

# Bluetooth Car

## Autor

[Laura-Gabriela Dumitrel](#)

## Introducere

Proiectul isi propune controlarea unei masinute folosind pozitia telefonului mobil.

Am ales acest proiect deoarece consider ca pot imbina cunostinte atat Hardware, cat si Software dobandite in cadrul materiei de PM in constructia acestei masinute. Proiectul are atat scop didactic, cat si unul distractiv pentru pasionatii de jocuri.

## Descriere generala

Aplicatia de pe telefonul mobil va transmite date in timp real (de la accelerometer sensor prin intermediul unei aplicatii Android). Comenzile sunt interpretate de modul de Bluetooth si transmise mai departe catre microcontroller-ul ATMEGA324PA. In cele din urma, informatia se va transmite catre driver-ul de motoare (care va controla cele doua motoare).

Prin intermediul acesteia aplicatii, masinuta va raspunde la comenzi transmise prin Bluetooth precum:

- mers inainte (am folosit mai multe trepte de viteze - controlate folosind Fast PWM)
- mers inapoi
- rotire stanga
- rotire dreapta
- viraj stanga
- viraj dreapta
- oprire

## Schemă bloc



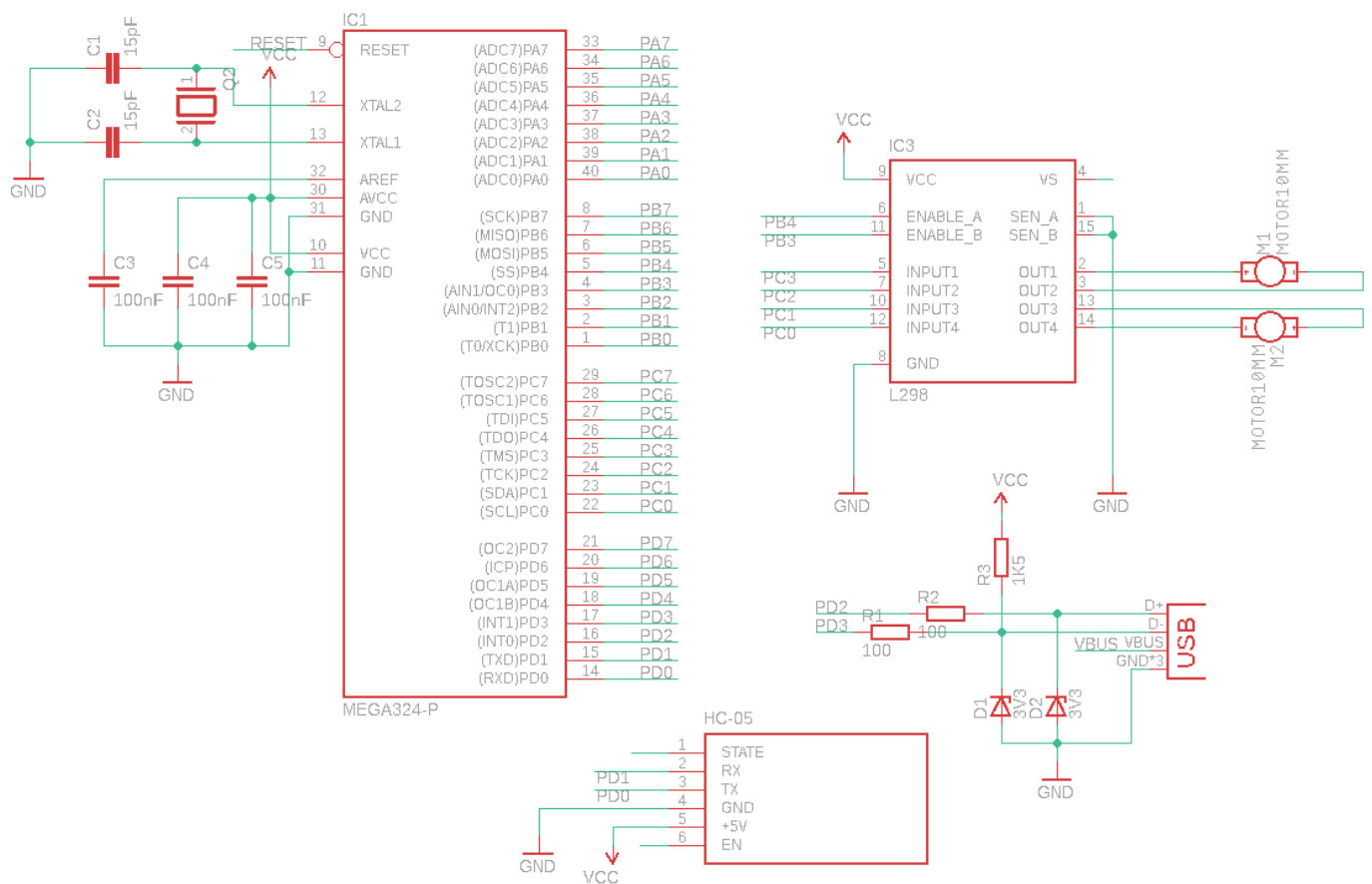
# Hardware Design

## Listă componente:

- ATmega324
- Bluetooth module HM-05
- Motor Drive Controller L298N
- Sasiu ce contine:
  - \* DC Motor x 2
  - \* Platforma
  - \* Compartiment pentru cele 4 baterii (4 \* 1.5V = 6V)
  - \* Anvelope
  - Fire

Modulul Bluetooth HC-05 lucreaza la o tensiune de 3.3V, deci placuta a fost setata sa functioneze la 3.3V.

## Schemă electrică:



# Software Design

Proiectul contine urmatoarele fisiere:

- main.c

- usart.c
- usart.h
- Makefile
- aplicatia Android

Initial, aplicatia folosita pentru a trimite comenzi Bluetooth a fost BLE Terminal HC-05 (Bluetooth Terminal HC-05). Am definit butoane care trimit comenzi date de catre utilizator modulului Bluetooth si care sunt transmise apoi de acesta microcontrollerului.

Dupa ce am testat conectivitatea la Bluetooth, am folosit Android Studio pentru a face o aplicatie Android. Am implementat o functie care interpreteaza datele primite de la device-ul Android (de la accelerometer sensor) si controleaza motoarele in functie de acestea. Datele sunt receptionate prin Bluetooth - folosind protocolul USART pentru comunicare.

In fisierul sursa al proiectului am facut o bucla infinita in care se primesc permanent comenzi in functie de pozitia pe care utilizatorul o da telefonului.

Am setat viteza de deplasare a masinutei conectand ENABLE la un canal Fast PWM. Viteza masinutei va fi setata prin cresterea, respectiv scaderea factorului de umplere a semnalului. Cele doua motoare sunt conectate la timerul 0 si variaz valoarea registrelor OCR0A si OCR0B pentru a seta factorul de umplere intre 0 si 255. Timerul 0 este setat cu prescaler de 1024. Comenzile pe care le-am dat din aplicatie in functie de axele X, Y returnate de pozitia telefonului prin intermediul aplicatiei Android sunt:

- 'f' : mers inainte ( $Y < 0$ )
- 'b' : mers inapoi ( $Y > 0$ )
- 'l' : viraj stanga ( $X > 0$ )
- 'r' : viraj dreapta ( $X < 0$ )
- 's' : stop

Am folosit 2 butoane prin care realizez rotatia:

- 'p' - stanga
- 'q' - dreapta

Am folosit 3 butoane prin care setez viteza de deplasare, astfel:

- incet '1' :  $OCR0A/OCR0B = 128$  (jumătate din viteza maxima)
- mediu '2' :  $OCR0A/OCR0B = 192$  (75 % din viteza maxima)
- repede '3' :  $OCR0A/OCR0B = 255$  (viteza maxima)

**Mers inainte:** ambele motoare se deplaseaza pe directia fata

**Mers inapoi:** ambele motoare se deplaseaza pe directia spate

**Viraj stanga:** motorul stanga va avea un factor de umplere cu 50% mai mic decat cel din dreapta

**Viraj dreapta:** motorul dreapta va avea un factor de umplere cu 50% mai mic decat cel din stanga

**Rotire stanga:** motorul stanga se deplaseaza in fata, si motorul dreapta in spate

**Rotire dreapta:** motorul dreapta se deplaseaza in fata, si motorul stanga in spate

**Oprire:** opresc ambele motoare

## Rezultate Obtinite

Masina este controlata de o aplicatie Android in functie de pozitia telefonului. Are urmatoarele functionalitati:

- <fc #008000>deplasare fata/spate</fc>
- <fc #008000>viraj stanga/dreapta</fc>
- <fc #008000>rotire stanga/dreapta</fc>
- <fc #008000>deplasare cu mai multe trepte de viteza(3 trepte de viteza)</fc>

Am atasat imagini cu masinuta in starea finala si un filmulet cu functionalitatile acesteia:

[Mwe848w4GXQ](#)



## Concluzii

Am reusit sa realizez o masinuta functionala controlata dintr-o aplicatie Android. Proiectul a necesitat destul de mult efort deoarece pe parcurs am intalnit diverse bug-uri atat Software, cat si Hardware, insa in final a iesit functionalitatea dorita. Am invatat multe din acest proiect si consider ca petrecute in realizarea lui au adus in final entuziasmul de a vedea ca merge ceva care nu te asteptai ca va iesi vreodata.


## Download

[dumitrellaura\\_btcar.zip](#)

[laura-androidapp.zip](#)

## Jurnal

- 21.04 - Adaugare introducere, descriere generala, schema bloc si lista pieselor utilizate
- 04.05 - Am adaugat schema electrica in EAGLE
- 08.05 - Am alimentat placuta si am conectat driver-ul de motoare si modulul Bluetooth
- 10.05 - Am incarcat un program de test pentru a testa functionalitatea (daca se poate deplasa inainte)

- 11.05 - Cu ajutorul aplicatie BLE Terminal HC-05 am trimis comenzi si am verificat receptionarea comenzilor si functionarea corespunzatoare (mers inainte/inapoi, viraj stanga/ dreapta, oprire)
- 16.05 - Am realizat partea software a proiectului
- 21.05 - Am realizat aplicatia Android
- 22.05 - Am finalizat codul in C pentru microcontroller si l-am incarcat pe placuta. <fc #0000FF>Masinuta este functionala</fc> 

## Bibliografie, resurse

[doc8272.pdf](#)

[istd016A.pdf](#)

[Această pagină în format PDF](#)

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2019/ctranca/lgd\\_project](http://ocw.cs.pub.ro/courses/pm/prj2019/ctranca/lgd_project) 

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