ramionami kluczowe jest, aby doprowadzić do zderzenia frontalnego, bo to w nim ma on mieć przewagę. Wydaje się, że konstruktor "Huggy Beara" oddaje nieco swojej skuteczności w takim starciu dla pewnych defensywnych zysków. W niniejszej pracy prezentowane będzie jednak podejście odwrotne, aby to frontalne starcie było jak najbardziej skuteczne.

Kolejną istotną przewagą robota opisanego w niniejszej pracy nad wspominanym robotem może być przyczepność ramion. Wykonanie ich z wałków malarских było zapewne podyktowane ich sprężystością, która miała pomagać w podnoszeniu przeciwnika. Wydaje się, że to dobra koncepcja, jednak można osiągnąć podobny efekt stosując np. niewielkie luzy w ramionach. Można wówczas pokryć kręczącą się część ramion materiałem o większej przyczepności, aby uzyskać jeszcze skuteczniejszy efekt.

Strategie walki w robotach minisumo nie muszą być wcale wyszukane, aby skutkowały osiągnięciem zadowalających wyników. Często w programie sterującym wystarczy tzw. "kilka ifów", jeżeli ktoś przyłożył większe starania do np. doboru silnika lub opon. Poza robotami o bardziej złożonych algorytmach, możemy podzielić proces sterowania działaniem robota na kilka stanów [9]. Pierwszym z nich jest zwykle szukanie przeciwnika,



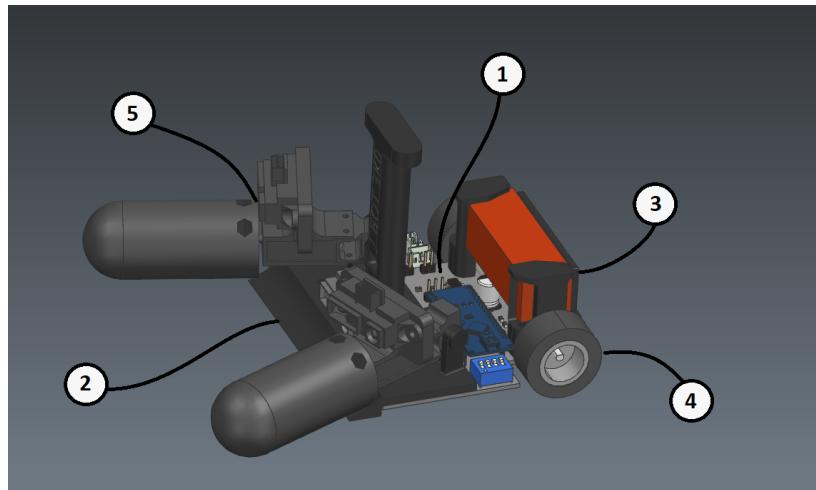
Rysunek 1.9. Maszyna stanów zaproponowana przez jednego z konstruktörów minisumo [9]

które może odbywać się poprzez powolne przesuwanie się robota do przodu, starając się jednocześnie nieco zmieniać orientację w celu wyeliminowania tzw. martwych stref. W przypadku robotów niewyposażonych w czujniki podłożu, a niekiedy i takie osiągają wysokie noty w zawodach, szukanie przeciwnika odbywa się poprzez wyłącznie zmianę orientacji, dzięki czemu eliminuje się ryzyko wypadnięcia poza ring. Najważniejsze jest, aby zgodnie z regulaminem większości zawodów [3] pozostawać w ruchu, gdyż to takie roboty są premiowane w przypadkach remisowych.

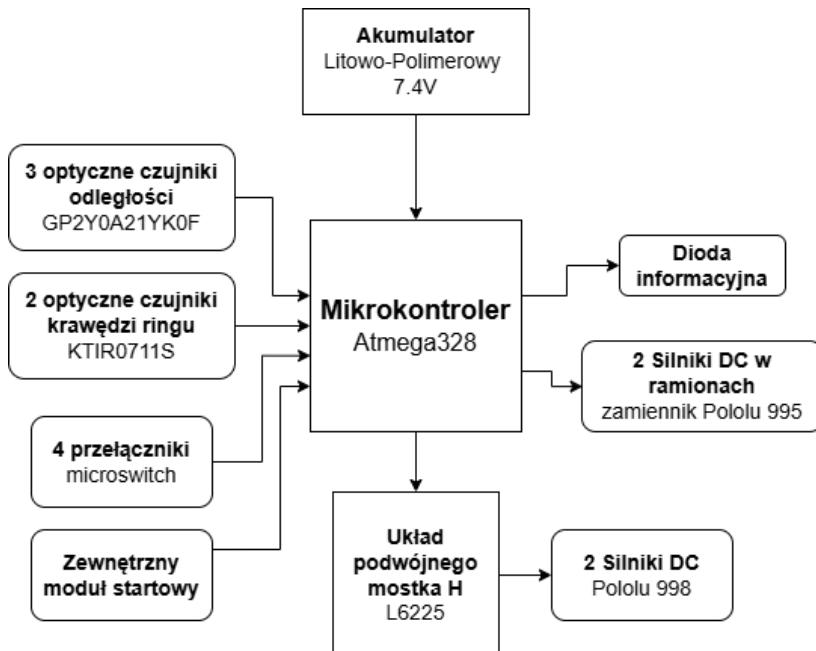
Poza szukaniem przeciwnika robot ma także stan przetrwania, który można bardziej celnie nazwać stanem uciekania z linii. Na rys. 1.9 widzimy też stan ataku, który tutaj może przechodzić w stan przetrwania w przypadku wykrycia linii. Rozwiążanie takie jest na pewno bardziej efektywne, gdyż robot po wypchnięciu przeciwnika sam nie wyjedzie poza ring, jednak regulaminy zawodów zwykle nie są zainteresowane stanem po opuszczeniu ringu przez pierwszego robota. Zatem niekiedy tworzone są algorytmy sterowania z inną kolejnością instrukcji warunkowych. Robot jedzie wtedy z przeciwnikiem poza ring, eliminując przy tym ewentualne krańcowe przypadki, w których nie udałoby się wypchnąć przeciwnika do końca z powodu wcześniejszego wykrycia linii.

2. Projektowanie

2.1. Wstępna koncepcja



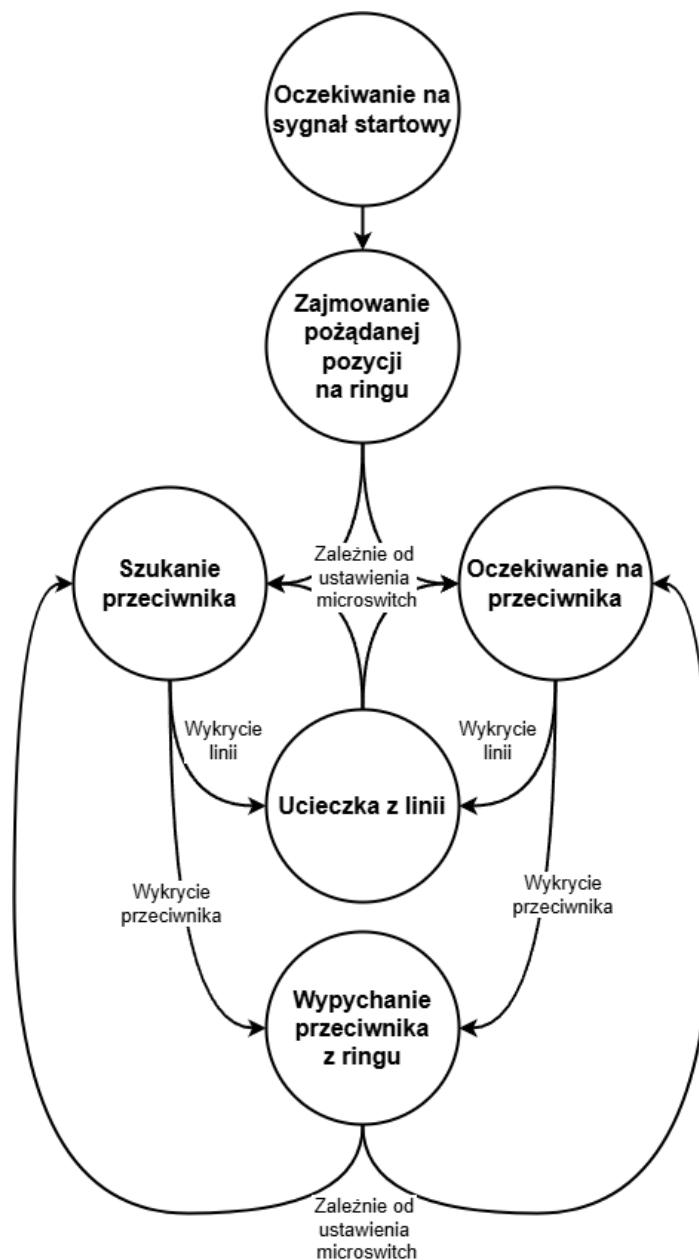
Rysunek 2.1. Projekt robota minisumo wyposażonego w ramiona z oznaczonymi elementami konstrukcji (opis w tekście)



Rysunek 2.2. Schemat blokowy układu sterowania robota minisumo wyposażonego w ramiona

Podstawą konstrukcji robota jest płytka PCB zaznaczona na rys. 2.1 numerem 1. Na niej znajduje się układ elektroniczny odpowiedzialny za sterowanie robotem oraz złącza do podłączenia pozostałych elementów. Numerem 2 oznaczono plug, który poza swoim standardowym zastosowaniem, czyli próbą podniesienia przeciwnika, jest wyposażony w miejsce na czujnik odległości i dwa ramiona oznaczone numerem 5. W tylnej części

konstrukcji numerem 3 zaznaczono obudowę przytrzymującą silniki oraz baterie. Robot jest dwukołowcem o napędzie różnicowym z kołami oznaczonymi numerem 4. Każdy z elementów jest dokładniej opisany w podrozdziale 2.2.

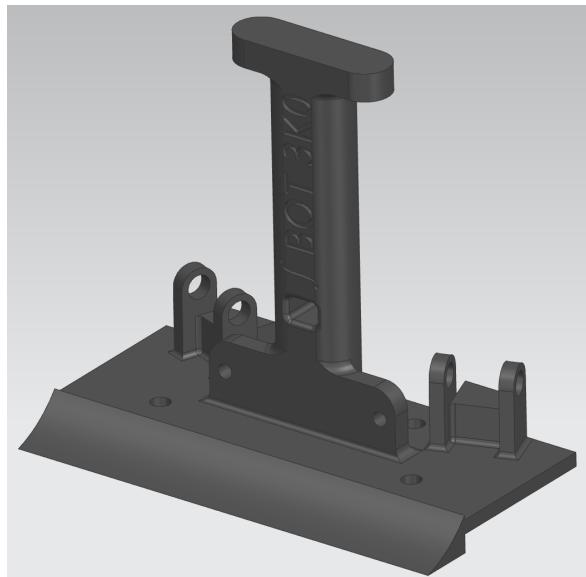


Rysunek 2.3. Graf stanów opisywanego robota minisumo

Robot jest wyposażony w układ sterowania, którego schemat blokowy jest widoczny na rys. 2.2 i realizuje program zgodny z grafem stanów widocznym na rys. 2.3. Jak zaznaczono na grafie, robot, w zależności od przełączników microswitch, może szukać lub jedynie czekać na rywala. Pozostałe przełączniki są wykorzystywane do zajmowania dogodnej pozycji startowej, a dokładniejszy opis algorytmu sterowania znajduje się w rozdziale 4.2.

2.2. Konstrukcja mechaniczna

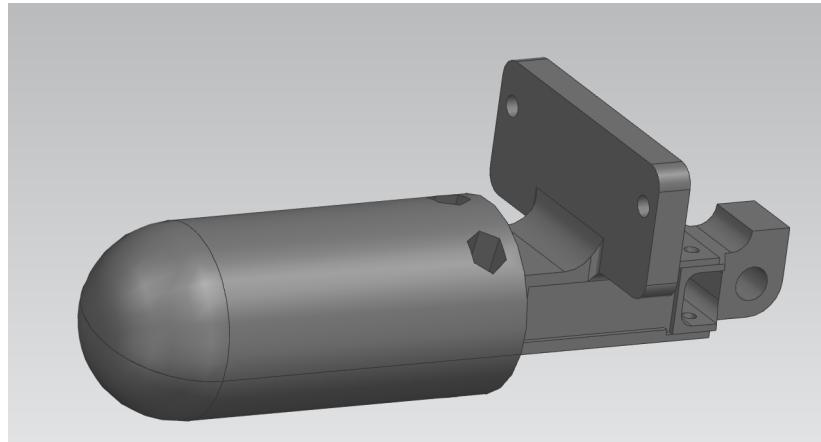
2.2.1. Plug



Rysunek 2.4. Plug robota

Ze względu na małą gęstość i stosunkowo dużą wytrzymałość zdecydowano się na konstrukcję wykonaną techniką druku 3D. Pracę rozpoczęto od budowy płyta. Jest to jeden z kluczowych elementów każdego robota minisumo. Podniesienie przeciwnika jest często czynnikiem warunkującym zwycięstwo, dlatego konstruktory starają się, aby płyta była możliwie jak najniżej. Przy konstrukcji dwukołowej jest on najczęściej po prostu oparty o podłoże i ślizgając się po macie stanowi trzeci punkt podparcia robota. Także w tym przypadku zdecydowano się na taką konstrukcję.

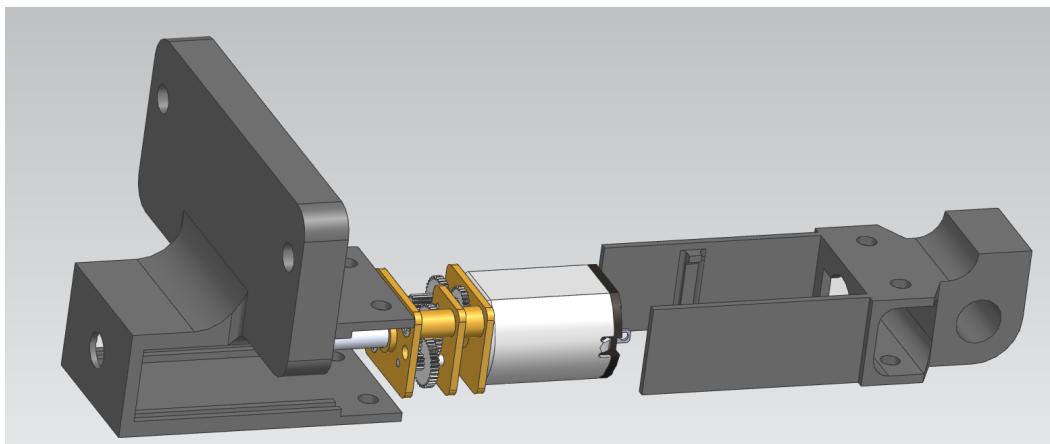
Frontalna część w opisywanym robocie ma jednak inne ważne zadania, które zostały uwzględnione w jego budowie. Cztery otwory ϕ 4 służą do połączenia robota z płytą sterującą, która jednocześnie będzie częścią konstrukcji. Opcjonalnie w razie potrzeby można wykorzystać otwory do montażu obciążników ołowianych w celu dociążenia konstrukcji. W centralnej części płyty dostępne jest miejsce do zamontowania czujnika Sharp GP2Y0A21YK0F (uzasadnienie wyboru w dalszej części pracy (2.3.3)). Tuż nad nim znajduje się ślupek, który jest częścią prostego mechanizmu rozkładania się ramion robota. W środku słupka wykonano otwór na przewody w celu lepszej ich organizacji. Symetrycznie po obu stronach płyty będą, przy pomocy sworzni, montowane ramiona robota. Jednocześnie konstrukcja ta wraz z konstrukcją ramienia tworzą ogranicznik, aby ramię nie mogło rozłożyć się do tyłu.



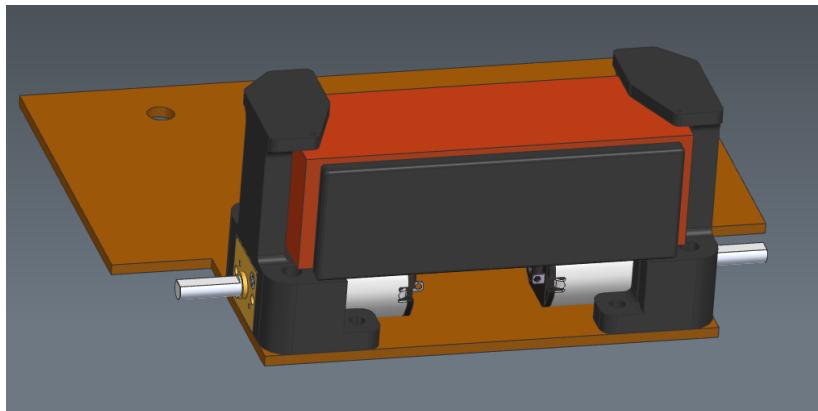
Rysunek 2.5. Ramię robota

2.2.2. Ramiona robota

Charakterystycznym elementem opisywanej konstrukcji są ramiona. Obudowa silnika robota składa się z dwóch skręcanych ze sobą elementów. Pomiędzy nimi znajduje się silnik, zamiennik Pololu 998 (o którym więcej w 2.3.2). Współpracę tą widać na rys. 2.6. Ramię jest tak skonstruowane, że silnik pomiędzy dwoma jego częściami jest nie tylko unieruchomiony, ale także odizolowany od kurzu czy włosów, co jest istotne dla niezawodności działania przekładni. W górnej części ramienia skonstruowano mocowanie dla kolejnego czujnika GP2Y0A21YK0F. Poza otworami na śruby łączące obie części obudowy silnika przewidziano też otwór na wał silnika. Tylna część konstrukcji ramienia ma też otwór na przewody do silnika, a także otwór do współpracy z plugiem, o czym mowa w rozdziale 2.2.1. Na wale silnika skręcana jest także część ramienia przeznaczona do podnoszenia przeciwnika, która jest pokryta gumą.



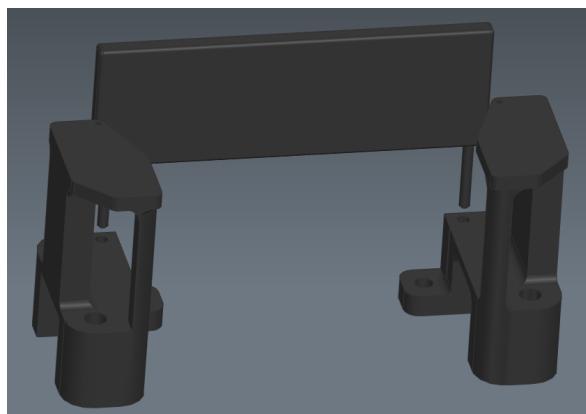
Rysunek 2.6. Rozłożone ramię



Rysunek 2.7. Mocowanie baterii i silników wraz z poglądowymi modelami baterii (czerwony) i płytka (pomarańczowy)

2.2.3. Mocowanie silników i baterii

Tylna część konstrukcji robota jest odpowiedzialna za trzymanie baterii, oraz mocowanie silników do płytka PCB. Można podzielić ją na 3 części. Dwa odbite symetrycznie elementy przytwierdzone do płytka przy pomocy 3 śrub M3 utrzymują silniki w bezruchu, ale także (ponownie) izolują przekładnie od ewentualnych zabrudzeń. Pozostała część tych elementów jest zaprojektowana tak, by odebrać możliwość ruchu baterii we wszystkich kierunkach poza tyłem robota. Tam zaprojektowano trzeci element, który można wyjąć na czas zmiany akumulatora, lub włożyć tak, aby zupełnie uniemożliwić jego ruch.



Rysunek 2.8. Rozłożone mocowanie baterii i silników

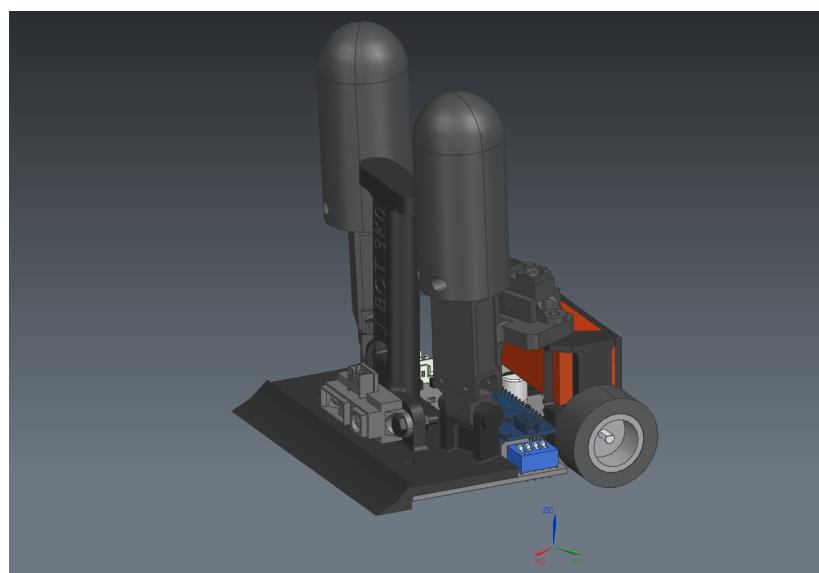
2.2.4. Koła

Zdecydowano się wykorzystać sprawdzone, fabryczne koła Solarbotics. Modele RW2 i RW2i różnią się rodzajem mocowania. Zdecydowano się wykorzystać te mocowane wewnętrznie dla większej wygody projektowania PCB. Warto zaznaczyć że pomimo, że koła od Pololu są powszechnie stosowane, popularniejsze wydaje się być wykorzystywanie kół i opon własnej konstrukcji. Będzie to rozważane jako potencjalna rozbudowa niniejszej konstrukcji w przyszłości.

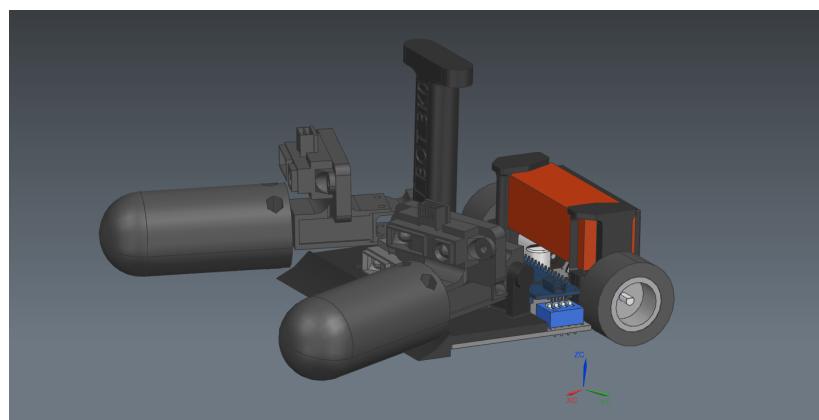


Rysunek 2.9. Koła Solarbotics RW2i

2.2.5. Złożenie



Rysunek 2.10. Projekt robota w ułożeniu startowym



Rysunek 2.11. Projekt robota po rozłożeniu ramion

Łącząc wszystkie wyżej wymienione elementy oraz dodając kilka gotowych otrzymujemy projekt konstrukcji mechanicznej robota minisumo wyposażonego w ramiona, zdolne do podnoszenia przeciwników. Robot rozpoczyna walkę w konfiguracji widocznej na rys. 2.10 . Następnie następuje start silników w ramionach i robot wykorzystując zamontowany na pługu ślupek przechodzi do konfiguracji widocznej na rys. 2.11.

2.3. Wybór komponentów

2.3.1. Silniki dla kół



Rysunek 2.12. Silnik Pololu HP

Obecnie odchodzi się od popularnych dawniej "przeróbek" serwomechanizmów [9], które dominowały w konstrukcjach robotów minisumo jeszcze kilka-kilkanaście lat temu. Stosunkowo niewielkie rozmiary i cena, które zapewniało tamto podejście, są wypierane również niewielkimi silnikami HP Pololu. Na uwagę zasługują także silniki firmy Faulhaber, które przewyższają w swoich osiągach silniki od Pololu. Cena jest jednak na tyle odstraszająca, że nie zapowiada się, aby silniki Pololu szybko zniknęły z zawodów minisumo oraz innych amatorskich konstrukcji urządzeń mechanicznych.

W niniejszym robocie zdecydowano się na skorzystanie z dostępnych we własnym warsztacie silników Pololu HP o identyfikatorze 998, z przekładnią 50:1. Jeżeli chodzi o silniki tego producenta, na zawodach można spotkać przekładnie 30:1 i 50:1. Oba modele zapewniają dużą szybkość, niezbędną żeby znaleźć przeciwnika, oraz duży moment obrotowy potrzebny w zwarciu frontalnym. Różnice między nimi przedstawia tabela 2.1.

Robot opisywany w niniejszej pracy będzie więc wyposażony w nieco wolniejszy napęd, ale za to o większym momencie. Współgra to z założeniami konstrukcyjnymi. Robot wyposażony w ramiona ma wygrywać w starciu frontalnym, a nie wyprzedzać przeciwnika. Przeciwdziałaniem próbom zajścia przeciwnika od tyłu niniejszego robota zajęto się programowo w dalszej części pracy (rozdział 4.2).

Tabela 2.1. Charakterystyka silników Pololu z różnymi przekładniami przy maksymalnej efektywności

Przekładnia	30:1	50:1
Rzeczywiste przełożenie	29,86:1	51,45:1
Prędkość [RPM]	830	490
Moment [kg · mm]	1	1,5
Prąd [A]	0,36	0,32
Prąd przy zatrzymanym wale [A]	1,6	1,6

2.3.2. Silniki dla ramion

Silniki do ramion robota powinny być wolniejsze od tych na kołach. Zdecydowano się na przekładnię 250:1, która zapewnia około 2 obroty na sekundę. W przekładni o takim przełożeniu jest wyposażony silnik Pololu 995. Skorzystano z zamiennika firmy Wagney, oszczędzając istotnie zasoby pieniężne, być może kosztem niektórych parametrów lub dokładności. Biorąc pod uwagę umiejscowienie tych silników i ich zadanie, wydaje się to jednak być zasadne. Silniki nie muszą kręcić się z dokładnie tą samą prędkością, czy posiadać taką samą wartość momentu, zatem dokładność wykonania¹ nie jest aż tak istotna.

2.3.3. Wykrywanie przeciwnika – czujnik odległości



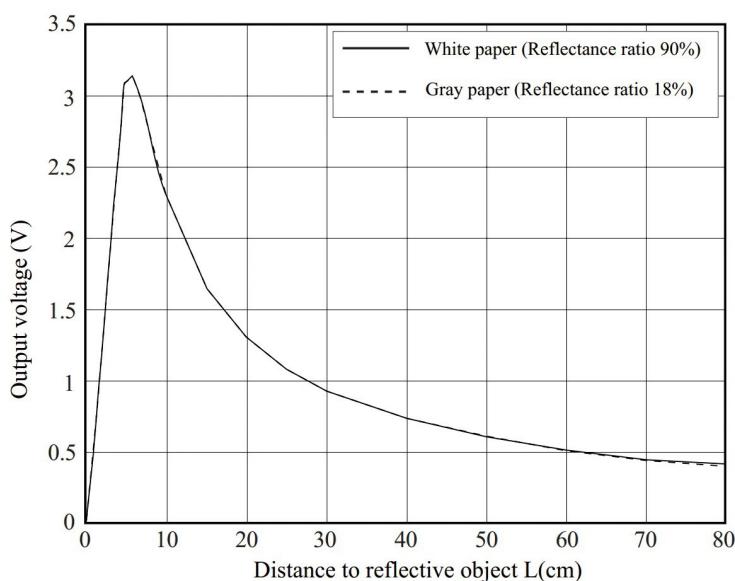
Rysunek 2.13. Czujnik odległości Pololu GP2Y0A21YK0F

W kategorii takiej jak walki minisumo czujniki odległości są w zasadzie jedynymi danymi wejściowymi algorytmu lokalizacji przeciwnika, dlatego tak ważny jest ich odpowiedni dobór. Częstymi wyborami dla minisumo są czujniki optyczne marki Sharp i Pololu. Wcześniej doświadczenia autora sugerują, że czujniki marki Pololu mimo

¹ Zakładając, że cena wynika z jakości wykonania

małych rozmiarów, mogą powodować dodatkowe problemy. Ze względu na ich szeroki kąt i potencjalne interpretowanie siebie (robotu) lub podłogi jako przeciwnika, należałoby zastosować dodatkową obudowę. Wybrano więc większy i łatwy w montażu czujnik marki Sharp GP2Y0A21YK0F.

Czujnik działa na podczerwień na zasadzie triangulacji. Jego zasada działania nie jest jednak tak istotna, ponieważ to, co odbierać będzie mikrokontroler, to napięcie z zakresu 0 – 3 V [10]. Istotnym z punktu widzenia robota minisumo może być także widoczny na rys. 2.14 wykres przykładowej charakterystyki. Wynika z niej, że odległość 0 – 2 cm algorytm mógłby później zinterpretować jako brak przeciwnika w zasięgu. Nie jest to jednak zagrożenie ani w przypadku czujników w ramionach, ani czujnika frontalnego. Te pierwsze z uwagi na wielkość ramienia naturalnie zachowują dystans od przeciwnika. Oponent nie powinien zbliżyć się także wystarczająco do głównego przedniego czujnika ze względu na kręcące się obok niego ramiona.



Rysunek 2.14. Przykładowa charakterystyka z dokumentacji czujnika odległości Pololu GP2Y0A21YK0F [10]

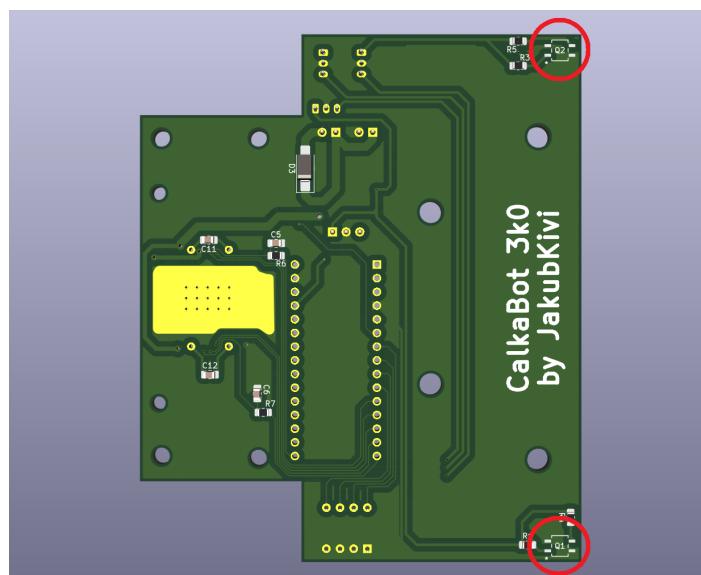
2.3.4. Wykrywanie linii – czujnik zmian jasności podłoża

Jak większość robotów minisumo, niniejszy również wykrywa linie na podłożu za pomocą transoptora. Opisywane w rozdziale 1.3 roboty wykorzystywały układy QRB1134 [9] i TCRT1010 [5]. Oba cechuje dość duży rozmiar, który nie jest do przyjęcia biorąc pod uwagę pozostałe cechy robota. Wysunięte ramiona nie tylko ograniczają przestrzeń, ale także wysuwają do przodu środek ciężkości konstrukcji. Szybkie wykrycie linii może okazać się kluczowe, aby robot nie wypadł z planszy mimo hamowania czy zwracania. W związku z tym zdecydowano się wysunąć transenty możliwie jak najbardziej do



Rysunek 2.15. Transoptor KTIR0711S

przodu robota, co zaznaczono na rys. 2.16. Aby to osiągnąć skorzystano z transoptorów KTIR0711S. Cechuje je wysokość jedynie 1,5 mm [11], co dobrze sprawdzi się w niniejszym zastosowaniu.



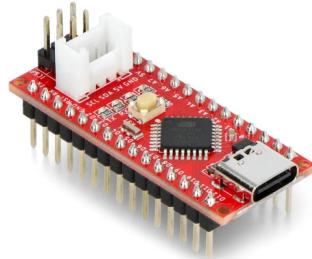
Rysunek 2.16. Projekt płytki PCB 2.4 z zaznaczonymi na czerwono położeniami transoptorów

2.3.5. Główny układ sterujący

Ze względu na własne doświadczenie zdecydowano się na użycie mikrokontrolera AVR jako głównego modułu sterującego. Jednak w związku z potrzebą minimalizacji błędów na PCB zdecydowano się na użycie gotowego modułu Arduino Nano. Takie rozwiązanie pozwala uprościć projekt własnej płytki, ponieważ Arduino Nano jest już wyposażony w układ do programowania mikrokontrolera przez USB, zewnętrzny oscylator kwarcowy czy przycisk reset. Gotowy moduł jest także łatwo wymienialny w przypadku problemów z mikrokontrolerem. To samo można też osiągnąć stosując mikrokontroler w obudowie THT z podstawką, ale słabe doświadczenia w projektowaniu układów PCB połączone z dość długim czasem walidacji poprawności takiej płytki (zamówienie, sprawdzenie

2. Projektowanie

"poprawności" przez producenta, dostawa, lutowanie elementów, by na koniec dowieźć się czy działa zgodnie z założeniami) przesądziły o wykorzystaniu Arduino Nano, a dokładniej tańszej wersji niezależnego producenta, czyli tzw. klona, Seeeduino Nano. Wybranie tego konkretnego modelu podiktowane jest wyłącznie wzgledami estetycznymi, dlatego że ma on czerwoną soldermaskę². Układ programatora płytki Seeeduino jest inny, niż w oryginalnym Arduino, jednak niedogodność związana z instalacją dodatkowego sterownika była jedyną napotkaną różnicą.



Rysunek 2.17. Płytnka Seeeduino Nano

2.3.6. Akumulator

Projekt konstrukcyjny przewidywał dokładne rozmiary konkretnego pakietu Li-Po Dualsky o pojemności 520 mAh. Akumulatory litowo-polimerowe to najpopularniejszy [5][6] wybór, jeśli chodzi o roboty minisumo w związku z swoją dużą wydolnością prądową, która potrzebna jest do zasilania silników. Niegdyś ze względu na swoją wysoką cenę ustępowały miejsca akumulatorom litowo-jonowym czy niklowym [12], dziś jednak cena akumulatora nie jest aż tak odstraszająca dla konstruktorów.

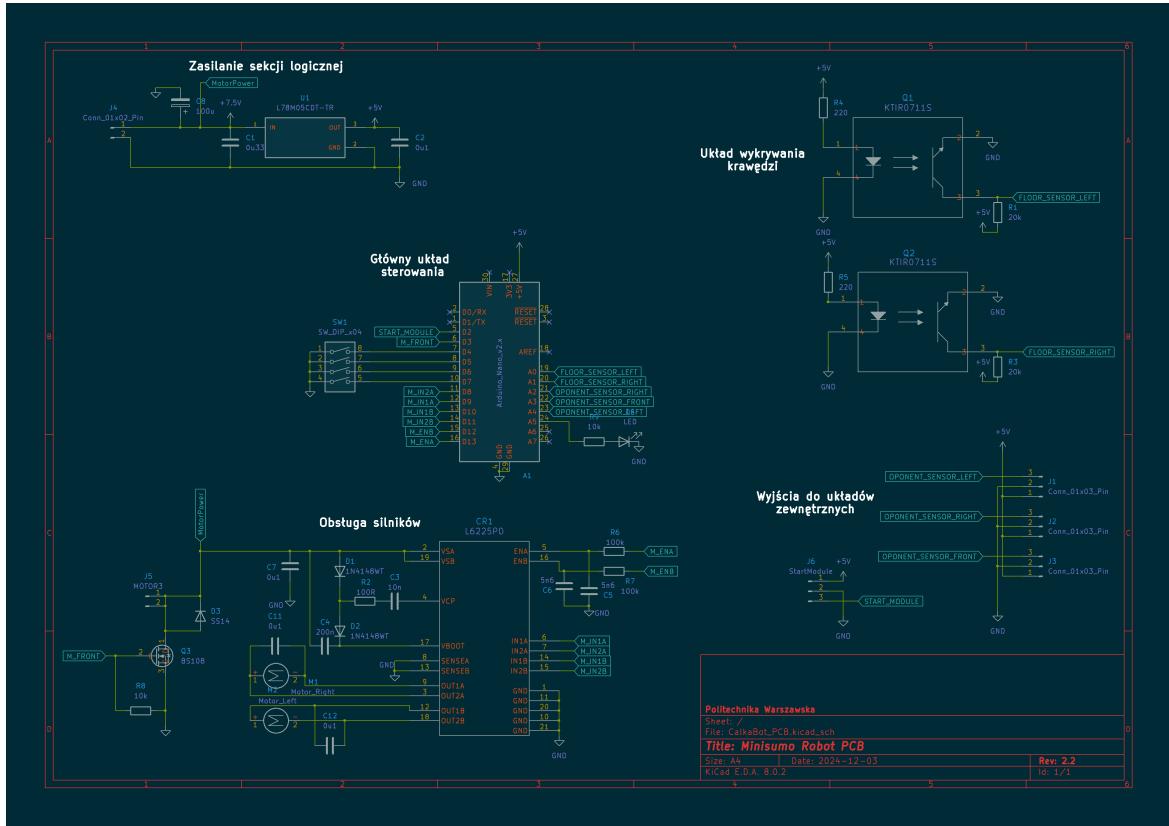
2.4. Projekt PCB

Podczas projektowania PCB zdecydowano się wykorzystać oprogramowanie KiCAD. Jego duża baza gotowych komponentów powiększona o biblioteki udostępniane przez użytkowników jest zachęcająca, choć jak ma się później okazać, także problematyczna. Schemat podzielono na sekcje widoczne na rys. 2.18.

2.4.1. Układ zasilania

Do zasilania sekcji logicznej wykorzystano popularny stabilizator L78M05, który wykorzystany zgodnie z założeniami dokumentacji technicznej [13], nie będzie miał problemu

² Soldermaska (ang. solder mask) to warstwa ochronna naniesiona na powierzchnię obwodu drukowanego (PCB), która zabezpiecza ścieżki miedziane przed utlenianiem i zwarciem oraz ułatwia lutowanie poprzez odsłonięcie jedynie pól lutowniczych (tzw. padów). Najczęściej występuje w formie zielonej powłoki polimerowej.



Rysunek 2.18. Schemat płytki PCB do sterowania robotem z ramionami

z zasileniem głównego układu logicznego. Dodano także to tej sekcji kondensator elektrolityczny, który umieszczony pomiędzy układem sterowania, a układem obsługi silników pełni funkcje zabezpieczające przed "szpilkami" i spadkami napięcia. Warto dodać, że cały układ zasilania jest niejako pozostałością po wstępnej koncepcji wykorzystującej mikrokontroler AVR bezpośrednio. Wykorzystana płytka Seeeduino ma w sobie już układ stabilizacji napięcia i można było go wykorzystać.

2.4.2. Układ sterowania

Główny układ sterowania poza przydatnymi oznaczeniami pinów opisywanego już wcześniej (2.3.5) Seeeduino, ma także 4 przełączniki typu microswitch. To częste rozwiązanie w robotach minisumo, daje dodatkowe możliwości manewrowe podczas zawodów. Konstruktor może wybrać, który z wcześniejszej zaimplementowanych algorytmów sterowania wykorzystać przeciwko konkretnemu przeciwnikowi nawet pomiędzy rundami. W związku z wykorzystaniem pinu D13³ głównej płytki do obsługi silników, dodano także diodę informacyjną do innego wolnego pinu.

³ Zgodnie z standardem Arduino, płytka Seeeduino na pinie D13 (PORTD3) wyposażona jest w informacyjną diodę

2.4.3. Układ sterowania silnikami

Obsługę silników kół napędowych zrealizowano przy użyciu układu podwójnego mostka L6225. Jak czytamy w dokumentacji technicznej układu [14], napięcie zasilania silników (opisywany 2.3.6 akumulator 7,4 V) jest znacznie poniżej maksymalnej wartości 52V. Napięcie na pinach wyjściowych mikrokontrolera AVR, a więc 5 V lub 0 V, będzie poprawnie interpretowane przez część logiczną mostka jako stan wysoki lub niski. Zabezpieczenie przeciwprądowe, nie odłączy silników nawet w pełnym zwarciu gdyż:

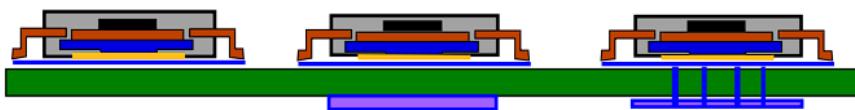
$$I_{SOVER} > I_{max}$$

$I_{SOVER} \in < 2; 3,55 >$ A - Napięcie graniczne płynące przez kanał mostka przy którym zadziała zabezpieczenie przeciwprądowe [14]

$I_{max} = 1.6$ A - Teoretycznie wyznaczona przez producenta silnika, maksymalna wartość prądu przy zatrzymanym wale silnika [15]

Ważnym czynnikiem przy projektowaniu PCB zawierającego układ L6225 jest zarządzanie ciepłem. Dla minimalizacji rozmiarów zdecydowano się skorzystać z układu w obudowie PowerSO20, która swoje małe rozmiary zawdzięcza między innymi swojej metalowej obudowie na spodzie. Jest ona podłączona do masy i przeznaczona do odprowadzania ciepła. Zatem żeby zminimalizować ryzyko przegrzania układu zastosowano otwory przelotowe wraz z powierzchnią do odprowadzania ciepła na spodzie, która nie została pokryta polimerem jak reszta płytki.

Sposoby montażu układów podatnych na przegrzanie widać na rys. 2.19. Zielonym kolorem oznaczono wnętrze PCB, niebieski kolor wskazuje na połączenia z miedzi lub cyny, żółty kolor to element, od którego odbierane jest ciepło. Od lewej: Element odbierający ciepło przylutowany do PCB z obszarem rozpraszania ciepła po tej samej stronie płytki. Na środkowym widać dodatkowy element odprowadzający ciepło również z drugiej strony płytki. Po prawej, najbardziej efektywne rozwiązanie, gdzie obie strony są dodatkowo połączone elementem lepiej przewodzącym ciepło niż włókno szklane płytki PCB

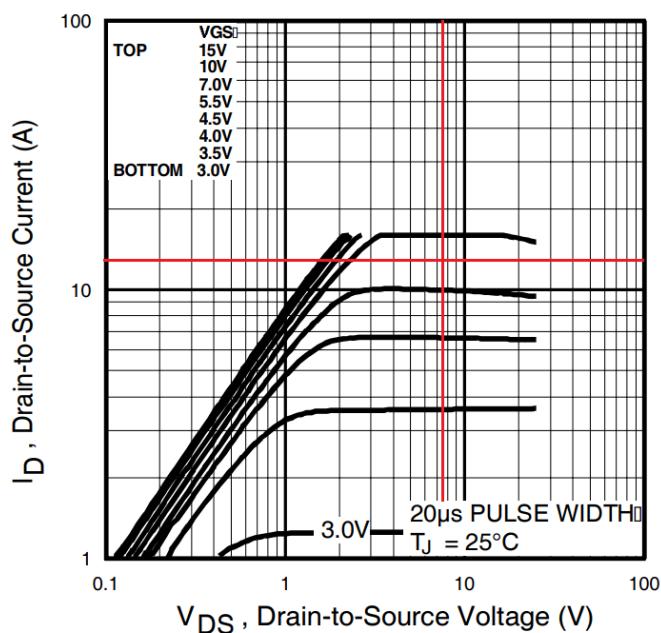


Rysunek 2.19. Przekrój boczny płytka PCB prezentujący trzy sposoby montażu układów podatnych na przegrzanie. [14]

Zastosowano się do opisywanego w dokumentacji typowego zastosowania, a więc poza filtrującymi kondensatorami i zasilaniem zaprojektowano także układ pompy ładunkowej. Do układu podłączono także piny sterujące mikrokontrolera.

Do sterowania silników ramion wykorzystano tranzystor MOSFET, w związku z tym, że silniki ramion kręcić się mają zawsze w jedną stronę. Nie trzeba sterować kierunkiem

obrotów, a jedynie odizolować układ sterowania od wyższego napięcia. Tranzystor MOSFET IRLL014N w swojej niewielkiej obudowie spełnia powyższe założenia. Dodatkowo na rys. 2.20 zaznaczono dla $V_{GS} = 5$ V (napięcie na pinach wyjściowych mikrokontrolera AVR) i $V_{DS} = 7,4$ V (napięcie zasilania z akumulatora) odczytano maksymalny prąd drenu $I_D \approx 12$ A. Jest to maksymalny prąd jaki mógłby płynąć przez ten MOSFET. Przy zatrzymanych wałach obu silników prąd płynący przez tranzystor wyniesie około 3.2 A[15], a więc wielokrotnie mniej. Zamontowano także diodę spolaryzowaną przeciwnie do kierunku zasilania silników, aby po wyłączeniu silników napięcie powstałe w wyniku ich bezwładnego kręcenia i indukujące się w uzwojeniach silnika w chwili wyłączenia zasilania nie uszkodziło tranzystora.



Rysunek 2.20. Charakterystyka układu IRLL014N z zaznaczonym odczytem [16]

2.4.4. Układ wykrywania krawędzi

Układ wykrywania krawędzi ringu zrealizowano zgodnie z wytycznymi dokumentacji technicznej [11] wspomnianego wcześniej układu KTIR0711S (2.3.4), a więc z wykorzystaniem rezystora do ograniczenia prądu płynącego przez diodę oraz rezystora pull-up przy pinie sygnałowym. Przyłożono szczególną wagę do umiejscowienia czujników jak najbliżej krawędzi płytka, by wykrywanie linii było możliwie najszybsze. Położenie czujników zilustrowano na rys. 2.16 .

2.4.5. Wyjścia do układów zewnętrznych

Zastosowano trzy złącza JST jako wyprowadzenia dla czujników odległości. Takie samo złącze jest wbudowane w układ GP2Y0A21YK0F. Trzypinowe złącze przygotowano dla popularnych na zawodach [3] modułów startowych. To zewnętrzne, często przygotowywane

2. Projektowanie

przez organizatora, układy zdalnego uruchamiania dające bardziej wyrównane szanse niż ręczne włączanie robotów, np. *na trzy*. Na zawodach z ręcznym startem również będzie można wykorzystać to złącze podłączając inny moduł.

2.4.6. Grubość ścieżek

Podczas pracy z większymi prądami niebagatelna okazuje się kwestią grubości ścieżek. W projekcie zastosowano laminat PCB o podwójnej grubości miedzi ($70 \mu m$), co wynika z konieczności przewodzenia znacznych prądów płynących przez układ sterowania silników. Dwa silniki o łącznym prądzie maksymalnym $3,2A$ zgodnie z normą IPC-2221 [17] potrzebowaliby ścieżki o grubości co najmniej $1,5 mm$. Ograniczenia wymiarowe sprawiają, że dodatkowy zapas grubości ścieżki zdecydowano się zrealizować w postaci podwójnej grubości.

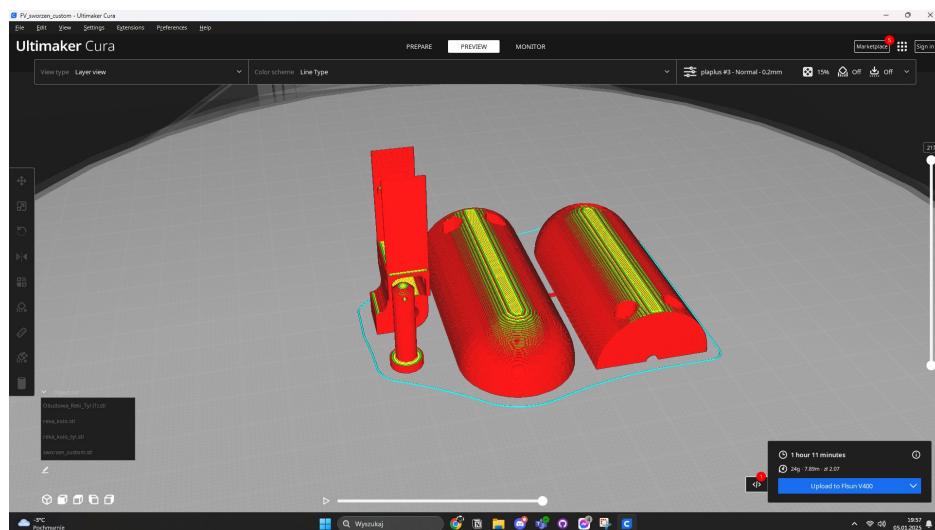
Ścieżki sygnałowe, w których prąd nie powinien przekraczać $20 mA$, mogą być znacznie cieńsze, dlatego dla nich zastosowano $0,5 mm$ grubości.

Przy zmianie grubości ścieżek w dwuwarstwowej płytce (a taką zdecydowano się zastosować) nie można również zapomnieć o grubości przelotek. Tutaj również zastosowano się do norm IPC-2221 z dodatkowym zapasem.

3. Wykonanie robota

3.1. Druk 3D

Do wykonania elementów konstrukcyjnych robota wykorzystano drukarkę 3D Flsun V400. Modele zostały przygotowane do druku za pomocą programu Ultimaker Cura. Jako materiał użyto filamentu ESun PLA Plus, wyróżniającego się podwyższoną wytrzymałością i dobrą jakością powierzchni.



Rysunek 3.1. Wybrane elementy robota przygotowane do druku w programie Ultimaker Cura

Proces druku przeprowadzono z wysokością warstwy 0,2 mm, co pozwoliło na uzyskanie precyzyjnych detali przy zachowaniu rozsądnego czasu realizacji. Grubość ściany wynosiła 0,8 mm, zapewniając wystarczającą wytrzymałość elementów. Druk odbywał się w temperaturze 215°C, co zapewniło dobre zespolenie warstw.

Po usunięciu wsporników oraz wygładzeniu niektórych powierzchni papierem ściernym oraz pilnikiem, części były gotowe, a elementy współpracujące zachowywały się w zaplanowany sposób.

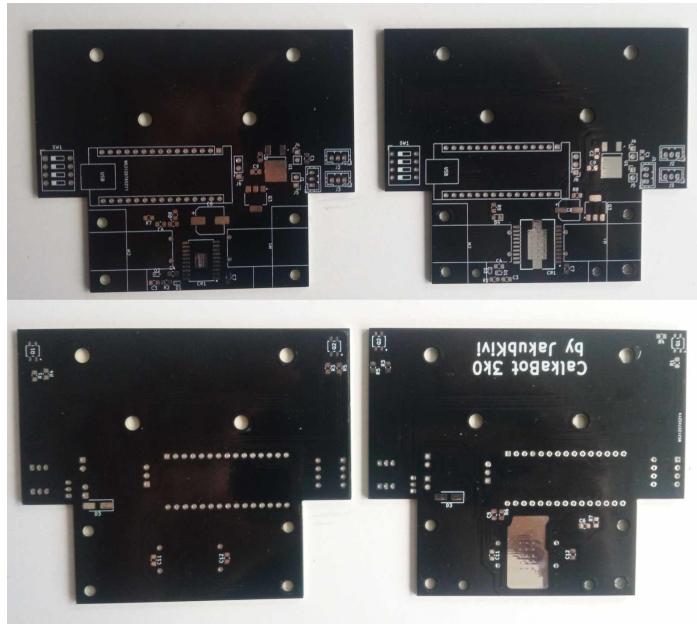
3.2. Wykonanie PCB

Ze względu na zawodność i małe doświadczenie w samodzielnym wytrawianiu płytEK PCB, zdecydowano się zamówić gotowe płytki.

Niestety pierwsza wersja płytKI miała błędnie dobrane (gotowe) wzory montażowe elementów układów do obsługi silników i do wykrywania linii. Jedyne co udało się zatem przetestować na pierwszej wersji płytKI to czujniki odległości, układ zasilania sekcji logicznej oraz przełączniki. Nanieśono poprawki na projekt samemu projektując footprint⁴

⁴ Footprint to obrys lub układ fizyczny elementu na płytce drukowanej (PCB), zawierający pola lutownicze (pady), wymiary obudowy oraz oznaczenia montażowe.

3. Wykonanie robota



Rysunek 3.2. Dwie wersje zamówionej płytki. Po lewej płytka z błędami, a po prawej druga poprawiona i usprawniona wersja

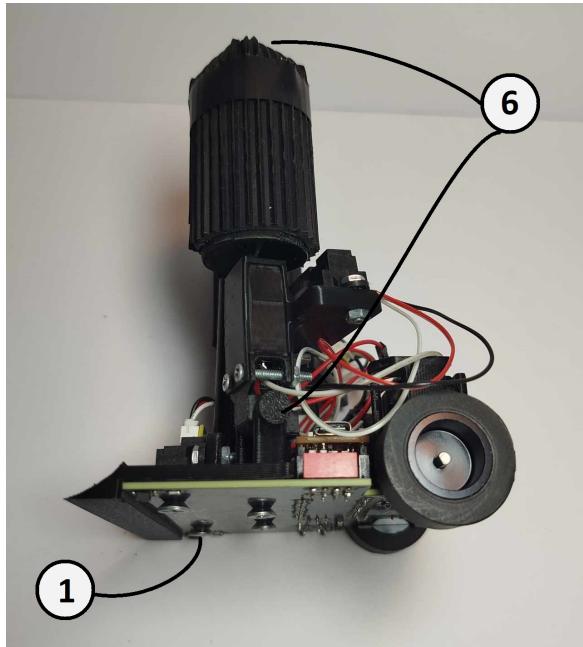
układu L6225PD, dokonano kilku usprawnień widocznych na rys. 3.2 oraz zamówiono drugą wersję płytki.

W związku z opisywanym wcześniej problemem odprowadzania ciepła z układu L6225 przylutowanie go zwykłą lutownicą mogłoby być problematyczne, a ewentualne negatywne skutki błędu (nie rozgrzanie wystarczająco lub przegrzanie) mogłyby się nie ujawnić od razu. Brak specjalistycznego sprzętu jak piec do lutowania rozpływowego udało się jednak zrekompensować domowym sprzętem.

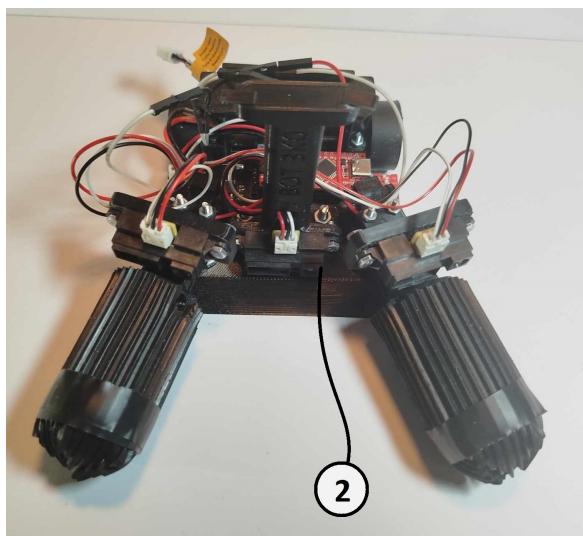
Na patelni rozsypano około 1 cm warstwy piasku przesypywanego przez sitko. Na nim położono dla izolacji od brudu cienką folię aluminiową. Na płytę PCB nałożono pastę lutowniczą w odpowiednich miejscach oraz ułożono elementy. Tu przydatnym dodatkiem do zamówionego PCB może być metalowy szablon do nanoszenia pasty, jednak w związku z niewielkim rozmiarem płytki i dużym kosztem takiego szablonu zdecydowano poradzić sobie bez niego. Tak przygotowaną płytę ułożono na wspomnianej wcześniej folii oraz monitorując temperaturę podgrzewano i ochładzano zgodnie z charakterystyką rozpływów zakupionej pasty. Po kilku poprawkach ręcznych układ przetestowano z powodzeniem. Przyutowano także elementy montowane poprzez przewlekanie oraz te na drugiej stronie płytki, po czym ponownie przeprowadzono udany test.

3.3. Montaż konstrukcji

Zgodnie ze projektem konstrukcji mechanicznej połączono płytę PCB i plug śrubami M3, co zaznaczono na rys. 3.3 numerem 1. Zamontowano środkowy czujnik, oznaczony



Rysunek 3.3. Bok robota z oznaczonymi elementami



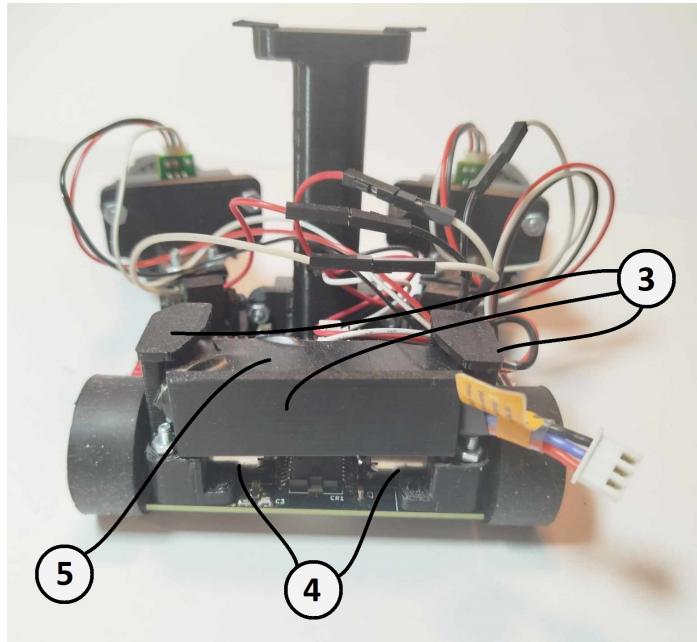
Rysunek 3.4. Górná časť robota z oznaczonymi elementami

na rys. 3.4 numerem 2 i podłączono go do jednego z złącz JST na płytce. Wszystkie złącza widoczne są na rys. 3.7.

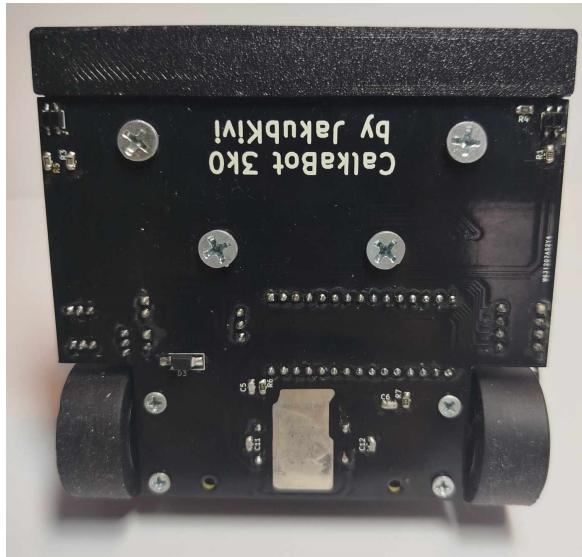
Z tyłu zamontowano mocowanie baterii i silniki, oznaczone odpowiednio numerem 3 i 4 na rys 3.5. Baterie oklejono czarną taśmą dla estetyki i kamuflażu i umieszczone na swoim miejscu, zaznaczonym na rys. 3.5 numerem 5. Silniki przylutowano bezpośrednio do płytki. Przykręcono na ich wale wspomniane wcześniej koła z wykorzystaniem środka do zabezpieczania gwintów, który przy tak wibrującej konstrukcji wydaje się rozsądny. Koło wraz z wałem jest dobrze widoczne na rys. 3.3.

W ramionach umieszczono silniki z wyprowadzonymi przewodami i skręcono śru-

3. Wykonanie robota

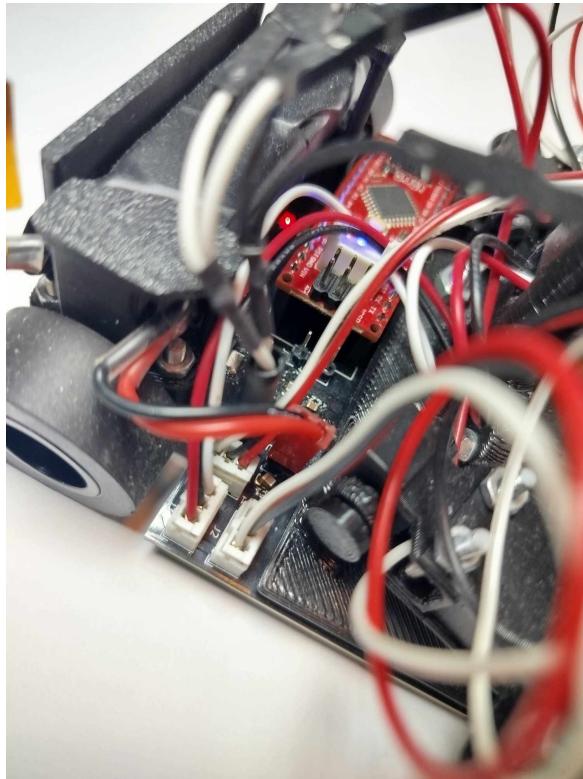


Rysunek 3.5. Tył robota z oznaczonymi elementami



Rysunek 3.6. Dolna część robota

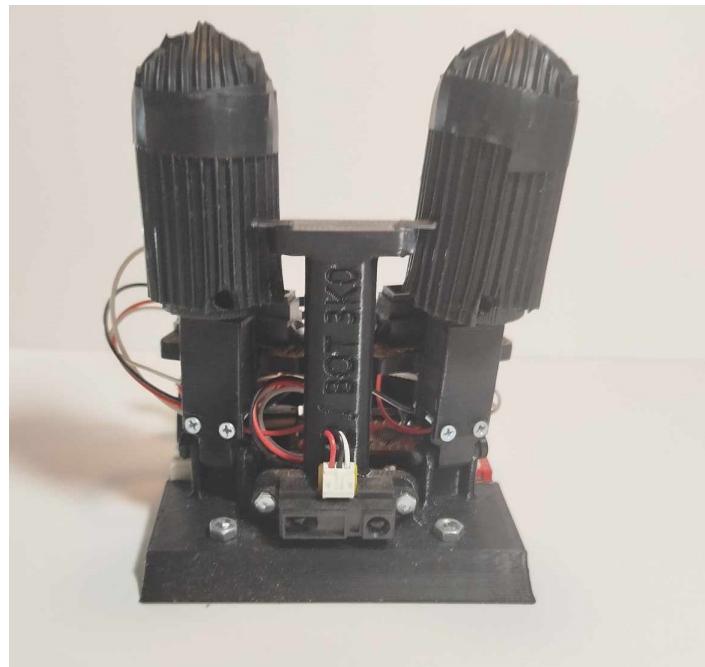
bami zgodnie z projektem na rys. 2.6. Na wale silnika skręcono wydrukowane walce. Oklejono je następnie ryflowaną gumową wykładziną, dla zwiększenia przyczepności. Obręcz kontaktu z podłożem oklejono taśmą dla minimalizacji tarcia. Do górnej części ramion przykręceno pozostałe dwa czujniki. Złożone ramię jest widoczne i zaznaczone numerem 6 na rys. 3.3 Tak przygotowane ramiona zamontowano na wydrukowanych sworzniach i podłączono pod pozostałe złącza JST. Silniki napędowe ramion podłączono do dwupinowego złącza.



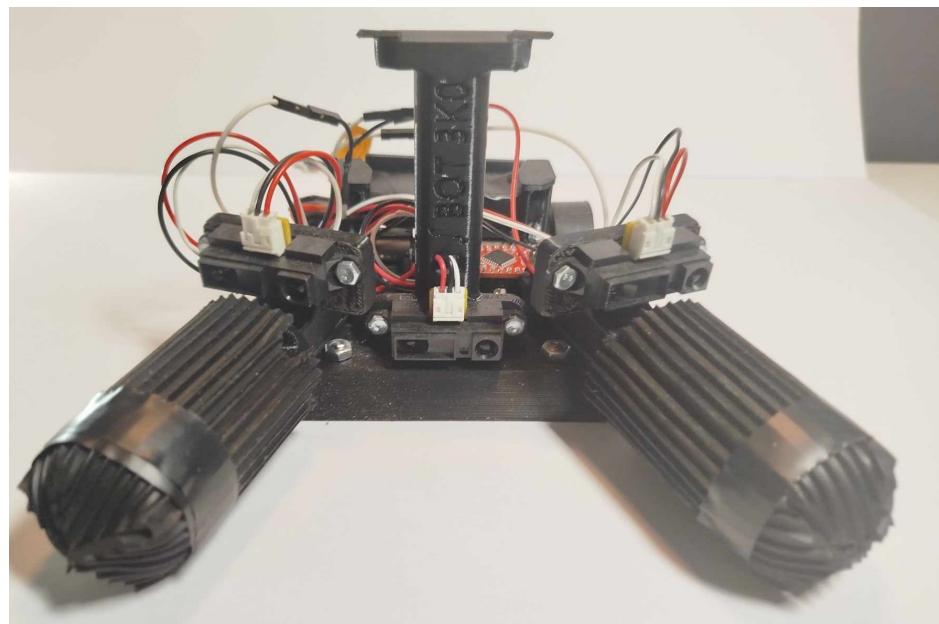
Rysunek 3.7. Złącza robota na płytce PCB

3.4. Dodatkowe obciążenie

Po zmontowaniu robota zważono go uzyskując wynik 300 g. Zgodnie z wspominanymi dobrymi normami robota należało dociążyć. Wykonano ołowiany prostopadłościan $1 \times 3.3 \times 5$ cm. Umieszczono go w przewidzianym wcześniej miejscu w plugu uprzednio oklejając taśmą izolacyjną dla kamuflażu i estetyki.



Rysunek 3.8. Wykonany robot w pozycji startowej



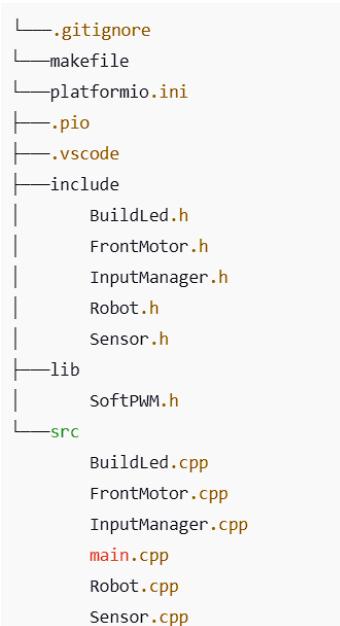
Rysunek 3.9. Wykonany robot z rozłożonymi ramionami

4. Oprogramowanie

4.1. Architektura systemu

Najpopularniejszym wyborem do programowania mikrokontrolerów AVR jest język C. Jego niskopoziomowe możliwości i kontrola programu są jak najbardziej pożądane. Nie tracąc jednak wiele na pamięci operacyjnej sterownika, której i tak w pełni nie wykorzystano, zdecydowano się na podejście obiektowe i skorzystanie z możliwości języka C++. Taki modularny kod będzie znacznie łatwiejszy do wykorzystania w kolejnych robotach minisumo i nie tylko.

Obiektywność znacznie zwiększa również czytelność kodu w głównym programie, gdyż wszystkie klasy wydzielono do plików *.h* i *.cpp*. Na potrzeby niniejszego robota powstała struktura projektu widoczna na rys. 4.1.



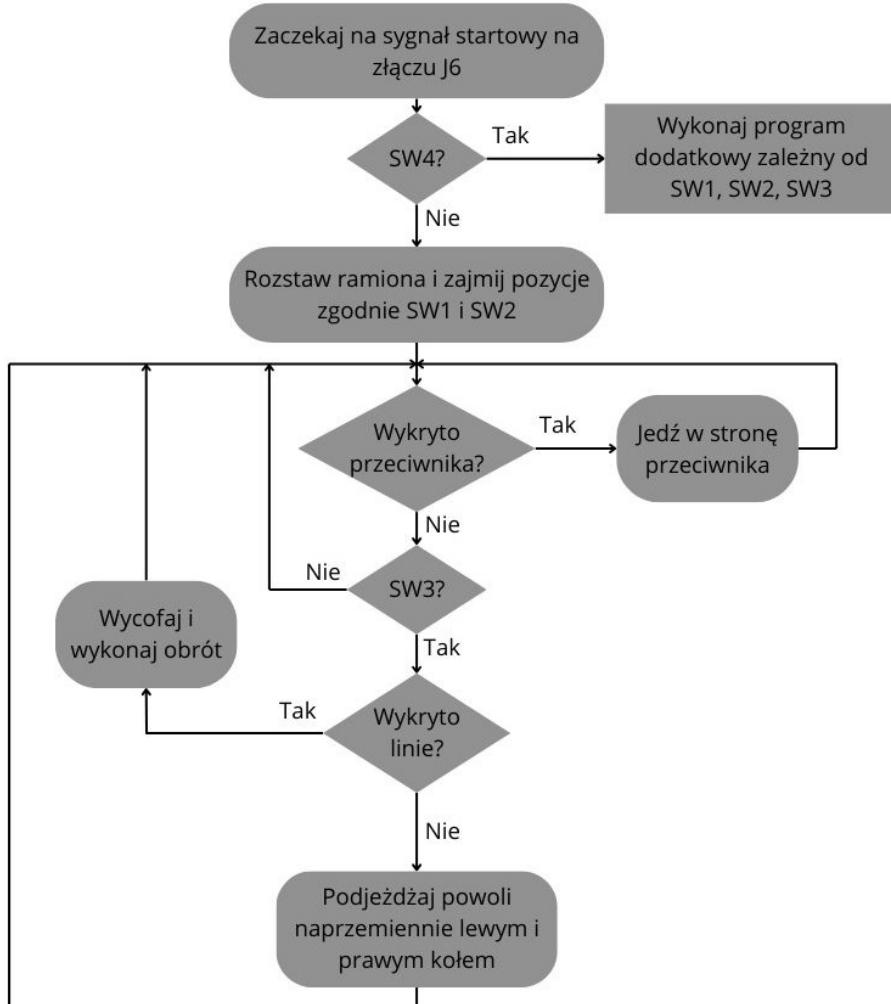
Rysunek 4.1. Struktura projektu oprogramowania robota minisumo

Początkowo planowano nie korzystać z biblioteki Arduino, wynikiem czego jest widoczny plik *makefile*. Pliki edytowane w VSCode komplikowano za pomocą *avr-gcc* i wgrywano używając *avrdude*. Wszystko działało prawidłowo, dlatego konstruktory klas przyjmują adresy portów i numery pinów jako parametry wejściowe. W przyszłości można rozbudować klasy o konstruktory z wykorzystaniem numeracji biblioteki Arduino, dla zwiększenia uniwersalności.

W toku prac zauważono jednak niewielki błąd w projekcie PCB. Piny odpowiedzialne za ustalanie prędkości silników w kołach robota nie były wyposażone w PWM. Zgodnie z zasadą "braki w hardwarze nadrabaj softwarem", zdecydowano się na wykorzystanie biblioteki SoftPWM. Samodzielne próby implementacji mechanizmu PWM z wykorzystaniem przerwań zegarowych mikrokontrolera działały poprawnie, jednak były znacznie

mniej niezawodne i dostosowywalne. Biblioteka SoftPWM sama korzysta z biblioteki Arduino. W związku z tym zrezygnowano z dalszego unikania użycia biblioteki Arduino, którą starano się omijać ze względów oszczędności zasobów i czasu (wgrywania, ale i działania programu). Do projektu w VSCode dodano zatem narzędzie PlatformIO. Konfiguracja komplikacji i wgrywania odbywa się teraz zatem w pliku platformio.ini.

4.2. Algorytmy sterowania



Rysunek 4.2. Schemat blokowy oprogramowania robota minisumo

Jak przedstawiono na schemacie blokowym układu sterowania w podrozdziale 2.2, robot wyposażony jest w 5 analogowych czujników, 4 przełączniki microswitch oraz moduł startowy. Na podstawie tych danych wejściowych mikrokontroler steruje poprzez układ po-dwójnego mostka zwrotem i prędkością dwóch niezależnych silników, diodą informacyjną oraz silnikami w ramionach, choć tu jedynie może je włączyć lub wyłączyć.

Jak przedstawiono na schemacie blokowym oprogramowania na rys. 4.2, na początku robot oczekuje sygnału startowego. Opisywane wcześniej moduły startowe (podrozdział 1.2), będą podłączone do złącza J6, o którym pisano w poprzedniej części pracy w podrozdziale 2.4.5. Do tego samego złącza można również podłączyć prosty układ do ręcznego sterowania robota. Z punktu widzenia mikrokontrolera nie ma jednak znaczenia co jest tam podłączone. Robot zawsze oczekuje na sygnał na złączu J6.

Następnie, jeżeli włączono 4. microswitch (SW4), robot może przejść do jednego z dodatkowych programów, które opisano dalej, lub przechodzi do stanu zaznaczonego na grafie stanów (rys. 2.3) jako "Zajmowanie pożądanej pozycji na ringu".

Na podstawie zaprezentowanych wcześniej przykładów (podrozdział 1.3) można dojść do wniosku, że robot z ramionami wygrywa starcia frontalne. Za kluczowe uznano zatem korzystne ustawienie robota na początku walki.

Sposób rozstawienia robotów na początku walki jak wspominano jest różny i uzależniony od regulaminu konkretnych zawodów. Aby zatem zwiększyć uniwersalność niniejszej konstrukcji, przełączniki (SW1, SW2...) wykorzystano w następujący sposób:

- **SW1** — Określenie orientacji początkowej robotów:
 - **1 – Przodem do siebie** — roboty są ustawione pługami ku sobie.
 - **0 – Tyłem do siebie** — roboty są ustawione pługami w przeciwnych kierunkach.
- **SW2** — Dodatkowa korekta pozycji. Regulaminy minisumo nakazują niekiedy uczestnikom ustawienie swoich konstrukcji wzdłuż pewnej linii [3]. Wówczas aby obrócić się przodem do przeciwnika, należy dokonać korekty:
 - **1 – W lewo**
 - **0 – W prawo**
- **SW3** — Poszukiwanie przeciwnika. Ustawienie odpowiedzialne za zachowanie robota po korekcie pozycji:
 - **1 – Włączone** — Powolne przeszukiwanie pola walki.
 - **0 – Wyłączone** — Zatrzymanie robota i czekanie na wykrycie przeciwnika.
- **SW4** — Tryb serwisowy i programy testowe. Po przełączeniu tego przełącznika, przełączniki SW1-SW3 zamiast działać jak opisano wyżej, wybierają jeden z 8 zapisanych programów:
 - **1 – Włączone**
 - **0 – Wyłączone**

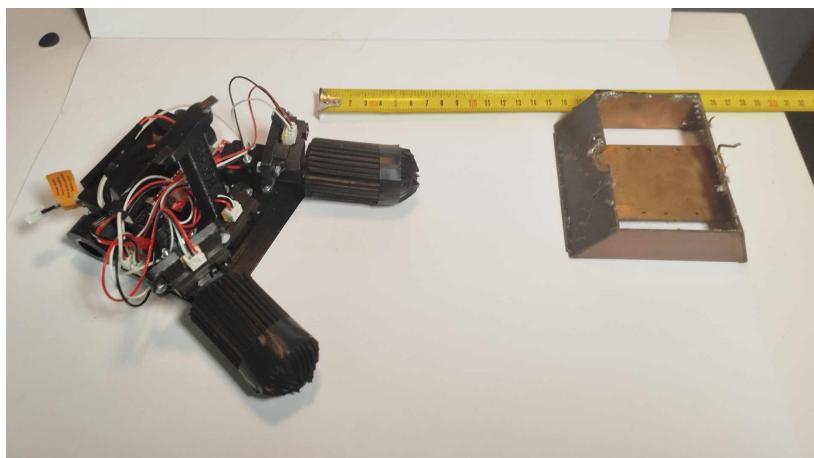
To dodatkowe wykorzystanie komunikacji między konstruktorem a algorytmem powinno zwiększyć skuteczność robota w walkach. Przełączniki SW1–SW3 pozwolą na dostosowanie się do konkretnego regulaminu zawodów oraz do konkretnego ułożenia na planszy i niekiedy do zachowania przeciwnika. Przełącznik SW4 pozwala na aż 8 dodatkowych programów serwisowych, testowych lub też specyficznych dla konkretnych zawodów. Do takich programów będą się zaliczać np.:

4. Oprogramowanie

- Program do czyszczenia kół – kręcenie się kół w jedna stronę niezależnie od odczytów,
- Dodatkowe programy, gdy regulamin zawodów pozwala na ustawienie robotów na wprost,
- Dodatkowe programy testowe lub pokazowe,
- Dodatkowe programy przygotowane przeciwko konkretnym przeciwnikom lub na konkretne zawody.

4.3. Wstępna kalibracja czujników

W celu ustalenia zakresów odczytów przeprowadzono eksperyment z wykorzystaniem faktycznego ringu oraz ramy innego robota minisumo. Rama była czarna, co pozwoliło urzeczywistnić warunki testowe. Warto jednak mieć na względzie, że konstrukcje przeciwników mogą być niższe, mogą być pod ostrym kątem względem podłoża, a autorowi znany jest przykład robota nastawionego wyłącznie na kamuflaż, używającego gładkich powierzchni płyta, co prawie uniemożliwiało jego wykrycie przez czujniki optyczne. Aby zminimalizować niebezpieczeństwo przeoczenia przeciwnika, poniższe odczyty potraktowano jako dobry punkt startu, a ostateczne ustawienia czujników będą powstawać po kolejnych testach praktycznych.



Rysunek 4.3. Fotografia stanowiska do kalibracji czujników

Tabela 4.1. Odczyty z czujników Sharp w różnych odległościach

Odległość mm	Lewy	Środkowy	Prawy
Bez przeszkody	150 ± 30	140 ± 20	150 ± 20
300	190 ± 20	230 ± 20	190 ± 20
150	320 ± 20	350 ± 20	310 ± 20
75	320 ± 20	620 ± 20	320 ± 20

Jak widać w tabeli 4.1, wszystkie czujniki są w stanie wykryć przeszkody.

Wartości w tab. 4.1 i tab. 4.2 odpowiadają napięciu 0–5 V zapisanemu w rozdzielcości 10-bitowej (rozdzielcość przetwornika analogowo cyfrowego w Atmega328). W związku z tym, że z tej samej rozdzielcości korzystano podczas programowania, zamiana wartości wydała się zbyteczna.

Tabela 4.2. Odczyty z czujnika wykrywania linii

Powierzchnia	Lewy	Prawy
Czarna - Standardowa powierzchnia ringu	760 ± 20	500 ± 20
Białą - Linia	0-48	0-43

Odczyty z czujnika wykrywania linii różnią się nieco od siebie mimo symetrii konstrukcji. Może to wynikać np. z niedokładności rezystorów pull-up przy transoptorze. Po przeprowadzeniu inspekcji wizualnej widać jednak niesymetrię w montażu SMD samych transoptorów. Różnica pomiędzy odczytami z wnętrza ringu, a linii jest jednak na tyle duża, że rozsądny wydaje się to zignorować.

4.4. Test nośności

W niniejszej pracy korzystano z założenia, że przy starciu frontalnym niniejszy robot podniesie przeciwnika, co da mu wystarczającą przewagę, aby wypchnąć przeciwnika z ringu. Aby przetestować to założenie przeprowadzono prosty test.

Na ringu do minisumo umieszczono niniejszą konstrukcję wraz z ramą innego robota minisumo. Dzięki temu można przewidywać, że warunki "chwytu" robota są podobne do tych na ringu. Przeciwnik, czyli wspomniana rama, został dodatkowo dociążony świeczką wypaloną tak, by jej masa wraz z ramą odpowiadała przybliżonej masie robota minisumo. Stanowisko pomiarowe z opisywanym "przeciwkiem" przedstawiono na rys. 4.4.

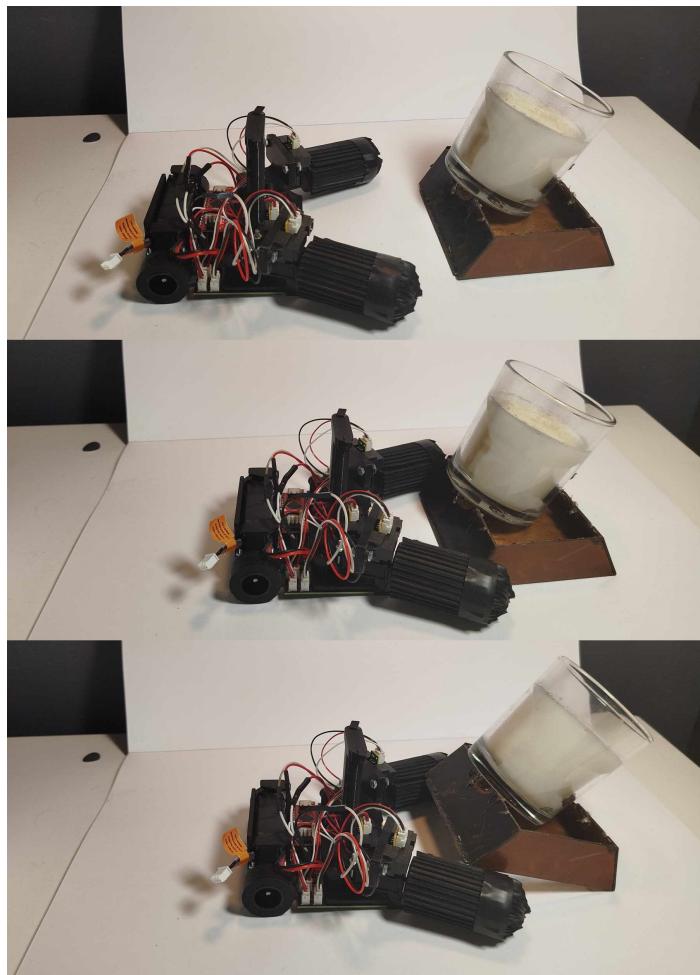


Rysunek 4.4. Fotografia stanowiska do testu nośności

Do mikrokontrolera został wgrany program, dzięki któremu robot ma jechać do przodu i kręcić silnikami w ramionach. Pomimo widocznej na rys. 4.4 masy ok. 480 g, czyli standardowej masy robota minisumo, niniejsza konstrukcja nie ma problemu z podnoszeniem

4. Oprogramowanie

ramy. Na rys. 4.5 przedstawiono 3 klatki nagrania eksperymentu, które ilustrują zamierzane działanie konstrukcji.



Rysunek 4.5. Trzy klatki z nagrania opisywanego eksperymentu

W opisywanym eksperymencie rama nie próbuje przepchnąć robota jak na prawdziwych zawodach. Jeśli jednak podczas zawodów zaobserwowane zostanie przegrywanie robota w starciach frontalnych, będzie to interpretowane jako słabość opon i zostanie przeprowadzona wymiana kół na inną konstrukcję.

5. Podsumowanie

5.1. Samokrytyka

Na słowo samokrytyki zasługują na pewno błędy w projekcie PCB. Pomimo dwóch iteracji zamówień nie udało się wyeliminować wszystkich błędów. Niezadbanie o piny PWM do obsługi prędkości kół kosztowało nieco pamięci i prędkości mikrokontroler, jednak nie przekreśliło to działania robota.

Kolejnym niepodważającym szans robota błędem jest niewykorzystanie pinu *AREF*⁵ mikrokontrolera. Czujniki odległości dają sygnał z zakresu 0 – 3[V], a odczyty z czujnika podłożą wskazują zakres 0 – 4[V]. Podłączenie pinu *AREF* do napięcia 3[V] zwiększyłoby rozdzielcość odczytu z czujnika odległości nie zaburzając pracy czujników podłożą.

Rozwiązaniem jeszcze dokładniejszym byłoby zaprojektowanie na płytce dzielnika napięcia. Wtedy mikrokontroler otrzymywałby sygnał cyfrowy o wykryciu przeciwnika lub linii. Takie rozwiązanie nie tylko przyspieszyłoby reakcje robota, skracając czas konwersji analogowo-cyfrowej, ale umożliwiłoby również wykorzystanie przerwań mikrokontrolera.

Przeoczeniem i archaizmem pozostałym po wcześniejszej koncepcji robota jest wspominany wcześniej układ zasilania części logicznej. W związku z wykorzystaniem gotowego modułu klonu Arduino Nano, jest on w zasadzie zbędny w konstrukcji. Tym niemniej w żaden sposób nie wpływa to negatywnie na pracę robota.

Niestety, z uwagi na brak dostępu do innego robota klasy minisumo, nie udało się przeprowadzić pojedynek sparingowych, które mogłyby dokładniej zweryfikować skuteczność systemu podnoszenia przeciwnika w warunkach rzeczywistych.

5.2. Potencjalne usprawnienia lub rozbudowy

Przygotowano trzecią wersję płytki, z niektórymi usprawnieniami opisanymi w poprzednim podrozdziale. Koszt przygotowania płytka jest jednak zbyt duży i zdecydowano wstrzymać się z zakupem, aby wprowadzić ewentualną dodatkową funkcjonalność albo rozwiązać ewentualny niezauważony dotąd błąd.

Taką dodatkową funkcjonalnością mogłyby być użycie akcelerometru. Podczas działania algorytmu bez przeszukiwania planszy, kiedy to robot czeka na przeciwnika, informacja o tym, że robot nabiera prędkości jednoznacznie wskazywałaby na to, że przeciwnik przesuwa robota, ale jest poza zasięgiem czujników (z tyłu lub boku).

Zagadnieniem, któremu nie poświęcono wystarczająco uwagi w niniejszej pracy są koła robota. Wykorzystano gotowe koła od producenta silników. To wygodne rozwiązanie pozwoliło skupić się na innych zagadnieniach pracy. Mimo poprawnego działania doświadczenia pokazują, że własnoręcznie wykonane koła często mają znacznie lepszą

⁵ AREF to pin w mikrokontrolerach Arduino, który pozwala na ustawienie napięcia odniesienia dla przetwornika analogowo-cyfrowego (ADC). Używając AREF, można uzyskać dokładniejsze pomiary napięć w zakresie od 0 do napięcia na pinie AREF.

5. Podsumowanie

przyczepność. Koła są wymienialne, a ich ulepszanie będzie przedmiotem dalszej rozbudowy robota.

Optymalizacji można poddać rozłożenie masy dodatkowego obciążenia. Wymiana jednego dodatkowego obciążnika na kilka mniejszych powinna nieco poprawić (obniżyć) położenie środka ciężkości. Jeszcze lepszym rozważanym rozwiązaniem jest wymiana płyta na metalowy, tak aby wyeliminować potrzebę dodatkowego obciążania robota.

Minimalizacja liczby bezpośrednich lutów do płytka oraz unikanie trwałego montażu elementów sprawia, że konstrukcja jest rozkładalna i można łatwo wymienić każdą część. Takie rozwiązanie pozwala na rozbudowę i wymianę części robota w przypadku zaobserwowania nieprawidłowości któregoś z elementów, ale również na ciągłe ulepszanie minimalnym kosztem. Ponadto umożliwia szybką wymianę uszkodzonej części np. podczas zawodów.

5.3. Wnioski

Realizację głównego celu pracy, którym było wykonanie robota minisumo z systemem podnoszenia przeciwnika, zgodnego z regulaminami zawodów, uznaje się za udaną. Niniejszy robot potrafi namierzać przeciwnika korzystając z zbliżeniowych czujników optycznych. Czujniki podają poprawnie interpretując linie i przy niewielkich prędkościach robot potrafi wyhamować, dzięki czemu sam nie opuści ringu. Konstrukcja mechaniczna działa jak przewidywano. Robot startuje z ramionami w górze, aby po starcie, zgodnie z jednym z algorytmów sterowania, rozłożyć ramiona w trakcie zajmowania dogodnej pozycji.

Robot jest w stanie podnosić obiekty o masie robota minisumo w starciu frontalnym, co powinno zapewniać mu przewagę nad przeciwnikiem. W celu znalezienia się w tak faworyzującej sytuacji, niniejsza konstrukcja jest wyposażona w przełączniki umożliwiające operatorowi wybór algorytmu startowego zależnie od sytuacji startowej na ringu.

Ocena skuteczności robota będzie przedmiotem późniejszych obserwacji w trakcie zawodów, jednak zastosowana architektura systemu oraz sposób montażu pozwala na łatwe rozbudowy i wymianę dowolnego z elementów.

Bibliografia

- [1] RoboGames, *Unified Sumo Robot Rules*, Dostęp zdalny (9.11.2024): <https://robogames.net/rules/all-sumo.php>, 2004.
- [2] Robothon, *Mini sumo - Official Contest Rules*, Dostęp zdalny (9.11.2024): <https://robothon.org/rules-sumo/>, 2022.
- [3] XChallenge, *Regulamin kategorii sumo*, Dostęp zdalny (11.11.2024): https://xchallenge.pl/regulations/ROBOmotion_2024_Sumo_EN.pdf, 2024.
- [4] S. Challenge, *Regulamin Sumo*, Dostęp zdalny (5.01.2024): https://skaner.p.lodz.pl/sumochallenge/files/pl/sumo.pdf?utm_source=chatgpt.com, 2023.
- [5] M. Drwięga, *Kołowy robot mobilny minisumo z czujnikami Sharp*, Dostęp zdalny (10.11.2024): <https://mdrwiega.com/robot-minisumo-banshee/>, 2015.
- [6] F. Dede, *How to Make Mini Sumo Robot with XMotion*, Dostęp zdalny (10.11.2024): <https://blog.jsumo.com/how-to-make-mini-sumo-robot-with-xmotion/>, 2018.
- [7] D. A. Heatherington, *Huggy Bear mini sumo*, Dostęp zdalny (8.12.2024): <https://robots.wa4dsy.net/huggy-bear-mini-sumo>, 2016.
- [8] A. H. R. Club, *2016 Robot Rally*, Dostęp zdalny (8.12.2024): <https://botlanta.org/AHRC/archives/rally-2016>, 2016.
- [9] R. LLC, *Building a Mini-Sumo Robot*, Dostęp zdalny (11.11.2024): <http://www.ridgesoft.com/articles/minisumo/building%20and%20programming%20a%20mini-sumo.pdf>, 2018.
- [10] SHARP, *GP2Y0A21YK0F-Distance Measuring Sensor Unit*, Dostęp zdalny (13.12.2024): https://global.sharp/products/device/lineup/data/pdf/datasheet/gp2y0a21yk_e.pdf, 2006.
- [11] Kingbright, *Optoelectronic components - TECHNICAL NOTES*, Dostęp zdalny (25.11.2024): https://www.tme.eu/Document/49118d934bd7dd8719a54163c0a31782/KINGBRIGHT_TECH_NOTES.pdf, 2016-2018.
- [12] J. Malewicz, "Ja mam robot!", *Komputer Świat. Ekspert.*, nr. 5, s. 12–15, 2007.
- [13] STMicroelectronics, *Precision 500 mA regulators*, Dostęp zdalny (3.12.2024): <https://www.tme.eu/Document/4c19afbd0cbf5bc0f90e84076ef5e976/78Mxx.pdf>, 2012.
- [14] STMicroelectronics, *L6225 - DMOS DUAL FULL BRIDGE DRIVER*, Dostęp zdalny (4.12.2024): <https://www.tme.eu/Document/36e4a95b12d9825f6867d8275e8e9c9c/L6225D.pdf>, 2003.
- [15] Pololu, *Micro Metal Gearmotors*, Dostęp zdalny (20.11.2024): <https://www.pololu.com/file/0J1487/pololu-micro-metal-gearmotors-rev-6-1.pdf>, 2024.
- [16] IRF, *IRLL014NPbf - Power MOSFET*, Dostęp zdalny (5.01.2024): <https://www.tme.eu/Document/e17313f9c5f01a58164fb9480e11a8c7/irll014npbf.pdf>, 2004.

5. Bibliografia

- [17] IPC, *Generic Standard on Printed Board Design*, Dostęp zdalny (6.12.2024): [https://www-eng.lbl.gov/~shuman/NEXT/CURRENT_DESIGN/TP/MATERIALS/IPC-2221A\(L\).pdf](https://www-eng.lbl.gov/~shuman/NEXT/CURRENT_DESIGN/TP/MATERIALS/IPC-2221A(L).pdf), 1998.

Spis rysunków

1.1	Dohyo z oznaczonymi wielkościami charakterystycznymi	10
1.2	Komercyjnie sprzedawany moduł startowy firmy JSumo	10
1.3	Szkic przedstawiający koncepcję rozkładania się robota	11
1.4	Szkic przedstawiający koncepcję podnoszenia przeciwnika ramionami	11
1.5	Projekt robota, opisanego na forum internetowym [5]	12
1.6	Komercyjnie sprzedawany robot firmy JSumo [6]	12
1.7	Dwie koncepcje ustawiania czujników zbliżeniowych	13
1.8	Robot "Huggy bear", jedyny znaleziony robot realizujący koncepcję walki z ramionami do podnoszenia przeciwnika [7]	14
1.9	Maszyna stanów zaproponowana przez jednego z konstruktorów minisumo [9]	15
2.1	Projekt robota minisumo wyposażonego w ramiona z oznaczonymi elementami konstrukcji (opis w tekście)	16
2.2	Schemat blokowy układu sterowania robota minisumo wyposażonego w ramiona	16
2.3	Graf stanów opisywanego robota minisumo	17
2.4	Pług robota	18
2.5	Ramię robota	19
2.6	Rozłożone ramię	19
2.7	Mocowanie baterii i silników wraz z poglądowymi modelami baterii (czerwony) i płytki (pomarańczowy)	20
2.8	Rozłożone mocowanie baterii i silników	20
2.9	Koła Solarbotics RW2i	21
2.10	Projekt robota w ułożeniu startowym	21
2.11	Projekt robota po rozłożeniu ramion	21
2.12	Silnik Pololu HP	22
2.13	Czujnik odległości Pololu GP2Y0A21YK0F	23
2.14	Przykładowa charakterystyka z dokumentacji czujnika odległości Pololu GP2Y0A21YK0F [10]	24
2.15	Transoptor KTIR0711S	25
2.16	Projekt płytki PCB 2.4 z zaznaczonymi na czerwono położeniami transoptorów	25
2.17	Płytnica Seeeduino Nano	26
2.18	Schemat płytki PCB do sterowania robotem z ramionami	27
2.19	Przekrój boczny płytki PCB prezentujący trzy sposoby montażu układów podatnych na przegrzanie. [14]	28
2.20	Charakterystyka układu IRLL014N z zaznaczonym odczytem [16]	29
3.1	Wybrane elementy robota przygotowane do druku w programie Ultimaker Cura	31
3.2	Dwie wersje zamówionej płytki. Po lewej płytka z błędami, a po prawej druga poprawiona i usprawniona wersja	32

3.3	Bok robota z oznaczonymi elementami	33
3.4	Górna część robota z oznaczonymi elementami	33
3.5	Tył robota z oznaczonymi elementami	34
3.6	Dolna część robota	34
3.7	Złącza robota na płytce PCB	35
3.8	Wykonany robot w pozycji startowej	36
3.9	Wykonany robot z rozłożonymi ramionami	36
4.1	Struktura projektu oprogramowania robota minisumo	37
4.2	Schemat blokowy oprogramowania robota minisumo	38
4.3	Fotografia stanowiska do kalibracji czujników	40
4.4	Fotografia stanowiska do testu nośności	41
4.5	Trzy klatki z nagrania opisywanego eksperymentu	42

Spis tabel

1.1	Specyfikacja wymiarów i wag dla różnych klas robotów sumo ze strony robogames.net	9
1.2	Wymiary dohyo dla różnych kategorii robotów na zawodach "Robothon" w USA [2]	10
2.1	Charakterystyka silników Pololu z różnymi przekładniami przy maksymalnej efektywności	23
4.1	Odczyty z czujników Sharp w różnych odległościach	40
4.2	Odczyty z czujnika wykrywania linii	41

Spis załączników

Do pracy dołączono poniżej wymienione pliki źródłowe z oprogramowaniem sterującym robota.

- **src:**
 - Sensor.cpp
 - Robot.cpp
 - main.cpp
 - InputManager.cpp
 - FrontMotor.cpp
 - BuildLed.cpp

- **include:**

- Sensor.h
- Robot.h
- InputManager.h
- FrontMotor.h
- BuildLed.h

- **lib:**

- SoftPWM.h