



**PAŃSTWOWA WYŻSZA SZKOŁA ZAWODOWA  
W ELBLĄGU  
INSTYTUT INFORMATYKI STOSOWANEJ  
IM. KRZYSZTOFA BRZESKIEGO**

## **PRACA DYPLOMOWA INŻYNIERSKA**

**Realizacja prototypu pojazdu autonomicznego do pomiarowania  
pomieszczeń wewnątrz budynku**

autor: Marcin Kurzawski

opiekun pracy: dr inż. Rober Smyk

Elbląg, 2021

## Spis treści

1. WSTĘP.....	7
1.1. Cel pracy.....	7
1.2. Zakres pracy.....	7
1.3. Założenia projektowe .....	8
1.4. Struktura pracy.....	10
2. OGÓLNE ZAŁOŻENIA I WYMAGANIA FUNKCJONALNE .....	11
3. ZAKRES PROBLEMATYKI PORUSZANEJ W PROJEKCIE .....	12
4. ANALIZA NARZĘDZI I URZĄDZEŃ WYKORZYSTYWANYCH DO POMIAROWANIA POMIESZCZEŃ.....	13
4.1. Przegląd wybranych platform sprzętowych .....	13
4.2. Przegląd wybranych czujników i sensorów .....	16
5. DOBÓR ALGORYTMU STERUJĄCEGO.....	21
5.1. Przegląd wybranych metod reprezentacji otoczenia i planowania trasy.....	21
5.2. Przegląd wybranych metod planowania ruchu i określania pozycji robota .....	26
5.3. Propozycja reprezentacji otoczenia poprzez macierz zajętości.....	28
5.4. Propozycja algorytmu przeszukiwania DFS .....	28
5.4.1. Zakłócenia pola magnetycznego w badanych pomie- szczeniach .....	30
5.5. Projekt alternatywnego algorytmu sterującego.....	36
5.5.1. Badanie pomieszczenia na podstawie przejazdu wzdłuż ścian .....	36
6. REALIZACJA PROTOTYPU POJAZDU AUTONOMICZNEGO DO POMIAROWANIA POMIESZCZEŃ WEWNĄTRZ BUDYNKU.....	38
6.1. Konstrukcja mechaniczna .....	38
6.2. Układ sterujący .....	39
6.3. Pomiar przyspieszenia i pola magnetycznego.....	40
6.4. Pomiar odległości .....	41
6.5. Sterowanie pracą silników.....	43
6.6. Komunikacja bezprzewodowa .....	44
6.7. Sekcja zasilania .....	44
6.8. Implementacja algorytmu .....	45
6.8.1. Przejazd wzdłuż ścian z wykorzystaniem czujników odległości.....	46
6.8.2. Określenie kierunku i położenia robota przy ścianie.....	46
6.8.3. Kontrola jazdy i korekta kierunku .....	47
6.8.4. Warunek zakończenia pomiarów i korekta współrzędnych.....	49
6.8.5. Opis procesu poprawy i modyfikacji zaimplementowanych algorytmów .....	49
6.8.6. Schemat blokowy algorytmu .....	51
6.9. Podsumowanie finalnej wersji prototypu .....	55

7. PERYFERIA I SPIS WYPROWADZEŃ .....	57
8. TESTY ROBOTA I ZAPROJEKTOWANEGO ALGORYTMU W WARUNKACH RZECZYWISTYCH .....	58
9. STOPIEŃ REALIZACJI WYMAGAŃ FUNKCJONALNYCH I PIERWOTNYCH ZAŁOŻEŃ .....	65
10. WADY I ZALETY METOD OPOMIAROWANIA ZAPROPO- NOWANYCH W PROJEKCIE .....	66
11. DALSZE PRACE .....	67
12. PODSUMOWANIE I WNIOSKI .....	68
Zawartość nośnika dołączonego do pracy: .....	69
Bibliografia .....	69

# 1. WSTĘP

W nowoczesnej robotyce badanie i pomiarowanie pomieszczeń nabiera coraz większego znaczenia. Eksploracja nieznanego obszaru przy jednoczesnym dokładnym określaniu lokalizacji robota wykorzystywane jest na masową skalę w wielkopowierzchniowych zakładach produkcyjnych (np. roboty odpowiedzialne za dostarczanie produktów w magazynie lub transportujące materiały pomiędzy poszczególnymi stanowiskami produkcyjnymi [1]). Autonomiczne pojazdy dokonujące rozpoznania nieznanego pomieszczenia wykorzystywane są także powszechnie w codziennym użytku domowym [2]. Najlepszym przykładem są nowoczesne odkurzacze, które pozwalają na sprząatanie i czyszczenie pomieszczeń bez jakiegokolwiek ingerencji człowieka. Najbardziej znana seria produktów tego typu to urządzenie Roomba produkcji firmy iRobot.

## 1.1. Cel pracy

Celem niniejszej pracy była realizacja prototypu pojazdu autonomicznego do pomiarowania pomieszczeń wewnątrz budynku. Przeznaczenie robota to badanie nieznanego środowiska bez ingerencji człowieka. Pojazd wykonuje automatyczny przejazd dopóki nie zgromadzi wszystkich danych potrzebnych do wyznaczenia kształtów i rozmiaru pomieszczenia. Do eksploracji otoczenia urządzenie wykorzystuje zainstalowane na jego pokładzie czujniki i sensory. Po skończonej pracy pojazd generuje mapę zbadanego obszaru. Mapa ta przedstawia kształt mierzonego pomieszczenia i wszystkie napotkane przez robota przeszkody. W pracy przedstawiono poszczególne etapy prac nad prototypem pojazdu, sposoby rozwiązywania napotkanych problemów. Na końcu pracy znajduje się sprawozdanie z testów pojazdu i poprawności zaimplementowanych algorytmów w warunkach rzeczywistych. Ograniczone fundusze miały istotny wpływ na przebieg pracy i zastosowane rozwiązania.

## 1.2. Zakres pracy

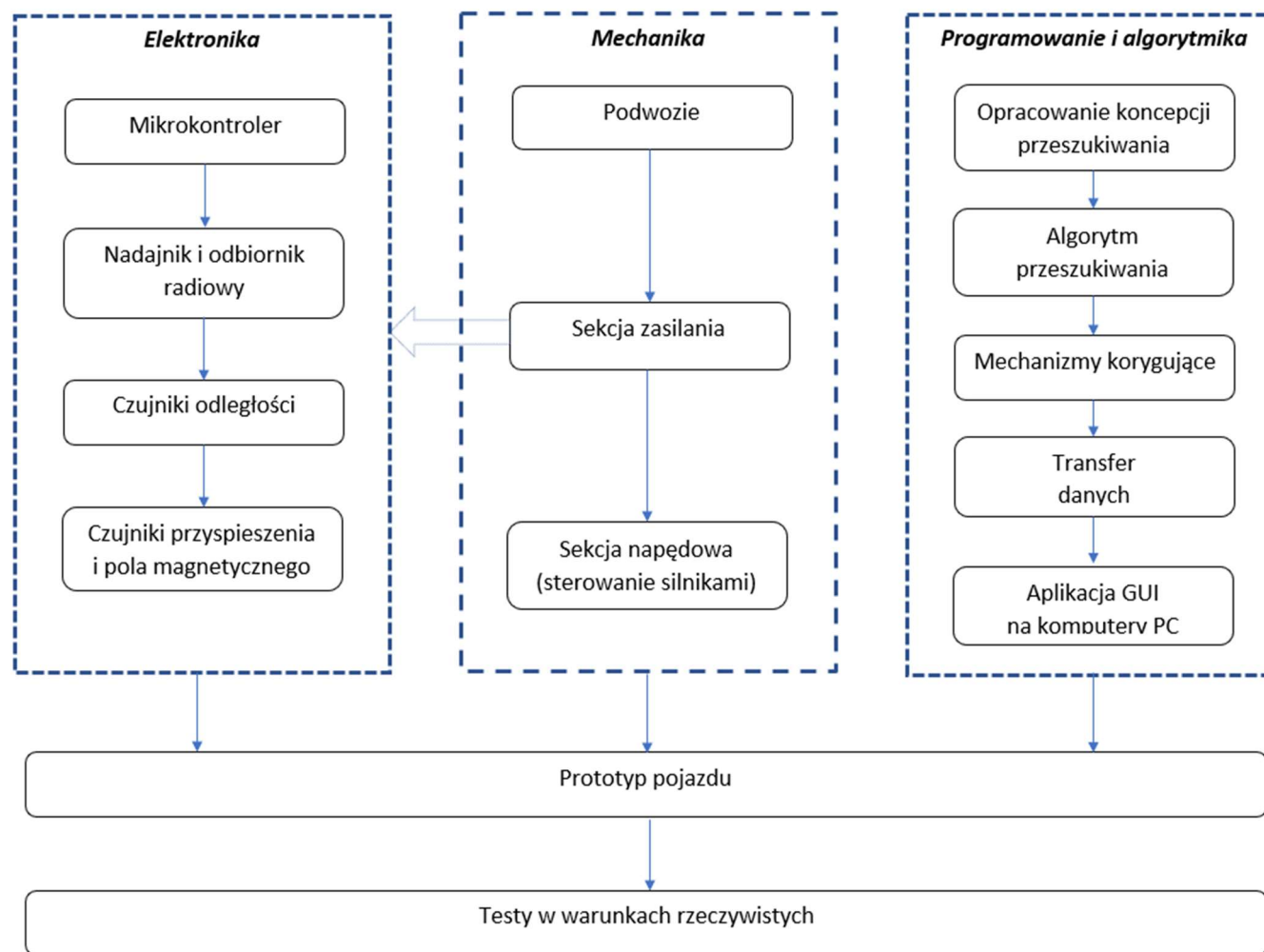
Zakres pracy można podzielić na pięć zasadniczych punktów. Niniejszy projekt porusza tematykę z pogranicza różnych dziedzin nauk technicznych: informatyki, elektroniki oraz robotyki. Zakres pracy obejmuje:

1. Przegląd wybranych algorytmów i metod używanych przy mapowaniu pomieszczeń przez roboty.

2. Przegląd wybranych urządzeń i oprzyrządowania, które wykorzystuje się w robotach przeznaczonych do pomiarowania pomieszczeń.
3. Stworzenie projektu robota i zaprogramowanie go w taki sposób, aby badał kształt i rozmiary pomieszczenia oraz generował odpowiadającą mu mapę.
4. Stworzenie programu komputerowego, który komunikuje się z robotem. Program prezentuje mapę pomieszczenia wygenerowaną podczas przejazdu robota na badanym obszarze.
5. Testy robota i zaimplementowanych algorytmów oraz analiza uzyskanych efektów.

### **1.3. Założenia projektowe**

Główne założenie projektu to stworzenie pojazdu, który wykonuje przejazd na nieznanym obszarze. W projekcie możemy wyróżnić kilka zasadniczych warstw. Pierwszą z nich jest warstwa związana z elektroniką i doбором platformy sprzętowej odpowiedniej klasy, a także wyposażenie robota w sensory odległości oraz czujniki pola magnetycznego i przyspieszenia. Na część dotyczącą konstrukcji mechanicznej składają się prace nad podwoziem, sekcją zasilania oraz sekcją napędową. Elementy związane z programowaniem to opracowanie koncepcji mechanizmu badania pomieszczenia, jej implementacja oraz opracowanie systemu kontroli błędów. Dodatkowo do tej fazy prac możemy zaliczyć także stworzenie graficznej aplikacji użytkownika (ang. *GUI*) przeznaczonej na komputery PC. Umożliwia ona sterowanie robotem, rozpoczynanie i kończenie pomiarów oraz prezentację wyników wykonywanych badań. Na wykresie poniżej (rys. 1.1) przedstawiono podział poszczególnych faz projektu ze względu na tematykę, której dotyczą.



Rys. 1.1 Podział prac nad prototypem

## **1.4. Struktura pracy**

Struktura niniejszej pracy odwzorowuje kolejne etapy prac nad projektem. W pierwszych rozdziałach przedstawiono ogólne założenia i wymagania funkcjonalne, a także dokonano przeglądu dostępnych platform sprzętowych i możliwych do wykorzystania urządzeń i sensorów. Następnie w rozdziale piątym omówiono dobór algorytmu sterującego, pierwotnie opracowaną koncepcję badań oraz problemy, które pojawiły się podczas jej testowania. W rozdziale tym przedstawiono także alternatywną koncepcję wykonywania pomiarów w odpowiedzi na problemy, które się pojawiły. Rozdział szósty przybliży poszczególne elementy składowe, z których składa się prototyp, a także zawiera omówienie zaimplementowanego algorytmu. W rozdziale dziewiątym przedstawiono proces testowania robota w warunkach rzeczywistych i analizę osiągniętych rezultatów. Ostatnie kilka rozdziałów zawiera ogólną podsumowanie, wnioski, krótki opis dalszych możliwości rozbudowy prototypu oraz analizę osiągniętych efektów.

## 2. OGÓLNE ZAŁOŻENIA I WYMAGANIA FUNKCJONALNE

Przed przystąpieniem do wyboru koncepcji i realizacji projektu dokonano opracowania ogólnych założeń i wymagań funkcjonalnych. Poniżej przedstawiono najważniejsze punkty przeprowadzonej analizy.

1. Przed przystąpieniem do pomiarów robot nie posiada informacji ani o wymiarach pomieszczenia, które bada, ani o jego kształcie i może zostać umieszczony w dowolnym miejscu w pomieszczeniu.
2. Robot kończy pracę w tym samym miejscu, w którym rozpoczynał pomiary.
3. Mapa pomieszczenia reprezentowana jest jako macierz komórek (macierz zajętości). Komórka może przyjmować trzy wartości : "wolne", "zajęte", "nieznane".
4. Praca robota odbywa się bez żadnego udziału człowieka i jest w pełni automatyczna. Do poprawnej pracy nie jest wymagane żadne wcześniejsze przygotowanie pomieszczenia bądź robota.
5. Robot wykonuje przejazd po pomieszczeniu i na podstawie odczytu z czujników generuje mapę odpowiadającą rozkładowi pomieszczenia.
6. Obsługa robota i prezentacja wyników odbywa się w aplikacji komputerowej, stworzonej specjalnie do celów niniejszego projektu. Użytkownik aplikacji w każdej chwili może zatrzymać pracę pojazdu.
7. Po zakończonym przejeździe robot wysyła dane o zbadanym obszarze drogą bezprzewodową do komputera.
8. Robot posiada odrębny tryb ręcznego sterowania, który daje użytkownikowi możliwość swobodnej jazdy. W trybie tym użytkownik aplikacji ma możliwość bezpośredniego odczytywania wskazań czujników robota (wskazania odległości oraz poprzecznego i podłużnego nachylenia robota). Dane te wyświetlane są w czasie rzeczywistym.



### **3. ZAKRES PROBLEMATYKI PORUSZANEJ W PROJEKCIE**

Tematyka związana z robotami mobilnymi porusza zagadnienia z pogranicza robotyki, elektroniki, informatyki i algorytmiki. Pierwszym ważnym aspektem z jakim mamy do czynienia podczas pracy nad robotami mobilnymi to dobór lub opracowanie metody badania pomieszczeń. Obrana metoda wpływa na dobór czujników potrzebnych w przyjętym systemie badań. Bardzo ważnym zagadnieniem jest także wybór odpowiedniej platformy sprzętowej, która odpowiada za sterowanie robota i obsługę wszystkich czujników. Platforma ta musi posiadać odpowiednio wysokie taktowanie, być wyposażona w odpowiednią liczbę portów i interfejsów, wystarczającą do obsługi wszystkich zamontowanych w robocie czujników i urządzeń. Kolejną kwestią jest dobór odpowiedniej konstrukcji mechanicznej. Zależy ona nie tylko od przyjętego modelu badań oraz implementowanych mechanizmów korekcyjnych, ale także od rodzaju terenu w jakim porusza się robot. Pod względem konstrukcji mechanicznej najczęściej napotkamy na jeden z pięciu głównych typów robotów: kołowe, gąsienicowe, kroczące, latające i pływające. Nieodpowiedni dobór konstrukcji mechanicznej i mechanizmów korekcyjnych może mieć katastrofalny wpływ na rezultaty badania i systemy pozycjonowania robota.

## **4. ANALIZA NARZĘDZI I URZĄDZEŃ WYKORZYSTYWANYCH DO POMIAROWANIA POMIESZCZEŃ**

Dobór odpowiedniej platformy sprzętowej oraz dodatkowych czujników, w ścisły sposób powiązany jest z przyjętą koncepcją badania pomieszczenia oraz samą konstrukcją mechaniczną robota. W dwóch kolejnych podrozdziałach dokonano ogólnego omówienia najważniejszych mikrokontrolerów oraz wybranych sensorów i urządzeń powszechnie stosowanych w pojazdach autonomicznych, których przeznaczeniem jest eksploracja nieznanego środowiska wewnątrz budynku.

### **4.1. Przegląd wybranych platform sprzętowych**

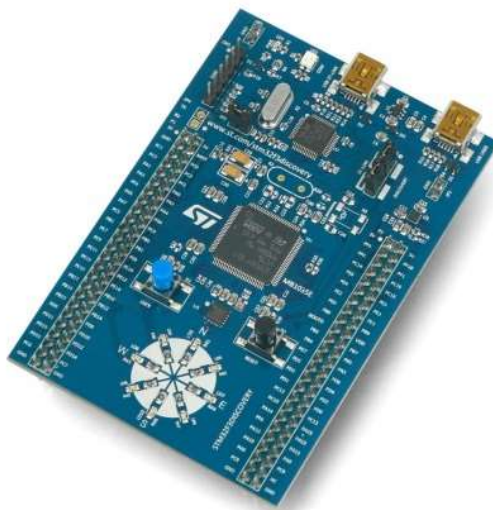
Dobór odpowiedniej platformy zależy przede wszystkim od przyjętej metody pomiarowania. Niektóre z metod eksploracji, takie jak badanie otoczenia przy użyciu kamer, charakteryzują się zwiększonym zapotrzebowaniem na moc obliczeniową. Dobrany mikrokontroler musi być w stanie przetworzyć wszystkie dane dostarczane przez czujniki robota. Spośród platform sprzętowych dostępnych na rynku najbardziej rozpowszechnione są trzy klasy mikrokontrolerów. Każda z nich posiada różną charakterystykę pracy i zakres wykorzystania.

Pierwszą popularną grupą platform sprzętowych jest rodzina układów 8 – bitowych [3]. Najbardziej rozpowszechnionym mikrokontrolerem z tej rodziny jest moduł Arduino Uno (podstawowa platforma z rodziny Arduino), widoczny na rys. 4.1. Na jego pokładzie znajduje się mikrokontroler AVR ATmega328. Zestawy z tej rodziny posiadają jednak dość skromne parametry. Maksymalna częstotliwość zegara wynosi zaledwie 16 MHz. Pamięć SRAM to 2 kB, a pamięć Flash 32 kB, z czego 5 kB zarezerwowane jest dla bootloadera. Moduł Arduino Uno posiada 14 portów I/O, z czego 6 portów może pracować w trybie PWM. Arduino posiada także interfejsy szeregowe UART, SPI oraz I2C. Obsługuje także zewnętrzne przerwania. Posiada gniazdo USB A przeznaczone do programowania oraz złącze DC 5,5 X 2,1 mm umożliwiające podłączenie układu do zewnętrznego źródła zasilania. Istnieją także wersje Arduino wyposażone w większą ilość portów i nieco większe zasoby sprzętowe, z kilkoma dodatkowymi portami pracującymi w trybie PWM. Koszty jakie trzeba ponieść przy zakupie modułów z rodziny Arduino są więc niewspółmierne do możliwości jakie oferuje produkt. Przykładowo, koszt modułu Arduino Uno Rev3 wynosi ok. 100 zł [4].



Rys. 4.1 Arduino Uno Rev3

Kolejną platformą, którą warto rozważyć przy budowie robota są mikrokontrolery 32 – bitowe. Dzięki większej objętości rejestrów układy te lepiej radzą sobie z przetwarzaniem większej ilości danych w porównaniu do układów 8 – bitowych. Zazwyczaj posiadają także lepsze parametry pod względem wydajności i nie ograniczają się do taktowania niskimi częstotliwościami [5]. Doskonałym przykładem tej grupy mikrokontrolerów jest seria produktów STM32 (rys. 4.2). Są to moduły z 32-bitowym mikrokontrolerem wyposażonym w rdzeń ARM Cortex. Platforma ta jest kompatybilna z nakładkami Arduino oraz platformą mbed. Rodzina tych układów posiada liczne timery, dużą ilość wyprowadzeń, liczne interfejsy (m.in. SPI, I2C, USART, UART, USB Full Speed, CAN, wirtualny port COM) [6]. Płytki wyposażone są zazwyczaj w dwa przyciski (jeden resetujący i jeden do dyspozycji użytkownika). Programując mikrokontroler istnieje możliwość zasilenia mikrokontrolera z zewnętrznego źródła napięcia w zakresie od 3,3 V do 12 V. Możliwości układów tego typu dodatkowo rozszerzają timery i przetworniki analogowo cyfrowe. Płytki bardzo często wyposażone są w dodatkowe układy i wewnętrzne czujniki, np. czujnik temperatury, wilgotności, magnetometr, akcelerometr, żyroskop, referencyjny pomiar napięcia itd. Bardzo ważną cechą układów STM32 jest bardzo niska cena w stosunku do oferowanych możliwości i jakości. Przykładowo mikrokontroler STM32 NUCLEO-F103RB kosztuje jedynie 50 zł, oferując przy tym dużą ilość wyprowadzeń, liczne interfejsy i wysoką wydajność (częstotliwość taktowania układu do 72 MHz).



Rys. 4.2 STM32F3 – Discovery [7]

Najwyższej klasy grupą platform sprzętowych omawianą w tym rozdziale jest seria układów 32 – bitowych klasy SoC (ang. System-On\_Chip). Urządzenia z tej grupy można traktować w kategoriach minikomputera, ponieważ wyposażone są one w bardzo dużą ilość pamięci i wysokowydajny procesor [8]. Flagowym reprezentantem urządzeń tej grupy jest produkt RaspberryPi. Jest to rodzaj minikomputera, który posiada duże zasoby obliczeniowe i szereg wysokiej jakości peryferiów. Najnowszym mikrokontrolerem z tej serii jest produkt Raspberry Pi 4 model B (rys. 4.3). Minikomputer wyposażony jest w cztery złącza USB (2 w standardzie 3.0 oraz 2 w standardzie 2.0), 2 złącza microHDMI, oraz złącze zasilania USB C [9]. Dodatkowo urządzenie wyposażone jest w dwuzakresowe WiFi 2,4GHz i 5 GHz, Bluetooth 5/BLE oraz port Ethernet o prędkości do 1000Mb/s. Na pokładzie płytki znajduje się 40 złącz GPIO, złącze SCI, DSI oraz slot kart pamięci microSD. Minikomputer produkowany jest w trzech różnych wersjach, odmiennych pod względem dostępnej pamięci RAM (2, 4 oraz 8 Gb). Serce układu stanowi procesor BCM2711. Jest to chipset z 64-bitowym rdzeniem Quad-core ARM-8 Cortex-A72 CPU, taktowany z częstotliwością 1,5 GHz. Powyższy opis doskonale obrazuje doskonałe parametry urządzenia oraz możliwości i zastosowania jakie oferuje. Zakup urządzenia tego typu pociąga za sobą duże koszty. Najnowszy model Raspberries Pi 4 kosztuje około 400 zł. Moduły z rodziny RaspberryPi charakteryzują się doskonałym stosunkiem jakości do ceny. Jednak jednostkowy koszt modułu jest na tyle duży, że warto rozważyć wybór innych produktów, szczególnie jeżeli możliwości minikomputera nie będą w pełni wykorzystane.



Rys. 4.3 Raspberry Pi model 4 [10]

## 4.2. Przegląd wybranych czujników i sensorów

Do poprawnego zbadania wymiarów i kształtów pomieszczenia konieczne jest zebranie odpowiednich danych za pomocą stosownych czujników. Prace koncepcyjne nad robotem wymagały przeglądu sensorów, które pasowałyby do wstępnie przyjętych wymagań i założeń funkcjonalnych.

Dość powszechnym rozwiązaniem w robotyce jest zastosowanie silników krokowych lub silników wyposażonych w enkodery (rys. 4.4). Silnik krokowy umożliwia precyzyjny obrót silnika o zadany kąt dzięki czemu posiadamy dokładną informację o przejechanej odległości. Enkoder jest licznikiem, który działa na podobnej zasadzie i zlicza ilość obrotów jakie wykonuje koło pojazdu. Zasada działania takiego mechanizmu polega na wykrywaniu zmian w naświetleniu elementów koła bądź wykrywaniu impulsu magnetycznego generowanego przez element umieszczony na kole, ewentualnie wykrywaniu zmian w rezystancji wbudowanych oporników. Enkoder zazwyczaj jest mniej dokładny, ale umożliwia jazdę z szybszą prędkością w porównaniu do silnika krokowego. Poważną wadą obu rozwiązań jest jednak wysoka cena silników krokowych i silników wyposażonych w enkodery [11].



Rys. 4.4 Silnik krokowy z przekładnią 28BYJ-48 [12]

Skutecznymi i tanimi sensorami do badania otoczenia robota są ultradźwiękowe czujniki odległości (rys. 4.5). Szczególnie popularne są sensory ze zintegrowanym transmitters i odbiornikiem wiązki. Czujniki tego typu mierzą odległość od wykrytej przeszkody wykorzystując emitowane przez sensor ultradźwięki. Wadą tego typu sensorów jest dość mały zakres działania i wykrywania przeszkód (zwykle jest to od 1 m do 4 m). Z racji wykorzystania wiązki akustycznej cechują się one niskim czasem pojedynczego pomiaru. Obliczenie odległości od wykrytej przeszkody wykonuje się na podstawie czasu zmierzonego od momentu wyemitowania fali akustycznej do momentu wykrycia odbitej wiązki przez odbiornik. Do wyliczeń wykorzystuje się prosty wzór (wzór numer 1) oraz wartość prędkości dźwięku w powietrzu (340m/s) [13].

$$dystans = czas\_zmierzony\_w\_sekundach * prędkość\_dźwięku / 2 \quad (1)$$



Rys. 4.5 Zintegrowany ultradźwiękowy czujnik odległości HY-SRF05

Kolejną grupą czujników są czujniki wykorzystujące promieniowanie światła podczerwonego (rys. 4.6). Sensor wysyła wiązkę promieniowania, a następnie wykrywa sygnał odbity od przeszkody. Podobnie jak ma to miejsce w przypadku sensorów ultradźwiękowych, do badania odległości czujnika od przeszkody wykorzystuje się pomiar czasu, który upłynął od momentu wyemitowania wiązki z transmittera do momentu wykrycia odbitej fali podczerwonej przez odbiornik. Dużą wadą sensorów tego typu jest niewielka odległość od czujnika jaką możemy skutecznie zbadać (zwykle około 2 m) [14].



Rys. 4.6 Zintegrowany czujnik odległości wykorzystujący podczerwień [15]

Jednym z najbardziej zaawansowanych sposobów badania otoczenia robota są czujniki i skanery laserowe. Jest to szeroka grupa sensorów, które charakteryzują się niezwykle dużą precyzją i dokładnością. Pomimo wykorzystania wiązki laserowej, produkowane czujniki tego typu są bezpieczne dla ludzkiego oka i nie wymagają szczególnych środków ostrożności. Możemy wyróżnić dwie podgrupy sensorów wykorzystujące wiązkę lasera [16]. Pierwszą grupą są czujniki wysyłające pojedynczą wiązkę laserową w jednym kierunku. Precyzyjne pomiarowanie ułatwia niewielki kąt wykrywania przeszkody, wynoszący zazwyczaj od  $0,5^\circ$  do  $1^\circ$ . Drugą podgrupę czujników wykorzystujących wiązkę laserową są skanery laserowe. Skanery wykonują cykliczne pomiary w zakresie  $360^\circ$ . Pomimo tego, że czujniki tego typu doskonale sprawdzają się przy pomiarowaniu pomieszczeń i budowaniu mapy nieznanego obszaru, mają one jednak jedną zasadniczą wadę, którą jest bardzo wysoki koszt sensora [17]. Na rysunku nr 4.7 widoczny jest przykładowy skaner laserowy, działający w zakresie 360 stopni.



Rys. 4.7 Skaner laserowy rplidar – 360 stopni [18]

Następnym analizowanym sensorem jest czujnik typu Time of Flight. W rozwiązaniu tym wykorzystuje się pomiar czasu jaki wygenerowana wiązka światła widzialnego potrzebuje na powrót do czujnika po odbiciu się od przeszkody.

W porównaniu do czujników ultradźwiękowych i wykorzystujących podczerwień, sensory te posiadają duży zasięg oraz cechują się szybszą prędkością działania. Zasada ToF jest metodą pomiarów dystansu pomiędzy sensorem a obiektem na podstawie różnicy czasu pomiędzy momentem wygenerowania wiązki świetlnej, a jej powrotem do sensora po odbiciu się od obiektu. Rozwiązanie takie jest w pełni bezpieczne dla ludzkiego oka i nie stanowi zagrożenia dla człowieka. Pomiarowanie pomieszczeń z wykorzystaniem metody ToF posiada jednak znaczące wady. Największe z nich jakie należy wymienić to większe zakłócenia w porównaniu do pomiarów czujnikami wykorzystującymi podczerwień i ultradźwięki. Zaletą takiego rozwiązania jest duży zasięg czujników. Profesjonalne sensory mogą badać obszar oddalony nawet o 60 metrów od czujnika, choć ich koszt jest wysoki (około 400 zł) [19]. Na rysunku 4.8 przedstawiono przykładowy sensor wykorzystujący zasadę ToF.



Rys. 4.8 Czujnik odległości tof reranger-evo-60m [20]

Do pomiarowania pomieszczeń z powodzeniem można wykorzystać kamery i systemy wizualne (rys. 4.9). Wykorzystanie bardziej zaawansowanych kamer pozwala na tworzenie zarówno dwu jak i trójwymiarowych map badanego obszaru. Możemy wyróżnić dwa główne podejścia, które wykorzystywane są podczas korzystania z systemów wizualnych [21], [22]. Pierwsze z nich polega na określaniu krawędzi, identyfikacji powierzchni, ścian i przeszkód na podstawie obrazu uzyskiwanego z kamery i odpowiednim przetworzeniu go. Połączenie obrazu z kamery oraz metod wykorzystywanych w symultanicznej lokalizacji i mapowaniu daje bardzo dobre efekty, ale zazwyczaj cechuje się zwiększonym zapotrzebowaniem na zasoby sprzętowe i moc obliczeniową. Drugi ze sposobów polega na wcześniejszym przygotowaniu pomieszczenia poprzez wyznaczenie w nim kilku punktów charakterystycznych (np. plakietki z symbolem lub linii po której pojazd ma podążać). Pojazd podczas pracy określa swoją lokalizację na podstawie wyznaczonych



wcześniej w ten sposób sztucznych znaczników bądź podąża po wcześniej przygotowanej drodze.



Rys. 4.9 System wizualny multisense-s7 do badania odłości [23]

Odrębną grupę sensorów stanowią czujniki do określania kierunku robota na podstawie wartości pola magnetycznego, przyspieszenia i prędkości kątowej. Są to odpowiednio: magnetometr, akcelerometr i żyroskop. Sensory takie występują zarówno pod postacią osobnych modułów dedykowanych pomiarom jedynie pojedynczych wielkości, jak i kilku czujników zintegrowanych w jednym układzie, wykorzystującym do komunikacji jeden interfejs. Zazwyczaj głównym modułem jest magnetometr, a pozostałe elementy pełnią jedynie rolę pomocniczą i korekcyjną. Rysunek 4.10 przedstawia układ scalony, który zawiera w sobie akcelerometr, magnetometr i żyroskop.



Rys. 4.10 Układ Pololu 852 (akcelerometr, żyroskop i magnetometr w jednym) [24]

## 5. DOBÓR ALGORYTMU STERUJĄCEGO

Przed przystąpieniem do prac nad konstrukcją robota przeprowadzono analizę algorytmów, których użycie byłoby potencjalnie pomocne w pełnej i łatwej realizacji przyjętych założeń i wymagań funkcjonalnych. Przeanalizowano także wybrane metody lokalizacji, planowania ruchu oraz reprezentacji pomieszczenia.

### 5.1. Przegląd wybranych metod reprezentacji otoczenia i planowania trasy

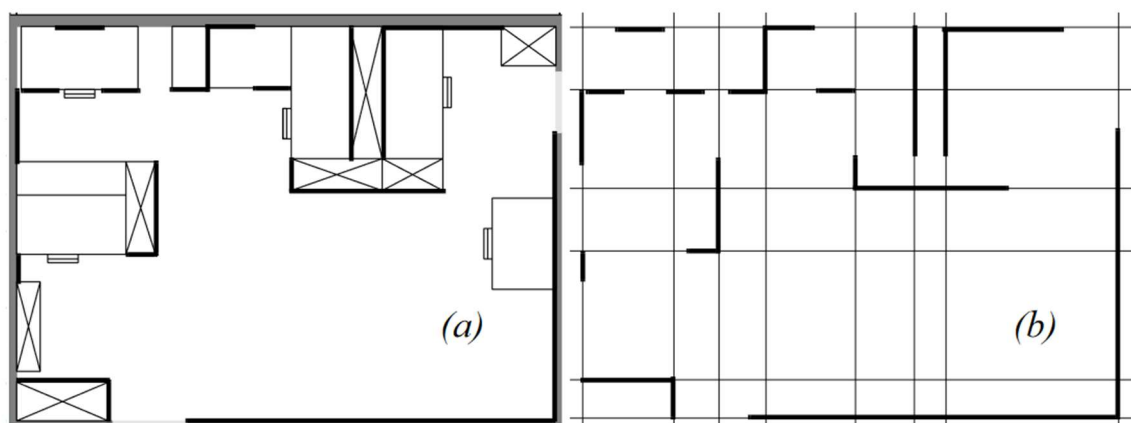
Do poprawnego zbadania pomieszczenia konieczne jest precyzyjne wyznaczenie pozycji oraz kierunku robota. Dobór odpowiedniej metody reprezentacji otoczenia bezpośrednio wiąże się ze sposobem w jaki określamy pozycję robota w przestrzeni, a także od samej konstrukcji mechanicznej pojazdu. Od używanego algorytmu zależy dalszy dobór platformy sprzętowej oraz sensorów i czujników pojazdu.

Rozważania na temat metod reprezentacji otoczenia i pozycjonowania robota należy rozpocząć od klasyfikacji warunków w jakich pracuje robot. W tym przypadku podział ze względu na charakter badanego środowiska przebiega pomiędzy metodami statycznymi a dynamicznymi [25], [26]. Statyczność sceny oznacza w tym przypadku niezmiennosc konfiguracji przestrzennej danego obszaru. W metodach statycznych, w czasie badania pomieszczenia konfiguracja przestrzenna, czyli rozmieszczenie poszczególnych przeszkód i obiektów nie ulega w trakcie ruchu robota żadnym zmianom. Jeżeli w trakcie ruchu robota obiekty zmieniają swoje położenie mamy do czynienia ze środowiskiem dynamicznym. Ważnym elementem jest w tym przypadku rozróżnienie przeszkód dynamicznych ze względu na ciągłość ruchu na dwie grupy ruchu obiektów. Pierwsza podgrupa to obiekt, który co pewien czas może być przestawiony na inną pozycję (obiekt quasi-dynamiczny). Przykładem takich obiektów są drzwi, krzesła, czy kosz na śmieci. Drugą podgrupą jest obiekt ruchomy, czyli wykonujący ruch w chwili obserwacji. Obiektem takim będzie człowiek poruszający się z robotem we wspólnym korytarzu lub drzwi w czasie ich otwierania i zamykania. Istnieje szereg algorytmów służących do poprawnego rozróżnienia pomiędzy podgrupą obiektów semi-statycznych a obiektami dynamicznymi (przykładem takiego algorytmu jest wykrywanie różnic pomiędzy kolejnymi pomiarami środowiska).

Kluczowym zagadnieniem podczas badania pomieszczeń jest dobór odpowiedniej metody reprezentacji badanego obszaru. Najbardziej rozpowszechnionym w literaturze kryterium podziału jest zależność od rodzaju opisu

mapy [27]. Możemy wyróżnić trzy podstawowe metody reprezentacji eksplorowanego środowiska: mapę metryczną, topologiczną oraz hybrydową [25].

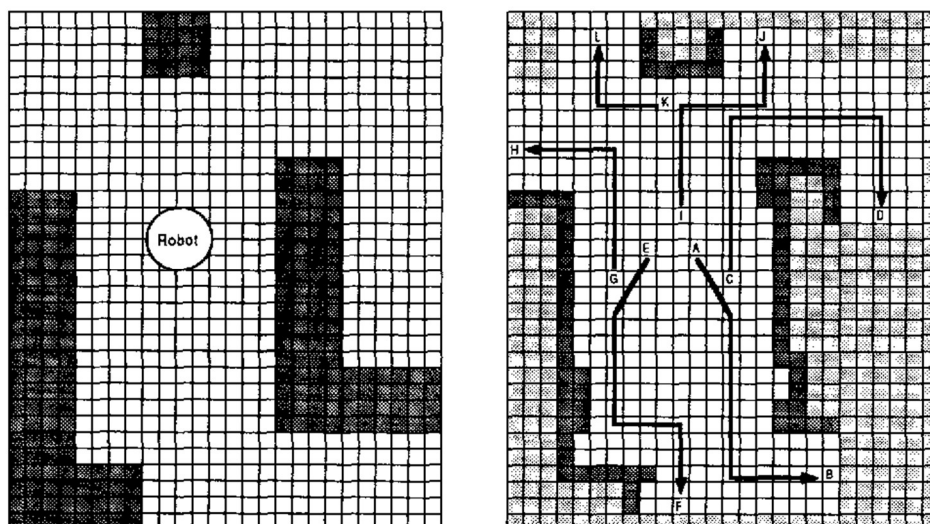
Pierwsza z wymienionych reprezentacji, do opisu charakterystyki otoczenia wykorzystuje układ współrzędnych. Metoda ta najczęściej obejmuje dekompozycję przestrzeni na siatkę komórek (wraz z uwzględnieniem zajętości danej komórki). Istnieją także podejścia, które opierają się na tworzeniu mapy jako cech charakterystycznych obserwowanego środowiska (określanie wierzchołków przeszkód i obiektów, a także definiowanie przeszkód i obiektów - przedstawianie przeszkód w formie linii). Przykład takiej reprezentacji przedstawiono na rysunku nr 5.1, gdzie robot tworzy lokalną mapę metryczną z wykorzystaniem odcinków, gdzie znajduje się przeszkoda. Reprezentacja metryczna wiąże się często z metodą określania pozycji robota przy użyciu dekompozycji komórkowej. Poprawne odwzorowanie otoczenia przy użyciu wyłącznie mapy metrycznej jest zagadnieniem skomplikowanym. Poważnym problemem w tym przypadku jest wyznaczenie odpowiedniej wielkości poszczególnych komórek w tworzonej mapie. Zbyt duży rozmiar komórek sprawi, że obraz środowiska stworzony podczas badania będzie obarczony dużym błędem. Kolejnym problemem jest pozyskiwanie odpowiednio dokładnych informacji dotyczących danych metrycznych. Zarówno o umiejscowieniu cech charakterystycznych badanego środowiska jak i samej pozycji robota, na podstawie której estymowana jest pozycja przeszkody.



Rys. 5.1 Rzeczywisty rozkład przeszkód w pomieszczeniu (a) oraz odwzorowanie przeszkód w lokalnej mapie metrycznej (b) [28]

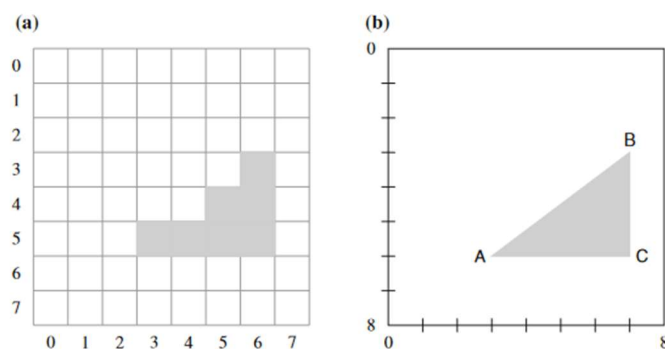
Mapy metryczne możemy podzielić na dwie osobne podgrupy: mapy metryczne ciągłe oraz mapy metryczne dyskretne [29]. Pierwszym sposobem metrycznego odwzorowania badanego obszaru jest dyskretna mapa metryczna (przedstawiona na

rysunku nr 5.2. W metodzie tej tworzona mapa przyjmuje formę siatki komórek o stałym rozmiarze. Im mniejsze wymiary pojedynczej komórki tym większa mapa. W metodzie tej pomieszczenie przedstawione jest jako dwuwymiarowa macierz, gdzie pojedynczy element może przyjmować wartości: nieznane, zajęte, wolne (ewentualnie wybrane dwie wartości spośród trzech wymienionych: wolne/nieznane, zajęte).



Rys. 5.2 Przykład wykorzystania mapy metrycznej do badania pomieszczeń [30]

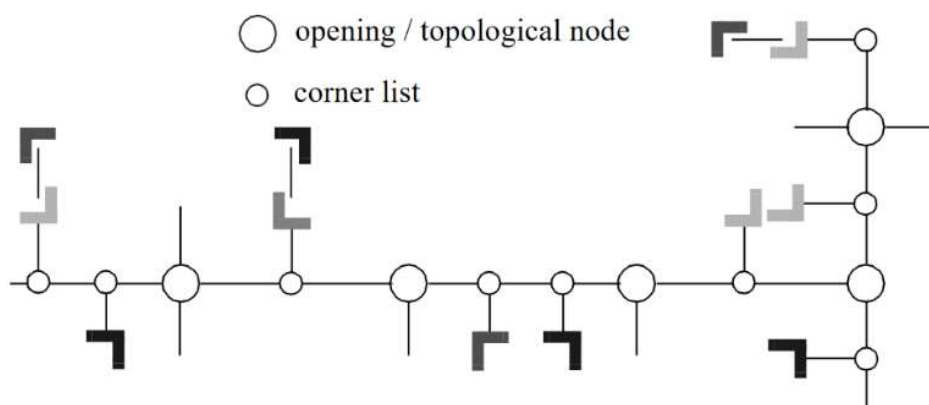
Nieco odmienną formą reprezentacji jest ciągła mapa metryczna. Robot zbierając dane z czujników pozyskuje dane o otoczeniu zazwyczaj z wykorzystaniem chmury punktów. Chmura punktów generowana jest na podstawie wartości odczytywanych z czujników robota (np. sensorów laserowych lub grupy czujników ultradźwiękowych). Zebrane punkty umieszczane są w kartezjańskim układzie współrzędnych. Następnie dane te przetwarzane są przez CPU lub GPU. Mapa generowana jest w absolutnym układzie odniesienia. Model ten bazuje na dokładnych wymiarach i pozycjach przeszkód w otoczeniu. Różnicę pomiędzy poszczególnymi podejściami doskonale obrazuje rysunek 5.3 obrazujący różnice pomiędzy podejściem mapy metrycznej dyskretnej a mapy metrycznej ciągłej. W pierwszym przypadku lokalizacja obiektu przechowywana jest w pamięci jako lista komórek zajętych. W przypadku ciągłej mapy metrycznej ten sam obiekt przechowywany jest jako lista współrzędnych jego wierzchołków. Poniższy przykład doskonale obrazuje jak ważny jest proces dyskretyzacji i przyjęcia odpowiednio wysokiej rozdzielczości macierzy komórek (tzw. macierzy zajętości).



Rys. 5.3 Zestawienie mapy metrycznej dyskretnej (rys. a) oraz mapy metrycznej ciągłej [31]

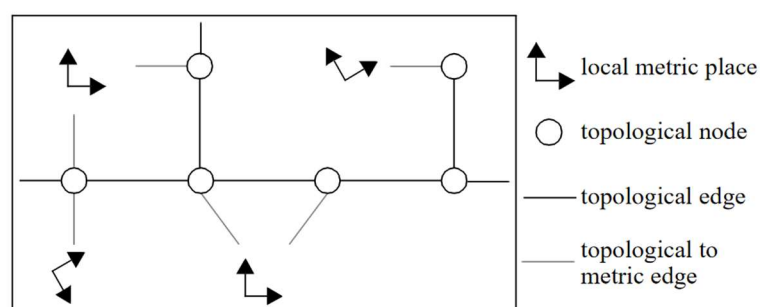
Ostatnią metodą, o której należy wspomnieć w kontekście reprezentacji badanego obszaru jest mapa topologiczna. W mapach tych badane środowisko reprezentowane jest jako graf. Poszczególne wierzchołki (węzły) grafu opisują miejsca, w których pojazd może się znaleźć. Dodatkowo dla każdego węzła w grafie przechowywane są dodatkowe informacje o cechach otoczenia, które mogą być w danym miejscu obserwowane. Informacje te pomagają w późniejszej identyfikacji pozycji robota w grafie. O dwóch wierzchołkach w grafie możemy powiedzieć, że są połączone w sytuacji, gdy istnieje bezkolizyjna droga łącząca opisywane miejsca. Wyznaczanie odpowiedniej drogi odbywa się poprzez znalezienie najkrótszej ścieżki w grafie. Określenie dokładnej lokalizacji pojazdu polega na wyszukaniu wierzchołka grafu, w którym zapamiętane cechy otoczenia są najbardziej podobne do aktualnie obserwowanych. Mapa topologiczna pozwala na efektywne zapamiętywanie informacji o otoczeniu robota i planowanie przejazdu w przypadku, gdy robot porusza się po bardzo dużych powierzchniach. Jeżeli pojazd posługuje się mapą topologiczną przedstawianą w formie grafu, nie ma wówczas możliwości sprawdzenia na podstawie samego grafu, w którym węzle aktualnie się znajduje. Jest to dość znacząca wada i z tego powodu w każdym wierzchołku grafu umieszczona jest dodatkowa informacja dotycząca danych metrycznych. Zazwyczaj informacja ta dotyczy informacji o otoczeniu, które w danym punkcie jest obserwowane. Mapa topologiczna tworzona jest więc jako siatka węzłów oraz informacji o sieciach połączeń pomiędzy nimi. Zazwyczaj w węzłach przechowywane są dane na temat odległości i kierunku innych sąsiadujących wierzchołków grafu (węzłów) oraz cechach charakterystycznych środowiska obserwowanego w danym wierzchołku grafu. Informacje te obejmują zwykle charakterystykę dotyczącą krawędzi oraz wolnych przestrzeni otaczających robota. Dane te pozwalają na podróż robota w odpowiednim kierunku i na odpowiednią odległość oraz poprawną identyfikację położenia w odniesieniu do wierzchołka.

Rysunek nr 5.4 przedstawia przykład mapy topologicznej oraz charakterystyczne cechy środowiska w poszczególnych punktach grafu. Węzły, w których w jednym z kluczowych kierunków wykryto przeszkodę oznaczone są znacznikiem małym. Natomiast węzły, w których robot nie wykrył żadnych przeszkód zaznaczono jako koła większe.



Rys. 5.4 reprezentacja otoczenia z wykorzystaniem mapy topologicznej [32]

Reprezentacja hybrydowa jest metodą łączącą dwie poprzednie metody: mapę topologiczną oraz mapę metryczną [27]. Istnieje wiele podejść różniących się stopniem rozbudowania i stopniem skomplikowania. Metoda hybrydowa może używać globalnej mapy topologicznej stworzonej z lokalnych map metrycznych lub też wykorzystywać globalną mapę metryczną oraz bardziej szczegółowe mapy topologiczne. Na rysunku 5.5 przedstawiony jest prosty przykład mapy hybrydowej łączącej globalną mapę topologiczną oraz lokalne mapy metryczne. Otoczenie reprezentowane jest przez miejsca umieszczone na mapie metrycznej oraz węzły reprezentujące lokalacje topologiczne. Podczas podróży pomiędzy węzłami pojazd przełącza się z systemu topologicznego na metryczny i odwrotnie. Przełączenie nie wymaga żadnych dodatkowych charakterystyk otoczenia.



Rys. 5.5 Reprezentacja otoczenia przy użyciu mapy hybrydowej [32]

## 5.2. Przegląd wybranych metod planowania ruchu i określania pozycji robota

Przyjęta reprezentacja związana jest z późniejszym pozycjonowaniem i planowaniem ruchu podczas pracy robota przy badaniu pomieszczeń. Planowanie trasy bardzo często przeprowadza się wykorzystując algorytm najkrótszej ścieżki. Przykładem takiego algorytmu jest powszechnie używany w informatyce algorytm Dijkstry. Przy omawianiu pozycjonowania robota jeden z głównych podziałów metod przebiega pomiędzy metodami globalnymi a metodami lokalnymi. Metody globalne wykorzystują globalny układ odniesienia, zaś metody lokalne bazują na informacjach dotyczących bezpośredniego otoczenia robota, zebranych przez jego czujniki.

Spośród lokalnych metod planowania trasy i wyznaczania kierunku robota wymienić należy dwie najważniejsze: BugAlgorithm oraz metoda Histogramu Pola Wektorowego (ang. Vector Field Histogram) [33]. Pierwszy z nich - Bug Algorithm, jest to sposób pomiarowania inspirowany zachowaniami owadów. Do poprawnego planowania trasy robot wykorzystuje przybliżone dane o kierunku napotkanej przeszkody oraz aktualne odczyty z czujników. Robot może poruszać się po linii prostej w kierunku celu. W momencie napotkania przeszkody pojazd otacza lub okrąża ją, po czym oddala się od niej w punkcie przecięcia prostej łączącej punkt startowy i docelowy. W metodzie Histogramu Pola Wektorowego dane na temat otoczenia robota zapisywane są w formie siatki w kartezjańskim układzie współrzędnych. Na podstawie odczytu z czujników robota każdej komórce przypisuje się wartość równą prawdopodobieństwu wystąpienia w niej przeszkody. Stworzona siatka zredukowana jest następnie do histogramu biegunowego. Każdy wycinek histogramu obrazuje prawdopodobieństwo wystąpienia przeszkody w danym kierunku. Dzięki temu robot może ominąć przeszkodę wykorzystując np. funkcję kosztu do obliczenia kierunku, który powinien przyjąć.

Jedną z metod globalnych określania pozycji robota jest metoda oparta na mapach drogowych [33]. Robot podczas przejazdu generuje graf określający sieć ścieżek po jakich może poruszać się robot. Graf zawiera informacje o relacjach pomiędzy poszczególnymi jego wierzchołkami. Jest to rodzaj mapy topologicznej, która opiera się na utworzeniu grafu przejścia między znacznikami, które w grafie umieszczane są jako węzły. W celu poprawnego korzystania z mapy robot musi zlokalizować siebie w jednym z węzłów oraz być w stanie przemieszczać się między

węzłami (wymagana duża precyzja: rzeczywista pozycja musi odpowiadać pozycji estymowanej – wyznaczanej przez konkretny węzeł).

Metoda orientacji robota wykorzystująca dekompozycję komórkową polega na podzieleniu przestrzeni na regiony (komórki) i wyznaczeniu trasy i kierunku na podstawie określenia, które puste komórki są sąsiadujące [34]. W zależności od przyjętego poziomu dokładności podziału, przejazd odbywa się na podstawie wyznaczonego na tej podstawie grafu spójności lub na podstawie generowanej siatki zajętości. Algorytm ten jest prosty do implementacji. Skutecznie działa w większych pomieszczeniach o regularnych kształtach. Jest on bardzo szybki, wymaga niewielkich zasobów sprzętowych, ale cechuje się niedokładnością i błędami przy niewielkich i nieregularnych przeszkodach. Komórka oznaczona jest jako obszar niedostępny, nawet jeżeli jest ona zajęta w niewielkim stopniu. Wyznaczenie granic pomiędzy komórkami może być wykonane poprzez określenie wszystkich punktów jednakowo odległych od wszystkich przeszkód. Inne podejście bazuje na podziale na komórki o stałej szerokości (najczęściej stosowany wariant tej metody).

Odminną formą pozycjonowania robota jest metoda wykorzystująca zewnętrzny sygnał radiowy [35]. Do poprawnego działania wymagane jest użycie przynajmniej trzech nadajników ułożonych w różnych miejscach w pomieszczeniu. Dzięki procesowi trilateracji istnieje możliwość określenia gdzie znajduje się robot. Poza oczywistą wadą, jaką jest konieczność wcześniejszego ustawienia nadajników, dodatkowo sygnał może być zagłuszony przez elementy konstrukcyjne budynku (ściany, zbrojenie, rury itd.) lub inne nadajniki pracujące na tej samej bądź zbliżonej częstotliwości.

W nowoczesnej robotyce coraz częściej spotyka się także metody, które do pozycjonowania wykorzystują rozpoznawanie obrazu przy użyciu kamer [36]. Z ich pomocą możliwe jest np. wykrycie krawędzi ścian i zmierzenie odległości do nich. Kamera umożliwia także określanie pozycji robota z wykorzystaniem rozmieszczonych przed pomiarem znaków charakterystycznych (sztuczne znaczniki - np. różnokolorowe kształty). Na rynku istnieją kamery 2 D, 2,5 D oraz 3 D. Te ostatnie umożliwiają tworzenie trójwymiarowego modelu środowiska. Metody wykorzystujące kamery charakteryzują się zwiększonym zapotrzebowaniem na moce obliczeniowe. Kamery wysokiej jakości wiążą się z dość wysokimi kosztami (zwłaszcza kamery 3D, umożliwiające badanie głębi obrazu). Do interpretowania obrazu stosuje się zazwyczaj specjalistyczne biblioteki i programy matematyczne takie jak MATLAB.



### **5.3. Propozycja reprezentacji otoczenia poprzez macierz zajętości**

Na etapie projektowania algorytmu pomiarowania pomieszczeń przyjęto założenie, że badany obszar reprezentowany jest jako macierz komórek. Każda komórka odpowiada kwadratowi o wymiarach 20 na 20 cm. Zakładamy, że robot nie posiada żadnej informacji o pomieszczeniu, które ma zbadać i działa on w środowisku statycznym (podczas badań nie dochodzi do zmian położenia przeszkód). Rozmiar dwuwymiarowej tablicy jest stały w czasie działania programu i nie ulega zmianie. Jako początkową wartość położenia pojazdu przyjmujemy połowę długości tablicy. Na potrzeby projektu założono, że wartości w macierzy mogą przyjmować jeden z trzech stanów: wolny, zajęty lub nieznany. Początkowo wszystkie komórki dwuwymiarowej macierzy przyjmują wartość nieznaną. Jedynie pozycja początkowa robota określona jest na początku jako wolna. Robot, wykonując automatyczny przejazd, sprawdza swoje sąsiedztwo w czterech kierunkach i na tej podstawie aktualizuje macierz zajętości. W przypadku wykrycia przeszkody oznacza odpowiednią komórkę jako zajętą. Robot wykonuje pomiary do momentu, w którym w macierzy zajętości pozostaje co najmniej jedna komórka o statusie nieznanym do której może dotrzeć i sprawdzić jej stan. Jeżeli w macierzy nie ma więcej takich komórek robot powraca w miejsce od którego zaczynał pomiary. Jednocześnie z macierzą zajętości aktualizowana jest macierz odwiedzonych już komórek.

### **5.4. Propozycja algorytmu przeszukiwania DFS**

W początkowym etapie projektu założono, że robot będzie badał całe pomieszczenie na podstawie algorytmu bardzo zbliżonego do rozpowszechnionego w informatyce algorytmu przeszukiwania w głąb (ang. Depth-First-Search). Poniżej opisano podstawowe założenia opracowanego algorytmu.

Pojazd może poruszać się w cztery strony (symbolicznie północ, południe, wschód, zachód). Robot może zostać umieszczony w dowolnym punkcie pomieszczenia, a jego symboliczne kierunki zostają wyznaczone na podstawie odczytu magnetometru. Kierunek, w którym początkowo zorientowany jest robot traktuje się jako „północ” (kierunek 0 stopni). Reszta symbolicznych kierunków jest wyznaczana przez dodanie do aktualnej wartości odpowiednio: 90, 180 oraz 270 stopni. Następnie dzięki dzieleniu modulo przez 360 otrzymujemy wszystkie symboliczne strony świata. Traktujemy je jako osie, po których może poruszać się robot. Według tych kierunków następuje także korekta kursu, jeśli robot odchyli się za bardzo z osi jazdy.

Badanie bezpośredniego sąsiedztwa robota przeprowadzane jest za pomocą czterech czujników ultradźwiękowych umieszczonych na każdym boku robota. Przeszkoda wykrywana jest gdy odczyt z czujnika odległości wynosi mniej niż 25 cm. Dodatkowym zabezpieczeniem jest także czujnik dotykowy umieszczony z przodu robota. Zabezpiecza on przed niebezpieczną sytuacją, w której robot stoi nachylony do ściany pod dużym kątem. W takim przypadku odczyt z czujnika odległości będzie drastycznie zawyżony, a robot zderzy się ze ścianą lub inną przeszkodą, czego konsekwencją może być jego całkowite zablokowanie. Dzięki czujnikowi, po zderzeniu z przeszkodą robot zatrzymuje się, wraca do punktu poprzedniego i oznacza obszar gdzie występuje przeszkoda jako zajęty.

Niezwykle ważnym aspektem w początkowo przyjętej metodzie pomiarowania jest dodatkowa macierz z mapą odwiedzonych komórek (jej rozmiar odpowiada rozmiarowi macierzy zajętości). Komórki w tej tablicy mogą przyjmować dwie wartości: odwiedzone lub nieodwiedzone, co w programie komputerowym odpowiada wartości logicznej 0 lub 1. Początkowo wszystkie komórki przyjmują wartość nieodwiedzone (wartość 0). Jedynym wyjątkiem jest tutaj komórka odpowiadająca pozycji początkowej robota. Analogicznie jak w macierzy zajętości jej współrzędnym odpowiada połowa długości tablicy. Zastosowanie dodatkowej tablicy reprezentującej odwiedzone już miejsca jest konieczne, żeby robot nie badał wielokrotnie tego samego obszaru. Inspiracją do pomiarowania pomieszczeń z wykorzystaniem dodatkowej tablicy reprezentującej zbadany już obszar był algorytm Trémauxa. Jest to trywialny algorytm, którego pierwotnym przeznaczeniem jest rozwiązywanie labiryntu. Jednak mimo to, doskonale nadaje się on do adaptacji przy badaniu pomieszczeń metodą podobną do metody DFS. Połączenie magnetometru, macierzy zajętości i macierzy miejsc odwiedzonych pozwala na skuteczne i łatwe w implementacji rozwiązanie przeznaczone do badania każdego rodzaju pomieszczenia, nawet takiego o nieregularnych kształtach.

#### **5.4.1. Zakłócenia pola magnetycznego w badanych pomieszczeniach**

W czasie prac nad pierwotną wersją prototypu napotkano na niespodziewane i bardzo poważne problemy związane z kluczowym komponentem elektronicznym robota, jakim jest kompas elektroniczny (moduł Pololu LSM303D). Przy próbie zaimplementowania mechanizmu skrętu robota zgodnie z głównymi kierunkami w jakich może się on poruszać, pojazd wykazywał się niestabilnym i nieprzewidywalnym działaniem. W celu zdiagnozowania źródła problemu wykonano badania i diagnostykę magnetometru oraz dokonano weryfikacji pomiarów pola magnetycznego.

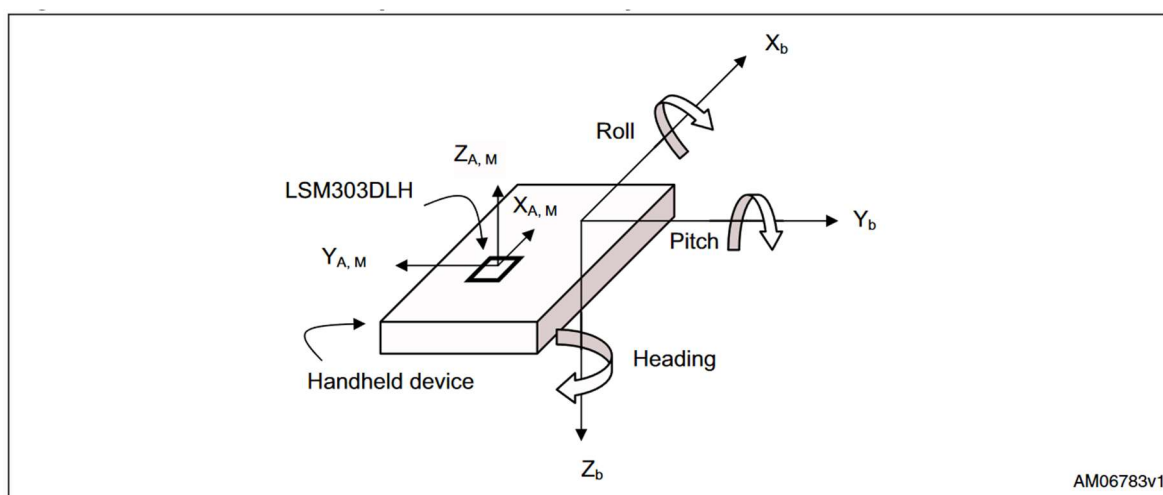
Pierwszym działaniem przeprowadzonym w ramach diagnostyki magnetometru było podłączenie go do mikrokontrolera Arduino Auno oraz sprawdzenie odczytu kompasu elektronicznego i wartości przyspieszeń przy użyciu dedykowanej biblioteki udostępnionej specjalnie dla tego modelu płytki przez firmę Pololu [37] [38] [39]. Biblioteka pozwala na szybką kalibrację czujnika oraz udostępnia wygodny interfejs do odczytywania aktualnych wartości przyspieszenia, kierunku i wartości pola magnetycznego. Na obu przetestowanych mikrokontrolerach (STM32RBT6 i Arduino Uno) wskazania magnetometru były bardzo zbliżone, wobec czego przystąpiono do dalszej weryfikacji wartości pola magnetycznego z wykorzystaniem innego modułu magnetometru.

Do celu dalszej diagnostyki pola magnetycznego wykorzystano mikrokontroler z rodziny STM Nucleo STM32F411E Discovery (rys. 5.6). Posiada ona wbudowany moduł magnetometru LSM303DLHC zintegrowany z akcelerometrem [40]. Zastosowana płytka nie posiada wbudowanego portu COM, dlatego do odczytania aktualnych wskazań pola magnetycznego wykorzystano konwerter UART – USB, która pozwala na komunikację płytki z komputerem i umożliwia wygodną transmisję danych. Porównując wskazania modułu LSM303DLHC oraz modułu LSM303D nie zauważono jednak znaczących różnic przy pomiarach pola przy dwóch ustawionych obok siebie modułach. Szczególną uwagę zwrócono natomiast na różnice wskazań kierunku w zależności od umiejscowienia kompasu elektronicznego na powierzchni stołu, na którym wykonywano pomiary. Aby upewnić się, że wartości odczytane z sensorów magnetycznych cechują się jak najmniejszym błędem, przeprowadzono dalszą analizę problemu z uwzględnieniem możliwych niedokładności powstałych na skutek wychylenia układu względem jego płaszczyzny poziomej.

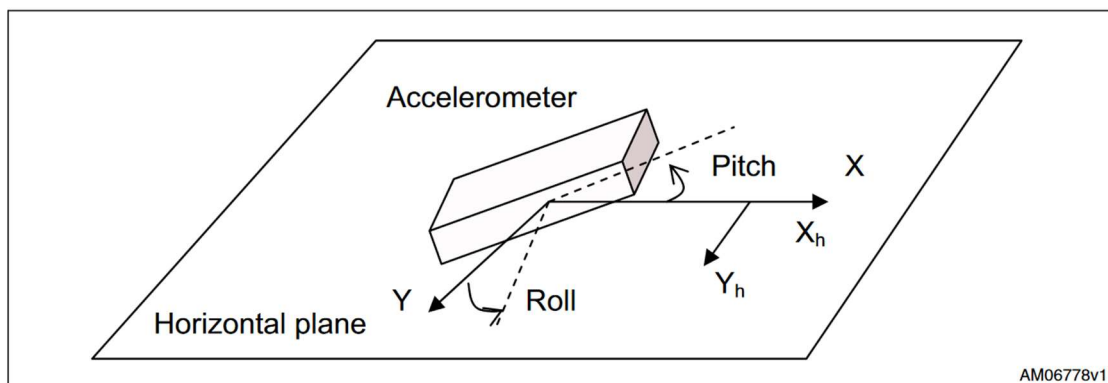


Rys. 5.6 Płytką STM Nucleo STM32F411E Discovery

W celu eliminacji błędów związanych z wypoziomowaniem układu, wykorzystano metodę kompensacji przechyłu. Metoda ta polega na jednoczesnym wykorzystaniu wskazań magnetometru i wartości odczytanych z akcelerometru. Na rys. 5.7 i 5.8 przedstawiono schematy z noty aplikacyjnej producenta modułu, obrazujące omawiane przechylenia. Istnieje kilka różnych metod wyznaczenia kątów wychylenia i skorygowanych wartości pola magnetycznego. W projekcie skorzystano ze sposobu opisanego w instrukcji użytkownika i w notce aplikacyjnej układu LSM303D. Dodatkowo producent urządzenia udostępnia dedykowane biblioteki, pozwalające na szybką i łatwą obsługę czujnika.



Rys. 5.7 Fragment z noty aplikacyjnej producenta magnetometru LSM303DLH przedstawiający osie, względem których badane jest pole magnetyczne [41]



Rys. 5.8 Fragment z noty aplikacyjnej producenta magnetometru LSM303DLH obrazujący badane wychylenia [41]

Do wyznaczenia wychylenia podłużnego i poprzecznego (ang. *pitch and roll*) posłużono się dwoma wzorami wykorzystującymi dane z akcelerometru (wzór numer 2 i 3).

$$Roll = atan(A_z / A_y) \quad (2)$$

$$Pitch = atan(-A_x / (A_y * \sin(Roll) + A_z * \cos(Roll))) \quad (3)$$

Gdzie  $A_x$ ,  $A_y$ ,  $A_z$  oznaczają wartości odczytane z sensora przyspieszenia. Obliczone w ten sposób wartości wychyleń wykorzystujemy następnie do wyznaczenia znormalizowanych wartości wektorów pola magnetycznego (wzór numer 4 i 5).

$$X_h = Y_M * \cos(Roll) - Z_M * \sin(Roll) \quad (4)$$

$$Y_h = X_M * \cos(Pitch) + Y_M * \sin(Pitch) * \sin(Roll) + Z_M * \sin(Pitch) * \cos(Roll) \quad (5)$$

Gdzie  $X_h$ ,  $Y_h$  oznaczają znormalizowane wartości wektora magnetycznego, a wartości  $X_M$ ,  $Y_M$ ,  $Z_M$  oznaczają nieprzetworzone wartości odczytane z sensora magnetycznego. Aby obliczyć znormalizowany kierunek wstawiamy wyliczone wartości do wzoru nr 6:

$$Heading = atan(Y_h / X_h) \quad (6)$$

Obliczona wartość może jednak przyjąć wartość ujemną, dlatego ostateczną wartość naszego kierunku otrzymujemy wykonując prostą operację modulo zgodnie ze wzorem umieszczonym poniżej (wzór nr 7).

$$H = (Heading + 360) \bmod 360 \quad (7)$$

Testy zaimplementowanej kompensacji przechyłu na mikrokontrolerze F4 wykazały, że znormalizowane wartości wektora pola magnetycznego wykazują się

dużą stabilnością. Przy wychyleniu 45 stopni błąd bezwzględny kierunku wynosił około 2 stopnie. Przed zastosowaną normalizacją nie było możliwości poprawnego odczytu kierunku przy tak dużym odchyleniu układu. Porównanie wskazań magnetometru LSM303D oraz modułu LSM303LDHC nie przyniosło jednak odpowiedzi na pytanie o przyczynę nieprzewidywalnej jazdy pojazdu przy korzystaniu z magnetometru. Po weryfikacji poprawności wskazań układu LSM303D kolejnym krokiem było więc zbadanie odczytów z magnetometru umieszczonego bezpośrednio na robocie.

Aby przetestować działanie magnetometru podczas pracy robota i jego różnych wychyleń, moduł LSM303D odseparowano od innych urządzeń i przewodów umieszczając go na osobnej płytce stykowej. Następnie wykonano serię odczytów przy włączonych i wyłączonych silnikach. Wartości przyspieszeń miały wartości poprawne, jednak odczyty z sensora magnetycznego wykazywały się dużą zależnością od miejsca ułożenia płytki stykowej z modułem oraz pracy silników elektrycznych. Moduł LSM303D jest urządzeniem czułym, mocno reagującym na kontakt ze wszystkimi metalowymi elementami i urządzeniami wytwarzającymi pole elektromagnetyczne. Aby zaradzić temu problemowi zastosowano płytkę stykową, przyklejoną do serwomechanizmu dodanego na przedzie robota. Serwomechanizm dodano w celu weryfikacji odczytów z magnetometru przed i po wykonaniu obrotu przez pojazd i ułatwieniu przy wyszukiwaniu potencjalnego błędu. Robota zaprogramowano w taki sposób, aby sam wykonał próbę skrętu o 90 stopni. Przebieg badania wyglądał następująco:

- obrót serwa o 90 stopni i odczyt kąta wartości referencyjnej  
powrót serwa do położenia 0 stopni
- powolny skręt robota przy jednoczesnym odświeżaniu aktualnych wartości z sensora magnetycznego
- zatrzymanie robota w momencie kiedy wartość bezwzględna różnicy wartości referencyjnej i aktualnego kąta jest mniejsza od 15 stopni (umyślna duża tolerancja – pewność, że nie skręcimy za daleko)

W wyniku testu okazało się jednak, że zastosowanie modułu LSM303D w takim układzie jeszcze bardziej potęguje problem. Ostatecznie do serwomechanizmu doklejona została cienka, długa płytka z tworzywa sztucznego, na końcu której umieszczono magnetometr. Pozwoliło to na całkowite odseparowanie magnetometru od wszelkich źródeł pola magnetycznego. Początkowo obiecujące testy wykazały jednak, że stabilność pracy pojazdu wykorzystującego do skrętów magnetometr jest silnie skorelowana z miejscem w jakim znajduje się robot. Zachowanie takie dało

powody do przypuszczeń, że błędy w pracy prototypu spowodowane są bardzo dużymi zewnętrznymi zakłóceniami magnetycznymi. W ostatecznej wersji projektu serwomechanizm nie był użytkowany, ale magnetometr pozostał na stworzonej konstrukcji (oddalenie od elementów generujących zakłócenia).

W celu potwierdzenia podejrzeń o wpływie zewnętrznego pola magnetycznego na pracę robota, przeprowadzono szereg pomiarów w pomieszczeniach, w którym robot był do tej pory budowany i testowany. Pomiary rozpoczęto od zbadania odczytu wszystkich wektorów z sensora magnetycznego opisanego wcześniej modułu LSM303LDHC wbudowanego w mikrokontroler STM32 F4 (wraz z kompensacją przechyłu). Zbadane wartości drastycznie różniły się w zależności od umiejscowienia magnetometru w pomieszczeniu. Ogromnym różnicom podlegały nie tylko odczyty z sensorów w osi poziomej (X, Y), ale także odczyt z sensora osi pionowej. Pionowy wektor pola magnetycznego jest stały dla danej szerokości geograficznej i nie zmienia się nawet w dużej odległości od konkretnego punktu. Kierunek wskazywany przez magnetometr porównano także z kompasem elektronicznym w telefonie komórkowym. W obu przypadkach otrzymano identyczne odchylenia od normy. Ostatecznego potwierdzenia wpływu zewnętrznego pola magnetycznego na pomiary dokonano tradycyjnym kompasem analogowym. Również w tym przypadku kierunek północny zależał od miejsca ułożenia kompasu. Wskazania z kompasu analogowego zgadzały się ze wskazaniami kierunku z modułu LSM303D. Trudnością w szybkim zidentyfikowaniu problemu było testowanie wskazań układu w jednym miejscu podczas pracy z modułem przy biurku i komputerze. Magnetometr testowany wraz z kompensacją przechyłu tylko w jednym punkcie, działał stabilnie i zgodnie z przewidywaniami. Utrudniło to szybką identyfikację problemu. Na rysunkach 5.9 i 5.10 przedstawiono zdjęcia z pomiarów pola magnetycznego kompasem analogowym. Rysunki te doskonale prezentują skalę zróżnicowania zakłóceń magnetycznych (do niniejszej pracy dołączony został nośnik zawierający film z przeprowadzonych pomiarów).





Rys. 5.9 Pomiary pola magnetycznego na długości 1,5 m



Rys. 5.10 Wyniki pomiarów pola magnetycznego odpowiednio od prawej: w punkcie początkowym, 60 cm od punktu początkowego, 150 cm od punktu początkowego (największa zbadana różnica wynosi około 80-90 stopni)

Podczas przeprowadzonych testów nie udało się określić jednego punktu, z którego pochodziłyby zakłócenia. Nie zaobserwowano, żeby rozchodziły się one koncentrycznie lub liniowo. Szczególnym zaskoczeniem okazała się być niestabilność wektora magnetycznego skierowanego pionowo do ziemi. Za główną przyczynę problemu można podejrzewać bezpośrednie sąsiedztwo linii wysokiego napięcia z badanymi pomieszczeniami (testy prowadzono w budynku stojącym 30 metrów od linii wysokiego napięcia). Dodatkowo zaobserwowano silną korelację pomiędzy bliskością podłoża (podłoga) a skalą błędów. Z tego powodu można wywnioskować, że wpływ na błędne wskazania mają w tym przypadku także elementy zbrojenia budynku. Używanie magnetometru wewnątrz pomieszczeń wiąże się zwykle z niewielkimi zakłóceniami, ale nie wpływają one znacząco na jego działanie. Układy tego typu z powodzeniem stosowane są nawet w amatorskiej automatyce i robotyce [42]. Dlatego skala zjawiska napotkana podczas pracy na prototypem robota



w badanych pomieszczeniach nie była brana na początku pod uwagę. Ogromne zakłócenia przy pomiarach pola magnetycznego sprawiają, że robot opierający swe działanie na pomiarach kierunku pozbawiony jest uniwersalności, a jego użycie ogranicza się do miejsc, w których nie występuje promieniowanie elektromagnetyczne tła. Z tego powodu na dalszym etapie prac dokonano modyfikacji prototypu i początkowych założeń projektowych w taki sposób, aby w jak największym stopniu zachować uniwersalność pojazdu.

## **5.5. Projekt alternatywnego algorytmu sterującego**

Niespodziewane problemy przy wyznaczaniu kierunku z wykorzystaniem magnetometru sprawiły, że pojawiła się konieczność zmiany pierwotnie opracowanej koncepcji. Zmiana sposobu pomiarów wiązałaby się z koniecznością zamiany zastosowanych części i elementów elektronicznych. Pociągnęłoby to za sobą znaczne zwiększenie kosztów finansowych poniesionych na budowę prototypu. Z tego względu podjęto decyzję o zachowaniu dotychczasowej konstrukcji mechanicznej robota. Zdecydowano się na opracowanie nowej koncepcji pomiarów bazującej na pomiarowaniu pomieszczeń w oparciu o usytuowanie ścian. Koncepcję przejazdu na całym obszarze badanego pomieszczenia zastąpiono systemem pomiarowania w oparciu o przejazd wzdłuż ścian.

### **5.5.1. Badanie pomieszczenia na podstawie przejazdu wzdłuż ścian**

W przyjętej koncepcji alternatywnej cały ciężar badania oparto na czujnikach ultradźwiękowych. Zmiana koncepcji badania pociągnęła za sobą konieczność zmiany ułożenia czujników i opracowanie szeregu mechanizmów korygujących. Nowa forma pomiarów wymusiła dostosowanie wcześniej przyjętego założenia o sprawdzeniu całej powierzchni badanych pomieszczeń. Założenie o dowolnej pozycji startowej przekształcono tak aby umożliwić realizację stojącego przed robotem zadania bez konieczności uzupełniania konstrukcji o nowe sensory. W zmodyfikowanym systemie pomiarowania, robot powinien zostać ustawiony w taki sposób, aby po jego prawej stronie w odległości około 5 – 25 cm znajdowała się jedna ze ścian. Nowo opracowana koncepcja wymusiła zmiany dotyczące macierzy zajętości. W algorytmie pomiarowania macierz zajętości przyjmuje dwa stany (wolny i zajęty). Dodatkowa macierz miejsc odwiedzonych, w tym przypadku nie jest używana przez co została ostatecznie odrzucona.

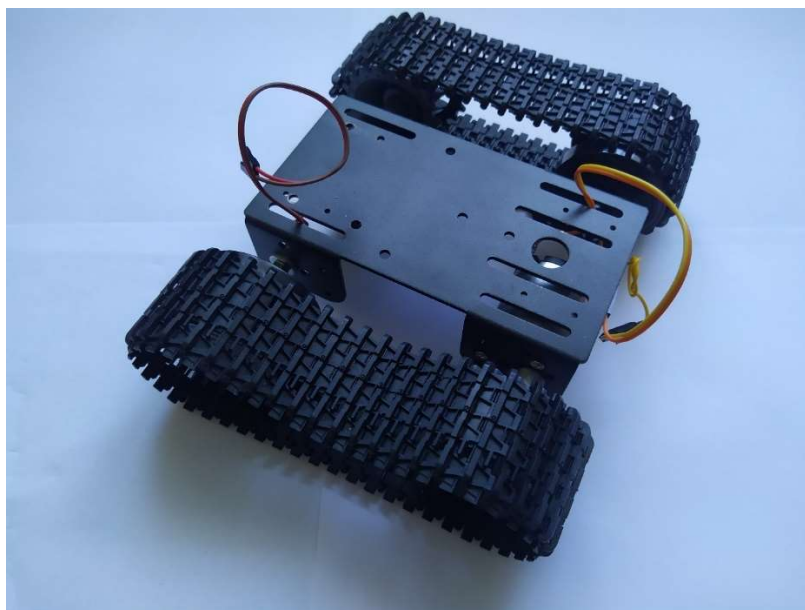
Na potrzeby projektu opracowano specjalną metodę przejazdu robota wraz z mechanizmem korekcji błędów. Mechanizm pomiarowania i tworzenia mapy pomieszczenia wykorzystuje zliczanie ilości ruchów wykonywanych w poszczególnych osiach, w których robot może się poruszać, (ale bez wykorzystania enkoderów). Zastosowanie silników krokowych bądź enkoderów ułatwiłoby zadanie, ale pociągnęłoby za sobą znaczące koszty. Robot wykonuje pomiary, dopóki nie przejedzie całego pomieszczenia wzdłuż jego ścian. W momencie, gdy pojazd skierowany jest w kierunku startowym oraz znajduje się w tym samym miejscu, w którym zaczynał pomiar badanie pomieszczenia kończy się.

## **6. REALIZACJA PROTOTYPU POJAZDU AUTONOMICZNEGO DO POMIAROWANIA POMIESZCZEŃ WEWNĄTRZ BUDYNKU**

Faza prac konstrukcyjnych była poprzedzona etapem, w którym dokonano opracowania koncepcji działania pojazdu. Po przeanalizowaniu literatury i dostępnych materiałów, powszechnie stosowanych rozwiązań i dostępnych urządzeń opracowano system, w którym najważniejszymi elementami są ultradźwiękowe czujniki odległości oraz kompas elektroniczny. Przyjęto, że robot powinien być uniwersalny i na każdym podłożu działać z taką samą skutecznością. Początkowo założono, że do określania kierunku i korekcji kursu robota wykorzystany zostanie magnetometr, a za badanie bliskiego sąsiedztwa robota odpowiedzialne będą sensory ultradźwiękowe. Ostatecznie jednak cały ciężar poprawnego badania pomieszczenia i dokładnego pozycjonowania robota przejęły czujniki ultradźwiękowe. Rozważny dobór sposobu pomiaru i urządzeń pozwolił na skonstruowanie robota przy rozsądnych kosztach. W kilku następnych podrozdziałach opisano poszczególne elementy robota oraz zaimplementowany algorytm.

### **6.1. Konstrukcja mechaniczna**

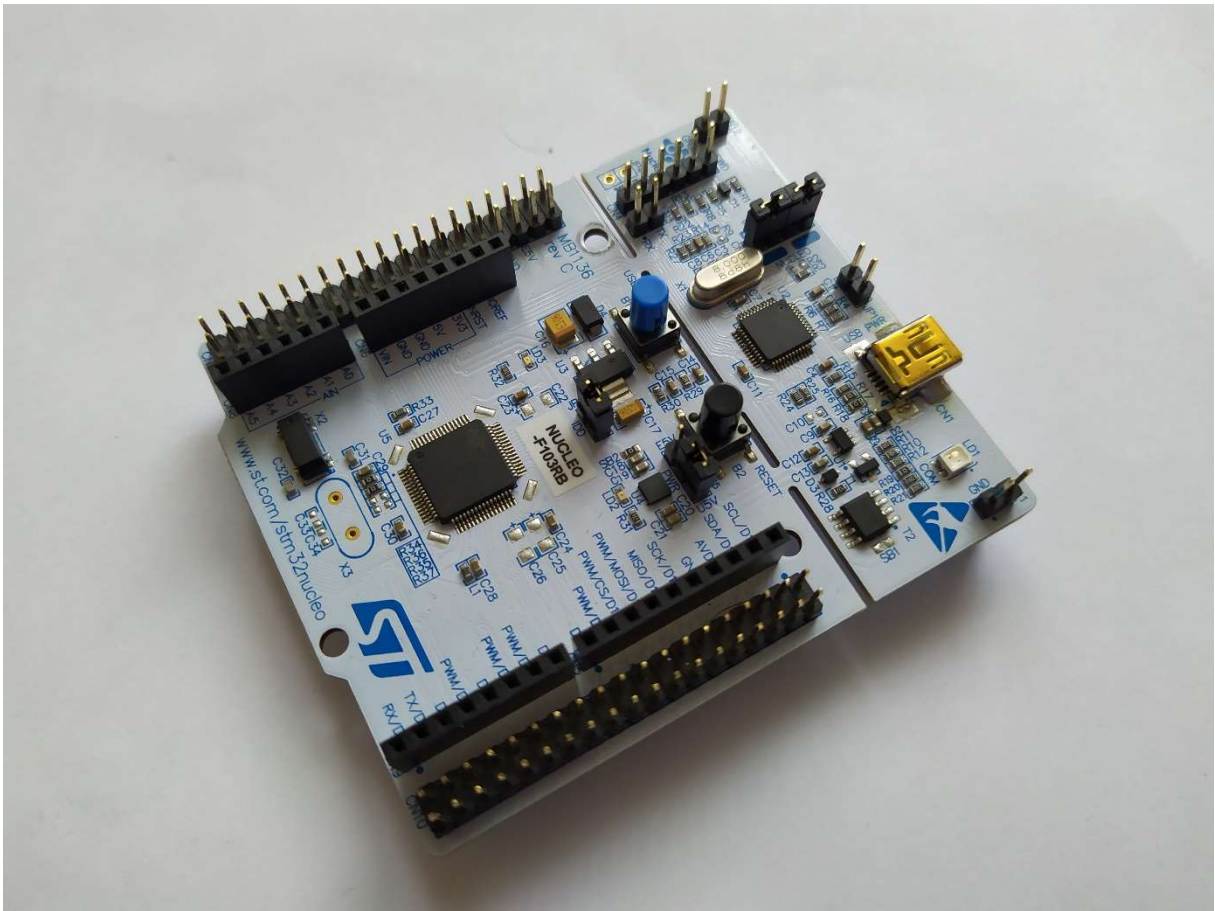
Podwozie robota zostało złożone przy użyciu elementów z zestawu Black Gladiator firmy DFRobot. Szkielet pojazdu stanowi aluminiowa rama, do której przymocowano pozostałe elementy konstrukcji (rys 6.1). Do poruszania się robot wykorzystuje gąsienice napędzane dwoma silnikami prądu stałego. Zarówno gąsienice jak i ich prowadnice wykonane są z tworzywa sztucznego. Zakres napięcia zasilania silników wynosi od 6 V do 12 V. Ich prędkość bez obciążenia wynosi od 170 do 350 obrotów na minutę (silniki bez enkoderów). W czasie pracy bez obciążenia silniki pobierają 100 mA. Nadwozie robota stanowi lekka drewniana płytka odseparowana od podwozia za pomocą tulei dystansowych o długości 25 mm. W płytce znajduje się szereg wycięć i otworów ułatwiających podłączenie wszystkich potrzebnych komponentów elektronicznych.



Rys. 6.1 Podwozie zastosowane przy budowie prototypu

## 6.2. Układ sterujący

Podczas tworzenia prototypu podjęto decyzję o wykorzystaniu zestawu NUCLEO-F103RB (rys. 6.2) z rodziny STM32 Nucleo64. Jest on wyposażony w mikrokontroler STM32F103RBT6 w obudowie LQFP64 [43] [44]. Jego rdzeń to 32-bitowy mikroprocesor ARM Cortex M3 o częstotliwości taktowania do 72MHz. Użyta płytką posiada siedem Timerów, 3 interfejsy USART, dwa przetworniki analogowo-cyfrowe (zarówno 12 jak i 16-kanalowy), dwa przyciski oraz debugger ST-Link/V2. Mikrokontroler wyposażony jest także w dwa interfejsy SPI i I2C. Złącze miniUSB udostępnia nam trzy różne interfejsy: wirtualny port COM, pamięć masową oraz port do programowania i debuggowania. Niezwykle ważną zaletą zestawu jest elastyczny sposób zasilania. Nucleo64 umożliwia podłączenie zasilania w trzech różnych wariantach. Nucleo-F103RB może działać poprzez zasilanie przez złącze USB lub przez zewnętrzne źródło zasilania. W drugim przypadku mamy do wyboru wejście zasilania VIN, tolerujące napięcia od 7V do 12 V (przy maksymalnym obciążeniu prądowym 800mA) oraz wejście zasilania E5V, tolerujące napięcie w zakresie 4,75 – 5,35 V (przy maksymalnym obciążeniu prądowym 500mA). Do zaprogramowania mikrokontrolera posłużono się biblioteką HAL przygotowaną przez firmę STM [45].



Rys. 6.2 Mikrokontroler Nucleo STM32RBT6

### 6.3. Pomiar przyspieszenia i pola magnetycznego

Podczas początkowych prac koncepcyjnych założono, że najważniejszym czujnikiem robota będzie moduł LSM303D produkcji firmy Pololu. Układ zawiera w sobie 3-osiowy akcelerometr oraz magnetometr. Wbudowany regulator napięcia pozwala na zasilenie modułu napięciem z zakresu od 2,5V do 5,5V. LSM303D (rys. 6.3) obsługuje dwie magistrale: I2C oraz SPI (w prototypie wykorzystano I2C). Rozdzielczość akcelerometru i magnetometru wynosi 16 bitów na każdą oś. Układ może pracować w kilku zakresach czułości przyspieszenia/indukcji magnetycznej. Dla akcelerometru są to odpowiednio wartości:  $\pm 2$ ,  $\pm 4$ ,  $\pm 6$ ,  $\pm 8$ ,  $\pm 16$  g. Dla magnetometru: 2,  $\pm 4$ ,  $\pm 8$ ,  $\pm 12$  Gs. Moduł wyposażony jest dodatkowo w zintegrowany czujnik temperatury. Pobór prądu przez moduł LSM303D podczas pracy wynosi jedynie 5mA.



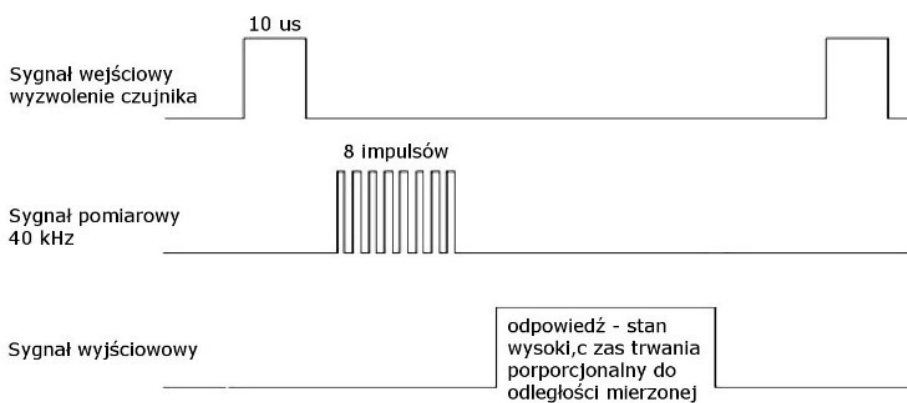
Rys. 6.3 3-osiowy akcelerometr i magnetometr Pololu LSM303D

#### 6.4. Pomiar odległości

Do monitorowania pomieszczenia, w którym znajduje się robot wykorzystano popularne czujniki ultradźwiękowe HC-SR04 i HY-SRF05. Przy doborze elementów odpowiedzialnych za sprawdzanie sąsiedztwa robota głównym czynnikiem była ich przystępność cenowa. Czujnik posiada nieduże wymiary 45 x 20 x 15 mm. Do jego stabilnego zamocowania w robocie użyto dedykowanych uchwytów montażowych. Sensory HC-SR04 pracują w zakresie od 2 cm do 200cm. Ich dużą wadą jest jednak wysokie napięcie zasilania potrzebne do poprawnej pracy, wynoszące 5 V. Ogranicza to możliwość swobodnego podłączenia czujnika do mikrokontrolera i wymusza korzystanie z wyprowadzeń tolerujących napięcie 5 V (ang. 5 V tolerant I/O). Obsługa czujnika polega na wystawieniu stanu wysokiego na odpowiednie wejście przez 10 mikrosekund. Czujnik wysyła falę akustyczną o częstotliwości 40 kHz. Odbita wiązka powraca do sensora. Sygnalizowane jest to stanem wysokim na wyjściu czujnika. Schemat procedury pomiarowej przedstawiono na rysunku 6.5. Na rysunku nr 6.4 przedstawiony został czujnik HC-SR04.



Rys. 6.4 Ultradźwiękowy czujnik odległości HC-SR04



Rys. 6.5 Sposób przeprowadzania pomiaru odległości z wykorzystaniem czujnika odległości [46]

Prędkość rozprzestrzeniania się fali dźwiękowej w powietrzu wynosi 340 m/s. Monitorując czas jaki upłynął od wysłania do odebrania wiązki z łatwością możemy obliczyć dystans przemierzony przez wiązkę zgodnie ze wzorem 8.

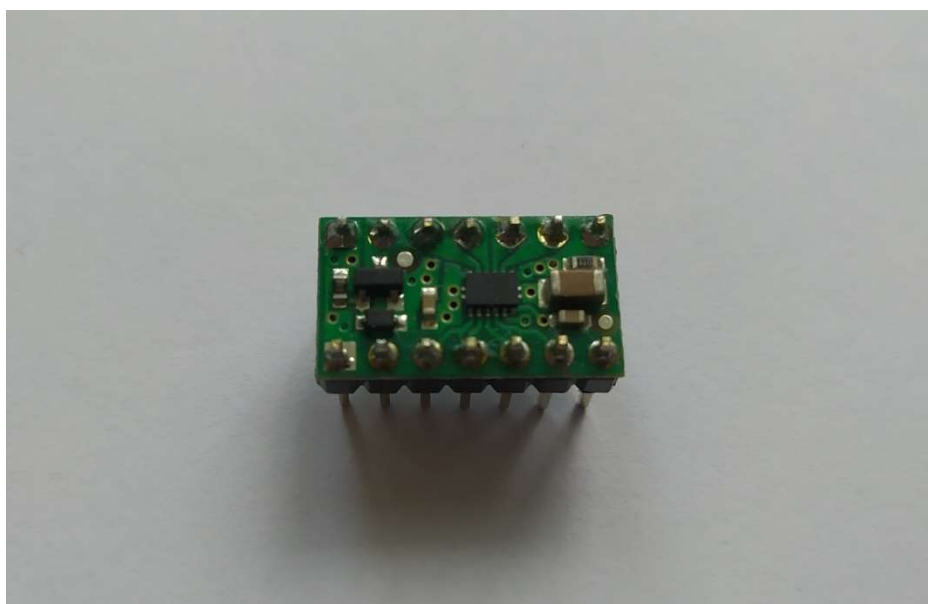
$$Test\_distance = (czas\_stanu\_wysokiego \times prędkość\_dźwięku (340m/s) / 2 \quad (8)$$

Ostatecznie do wyliczenia odległości w programie komputerowym użyto wzoru numer 9.

$$dystans [cm] = (czas\_stanu\_wysokiego [\mu s] * 34) / 1000 / 2 \quad (9)$$

## 6.5. Sterowanie pracą silników

Do sterowania pracą dwóch silników napędzających gąsienice użyto dwukanałowego sterownika silników DRV8835 (rys. 6.6) produkcji firmy Pololu [47]. Układ ten pracuje z napięciem w zakresie od 2 V do 11V oraz prądem na kanał do 1,2 A. Moduł wytrzymuje kilkusekundowe skoki natężenia prądu maksymalnie do 1,5 A. Część logiczna zasilana jest napięciem 3,3 V. Prędkość obrotową silników kontroluje się przy pomocy sygnału PWM o częstotliwości do 250 kHz. Moduł może pracować w dwóch trybach. Pierwszy z nich to prosty tryb PHASE/ENABLE, przy którym prędkość silników jest wprost proporcjonalna do stopnia wypełnienia sygnału PWM. Drugi z nich to zaawansowany tryb IN/IN MODE, który pozwala zarówno na sterowanie wprost proporcjonalne, jak i odwrotnie proporcjonalne do stopnia wypełnienia sygnału PWM. Sterownik posiada zabezpieczenie przed nieprawidłowym podłączeniem zasilania, co dodatkowo chroni konstrukcję elektroniczną robota przed przypadkowymi błędami montażowymi. Przy tworzeniu prototypu wykorzystano tryb PHASE/ENABLE.

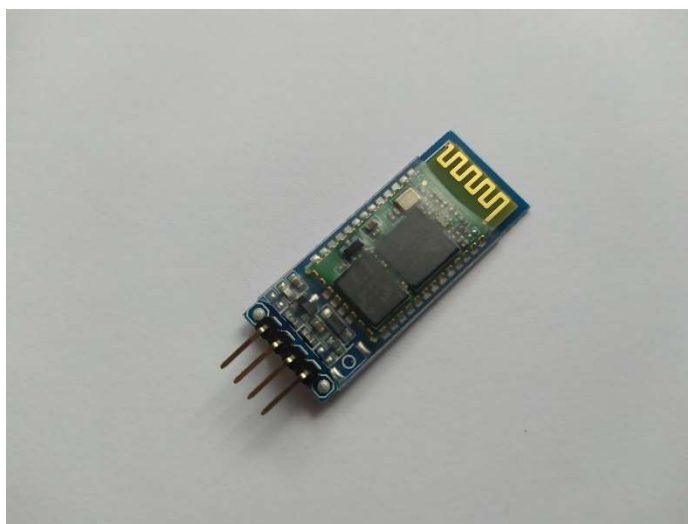


Rys. 6.6 Dwukanałowy sterownik silników DRV8835 (Pololu 2135)



## 6.6. Komunikacja bezprzewodowa

Komunikację bezprzewodową zrealizowano za pomocą modułu Bluetooth HC-06-ZS-040, który przedstawiono na rysunku 6.7. Urządzenie komunikuje się poprzez szeregowy interfejs UART. Posiada ono piny do transmisji i odbioru danych (TX i RX), które pracują z napięciem 3,3 V. Sam moduł może być zasilany napięciem z zakresu 3,6 V do 6 V. Podczas pracy układ pobiera około 50 mA. Nadajnik posiada zasięg do 10 m, a jego moc i czułość wynoszą odpowiednio 4 dBm i 85 dBm. Urządzenie wspiera komendy AT, dzięki czemu istnieje możliwość zmiany domyślnych ustawień modułu. Do wyprowadzenia statusowego urządzenia została podłączona dioda LED, która sygnalizuje status poprawnego połączenia. Moduł pozwala na wygodną komunikację pomiędzy robotem a komputerem [48].



Rys. 6.7 Moduł HC-06-ZS-040 do komunikacji bezprzewodowej, wykorzystujący technologię bluetooth.

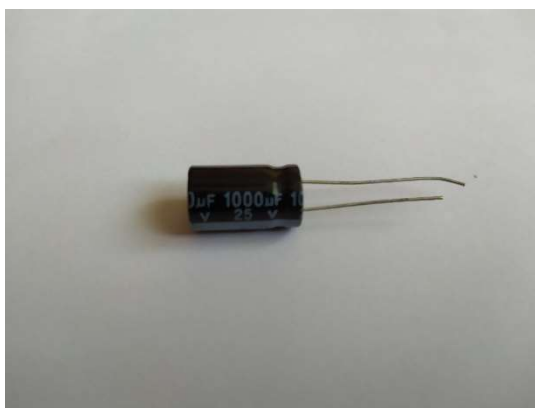
## 6.7. Sekcja zasilania

Sekcja zasilania składa się z dwóch akumulatorów oraz pojedynczego kondensatora elektrolitycznego o wysokiej pojemności. Pojazd zasilany jest z dwóch ogniw litowo – jonowych firmy Xtar o pojemności 2600 mAh zbudowanych w oparciu o oryginalne ogniwo firmy Samsung (rys. 6.8). Posiada on zabezpieczenie przed nadmiernym rozładowaniem, przeładowaniem, przeciążeniem i zwarcie. Użyte ogniwa, choć droższe od standardowych ogniw (o 20%) zabezpiecza przed uszkodzeniami innych elementów elektronicznych użytych przy tworzeniu prototypu. Napięcie nominalne każdego ogniwa wynosi 3,7 V.



Rys. 6.8 Akumulator Li-ion używany w module zasilania

Do stabilizacji układu zasilania użyto pojedynczego kondensatora elektrolitycznego o pojemności 1000  $\mu\text{F}$  i maksymalnym napięciu 25V. Zarówno oba silniki jak i serwomechanizm podłączone zostały do wspólnego źródła zasilania co mikrokontroler. Wymusza to zastosowanie filtracji zasilania w celu eliminacji zakłóceń i stabilizacji działania mikrokontrolera. Zastosowanie osobnego źródła zasilania dla każdego modułu niepotrzebnie zwiększyłoby masę oraz stopień skomplikowania prototypu.



Rys. 6.9 Kondensator elektrolityczny użyty do stabilizacji napięcia

## 6.8. Implementacja algorytmu

Ostateczną koncepcją algorytmu przyjętą w projekcie jest pomiarowanie i badanie pomieszczenia poprzez przejazd robota wzdłuż ścian. Na algorytm składa się sterowanie silnikami i korekta kursu robota, badanie bezpośredniego sąsiedztwa pojazdu czujnikami odległości oraz wyznaczanie osi kierunku, a także monitorowanie położenia robota w kartezjańskim układzie współrzędnych i aktualizacja macierzy zajętości w czasie rzeczywistym.

### **6.8.1. Przejazd wzdłuż ścian z wykorzystaniem czujników odległości**

W ostatecznej (alternatywnej) wersji algorytmu główną rolę przy pomiarowaniu spełnia odometria, poprzez zliczanie wykonanych ruchów w daną stronę. Pojazd stale monitoruje kierunek, w którym się porusza. Jadąc wzdłuż ściany, w czasie rzeczywistym modyfikuje macierz zajętości. Po każdym ruchu na bieżąco aktualizuje także swoją obecną pozycję. Do orientacji w przestrzeni robot wykorzystuje czujniki odległości HC-SR04 i HY-SRF05. Przedni sensor sprawdza, czy przed pojazdem znajduje się przeszkoda i czy istnieje możliwość jazdy do przodu. Dwa boczne czujniki mają na celu monitorowanie otoczenia po prawej stronie pojazdu. Za ich pomocą dokonujemy sprawdzenia, czy istnieje możliwość skrętu w prawo, a także określamy odpowiedni kurs pojazdu. Po sprawdzeniu, czy przed robotem nie ma przeszkody robot przesuwa się do przodu o odległość 20 cm. Zastosowanie gąsienic i dopasowanie pracy ich prowadnic oraz ograniczenie maksymalnych prędkości pozwoliło uniknąć niepożądanego poślizgu.

### **6.8.2. Określenie kierunku i położenia robota przy ścianie**

Do określania kierunku robota wykorzystano założenie, że pojazd może poruszać się w czterech kierunkach (północ/góra, południe/dół, wschód/prawo, zachód/lewo). Przeważająca większość pomieszczeń posiada proste, regularne kształty, a ściany w nich, nachylone są do siebie pod kątem 90 stopni. Dzięki temu robot sprawdza się przy pomiarowaniu większości wnętrza budynków. Bardzo ważnym mechanizmem, który zaimplementowano w celu bezbłędnego skręcania pojazdu jest system kontroli odległości od elewacji przed i w czasie obrotu robota. Wyznaczenie pozycji robota określa się w oparciu o kartezjański układ współrzędnych. Współrzędne w projekcie traktowane są jako para liczb całkowitych.

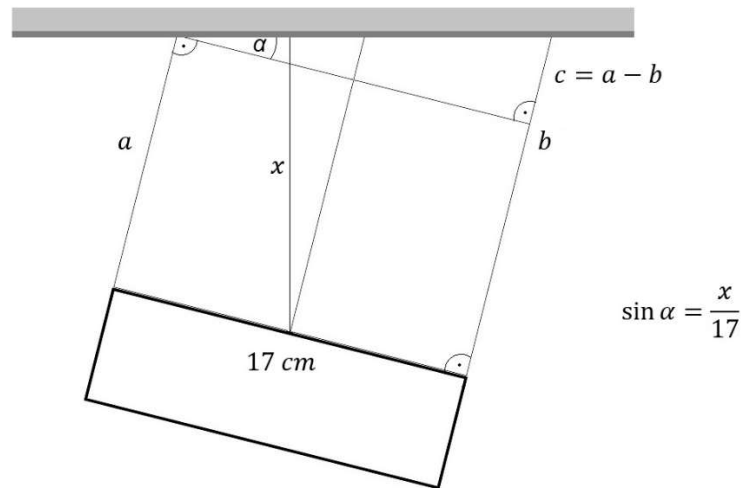
Początkowo określanie pozycji robota miało odbywać się z wykorzystaniem odczytów z frontowego czujnika odległości HY-SRF05. Zaplanowana początkowo procedura wyglądała następująco. W momencie rozpoczęcia jazdy odczytujemy odległość z przedniego czujnika. Dokonujemy jazdy prosto do momentu, w który mamy możliwość wykonania skrętu w lewo lub prawo. Wykonujemy skręt, a następnie pomiar odległości. Powtarzamy czynności aż do znalezienia się w tym samym punkcie, względnie zliczenia podobnych odległości w kierunkach północ-południe oraz wschód-zachód. Niestety przy natrafieniu na załamania, filary itp. czujnik ultradźwiękowy nie jest w stanie dostarczyć precyzyjnych wskazań odległości. Badanie możliwości

pozycjonowania z wykorzystaniem czujników odległości nie przyniosło oczekiwanych rezultatów, dlatego do określania pozycji robota wykorzystano metodę wykorzystującą zliczanie cykli jazdy w danym kierunku. Robota zaprojektowano w taki sposób, aby nie poruszał się ruchem jednostajnym, lecz badał pomieszczenie etapami. Robot porusza się do przodu fazami, w każdym z etapów przejeżdżając odległość 20cm. Ze względu na zastosowaną konstrukcję i napęd gąsienicowy robot nie wykorzystuje enkoderów, a wartość 20 cm została wyznaczona empirycznie poprzez dobranie odpowiednich parametrów pracy silników.

### **6.8.3. Kontrola jazdy i korekta kierunku**

Jednym z kluczowych mechanizmów korygujących jazdę jest system kontroli położenia robota względem ściany, wzdłuż której porusza się robot.

Podczas tworzenia prototypu eksperymentalnie dobrano odpowiednią wartość wypełnienia sygnału PWM dostarczanego do sterownika silników. Jest ona dobrana w taki sposób, aby robot bez problemu ruszał z miejsca, ale prędkość gąsienic nie była zbyt duża, ponieważ zbyt szybkie przesuwanie gąsienic na prowadnicach powoduje niepożądane poślizgi. Prędkość lewego silnika jest o około 20% większa niż silnika prawego. Wartość tę dobrano eksperymentalnie. Podejście takie pozwala na stabilną jazdę wzdłuż elewacji. Pojazd nie odchyła się znacząco od oczekiwanego kierunku jazdy. Podczas poruszania się na wprost robot nieustannie koryguje swój kurs względem badanej elewacji. Korekta ta polega na sprawdzeniu kąta nachylenia oraz odległości środka robota do ściany. Do tego celu wykorzystano dwa czujniki umieszczone po prawej stronie robota rozlokowane 17 cm od siebie. Jeżeli odległość do wykrytej przeszkody od prawego czujnika przedniego jest mniejsza niż 20 cm, a kąt jest większy niż 45 stopni, w celu natychmiastowego wyrównania kierunku jazdy uruchamiany jest mechanizm wyrównania kąta i przyjęcia minimalnej odległości (rys. 6.10). Wpływ tego systemu na pomiar przemierzonych odległości jest znikomy i nie zakłóca pomiarów.



Rys. 6.10 Określanie pozycji robota względem sąsiadującej z robotem ściany

Przed wykonaniem skrętu w prawo pojazd dokonuje odczytu z prawego czujnika odległości. Odczyt ten traktowany jest jako wartość referencyjna. Robot obraca się następnie dopóki odczyt z przedniego czujnika odległości nie będzie taki sam jak odległość referencyjna. W celu eliminacji błędów przy obrocie zastosowano tolerancję 8 cm. Ten sam schemat realizowany jest dla skrętu w lewo, z tą różnicą, że wartość referencyjna pobierana jest z czujnika przedniego, a odległość aktualną odczytujemy z czujnika prawego. Dodatkowo po wykonanym skręcie robot przeprowadza kontrolę równoległego położenia względem ściany. Pojazd dokonuje sprawdzenia czy nachylenie robota do ściany jest nie większe niż 4 stopnie. W przeciwnym wypadku robot dokonuje dalszej korekty, aby robot ustawił się równolegle do ściany.

Tuż po obrocie w prawo czujniki odległości wykrywają wolną przestrzeń po jego prawej stronie. Aby robot nie zawrócił zastosowano w tym przypadku dodatkowy mechanizm protekcji trasy i dodatkowy licznik (jeżeli robot nie napotyka wolnej przestrzeni po swojej prawej stronie licznik automatycznie przywracany jest do stanu początkowego). Jeżeli obszar przed robotem i po jego prawej stronie jest wolny, wówczas licznik jest inkrementowany a robot porusza się do przodu. Jeżeli po tym ruchu sytuacja się powtarza robot ponownie wykonuje ruch i inkrementuje licznik. Dopiero po tych czynnościach robot może ponownie skręcić w prawo. System taki zapobiega zawracaniu robota i jego ciągłemu obracaniu się wokół własnej osi tuż po skręcie, a dwukrotne przejechanie w przód chroni także przed błędnymi wyliczeniami kątów i odległości z sensorów ultradźwiękowych. Jeżeli robot nie napotka wolnej przestrzeni licznik przywracany jest do stanu początkowego.

Dodatkowym zabezpieczeniem przed wpadnięciem robota na przeszkodę (co mogłoby spowodować błędy w pomiarach lub całkowite zablokowanie się) jest kontrola odległości do przeszkody wykonywana przed skrętem. W przypadku zbyt bliskiej odległości do przeszkody robot wykonuje manewr cofnięcia się o 5 cm.

#### **6.8.4. Warunek zakończenia pomiarów i korekta współrzędnych**

Istotnym mechanizmem zaprojektowanym do korekty błędów i niedokładności przy pomiarowaniu pomieszczeń jest korekta błędów przy wyznaczaniu pozycji robota i korekta jego współrzędnych. Robot kończy prace po przejechaniu trasy wzdłuż ścian pomieszczenia. Zakończenie działania pracy pojazdu uwarunkowane jest znalezieniem się w tym samym miejscu, w którym rozpoczęto pracę. Dodatkowo aktualny kierunek robota musi pokrywać się z kierunkiem początkowym. W zaimplementowanym algorytmie kierunek początkowo zawsze przyjmowany jest jako północ (góra). Istnieje niebezpieczeństwo, że przy licznych skrętach i niekorzystnym ułożeniu ścian i przeszkód robot błędnie określi swoje współrzędne. W celu zabezpieczenia się przed taką sytuacją warunek zakończenia pracy obudowano dodatkowym sprawdzeniem, czy różnica początkowej współrzędnej jej aktualnej wartości nie jest większa lub równa jeden (dotyczy obu współrzędnych X i Y w kartezjańskim układzie współrzędnych).

#### **6.8.5. Opis procesu poprawy i modyfikacji zaimplementowanych algorytmów**

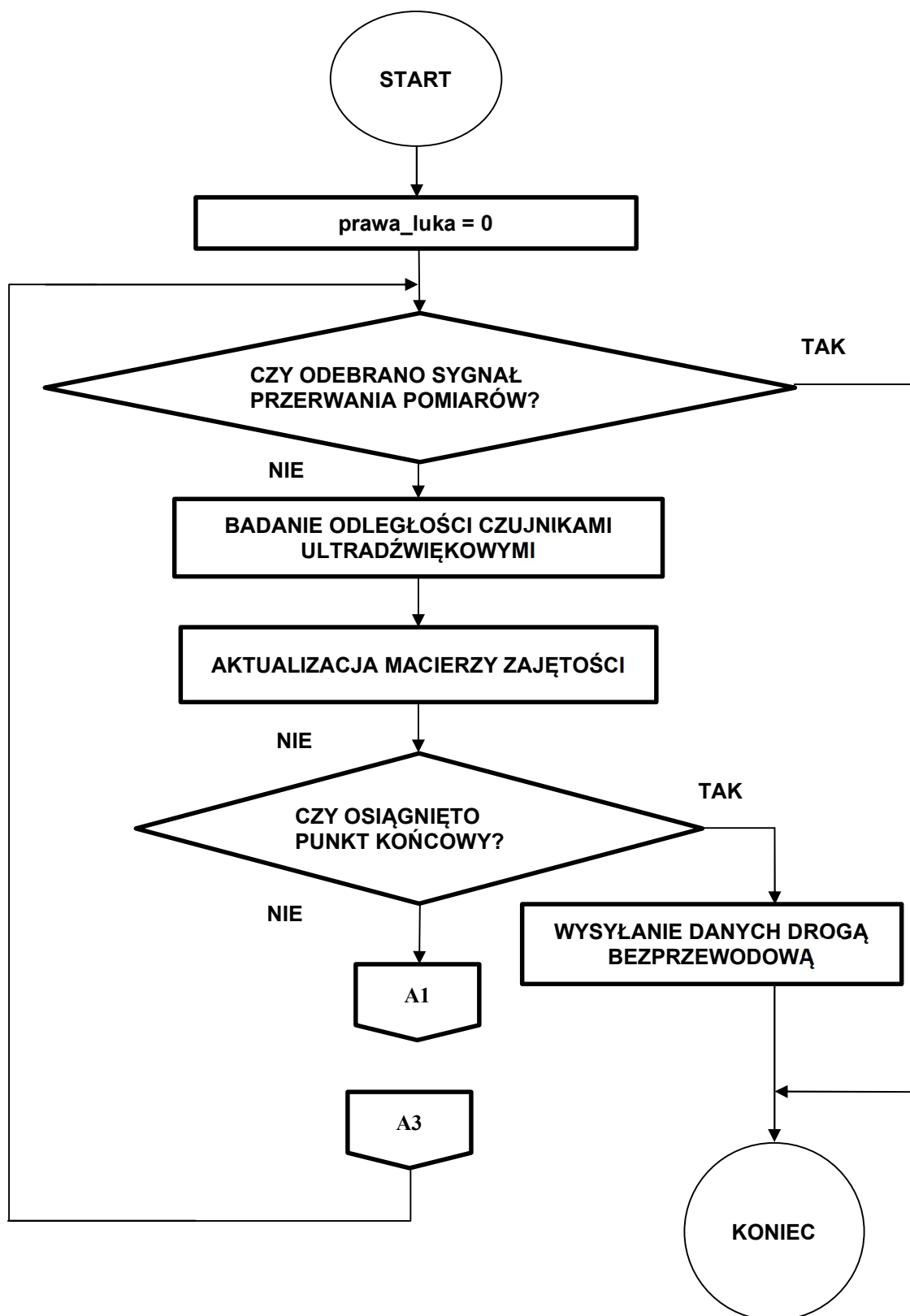
Procesowi tworzenia prototypu towarzyszyło ciągle przeprowadzanie testów zaimplementowanych algorytmów i poprawności ich działania. Badania takie były konieczne do osiągnięcia jak najwyższej precyzji. Testy przeprowadzano w taki sposób, aby sprawdzać zachowanie robota w warunkach brzegowych. Pierwszą próbą była sytuacja, w której robot otoczony jest ze wszystkich stron znajdującą się w niewielkiej odległości przeszkodą. Robot czterokrotnie powinien odnotować przeszkodę i brak możliwości skrętu w prawo lub jazdy na wprost, po czym powinien obrócić się w lewo. Czterokrotne powtórzenie tej czynności kończy pomiary, a ich wynikiem jest odnotowanie w macierzy zajętości pojedynczej komórki otoczonej przez cztery komórki oznaczone jako przeszkoda. Testy te pokazały, że korekta błędów przy skręcaniu w lewo jest niewystarczająca. Pojazd przy skręcie początkowo korygował swoją pozycję do momentu, w którym kąt nachylenia do ściany był nie większy niż 15 stopni. Kąt ten okazał się zbyt duży, a dalsze eksperymenty pokazały, że do

osiągnięcia wymaganej precyzji potrzeba ograniczenia wartości progowej poniżej 8 stopni. Kolejnym testem była seria przejazdów robota w przygotowanym korytarzu o szerokości około 45cm. Podczas prób okazało się, że zaproponowana korekta błędu dotycząca korekty odchylenia nie spełnia w wystarczającym stopniu swoich założeń. Pojazd podczas jazdy zachowywał się bardzo dużą niestabilnością jeżeli chodzi o nachylenie do ściany podczas jazdy (miejscami nachylenie wynosiło nawet ok. 45 stopni). Aby poprawić efektywność mechanizmu korekcji błędów podczas jazdy przy ścianie wprowadzono serię poprawek i ulepszeń. Zmniejszono minimalną odległość wymaganą podczas jazdy przy ścianie. W konsekwencji zredukowano także dystans wykrycia ściany przed robotem. Powstało jednak niebezpieczeństwo, że robot może zahaczyć o przeszkodę, która znajdzie się zbyt blisko, co zniekształci pomiary, a w najgorszym przypadku całkowicie zablokuje pracę pojazdu. Spowodowało to jednak konieczność wprowadzenia mechanizmu wycofującego robota o 5 – 10 cm jeżeli znajduje się on niemal na przeszkodzie. Na koniec zmniejszono graniczne wartości kąta, przy której następuje korekta kursu co również znacząco wpłynęło na płynność jazdy. Wszystkie zastosowane techniki pozwoliły na znaczące ustabilizowanie toru jazdy robota co ma kluczowe znaczenie przy określaniu wartości danej komórki w macierzy zajętości. Żeby dobrze odwzorować pomieszczenie robot musi zachowywać się w sposób przewidywalny i stabilny. Kolejnym etapem było sprawdzenie poprawności działania robota na w teście, który wymaga zarówno skrętu w lewo jak i w prawo. Podczas próby robot bardzo dobrze poradził sobie z odwzorowaniem badanego obszaru, a ulepszenia zaimplementowane do skrętu w lewo oraz ich odpowiednik do skrętu w prawo działały bez zarzutu. Po wszystkich wymienionych testach przyszedł czas na przetestowanie robota w prawdziwych i realnych pomieszczeniach. Robot został sprawdzony w kilku pomieszczeniach. Zarówno tych prostokątnych, jak i tych nieregularnych, wymagających skrętów zarówno w lewo jak i prawo. Podczas testów zmodyfikowana jeszcze wartość graniczną kąta sprawdzenia nachylenia do ściany po skręcie (dwukrotne zmniejszenie wartości). Po zmianie kąta powtórzono całą serię testów od samego początku. Zachowanie robota dla wszystkich wcześniej wymienionych testów okazało się równie skuteczne jak poprzednio, a w testach w warunkach rzeczywistych na większym obszarze dało w rezultacie większą stabilność toru jazdy.

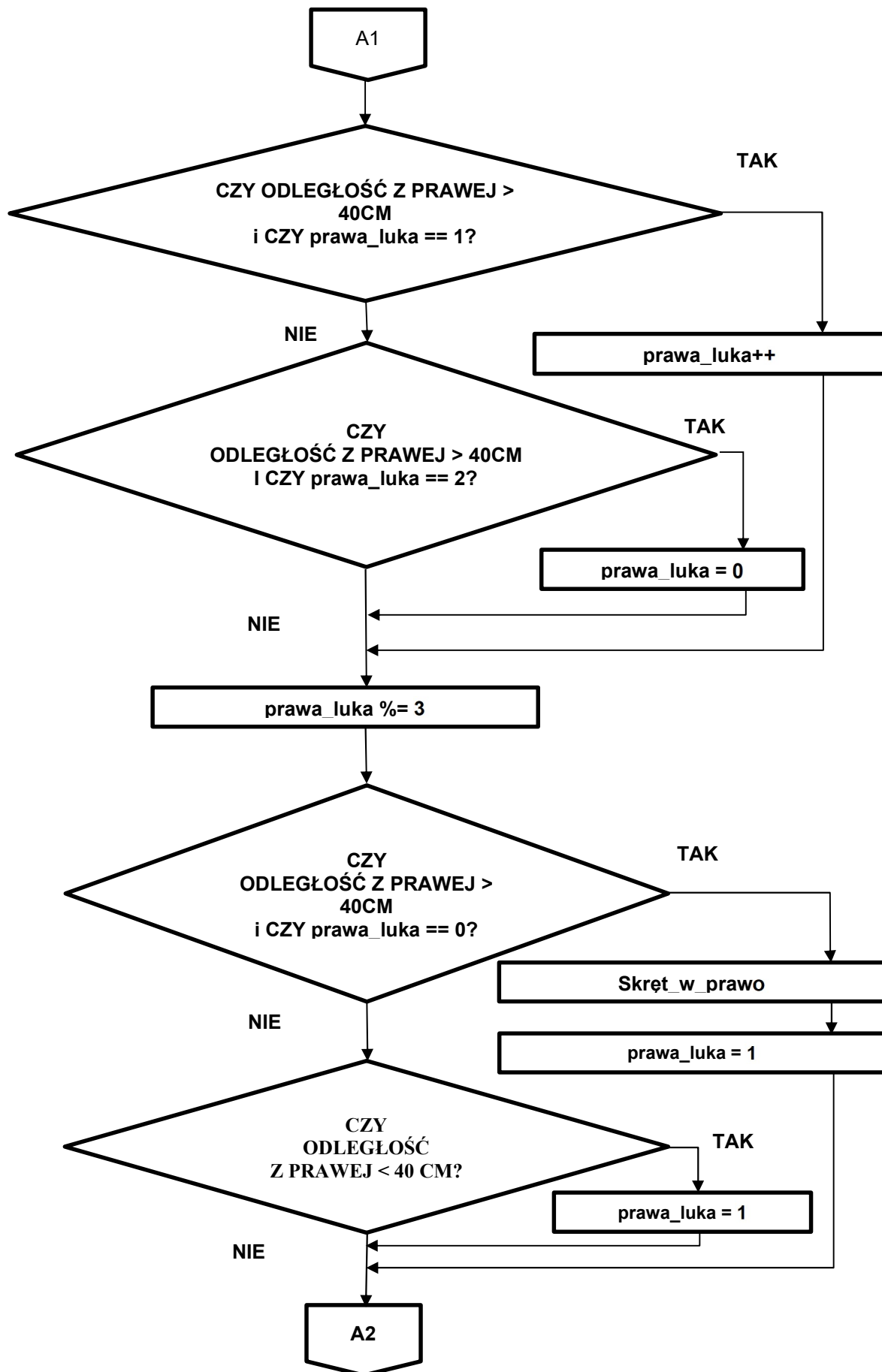
### **6.8.6. Schemat blokowy algorytmu**

W niniejszym podrozdziale przedstawiony został schemat blokowy ostatecznie zaimplementowanego algorytmu. Rysunki 6.11, 6.12 oraz 6.13 przedstawiają bardzo uproszczony algorytm pracy robota. Schemat podzielony został na trzy części w celu poprawienia jego czytelności. Zmienna `prawa_luka` prezentuje licznik, który stanowi zabezpieczenie przed wykonywaniem ponownego skrętu zaraz po wykryciu luki z prawej strony robota. Wykonanie skrętu po dwukrotnym wykryciu wyłomu z prawej strony zabezpiecza także przed wykonaniem przejazdu w zbyt ciasnym miejscu. Gwarantuje także bezpieczeństwo w razie przypadkowego błędnego odczytu z czujników odległości.

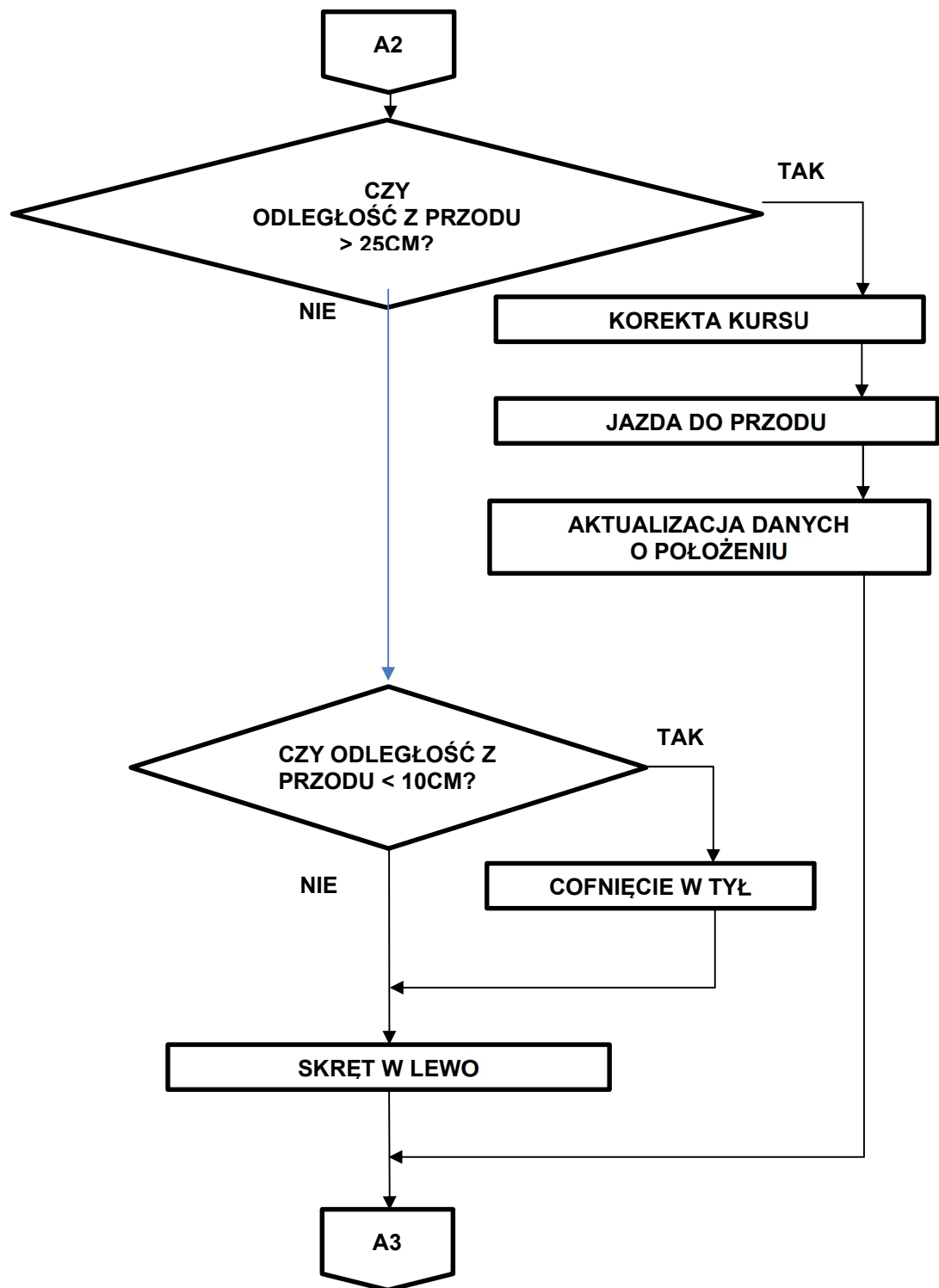




Rys. 6. 11 Schemat blokowy algorytmu - część pierwsza



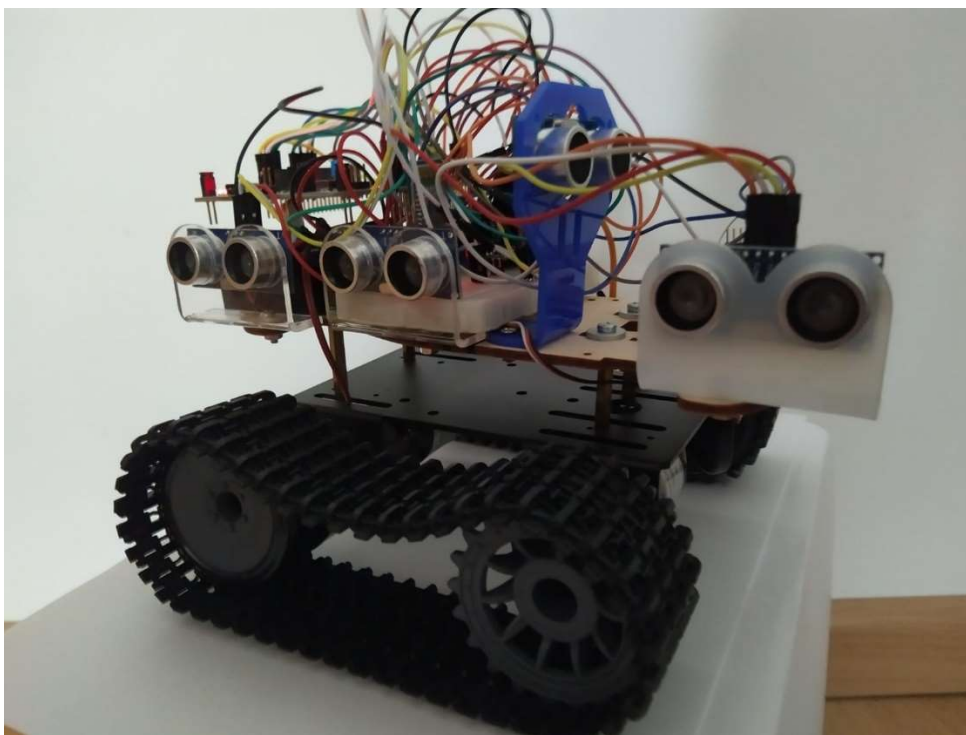
Rys. 6. 12 Schemat blokowy algorytmu - część druga



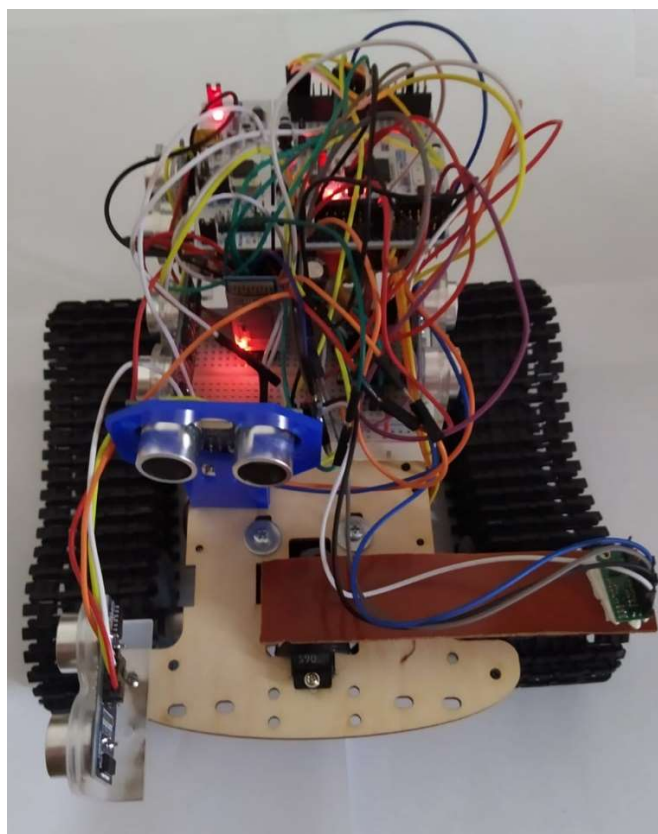
Rys. 6. 13 Schemat blokowy algorytmu - część trzecia

## 6.9 Podsumowanie finalnej wersja prototypu

Początkowa koncepcja alternatywnej pracy robota obejmowała dokonywanie pomiarów przed i po każdym skręcie (zarówno w lewo jak i prawo). Z tego powodu czujnik odległości HC-SR04, który znajdował się na przodzie robota zamieniono na bardziej efektywny czujnik HY-SRF05. Charakteryzuje się on lepszymi parametrami pomiarów, ponieważ potrafi wykrywać przeszkody w zakresie od 2 do 400 cm. W ostatecznej koncepcji zmodyfikowano metodę pozycjonowania, ponieważ początkowo przyjęta koncepcja okazała się nieskuteczna. Z tego powodu, zamiast pomiarów przed i po skręcie zastosowano metodę wykorzystującą odometrię (zliczanie cykli przejazdów w danym kierunku). Dodatkowo należało ulepszyć metodę badania przeszkód po prawej stronie robota. W prototypie przyjęto, że robot będzie poruszał się w bezpośrednim sąsiedztwie ściany i będzie badał położenie przeszkód z przodu i po swojej prawej stronie. W celu dokładnego określania odległości od ściany i kąta odchylenia robota, na prawym boku zamontowano dwa czujniki HC-SR04, w odległości 17 cm od siebie. Obszar w środku pomieszczenia w ostatecznie zaimplementowanym algorytmie nie był badany ze względu na opisane we wcześniejszej zakłócenia elektromagnetyczne. Robot po zakończonych pomiarach wysyła zebrane dane drogą bezprzewodową. Dodatkowym atutem robota jest możliwość zdalnego sterowania robotem za pomocą łączności bluetooth. Ostateczną konstrukcję robota przedstawiono na rysunku nr 7.1 oraz 7.2..



Rys. 7.1 Widok boczny na ostateczną wersję prototypu



Rys 7.2 Widok z góry na ostateczną wersję prototypu

## 7. PERYFERIA I SPIS WYPROWADZEŃ

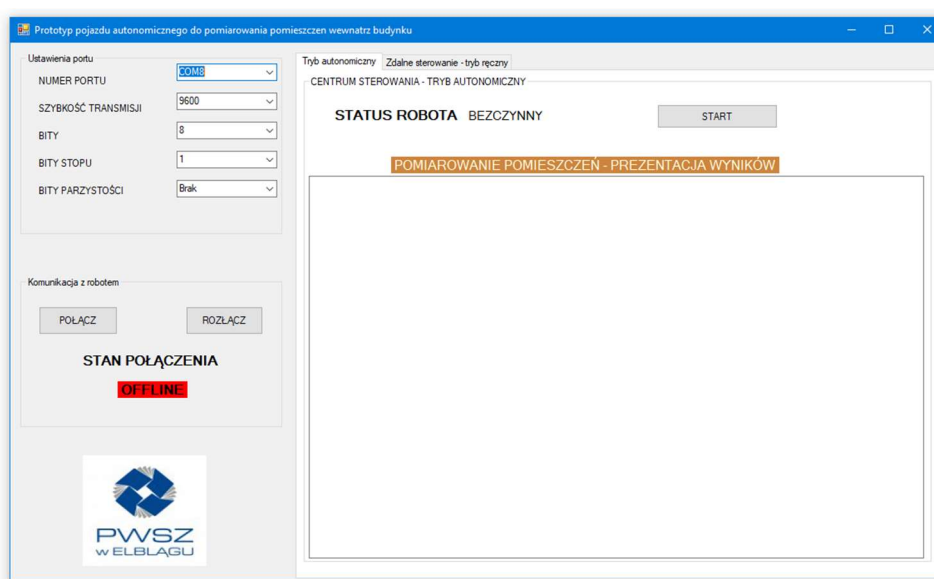
Wszystkie czujniki, moduły i urządzenia podłączone do mikrokontrolera sterującego robotem można podzielić na sześć grup. Wśród nich najważniejszy jest zestaw pinów odpowiedzialnych za kontrolę nad prędkością i kierunkiem jazdy robota (sterowanie), badanie otoczenia robota (pomiar odległości) oraz komunikację pomiędzy robotem a komputerem z aplikacją sterującą robotem (transfer danych). W tabeli nr 7.1 przedstawiono szczegółowy opis wszystkich wyprowadzeń, ich przeznaczenia oraz grupy (zastosowania), do której należą.

Tab. 7.1 Spis wyprowadzeń

PIN	WYPROWADZENIE	ZASTOSOWANIE
PB8	KIERUNEK OBROTU SILNIKA 1	STEROWANIE
PC9	KIERUNEK OBROTU SILNIKA2	
PC0	PWM SILNIKA 1	
PC1	PWM SILNIKA 2	
PC6	PIN WYZYWAJĄCY CZUJNIKA 1	POMIAR ODLEGŁOŚCI
PC8	PIN WYZYWAJĄCY CZUJNIKA 2	
PC10	PIN WYZYWAJĄCY CZUJNIKA 3	
PB12	PIN WYZYWAJĄCY CZUJNIKA 4	
PC7	PIN ECHO CZUJNIKA 1	
PC9	PIN ECHO CZUJNIKA 2	
PC11	PIN ECHO CZUJNIKA 3	
PC12	PIN ECHO CZUJNIKA 4	
PC13	PRZYCISK UŻYTKOWNIKA	ODBLOKOWANIE ROBOTA
PA6	SERVO PWM	DODATKOWY SERWOMECHANIZM
PA9	UART1_RX	TRANSFER DANYCH
PA10	UART1_TX	
PA2	UART2_TX	
PA3	UART2_RX	
PB6	I2C SCL	PRZYSPIESZENIE / POLE MAGNETYCZNE
PB7	I2C SDA	

## 8. TESTY ROBOTA I ZAPROJEKTOWANEGO ALGORYTMU W WARUNKACH RZECZYWISTYCH

Do sterowania prototypem stworzono aplikację napisaną w środowisk .NET z wykorzystaniem języka C# oraz technologii Windows Forms. Przy projektowaniu zachowania robota przewidziano dwa główne tryby działania: tryb automatyczny oraz dodatkowy tryb ręcznego sterowania. Po uruchomieniu aplikacji konieczne jest nawiązanie połączenia z robotem. Dodatkowo po włączeniu zasilania robot wymaga uzbrojenia, poprzez wciśnięcie przycisku na mikrokontrolerze. Stan początkowy aplikacji przedstawiono na rysunku 8.1.

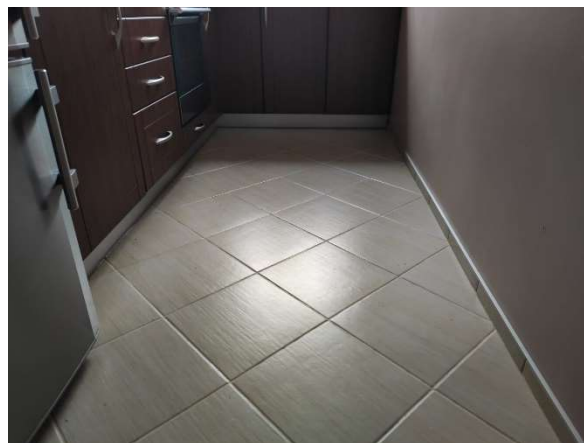
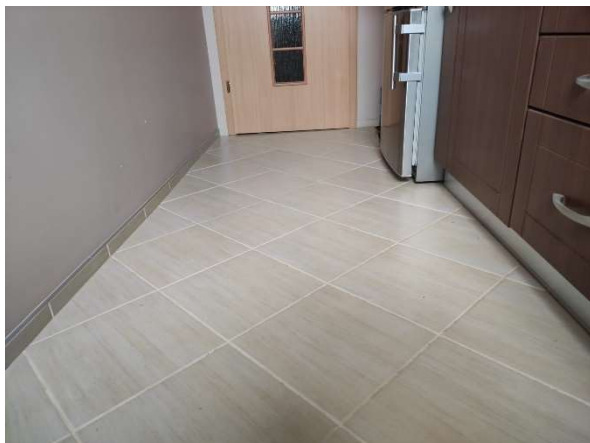


Rys. 8.1 Aplikacja do sterowania robotem – stan bezpośrednio po uruchomieniu

Najważniejszy tryb pracy robota to tryb autonomicznego badania i opomiarowania pomieszczeń. W trybie autonomicznej pracy użytkownik wydaje robotowi polecenie rozpoczęcia wykonania pomiarów, a następnie czeka aż robot zbada całe pomieszczenie i zakończy pomiary. Po zakończonych pomiarach robot drogą bezprzewodową przesyła stworzoną macierz zajętości do komputera. Macierz przedstawiana jest następnie w aplikacji. Użytkownik może w każdej chwili przerwać wykonywane badania. Po zakończeniu pomiarów użytkownik może przenieść robota w inne miejsce i uruchomić pomiary bez konieczności ponownego uruchamiania aplikacji. Prezentacja wyników przedstawiona jest jako macierz komórek, które robot generuje na podstawie przejazdu podczas autonomicznej jazdy. Wszystkie komórki macierzy początkowo przyjmują wartość '\_'. W miejscach, w których robot podczas

pomiarów w trybie autonomicznym wykrył przeszkodę w macierzy zapisywana jest wartość 'X'. Pozwala to na wygodną i szybką prezentację wyników.

Na rysunku 8.2 przedstawiono pomieszczenie, które posłużyło do badania pracy robota w warunkach rzeczywistych. Rysunek 8.3 zawiera zdjęcia wykonane podczas przejazdu robota w trybie autonomicznym. Rysunek 8.4 przedstawia rezultaty pomiarów wykonanych przez robota. Doskonale widać na nim, że zaimplementowany algorytm pozwala na poprawne odwzorowanie kształtów i wymiarów pomieszczenia.

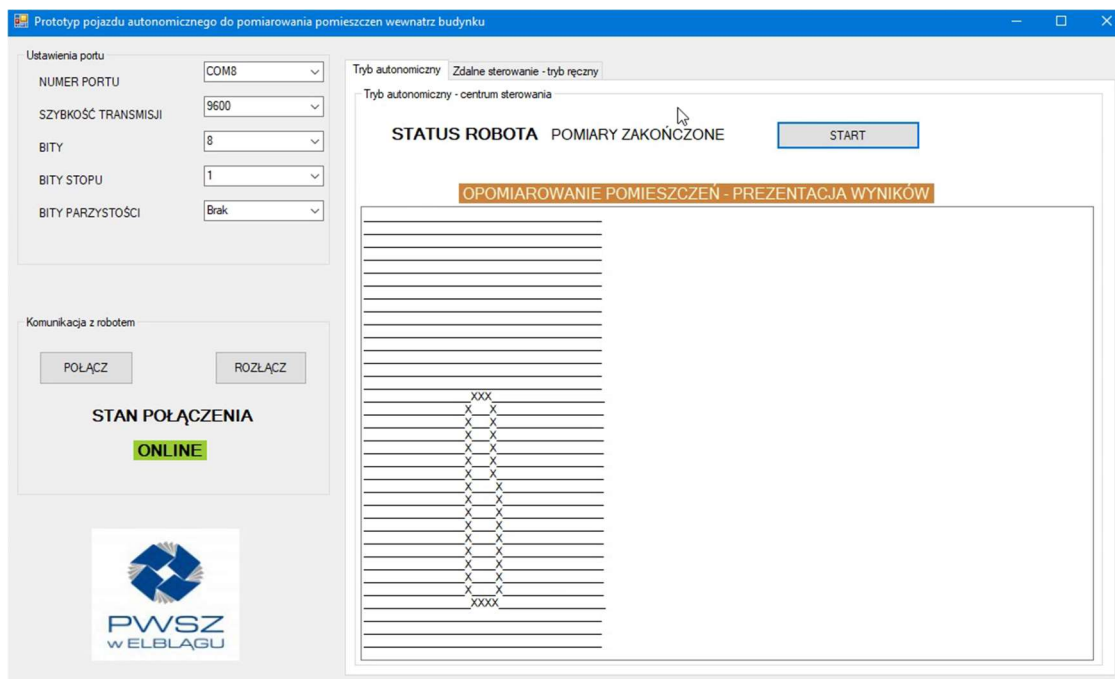


Rys. 8.2 Widok na pomieszczenie testowe nr 1





Rys. 8.3 Robot w czasie testów rzeczywistych – badania w trybie autonomicznym

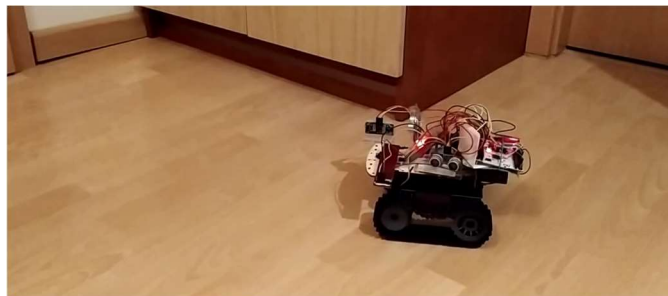
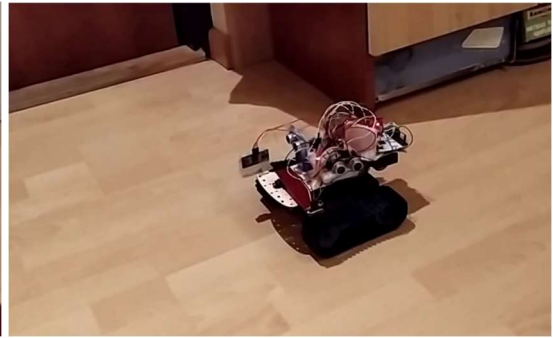
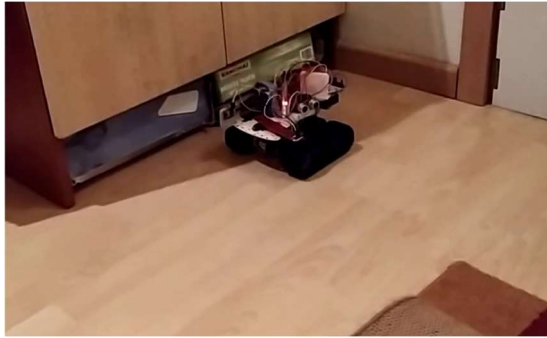


Rys. 8.4 Rezultat badania pomieszczenia nr 1

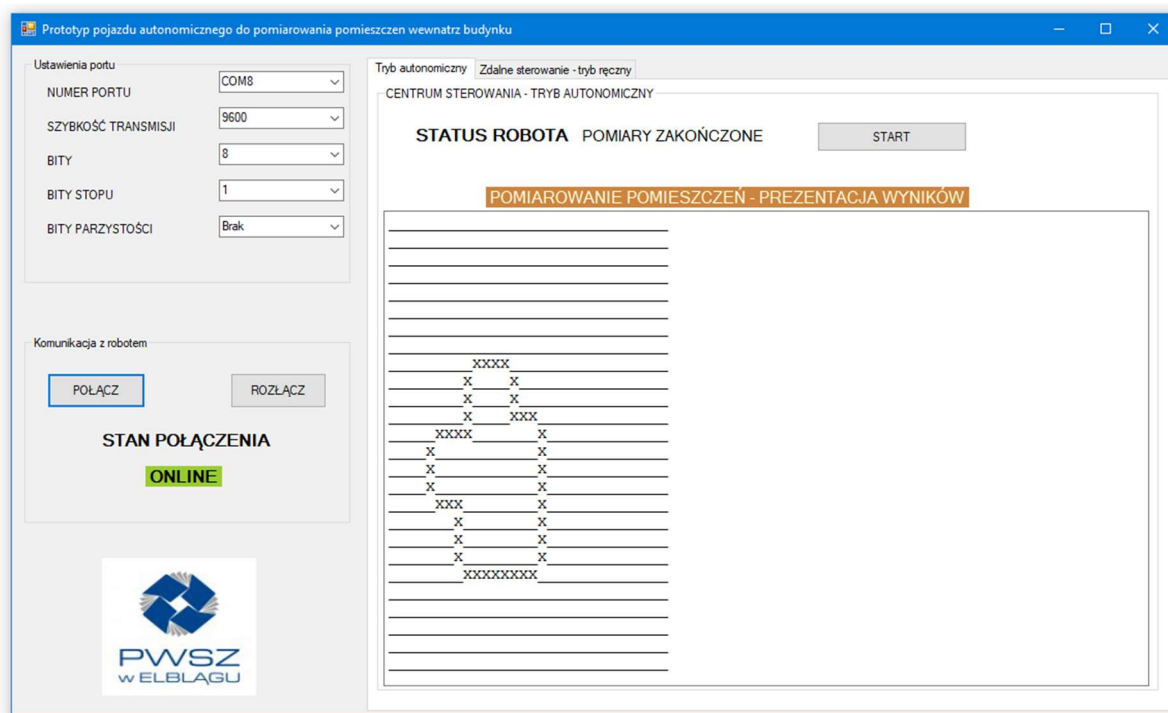
Kolejnym etapem było sprawdzenie poprawności działania prototypu w bardziej wymagających warunkach. Układ kolejnego pomieszczenia (pomieszczenie testowe nr 2), w którym testowany był robot przedstawia rysunek 8.5. Biorąc pod uwagę zaimplementowany system wyznaczania kierunku i pozycjonowania pojazdu test ten można uznać za test w warunkach ekstremalnych. Badanie to wymagało precyzyjnego manewrowania i wykonania aż 10 skrętów na dość niewielkim obszarze (zarówno w lewo jak i prawo). Miejscami działanie mechanizmów korekcyjnych testowano więc na bardzo krótkiej odległości. Warunki pracy robota były więc niezwykle wymagające, ale dzięki temu pozwoliły na dokładne sprawdzenie poprawności działania całego systemu oraz wszystkich zastosowanych mechanizmów korygujących. Rysunek 8.6 obrazuje poszczególne etapy badania pomieszczenia. Na rysunku 8.7 przedstawiono rezultaty pomiarów. Widzimy na nim, że robot bardzo dobrze poradził sobie nawet w wymagającej sytuacji. Stworzona mapa doskonale odzwierciedla rzeczywiste kształty i wymiary badanego obszaru. Warto zwrócić uwagę, że podczas badania robot wykonywał zarówno skręty w lewo jak i prawo, co pozwoliło na dokładne sprawdzenie zaimplementowanych metod korekty skrętu. Test obrazuje także, że obmyślona koncepcja pozycjonowania robota działa poprawnie i zgodnie z założeniami.



Rys. 8.5 Układ pomieszczenia testowego numer 2



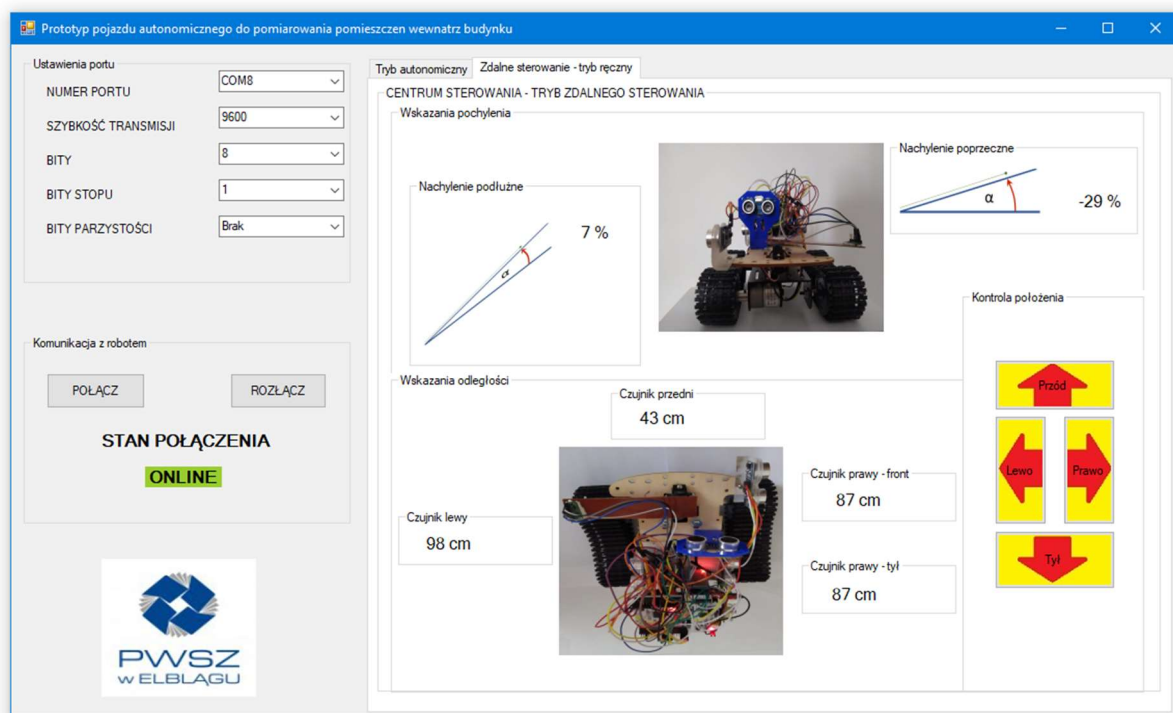
Rys. 8.6 Kolejne fazy pracy robota w pomieszczeniu testowym nr 2



Rys. 8.7 Rezultat badania pomieszczenia nr 2

Przeprowadzone testy robota wypadły pomyślnie, a otrzymane mapy bardzo dobrze odzwierciedlają kształt i wymiary pomieszczenia. Robot sprawdza się nawet w niesprzyjających warunkach, gdzie wymagana jest duża precyzja jazdy i zachodzi konieczność wykonania licznych skrętów. Podczas testów robota pojawiła się tylko jedna niedokładność w drugim teście, ale miało to miejsce w bardzo ciasnym miejscu. Niedokładność w pomiarowaniu wyniosła zaledwie 20cm co jest doskonałym wynikiem biorąc pod uwagę trudny kształt pomieszczenia. Wykonane badania w pełni potwierdziły poprawność zaimplementowanego algorytmu oraz wymyślonych mechanizmów korekcyjnych. Udało się więc potwierdzić, że robot spełnia wszystkie postawione mu zadania, a uzyskane wyniki były niezwykle zadowalające.

Dodatkowym trybem obsługi robota jest tryb zdalnego sterowania. Pracując w tym trybie, robot zachowuje się jak robot inspekcyjny. Użytkownik posiada wygodny interfejs służący do sterowania jazdą robota. Za pomocą grupy przycisków wydaje rozkazy do jazdy w przód, tył, lewo i prawo. Program wysyła żądania użytkownika bezpośrednio do robota za pośrednictwem komunikacji bezprzewodowej. W trybie inspekcyjnym robot nieustannie monitoruje użytkownika aplikacji o swoim stanie. Aplikacja w trybie rzeczywistym aktualizuje informację o stanie robota, wskazania ze wszystkich czujników odległości zamontowanych w robocie, a także z czujników z układu LSM303D zamontowanego na pokładzie robota. Dzięki temu użytkownik widzi odległości oraz pochylenia podłużne i poprzeczne całej konstrukcji (rys. 8.8).



Rys. 8.8 Aplikacja do zdalnego sterowania robotem (na zrzucie ekranu widać wskazania poszczególnych czujników odległości oraz kąty nachylenia robota względem płaszczyzny poprzecznej i podłużnej)

## **9. STOPIEŃ REALIZACJI WYMAGAŃ FUNKCJONALNYCH I PIERWOTNYCH ZAŁOŻEŃ**

Podczas pracy nad prototypem udało się zrealizować praktycznie wszystkie założenia i wymagania przyjęte przed rozpoczęciem pisania pracy i tworzeniem projektu. Problemy z poprawnymi wartościami wskazań kompasu elektronicznego wewnątrz wszystkich badanych pomieszczeń okazały się bardzo poważnym problemem, który można obejść przy zastosowaniu o wiele bardziej zaawansowanych algorytmów w połączeniu np. z czujnikami laserowymi. Poważnie podniosłoby to jednak koszt całego projektu. Istnieją także inne metody pomiarów wiążące się także m.in. z koniecznością wcześniejszego oznaczenia kilku punktów odniesienia do poprawnej pracy robota. Stoi to jednak w sprzeczności z początkowo przyjętym założeniem o w pełni autonomicznej pracy bez udziału człowieka w całym procesie. Początkowo pomiarowanie pomieszczeń miało być wykonane w oparciu o przejazd po całym obszarze. Robot miał wykonywać badania dopóki w macierzy zajętości pozostaje chociaż jedna niezbadana komórka. Problemy błędnych wskazań magnetometru wymusiły modyfikację tego założenia. Konsekwencją tego jest także to, że robot umieszczony w dowolnym punkcie pomieszczenia nie odnajdzie poprawnej drogi. W ostatecznej wersji prototypu powinien zostać umieszczony w pobliżu ściany. Zmodyfikowany algorytm działa jednak w dokładny i precyzyjny sposób. Chociaż jego poprawne zaimplementowanie okazało zadaniem złożonym i czasochłonnym. Szczególnie, że wprowadzenie zmian i ulepszeń na etapie znacznego zaawansowania projektu skutkowało koniecznością sprawdzenia wszystkich warunków brzegowych i przeprowadzenia serii testów.

## 10. WADY I ZALETY METOD OPOMIAROWANIA ZAPROPO- NOWANYCH W PROJEKCIE

Pierwotnie opracowana koncepcja pomiarowania pomieszczeń wewnątrz budynku pozwala na zbudowanie pełnej i kompletnej mapy całego badanego obszaru. Metoda polegająca na wykorzystaniu magnetometru w połączeniu z czujnikami ultradźwiękowymi pozwala na precyzyjne zmapowanie całego pomieszczenia. Jak okazało się podczas prac projektowych, może ona być zależna od zewnętrznych zakłóceń elektromagnetycznych. Robot taki będzie działał tylko w sprzyjających warunkach. W przypadku pojawienia się jakichkolwiek zakłóceń mogą one całkowicie uniemożliwić poprawne i stabilne działanie pojazdu. Uzupełnienie kompasu elektronicznego o żyroskop, ze względu na źródło, charakter i skalę problemu nie przyniesie w tym przypadku pożądanych efektów. Alternatywne rozwiązanie opracowane na potrzeby niniejszej pracy uniezależnia poprawne działanie prototypu od wszelkich zakłóceń elektromagnetycznych. Wykorzystanie do badania pomieszczeń koncepcji bazującej na czujnikach odległości pozwoliło wyeliminować problem zakłóceń elektromagnetycznych. Koncepcja alternatywna ogranicza co prawda opomiarowanie do przejazdu wzdłuż ścian, ale testy prototypu w warunkach rzeczywistych pokazały, że zaimplementowana metoda jest niezwykle dokładna i skuteczna. Jeżeli wiązka ultradźwięków napotka na nieregularną powierzchnię (np. ościeżnica z załamaniami) istnieje niebezpieczeństwo, że obliczony kąt nie będzie w pełni poprawny. Robot najlepiej sprawdza się więc w pomieszczeniach o regularnych, klasycznych kształtach takich jak korytarze, pokoje, sale i pomieszczenia o układzie ścian nachylonych wzajemnie pod kątem 90 stopni. Problemem może się także okazać nagłe i skokowe zmiany geometrii ściany takie jak niewielki filar, załamanie, wyłom). Ponadto badanie pomieszczenia jest niezależne od natężenia światła na badanym obszarze, jak ma to miejsce w przypadku czujników ToF. Zastosowane rozwiązanie uniezależnia pracę pojazdu od jakichkolwiek czynników zewnętrznych i zakłóceń pola magnetycznego, które mogłyby wpłynąć na poprawność wykonywanych przez robota badań.

## 11. DALSZE PRACE

Skonstruowany prototyp porusza wiele trudnych i skomplikowanych zagadnień. Z tego powodu istnieje duży potencjał do dalszego rozwijania projektu. Zastosowana konstrukcja umożliwia łatwe rozwijanie pojazdu. Przede wszystkim robot może zostać rozbudowany o urządzenia potrzebne do badania pomieszczeń z wykorzystaniem algorytmu SLAM. Zastosowanie napędu gąsienicowego o dobrych parametrach umożliwia duże zwiększenie masy pojazdu i wyposażenie go w profesjonalny skaner laserowy. Pojazd można uzupełnić także o dodatkowe czujniki i sensory takie jak czujnik wilgotności, czujnik natężenia światła itd. Dodatkowo po dodaniu kamery robot może pracować jako robot inspekcyjny bez konieczności bezpośredniej obserwacji pojazdu przez użytkownika. Prototyp może zostać także rozbudowany o mechanizm wykrywania nagłej i skokowej zmiany bliskości ściany po jego prawej stronie. Obecnie, jeżeli w pomieszczeniu występuje tylko jedno takie miejsce, najlepszym środkiem zaradczym na problem tego typu jest rozpoczęcie pomiarów właśnie od takiego punktu. Jeżeli w pomieszczeniu występuje kilka takich miejsc konieczne może się okazać zaimplementowanie dodatkowego systemu kontrolującego sytuację tego typu. Bardzo długa praca robota powoduje stopniowe wyczerpywanie baterii, co może mieć niekorzystny wpływ na sprawność i szybkość poruszania się robota. Dodanie komparatora napięć pozwoli na zwiększanie mocy silników proporcjonalnie do stopnia ich rozładowania.



## 12. PODSUMOWANIE I WNIOSKI

Przedstawione w niniejszej pracy etapy projektu doskonale obrazują jak skomplikowanym zagadnieniem jest tematyka związana z konstrukcją robotów autonomicznych do eksplorowania pomieszczeń. Do poprawnego działania pojazdu tego typu wymagane jest precyzyjne określenie jego położenia oraz jednocześnie badanie obszaru w bezpośrednim sąsiedztwie robota. W niniejszej pracy udało się udowodnić, że możliwe jest stworzenie skutecznie działającego pojazdu w szybkim czasie, używając jedynie tanich elementów elektronicznych. Prace nad prototypem uwidoczniły problemy, które mogą pojawić się w podczas tworzenia urządzenia. Niespodziewane zakłócenia magnetometru, doskonale pokazują, że do niezawodnego działania robotów autonomicznych w każdych warunkach wymagane jest zastosowanie wysokiej jakości komponentów i czujników (np. skanery laserowe). Oczywiście tego konsekwencją jest jednak drastyczne zwiększenie kosztów ze względu na cenę takich urządzeń.. Niespodziewane problemy z działaniem magnetometru w pierwotnie opracowanej koncepcji sprawiły, że pojawiła się konieczność dostosowania założeń projektowych i wymagań funkcjonalnych robota. Opracowanie odpowiednich algorytmów, skonstruowanie robota oraz zaimplementowanie wszystkich potrzebnych mechanizmów korygujących było zajęciem czasochłonnym i wymagającym. Pomimo dużych trudności napotkanych w projekcie, dzięki obmyśleniu szeregu mechanizmów korekcyjnych, korygujących oraz sprawdzających położenie i orientację robota udało się opracować skuteczny system pomiarowania pomieszczeń wewnątrz budynku. Generowane podczas testowych przejazdów mapy bardzo dobrze odwzorowują kształt i wymiary badanych obszarów. Jest to duże osiągnięcie zważywszy na fakt mocno ograniczonych funduszy jakimi dysponowano podczas tworzenia projektu oraz krótkiego czasu realizacji prototypu.

## Zawartość nośnika dołączonego do pracy:

- nagranie 1: nagranie pracy robota – test nr 1
- nagranie 2: nagranie z pracy robota - test nr 2
- nagranie 3: pomiary pola magnetycznego kompasem analogowym
- kody źródłowe projektu
- niniejsza praca w formacie pdf

## Bibliografia

- [1] National Robotics Engineering Center,  
<https://www.rec.ri.cmu.edu/solutions/manufacturing/autonomous-material-transport.html>,  
(dostęp 29.06.2021)
- [2] IRobot, iRobot Roomba i7, <https://irobot.pl/pl/roomba/384-irobot-roomba-i7-i7158-5060359287311.html>, (dostęp 29.06.2021)
- [3] Ekosystem Arduino, <https://store.arduino.cc/arduino-uno-rev3>, (dostęp 29.06.2021)
- [4] Arduino Uno Rev 3 , Botland, <https://botland.com.pl/arduino-moduly-glowne/1060-arduino-uno-rev3-a000066-8058333490090.html>, (dostęp 29 06 2021)
- [5] Charakterystyka produktu, STM32, <https://www.st.com/en/evaluation-tools/stm32-mcu-mpu-eval-tools.html>, (dostęp 29 06 2021)
- [6] STM32F3DISCOVERY, Botland, <https://botland.com.pl/stm32-discovery/957-stm32f3-discovery-stm32f3discovery.html>, (dostęp 29 06 2021)
- [7] STM32F3DISCOVERY - grafika, Botland, <https://botland.com.pl/stm32-discovery/957-stm32f3-discovery-stm32f3discovery-5904422301132.html>. (dostęp 26 06 2021)
- [8] Produkty RaspberryPi, RaspberryPi, <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>, (dostęp 29.06.2021)
- [9] RaspberryPi, Botland, <https://botland.com.pl/moduly-i-zestawy-raspberry-pi-4b/14646-raspberry-pi-4-model-b-wifi-dualband-bluetooth-2gb-ram-15ghz-765756931175.html>,  
(dostęp 29 06 2021)
- [10] RaspberryPi - grafika, Botland, <https://botland.com.pl/moduly-i-zestawy-raspberry-pi-4b/14646-raspberry-pi-4-model-b-wifi-dualband-bluetooth-2gb-ram-15ghz-765756931175.html>, (dostęp 29.06.2021)
- [11] Silnik krokowy z przekładnią, Botland, <https://botland.com.pl/silniki-krokowe/12807-silnik-krokowy-z-przekladnia-28byj-48-5v-01a-003nm-ze-sterownikiem-uln2003.html>,  
(dostęp 29.06.2021)

- [12] Silnik krokowy z przekładnią 28-BYJ - grafika, Botland, <https://botland.com.pl/silniki-krokowe/12807-silnik-krokowy-z-przekladnia-28byj-48-5v-01a-003nm-ze-sterownikiem-uln2003.html>, (dostęp 29.06.2021)
- [13] Ultradźwiękowy czujnik odległości HC-SR04, Botland, <https://botland.com.pl/ultradzwiekowe-czujniki-odleglosci/15717-ultradzwiekowy-czujnik-odleglosci-a02yyuw-3-450cm-wodoodporny-dfrobot-sen0311.html>. [Data uzyskania dostępu: 29 06 2021], (dostęp 29.06.2021)
- [14] Czujnik na podczerwień - moduł rozszerzeń Unit, Botland, <https://botland.com.pl/moduly-rozszerzen-unit/18074-nadajnikodbiornik-podczerwieni-modul-rozszerzen-unit-do-modulow-deweloperskich-m5stack-6972934170538.html>, (dostęp 29 06 2021)
- [15] Czujnik na podczerwień - moduł rozszerzeń Unit - grafika, Botland, <https://botland.com.pl/moduly-rozszerzen-unit/18074-nadajnikodbiornik-podczerwieni-modul-rozszerzen-unit-do-modulow-deweloperskich-m5stack-6972934170538.html>, (dostęp 29.06.2021)
- [16] Skanery laserowe i laserowe czujniki odległości, Botland, <https://botland.com.pl/443-skanery-laserowe>, (dostęp 29 06 2021)
- [17] Skaner rplidar, Botland, <https://botland.com.pl/skanery-laserowe/19625-skaner-laserowy-rplidar-a1m8-r6-360-stopni-12m-seeedstudio-114992561-5903351248624.html>, (dostęp 29.06.2021)
- [18] Skaner rplidar - grafika, Botland, <https://botland.com.pl/skanery-laserowe/19625-skaner-laserowy-rplidar-a1m8-r6-360-stopni-12m-seeedstudio-114992561-5903351248624.html>, (dostęp 29.06.2021)
- [19] Czujniki ToF, Terabee, <https://www.terabee.com/shop/lidar-tof-range-finders/teraranger-evo-60m/>, (dostęp 29.06.2021)
- [20] Czujnik Terabee teraranger-evo-60m - grafika, Terabee, <https://www.terabee.com/shop/lidar-tof-range-finders/teraranger-evo-60m/>, (dostęp 29.06.2021)
- [21] Hallmann I., Siemiątkowska B., Nawigacja robota mobilnego z użyciem sztucznych znaczników, VII krajowa Konf. Robotyki, Prace Naukowe ICT PW, Wrocław 2001, s. 177-184.
- [22] Rendon-Mancha J. M., Cardenas A., Romero M., Gonzalez-Galvan E, Robot Positioning Using Camera-Space Manipulation With a Linear Camera Model, [https://www.researchgate.net/publication/224145518\\_Robot\\_Positioning\\_Using\\_Camera-Space\\_Manipulation\\_With\\_a\\_Linear\\_Camera\\_Model](https://www.researchgate.net/publication/224145518_Robot_Positioning_Using_Camera-Space_Manipulation_With_a_Linear_Camera_Model), (dostęp 29.06.2021)
- [23] System wizualny multisense, Carnegie Robotics, <https://carnegierobotics.com/products/multisense-s7/>, (dostęp 29.06.2021)
- [24] Pololu 852, Botland, <https://botland.com.pl/produkty-wycofane/852-minimu-9-v2-akcelerometr-zyroskop-i-magnetometr-imu-i2c-modul-pololu.html>, (dostęp 29.06.2021)
- [25] Siemiątkowska B, Hybrydowa reprezentacja otoczenia robota mobilnego, Przemysłowy Instytut Automatyki i Pomiarów PIAP, *Pomiary Automatyka Robotyka*, [http://ent.ipipan.waw.pl/robot-ent/img/pub/bsiemiat\\_hybryd.pdf](http://ent.ipipan.waw.pl/robot-ent/img/pub/bsiemiat_hybryd.pdf), 2007 03 16, (dostęp 29.06.2021)

- [26] B. Siemiątkowska, R. Chojecki, M. Różańska-Walczyk, M. Przybylski, P. Węclewski, M. Wiśniowski, Planowanie trasy robota Kurier w środowisku dynamicznym z wykorzystaniem sieci komórkowych, Wydział Mechatroniki, Instytut Automatyki i Robotyki, Politechnika Warszawska, 2/2012 Pomiar automatyka Robotyka, (dostęp 29.06.2021)
- [27] K. Kułakowski, J. Wąs, M. Szpryka, Dynamiczny model świata w sterowaniu autonomicznym robotem mobilnym, Automatyka, 2008, Tom 12, Zeszyt 3
- [28] M. Ben-Ari and F. Mondada, N. Tomatis, I. Nourbakhsh, R. Siegwart, Elements of Robotics, Combining Topological and Metric: A Natural Integration for Simultaneous Localization and Map Building, Workshop Paper, 4th European Workshop on Advanced Mobile Robots (EUROBOT '01), *September, 2001*, [https://www.ri.cmu.edu/pub\\_files/pub3/tomatis\\_nicola\\_2001\\_1/tomatis\\_nicola\\_2001\\_1.pdf](https://www.ri.cmu.edu/pub_files/pub3/tomatis_nicola_2001_1/tomatis_nicola_2001_1.pdf), (dostęp 29.06.2021)
- [29] M. Ben-Ari, Elements of Robotics, [https://link.springer.com/chapter/10.1007/978-3-319-62533-1\\_9](https://link.springer.com/chapter/10.1007/978-3-319-62533-1_9), 201, (dostęp 29.06.2021)
- [30] A. Zelinsky, Mobile robot map making using sonar, <https://www.semanticscholar.org/paper/Mobile-robot-map-making-using-sonar-Zelinsky/da03753bb567228647f3a592aaec39be17d4a54e>, (dostęp 29.06.2021)
- [31] M. Ben-Ari and F. Mondada, Elements of Robotics, [https://doi.org/10.1007/978-3-319-62533-1\\_9](https://doi.org/10.1007/978-3-319-62533-1_9), rozdział 9, (dostęp 29.06.2021)
- [32] N. Tomatis, I. Nourbakhsh, R. Siegwart, Combining Topological and Metric: A Natural Integration for Simultaneous Localization and Map Building, Swiss Federal Institute of Technology Lausanne (EPFL) , 01 2001, [https://www.researchgate.net/publication/37441174\\_Combining\\_Topological\\_and\\_Metric\\_A\\_Natural\\_Integration\\_for\\_Simultaneous\\_Localization\\_and\\_Map\\_Building](https://www.researchgate.net/publication/37441174_Combining_Topological_and_Metric_A_Natural_Integration_for_Simultaneous_Localization_and_Map_Building), (dostęp 29.06.2021)
- [33] I. N. R. Siegwart, Introduction to Autonomous Mobile Robots, Cambridge, MIT Press 2004
- [34] I. N. R. Siegwart, Introduction to Autonomous Mobile Robots, Cambridge, MIT Press, 2004
- [35] H. M. D Herrero, Fuzzy mobile-robot positioning in intelligent spaces using wireless sensor networks, Sensors, 2011, <https://www.mdpi.com/1424-8220/11/11/10820>, (dostęp 29.06.2021)
- [36] D. L. J. L. Stephen Se, Local and Global Localization for Mobile Robots using Visual Landmarks, Department of Computer Science University of British Columbia Vancouver, B.C. V6T 1Z4, Canada, Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems Maui, Hawaii, USA, Oct. 29 - Nov. 03, 2001, <https://www.cs.ubc.ca/~lowe/papers/iros01.pdf>, (dostęp 29.06.2021)
- [37] Biblioteka do obsługi lsm303, Pololu, Available: <https://github.com/pololu/lsm303-arduino>, (dostęp 29.06.2021)
- [38] Karta produktu LSM303D, Pololu, <https://www.pololu.com/file/0J703/LSM303D.pdf>, (dostęp 29.06.2021)
- [39] Akcelerometr na mikrokontrolerze STM32, Forbot, <https://forbot.pl/blog/kurs-stm32-f1-hal-i2c-w-praktyce-akcelerometr-id24350>, (dostęp 29.06.2021)

- [40] Karta produktu STM32F411VE, <https://www.st.com/resource/en/datasheet/stm32f411ve.pdf>, (dostęp 29.06.2021)
- [41] Nota aplikacyjna LSM303DLH, Pololu, <https://www.pololu.com/file/0J434/LSM303DLH-compass-app-note.pdf>, (dostęp 29.06.2021)
- [42] D. A. Patil, Design and Implementation of Mapping Robot using Digital Magnetic Compass and Ultrasonic Sensor, International Journal of Engineering Research & Technology (IJERT), 04 06 2015, IJERT, <https://www.ijert.org/research/design-and-implementation-of-mapping-robot-using-digital-magnetic-compass-and-ultrasonic-sensor-IJERTV4IS060839.pdf>, (dostęp 29.06.2021)
- [43] Karta produktu STM32F103, STM, <http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/CD00161566.pdf>, (dostęp 29.06.2021)
- [44] Instrukcja obsługi produktów serii F1, [http://www.st.com/st-web-ui/static/active/en/resource/technical/document/reference\\_manual/CD00171190.pdf](http://www.st.com/st-web-ui/static/active/en/resource/technical/document/reference_manual/CD00171190.pdf), (dostęp 29.06.2021)
- [45] Instrukcja użytkowania biblioteki HAL dla produktów F1, STM, [http://www.st.com/content/ccc/resource/technical/document/user\\_manual/72/52/cc/53/05/e3/4c/98/DM00154093.pdf/files/DM00154093.pdf/jcr:content/translations/en.DM00154093.pdf](http://www.st.com/content/ccc/resource/technical/document/user_manual/72/52/cc/53/05/e3/4c/98/DM00154093.pdf/files/DM00154093.pdf/jcr:content/translations/en.DM00154093.pdf), (dostęp 29.06.2021)
- [46] Ultradźwiękowy czujnik odległości HC-SR04 - grafika, Botland, <https://botland.com.pl/ultradzwiekowe-czujniki-odleglosci/1420-ultradzwiekowy-czujnik-odleglosci-hc-sr04-2-200cm-5903351241366.html>, (dostęp 29.06.2021)
- [47] Dwukanałowy sterownik silników, Pololu, <https://www.pololu.com/product/2135>, (dostęp 29.06.2021)
- [48] Komunikacja z komputerem przez uart, Forbot, <https://forbot.pl/blog/kurs-stm32-f1-hal-komunikacja-z-komputerem-uart-id22896>, (dostęp 29.06.2021)
- [49] N. Tomatis, I. Nourbakhsh, R. Siegwart, Combining Topological and Metric: A Natural Integration for Simultaneous Localization and Map Building, Autonomous Systems Lab Swiss Federal Institute of Technology Lausanne (EPFL), 2001
- [50] M. Ben-Ari, F. Mondada, Elements of Robotics, rozdział 9, SpringerOpen, 2018
- [51] Nota aplikacyjna LSM303DLH, Pololu, <https://www.pololu.com/file/0J434/LSM303DLH-compass-app-note.pdf>, (dostęp 29.06.2021)
- [52] Specyfikacja techniczna STM32F103RB, STM, <https://www.st.com/resource/en/datasheet/stm32f103rb.pdf>, (dostęp 29.06.2021)

