



Proiectare cu microprocesoare

Laboratory activity

Name: Rosca Alexandru & Nechifor Cosmin

Group: 30232

Email: alex.rosca1996@gmail.com & cosmin.nechifor96@gmail.com



Contents

1	Documentatie	3
2	Source Code	9

Chapter 1

Documentatie

Introducere

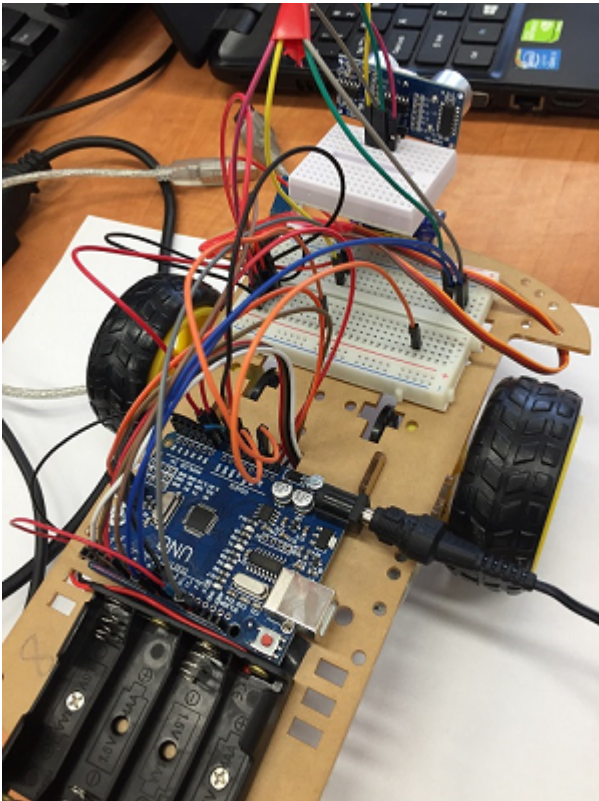
În cadrul proiectului la Proiectare cu MicroProcesoare am ales să implementăm un robot al cărui scop este să curețe casa de agenții dăunători sanatații.

Structura robotului

Robotul este format din următoarele componente:

- *Placa microcontroller compatibilă Arduino Uno*
- *Driver motoare L298N Dual H-Bridge*
- *2x Motor DC* - convertesc energia electrică în lucru mecanic. Viteza de rotație a unui motor este proporțională cu tensiunea de alimentare de la bornele acestuia, iar direcția de rotație depinde de polaritate (conectarea celor 2 fire de alimentare ale motorului la +Vcc și Gnd, sau vice-versa).
- *1 Motor Servo* - sunt folosite pentru a obține rotații parțiale, stabile și controlate, pentru efectuarea unor operații cu amplitudine mică dar cu precizie ridicată: acționare mecanism de închidere-deschidere, poziționare senzori, efectuarea unor gesturi, etc
- *Carcasa baterii 4xAA (R6)*
- *2 roți conectate la motoare, 1 roată mobilă*
- *Două plăci prototipizate*
- *1 Senzor sonar (HC-SR04)* - pentru a detecta distanța senzorul emite un ultrasunet la frecvența de 40000 Hz care merge prin aer și dacă un obiect se găsește pe direcția de propagare acesta va fi reflectat, știind timpul dintre emisie și recepție a semnalului și viteza de deplasare a sunetului putem afla distanța
- *1 Senzor lumină* - sesizează nivelul de iluminare al mediului, valoarea iluminării variind liniar

Montaj:



Cum functioneaza?

La inceputul drumului merge inainte folosindu-se de motoarele DC, iar daca detecteaza un obstacol in fata lui, atunci se opreste pentru a verifica daca in stanga sau in dreapta lui exista un drum liber.

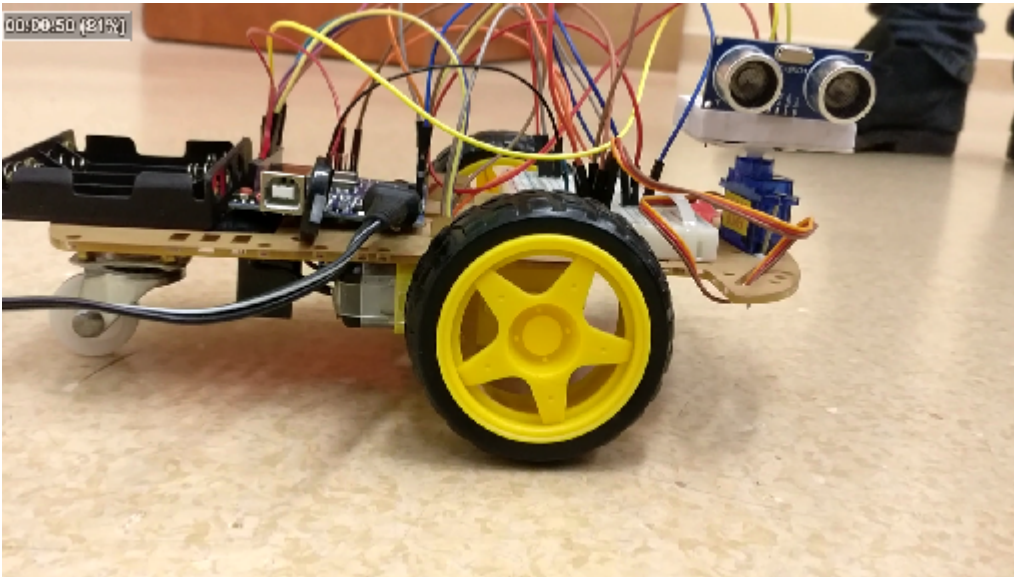
Acest lucru este realizat folosindu-se de sonar si de motorul servo. Servo-ul orienteaza sonarul la stanga si la dreapta pentru verifica unde se afla cea mai mare distanta. Odata aflata directia in care distantaeste cea mai mare, robotul se va orienta spre directia respectiva.

Iar pentru partea de "aspirat" se foloseste de senzorul de luminozitate reuseste sa identifice zonele murdare, datorita anomaliilor din spatiul respectiv. Presupunerea noastra este ca spatiile murdare sunt negre ca atunci cand senzorul de lumina trece peste ele sa le detezeze.

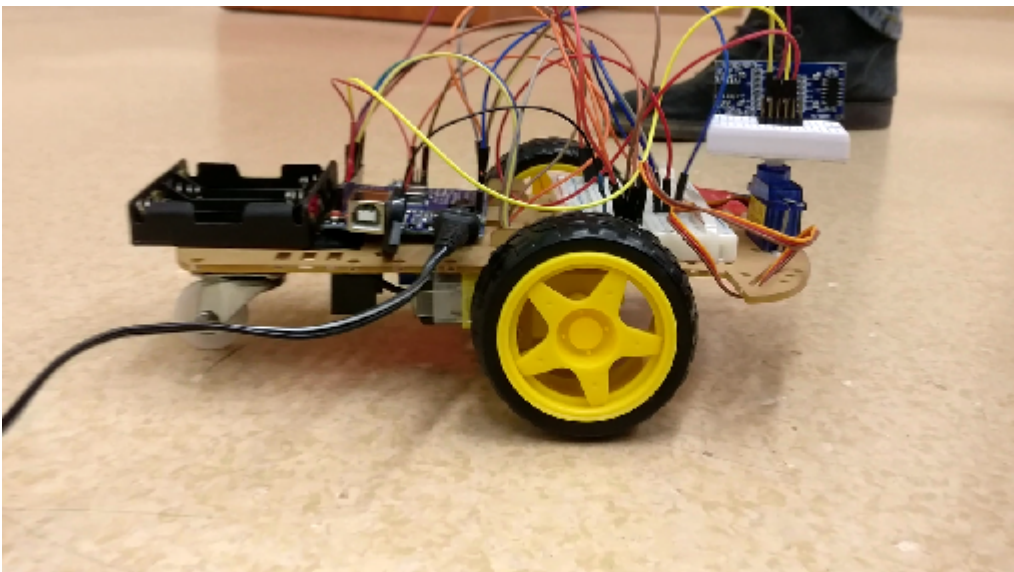
Dupa ce termina de aspirat, verifica iarasi daca in stanga si in dreapta lui se afla obstacole si alege directia corecta.

Demo Link

Position 1:



Position 2:



Dezvoltari ulterioare



Robotul nostru este departe de a fi perfect. Pentru a permite o mai bună recunoaștere a mediului, se mai pot adăuga senzori, cum ar fi:

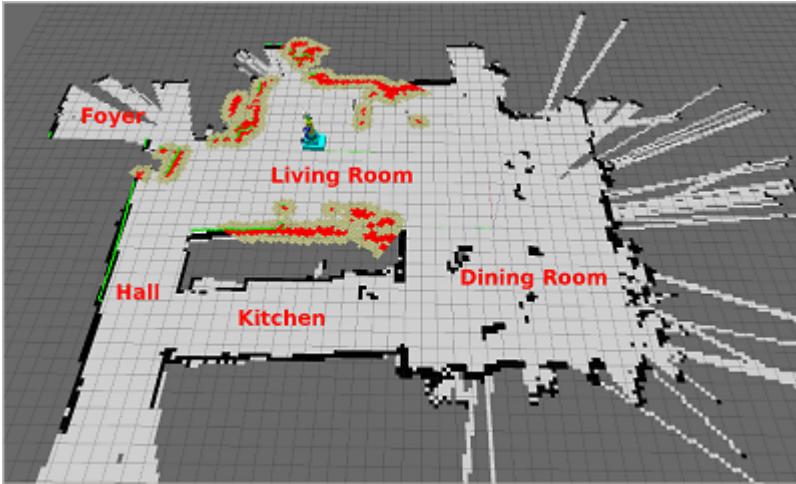
- temperatura
- mai multe sonare și lumina pentru o detecție mai bună a obstacolelor
- umiditate - să știe dacă o zonă conține apă, întrucât senzorul de lumină este incapabil să detecteze elementele transparente
- etc.

Se mai poate dezvolta algoritmul de evitare a obstacolelor astfel încât robotul să nu se poată bloca într-o încăpere.

Se poate înlocui sonarul cu o cameră web, care să permită procesarea imaginilor din jur și detectarea obiectelor și obstacolelor.



Avand aceasta camera robotul ar putea sa memoreze structura spatiului in care el trebuie sa aspire (ex: structura casei, camerei...) si sa optimizeze modul de deplasare astfel incat sa se reduca timpul destinat curateniei si eficientizarea consumului de energie.



Daca se merge la un nivel mai inalt, folosind *reinforcement learning* robotul poate atat de inteligent incat sa nu mai necesite omul sa-l porneasca sau sa-l pozitioneze in o camera pentru a aspira, cu alte cuvinte am avea astfel utilajul de curatenie perfect.

Procesul de invatare poate fi stimulat daca se introduce un modul wifi, permitand robotului sa se conecteze la internet si sa schimbe datele cu ceilalti roboti aspiratori.



Bibliografie

- Laboratoare
- Smart vacuum Photo
- P2P Photo
- Map photo
- Robot with camera

Chapter 2

Source Code

Github source code

Bibliography

