

## Spis treści

Wstęp .....	4
Cel i zakres pracy .....	4
Struktura pracy .....	4
Ogólne założenia i wymagania funkcjonalne .....	4
Podstawowe zagadnienia dotyczące robotów mobilnych .....	4
Algorytmy i metody używane przy badaniu otoczenia .....	4
Metody określania dokładnej pozycji robota .....	4
Sposoby reprezentacji badanego obszaru .....	4
Przegląd narzędzi i urządzeń wykorzystywanych do opomiarowania pomieszczeń...	4
Platformy sprzętowe .....	4
Arduino .....	4
RaspberryPi .....	4
STM32 .....	4
Czujniki i sensory używane w robotyce .....	4
Enkodery i silniki krokowe .....	4
Czujniki ultradźwiękowe .....	4
Czujniki na podczerwień .....	4
Czujniki i skanery laserowe .....	4
Czujniki ToF .....	4
Kamery .....	4
Realizacja prototypu pojazdu autonomicznego do pomiarowania pomieszczeń wewnątrz budynku .....	4
Konstrukcja mechaniczna .....	4
Płytki Nucleo STM32F103RB .....	5
Magnetometr i akcelerometr Pololu LSM303D .....	5
Czujniki ultradźwiękowe HC-SR04 .....	5

Dwukanałowy sterownik silników Pololu DRV8835 .....	6
Komunikacja bezprzewodowa: moduł Bluetooth HC-06 ZS-040 .....	7
Moduł zasilania: ogniwa Li-ion 2600mAh .....	7
Stabilizacja zasilania: kondensator 1000µF 25V .....	8
Wykorzystanie algorytmu DFS w połączeniu z algorytmem Trémauxa .....	8
Reprezentacja pomieszczenia: dyskretyzacja pomieszczenia i macierz zajętości ..	9
Problemy dotyczące poprawnego działania magnetometru w pomieszczeniach.....	10
Weryfikacja wskazań magnetometru: Nucleo STM32F411E Discovery wraz z układem LSM303DLHC.....	10
Magnetometr: kalibracja i kompensacja przechyłu .....	10
Zmiana położenia modułu z magnetometrem i wykorzystanie serwomechanizmu	13
Niestabilność pola magnetycznego przy badaniu pomieszczeń.....	14
Źródła i przyczyny niepoprawnych wskazań magnetometru .....	15
Opracowanie alternatywnego sposobu opomiarowania pomieszczeń.....	15
Badanie otoczenia na podstawie przejazdu robota wzdłuż ścian .....	15
Przebudowa robota .....	15
Wykorzystanie odeometrii i czujników odległości .....	16
Określenie kierunku i położenia robota .....	16
Korekta błędów.....	16
Peryferia i spis wyprowadzeń .....	16
Aplikacja do komunikacji pomiędzy robotem a komputerem (środowisko .NET) .....	16
Tryb autonomiczny .....	16
Tryb zdalnego sterowania .....	17
Testy robota i zaprojektowanego algorytmu oraz aplikacji do komunikacji w warunkach rzeczywistych .....	18
Stopień realizacji wymagań funkcjonalnych i pierwotnych założeń .....	21
Wady i zalety metod opomiarowania zaproponowanych w projekcie .....	21
Dalsze prace.....	21

Podsumowanie i wnioski.....	22
-----------------------------	----

Wstęp

Cel i zakres pracy

Struktura pracy

Ogólne założenia i wymagania funkcjonalne

Podstawowe zagadnienia dotyczące robotów mobilnych

Algorytmy i metody używane przy badaniu otoczenia

Metody określania dokładnej pozycji robota

Sposoby reprezentacji badanego obszaru

Przegląd narzędzi i urządzeń wykorzystywanych do opomiarowania pomieszczeń

Platformy sprzętowe

Arduino

RaspberryPi

STM32

Czujniki i sensory używane w robotyce

Enkodery i silniki krokowe

Czujniki ultradźwiękowe

Czujniki na podczerwień

Czujniki i skanery laserowe

Czujniki ToF

Kamery

Realizacja prototypu pojazdu autonomicznego do pomiarowania pomieszczeń  
wewnątrz budynku

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. Do poruszania się robot wykorzystuje gąsienice napędzane dwoma silnikami prądu stałego. Zarówno prowadnice gąsienic jak i same gąsienice 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. W czasie pracy bez obciążenia silniki pobierają 100 mA. Nadwozie robota stanowi lekka drewniana płyta odseparowana od podwozia za pomocą tulei dystansowych o długości 25 mm. W płytce znajduje się szereg wycięć i otworów ułatwiających podłączenie komponentów elektronicznych potrzebnych do poprawnego działania prototypu.

### Płytki Nucleo STM32F103RB

Podczas tworzenia prototypu podjęto decyzję o wykorzystaniu zestawu NUCLEO - F103RB z rodziny STM32 Nucleo64. Jest on wyposażony w mikrokontroler STM32F103RBT6 w obudowie LQFP64. Jego rdzeń to 32-bitowy mikroprocesor ARM Cortex M3 o częstotliwości taktowania do 72MHz. Użyta płytki posiada siedem Timerów, 3 interfejsy USART, dwa przetworniki analogowo-cyfrowe (zarówno 12 jak i 16-kanalowy), dwa przyciski (jeden przycisk RESET i jeden do dyspozycji użytkownika) 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 całego układu. Nucleo64 umożliwia podłączenie zasilania w trzech różnych wariantach. Nucleo-F103RB może być zasilany poprzez złącze USB, lub za pomocą zewnętrznego źródła zasilania. W drugim przypadku mamy do wyboru wejście zasilania VIN, które można podłączyć do napięcia od 7V do 12 V (przy maksymalnym obciążeniu prądowym 800mA) oraz wejście zasilania E5V, które pozwala podłączyć napięcie w zakresie 4,75 – 5,35 V (przy maksymalnym obciążeniu prądowym 500mA).

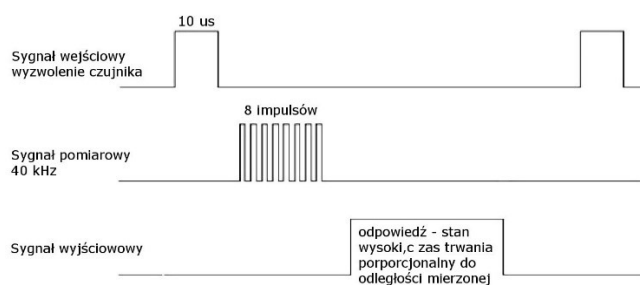
### Magnetometr i akcelerometr Pololu LSM303D

Podczas prac koncepcyjnych nad robotem 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 zasilanie modułu napięciem z zakresu od 2,5V do 5,5V. LSM303D obsługuje dwie magistrale: I2C oraz SPI. 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 zaledwie 5mA.

### Czujniki ultradźwiękowe HC-SR04

Do monitorowania pomieszczenia, w którym znajduje się robot wykorzystano popularne czujniki ultradźwiękowe HC-SR04. 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 mu uchwytów montażowych. Sensory te pracują w zakresie od 2 cm do 200cm. Ich dużą wadą jest jednak wysokie napięcie zasilania potrzebne do poprawnej pracy. Wynosi ono aż 5 V. Ogranicza to możliwość swobodnego podłączenia czujnika do płytki, ponieważ musi on zostać podłączony do pinów 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 poniżej.



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.

$$\text{Test\_distance} = (\text{high\_level\_time} \times \text{velocity\_of\_sound (340m/s)} ) / 2$$

Ostatecznie do wyliczenia odległości w programie komputerowym użyto wzoru przedstawionego poniżej.

$$\text{distance [cm]} = ( \text{high level time [us]} * 34 ) / 1000 / 2$$

### Dwukanałowy sterownik silników Pololu DRV8835

Do sterowania pracą obu silników w jakie wyposażony jest pojazd użyto dwukanałowego sterownika silników DRV8835 produkcji firmy Pololu. Układ ten pracuje z napięciem w zakresie od 2 V do 11V oraz prądem na kanał do 1,2 A. Wytrzymuje on także 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 przy konstruowaniu prototypu. Do tworzenia prototypu wykorzystano tryb PHASE/ENABLE.

#### **Komunikacja bezprzewodowa: moduł Bluetooth HC-06 ZS-040**

Komunikację bezprzewodową zrealizowano za pomocą modułu Bluetooth HC-06 ZS-040. Urządzenie komunikuje się poprzez szeregowy interfejs UART. Posiada piny do transmisji i odbioru danych (TX i RX), które pracują z napięciem 3,3 V. Sam moduł może być zasilana 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 zasilania: ogniwa Li-ion 2600mAh**

Na układ zasilania całego urządzenia składa się moduł z zestawem dwóch akumulatorów litowo-jonowych. Pojemność każdego ogniwa wynosi

Na moduł zasilania całej konstrukcji składa się moduł złożony z dwóch wydajnych akumulatorów litowo – jonowych firmy Xtar o pojemności 2600 mAh zbudowany w oparciu o oryginalne ogniwo firmy Samsung. Posiada on zabezpieczenie przed nadmiernym rozładowaniem , przeładowaniem, przeciążeniem, a także zwarcie. Użyte ogniwa, choć droższe od standardowych ogniw o około 20% zabezpiecza nas przed uszkodzeniami innych elementów elektronicznych użytych przy tworzeniu prototypu. Prąd ładowania akumulatorów wynosi 2 A, przy prądzie rozładowania 5 A. Napięcie nominalne każdego ogniwa wynosi 3,7 V. Wymiary obu baterii wynoszą 18,4 x 69,2 mm.

### Stabilizacja zasilania: kondensator 1000 $\mu$ F 25V

Do stabilizacji układu zasilania użyto pojedynczego kondensatora elektrolitycznego o pojemności 1000  $\mu$ F i maksymalnym napięciu 25V. Zarówno oba silniki jak i serwomechanizm podłączony został do tej samej linii co mikrokontroler. Wymusza to zastosowanie filtracji zasilania w celu eliminacji zakłóceń powstających przy działaniu tych elementów. Dzięki temu mikrokontroler nie resetuje ani nie zaciąga się i zawsze działa stabilnie zgodnie z naszymi oczekiwaniami. Zastosowanie osobnego źródła zasilania dla każdego modułu niepotrzebnie zwiększyłoby masę oraz stopień skomplikowania prototypu.

### Wykorzystanie algorytmu DFS w połączeniu z algorytmem Trémauxa

W początkowym etapie projektu założono, że robot będzie badał całe pomieszczenie na podstawie algorytmu bardzo zbliżonego do 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 znajduje się robot jest traktowany jako „północ”. Reszta symbolicznych kierunków jest wyznaczana przez dodanie do aktualnego 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 z każdej jego strony. 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. Zastosowanie czujnika



dotykowego pozwala zapobiec takiej sytuacji. Dzięki temu po zderzeniu z przeszkodą robot zatrzymuje się, wraca do punktu poprzedniego i oznacza obszar jako zajęty.

Niezwykle ważnym aspektem w zaprojektowanej 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 boolowskiej. Początkowo wszystkie komórki przyjmują wartość nieodwiedzone (logiczne 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 miejsca. 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 pomimo tego, doskonale nadaje się on do adaptacji przy badaniu pomieszczeń metodą podobną do metody DFS. Połączenie macierzy zajętości i macierzy miejsc odwiedzonych pozwala na skuteczne i łatwe w implementacji rozwiązanie przeznaczone do badania każdego rodzaju pomieszczenia, nawet tego o nieregularnych kształtach.

### Reprezentacja pomieszczenia: dyskretyzacja pomieszczenia i 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. Rozmiar dwuwymiarowej tablicy jest stały w czasie działania programu i nie ulega zmianie. Może on jednak zostać rozszerzony lub zmniejszony przed przystąpieniem do pomiarów. Jako początkową wartość położenia pojazdu przyjmujemy połowę długości tablicy. Na potrzeby projektu założono początkowo, 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.

### Problemy dotyczące poprawnego działania magnetometru w pomieszczeniach

W czasie prac nad pierwotną wersją prototypu urządzenia 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 dokonano badania i diagnostyki magnetometru oraz dokonano weryfikacji pomiarów pola magnetycznego.

### Weryfikacja wskazań magnetometru LSM303D: Arduino Uno oraz dedykowana biblioteka Pololu

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. 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.

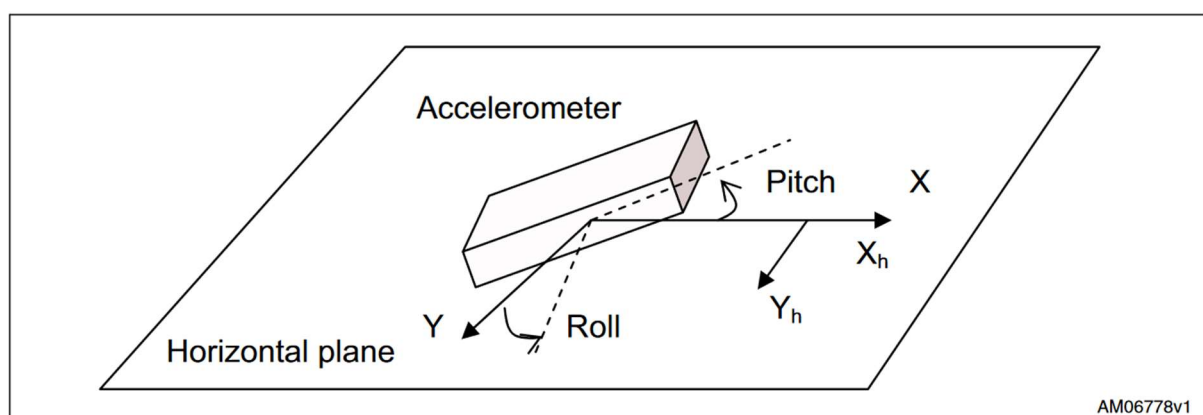
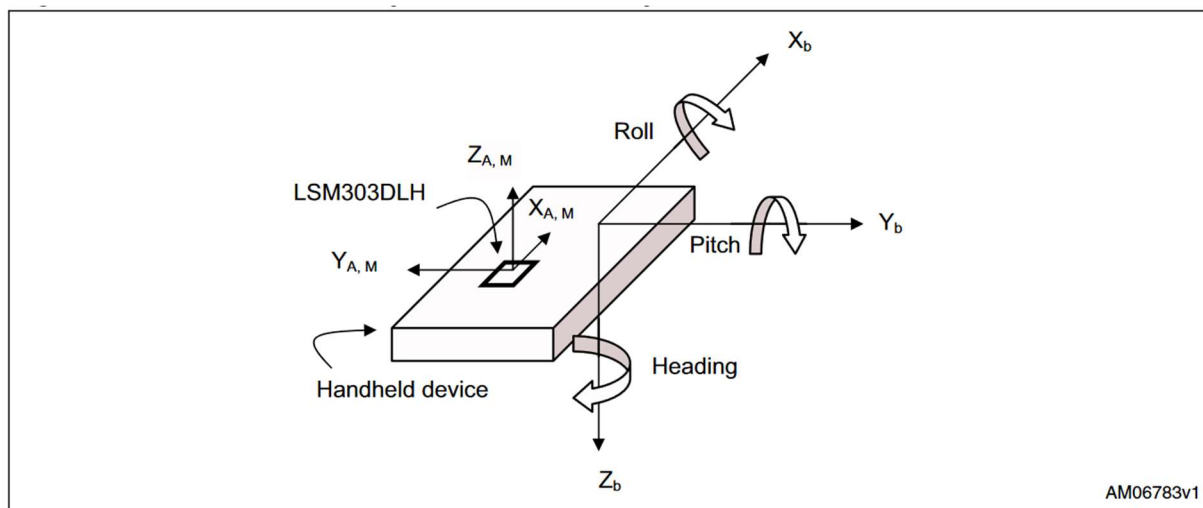
### Weryfikacja wskazań magnetometru: Nucleo STM32F411E Discovery wraz z układem LSM303DLHC

Do celu dalszej diagnostyki wartości pola magnetycznego wykorzystano mikrokontroler z rodziny STM Nucleo STM32F411E Discovery. Posiada ona wbudowany moduł magnetometru LSM303DLHC zintegrowany z akcelerometrem. 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 oraz odczytanie transmitowanych wyników w terminalu. 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. 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.

### **Magnetometr: kalibracja i kompensacja przechyłu**

W celu eliminacji błędów związanych z wypoziomowaniem płytki wykorzystano metodę kompensacji przechyłu. Metoda ta polega na jednoczesnym wykorzystaniu zarówno wskazań magnetometru jak i wartości odczytanych z akcelerometru. Na rysunkach poniżej 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.



Do wyznaczenia wychyleń podłużnego i poprzecznego (ang. pitch and roll) posłużono się dwoma wzorami wykorzystujące dane z akcelometru.

$$\text{Roll} = \text{atan}(A_z / A_y)$$

$$\text{Pitch} = \text{atan}(-A_x / (A_y * \sin(\text{Roll}) + A_z * \cos(\text{Roll})))$$

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.

$$X_h = Y_M * \cos(\text{Roll}) - Z_M * \sin(\text{Roll})$$

$$Y_h = X_M * \cos(\text{Pitch}) + Y_M * \sin(\text{Pitch}) * \sin(\text{Roll}) + Z_M * \sin(\text{Pitch}) * \cos(\text{Roll})$$

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:

$$\text{Heading} = \text{atan}(Y_H / X_H)$$

Obliczona wartość może jednak przyjąć wartość mniejszą od zera, dlatego ostateczną wartość naszego kierunku otrzymujemy wykonując prostą operację modulo zgodnie ze wzorem umieszczonym poniżej.

$$H = (\text{Heading} + 360) \bmod 360$$

Testy zaimplementowanej kompensacji przechyłu 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.

### Zmiana położenia modułu z magnetometrem i wykorzystanie serwomechanizmu

Aby przetestować działanie magnetometru podczas pracy robota moduł LSM303D odseparowano od innych urządzeń i przewodów umieszczając go na osobnej płytce stykowej. Następnie wykonano serię odczytów przy włączonych i wyłączonych silnikach. Wszystkie odczytane przyspieszenia 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 podkładkę przyklejoną do serwomechanizmu dodanego na przedzie robota. Serwomechanizm dodano w celu weryfikacji odczytów z magnetometru przed i po wykonaniu obrotu przez pojazd. 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 płytką z tworzywa sztucznego, na której końcu 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.

### Niestabilność pola magnetycznego przy badaniu pomieszczeń

W celu potwierdzenia podejrzeń o wpływie zewnętrznego pola magnetycznego na pracę robota, przeprowadzono szereg pomiarów w miejscu i 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 (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 LSM. Trudnością w szybkim zidentyfikowaniu problemu było testowanie wskazania kierunku północnego w jednym miejscu podczas pracy z modułem przy biurku i komputerze. Moduł LSM testowany wraz z kompensacją przechyłu tylko w jednym punkcie działał stabilnie i zgodnie z przewidywaniami. Utrudniło to szybką identyfikację problemu.

## Źródła i przyczyny niepoprawnych wskazań magnetometru

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 do ziemi. Za główną przyczynę problemu można podejrzewać bezpośrednie sąsiedztwo linii wysokiego napięcia z badanymi pomieszczeniami (testy prowadzono w budynku stojącym 30 metrów od słupów energetycznych). Dodatkowo zaobserwowano silną korelację pomiędzy bliskością podłoża 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 działanie układu. Układy tego typu z powodzeniem stosowane są nawet w amatorskiej automatyce i robotyce. 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 początkowych założeń projektowych w taki sposób, aby zachować uniwersalność robota.

## Opracowanie alternatywnego sposobu opomiarowania pomieszczeń

### Badanie otoczenia na podstawie przejazdu robota wzdłuż ścian

#### Przebudowa robota

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. Dodatkowo należało ulepszyć metodę badania przeszkód na prawym boku robota. W prototypie przyjęto, że robot będzie badał położenie ścian 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.



### Wykorzystanie odeometrii i czujników odległości

W alternatywnej wersji algorytmu główną rolę przy pomiarowaniu spełnia odometria. 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 wspomniany powyżej 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.

### Określenie kierunku i położenia robota

W alternatywnej wersji projektu przyjęto bardzo zbliżone założenia dotyczące określania kierunku robota. Zakładamy, że pojazd może poruszać się w czterech różnych 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 obszarów wewnątrz budynków.

### Korekta błędów

#### Peryferia i spis wyprowadzeń

#### Aplikacja do komunikacji pomiędzy robotem a komputerem (środowisko .NET)

Do sterowania prototypem stworzono aplikację przeznaczoną na komputery PC. Aplikacja została napisana w środowisku .NET z wykorzystaniem języka C# oraz technologii Windows Forms.

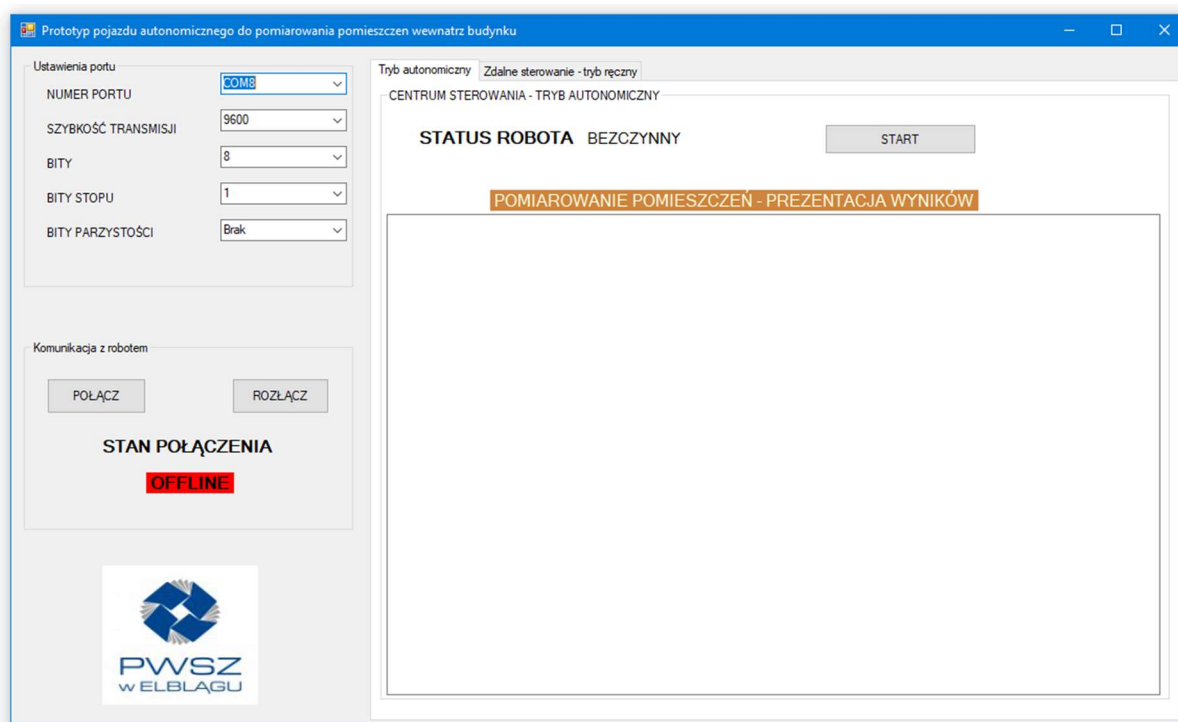
Przy projektowaniu zachowania robota przewidziano dwa główne tryby działania.

### Tryb autonomiczny

Pierwszy zaprojektowany tryb pracy robota to tryb autonomicznego badania i opomiarowywania pomieszczeń. W trybie tym użytkownik wydaje robotowi polecenie rozpoczęcia wykonania pomiarów i czeka aż robot zbada całe pomieszczenie i zakończy pomiary. Użytkownik może w każdej chwili ręcznie przerwać pomiary. W trybie tym robot na bieżąco aktualizuje mapę z badanym pomieszczeniem.



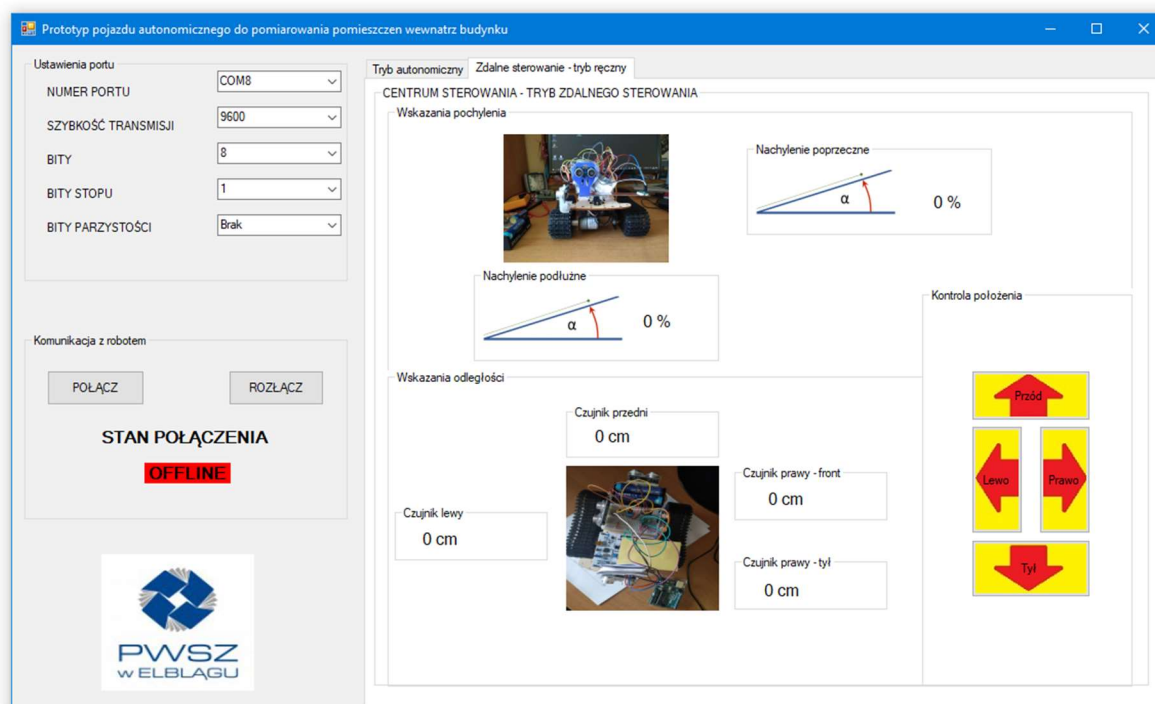
Prezentacja wyników przedstawiona jest jako macierz komórek, które robot generuje na podstawie przejazdu podczas autonomicznej jazdy i badania pomieszczenia. Wszystkie komórki macierzy początkowo przyjmują wartość '\_'. W miejscach, w których robot podczas pomiarów w trybie autonomicznym wykrył przeszkodę w macierzy zapisywana jest wartość 'X'. Pozwala to na wygodną i szybką prezentację wyników pomiarów.



### Tryb zdalnego sterowania

Drugim zaprojektowanym modelem działania jest tryb zdalnego sterowania. Pracując w tym trybie, robot zachowuje się jak robot inspekcyjny. Użytkownik posiada wygodny interfejs do służący do sterowania i jazdy robotem. 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.



### Testy robota i zaprojektowanego algorytmu oraz aplikacji do komunikacji w warunkach rzeczywistych

Prototyp robota poddano licznym testom i próbom. Testy rozpoczęto od sprawdzenia zachowania robota przy warunkach brzegowych, w których 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

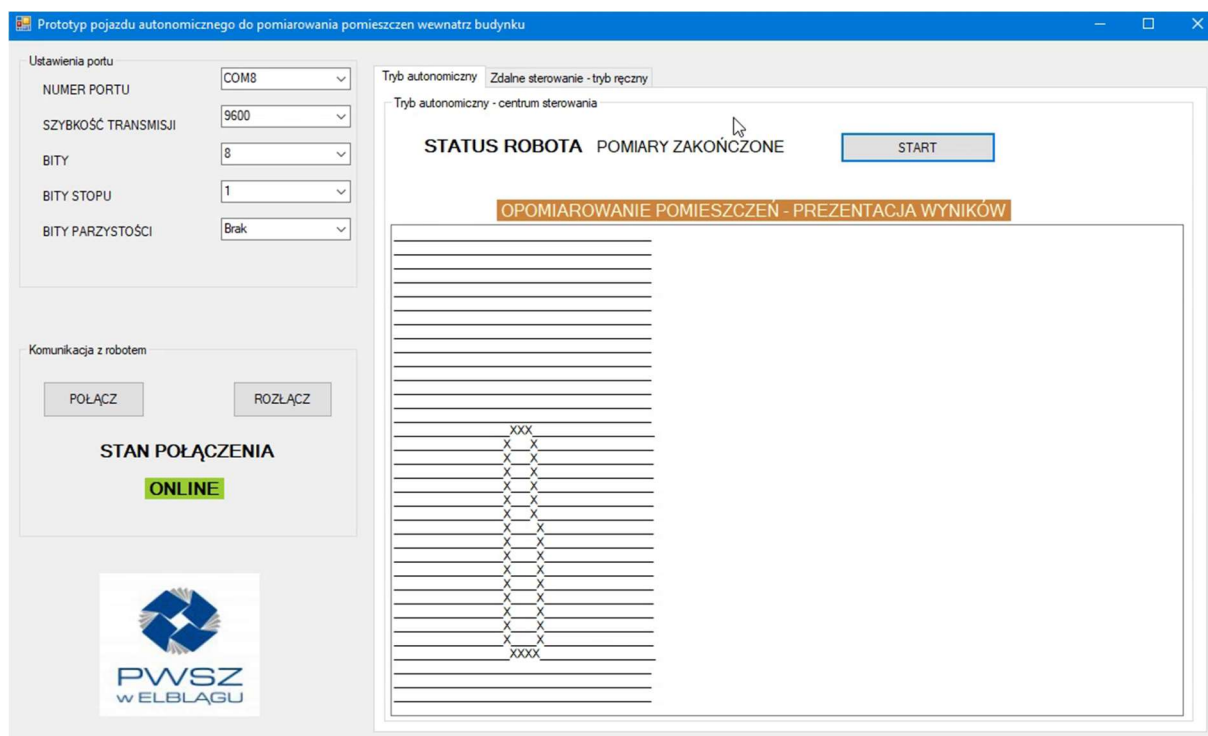
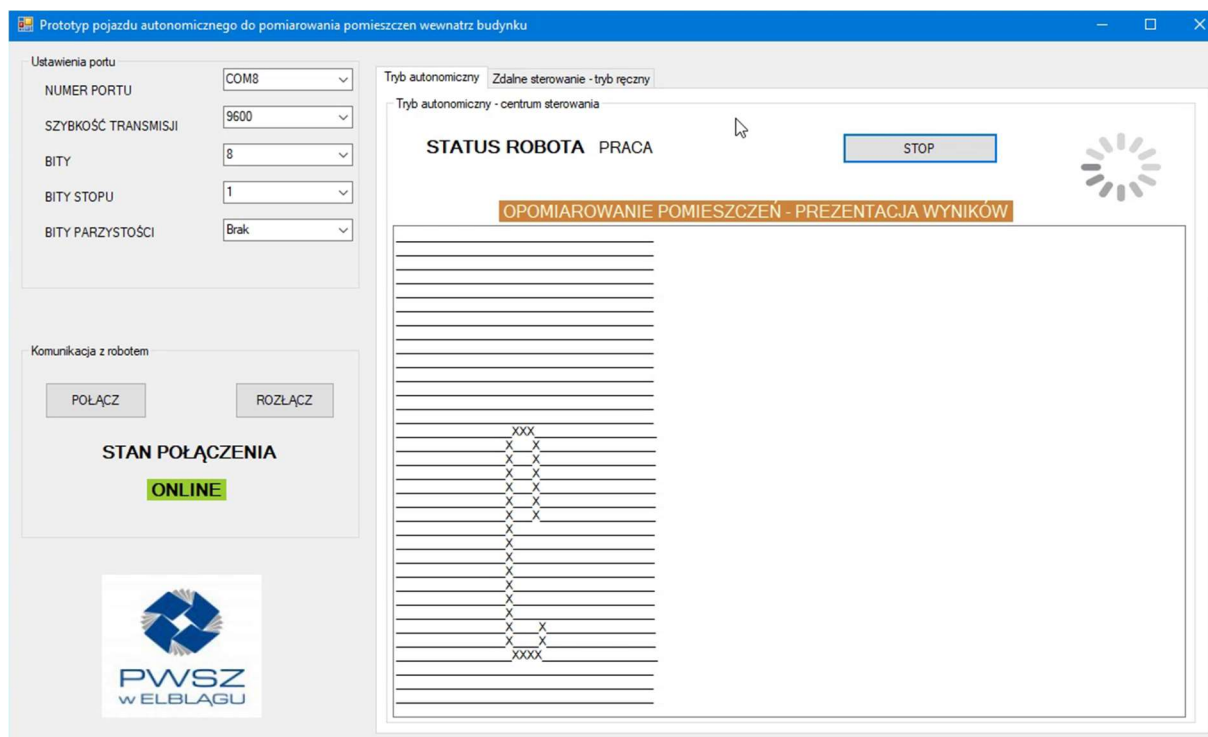
nachylenie do ściany podczas jazdy (miejscami nachylenie wynosiło nawet ok. 45 stopni). Aby poprawić efektywność mechanizmu korekcji błędów podczas jazdy przy ścianie wprowadzono serię poprawek i ulepszeń. Zmniejszono minimalną odległość wymaganą podczas jazdy przy ścianie. W konsekwencji zredukowano 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.

Do prezentacji wyników i weryfikacji poprawności działania algorytmu i zastosowanych mechanizmów wykorzystano stworzoną uprzednio aplikację z graficznym interfejsem użytkownika napisaną z wykorzystaniem technologii .NET. Aplikacja okazała się być niezwykle użyteczna przy prezentacji wyników i pomagała w wygodny sposób na weryfikację rezultatów każdego pomiarowania, szczególnie,

że pole z wynikami pozwala nam nie tylko na prezentację wartości macierzy, ale także wydrukowanie innych dodatkowych informacji. Przez cały okres testów aplikacja działała stabilnie, ani razu nie zrywając połączenia.



## Stopień realizacji wymagań funkcjonalnych i pierwotnych założeń

Podczas pracy nad prototypem udało się zrealizować większość założeń i wymagań przyjętych 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 robota bez udziału człowieka w całym procesie.

Elementami pracy, którego nie udało się w pełni osiągnąć jest bardzo dokładne zbadanie całego obszaru wewnątrz pomieszczenia. 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 okazały się jednak przeszkodą nie do pokonania i wymusiły drastyczną modyfikację tego założenia. Konsekwencją tego jest także to, że robot umieszczony w dowolnym punkcie pomieszczenia nie odnajdzie poprawnej drogi, przez co powinien zostać umieszczony w pobliżu ściany. Zmodyfikowany algorytm badania działa jednak w bardzo zadowalający sposób. Chociaż jego poprawne zaimplementowanie okazało się być trudnym, złożonym i czasochłonnym zadaniem. Szczególnie, że wprowadzenie zmian i ulepszeń na etapie znacznego zaawansowania projektu skutkowało koniecznością sprawdzenia wszystkich warunków brzegowych i przeprowadzenia serii wszystkich testów na nowo.

## Wady i zalety metod opomiarowania zaproponowanych w projekcie

### Dalsze prace

Skonstruowany prototyp porusza wiele trudnych i skomplikowanych zagadnień. Nie wszystkie tematy zostały w pełni rozwinięte. Zastosowana konstrukcja umożliwia jednak dalsze rozwijanie stworzonego pojazdu. Przede wszystkim robot może zostać rozbudowany o urządzenia potrzebne do badania pomieszczeń z wykorzystaniem algorytmu SLAM. Zastosowanie mocnego 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.

## Podsumowanie i wnioski