



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

# **Robot controlat prin Bluetooth care detectează obstacolele și le evită în mod automat**

**Studenti: Făgădar Ionela-Cătălina**

**Hădărean Andreea-Roxana**

**Grupa: 30236**

**Profesor îndrumător: Dănescu Radu**

**Data: 16.01.2020**



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE**

## **Cuprins**

<b>1. Obiective.....</b>	<b>2</b>
<b>2. Introducere.....</b>	<b>2</b>
<b>2.1 Componente utilizate.....</b>	<b>2</b>
<b>2.2 Schema hardware .....</b>	<b>5</b>
<b>3. Algoritm.....</b>	<b>5</b>
<b>4. Rezultate.....</b>	<b>11</b>
<b>5. Concluzii.....</b>	<b>12</b>



# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

## 1. Obiective

Principalul obiectiv al proiectului a fost programarea unui robot autonom, folosind cunostintele acumulate de-a lungul orelor de laborator. Acesta are capacitatea de a se misca, de a detecta obiectele din mediul inconjurator cu ajutorul senzorului ultrasonic si de a le ocoli pentru a evita posibilele coliziuni, generand totodata un sunet de frecventa proportionala cu distanta la care se afla obiectele.

## 2. Introducere

Tema proiectului consta in proiectarea si implementarea unui robot a carui actiuni pot fi controlate prin intermediul modulului Bluetooth, putand realiza miscari in 4 directii: N, S, E, V. De asemenea, daca intalneste obstacole in drum, acesta a fost proiectat sa isi analizeze mediul inconjurator, hotarand autonom ce directie sa urmeze pentru a ocoli eventualul obstacol. In momentul detectarii obstacolului, LED-ul de pe pinul 13 se aprinde, iar buzzer-ul conectat creeaza un sunet a carui frecventa este direct proportionala cu distanta la care se afla respectivul obstacol.

### 2.1 Componente utilizate

Principala componenta utilizata in proiectarea si implementarea robotului este reprezentata de placa de dezvoltare Arduino UNO Rev3, placa care a fost programata pentru a realiza functionalitatile descrise mai sus.



Figură 1 - Placa de dezvoltare Arduino UNO



# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

## FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

Au fost utilizate module variate pentru a se obtine modul de functionare dorit, astfel:

- Modulul bluetooth 4.0 BLE CC2540 CC2541 – pentru conexiunea cu telefonul si receptia comenzilor de catre robot
- Senzor ultrasonic HC-SR04 – pentru detectia obiectelor
- Buzzer piezoelectric activ 3-24V HND-2312 – pentru a emite un sunet la aparitia unui obstacol



Figură 2 - Modul Bluetooth



Figură 3 - Senzor ultrasonic



Figură 4 – Buzzer

Pentru miscarea robotului, s-au folosit 2 motoare DC impreuna cu o punte H, deoarece aceste motoare necesita o intensitate a curentului semnificativa. Viteza de rotatie a acestor motoare depinde de tensiunea de alimentare de la bornele acestuia, iar directia depinde de polaritatea acesteia. Aceste motoare actioneaza asupra a doua roti, ce ajuta la miscare, fiind prezenta si o a treia roata de suport.



Figure 5 - Motor DC

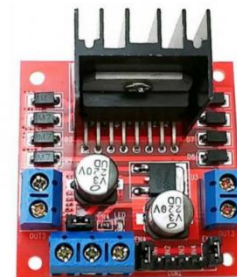


Figure 6 - punte H

Fiecare motor are 3 pini de control. Primul motor este actionat de pinii EnA ("1"), In1 si In2, pe cand celalat de EnB ("1"), In3 si In4.

Table 1 - Comenzii pentru motor

In1 (In3)	In2 (In4)	Efect
0	0	Motor oprit
0	1	Motor pornit – inainte
1	0	Motor pornit – inapoi
1	1	Motor oprit



# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

Pentru rotatia senzorului ultrasonic, s-a folosit un motor servo, deoarece acesta este un motor cu precizie ridicata. Motorul servo are 3 fire: Vcc (5V) care este de obicei de culoare rosu, GND, care are culoare neagra sau maro si un fir de comanda. De obicei, acest motor servo nu efectueaza o rotatie completa, ci doar o rotatie cu un unghi. Pentru actionarea lui, s-a folosit biblioteca Servo.

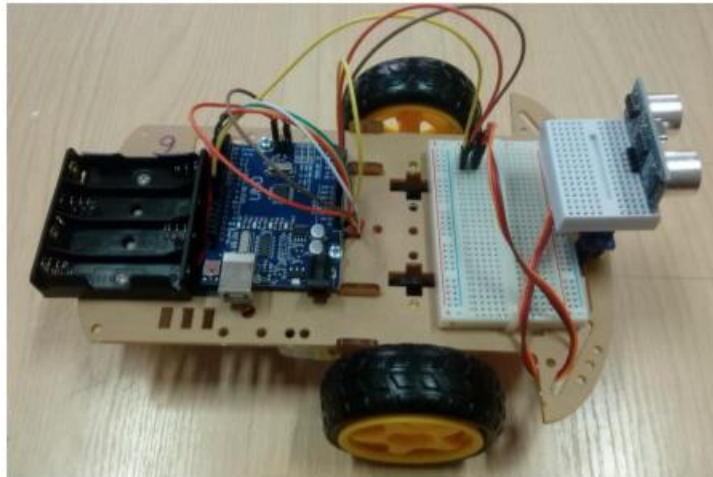


Figure 7 - robot privit de sus (fara modulul Bluetooth si buzzer)

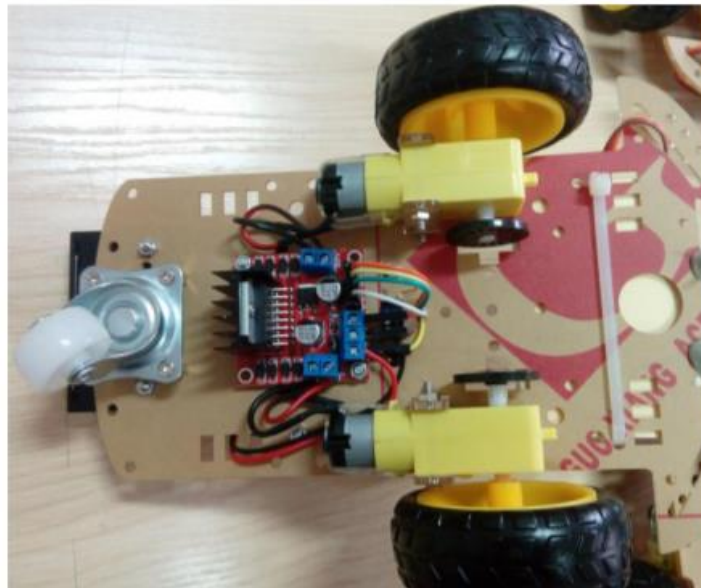
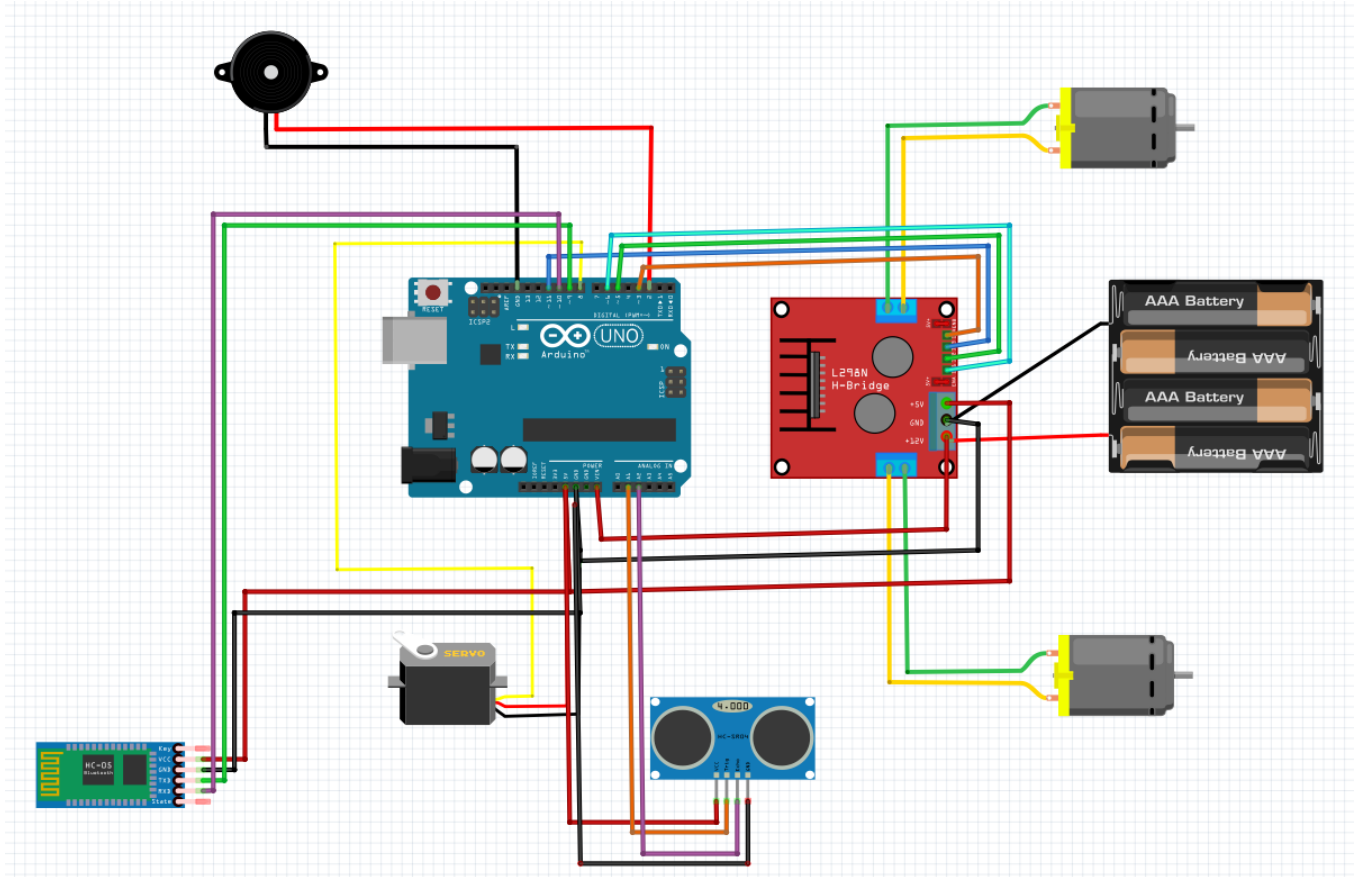


Figure 8 - robot privit de jos



**UNIVERSITATEA TEHNICĂ**  
DIN CLUJ-NAPOCA  
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

## 2.2 Schema hardware



## 3. Algoritm

```
#include <Servo.h>
```

```
#include <SoftwareSerial.h>
```

```
SoftwareSerial mySerial(9, 10); //rxPin = 9 txPin = 10;
```

```
#define mpin00 5
```

```
#define mpin01 6
```

```
#define mpin10 3
```

```
#define mpin11 11
```

```
#define echoPin A2
```

```
#define trigPin A1
```



# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

## FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

```
int b = 0;

float durata = 0;

float distanta = 0;

char dir;

Servo srv;

void setup() {
  mySerial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(13, OUTPUT);
  pinMode(2, OUTPUT);
  digitalWrite(mpin00, 0);
  digitalWrite(mpin01, 0);
  digitalWrite(mpin10, 0);
  digitalWrite(mpin11, 0);
  pinMode (mpin00, OUTPUT);
  pinMode (mpin01, OUTPUT);
  pinMode (mpin10, OUTPUT);
  pinMode (mpin11, OUTPUT);
}

void StartMotor (int m1, int m2, int forward, int speed)
{
  if (speed == 0) // oprire
  {
    digitalWrite(m1, 0);
    digitalWrite(m2, 0);
  }
  else
  {
    if (forward)
```



# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

```
{  
    digitalWrite(m2, 0);  
    analogWrite(m1, speed); // folosire PWM  
}  
else  
{  
    digitalWrite(m1, 0);  
    analogWrite(m2, speed);  
}  
}
```

void delayStopped(int ms)

```
{  
    StartMotor (mpin00, mpin01, 0, 0);  
    StartMotor (mpin10, mpin11, 0, 0);  
    delay(ms);  
}
```

void playWithServo(char dir, int pin)

```
{  
    srv.attach(pin);  
    float poz = srv.read();  
    if (dir == 'N') {  
        srv.write(90);  
    }  
    else if (dir == 'V') {  
        srv.write(180);  
    }  
    else if (dir == 'E') {  
        srv.write(0);  
    }  
}
```





# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

## FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

```
delay(1000);  
srv.detach();  
}  
  
float calculeazaDistanța() {  
    digitalWrite(trigPin, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(trigPin, LOW);  
    float durata_1 = pulseIn(echoPin, HIGH);  
    digitalWrite(2, b);  
    b = 1 - b;  
    float distanta_1 = durata_1 / 58.2;  
    return distanta_1;  
}  
  
void pornesteN() {  
    StartMotor (mpin00, mpin01, 1, 128);  
    StartMotor (mpin10, mpin11, 0, 128);  
    delay (500);  
    delayStopped(500);  
}  
  
void pornesteS() {  
    StartMotor (mpin00, mpin01, 0, 128);  
    StartMotor (mpin10, mpin11, 1, 128);  
    delay (500);  
    delayStopped(500);  
}  
  
void pornesteV() {  
    StartMotor (mpin00, mpin01, 1, 128);
```



# UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

## FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE

```
StartMotor (mpin10, mpin11, 0, 0);  
delay (500);  
delayStopped(500);  
}
```

```
void pornesteE() {  
    StartMotor (mpin00, mpin01, 1, 0);  
    StartMotor (mpin10, mpin11, 0, 128);  
    delay (500);  
    delayStopped(500);  
}
```

```
void gasesteCale(char dir) {  
    int merge = 0;  
    float d = 0;  
    if (merge == 0){  
        if (dir != 'N') playWithServo('N', 8);  
        d = calculeazaDistanta();  
        if (d > 20) {  
            pornesteN();  
            merge = 1;  
        }  
    }  
    if (merge == 0){  
        if (dir != 'V') playWithServo('V', 8);  
        d = calculeazaDistanta();  
        if (d > 20) {  
            pornesteV();  
            merge = 1;  
        }  
    }  
    if (merge == 0){
```



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE**

```
if (dir != 'E' ) playWithServo('E', 8);  
d = calculeazaDistanta();  
if (d > 20) {  
    pornesteE();  
    merge = 1;  
}  
}  
if (merge==0){  
    pornesteS();  
    merge=1;  
}  
}  
  
void loop() {  
    if (mySerial.available()) {  
        dir = mySerial.read();  
        if (dir != 'S') {  
            playWithServo(dir, 8);  
            distanta = calculeazaDistanta();  
        }  
        if ((distanta <= 20 && distanta >= 0) && dir != 'S') {  
            digitalWrite(13, HIGH);  
            gasesteCale(dir);  
        }  
        else {  
            digitalWrite(13, LOW);  
            if (dir == 'N') {  
                pornesteN();  
            }  
            else if (dir == 'S') {  
                pornesteS();  
            }  
            else if (dir == 'V') {
```



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE**

```
    pornesteV();
}
else if (dir == 'E') {
    pornesteE();
}
}
}
else {
    distanta = calculeazaDistanta();
    if (distanta <= 20 && distanta >= 0) {
        digitalWrite(13, HIGH);
    }
    else {
        digitalWrite(13, LOW);
    }
}
}
```

## 4. Rezultate

Atata timp cat robotul este controlat prin intermediul modului Bluetooth, el se va deplasa, daca nu exista obstacole, in directia aleasa dintre cele patru directii posibile: N, S, V, E. Daca se alege o directie ce nu poate fi executata (exista obstacol pe acea directie), robotul cauta cu ajutorul senzorului cu ultrasunete o posibila directie de deplasare. Daca acesta are obstacole pe directiile N, E si V, el se va deplasa spre S. In momentul in care se detecteaza un obstacol la o distanta mai mica de 20 de cm, LED-ul conectat la pinul 13 al placii de dezvoltare Arduino UNO se va aprinde si va ramane aprins atata timp cat obstacolul ramane la o distanta mai mica decat 20 de cm. De asemenea, buzzer-ul conectat va incepe sa genereze sunete de frecventa redusa cand obstacolul este mai departe, frecventa acestora crescand cu apropierea obstacolului.



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE – CATEDRA CALCULATOARE**

## **5. Concluzii**

În concluzie, robotul controlat cu ajutorul modulelor prezentate mai sus are două moduri posibile de deplasare: fie controlat cu ajutorul telefonului mobil sau un laptop care are funcția de Bluetooth activată, fie autonom în momentul în care întâlnește obstacole. Tehnologia ce stă la baza robotului poate fi folosită pentru proiectarea unui aspirator inteligent sau a unei mașinuțe de jucărie. Algoritmul proiectului poate fi extins pentru a fi posibilă desenarea traseului urmat de robot, astfel putând fi utilizat pentru maparea și măsurarea diferitelor terenuri.