

Surveillance Car

Nume: Coman Calin-Alexandru
Grupa: 331CAb

Introducere

Proiectul consta intr-o masina autonoma care poate detecta obstacole astfel incat sa le evite si poate detecta si intersectiile, pe care sa le urmeze. In acelasi timp, masina transmite live ceea ce filmeaza si avertizeaza prin semnale luminoase si acustice atunci cand intalneste un obstacol si trebuie sa-si schimbe traseul.

Scopul masinii este de a monitoriza un perimetru in timp real, fara a fi nevoie de controlul manual al utilizatorului. Initial am pornit de la ideea de a face un sistem static de monitorizare, dar apoi m-am gandit ca ar fi mai interesant sa adaug si componenta de autonomie. De asemenea, proiectul ar putea fi extins prin adaugarea unui sistem de recunoastere a obiectelor din stream-ul video.

Descriere generală

Pentru a putea detecta obstacolele, masina are montat in fata un senzor infrarosu sharp care este rotit de un servomotor intr-un perimetru de 180 de grade. De asemenea, tot pe masina este montata si o placuta ESP32 CAM, care transmite live filmarea prin WiFi catre un smartphone/laptop.

Functionalitatea dorita este urmatoarea:
Initial senzorul IR este indreptat in fata.

- Daca acesta detecteaza un obstacol, se roteste la 90 de grade in stanga si in dreapta. Apoi masina o ia in directia unde distanta pana la obstacol e cea mai mare.
- Daca nu detecteaza un obstacol, inseamna ca exista o posibilitate sa fim intr-o intersectie, asa ca senzorul se roteste iar la 90 de grade in stanga si in dreapta. Daca intr-o directie nu detecteaza nimic si in cealalta da, o ia in prima directie. Daca ambele directii sunt libere, o ia in stanga, in dreapta sau inainte, in mod aleatoriu. Daca ambele directii sunt ocupate de un obstacol, o ia inainte.

Atunci cand masina detecteaza un obstacol in fata, va aprinde un led RGB in rosu.



Hardware Design

- Arduino UNO R3 ATmega328P
- ESP32 cu OV2640 camera
- Servomotor SG-90
- GP2Y0A21YK0F 10-80 cm Infrared Distance Sensor
- Sasiu Masina + Roti
- L293D Motor Driver
- 2 motoare electrice
- 2 x Acumulator 18650 3.7V
- Led RGB
- Regulator de tensiune LM2596
- Buton
- Rezistente
- Breadboard
- Isolierband
- Banda dublu adeziva

O problema pe care am intalnit-o la partea de hardware a fost curentul insuficient la alimentare. Initial am folosit o baterie de 9V, dar aceasta s-a dovedit a fi ineficienta pentru a duce sarcina tuturor componentelor (din cauza curentului redus cateodata se resetau placutele Arduino si ESP32). In final, dupa niste research, am hotarat sa folosesc doua acumulatori 18650 de 3.7V 2500mAh legate in serie, care s-au dovedit a fi mai mult decat destule pentru alimentarea masinii. De asemenea, cum OUT-ul de 5V de la Arduino nu avea destul curent pentru a alimenta ESP32-ul si celelalte componente in acelasi timp, am folosit un regulator de tensiune LM2596 pentru a reduce tensiunea de 7.4V de la acumulatori la 5V (input pentru ESP32).

Pentru a controla motoarele am ales sa folosesc driver-ul L293D simplu (nu shield), deoarece shield-ul imi ocupa toti pinii digitali. Folosind acest driver, am putut sa setez atat sensul, cat si viteza motoarelor (cu PWM).

Schema electrica realizata in Eagle:



Software Design

Proiectul a fost realizat folosind Arduino IDE.

Bibliotecile folosite sunt:

- Servo.h (<https://github.com/arduino-libraries/Servo>) - pentru a putea comanda servomotorul SG-90
- SharpIR.h (<https://github.com/guillaume-rico/SharpIR>) - pentru a prelua datele de la senzorul infrarosu GP2Y0A21YK0F
- WiFi.h (<https://github.com/espressif/arduino-esp32/blob/master/libraries/WiFi/src/WiFi.h>) - pentru a interactiona cu modulul de WiFi de pe ESP32 CAM

Limitarile si setback-urile principale la implementarea software au fost date de senzorul infrarosu. In primul rand, distanta maxima de recunoastere a unui obstacol de catre acest senzor este de 80 de cm,

deci obstacolul trebuie sa se afle relativ aproape pentru ca masina sa-l poata detecta si deci sa il evite. In al doilea rand, desi in teorie este mai exact decat un senzor ultrasonic, am descoperit ca acest senzor IR are destul de multe spike-uri si intoarce des distante care nu sunt reale. Pentru acest lucru am scris functia `getDistance()`, care in loc sa citeasca o singura data output-ul analogic, citeste 25 de valori succesive si foloseste media lor aritmetica.

De asemenea, o parte destul de dificila a fost potrivirea valorilor PWM trimise motoarelor (adica viteza) cu timpii de intoarcere pentru a face miscarea masinii naturala.

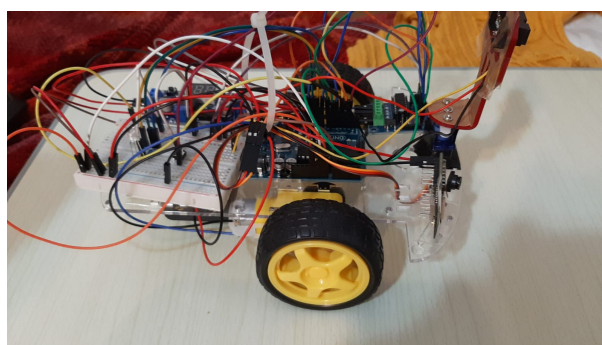
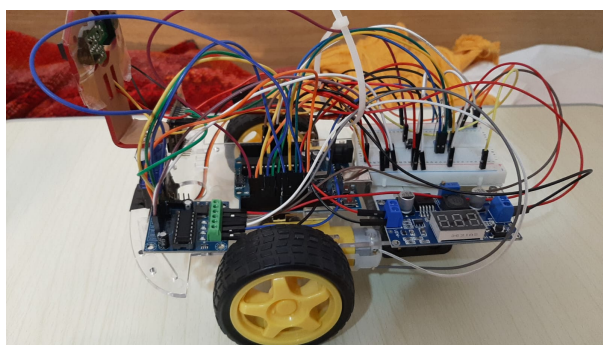
Descrierea codului pe Arduino:

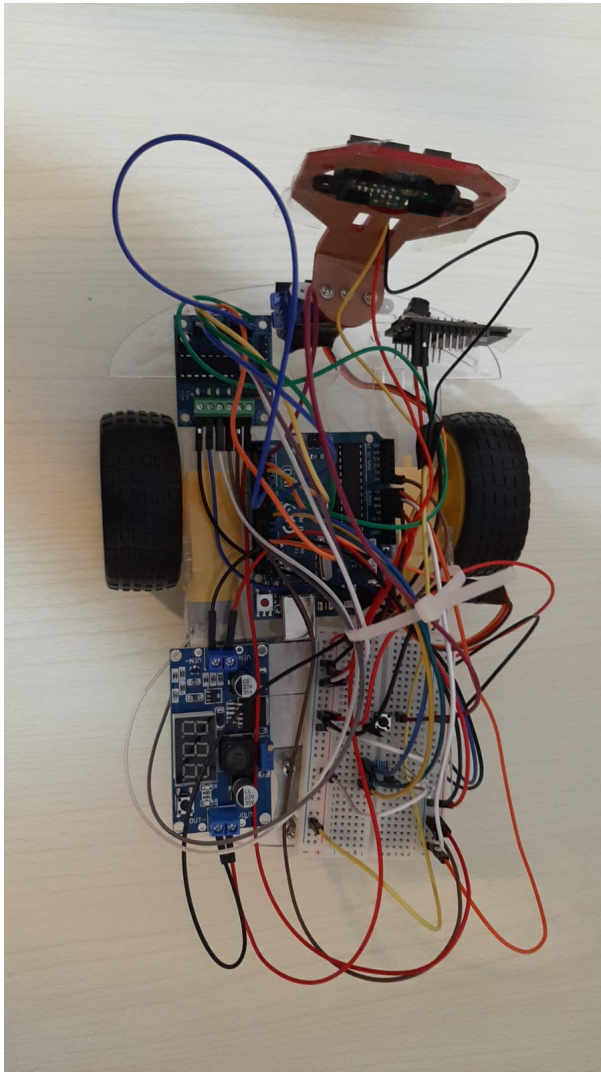
- In functia de `setup()` se seteaza toti pinii de output si input (senzorul IR, driver-ul de motoare, etc) si se ataseaza unui buton o intrerupere care va declansa pornirea masinii. Odata ce masina este alimentata, aceasta asteapta apasarea butonului, iar intreuperea generata declanseaza startul.
- In functia de `loop()` este urmatorul flow: Senzorul IR se uita intai in fata. Daca niciun obstacol nu este detectat, masina merge in linie dreapta. Daca avem un obstacol in fata, senzorul se uita si in stanga si in dreapta si masina alege sa mearga pe calea unde obstacolul este cel mai departat. Daca si in stanga si in dreapta obstacolele sunt aproape, masina se intoarce 180 de grade pentru a iesi din impas. Atunci cand se opreste, LED-ul RGB devine rosu, cand merge, este verde, si cand schimba directia este albastru.

Pentru a incarca cod pe placuta ESP32 CAM, am folosit Arduino-ul ca intermediar, deoarece aceasta nu are un port dedicat. Din cod se configureaza ca modulul WiFi sa fie folosit pentru a crea un AP (Access Point), astfel orice device se poate conecta la retea generata de placuta pentru a vedea imaginea live.

Laboratoarele utilizate sunt: Intreruperi, ADC, PWM.

Rezultate Obținute





<https://youtu.be/y-nZaCMrqL0>

Concluzii

A fost interesant proiectul, mi-a placut ca a implicat si partea hardware, lucru care pentru mine cel putin clar nu a fost o treaba usoara. Eu sunt multumit de rezultat, dat fiind si ca am fost foarte presat cu timpul semestrul asta. Din toate probleme pe care le-am avut am invatat ca este esential sa gandesti bine un plan, inainte de a te apuca sa il pui si in aplicare.

Download

[331ca_comancalinalexandru_codproiectpm.zip](#)

Bibliografie/Resurse

<https://lastminuteengineers.com/l293d-dc-motor-arduino-tutorial/>

<https://www.instructables.com/How-to-Use-the-Sharp-IR-Sensor-GP2Y0A41SK0F-Arduin/>

<https://docs.arduino.cc/learn/electronics/servo-motors>

<https://randomnerdtutorials.com/program-upload-code-esp32-cam/>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/gpatru/calin.coman>



Last update: **2023/05/30 04:18**