

Masina controlata prin bluetooth cu senzori

Autor

Radutoiu Vlad

Introducere

Proiectul consta in realizarea unei masini controlate prin bluetooth de pe telefon care se poate deplasa in orice directie cu senzor de oprire in cazul de intalnire a unui obstacol. Masina este creata cu ajutorul unei placute Arduino UNO.

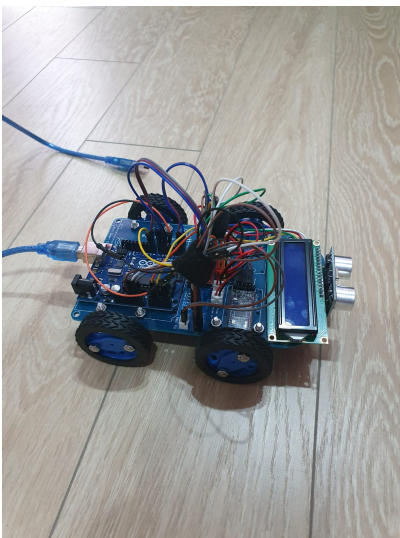
Placuta Arduino este conectata la un modul de antrenare pentru motoare unde vor fi conectate motoarele masinii. Se va folosi un senzor ultrasonic pentru a determina distanta fata de obiectele din jur si un ecran led pentru a afisa distanta fata de acestea.

Masina va avea o baterie si un power switch.

Descriere

Masina este pornita si asteapta comenzi din telefon. La detectarea unui obstacol la mai putin de o anumita distanta se va opri din deplasare in acea directie.

Schema Bloc



Schema Electrica



Componente Hardware

- Arduino UNO
- Modul Bluetooth HM-10
- ECRAN LCD 1602 - CHIP
- 4 x motoare N20 cu roți dințate
- Servomotor
- Senzor ultrasonic
- Power Switch
- Sasiu masina

Software Design

- Mediul de dezvoltare: Arduino IDE
- Biblioteci incluse: LiquidCrystal_I2C, SoftwareSerial, NewPing
- Functii implementate:
 - functia setup() initializeaza pinii de output pentru miscarea rotilor, initializeaza modulul de bluetooth pentru a primi comenzi si initializeaza conexiunea seriala la 9600 bps.
 - functia readPing este folosita pentru a citi distanta de pe senzorul ultrasonic
 - functia de loop citeste comenzi daca sunt primite pe modulul bluetooth si trateaza cazurile de miscare ale motoarelor. Comenzile de miscare sunt w,a,s,d iar pentru oprire este folosit x. Ecranul LCD afiseaza mereu distanta fata de obiectul din fata. Daca in fata mai sunt mai putin de 10 um masina se va opri si va afisa mesajul STOP.

Cod

```
#include <LiquidCrystal_I2C.h> #include <NewPing.h> #include <SoftwareSerial.h>
```

```
LiquidCrystal_I2C lcd(0x27,20,4);
```

```
// Motor A
```

```
int pwmA = 5; int in1A = 4; int in2A = 9;
```

```
char command; char mode;
```

```
Motor B int pwmB = 6; int in1B = 7; int in2B = 8; int distance = 100; SoftwareSerial HM10(0, 1);  
#define trig_pin A1 analog input 1 #define echo_pin A2 analog input 2 #define maximum_distance  
200 int MotorSpeed1 = 0; int MotorSpeed2 = 0; NewPing sonar(trig_pin, echo_pin,  
maximum_distance); sensor function
```

```
void setup() {
```

```
// put your setup code here, to run once:
```

```
Serial.begin(9600);  
HM10.begin(9600);  
pinMode(pwmA, OUTPUT);  
pinMode(pwmB, OUTPUT);  
pinMode(in1A, OUTPUT);  
pinMode(in2A, OUTPUT);  
pinMode(in1B, OUTPUT);  
pinMode(in2B, OUTPUT);
```

```
}
```

```
int readPing(){
```

```
delay(70);  
int cm = sonar.ping_cm();  
if (cm==0){  
    cm=250;  
}
```

```
return cm;
```

```
}
```

```
void loop() {
```

```
// put your main code here, to run repeatedly:
```

```
  HM10.listen();  
  if (HM10.available() > 0){  
    command = HM10.read();  
    lcd.print(command);  
    delay(1000);  
  }
```

```
  else {  
    command = ' ';  
  }
```

```
  lcd.init();  
  lcd.backlight();  
  lcd.clear();  
  lcd.setCursor(4,0);  
  distance = readPing();  
  lcd.print(distance);
```

```
if(distance < 10 && command != 's' && MotorSpeed1 !=0){  
  MotorSpeed1 = 0;  
  MotorSpeed2 = 0;  
  digitalWrite(in1A, HIGH);  
  digitalWrite(in2A, LOW);  
  digitalWrite(in1B, LOW);  
  digitalWrite(in2B, HIGH);  
  analogWrite(pwmA, MotorSpeed1);  
  analogWrite(pwmB, MotorSpeed2);  
  lcd.clear();  
  lcd.print("STOP");  
  delay(1000);  
  analogWrite(pwmA, MotorSpeed1);  
  analogWrite(pwmB, MotorSpeed2);  
}
```

```
if (command == 'w'&& distance > 10){  
  MotorSpeed1 += 100;  
  MotorSpeed2 += 100;  
  digitalWrite(in1A, LOW);  
  digitalWrite(in2A, HIGH);  
  digitalWrite(in1B, HIGH);  
  digitalWrite(in2B, LOW);  
  analogWrite(pwmA, MotorSpeed1);  
  analogWrite(pwmB, MotorSpeed2);  
  lcd.clear();  
  lcd.print("FORWARD");  
  delay(1000);
```

```
    analogWrite(pwmA, MotorSpeed1);
    analogWrite(pwmB, MotorSpeed2);

}
if (command == 's' ){
    MotorSpeed1 -= 100;
    MotorSpeed2 -= 100;
    digitalWrite(in1A, HIGH);
    digitalWrite(in2A, LOW);
    digitalWrite(in1B, LOW);
    digitalWrite(in2B, HIGH);
    analogWrite(pwmA, MotorSpeed1);
    analogWrite(pwmB, MotorSpeed2);
    lcd.clear();
    lcd.print("BACKWARD");
    delay(1000);
    MotorSpeed1 = 0;
    MotorSpeed2 = 0;
    analogWrite(pwmA, MotorSpeed1);
    analogWrite(pwmB, MotorSpeed2);

}
if (command == 'x' ){
    MotorSpeed1 = 0;
    MotorSpeed2 = 0;
    digitalWrite(in1A, HIGH);
    digitalWrite(in2A, LOW);
    digitalWrite(in1B, LOW);
    digitalWrite(in2B, HIGH);
    analogWrite(pwmA, MotorSpeed1);
    analogWrite(pwmB, MotorSpeed2);
    lcd.clear();
    lcd.print("STOP");
    delay(1000);
    analogWrite(pwmA, MotorSpeed1);
    analogWrite(pwmB, MotorSpeed2);

}
}
```

turn left if (command == 'a'){ MotorSpeed1 = 100; MotorSpeed2 = 100; digitalWrite(in1A, LOW); digitalWrite(in2A, HIGH); digitalWrite(in1B, LOW); digitalWrite(in2B, HIGH); analogWrite(pwmA, MotorSpeed1); analogWrite(pwmB, MotorSpeed2); lcd.clear(); lcd.print("LEFT"); delay(1000); MotorSpeed1 = 0; MotorSpeed2 = 0; analogWrite(pwmA, MotorSpeed1); analogWrite(pwmB, MotorSpeed2); } turn right

```
if (command == 'd'){
    MotorSpeed1 = 100;
    MotorSpeed2 = 100;
    digitalWrite(in1A, HIGH);
```

```
digitalWrite(in2A, LOW);  
digitalWrite(in1B, HIGH);  
digitalWrite(in2B, LOW);  
analogWrite(pwmA, MotorSpeed1);  
analogWrite(pwmB, MotorSpeed2);  
lcd.clear();  
lcd.print("RIGHT");  
delay(1000);  
MotorSpeed1 = 0;  
MotorSpeed2 = 0;  
analogWrite(pwmA, MotorSpeed1);  
analogWrite(pwmB, MotorSpeed2);  
}
```

```
delay(100);
```

```
}
```

Video

[Video Demo](#)

Concluzii

Proiectul nu a prezentat dificultati mari pe partea de hardware, conexiunile fiind destul de simple de realizat. Partea de software a prezentat probleme mai mari la conectarea cu modulul de bluetooth. Consider ca am invatat multe lucruri noi din acest proiect.

[Export to PDF](#)

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/masina_controlata_prin_bluetooth 

Last update: **2021/06/04 11:37**