

PROJEKT

ROBOTY MOBILNE

Raport III

Minisumo

Skład grupy:

Bartosz PIECH

Termin: s



Prowadz c :

Mgr inż. Arkadiusz



16 marca 2026

Spis treści

1	Opis projektu	2
2	Konfiguracja mikrokontrolera	3
2.1	Konfiguracja pinów	5
3	Projekt elektroniki	5
4	Konstrukcja mechaniczna	5
5	Dokumentacja	10
6	Harmonogram pracy	11
6.1	Wykres Gantta	12
6.2	Podział pracy	13
7	Zadania niezrealizowane	13
7.1	Robot zastępczy sumo	13
7.2	Wygląd robota zastępczego	13
7.3	Algorytmy sterowania	13
8	Podsumowanie	15

1 Opis projektu

Celem projektu było skonstruowanie robota mobilnego klasy Minisumo, jeżdżącego po ringu (dohyo) o zewnętrznej średnicy 77cm, który spełniał następujące kryteria :

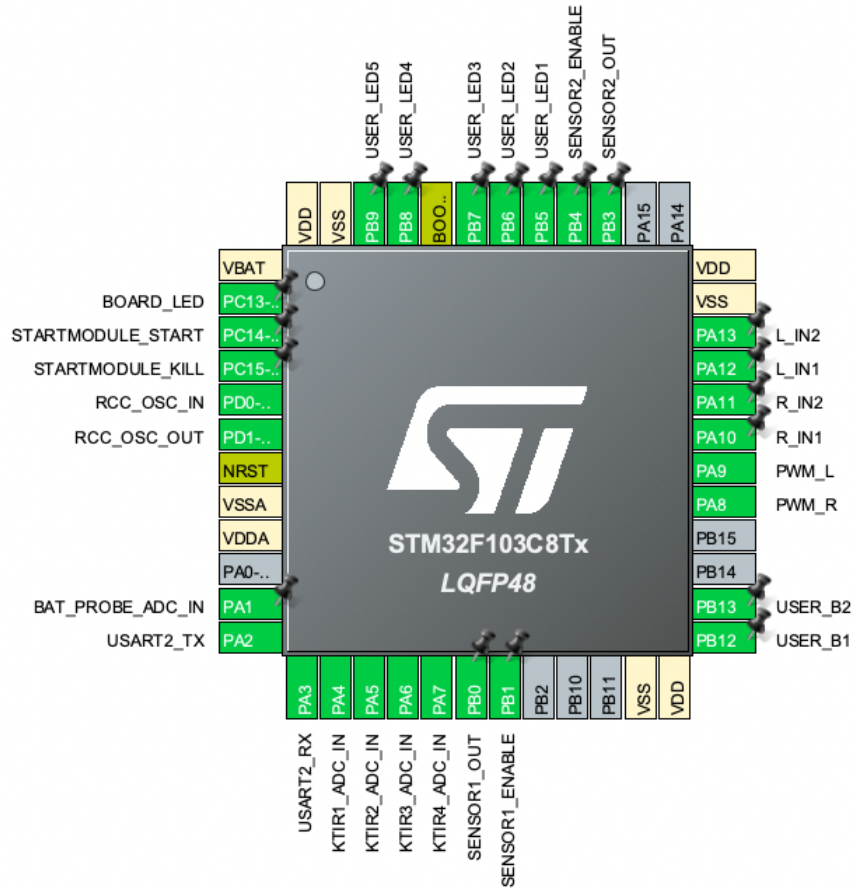
- maksymalne wymiary: 10cm x 10cm,
- maksymalna waga: 0,5kg,
- autonomiczna jazda

Podzespoły zostały wybrane na podstawie ceny, dostępności i parametrów. Robot miał być wyposażony w:

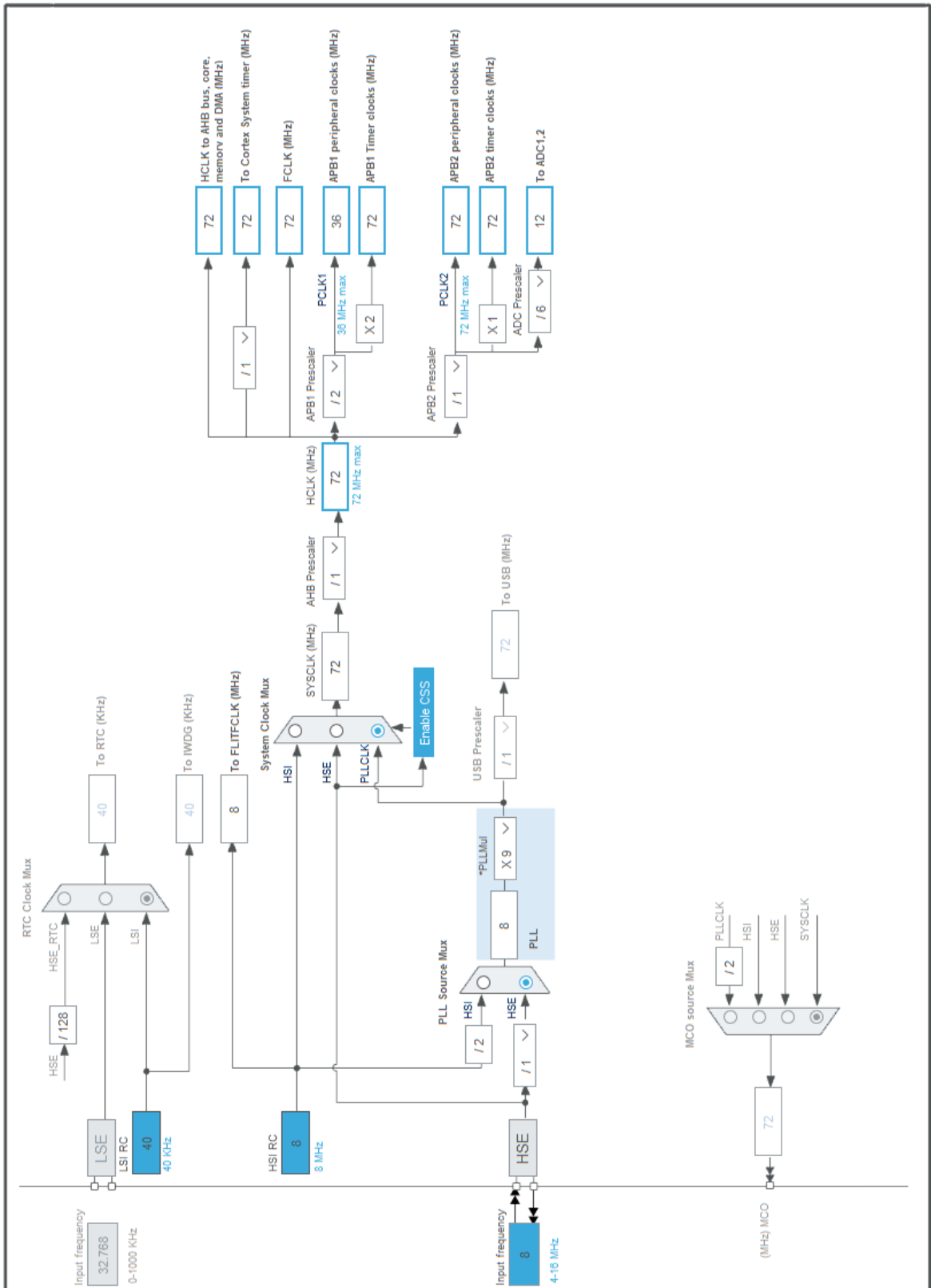
- STM32-F103 bluepill,
- 2x Silnik N20-BT05 micro 50:1 625RPM - 12V (o możliwości zasilania 3V - 12V),
- 2x Mostek H - TB6612FNG,
- 4x Czujnik białej linii KTIR0711S,
- 2x Cyfrowy czujnik odległości IR 38kHz - 60cm - Pololu 2578,
- Akumulator LiPol,
- Koła oraz opony

2 Konfiguracja mikrokontrolera

Mikrokontroler firmy STM32, model STM32F103C8T6 został skonfigurowany przy użyciu narzędzia STM32CubeMX.



Rysunek 1: Konfiguracja wyjść mikrokontrolera w programie STM32CubeMX



Rysunek 2: Konfiguracja zegarów mikrokontrolera

2.1 Konfiguracja pinów

Numer pinu	PIN	Tryb pracy	Funkcja/etykieta
2	PC13	GPIO_OUTPUT	BOARD_LED
3	PC14	GPIO_EXTI_14	STARTMODULE_START
4	PC15	GPIO_EXTI_15	STARTMODULE_KILL
5	PD0	OSC_IN*	RCC_OSC_IN
6	PD1	OSC_OUT*	RCC_OSC_OUT
11	PA1	ADC2_IN1	BAT_PROBE_ADC_IN
12	PA2	USART2_TX	USART_TX
13	PA3	USART2_RX	USART_RX
14	PA4	ADC1_IN4	KTIR1_ADC_IN
15	PA5	ADC1_IN5	KTIR2_ADC_IN
16	PA6	ADC1_IN6	KTIR3_ADC_IN
17	PA7	ADC1_IN7	KTIR4_ADC_IN
18	PB0	GPIO_Input	SENSOR1_OUT
19	PB1	GPIO_Output	SENSOR1_ENABLE
25	PB12	GPIO_EXTI_12	USER_B1
26	PB13	GPIO_EXTI_13	USER_B2
29	PA8	TIM1_CH1	PWM_R
30	PA9	TIM1_CH2	PWM_L
31	PA10	GPIO_Output	R_IN1
32	PA11	GPIO_Output	R_IN2
33	PA12	GPIO_Output	L_IN1
34	PA13	GPIO_Output	L_IN2
39	PB3	GPIO_Input	SENSOR2_OUT
40	PB4	GPIO_Output	SENSOR2_ENABLE
41	PA13	GPIO_Output	USER_LED1
42	PA13	GPIO_Output	USER_LED2
43	PA13	GPIO_Output	USER_LED3
45	PA13	GPIO_Output	USER_LED4
46	PA13	GPIO_Output	USER_LED5

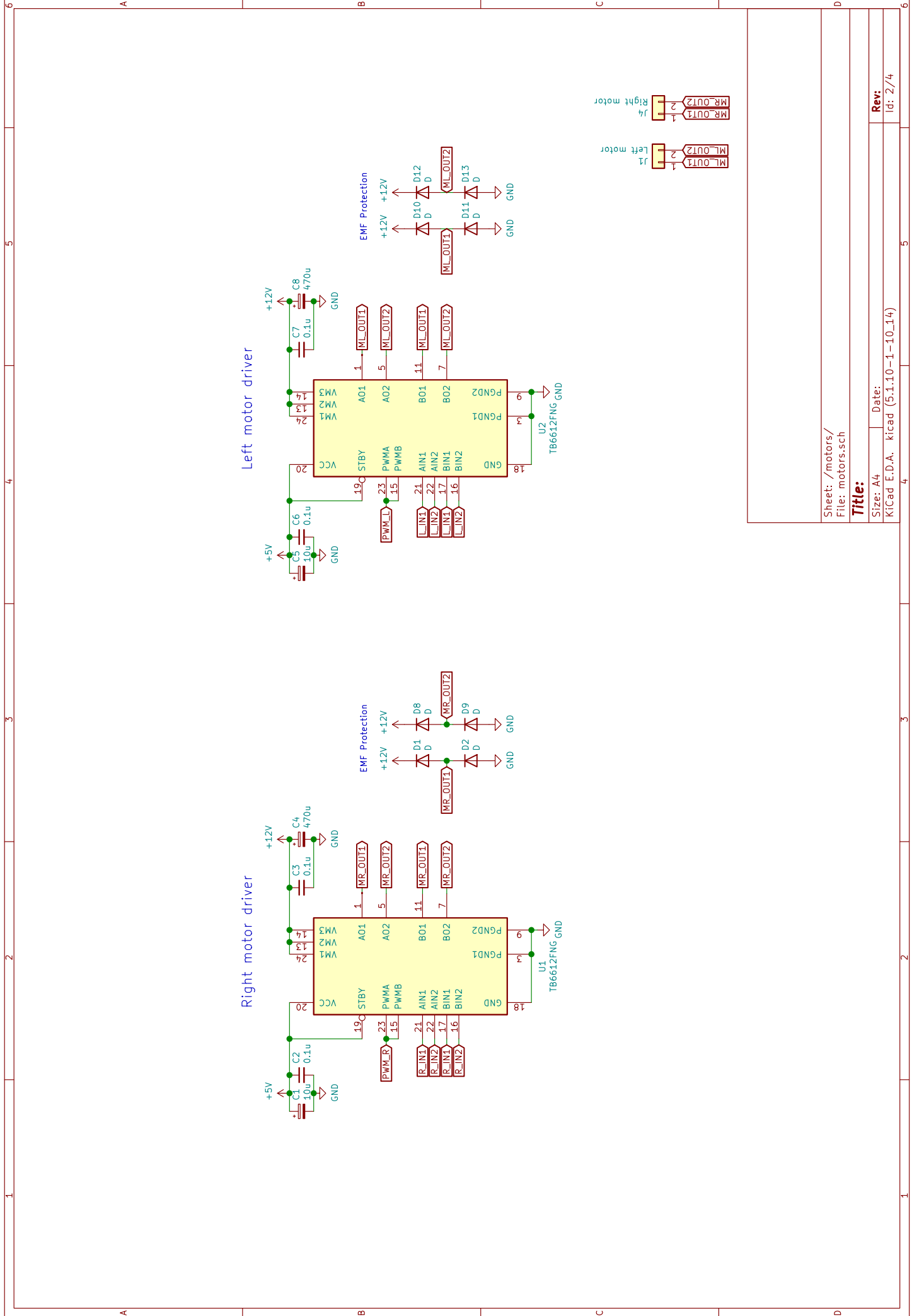
Tabela 1: Konfiguracja pinów mikrokontrolera

3 Projekt elektroniki

Projekt elektroniki został wykonany przy użyciu programu KiCad. Program został wybrany ze względu na fakt, iż jest darmowym i wolnym oprogramowaniem, ponadto ma intuicyjny interfejs użytkownika. Projekt elektroniki zajął zdecydowanie dłużej niż się spodziewano. Jest to spowodowane faktem, że należało nauczyć się interfejsu całego programu od początku, oraz wiedza na temat projektowanych układów elektroniki nie była na wystarczającym poziomie. Po kilku rewizjach projektu elektroniki udało się uzyskać zamierzony efekt i wyeliminować dużą ilość błędów. Projekt z KiCad'a oraz wizualizacja 3D są pokazane na następnej stronie.

4 Konstrukcja mechaniczna

Podstawę konstrukcji robota minisumo będzie stanowiła płytka drukowana, na której będą też osadzone główne elementy elektroniczne oraz czujniki. Obudowa robota zostanie wykonana z metalu lub w technologii 3D, gdy poprzednia opcja zawiedzie. Pług będzie wykonany z mocnego i wytrzymałego tworzywa, najprawdopodobniej stali lub węgla spiekane. Projekt mechaniki został opracowany w oparciu o dostępne w domu materiały oraz narzędzia. Planuję do testów zamontować koła znalezione w domu wymontowane z samochodu RC. Wynikowo planuję wykonać opony z silikonu w celu zwiększenia tarcia pomiędzy kołami i podłożem.

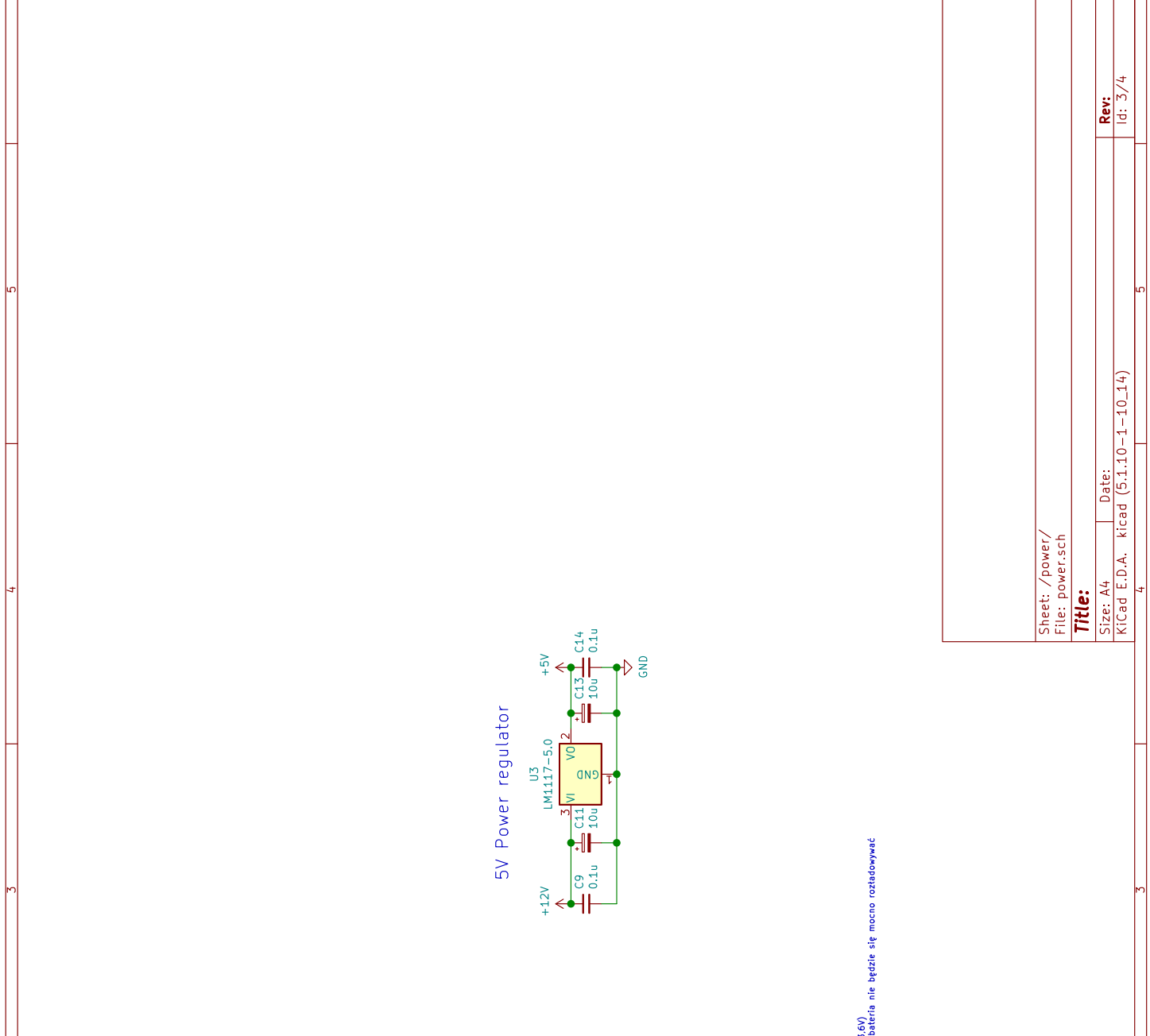


Sheet: /motors/
File: motors.sch

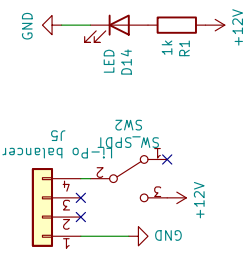
Title:

Size: A4 | Date:
KICad E.D.A. kicad (5.1.10-1-10.14)

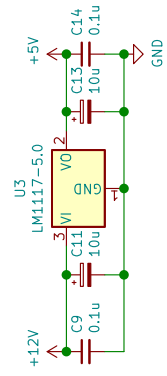
Rev:
Id: 2/4



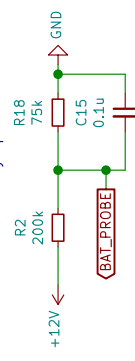
Li-Po balancer connector



5V Power regulator



Battery probe

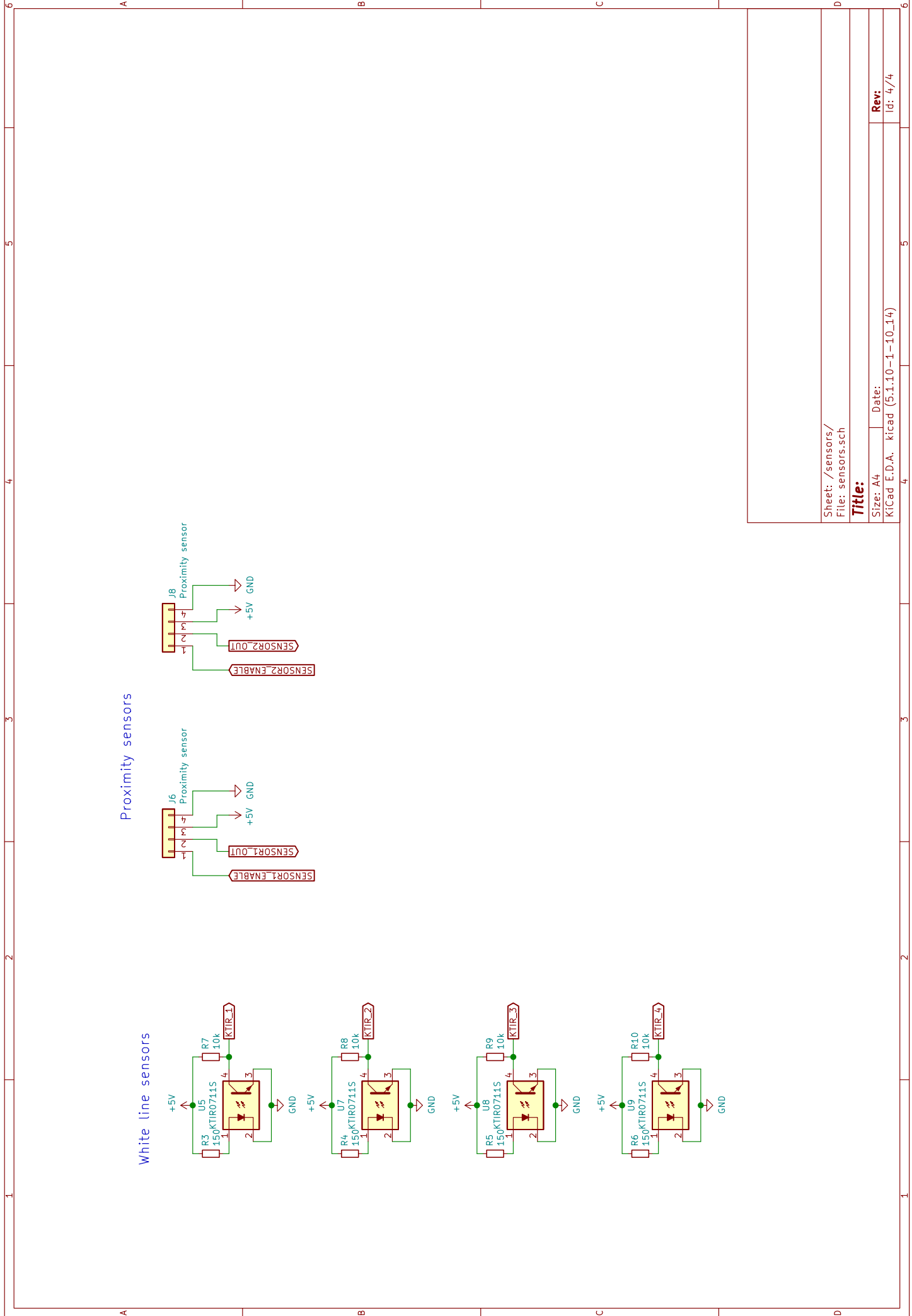


$V_{max} = 12.6 \text{ V}$
 $R_1 = 75 \text{ Ohm}$
 $R_2 = 200 \text{ Ohm}$
 $V_{out} = 3.436 \text{ V} \rightarrow$ mniej niż zakres ADC (3.6V)
 $I = 12.6 / 275k = 45\mu\text{A} \rightarrow$ małyte czyli bateria nie będzie się mocno rozładowywać

Sheet: /power/
File: power.sch

Title:

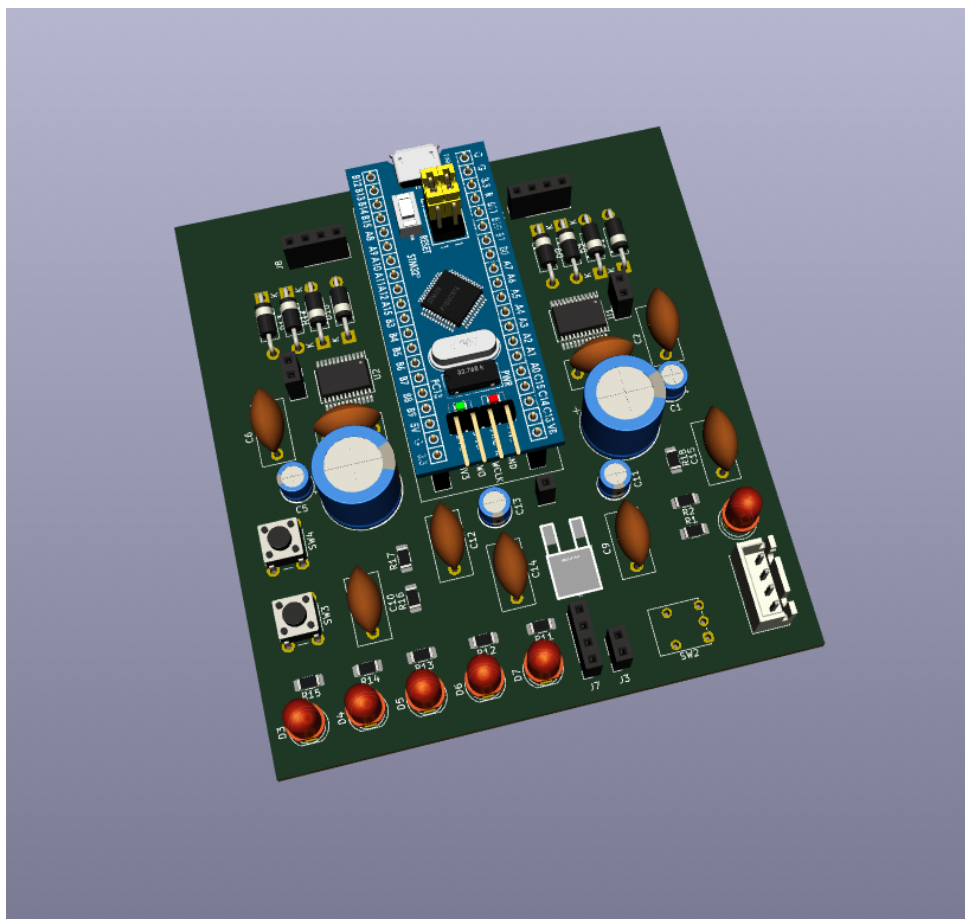
Size: A4 Date:
KICad E.D.A. kicad (5.1.10-1-10.14)



Proximity sensors

White line sensors

Sheet: /sensors/
 File: sensors.sch
Title:
 Size: A4 Date:
 KiCad E.D.A. kicad (5.1.10-1-10.14)
 Rev: Id: 4/4



Rysunek 3: Wizualizacja 3D płytki PCB

5 Dokumentacja

Dokumentacja projektu jest prowadzona przy użyciu systemu kontroli wersji git. Zdalne repozytorium zostało założone na platformie Github ze względu na wcześniej zdobyte doświadczenie w pracy z tym oprogramowaniem. Git umożliwił bezpieczne rozwijanie projektu bez zamartwiania się o to czy po wprowadzeniu nowych zmian wszystko będzie działało bez zarzutów, ponieważ ten system umożliwia wersjonowanie plików i bezpieczne wracanie do zapisanych wcześniej efektów.

W repozytorium obecnie znajdują się pliki KiCada, datasheety podzespołów, dodatkowe biblioteki KiCada oraz projekt programu wraz z kodem na mikrokontroler napisany przy użyciu STM32CubeIDE. Strona główna projektu: <https://github.com/bartoszipiech/minisumo>

6 Harmonogram pracy

W projekcie zostały wytyczone następujące kamienie milowe:

1. Mechanika:

- Projekt mechaniki Minisumo,
- Złożenie zamówienia bądź wykonanie elementów mechanicznych,
- Budowa robota,

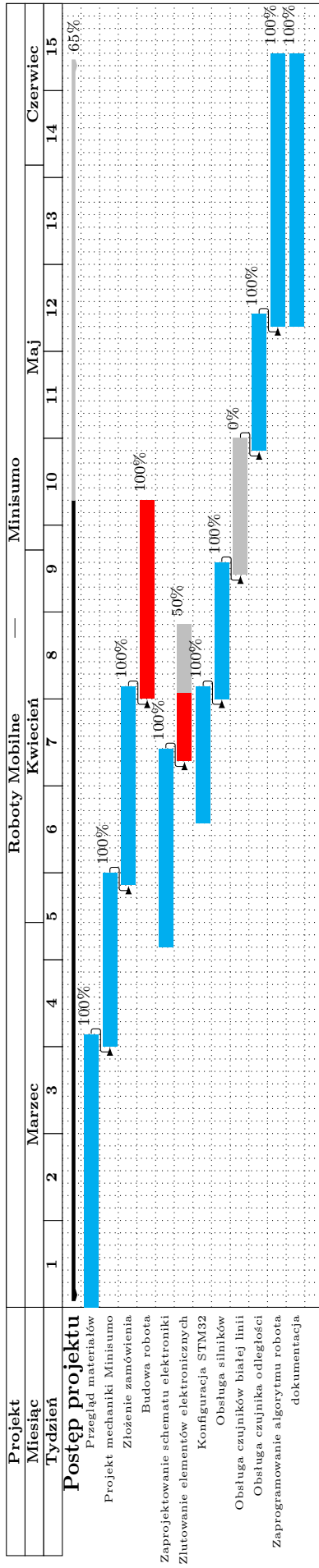
2. Elektronika:

- Zaprojektowanie schematu elektroniki,
- Złożenie oraz zlutowanie elementów elektronicznych,

3. Programowanie:

- Konfiguracja potrzebnych peryferiów z użyciem STM32CubeMX,
- Obsługa silników przez STM32,
- Obsługa czujników białej linii,
- Obsługa czujnika odległości,
- Zaprogramowanie algorytmu robota mobilnego,

6.1 Wykres Gantta



6.2 Podział pracy

Projekt wykonywany jest samodzielnie.

7 Zadania niezrealizowane

Projekt w wersji, którą zakładano nie został zrealizowany do końca ze względu na opóźnienia związane z zamówieniem, prace nad nim zostały zawieszono aż do przyjscia wszystkich elementów elektronicznych i płytek PCB. Postanowiono w ramach projektu zbudować wersję zastępczą robota sumo, który nie mieści się w przedziale wielkościowym oraz wagowym minisumo, jednak pozwala przetestować działanie czujników oraz implementację algorytmów walki.

7.1 Robot zastępczy sumo

Został wykonany robot typu sumo, którego części były dostępne w domu, dlatego nie trzeba było się martwić o dostępność oraz czas realizacji zamówień. Podzespoły użyte do wykonania robota mobilnego typu sumo:

- 2x silnik 65 x 26 mm 5 V z przekładnią 48:1 nieznannej marki,
- sklejka drewniana służąca jako podwozie robota do montowania wszystkich elementów,
- 2x koła i opony z robota RC,
- 1x koło swobodne podtrzymujące tył robota,
- śruby M3, podkładki i nakrętki służące do montowania komponentów,
- 1x cyfrowy czujnik odległości IR 38kHz - 60cm - Pololu 2578,
- 6x baterie AA służące jako zasilanie do silników i modułu sterowania,
- specjalny Shield do Arduino z wbudowanym mostkiem H służącym do zasilania silników

7.2 Wygląd robota zastępczego

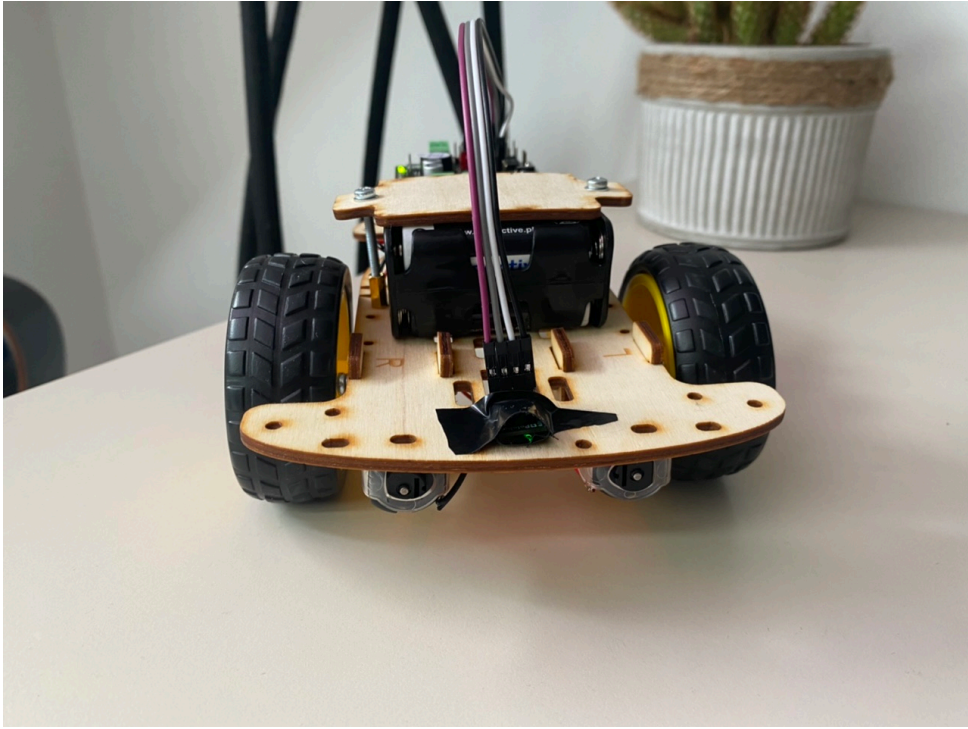
Ze względu na specyfikę użytego sensora odległości, trzeba było zakleić taśmą izolacyjną, ponieważ czujnik wykrywał wszystkie obiekty znajdujące się w pobliżu czujnika. Spowodowało to zmniejszenie zasięgu czujnika z około 60cm do 30-40cm. Po wykonanych modyfikacjach czujnik działa prawidłowo i wykrywa tylko obiekty znajdujące się wprost przed czujnikiem. W kolejnej wersji należałoby przemyśleć mocowanie takiego czujnika, na chwilę obecną przymocowany jest za pomocą taśmy izolacyjnej do podwozia. Spełnia to swoją funkcję, lecz nie jest tak trwałe, jak mocowanie za pomocą śrub.

7.3 Algorytmy sterowania

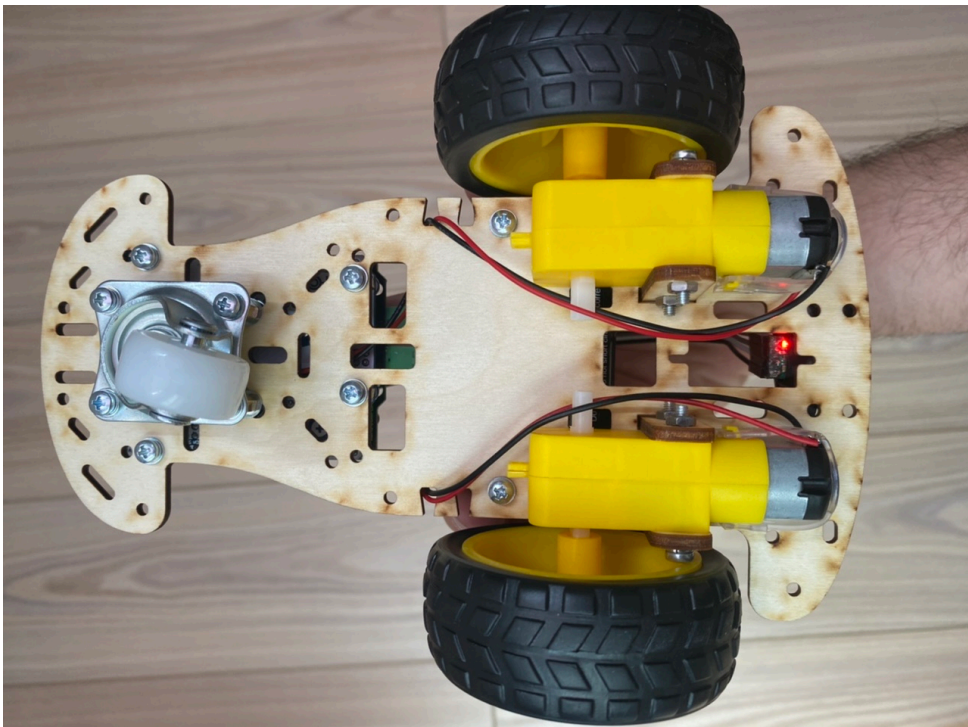
Aby przetestować algorytmy sterowania robotem typu sumo stworzono prowizoryczne dohyo. Przy użyciu kawałków taśmy izolacyjnej stworzono kształt podobny do koła, który miał około 70cm średnicy, tak jak klasyczne dohyo w kategorii minisumo.

Do sterowania robotem podczas walki w dohyo stworzono dwa algorytmy. Jeden polega na ciągłym obrocie robota w jedną ze stron w poszukiwaniu przeciwnika (tornado). Gdy czujnik odległości wykrył przed sobą przeszkodę, robot zmieniał na chwilę kierunek obrotu, aby ustawić się wprost do przeciwnika, następnie ruszał z dużą prędkością na przód i próbował zepchnąć przeciwnika z ringu. Nagrano filmik przedstawiający algorytm tornado: http://panamint.ict.pwr.wroc.pl/~bpiech/uploads/roboty_mobilne/sumo1.mp4

Drugi algorytm polegał na obracaniu przez określony czas w jedną ze stron, następnie przez ten sam czas w stronę odwrotną, dzięki temu robot skanował obszar w poszukiwaniu przeciwnika, po wykryciu przeszkody robot z dużą prędkością ruszał w jej stronę w celu wypchnięcia poza ring. Nagrano filmik przedstawiający algorytm prawo-lewo: http://panamint.ict.pwr.wroc.pl/~bpiech/uploads/roboty_mobilne/sumo5.mp4



Rysunek 4: Widok robota sumo od przodu



Rysunek 5: Widok na podwozie robota sumo

8 Podsumowanie

Robot pomimo, iż nie powstał w wersji, którą zakładano na początku, z pewnością zostanie dokończony jak tylko przyjdą potrzebne komponenty. Pomimo problemów nauka programu KiCad przyda się w przyszłości do konstruowania innych robotów. Zbudowanie zastępczego robota mobilnego sumo dostarczyło dużo doświadczenia i pozwoliło spróbować zaimplementować algorytmy sterowania do takiego robota, a następnie przetestować je na żywo. Po zbudowaniu robota zastępczego okazało się, że czujnik IR, który został wykorzystany w projekcie, nie pozwala w rzeczywistości uzyskać reklamowanego zasięgu wykrycia obiektu, ze względu na potrzebę zakrycia powierzchni znajdujących się dookoła samego czujnika, dlatego należy przemyśleć zmianę czujników odległości w finalnej wersji. Gdy planowany robot minisumo zostanie ukończony, weźmie udział w zawodach minisumo, które odbywają się co roku w całej Polsce (ze względu na COVID-19 zawody były od roku wstrzymane). Mam nadzieję, że stoczy kilka ciekawych walk.