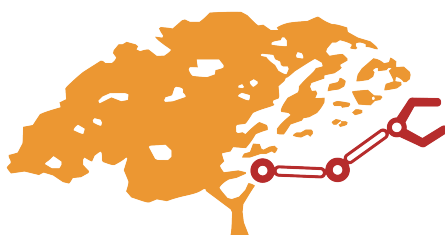

Dokumentacja

Robot mobilny klasy minisumo
„Sneak100”



KoPiAR

KOŁO NAUKOWE ROBOTYKÓW

Eryk Możdżeń

2 Września 2022

Spis treści

1	Wstęp	2
2	Założenia projektowe	2
3	Mechanika	2
3.1	Iteracje projektu	2
3.2	Projekt mechaniki	3
3.3	Dobór przekładni silników oraz baterii	4
3.4	Wydruki 3D	5
3.5	Pług	7
3.6	Koła	7
3.7	Obciążniki	8
4	Elektronika	9
4.1	Płytką główną	9
4.2	Czujniki linii	12
4.3	Enkodery	12
4.4	Czujniki wykrycia przeciwnika	13
5	Program	14
5.1	Budowa programu	14
5.2	Algorytm walki	15
6	Zdjęcia	16
7	Podsumowanie	17
8	Odnosnik do repozytorium GIT	17

1 Wstęp

W niniejszym dokumencie zostały zawarte informacje na temat budowy oraz działania robota „Sneak100”. Wyodrębnienie tego projektu spośród innych nastąpiło po powrocie z zawodów ”XII Robotic Arena” w styczniu 2020 roku, na których skryształizowała się wizja 4-kołowego, niskiego robota minisumo.

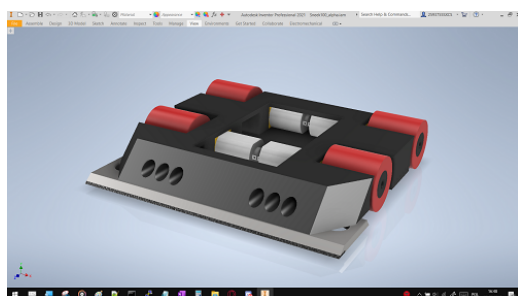
2 Założenia projektowe

- Działanie w pełni autonomiczne
- Użycie mikrokontrolera z rodziny STM32F4
- Zastosowanie 4 odrębnie napędzanych kół
- Wszystkie koła sterowane w zamkniętej pętli z użyciem enkoderów inkrementalnych
- Użycie więcej niż 2 odbiciowych czujników linii
- Zastosowanie w roli czujników wykrycia przeciwnika par: dioda nadawcza + fototranzystor
- Użycie maksymalnie dużej ilości czujników wykrycia przeciwnika
- Zastosowanie odbiornika podczerwieni do emulowania modułu startowego
- Konstrukcja możliwie niska
- Konstrukcja o obrysie mieszczącym się w polu 100x100mm
- Waga nieprzekraczająca 500g

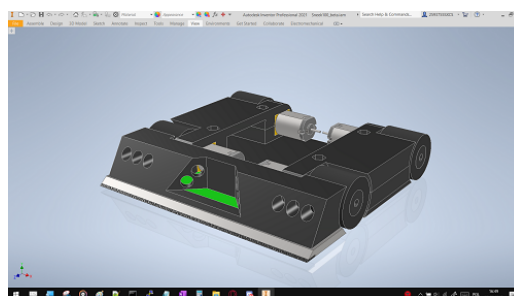
3 Mechanika

3.1 Iteracje projektu

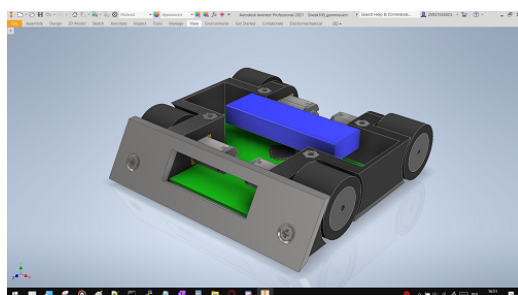
Początkowo projekt zakładał zastosowanie jednej płytki PCB w okolicach podwozia konstrukcji w którą na stałe miały być wlutowane wyprowadzenia silników, enkoderów, odbiciowe czujniki linii oraz czujniki wykrycia przeciwnika, aby zminimalizować ilość przewodów (jeden element elektroniczny konstrukcji). Pomysł ten został w końcu porzucony z uwagi na brak wystarczającego miejsca na płytce.



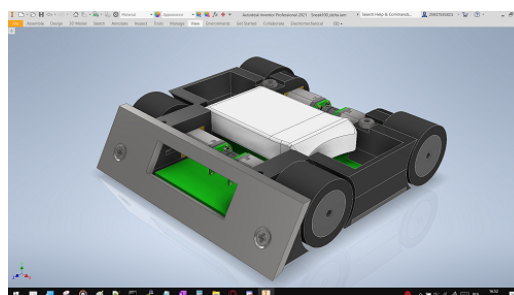
(a) wersja koncepcyjna Alpha, czerwone opony, czarny kadłub, nóż u dołu duża inspiracja robotem Flash [3]



(b) wersja koncepcyjna Beta, ruchoma głowica z pługiem i czujnikami



(c) wersja koncepcyjna Gamma, wyższa konstrukcja, pług przykręcony od przodu



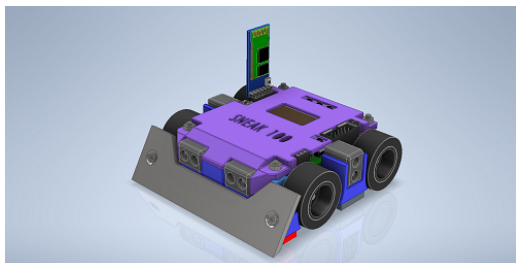
(d) wersja koncepcyjna Delta, "przymiarka" docelowej baterii

Rysunek 1: Ewolucja projektu, wersje koncepcyjne, program Autodesk Inventor 2021

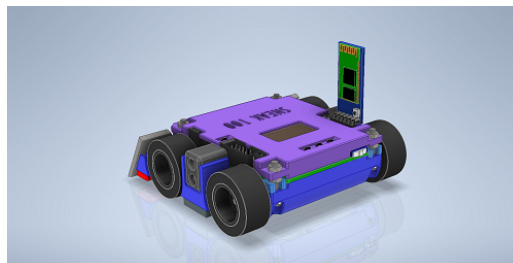
3.2 Projekt mechaniki

Ostatecznie, projekt zatracił właściwość jazdy po obu stronach, czyli po wywróceniu staje się bezbronny. W najwyższym punkcie konstrukcja osiąga niecałe 32mm ponad płaszczyznę poruszania się. Zaokrąglone ścięcie w tylnej części kadłuba zapewnia stały kontakt tylnych kół z podłożem, nawet podczas znacznego uniesienia frontu pojazdu. W roli pług został zastosowany nóż do strugarek ze stali narzędziowej z ostrzem z węgla spiekanego, w którym wykonano wycięcie na przednie czujniki wykrycia przeciwnika.

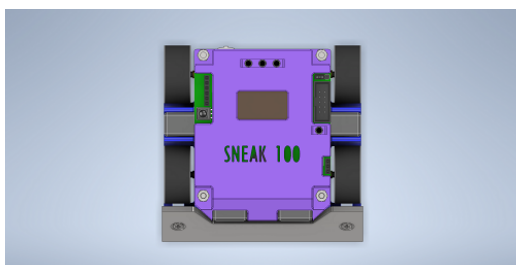
Główna płytką PCB jest umieszczona ponad baterią i silnikami, tak jak w robocie DevexDestroyerDroid [4]. Umożliwiło to wmontowanie ekranu OLED prosto w płytkę drukowaną.



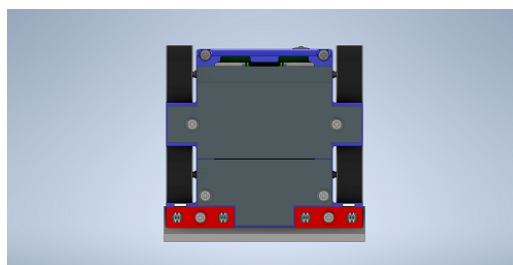
(a) widok pod skosem z przodu, obecny moduł Bluetooth HC-05



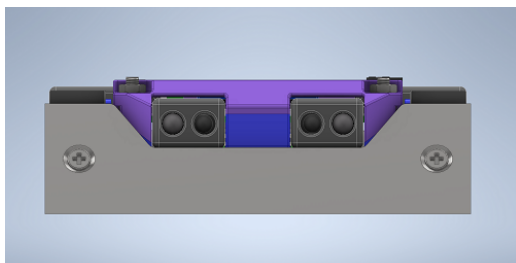
(b) widok pod skosem z tyłu, obecny moduł Bluetooth HC-05



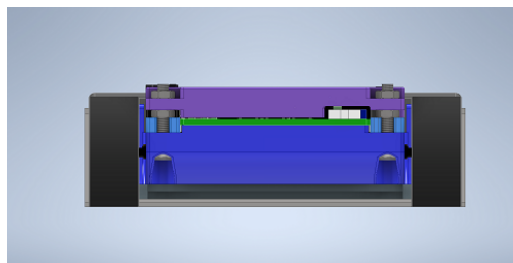
(c) widok z góry, od lewej: gniazdo modułu Bluetooth, odbiornik podczerwieni, 3 przyciski ekran OLED, gniazdo programatora



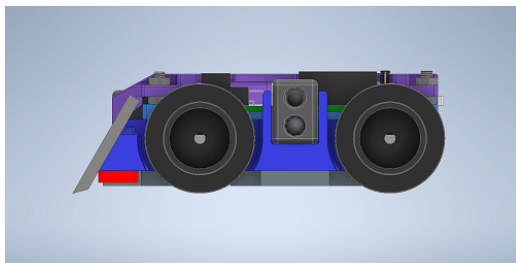
(d) widok z dołu, widoczna pokrywa baterii oraz odbiciowe czujniki linii



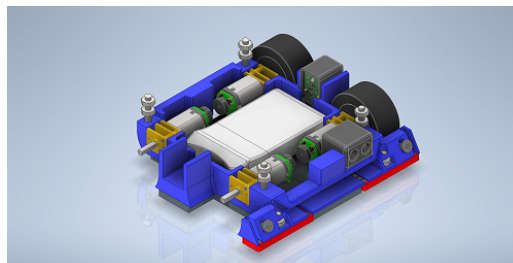
(e) widok z przodu



(f) widok z tyłu, widoczny włącznik



(g) widok z lewej



(h) widok od środka, widoczna bateria, silniki, enkodery, w przedniej części widoczne kanały przewodów czujników linii

Rysunek 2: Model 3D, program Autodesk Inventor 2021, niebieski - korpus, fioletowy - pokrywa górna, srebrny - pług, czerwony - obudowa czujników linii, szary - ołowiane obciążniki

3.3 Dobór przekładni silników oraz baterii

Jednostki napędowe (silniki wraz z przekładniami jako całość) to szeroko stosowane napędy typu Micro High Power firmy Pololu. Został wybrany wariant z karbonowymi szczotkami (HPCB), które zapewniają zwiększoną żywotność i odporność na pracę pod obciążeniem.

Do przeanalizowania dostępnych kombinacji baterii - przekładnia, zostały zamodelowane dwa czynniki: czynnik proporcjonalny do maksymalnej prędkości liniowej robota V oraz czynnik proporcjonalny do siły z jaką robot jest w stanie oddziaływać na przeciwnika M .

$$F_{push} \propto M = C \cdot R \cdot n \quad (1)$$

$$v_{max} \propto V = \frac{C}{R} \quad (2)$$

- C - ilość cel w baterii LiPO (np. dla 2S $C = 2$)
- R - przełożenie przekładni silnika (np. dla 30:1 $R = 30$)
- n - ilość kół

Idealny wybór maksymalizował by oba czynniki, jednak jest to niemożliwe. Aby zobrazować możliwe kombinacje, została stworzona tabela, w której punktem odniesienia został robot Scalak [2].

robot	Scalak (2 koła)	Sneak100 (4 koła)			
bateria	3S	2S		3S	
przekładnia	50:1	50:1	30:1	50:1	30:1
M	300 (100%)	400 (+33%)	240 (-20%)	600 (+100%)	360 (+20%)
V	0.06 (100%)	0.04 (-33%)	0.067 (+11%)	0.06 (+0%)	0.1 (+67%)

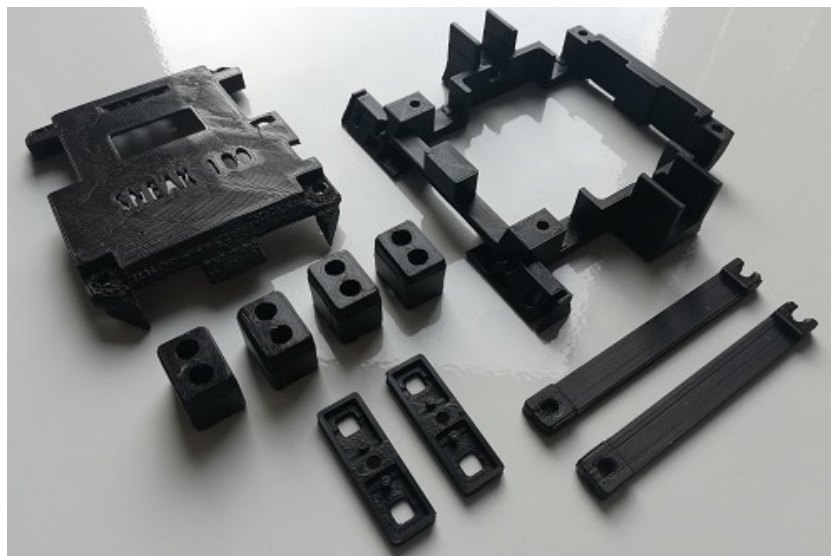
Tablica 1: Tabela doboru przekładni silników
kolorem zielonym został zaznaczony wybrany wariant

Jak widać najlepsze rezultaty osiągane są dla baterii 3S, jednak ze względu na swoje rozmiary musiała być wykluczona z dalszych rozważań.

Z uwagi na pierwotne założenia projektu (wysoka mobilność), został wybrany wariant baterii 2S wraz z przekładnią 30:1, który charakteryzował się większą prędkością liniową konstrukcji (robot Scalak czasami wydawał się zbyt powolny). Osłabienie w zwarcu powinno zostać nadrobione wymanewerowaniem przeciwnika.

3.4 Wydruki 3D

Pliki STL zostały bezpośrednio zapisane z Autodesk Inventor-a, a kod gcode został wygenerowany w programie Voxelizer 2.0 (dedykowany program producenta dostępnej drukarki). Elementy zostały wydrukowane z czarnego PLA. Kolor został wybrany tak, aby utrudniać wykrycie przez przeciwnika z czujnikami opartymi o światło podczerwone (obecnie większość robotów takich używa).



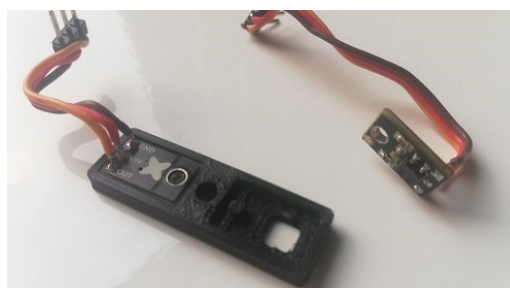
Rysunek 3: Wszystkie wydruki 3D tworzące konstrukcję robota, od lewej: obudowa górna, cztery obudowy czujników wykrycia przeciwnika, dwie obudowy czujników linii, korpus, dwie belki blokujące silniki

Mocowanie czujników linii

Czujniki odbiciowe w postaci małych modułów zostały zamontowane parami w niewielkich koszykach wy-modelowanych tak, aby zakrywały cały czujnik z wyjątkiem głowicy modułu. Całość została przykręcona na pojedynczą śrubę M3 z płaskim łbem.



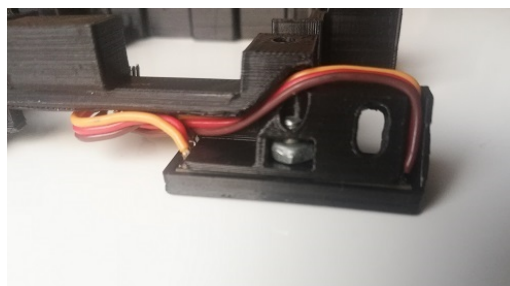
(a) widok z góry i z dołu, widoczne kwadratowe otwory na czujniki oraz okrągłe na śrubę



(b) czujnik osadzony w gnieździe, czujnik po za gniazdem



(c) widok z dołu z osadzonymi czujnikami, śruba z płaskim łbem "chowa się" wewnątrz obudowy



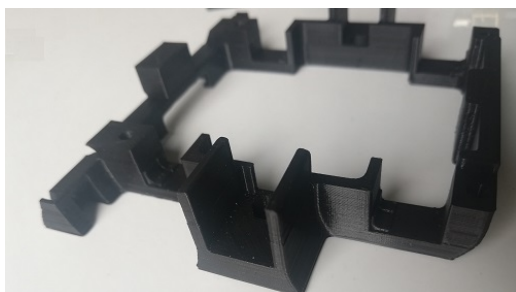
(d) widok z przodu, przewody ułożone w kanałach, śruba przykręcona do nakrętki umieszczonej w specjalnym wycięciu

Rysunek 4: Sposób montażu odbiciowych czujników linii

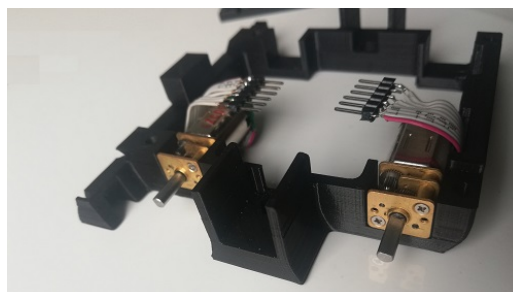
Korpus – mocowanie silników

Silniki zostały osadzone za pomocą specjalnych gniazdach imitujących fabryczne mocowania zastosowanych silników firmy Pololu. W każdym z nich zostało umieszczone odpowiednio wyprofilowane wycięcie, które umożliwia ruch przekładni (a wraz z tym silnika) do środka i na zewnątrz pojazdu, co było dużym problemem w robocie Scalak[2].

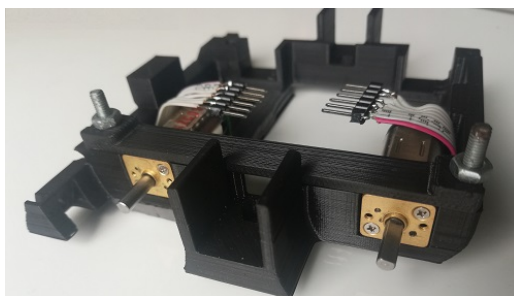
Belki blokujące zapobiegają wysuwaniu się przekładni w górę, a także stanowią podstawę pod płytkę elektroniczną, dzięki czemu powstaje przestrzeń o lepszych zdolnościach chłodzących silnik.



(a) miejsca na przekładnie silników, widoczne wycięcia zapobiegające wysuwaniu się zespołu napędowego w bok



(b) przekładnia silników w pełni osadzona w korpusie



(c) belka blokująca przykręcona na śruby dociska przekładnie do wnętrza wycięcia

Rysunek 5: Sposób osadzania silników w korpusie robota

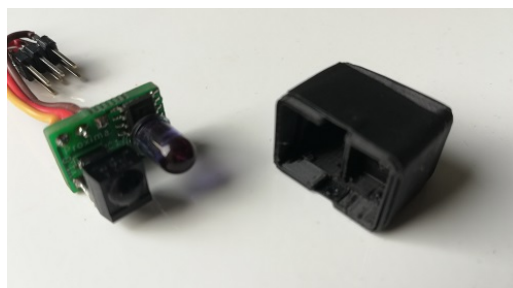
Czujniki wykrycia przeciwnika

Każdy pojedynczy czujnik wykrycia przeciwnika został zamknięty w obudowie, której głównym zadaniem było odseparowanie nadajnika (podczerwona dioda LED) oraz odbiornika w osobnych przestrzeniach przy jednoczesnym nieograniczaniu możliwości wykrywania obiektów na wprost sensora.

W praktyce przestrzeń wykrywania przypomina stożek, a umiejscowienie czujników w korpusie zapewnia częściowe pokrywanie się tych pól (estymacja położenia przeciwnika jest dzięki temu bardziej dokładna).



(a) porównanie stanu fabrycznego oraz spreparowanego

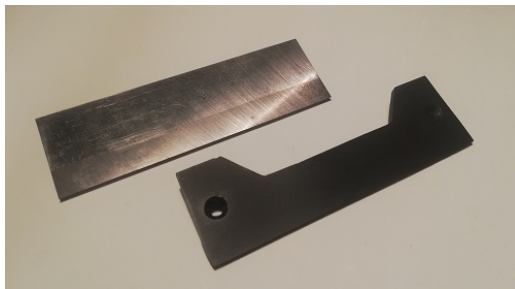


(b) widok z przodu, pług przykręcony do korpusu, śruby nie wystają po za płaszczyznę noża

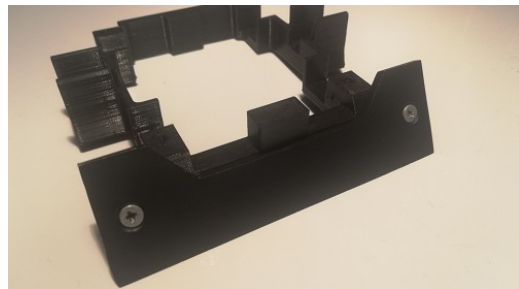
Rysunek 6: Wykonanie i montaż pług robota

3.5 Pług

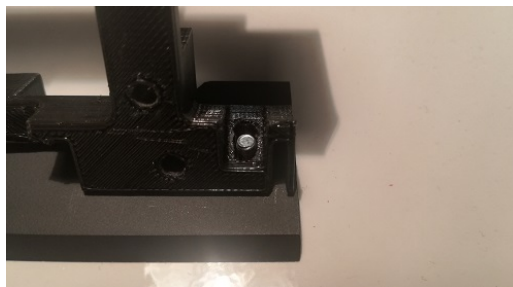
W roli pług został użyty nóż typu HM do strugarek. Ostrze zostało wykonane z węgla spiekane. Za pomocą szlifierki kątovej zostało wykonane podłużne okno dla czujników wykrycia przeciwnika. Całość została pomalowana czarną farbą w spray-u chroniącą przed korozją oraz utrudniającą wykrycie przez przeciwnika.



(a) porównanie stanu fabrycznego oraz spreparowanego



(b) widok z przodu, pług przykręcony do korpusu, śruby nie wystają po za płaszczyznę noża



(c) widok z dołu, widoczne wycięcie na nakrętkę, podłużny otwór umożliwia delikatną regulację pozycji ostrza noża względem podłoża

Rysunek 7: Wykonanie i montaż pług robota

3.6 Koła

Zostały zakupione popularne koła Solarbotics RW2i o średnicy 29 mm. Docelowo opona zostanie odlana z silikonu formierskiego, ponieważ fabryczne ogumienie nie posiada wystarczająco dobrej przyczepności do podłoża.



(a) felga oraz gumowa opona

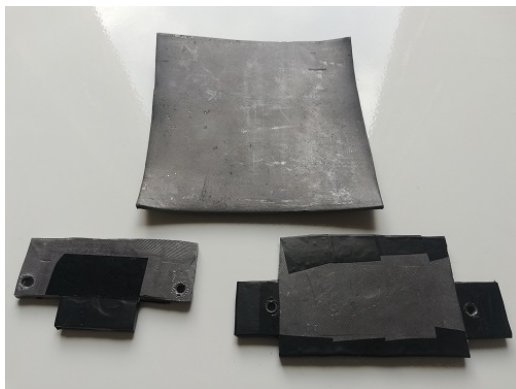


(b) korpus nie wystaje po za obrys opony co zapewnia ciągły kontakt tylnich kół z podłożem

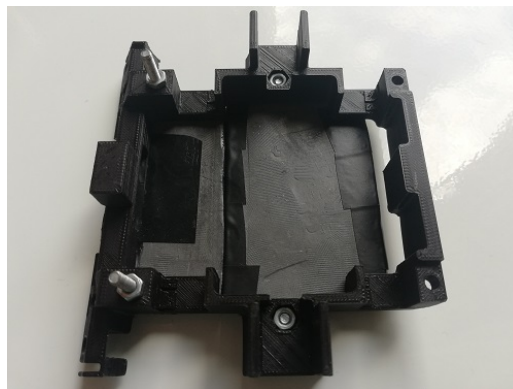
Rysunek 8: Koła robota

3.7 Obciążniki

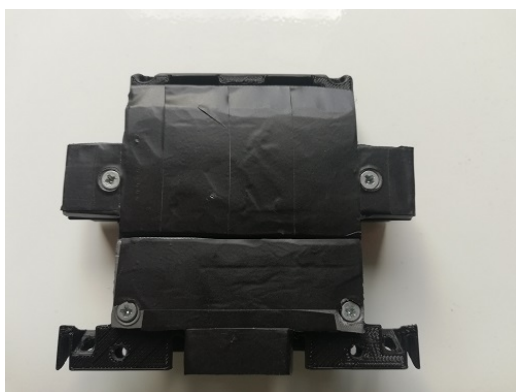
Dociążenie konstrukcji okazało się być konieczne, ponieważ konstrukcja uzbrojona we wszystkie komponenty elektroniczne ważyła lekko ponad połowę maksymalnej (docelowej) masy. W roli obciążników zastosowano płyty ołowiane. Z uwagi na komfort użytkownika, ołów został owinięty czarną taśmą izolacyjną (brak kontaktu skóry z ołowiem).



(a) płyta ołowiana o grubości 2mm,
obciążniki: mały 85g, duży 152g



(b) nakrętki umieszczone w sześciokątnych
gniazdach ułatwiających przykręcanie



(c) obciążniki pełnią rolę dna korpusu,
nisko położony środek ciężkości



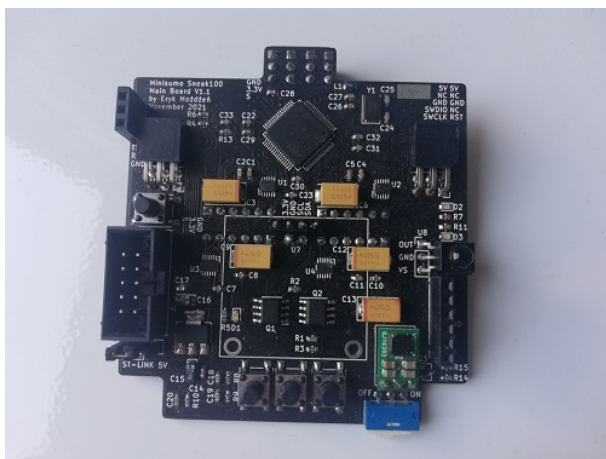
(d) każdy z obciążników jest wykonany z
dwóch warstw ołowiu o łącznej grubości 4mm

Rysunek 9: Obciążniki ołowiane

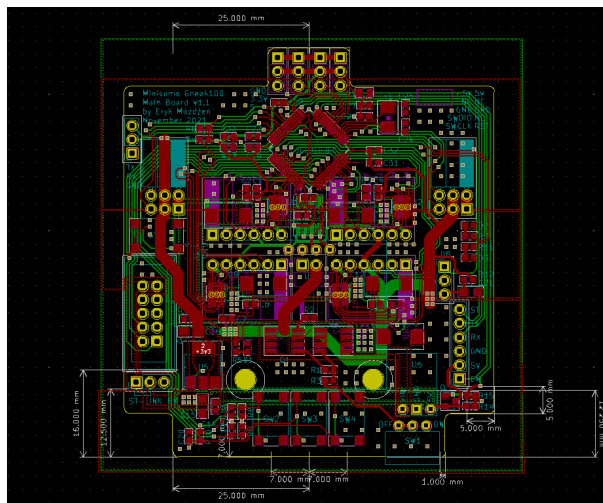
4 Elektronika

4.1 Płytką główną

Płytką wyposażoną w główne komponenty została zaprojektowana początkowo w Autodesk Inventor (tak jak cały robot), a następnie po wyznaczeniu możliwego obrysu i wymiarów mechanicznych został wykonany projekt w programie KiCad 5. Ścieżki pomieściły się na dwóch warstwach miedzi, co zapewnia niskie koszty produkcji płytki PCB. Elementy zostały umiejscowione tak, aby znajdowały się po jednej stronie co ułatwiało montaż (z jednym wyjątkiem). Czarny kolor został wybrany wyłącznie dla walorów estetycznych.



(a) widok z przodu,
ekran OLED został wylutowany

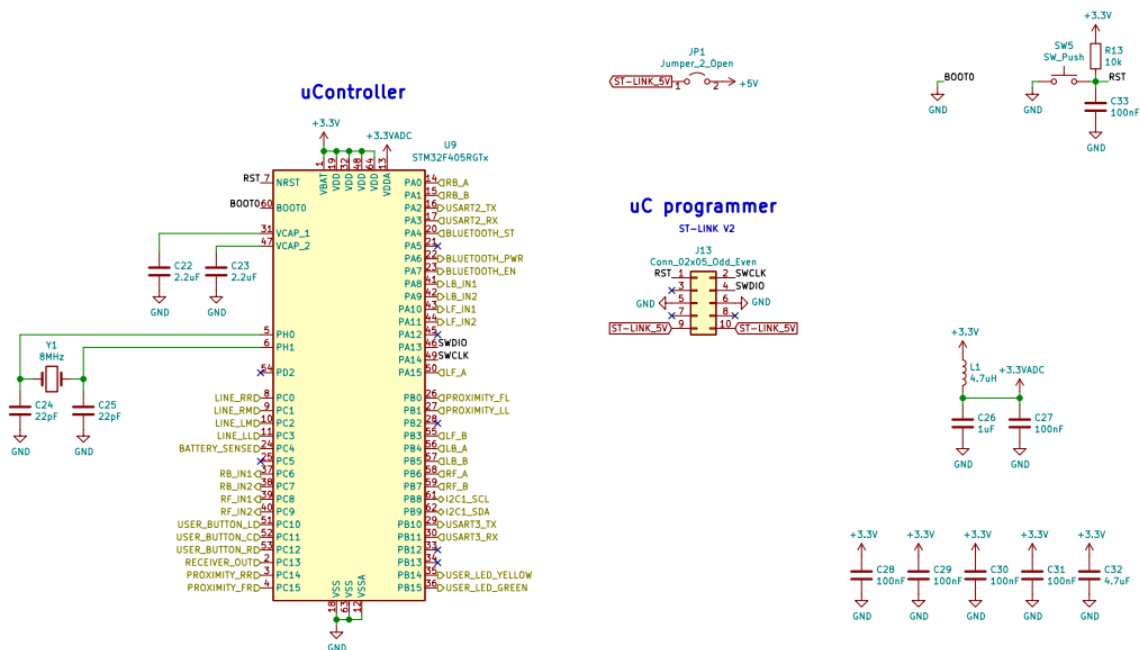


(b) projekt w programie KiCad 5

Rysunek 10: Płytką główną

Mikrokontroler

Jako mikrokontroler został wybrany STM32F405 z uwagi na wcześniejsze doświadczenia z tą jednostką oraz możliwość jego zakupu (w sierpniu 2021 roku wybór w sklepach nie był obfity). Zostało wykorzystane sprzętowe wsparcie dla czterech enkoderów inkrementalnych.

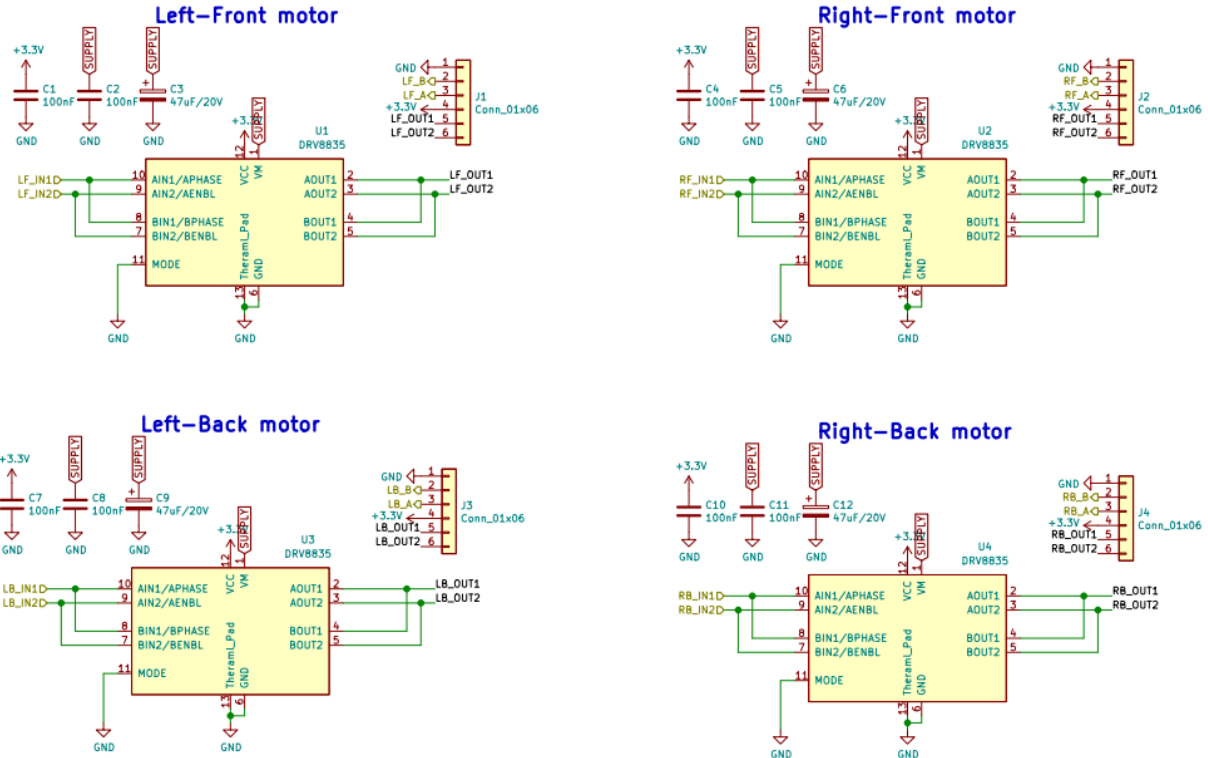


Rysunek 11: Schemat wyprowadzeń mikrokontrolera, oscylator, zworka wyboru zasilania, przycisk reset, złącze programatora

Mostki H

Do sterownia silników użyto czterech dwukanałowych sterowników DRV8835. Został polecony przez znajomych, którzy zachwalali jego niewielkie rozmiary oraz stosunkowo dobre parametry (maksymalny prąd dwóch zmostkowanych kanałów 3A). Rzeczywiście, niewielka obudowa ułatwiła tworzenie projektu, ponieważ wymagał on dwukrotnie większej liczby takich układów (zazwyczaj konstrukcje tego typu mają dwa koła).

H-Bridge control mode:
MODE = LOW -> IN/IN
MODE = HIGH -> PHASE/ENBL

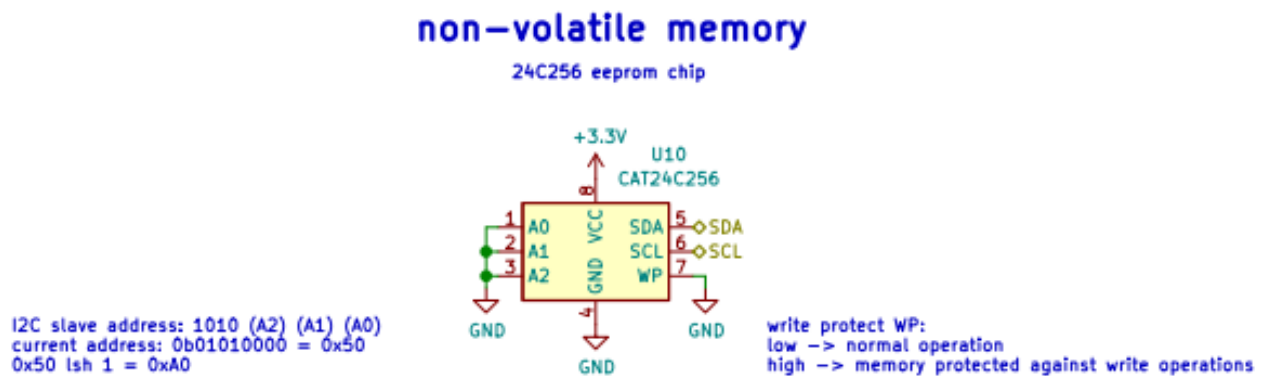


Rysunek 12: Schemat czterech układów z zmostkowanymi wyjściami, gniazda podłączeń silników

Pamięć nieulotna

Kość pamięci EEPROM ma na celu zapis stanu robota podczas walki, aby przy chwilowym braku zasilania lub wystąpieniu szpilki napięcia resetującą mikrokontroler (wywołaną nagłym uderzeniem) móc przywrócić maszynę stanów, dane o walce lub inne parametry do pamięci RAM.

Element ten został dodany do projektu jako ostatni w lekkim pośpiechu, więc został umieszczony na spodniej stronie płytki pomiędzy gniazdami podłączeń silników.

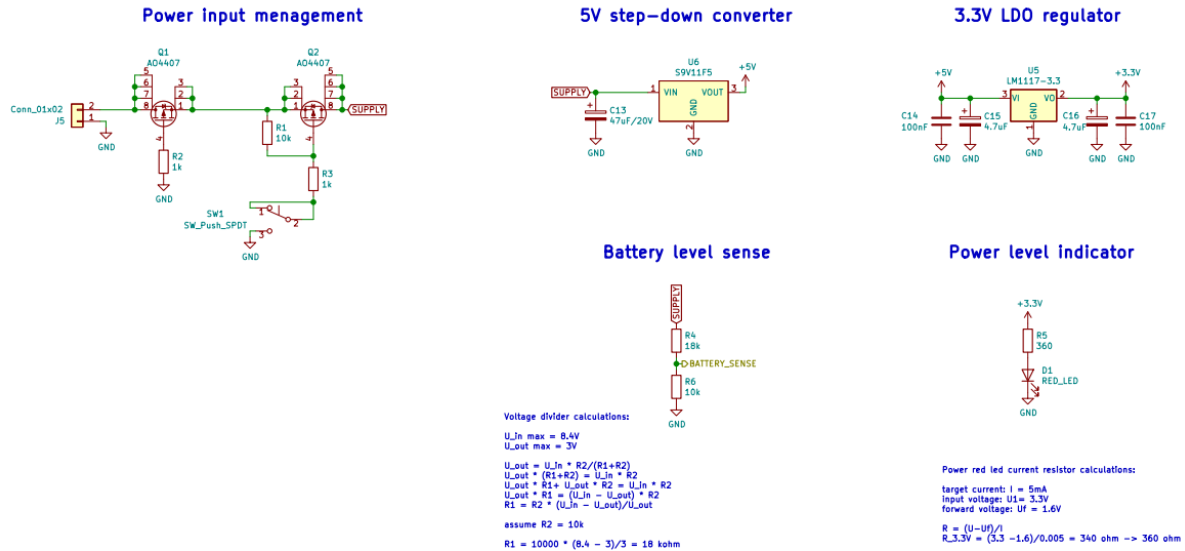


Rysunek 13: Schemat pamięci EEPROM

Zasilanie

W konstrukcji zostały wyszczególnione 3 rodzaje napięć:

- napięcie zasilania (bateryjne, max 11V) służące do zasilania mostków H i silników
- napięcie 5V (przejsciowe) do zasilania modułu bluetooth
- napięcie 3.3V (logiczne) służące do zasilania mikrokontrolera, czujników i interfejsów

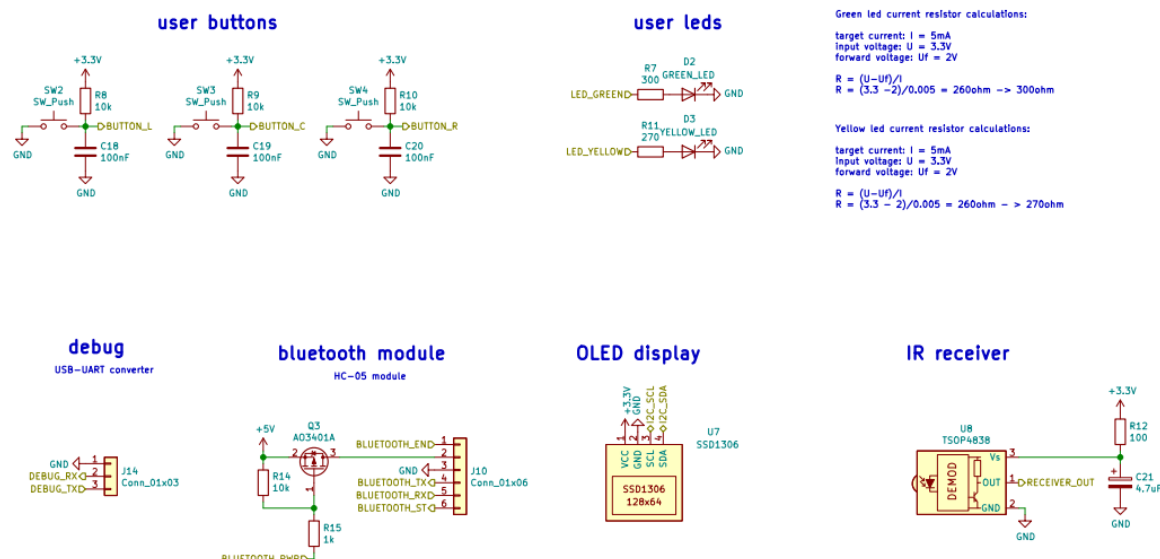


Rysunek 14: Schemat sekcji zasilania, od lewej: wejście baterii i ochrona przed odwrotną polaryzacją, włącznik główny, przetwornica 5V, stabilizator 3.3V, dzielnik do pomiaru napięcia baterii, dioda LED

Interfejs

W celu ułatwienia użytkowania, rozwoju i debugowania robota, zostały dodane:

- ekran OLED SSD1306 wraz z trzema przyciskami (interfejs graficzny)
- odbiornik podczerwieni wraz z żółtą diodą (wbudowany moduł startowy)
- moduł bluetooth HC-05 (możliwość wchodzenia w tryb AT z poziomu kodu)
- UART oraz dioda (debugowanie)



Rysunek 15: Schemat interfejsów od lewej: 3 przyciski, 2 diody, UART, moduł bluetooth, ekran, odbiornik IR

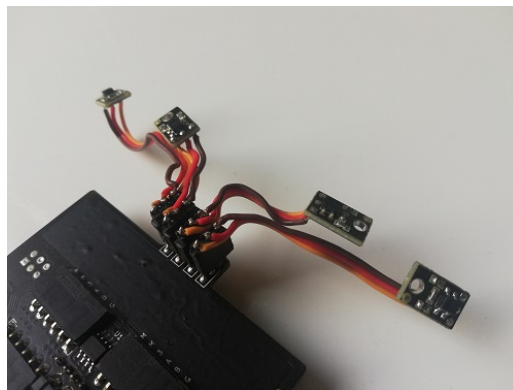
4.2 Czujniki linii

Do wykrycia białej linii na czarnym tle zostały wybrane popularne czujniki KTIR0711S w postaci klonów modułów QTR-1A oryginalnie produkowanych przez firmę Pololu. Moduły okazały się być bardzo awaryjne i konieczna była wymiana scalaków na nowe (można było to przewidzieć po dwukrotnie niższej cenie za sztukę).

W roli przewodów posłużyły trójżyłowe i giętkie kable odcięte od szeroko dostępnych 9-gramowych serwomechanizmów stosowanych w modelarstwie. Wtyk jest standardową listę pinów o rastrze 2,54 mm. Nóżki gniazda na płytce głównej zostały przed wlutowaniem uformowane tak, aby listwa żeńska nie ocierała o tarcze enkoderów.



(a) czujniki wraz kablem od serwomechanizmów



(b) podłączenie czujników do płytki głównej

Rysunek 16: Czujniki linii

4.3 Enkodery

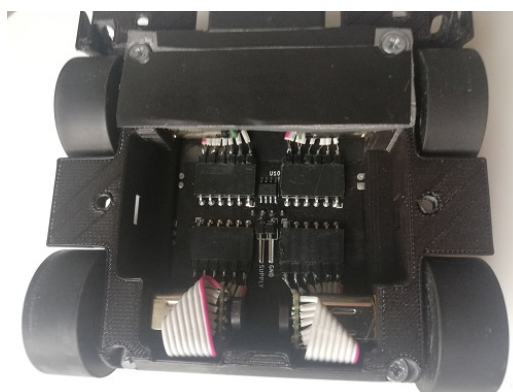
Kąt wychylenia koła można wyznaczyć poprzez zliczanie impulsów enkodera inkrementalnego. Do wybrano silnika przystosowane są enkodery także od firmy Pololu. Charakteryzują się 12 impulsami na obrót silnika, co razem z przekładnią 30:1 daje 360 impulsów na obrót koła (1°).

Niewielki raster wyprowadzeń 2mm uniemożliwia zastosowanie standardowych konektorów i listw 2.54mm, więc przylutowane zostały przewody taśmowe odzyskane ze starego komputera PC. Na linie zasilania silnika zostały użyte cztery żyły zmostkowane parami w celu minimalizacji oporu i wydzielanego ciepła.

Gniazda i konektory to standardowe listwy 2.54mm wybrane głównie z uwagi na niski profil (wyższe gniazda dawałyby zbyt mało miejsca na baterię).



(a) enkoder (zielony) wraz z tarczą (czarną) zamontowany na silniku (srebrny)



(b) podłączenie wiązek silników do płytki głównej

Rysunek 17: Enkodery

4.4 Czujniki wykrycia przeciwnika

Z braku na rynku czujników wykrywania przeszkód i obiektów o zadowalających parametrach, cenie i rozmiarze zostały opracowane i stworzone własne sensory.

Początkowo planowane było zastosowanie pary dioda IR + fototranzystor bazując na robocie typu micro-mouse „Squick”[1], jednak wstępne testy wykazały zbyt niskie możliwości wykrywania obiektów oddalonych powyżej 20cm (pożądany zasięg to 40cm).

Ostatecznie zostały zrealizowany pomysł jeszcze z roku 2019, w którym dioda nadawcza zasilana sygnałem prostokątnym o częstotliwości 48kHz wysyłała światło podczerwone, które po odbiciu od przeszkody wracał do sensora i był wykrywany przez popularny odbiornik TSOP4838. Konstrukcja ta osiągała zamierzone 40cm zasięgu oraz była łatwa i prost w budowie.

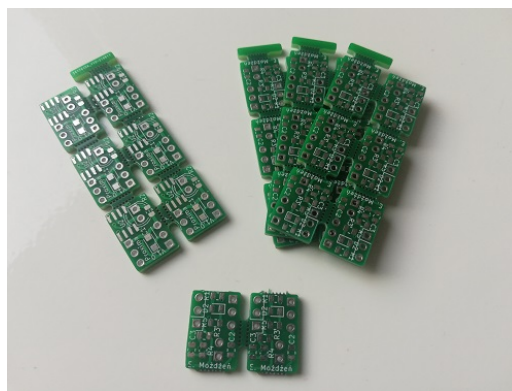
W celu implementacji do robota minisumo częstotliwość na której pracował czujnik musiała zostać zmieniona 56kHz, ponieważ 38kHz jest używane do nadawania wiadomości do modułów startowych przez sędziego. Nadawanie w tym paśmie mogłoby być powodem dyskwalifikacji robota z zawodów.

Wymiary zewnętrzne zostały zbliżone do rozmiaru czujnika JS40F. Wymusiło to niewielki rozmiar płytki PCB, która została zaprojektowana oraz ręcznie spanelizowana w programie KiCad 5.

Doświadczalne dobranie rezystora ograniczającego prąd diody nadawczej okazało się być wymagane, ponieważ zastosowany układ 555 w roli generatora przebiegu prostokątnego według dokumentacji (teoretycznie) nie powinien być zasilany napięciem 3.3V. Powodowało to nieprzewidywalną obciążalność prądową wyjścia i spadek napięcia.

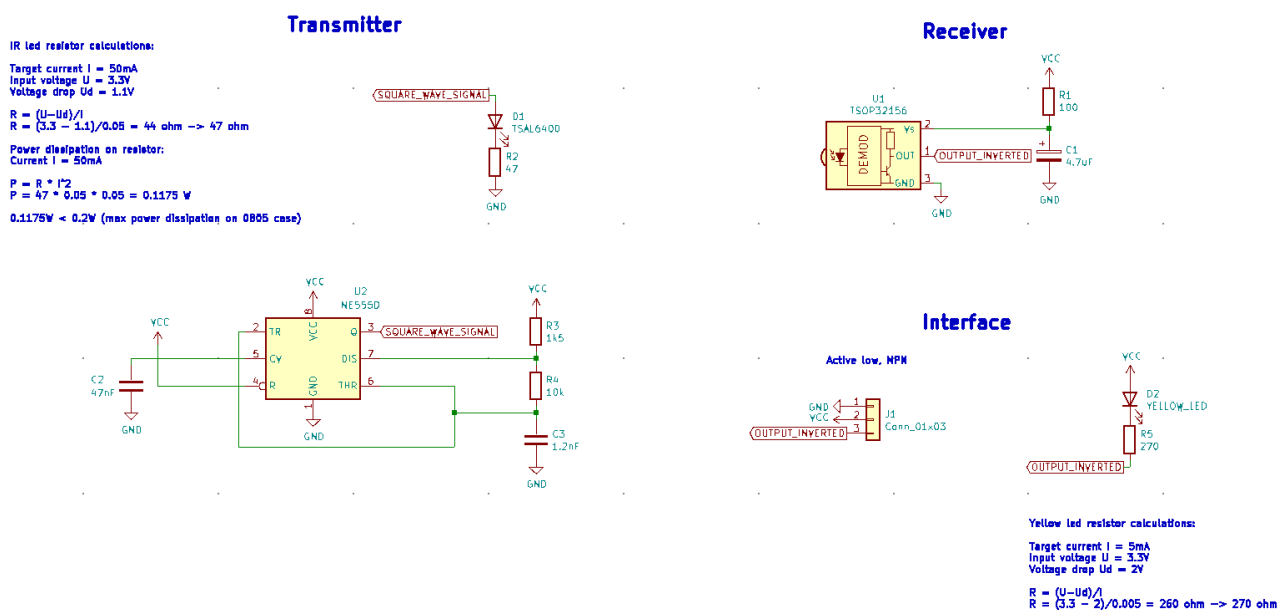


(a) gotowy czujnik, widoczny odbiornik i dioda nadawcza



(b) panelizacja płytki PCB

Rysunek 18: Czujnik wykrycia przeciwnika



Rysunek 19: Schemat płytki czujnika, od lewej: dioda nadawcza i generator przebiegu prostokątnego, odbiornik i wyjścia

5 Program

Program początkowo tworzony był w zintegrowanym środowisku STM32CubeIDE z wykorzystaniem bibliotek HAL oraz generatora kodu CubeMX. Po przejściu na śródownąSTM-kę z programowaniem na rejestrach, projekt został przepisany na wykorzystanie systemu czasu rzeczywistego FreeRTOS w edytorze VSCode.

5.1 Budowa programu

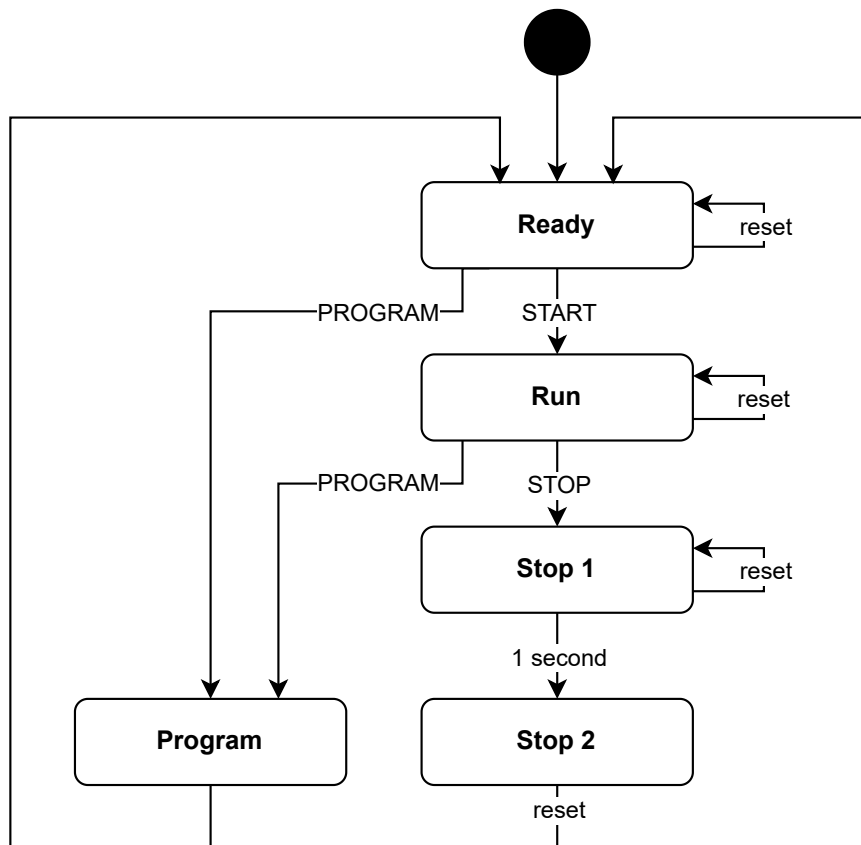
Program został oparty o pojęciu "maszyny stanów", której przejścia pomiędzy stanami są zależne od informacji docierającej do robota. Główne wyodrębnione stany:

- Ready – inicjalizacja peryferiów, oczekiwanie na sygnał
- Program – parowanie z pilotem sędziego
- Run – konstrukcja rozpoczęła algorytm walki sterowany odrębną maszyną stanów
- Stop 1 – stan zabezpieczający przed odbezpieczeniem zasilania robota w warunkach nagłego zatrzymania silników np.: szok elektromagnetyczny, zakłócenia zasilania (dotyczy głównie większych konstrukcji)
- Stop 2 – stan unieruchomienia robota, wymagane ponowne uruchomienie

Emulator modułu startowego

W celu uniknięcia potrzeby montowania modułu startowego w gotowego robota, konstrukcja została wyposażona w odpowiedni odbiornik IR oraz żółtą diodę led.

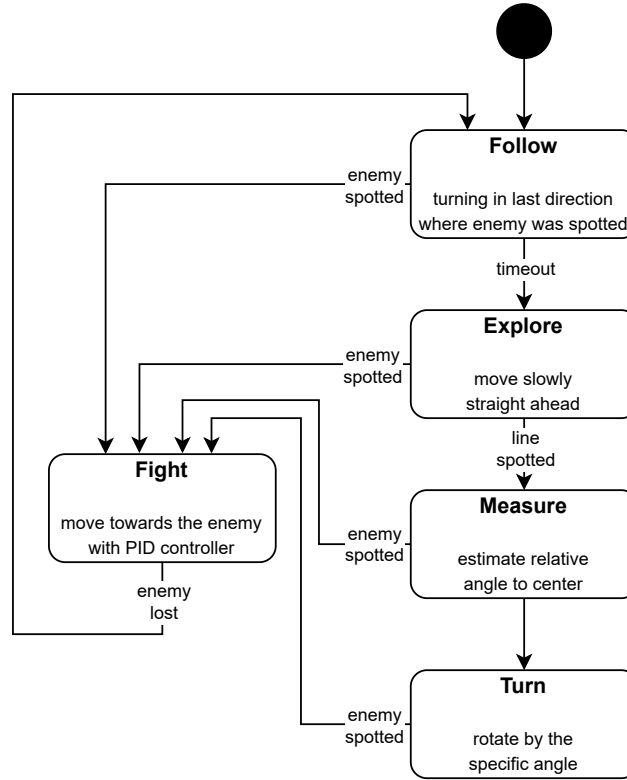
Paczki danych są zakodowane sygnałem RC5, który mikrokontroler odczytuje, co pozwala na rozszerzenie funkcji robota do bycia swoim własnym modułem startowym.



Rysunek 20: Schemat UML maszyny stanów podczas kontroli konstrukcji odbiornikiem RC5

5.2 Algorytm walki

Algorytm walki w większości polega na znalezieniu przeciwnika (szczególnie w kategoriach minisumo enchanted, gdzie średnia dyhlo osiąga 1.54m). Po nawiązaniu pierwszego kontaktu, stery przejmują kontrolę PID, który dąży do utrzymania kursu robota na wprost przeciwnika.



Rysunek 21: Schemat UML maszyny stanów podczas walki

Estymacja pozycji przeciwnika

Konstrukcja zawiera 4 czujniki wykrywający obecność przeciwnika w zakresie $\pm 90^\circ$. Znajomość tego które czujniki zgłaszają wykrycie obiektu i założenia, że na ringu znajduje się tylko jeden przeciwnik oraz dostępna przestrzeń ogranicza się do płaszczyzny ringu, pozwala na określenie przybliżonego kąta obecności przeciwnika względem kierunku frontu robota.

Odbyna się to poprzez nadanie wartości kątów poszczególnym czujnikom (w tym przypadku są to od lewej: -90° , -30° , 30° , 90°), a następnie wykonanie średniej arytmetycznej z wszystkich wartości, których czujniki stwierdziły wykrycie obiektu.

Estymacja pozycji względem krawędzi ringu

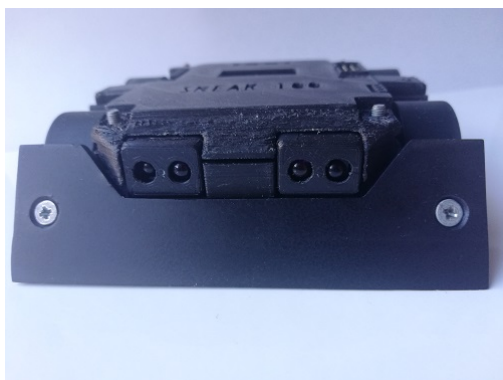
Przy założeniu, że krawędź ringu jest linią prostą (dla dużych promieni ringu generuje to mały błąd), nie występują poślizgi kół oraz że sytuacja nie jest ekstremalna (kąt jest bliski $\pm 90^\circ$), możliwe jest przybliżenie kąta pod jakim robot najeżdża na styczną do krawędzi ringu i o jaki kąt powinien skrócić, aby powrócić na środek planszy (kierunek prostopadły do stycznej krawędzi w punkcie położenia robota).

Robot posiada cztery czujniki wykrycia linii: dwa przy końcach ostrza (zewnętrzne), dwa przy centrum ostrza (wewnętrzne). Po wykryciu linii na którymś z zewnętrznych czujników odbiciowych, robot zwalnia tempo poruszania i zliczając impulsy enkoderów mierzy odległość którą przebywa do momentu wykrycia linii czujnikiem wewnętrznym.

Znając odległość pomiędzy daną parą czujników wewnętrznych i zewnętrznych $D_{cz} = const$ i przejechany dystans d kąt najeżdżania na krawędź θ_n i kąt wymaganego obrotu θ_o wynoszą odpowiednio:

$$\theta_{najezdu} = \tan^{-1} \left(\frac{d}{D_{cz}} \right) \quad \theta_{obrotu} = \pi - \theta_{najezdu}$$

6 Zdjęcia



(a) widok z przodu



(b) widok z boku



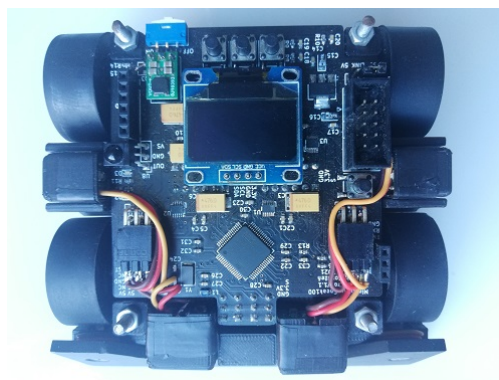
(c) widok od tyłu



(d) widok z dołu



(e) widok z góry



(f) widok z góry przy otwartej obudowie



(g) widok pod skosem z przodu



(h) widok pod skosem od tyłu

Rysunek 22: Konstrukcja złożonego robota

7 Podsumowanie

Projekt jako realizacja przemysłu na temat czterokołowego robota klasy minisumo został wykonany w większości po myśli konstruktora. To co na początku wydawało się być problemem np. solidność kadłuba wydrukowanego z PLA, okazało się być całkowicie przesadzonymi obawami.

Od początku konstrukcja miała być wykonana profesjonalnymi metodami, całkowicie unikając tz. „rzeźbienia”. Początkowy model 3D przeszedł bardzo długą drogę do moment aż wydał się możliwy do zrobienia. Nakreślił on ograniczenia mechaniczne nałożone na całą elektronikę (bateria, czujniki). Gdyby pominąć wielokrotne próby kompensowania precyzji wydruków z użytej drukarki, wszystkie elementy pasowały do siebie za pierwszym razem.

Założenie w którym płyta główna jest osobnym elementem do którego za pomocą przewodów wpinane są czujniki i efektory okazało się sprawiać więcej problemów niż pożytku. Powoduje to zmniejszanie się już i tak niewielkiej przestrzeni o przewody i gniazda, które cechują się dużą awaryjnością. Utrudnia to również szybki demontaż konstrukcji w celu nagłej naprawy.

Warto wspomnieć, że mimo braku świadomego wzorowania się na robocie „Enova”[5], zmiany wprowadzane do projektu powoli oddalały go od początkowego zamysłu właśnie w tą stronę. Wielokrotnie można było usłyszeć od starszych zawodników opinie, że jest to konstrukcja wzorowana na „Enovie”, co z uwagi na jej osiągnięcia można uznać za komplement. Faktem jest natomiast to, że wspomniany robot udowadnia, że można stworzyć wybitną konstrukcję minisumo wyposażoną w 4 koła.

Jednak mimo to odpowiedź na pytanie: „Co jest lepsze w minisumo, dwa czy cztery koła?” jest proste: dwa koła. Mimo wcześniejszej niepewności, która dała początek temu projektowi, praktyka pokazuje, że dwa koła dają zdecydowanie większą manewrowość, zmniejsza koszt o dwie jednostki napędowe oraz zwiększa ilość dostępnego miejsca na baterię i czujniki.

W konstrukcji czterokołowej pojawiają się takie problemy jak: brak lub mały docisk ostrza do płaszczyzny ringu, nierówny docisk kół do podłoża, pojawia się tarcie boczne działające na koła przy skrętach.

8 Odnosnik do repozytorium GIT

Projekt od początku właściwych prac jest dokumentowany na repozytorium GIT: <https://github.com/Eryk-Mozdzen/minisumo-sneak100>.

Bibliografia

- [1] Piotr Gorzelnik i Patryk Szydlík. *Robot mobilny typu MicroMouse "Squick"*. 2021.
- [2] Eryk Możdżeń. *Robot mobilny klasy minisumo "Scalak"*. 2021. URL: https://github.com/Eryk-Mozdzen/minisumo-scalak/blob/main/docs/raport/Dokumentacja_MS_Scalak.pdf.
- [3] Krzysztof Pochwała. *Robot mobilny klasy minisumo "Flash"*. 2015. URL: <https://forbot.pl/forum/topic/8309-minisumo-flash/>.
- [4] Devex Robotics. *Mini Sumo, from design to competing in Robot SM 2019*. URL: https://www.youtube.com/watch?v=_aM-ktyTo84.
- [5] Magnat Cyber Forge Team. *Enova*. 2014. URL: <https://forbot.pl/forum/topic/7020-enova/>.