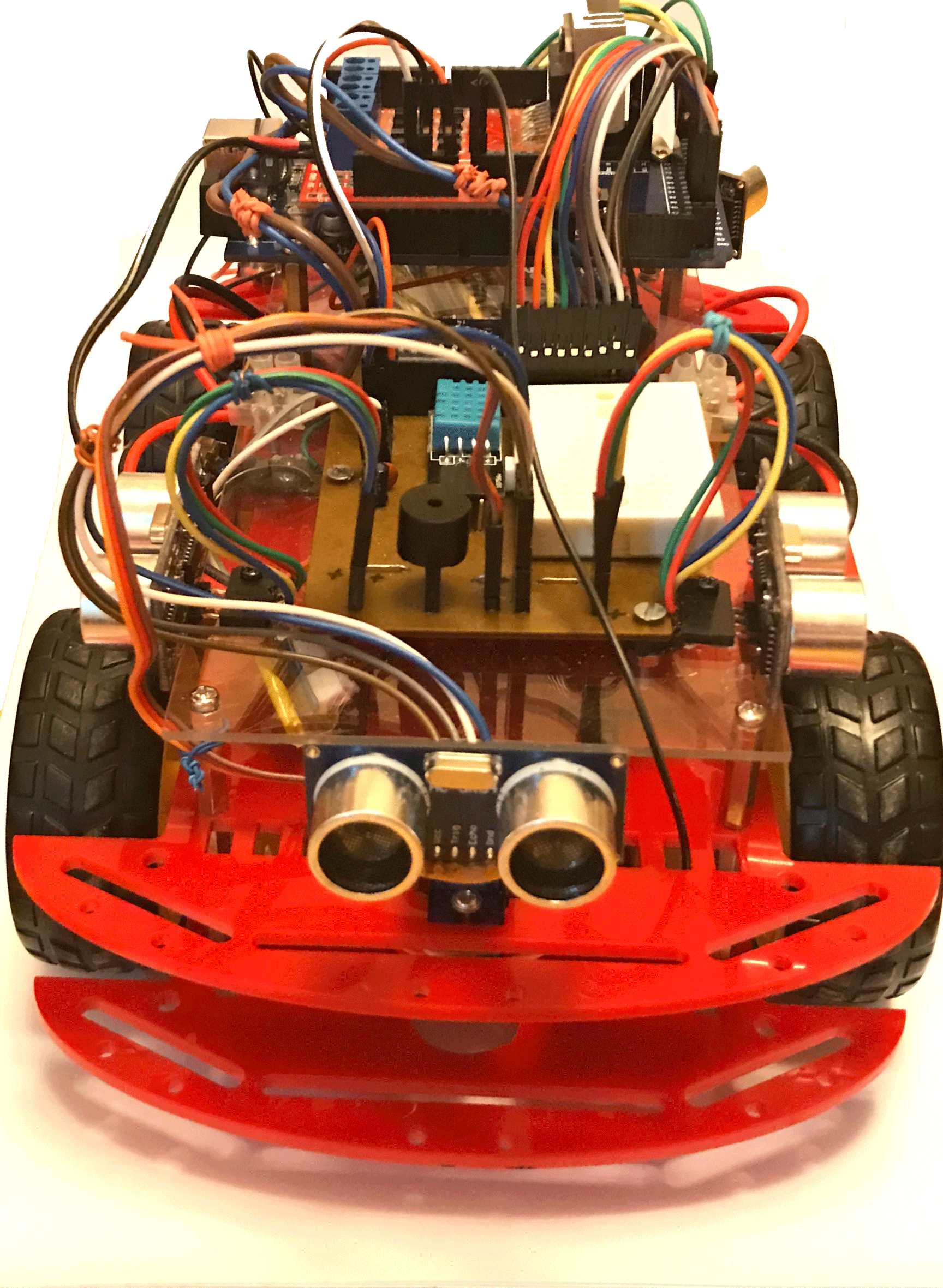
Infoeducație

Etapa Națională 31 iulie – 6 august 2017

**Referatul lucrarii**

“DriveJR” – Mașina inteligentă



Realizator: Prof. Coordonator:

**Josanu Rareș Ionuț**  **Pantelimonescu Remus**

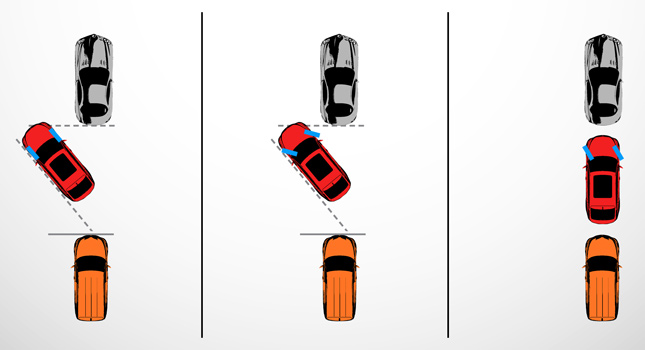
**I.Utilitate practică**

Mașina inteligentă este un subiect de ultimă ora pe plan național cat si pe plan internațional . Proiectul “DriveJR”, are ca scop dezvoltarea unei platforme universale bazată atat pe partea de **Hardware** cat si pe partea de **Software** ce ar putea transforma condusul, dar si autovehihulele in lucruri mult mai sigure.

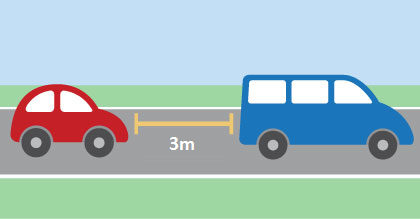
Masina la momentul actual dispune de functii cum ar fi:

* Obstacle Avoid (evită obstacolele aparute pe traseu)
* Distance Keeper (pastrează distanța față de autoturismul din față)
* Remote control (controlul mașinii prin intermediul unei aplicații, funcție ce poate fi utilizată în paralel cu funcția de Obstacle Avoid)
* Test Environment (indicare a nivelului de **umiditate** si al **temperaturii** din mediul inconjurător)
* Park Assist (Parcare laterala realizata in totalitate de autoturism, această functie nu este implementată în totalitate)

Pe viitor, mașina va dispune de sistem de navigație prin GPS, sistem de recunoastere a semnelor de circulație si al benzilor realizat prin intermediul unei camere video si prin intermediul platformei OpenCV

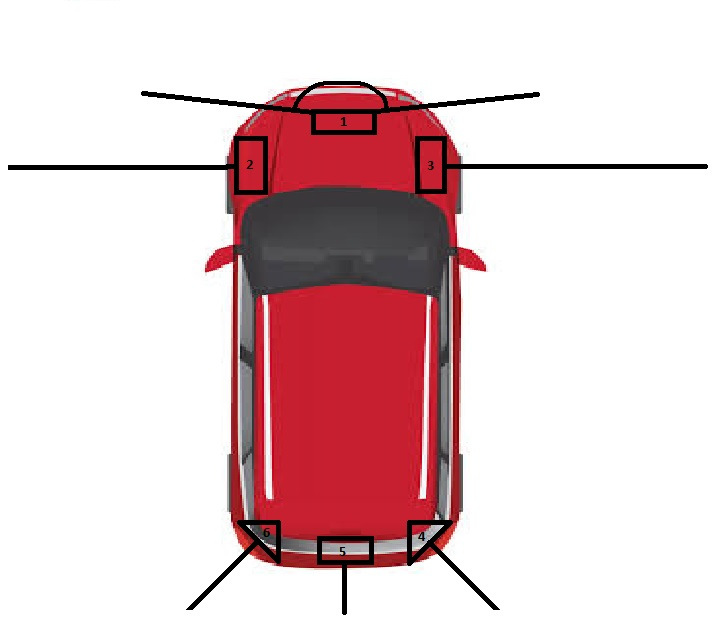
.

(Parcare laterală)



(Păstrarea distanței)

Spre deosebire de alte concepte de mașini inteligente, prototipul prezentat se poate realiza cu un cost redus, este compatibil cu orice model de mașină, atata timp cât mașina este dotata cu pilot automat si servodirecție.



(Poziționarea senzorilor de distanță)

**II.Mecanica**

Pe partea mecanică, robotul este construit pe baza unui șasiu ce predispune de 4 motoare de curent continuu (DC), un servomotor ce are rolul de a mișca senzorul de distanță frontal, 4roți cu diametrul de 65mm, 2 placi din plastic pe care au fost montate motoarele și o placă din plexiglas pe care au fost montate componentele electronice.



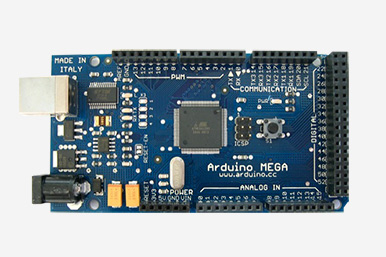
Robotul se poate deplasa liber, dar si cu limite impuse de funcția de ocolire a obstacolelor, functie ce ajută robotul sa evite obstacolele aparute în cale.

Pe șasiu sunt amplasați 6 senzori de distanță cu ultrasunete, 3 in spate, 2 pe laterale ce au un rol important în funcția de parcare si un senzor in față ce se poate mișca de la 35 pana la 145 de grade cu ajutorul servomotorului.

Datorită senzorilor și datorită modulului Bluetooth Low Energy CLASS I, robotul ajunge la un consum redus în raport cu dimensiunile sale, ceea ce înseamnă o autonomie mai mare.

**III.Componente electronice**

1. **Placa de dezvoltare Arduino Mega 2560**



Placa de dezvoltare Arduino Mega 2560 este platforma de dezvoltare ideală pentru un astfel de proiect, deoarece microcontrollerul ATMega2560 oferă un numar mare de intrari și iesiri I/O, dar si putere de procesare mare pentru a putea rula software-ul fară nici o problemă.

1. **Driverul de motoare L298 SHIELD**



Driverul de motoare L298 SHIELD a fost utilizat deoarece este o alternativă compatibilă cu placa Arduino Mega, astfel reducând numarul de conexiuni. Acest driver are rolul de a controla viteza si directia de rotație a motoarelor. Driverul oferă un output cuprins intre 3v si 12v – 3A.

1. **Modulul Bluetooth Mate**



Modulul Bluetooth Mate a fost utilizat pentru a realiza comunicarea dintre robot și pc/mobil. Robotul transmite către dispozitivul cu care este conectat date cum ar fi: Temperatura, Umiditatea, Distanta față de obiectul din față, Accelerație, Unghiul de inclinare față de sol, etc.

1. **Senzorul de distanță HC-SR04 Ultrasionic**



Senzorul HC-SR04 este un senzor de distanță ce funcționează prin intermediul ultrasunetelor. Spre deosebire de senzorii de distanță cu infraroșu, acești senzori nu sunt perturbați de lumina ambientală. Compând acest senzor cu un senzor de distanță cu infraroșu, acest tip de senzor (ultrasunete) este puțin mai lent, dar îndeajuns de rapid pentru a îndeplini sarcinile predefinite. Acest fenomen se petrece deoarece viteza sunetului este cu mult mai mică față de viteza luminii. ( sunet: 380m/s – lumina: 299 792 458m/s)

1. **Senzorul de temperatură și umiditate**



Senzorul de temperatură și umiditate este un senzor util atunci când factorii de mediu pot influenta modul in care funcționează mașina.

1. **Buzzer piezo**



Buzzerul piezo poate genera un sunet in intervalul

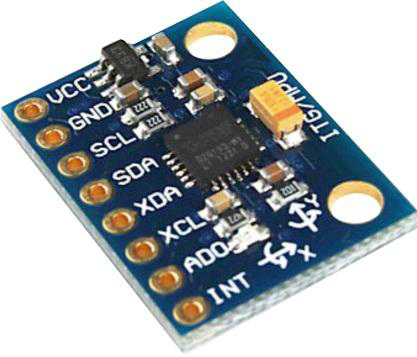
20hz – 20.000hz. Acesta a fost folosit în mai multe scopuri, cum ar fi, producerea unui beep atunci când robotul trece de la un mod la altul.

1. **Servomotor 9g**



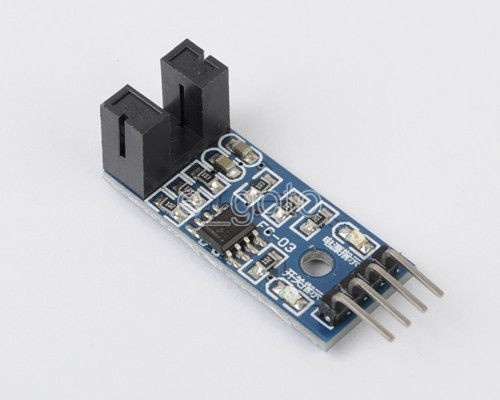
Acest servomotor a fost utilizat pentru a realiza mișcarea senzorului de distanță frontal.

1. **MPU6050 – ACCELEROMETRU SI GYROSCOP**



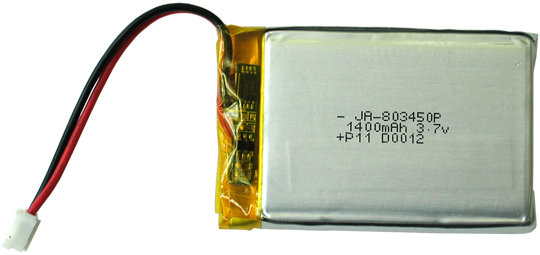
Senzorul MPU6050 a fost folosit pentru a aduna date din mediul înconjurator cum ar fi accelerația gravitaționala (calcularea vitezei de deplasare) si unghiul de înclinare față de sol cu ajutorul gyroscopului..

1. **Encoder optic**



Encoderul optic este utilizat pentru a determina cu precizie anumite distanțe în functie de numarul de rotații efectuate de motor. (Formula: nr.rotații \* 2πr , unde r reprezintă raza roții)

1. **Acumulatori**

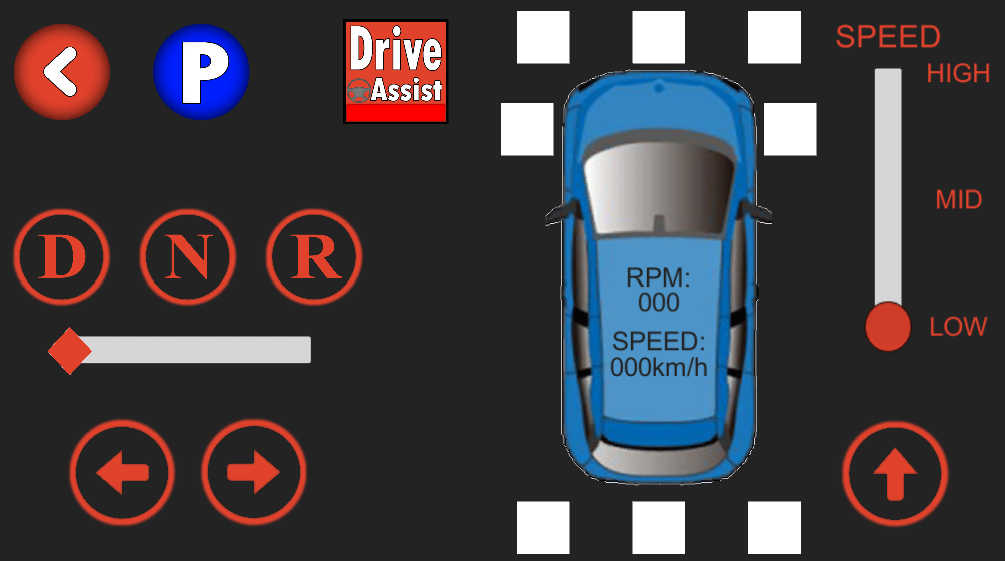


2x LiPo 3.7v 1400mah = 7.4v 1400mAh – alimentare driver motoare (serie)

4xBaterii Alkaline 1.5v 1500mah = 6v 1500mAh alimentare Arduino + Senzori (serie)

**IV.Complexitate**

În funcție de preferință, dar și de mediul înconjurător, robotul poate fi controlat prin intermediul aplicației pc/mobile via Bluetooth, asistat sau nu de funcțiile Avoid Obstacles sau Distance Keeper, funcții ce pot fi activate sau dezactivate din aplicația mașinii. (Drive Assist)



Una dintre operațiile complet autonome este funcția de parcare laterală, funcție ce poate fi accesată din aplicația mașinii, atunci când mașina este setata sa mearga cu viteză redusă.

Mediul în care robotul își desfășoară activitatea poate fi necunoscut sau cunoscut. Acest lucru nu este foarte important, deoarece robotul este programat să se adapteze la mediul de desfășurare.

Robotul poate fi controlat și în totalitate de utilizator, fară a activa funcțiile de Drive Assist.

**V.Software**

Software-ul robotului este unul complex, alcatuit dintr-o aplicație PC/Mobile din care se pot citi datele de la senzori și se pot trimite comenzi catre robot, și un cod ON-BOARD ce este scris pe microcontrollerul ATMega 2560, cod ce stă la baza mașinii. Limbajul de programare folosit: C#/C++.

**Aplicația PC/Mobile dispune de 4 meniuri:**

1.Remote Control

-Control teleghidat, functii parcare, Drive Assist, Viteze

2.Sensors Readings

-Citirea datelor trimise de senzori

3.Environment Mapping

-Reproducerea mediului înconjurator într-un plan 2D.

4.Headlight Controller.

-Controler de lumini(stopuri, semnalizari) – în dezvoltare

**Codul sursa de pe microcontroller dispune de urmatoarele functii:**

1.Transmisie / Receptie date Bluetooth

2.Funcția Avoid Obstacles

3.Funcția Distance Keeper (followFrontObject)

4.Funcția de Remote Control

5.Funcția de control asupra Servomotorului

6.Funcția de măsurare a distantei in funcție de sensor

7.Funcția de control al vitezei

8.Funcția de control al motoarelor

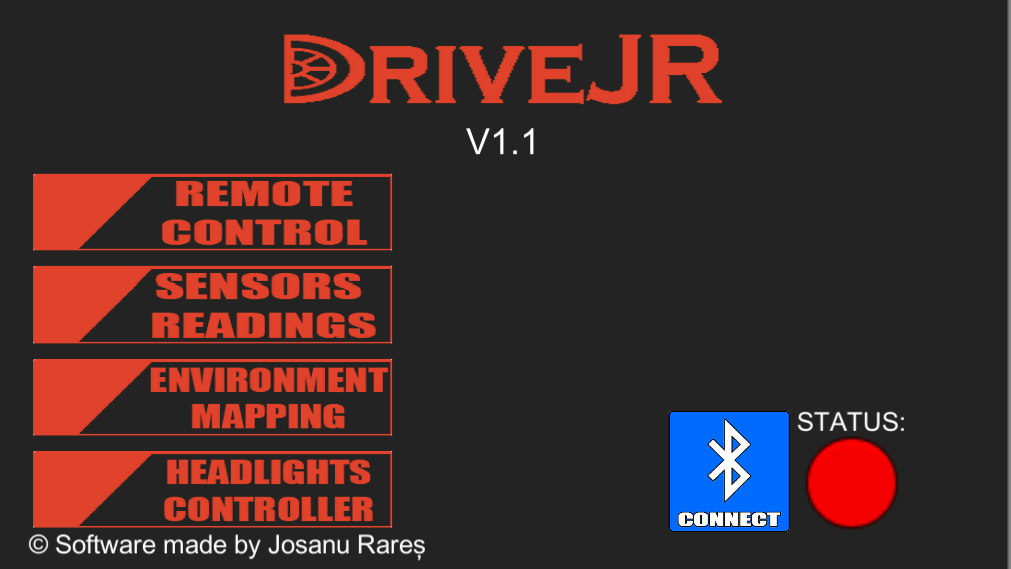
9.Funcția de parcare – în dezvoltare

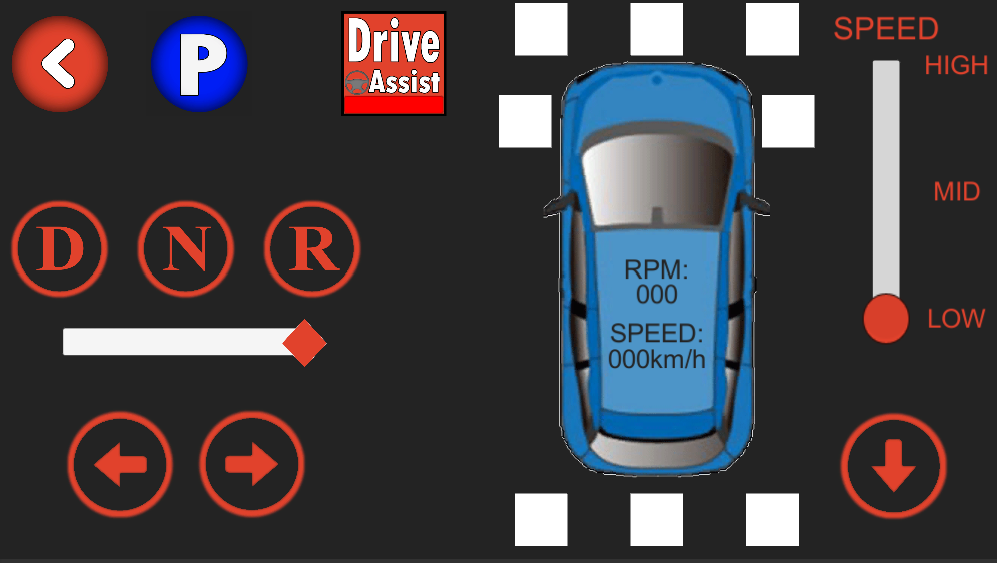
10.Funcția de citire a senzorilor aditionali (temp. umiditate, acc. gyro.)

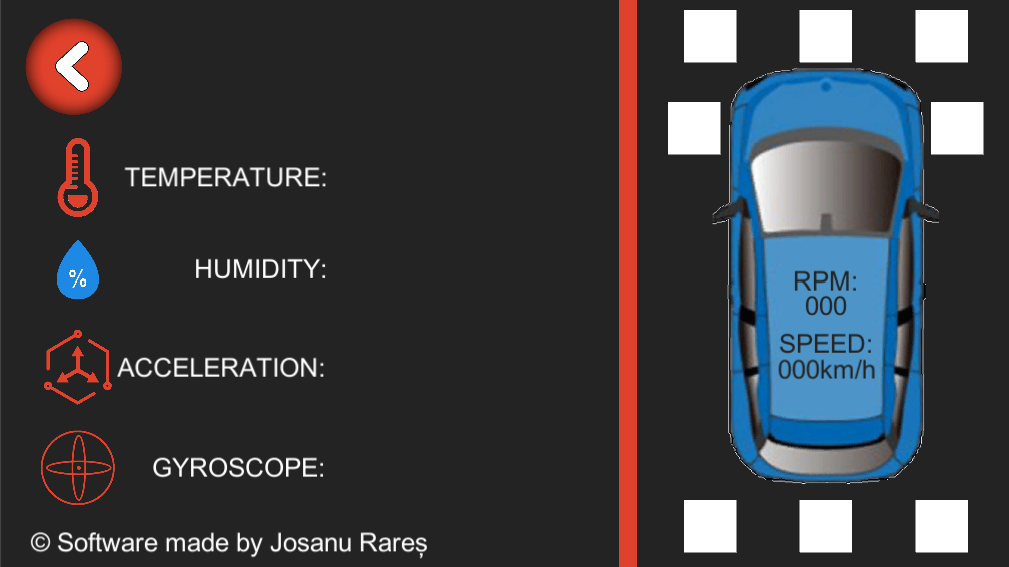
11.Funcția de generare a tonurilor pentru Buzzer

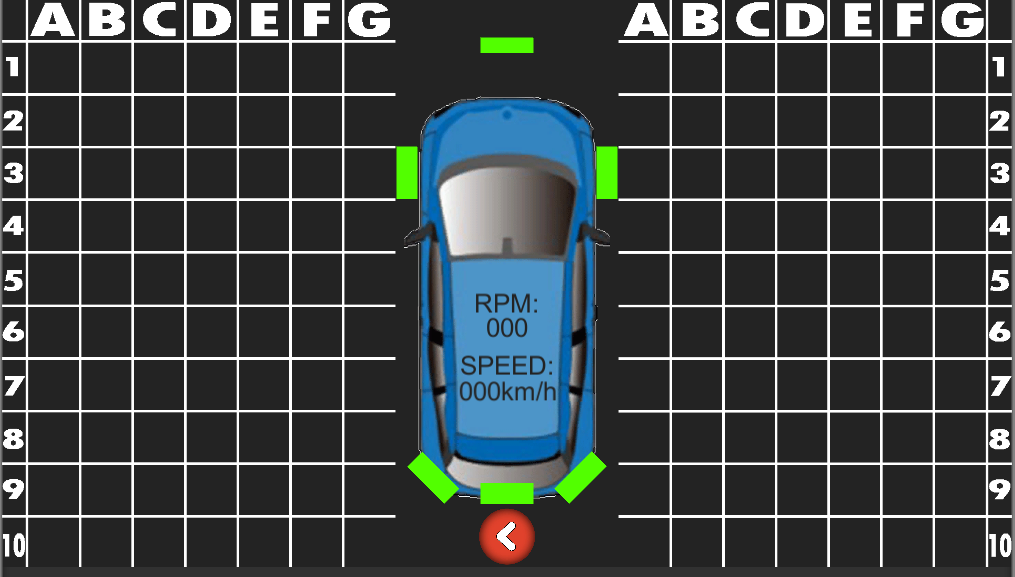
+ alte funcții esențiale.

**Screenshots din aplicația DriveJR**



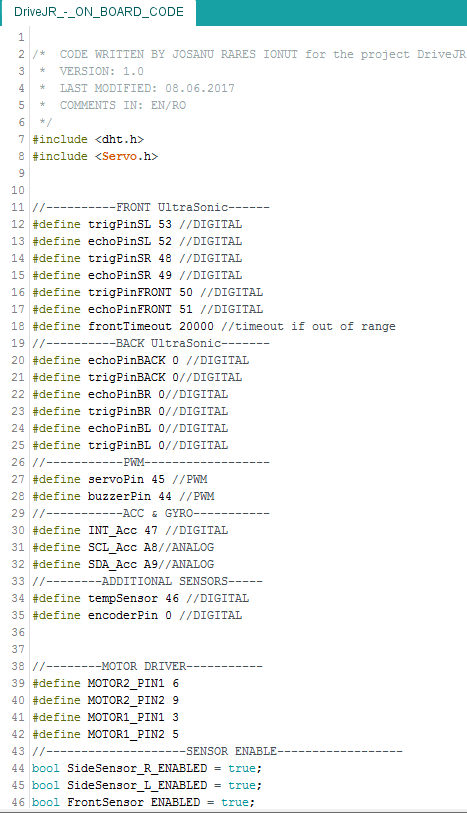


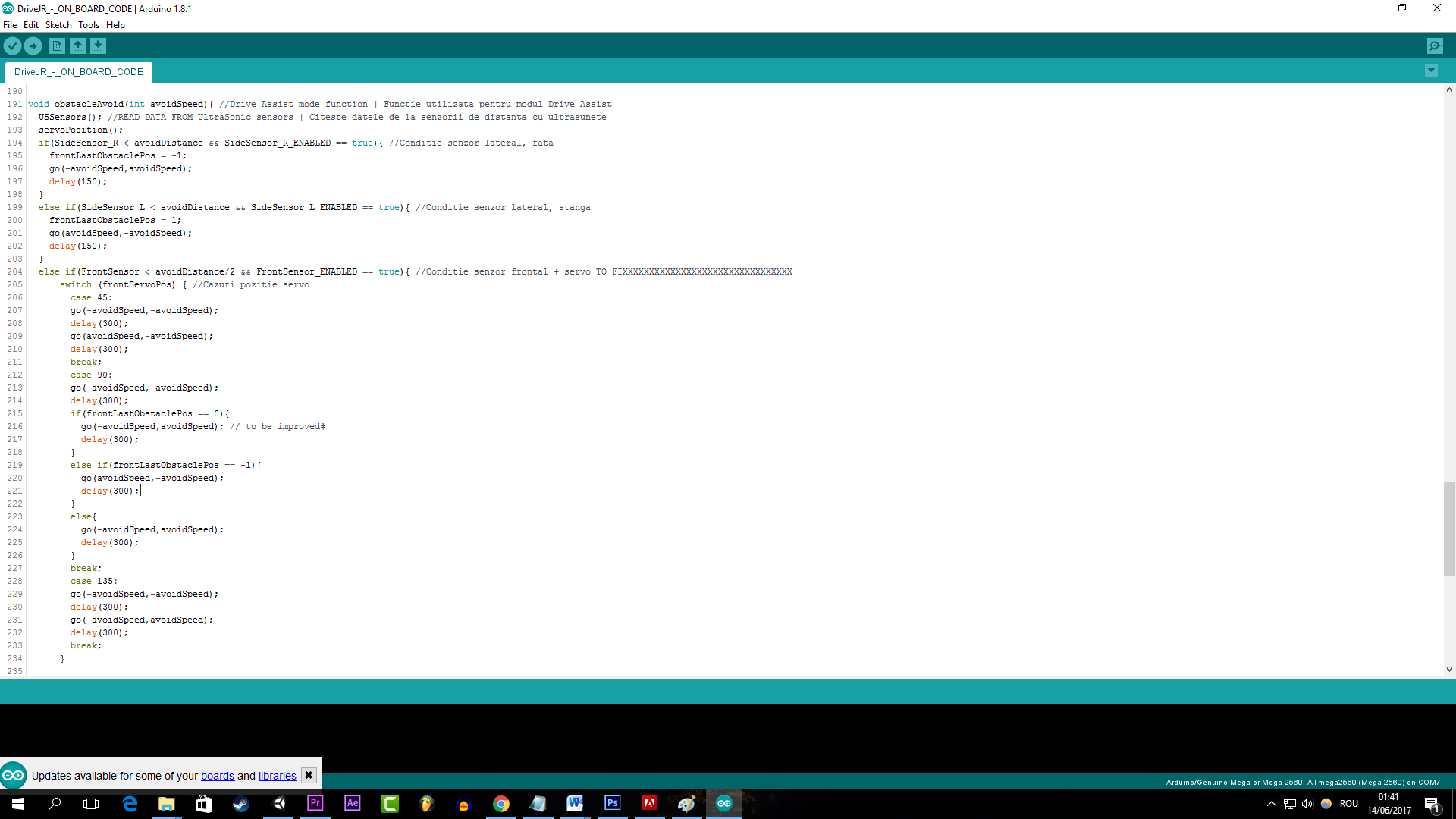
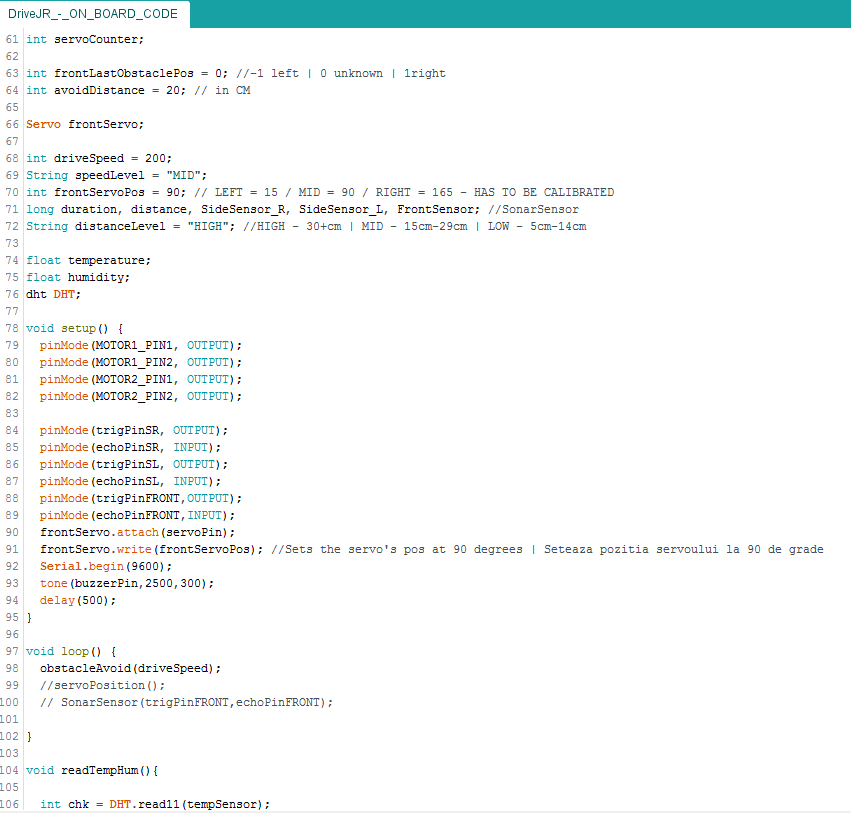


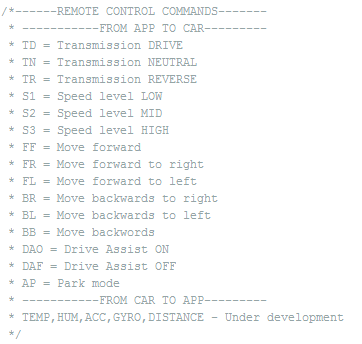


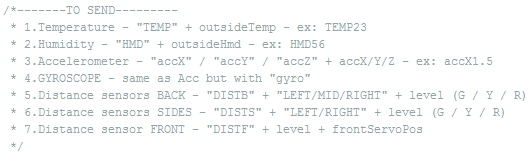
Aplicația a fost realizată în Unity.

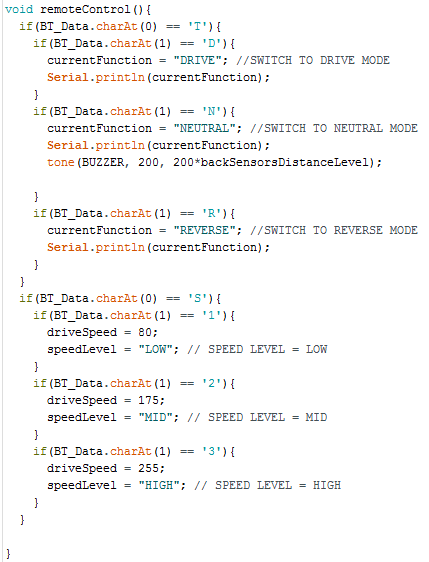
**Screenshots din Arduino IDE – cod sursă**





****

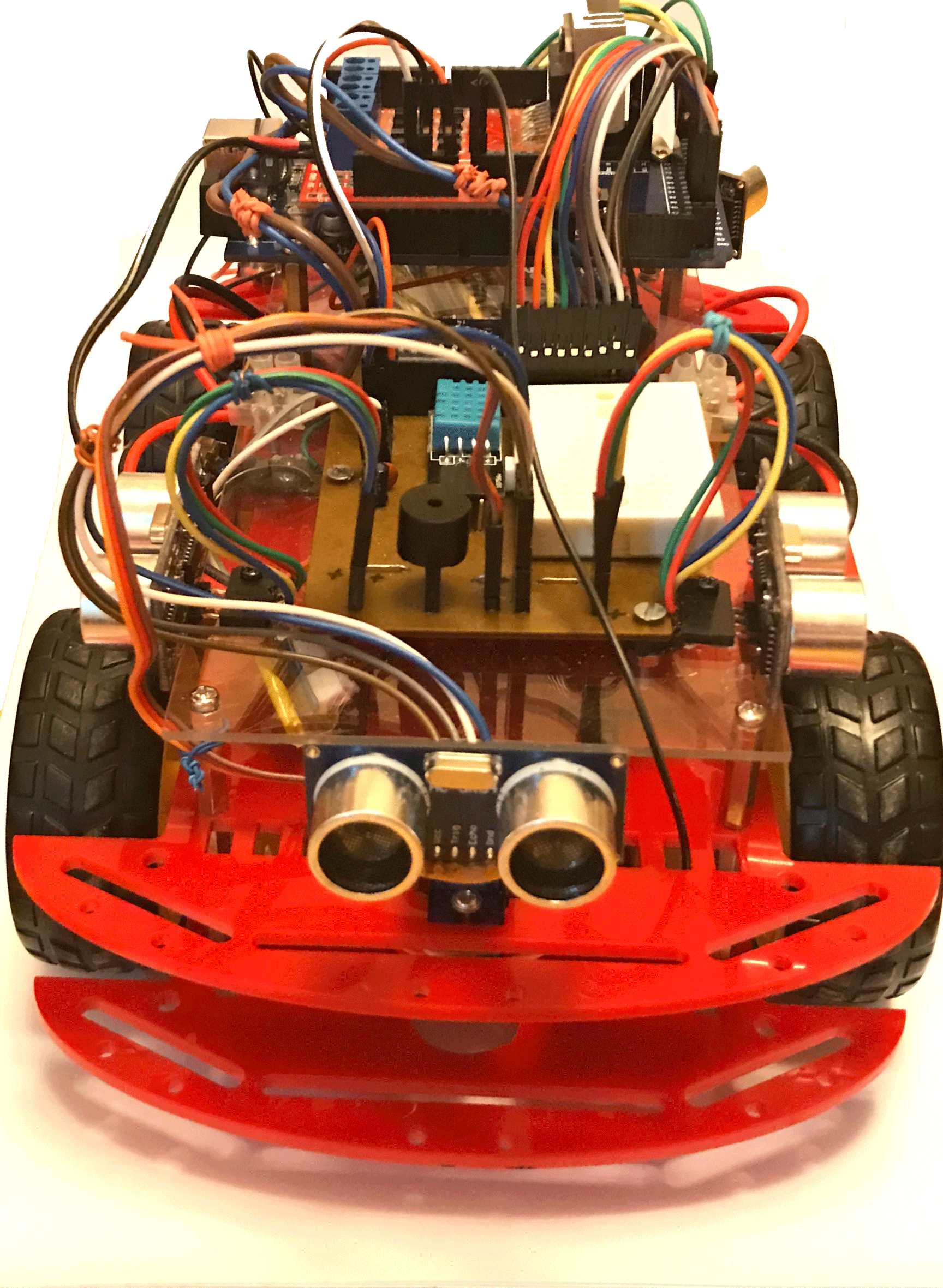
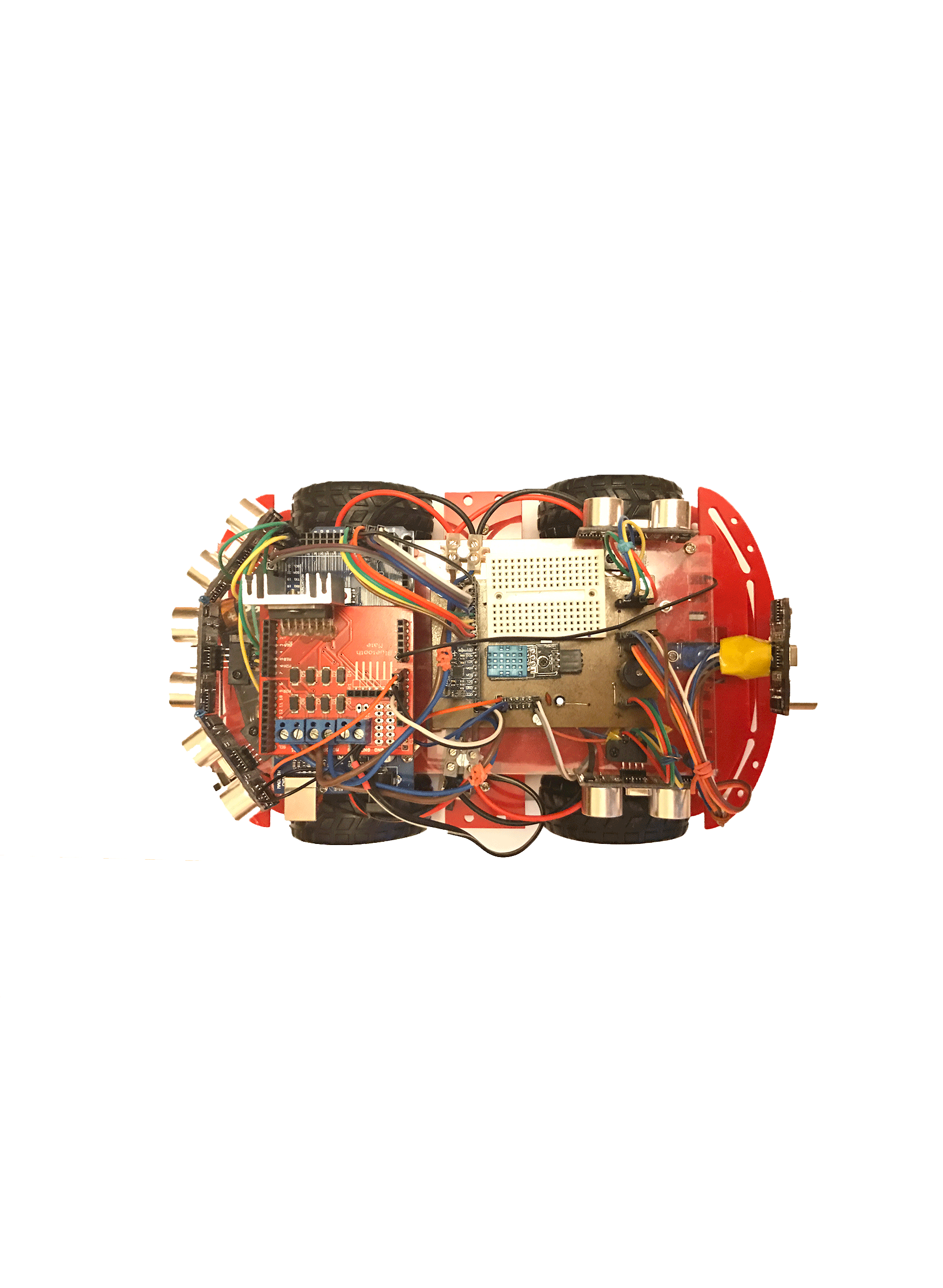
****

****

**VI.Design**

Senzorii de distanță au fost poziționați astfel încât autoturismul poate detecta orice obstacol indiferent de poziția orizontală a mașinii.

Acest sistem, fiind unul universal, poate fi instalat pe aproape orice mașină, singurele necesități fiind servodirectia si pilotul automat.



90% dintre componente sunt modulare. Acestea pot fi înlocuite foarte ușor în cazul în care se defectează.

**VII.Bibliografie**

Sparkfun datasheet – Bluetooth Mate

DHT datasheet – DHT11

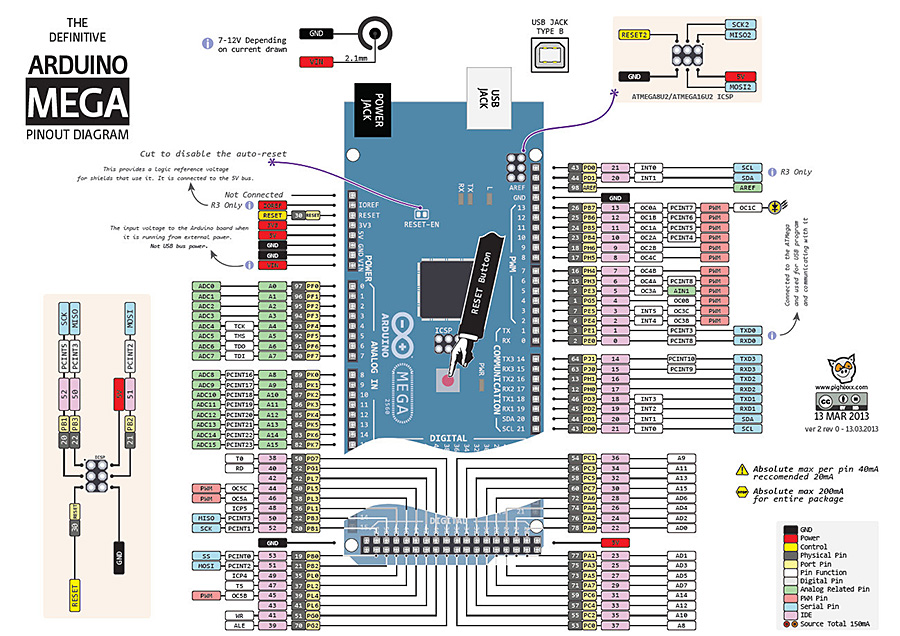
Arduino Playground – Arduino Bluetooth examples

Arduino Playground – MPU6050, BUZZER, HC-SR04

Robofun.ro – Documentatie Driver L298 shield

Unity API – clase si funcții C#

Arduino Mega PINOUT Diagram

****