

# Horia-Alexandru DRAGOMIR (87494) - BT Car

Date de contact: [Horia-Alexandru Dragomir](#)

## Introducere

Proiectul consta in realizarea unei masinute teleghidate prin intermediul unei aplicatii Android via Bluetooth.

Am ales acest proiect deoarece masinutele teleghidate au fost cele mai frumoase jucarii ale copilarii mele. Prin intermediul acestei teme, voi incerca sa pun in aplicare cunostintele teoretice dobandite pe parcursul laboratoarelor si sa-mi dezvolt creativitatea si imaginatia. Pe langa functiile de baza ale oricarui vehicul, adica mersul inainte/inapoi si virarea la stanga/dreapta, masinuta va fi capabila sa claxoneze prin intermediul Buzzer-ului si sa semnalizeze prin intermediul unor LED-uri. Masinuta va avea in partea din fata si niste lumini asemanatoare unor faruri, iar utilizatorul va putea alege viteza cu care masinuta se deplaseaza.

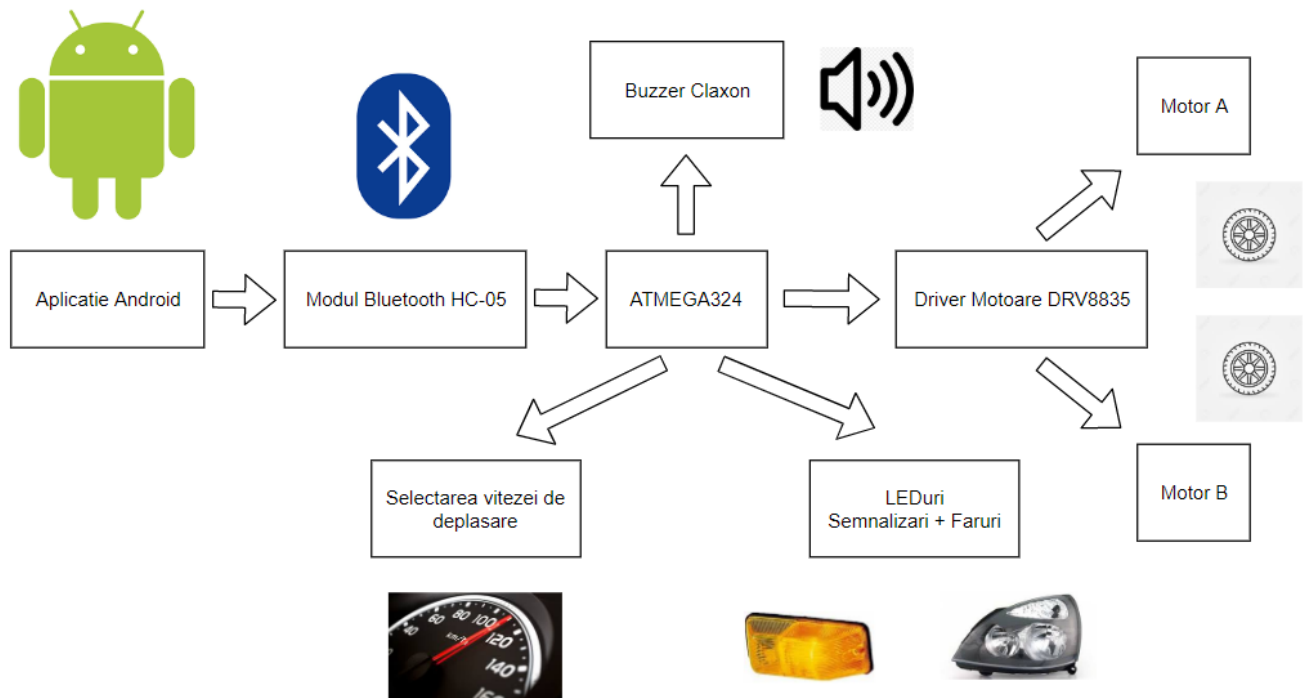
Pe viitor, imi propun sa continui dezvoltarea acestui proiect prin adaugarea unei camere video care transmite telefonului imaginile surprinse la bordul masinii, a unor senzori de distanta care detecteaza obstacolele apropiate masinutei si a unor senzori care masoara diferiti parametrii cum ar fi temperatura, umiditate, luminozitate, zgomot, etc.

## Descriere Generala

Folosind o aplicatie Android, masinuta va raspunde la urmatoarele comenzi transmise prin Bluetooth:

- oprire
- mers inainte
- mers inapoi
- viraj stanga
- viraj dreapta
- claxon
- avarii
- semnalizare stanga (fata-spate)
- semnalizare dreapta (fata-spate)
- faruri fata (stanga-dreapta)
- selectarea vitezei de deplasare (incet, normal, repede)

[Schema bloc a proiectului:](#)



## Hardware Design

Placuta functioneaza la o tensiune de 3.3V, deoarece modulul de Bluetooth HC-05 lucreaza la aceasta tensiune.

Piese utilizate:

- ATmega324P
- Modul Bluetooth HC-05
- Driver Motoare DRV8835
- Modul DC-DC Step-Down MP1584
- LED-uri: 4 galbene si 2 albe
- 2 x Rezistente 33Ω (pentru LED-uri galbene)
- Rezistenta 7.5Ω (pentru LED-uri albe)
- Buzzer pasiv
- Intrerupator
- Breadboard
- Fire mama-tata
- Fire breadboard

LED-urile functioneaza la un curent de aproximativ 20mA si o tensiune de 2V (LED-uri galbene) respectiv 3V (LED-uri albe).

Pe langa acestea, am achizitionat un sasiu ce contine:

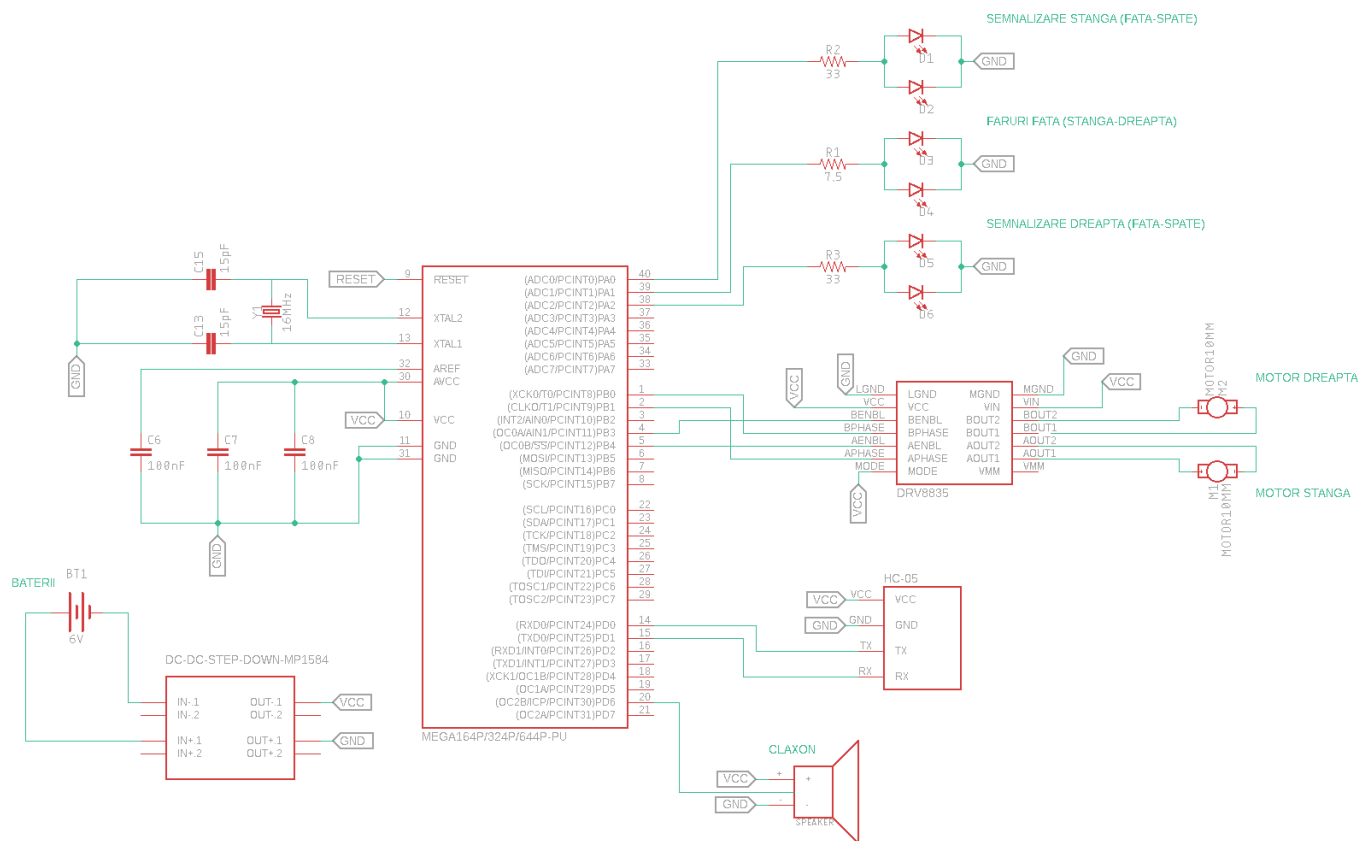
- Platforma mobila
- Compartiment pentru 4 baterii
- 2 x Motoare electrice de curent continuu
- 2 x Anvelope
- Roata omnidirectionala

Cele 4 baterii furnizeaza o tensiune de aproximativ 6V.

Fiecare motor functioneaza la o tensiune de 3.6V.

**Alimentarea placutei:** Bateriile se conecteaza la sursa coboratoare de tensiune si tensiunea de aproximativ 6V de la intrare trece in 3.3V la iesire. Placuta va fi alimentata folosind aceasta tensiune de iesire.

### Schema electrica:



## Software Design

Arhiva proiectului contine urmatoarele fisiere:

- bt-car.c
- usart.c + usart.h
- Makefile

Pentru medierea intre dispozitivul Android si modulul HC-05, am folosit o aplicatie din Google Play numita Bluetooth Terminal HC-05.

Aplicatia poate fi accesata la urmatorul link: [details](#)

Panoul de comenzi este urmatorul:



Fiecare comanda este codificata printr-un caracter ce este trimis modulului HC-05 atunci cand un

anumit buton este apasat. Modulul de Bluetooth functioneaza ca un intermediar intre utilizator si  $\mu C$ : primeste o comanda de pe dispozitivul Android si o trimite mai departe  $\mu C$ -ului.

Microcontroller-ul si modulul de Bluetooth comunica prin interfata seriala. Parametrii comunicatiei sunt: 9600, 8, N, 1. Acestia sunt configurati in functia `USART0_init()`. Pe langa aceasta functie, header-ul `usart.h` mai contine functia numita `USART0_receive()` prin intermediul careia  $\mu C$ -ul receptioneaza un caracter de la modulul de Bluetooth.

In functia `main()` a sursei `bt-car.c` se afla o bucla infinita in care  $\mu C$ -ul receptioneaza permanent comenzile utilizatorului. In functie de caracterul primit,  $\mu C$ -ul va executa comanda specifica.

Claxonul:

- functioneaza, prin intermediul buzzer-ului pasiv, la o frecventa de aproximativ 490 Hz
- este controlat prin PWM, pe canalul OC2B

Semnalizarile:

- functioneaza la o frecventa de 2 Hz (jumătate de secunda LED-ul este aprins si cealalta jumătate LED-ul este stins)
- aprinderea / stingerea LED-urilor se realizeaza prin intermediul unor intreruperi, configurate pe timer-ul 1
- partea stanga este configurata pe intreruperea Compare Match A (OCR1A), iar partea dreapta este configurata pe intreruperea Compare Match B (OCR1B)

Avariile:

- partea stanga trebuie sa se aprinda si sa se stinga concomitent cu partea dreapta
- pentru aceasta, am configurat ca OCR1A sa fie egal cu OCR1B

Farurile:

- sunt legate la un pin GPIO setat ca iesire

Interactiunea cu driver-ul de motoare:

- driver-ul de motoare contine pentru fiecare motor cate doua terminale numite PHASE si ENABLE
- terminalul PHASE este conectat la un pin GPIO si prin intermediul acestuia se poate selecta sensul de rotatie al motorului
- terminalul ENABLE este conectat la un canal PWM, iar factorul de umplere al semnalului coincide cu viteza de rotatie
- motorul din stanga este conectat la canalul OC0A, iar cel din dreapta la canalul OC0B

Deplasarea inainte/inapoi:

- in functie de directia de deplasarea dorita, se seteaza pinii PHASE ai celor doua motoare
- miscare inainte  $\Rightarrow$  cei doi pini sunt pusi pe HIGH
- miscare inapoi  $\Rightarrow$  cei doi pini sunt pusi pe LOW
- cele doua motoare au acelasi factor de umplere

Deplasarea stanga/dreapta:

- miscare stanga  $\Rightarrow$  motorul de pe partea stanga va avea un factor de umplere cu 50% mai mic decat

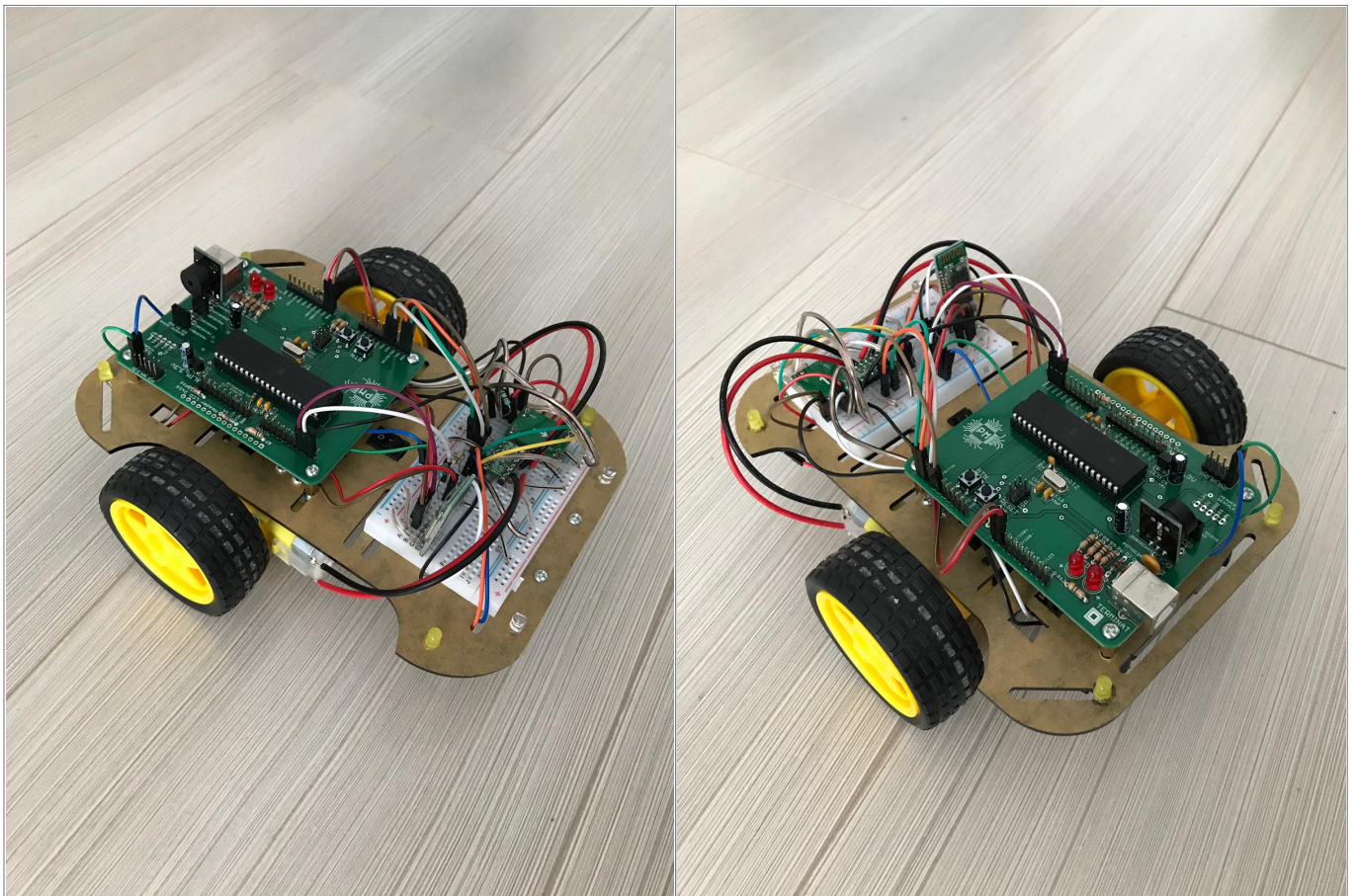
motorul din dreapta

- miscare dreapta  $\Rightarrow$  motorul de pe partea dreapta va avea un factor de umplere cu 50% mai mic decat motorul din stanga

Selectarea vitezei de deplasare:

- viteza de deplasare se configureaza schimbând factorii de umplere ai motoarelor
- exista 3 viteze de deplasare: incet, normal si repede
- incet  $\Rightarrow$  33%, normal  $\Rightarrow$  66%, repede  $\Rightarrow$  100%

## Rezultate Obtinute



Am realizat si un scurt videoclip in care este prezentata masinuta si functionalitatile acesteia: [watch](#)

## Concluzii

- probabil cea mai interesanta, placuta si captivanta tema pe care am avut pana acum in facultate
- satisfactia este si mai mare atunci cand realizez ca am construit aceasta masinuta de la zero, incepand cu lipirea placutei si terminand cu incarcarea codului
- desi la inceputul semestrului acest proiect mi s-ar fi parut imposibil de implementat, pe parcurs am realizat ca lucrurile nu sunt chiar atat de SF precum credeam si facand pasi mici am reusit sa obtin intregul, depasindu-mi asteptarile pe care le aveam la aceasta materie
- consider ca scopul proiectului a fost atins, pentru ca am reusit sa pun in practica ce am invatat pe

parcursul laboratoarelor de PM


- pe langa programarea si interactiunea cu  $\mu$ C-ul, mi-am imbunatatit si abilitatile de lucru cu hardware-ul

## Download

[Schema electrica EAGLE](#)

[Arhiva cod](#)

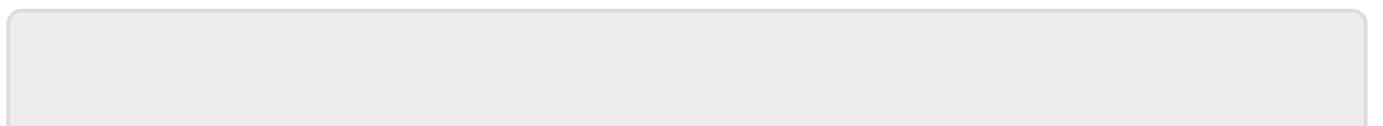
## Jurnal

- 21.04 - Adaugare Introducere, Descriere Generala (schema bloc) si Hardware Design (lista pieselor utilizate)
- 22.04 - Am terminat de lipit componentele de baza ale placutei, am incarcat bootloader-ul si am testat functionalitatea placutei prin incarcarea unui program
- 23.04 - Am trecut placa la 3.3V
- 03.05 - Am facut schema electrica in EAGLE
- 07.05 - Am prins suportul de baterii sub sasiul masinutei si am lipit intrerupatorul
- 07.05 - Am pus sursa coboratoare de tensiune si am verificat ca iesirea acesteia sa fie de 3.3V
- 08.05 - Am pus toate LED-urile si buzzer-ul si am testat functionalitatea acestora
- 09.05 - Am adaugat driver-ul de motoare si printr-un program de test am verificat ca masinuta se poate deplasa
- 10.05 - Am adaugat modulul de Bluetooth, am descarcat aplicatia din Google Play si am verificat ca telefonul poate comunica cu  $\mu$ C-ul prin intermediul acestuia
- 11.05 - Am aranjat toate componentele pe sasiul masinii astfel: am prins placuta cu niste inaltatoare in partea din spate a masinutei, am lipit breadboard-ul in partea din fata a acesteia, am aranjat toate componentele pe breadboard inclusiv firele si am lipit toate LED-urile pe sasiul masinutei
- 11.05 - Am scris codul si l-am incarcat pe placuta
- 11.05 - Masinuta este functionala! `<fc #008000>VROOM VROOM VROOM</fc>` 

## Bibliografie / Resurse

[Accesati această pagină în format PDF](#)

- ATmega324P  $\Rightarrow$  [doc8272.pdf](#)
- DRV8835  $\Rightarrow$  [2135](#)
- HC-05  $\Rightarrow$  [hc-05-bluetooth-module](#)



From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2019/ctranca/bt-car>



Last update: **2021/04/14 15:07**