

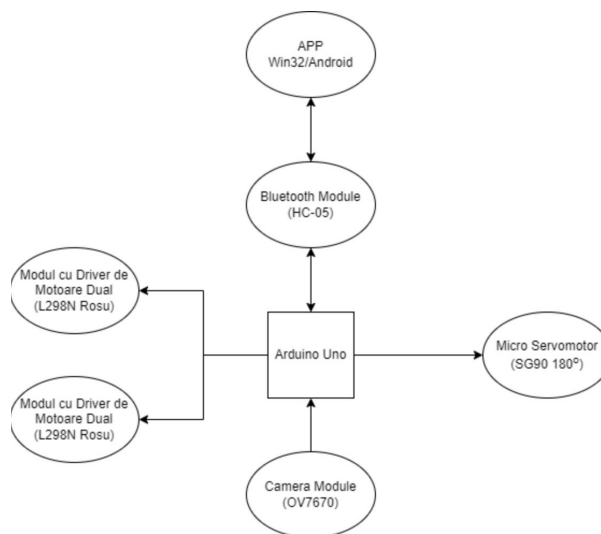
Robot Video

- Student: Judea Alexandru-Daniel
- Grupa: 341C4

Introducere

În acest proiect, se intenționează construirea un robot autonom care să supravegheze video într-o cameră, utilizând o cameră video și tehnologia de transmisie Bluetooth sau cardul SD. Robotul va fi capabil să se miște prin încăpere și va transmite feed-ul video către o aplicație suport, care va rula pe dispozitive Windows sau Android. De asemenea, robotul va filma doar când este în repaus, pentru a obține o calitate superioară a imaginii, din cauza frecvenței reduse de transmisie.

Descriere generală



Lista de piese

- Modul Camera OV7670
- Senzori Ultrasonici x2
- Modul cu Driver de Motoare Dual L298N Rosu
- Motoare x4
- Modul Bluetooth HC-05
- Modul Arduino Uno
- Acumulatori
- Fire de legatura

- Corp robot (roti, placute pentru suport, suruburi etc)

Interactiune module

În proiectul propus, Modulul Arduino Uno va fi elementul central al sistemului. Acesta va asigura comunicarea între celelalte module componente. Modulul Bluetooth HC-05 va fi responsabil de comunicarea între robot și aplicația suport. Acest modul va primi feed-ul video de la modulul Arduino și îl va transmite către aplicație, dar și va transmite comenzile primite de la aplicație înapoi către Arduino pentru procesare. Modulul de cameră va fi responsabil pentru captarea imaginilor din încăperea și transmiterea lor către Arduino și, ulterior, către aplicație. Modulele de Driver vor fi utilizate pentru a controla curentul livrat către motoarele care vor asigura mișcarea robotului. Toate conexiunile între module vor fi realizate prin intermediul firelor de legătură.

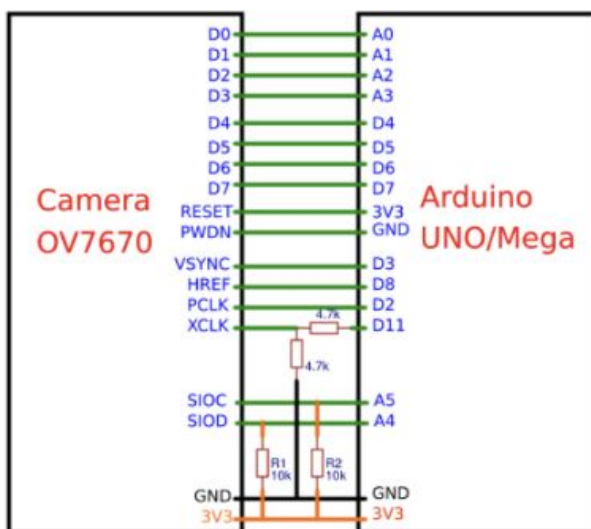
Hardware Design

Plăcuța Arduino primește o tensiune de 5V ca ieșire de la puntea H, care la rândul ei este alimentată cu 12V. Puntea H este conectată la Arduino prin pinii 2-7. Pinii (7;5) și (2;4) transmit tensiunea Low și High pentru fiecare motor, stabilind astfel direcția de deplasare (atunci când 7 și 4 sunt High, iar 5 și 2 sunt Low, motorul se va deplasa înainte). Pinii 6 și 3 reglează procentajul de tensiune transmisă către motoare. În mod normal, ajustarea tensiunii s-ar face pentru a schimba viteza de deplasare, dar în acest caz, este realizată pentru a asigura că tensiunea de alimentare nu este prea mare și motoarele nu se supraîncălzesc. Cei doi senzori ultrasonici sunt conectați la placuța prin pinii 8-11. Modulul Bluetooth, deoarece necesită comunicare serială, este conectat la pinii RX și TX.

Modulul Bluetooth este conectat la GND-ul de pe placuță, în timp ce celelalte componente sunt conectate la un "ground" global provenit de la sursa de alimentare.

A doua placuță Arduino este alimentată în mod similar cu prima. De la aceasta se alimentează camera OV7670 prin pin-ul de 3V3. Vor fi utilizate atât pinii digitali 2-8 și 11, cât și pinii analogici A0-5.

Schema electrica camera:



Software Design

Mediu de dezvoltare folosit:

- Unordered List Item Arduino IDE

Biblioteci folosite:

- SoftwareSerial.h
- stdint.h
- avr/io.h
- util/twi.h
- util/delay.h
- avr/pgmspace.h

Aplicatii ajutatoare:

- Se foloseste "Dabble" pentru comunicatia Bluetooth
- Se foloseste o aplicatie de vizualizat feed-ul video ce vine pe un port serial

Partea de software a fost complet dezvoltată folosind Arduino IDE. În ceea ce privește funcționalitatea de mișcare, există următoarele funcții disponibile:

- În funcția "setup", sunt setați pinii de intrare și ieșire (pinii 2-7 sunt configurați ca ieșire pentru motoare, iar pinii 8 și 10 sunt configurați ca intrare, iar pinii 9 și 11 ca ieșire pentru senzorii ultrasonici). De asemenea, se inițializează o variabilă "SoftwareSerial" pentru a configura comunicarea cu modulul Bluetooth.
- În funcția "mers_inainte", se trimite un semnal de Low și High pe pinii corespunzători fiecărui motor folosind funcția "digitalWrite", astfel încât deplasarea să se facă înainte. Utilizând funcția "analogWrite", se modifică tensiunea de alimentare și se ajustează viteza la o valoare potrivită. Apoi, se utilizează o întârziere de 400 de milisecunde ("delay") pentru a permite deplasarea.
- În funcția "mers_inapoi", aceasta este similară cu funcția anterioară, cu singura diferență că se inversează pinii High și Low între ei, astfel încât mișcarea să se facă în direcția opusă. Restul logicii și a utilizării funcțiilor "digitalWrite" și "analogWrite" rămân aceleași pentru ajustarea tensiunii și vitezei de deplasare, iar întârzierea "delay" de 400 de milisecunde este utilizată pentru a permite mișcarea.
- În funcția "rotire_stanga", se utilizează logica de rotire a tancurilor, unde roțile din stânga se mișcă într-o direcție, în timp ce cele din dreapta se mișcă în direcția opusă (ambele cu aceeași viteză), ceea ce produce o rotație pe loc. Sensul de rotație este determinat de roțile care se deplasează înainte, în acest caz fiind roțile din stânga. Utilizând funcțiile "digitalWrite" și "analogWrite", se controlează pinii corespunzători pentru a seta direcția și viteza de rotație. De asemenea, se utilizează o întârziere "delay" de 400 de milisecunde pentru a permite rotația.
- În funcția "rotire_dreapta", aceasta este similară cu funcția anterioară, conform logicii de rotire a tancurilor menționată anterior. În acest caz, sensul de rotație este determinat de roțile din dreapta care se deplasează înainte. Astfel, roțile din stânga se mișcă într-o direcție, în timp ce cele din dreapta se mișcă în direcția opusă, ambele având aceeași viteză. Prin intermediul funcțiilor "digitalWrite" și "analogWrite", se controlează pinii corespunzători pentru a seta direcția și viteza de rotație. Similar cu celelalte funcții, se utilizează și o întârziere "delay" de 400 de milisecunde pentru a permite rotația corespunzătoare.

- În funcția “resetare_roti”, pentru a opri mișcarea motoarelor, se setează pinii cu tensiunea High pe Low. Pentru a evita verificarea individuală a fiecărui pin în parte, toți pinii digitali utilizați pentru motoare sunt setați pe Low simultan. Aceasta asigură că toate motoarele se opresc, indiferent de starea inițială a pinilor.
- În funcția “senzor_fata”, se efectuează măsurători ale distanței față de primul obiect utilizând un senzor ultrasonic. În primul rând, se inițiază transmiterea ultrasunetului prin setarea tensiunii pe pinul de trigger pe High. Se așteaptă o perioadă de 10 microsecunde pentru ca sunetul să fie emis, după care se oprește transmiterea. Apoi, se preia valoarea de pe pinul de echo utilizând funcția “pulseIn”. Această valoare reprezintă durata totală de deplasare a sunetului între senzor și obiectul detectat. Utilizând o simplă regulă de trei simplă, se poate calcula distanța până la primul obiect bazată pe această durată măsurată.
- În funcția “senzor_spate”, funcționarea este similară cu funcția anterioară, dar se utilizează pinii corespunzători senzorului amplasat în partea din spate. Se inițiază transmiterea ultrasunetului prin setarea tensiunii pe pinul de trigger la nivel înalt (High). Se așteaptă o perioadă de 10 microsecunde pentru ca sunetul să fie emis, apoi se oprește transmiterea. Ulterior, se preia valoarea de pe pinul de echo utilizând funcția “pulseIn”. Această valoare reprezintă durata totală de deplasare a sunetului între senzor și obiectul detectat. Se poate calcula distanța până la primul obiect folosind o regulă simplă de trei, bazată pe această durată măsurată.
- În funcția “move_alone”, se permite robotului să se deplaseze “singur”. În primul rând, se verifică dacă poate avansa în față folosind senzorul din față. În cazul în care nu există obstacole în față, robotul se va deplasa înainte. În caz contrar, se va realiza o rotație de aproximativ 360 de grade până când nu mai există obstacole în fața sa. Dacă nu se găsește un spațiu liber, robotul se va deplasa în spate. Această logică permite robotului să se deplaseze autonom, evitând obstacolele și găsind un drum liber în fața sa sau reacționând în consecință în cazul în care nu există un spațiu disponibil.
- În funcția “loop”, se utilizează următoarea logică: se verifică dacă există vreun mesaj primit pe plăcuța Bluetooth. Dacă există un mesaj, se verifică conținutul acestuia și se aplică comanda primită doar în cazul în care este validă. În cazul în care s-a primit o comandă pentru oprirea mișcării, se realizează resetarea roților pentru a opri deplasarea robotului. În continuare, se verifică dacă robotul este setat în modul de mișcare autonomă. Dacă este așa, robotul se va deplasa “singur” folosind logica implementată în funcția “move_alone”. Această verificare se realizează de fiecare dată în bucla principală “loop”, permițând robotului să funcționeze în mod automat atunci când este setat în acest mod.

Pentru funcționalitatea de filmare, logica programului poate fi următoarea:

- Protocolul I2C/TWI este utilizat pentru transferul datelor de la camera OV7670 la plăcuța Arduino. Prin intermediul liniilor SDA și SCL, datele sunt transmise între cele două dispozitive, permițând capturarea imaginilor și prelucrarea acestora pe plăcuța Arduino.
- În funcția “setup”, se realizează inițializarea protocolului I2C/TWI și a camerei OV7670, împreună cu setările de rezoluție și culoare ale camerei.
- În funcția “loop”, începe capturarea imaginii în format 320p x 240p de la camera OV7670.
- În funcția “loop”, se realizează citirea și transmiterea imaginii de la camera OV7670 de la stânga la dreapta și de sus în jos. Astfel, datele de imagine sunt procesate și transmise în ordinea pixelilor, începând cu colțul stânga-sus și înaintând către colțul dreapta-jos al imaginii.
- În funcția “loop”, imaginea capturată de la camera OV7670 este transmisă în format serial către plăcuța Arduino. Aceasta poate fi apoi trimisă prin intermediul unei conexiuni seriale către un PC sau o altă aplicație de pe PC, unde poate fi afișată și procesată ulterior.

Diagrama setup de mișcare:

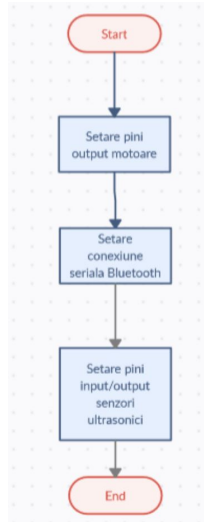
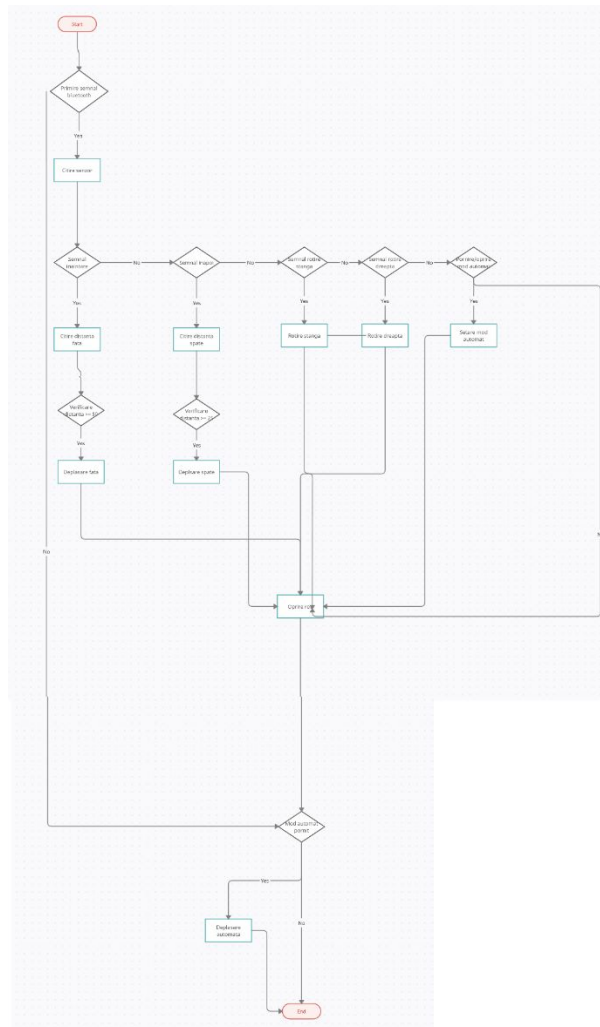
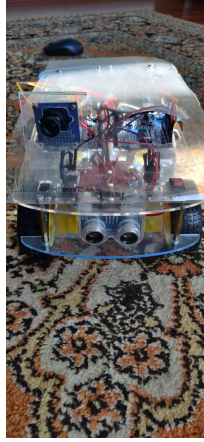


Diagrama Loop de miscare:



Rezultate Obținute

Robot:



Imagine obtinuta:



Concluzii

Am descoperit ca nu imi doresc sa mai fac robotei sau sa folosesc tehnologia bluetooth

Download

- [proiect_pm_robot_video.zip](#)
- Directorul "camera" conține codul specific plăcuței Arduino responsabilă de funcționalitatea camerei.
- Directorul "miscare_din_bluetooth" conține codul specific plăcuței Arduino responsabilă de controlul mișcării robotului prin intermediul Bluetooth.
- Directorul "SerialPortReader" conține aplicația responsabilă de vizualizarea imaginilor provenite de la camera, prin intermediul portului serial.

Bibliografie/Resurse

- [Camera](#)
- [Puntea H](#)
- [Bluetooth](#)
- [Senzor Ultrasonic](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/apredescu/robotvideo>



Last update: **2023/05/28 16:40**