

Maze solver

Introducere

Proiectul presupune realizarea unui robot care rezolva un labirint fara a avea informatii despre acesta. Robotul are la baza o placa Arduino care controleaza 2 motoare in functie de senzorii de proximitate si ultima miscare efectuata. In plus, user-ul poate interveni cu o sursa de lumina pentru a dirija robotul pe traseul dorit, evitand in acelasi timp peretii labirintului.

Proiectul este inspirat din cunoscutul joc "Ajuta soarecele sa gaseasca cascavalul" si are ca scop prezentarea unei variante de rezolvare a unui labirint indiferent de dimensiunea/dificultatea acestuia.

Descriere generală

Masinuta are atasati in partea frontala 3 senzori de proximitate HC-SR04 pentru a determina obstacolele si 2 fotorezistente care stabilesc directia sursei de lumina.

Principiul de functionare a masinutei este urmatorul:

- In cazul in care senzorii ultrasonici din lateralul masinii detecteaza peretii, atunci aceasta inainteaza
- In cazul in care toti senzorii ultrasonici detecteaza peretii, atunci masina executa manevra de intoarcere deoarece acest se afla la un capat de drum
- In cazul in care niciun senzor ultrasonic NU detecteaza peretii, atunci masina va efectua un viraj la dreapta, deoarece se afla intr-o intersectie

Utilizatorul are posibilitatea de a interveni in rezolvarea labirintului prin folosirea unei surse de lumina (ex: lanterna) care sa directioneze masina pe ruta dorita.



Hardware Design

Lista de piese:

- Arduino UNO ATMEGA328P
- Senzori Ultrasonici HC-SR04
- Fotorezistente

- 2 motoare electrice
- Rezistente
- Baterie
- Un buton
- Led-uri
- Buton

Schema Electrica:

Design-ul circuitului:



Software Design

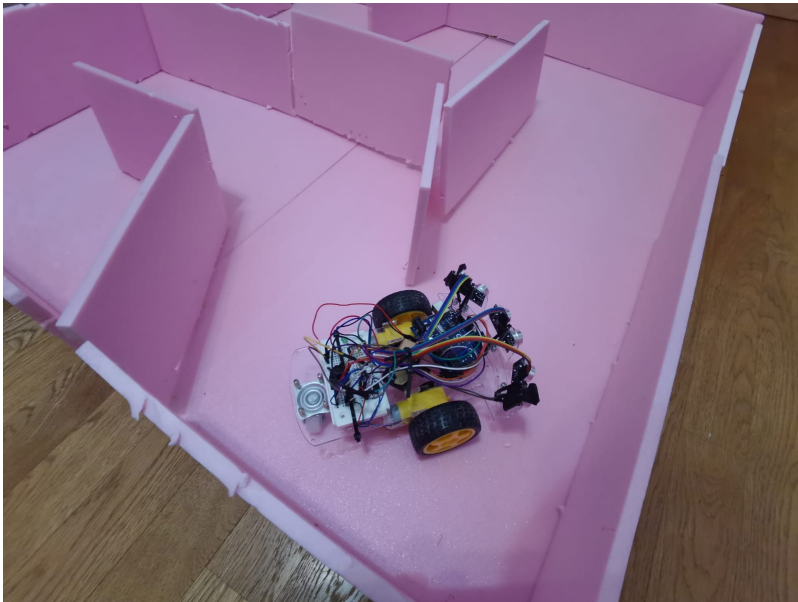
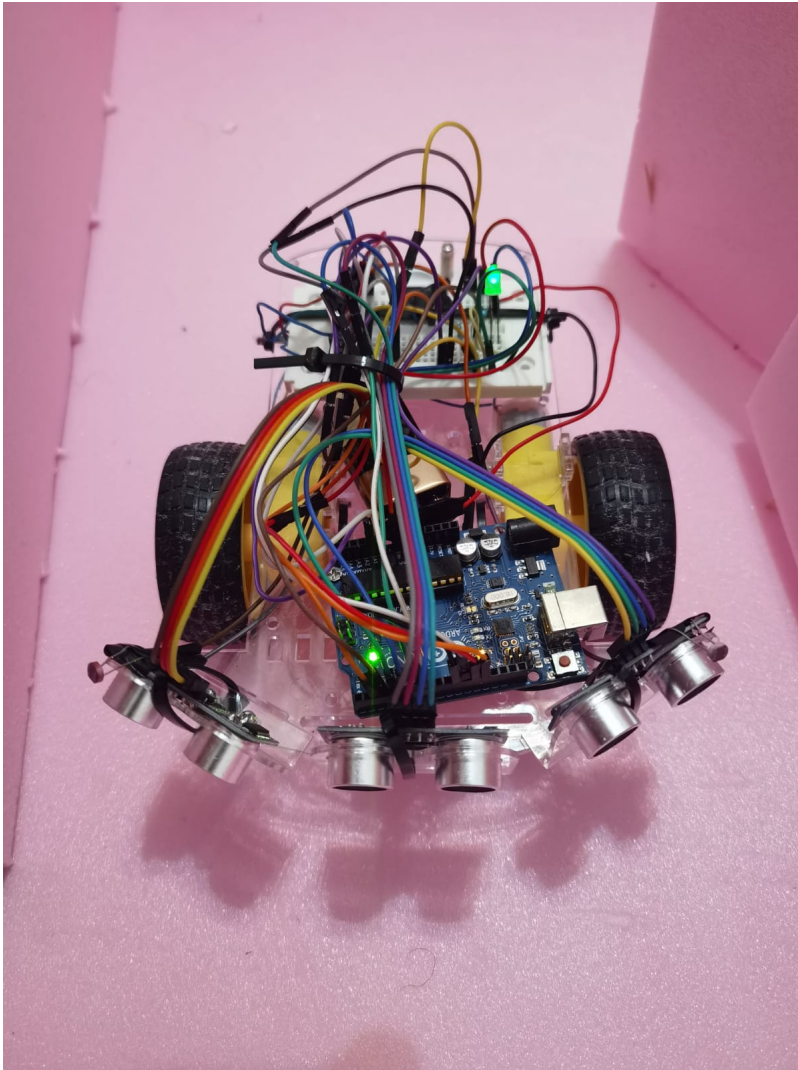
Proiectul este realizat utilizand mediul de dezvoltare Arduino IDE.

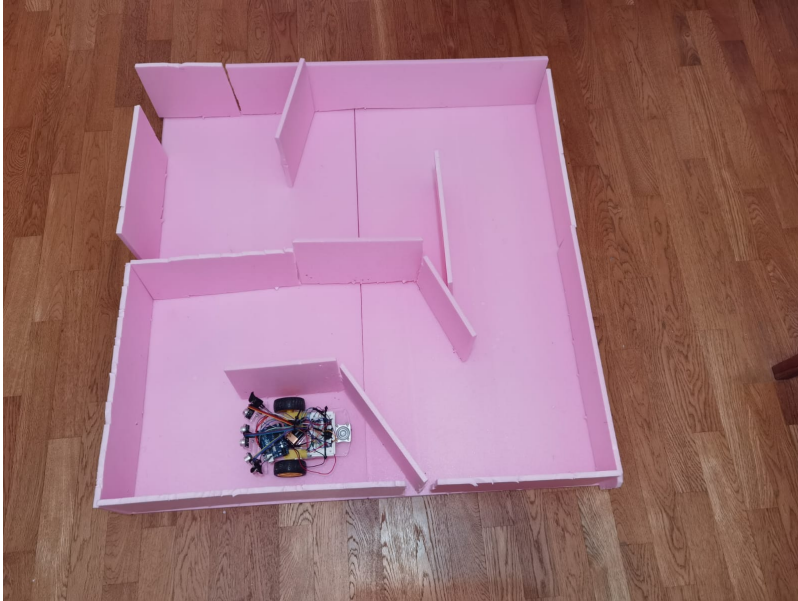
Descrierea codului aplicației:

`void setup()` - se configureaza intreruperile hardware si citirea analogica impreuna cu calibrarea senzorilor de lumina: `ISR(INT0_vect)`: intrerupere generata la apasarea butonului PD2 pentru finalizarea procesului de calibrare a senzorilor de lumina. `setupADC()` - configureaza citirea analogica de pe pinii PA0 si PA1 pentru fotorezistente. Cat timp robotul de afla in modul de calibrare, se calculeaza minimul si maximul indicati de senzorii de lumina.

`void loop()` - se verifica nivelul de luminozitate, iar in cazul in care se depaseste un prag, vitezele motoarelor se vor seta direct proportional cu valorile citite de la senzori, urmarind astfel sursa de lumina. In acelasi timp, se calculeaza distanta indicata de fiecare senzor de proximitate (`getDistLeft()`, `getDistMid()`, `getDistRight()`) si in functie de acestia, robotul va evita orice obstacol. Pentru rezolvarea labirintului, robotul va vira mereu la dreapta atunci cand este posibil.

Rezultate Obținute





Concluzii

Acest proiect a reprezentat o oportunitate pentru a pune in aplicare cunostintele dobandite la laboratoare. Modul de abordare pentru a imbina partea hardware cu cea software si de a le folosi la o capacitate cat mai mare, m-a ajutat sa realizez cat de importanta este proiectarea atenta a proiectului inainte de implementare. Pentru o solutie mai eficienta in rezolvarea labirintului ar fi fost necesar un sasiu mai mic/mai usor de manevrat in viraje.

Download

[pm_maze_solver.zip](#)

Jurnal

- 04.05 - Scrierea documentatiei
- 21.05 - Adaugare Hardware design
- 28.05 - Adaugare Software design

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2023/apredescu/maze_solver



Last update: **2023/05/30 06:04**