

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

Dzmitry Kuksik , Antonina Lobach

20-01-2017

# Spis treści

<b>1</b>	<b>Treść zadania</b>	<b>2</b>
1.1	Podążanie wzdłuż linii(Linefollower) . . . . .	2
1.2	Transporter . . . . .	2
<b>2</b>	<b>Budowa robota</b>	<b>3</b>
2.1	Elementy wykonawcze . . . . .	3
2.2	Czujniki zastosowane . . . . .	3
<b>3</b>	<b>Algorytm sterowania</b>	<b>4</b>
3.1	Kalibracja czujników . . . . .	4
3.2	LineFollower . . . . .	4
3.2.1	Dobieranie parametrów algorytmu . . . . .	4
3.3	Transporter . . . . .	5

# 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.

TUT BUDUT FOTKI

## 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 LineFollower

Zdecydowaliśmy zastosować schemat klasycznego regulatora proporcjonalno-całkująco-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 skręcał tak długo, dopóki nie wykryje linii.

#### 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-

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

### **3.3   Transporter**