

Robot line follower zbudowany i
zaprogramowany w oparciu o Lego
Mindstorms.
Wstęp do Robotyki

Dzmitry Kuksik , Antonina Lobach

28-01-2017

Spis treści

1	Treść zadania	2
1.1	Podążanie wzdłuż linii(Linefollower)	2
1.2	Transporter	2
2	Budowa robota	3
2.1	Elementy wykonawcze	3
2.2	Czujniki zastosowane	3
3	Algorytm sterowania	6
3.1	Kalibracja czujników	6
3.2	Line Follower	6
3.2.1	Dobieranie parametrów algorytmu	6
3.3	Algorytm transportera	7
4	Podsumowania	9

Rozdział 1

Treść zadania

Naszym zadaniem było zbudowanie robota głównym celem którego jest podążanie za czarną linią na białym tle.

Robot powinien być zbudowany z klocków oraz elementów dodatkowych wchodzących w skład zestawu *Lego Mindstorms EV3* oraz zaprogramowanego na komputerach dostępnych w laboratorium. Dostępne czujniki: 2 czujnika koloru, czujnik dotyku oraz czujnik podczerwieni. Dostępne silniki: 2 duże serwomechanizmy ,1 średni serwomechanizm.

1.1 Podążanie wzdłuż linii(Linefollower)

Zadaniem robota było przejechanie całej trasy po wyznaczonej linii.

1.2 Transporter

Zadaniem robota było przetransportowanie obiektów z punktów bazowych do punktów docelowych. Punkt bazowy oznaczony jest zielonym rozwidleniem trasy. Kolor punktu bazowego definiuje do jakiego punktu docelowego należy dostarczyć cargo. Droga do punktu docelowego oznaczona jest odpowiednim rozwidleniem trasy.

Rozdział 2

Budowa robota

2.1 Elementy wykonawcze

Budowa robota jest dość klasyczna - to przykład maszyny o napędzie różnicowym z jednym punktem podparcia w postaci koła sferycznego. Pod kostką po obu stronach zamontowane zostały dwa duże serwomechanizmy wraz z kołami o największej średnicy.

Koło sferyczne umieściliśmy z tyłu pojazdu w celu zachowania stabilności.

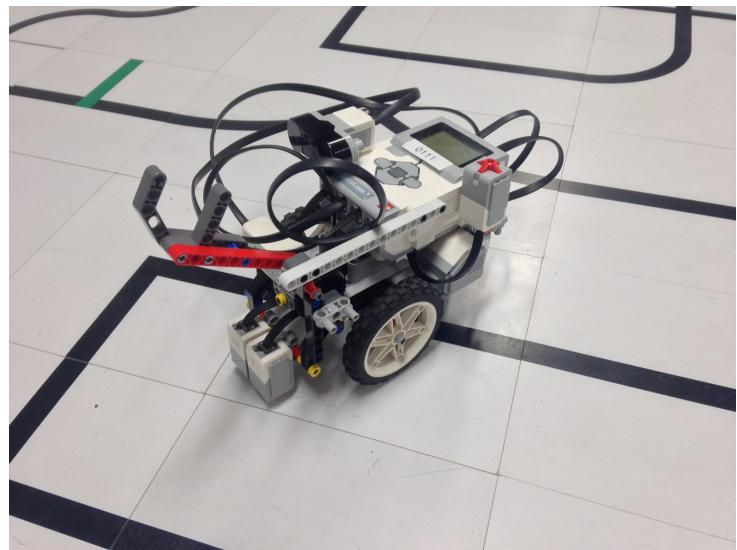
Dla realizacji zadania *Transporter* zamontowaliśmy średni serwomechanizm z boku robota.

2.2 Czujniki zastosowane

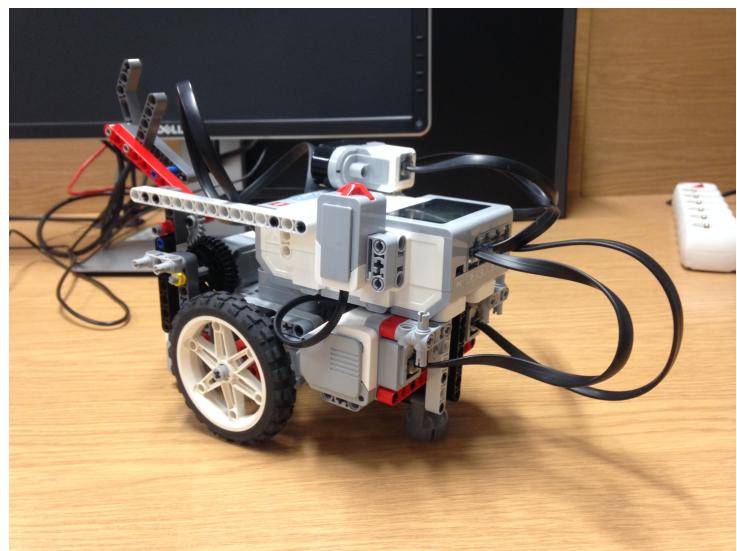
Do wykrywania czarnej linii zastosowaliśmy dwa czujniki koloru. Oba czujniki zapewniają odświeżanie o częstotliwości 1kHz. Umieściliśmy czujniki około 5mm od podłogi na wysięgniku, są maksymalnie blisko, oba pokrywają linię.

Do wykrywania obiektu zastosowaliśmy czujnik podczerwieni. Umieściliśmy dość nisko, żeby mógł wykryć obiekt.

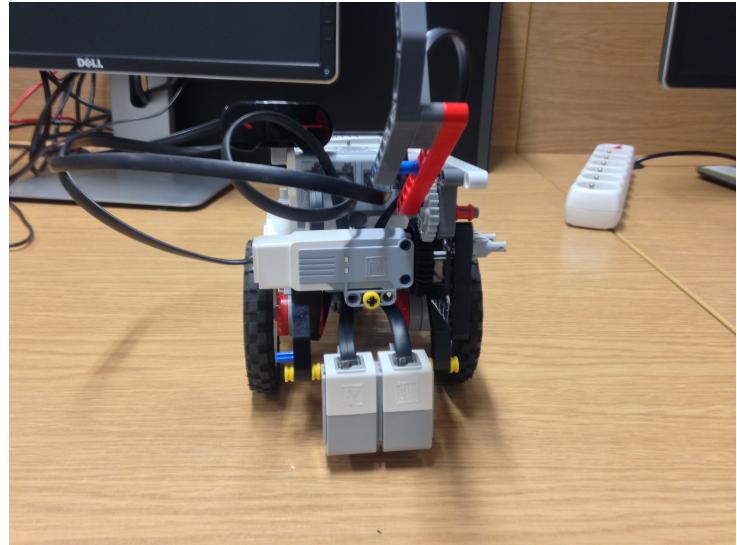
Umieściliśmy czujnik dotykowy na górze pojazdu jako sygnał uruchomienia robota oraz zatrzymania go w wybranym momencie.



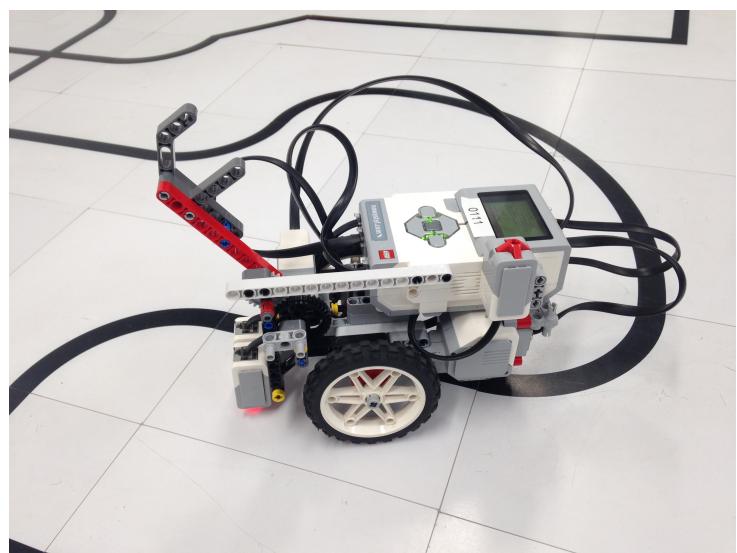
Rysunek 2.1: Widok z góry. Na zdjęciu dokładnie widać mikrokomputer .



Rysunek 2.2: Widok z boku. Dokładnie widać serwomechanizmy i koło sferyczne i podnośnik .



Rysunek 2.3: Widok z przodu. Dokładnie widać użyte czujniki .



Rysunek 2.4: Widok na planszy.

Rozdział 3

Algorytm sterowania

3.1 Kalibracja czujników

Do realizacji zadania otrzymaliśmy dwa czujniki koloru. Każdy z nich ma z dołu emiter światła oraz odbiornik. Czujniki wysyłają wiązkę czerwonego światła, które odbite od powierzchni powraca do odbiornika. Zakres czujników koloru to ok. 6 (dla czarnego) i ok. 45 (dla białego).

Stworzyliśmy funkcję, do odczytania wartości czarnej i białej dla obu czujników, która obraca robota symetrycznie w jedną i drugą stronę, zapisuje największą i najmniejszą odczytaną wartość - odpowiednio biały i czarny. I ustaliliśmy wartości krytyczne.

3.2 Line Follower

Zdecydowaliśmy zastosować schemat klasycznego regulatora proporcjonalno-całkującąco-różniczkującego. Całka zamienia się w sumę, różniczka w różnicę.

Zastosowaliśmy dwa regulatory PID, których działanie sumuje się na obu silnikach.

Przez konstrukcję robota regulator PID jest zbyt słaby, więc zdecydowaliśmy napisać funkcję, aby pojazd pokonywał zakręty większe niż 45 stopni. Funkcja wykrywa gdy robot całkowicie zjedzie z linii, następnie obróci się ok. 135 stopni, dopóki nie wykryje linii. Jeśli nie wykryje linii - skręci ponownie z większą prędkością. Będzie skrępiał tak długo, dopóki nie wykryje linię.

3.2.1 Dobieranie parametrów algorytmu

Parametry regulatora PID ustalaliśmy na początku metodą Zieglera-Nicholsa. Samo to nie wystarczało do pokonywania kątów prostych, więc zdecydowa-

Liśmy się na napisanie funkcji sterującej pojazdem w przypadku zjechania z czarnej linii.

3.3 Algorytm transportera

Dla zrealizowania zadania z transportem rozpisaliśmy poszczególne stany w których realizujemy algorytm transportera. Główna funkcja jechania po linii jest użyta z poprzedniego zadania, ale z od nowa dobieranymi parametrami PID. W tym zadaniu korzystaliśmy z dwóch czujników -czujnik koloru i wykrycia światła.Dla lepszego rozpoznawania koloru wartości z czujnika koloru odczytujemy w postaci wektora RGB;

Opis stanów:

- 0 - szukanie bazy z przedmiotem
- 1 - znalezienia bazy z przedmiotem, podnosimy przedmiot (zielonej)
- 2 - wyjechanie z bazy z przedmiotem
- 3 - szukanie bazy docelowej (czerwonej)
- 4 - znalezienie bazy docelowej, odkładanie przedmiotu
- 5 - wyjazd z bazy docelowej

```
1 elif (stateMode == 0 and (rR < 50 and gR > 150 and bR < 50)):  
2     straightRide(1)  
3     rotateRobotSym(90, True, 80)  
4     straightRide(1)  
5     stateMode = 1  
6 elif(stateMode == 1 and (rR < 50 and gR > 150 and bR < 50)):  
7     straightRide(2)  
8     takeCargo()  
9     sleep(2)  
10    rotateRobotSym(210, True, 80)  
11    stateMode = 2  
12 elif(stateMode == 2 and (rR < 50 and gR > 150 and bR < 50)):  
13     straightRide(2)  
14     rotateRobotSym(90, True, 80)  
15     stateMode = 3  
16 elif(stateMode == 3 and (rR > 155 and gR < 70 and bR < 20)):  
17     straightRide(1)  
18     rotateRobotSym(90, True, 80)  
19     straightRide(1)  
20     stateMode = 4
```

```
22 elif(stateMode == 4 and (rR > 155 and gR < 70 and bR < 20)):  
23     straightRide(2)  
24     leaveCargo()  
25     sleep(2)  
26     rotateRobotSym(210, True, 80)  
27     stateMode = 5  
28 elif(stateMode == 5 and (rR > 155 and gR < 70 and bR < 20)):  
29     straightRide(2)  
30     rotateRobotSym(90, True, 80)  
31     stateMode = 0  
32  
33
```

Rozdział 4

Podsumowania

Udało nam się dokonać postawione zadanie. Ale w naszym robocie można wyróżnić jak i wady, tak i zalety. Robot jest mobilny i bardzo dokładnie rucha się po linii pokonując całą trasę, ale widoczne są niewielkie oscylacje.

Nie udało się do końca robotu wykonać zadanie z transportem przedmiotu z powodu nie do końca dobrze skalibrowanych czujników, robot nie wykrył czerwonej linii i nie mógł skończyć zadanie. Ale po dokładniejszej kalibracji czujników, jak w przypadku z kolorem zielonym, udało się wypełnić zadanie.