

Social distancing z robotem - zdalne sterowanie



Autorzy: Aleksandra Walczybok, Zofia Książak, Tomasz Chyrowicz, Marcin Kurka

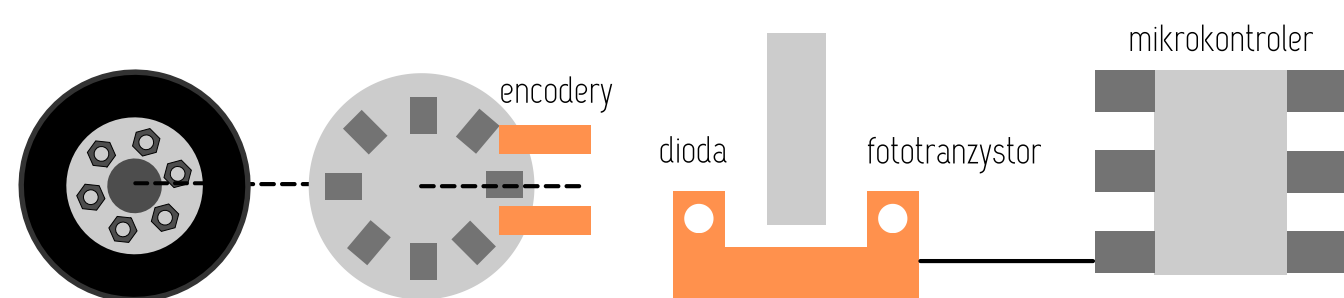
WPROWADZENIE

Za cel naszego projektu postawiliśmy sobie implementację zdalnego sterowania w mobilnym Almubocie. Aby to osiągnąć, zrealizowaliśmy dwa moduły:

- komunikacji bezprzewodowej
- odometryczny (nawigacyjny).

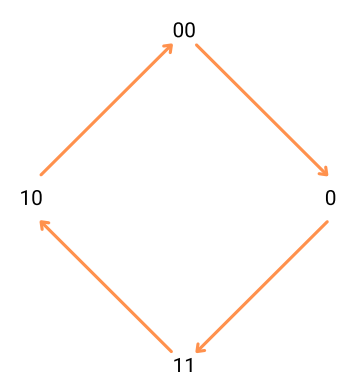
MODUŁ ODOMETRYCZNY

Moduł ten jest odpowiedzialny za zliczanie impulsów z encoder'ów, zamontowanych na osi napędowej kółek, a następnie za obliczanie aktualnej lokalizacji (pozycji oraz orientacji) na tej podstawie.



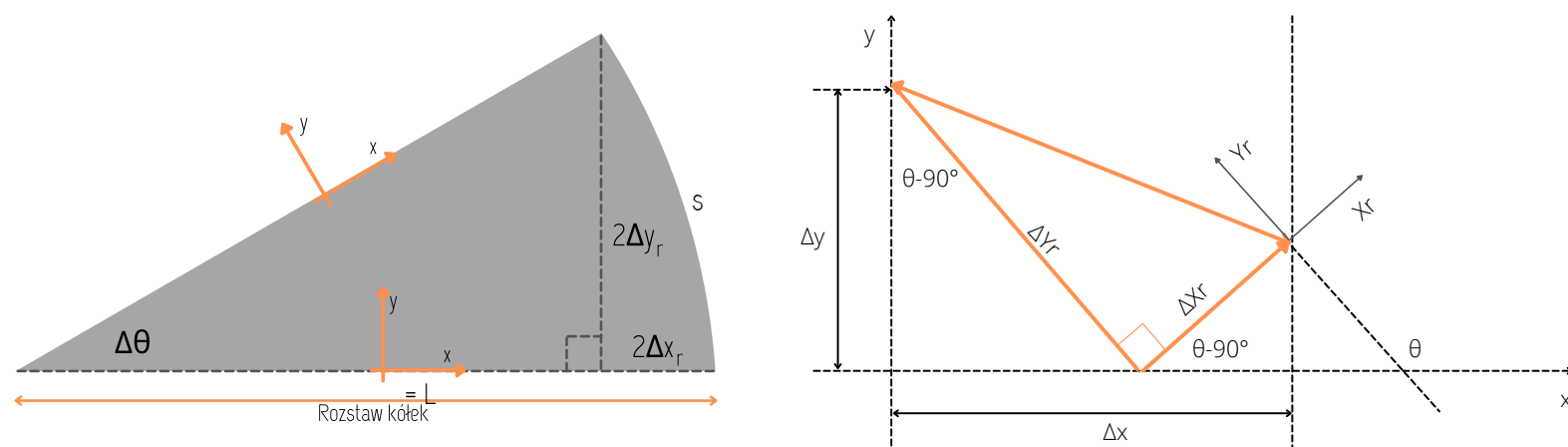
Fototranzystor rejestruje strumień światła przechodzący przez szczeliny tarczy encoder'a, co powoduje zmianę napięcia na wyjściu encoder'a. Skutkuje to generowaniem przerwania na kontrolerze. Wewnątrz procedury obsługi przerwania aktualizowana jest wartość położenia.

W celu określenia kierunku obrotu używamy po dwa encoder'y na stronę.



Aktualizacja pozycji robota składa się z dwóch kroków:

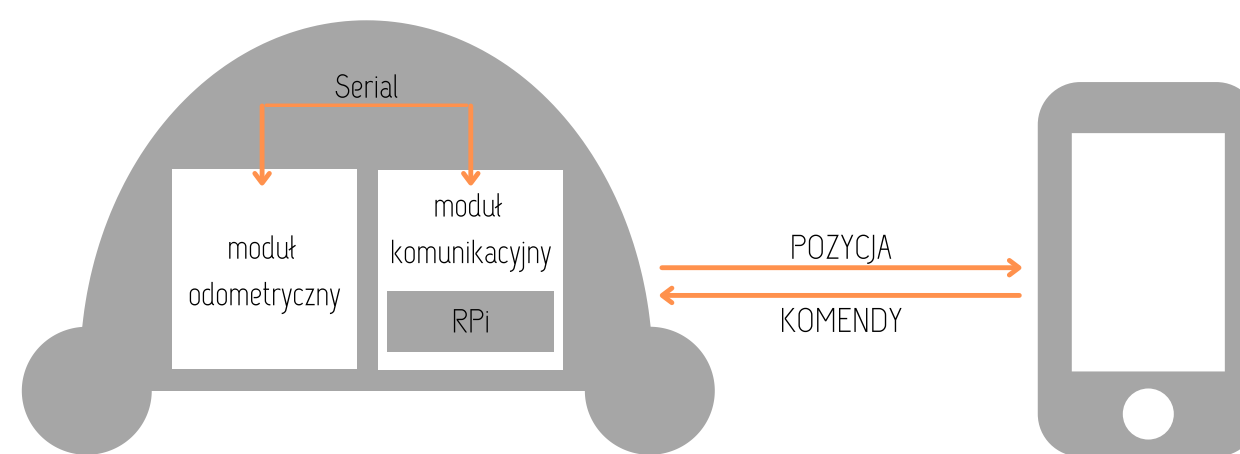
1. Obliczenie przesunięcia względem układu współrzędnych robota dla pojedynczych impulsów.
2. Przeliczenie przesunięcia z układów współrzędnych robota na układ współrzędnych względem punktu początkowego.



MODUŁ KOMUNIKACYJNY

Druga część naszego projektu opierała się na realizacji komunikacji przewodowej. Pierwszy krok polegał na tym, że Almubot odbierał pojedynczy znak, reprezentujący komendę, a następnie wykonywał akcję lub przysyłał wynik pomiarów, takich jak aktualna pozycja.

W drugim kroku połączyliśmy zapytania http z jednoznakowymi komendami poprzez implementację serwera www napisanego w Pythonie.



Przykładowe ścieżki działania:

1. Po wciśnięciu przycisku "stop" przez użytkownika wysyłane jest żądanie typu POST do serwera www, które następnie jest tłumaczone na jednoznakową komendę "q" i przesyłane do kontrolera bota, który ją interpretuje i zatrzymuje oba kółka.
2. Co sekundę przeglądarka wysyła żądanie typu GET do serwera www, które następnie jest tłumaczone na jednoznakową komendę "i" oraz przesyłane do kontrolera bota, który ją interpretuje i odsyła aktualną pozycję. Otrzymane dane są tłumaczone na odpowiedź w formacie JSON, odbierane przez przeglądarkę, a następnie prezentowane na stronie www u użytkownika.

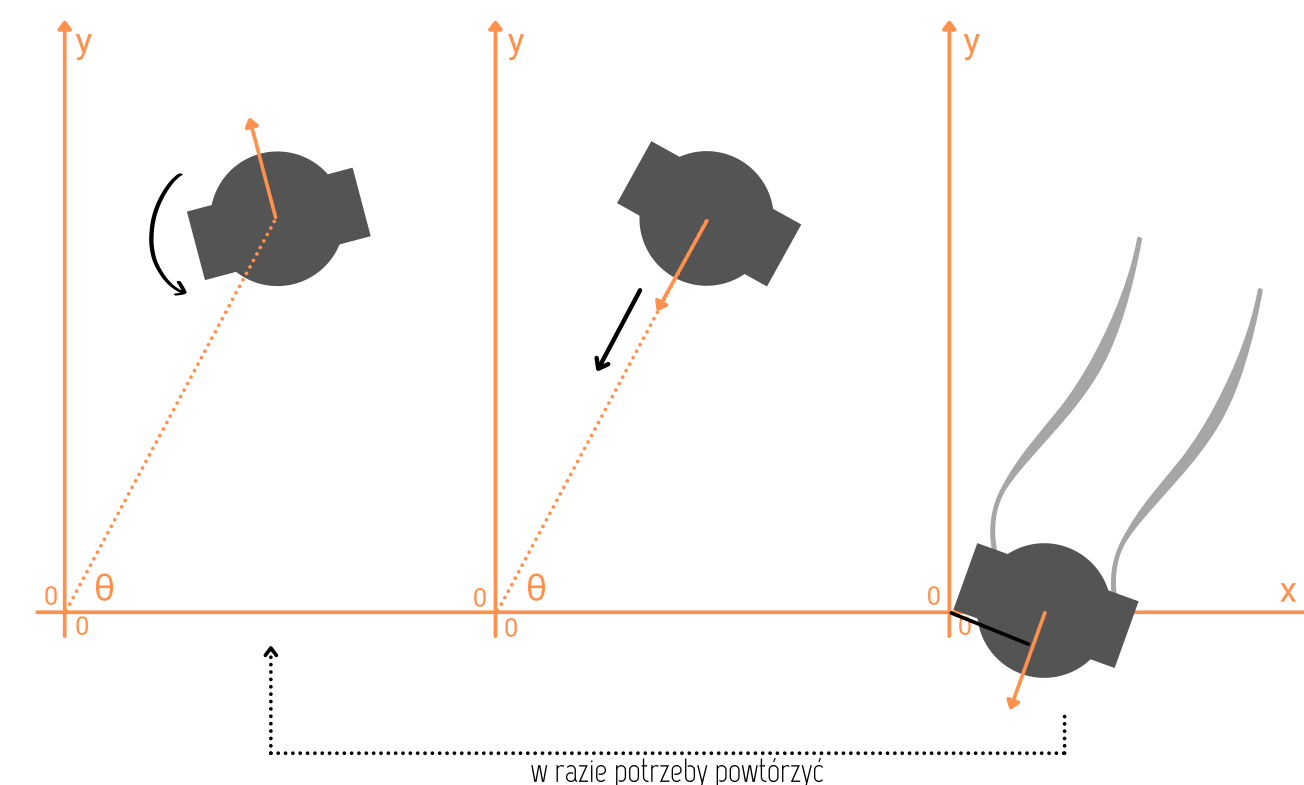
$$\Delta\theta = \frac{s}{L}$$
$$\Delta x_r = \frac{L \cdot (1 - \cos(\Delta\theta))}{2}$$
$$\Delta y_r = L \cdot \Delta\theta$$

$$\Delta x = \sin(\theta)\Delta x_r - \cos(\theta)\Delta y_r$$
$$\Delta y = \cos(\theta)\Delta x_r + \sin(\theta)\Delta y_r$$

'GO HOME' - dodatkowa funkcjonalność

Na koniec zdecydowaliśmy się dodać funkcję "go home" - powrotu do pozycji startowej. Zaimplementowaliśmy następujący algorytm:

1. Ustawienie robota w kierunku początku układu współrzędnych.
2. Jazda po prostej do osiągnięcia minimum odległości od początku układu współrzędnych - punktu (0, 0).
3. Sprawdzenie, czy robot dotarł w wyznaczone miejsce. Jeśli nie, powtórzenie algorytmu od punktu pierwszego.
4. Przywrócenie pierwotnej orientacji robota.



Wyniki

Udało nam się osiągnąć wyznaczony sobie cel - zaimplementowaliśmy zdalne sterowanie, więc robot reaguje na przyciśnięcie przycisków na stronie www wywołując odpowiednią akcję - jazda w przód, tył, skręt w prawo lub lewo, zatrzymanie, zresetowanie współrzędnych, funkcja 'go home'.

Posteruj Almubotem!

WIFI: Almubot

HASŁO: almuboty

