**ROBOT CU SONAR SI CLAXON**

**Rus Renato**

**Grupa 30236**



**CODUL SURSA PENTRU PLACA ARDUINO UNO**

#include <Servo.h>

#include <TimerOne.h>

#include "pitches.h"

int melody[] = {

NOTE\_C4

};

int ton = 0;

// Pinii motor 1

#define mpin00 5

#define mpin01 6

// Pinii motor 2

#define mpin10 2

#define mpin11 11

const int trigPin = 9;

const int echoPin = 10;

long randomNumber;

long duration;

int distancee;

int stare = 0;

int stareMuzica = 0;

int counter = 0;

Servo srv;

void setup() {

// configurarea pinilor motor ca iesire, initial valoare 0

digitalWrite(mpin00, 0);

digitalWrite(mpin01, 0);

digitalWrite(mpin10, 0);

digitalWrite(mpin11, 0);

pinMode (mpin00, OUTPUT);

pinMode (mpin01, OUTPUT);

pinMode (mpin10, OUTPUT);

pinMode (mpin11, OUTPUT);

// pin LED

pinMode(13, OUTPUT);

// Functie pentru controlul unui motor

// Intrare: pinii m1 si m2, directia si viteza

//Senzorul de distanta

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

Serial.begin(9600);

playWithServo(8);

Timer1.initialize(500000);

Timer1.attachInterrupt(checkDistance);

}

void StartMotor (int m1, int m2, int forward, int speed)

{

if (speed == 0) // oprire

{

digitalWrite(m1, 0);

digitalWrite(m2, 0);

}

else

{

if (forward)

{

digitalWrite(m2, 0);

analogWrite (m1, speed); // folosire PWM

}

else

{

digitalWrite(m1, 0);

analogWrite(m2, speed);

}

}

}

// Functie de siguranta

// Executa oprire motoare, urmata de delay

void checkDistance() {

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distancee = duration \* 0.034 / 2;

Serial.print("Distance: ");

Serial.println(distancee);

counter++;

if(distancee < 40){

tone(3, NOTE\_C4);

}else {

noTone(3);

}

}

void goForward() {

StartMotor (mpin00, mpin01, 1, 128);

StartMotor (mpin10, mpin11, 1, 128);

stare = 1;

}

void goLeft() {

StartMotor (mpin00, mpin01, 0, 400);

StartMotor (mpin10, mpin11, 1, 128);

stare = 1;

Serial.println("Stanga");

}

void goRight() {

StartMotor (mpin00, mpin01, 0, 400);

StartMotor (mpin10, mpin11, 1, 128);

stare = 1;

Serial.println("Dreapta");

}

void Stop() {

StartMotor (mpin00, mpin01, 1, 0);

StartMotor (mpin10, mpin11, 1, 0);

stare = 0;

}

void playWithServo(int pin)

{

srv.attach(pin);

srv.write(0);

delay(1000);

srv.write(150);

delay(1000);

srv.write(90);

delay(1000);

srv.detach();

}

void playWithServoRigt(int pin)

{

srv.attach(pin);

srv.write(90);

srv.detach();

}

void playWithServoLeft(int pin)

{

srv.attach(pin);

srv.write(180);

srv.detach();

}

void loop() {

goForward();

if (distancee < 40) {

Stop();

Serial.println("Obstacol in fata");

}

if (stare == 0) {

goLeft();

}

}

Descriere :

Robotul a fost programat astfel incat sa se poata deplasa singur iar in cazul in care intalneste un obiect la o distanta mai mica de 40 de cm acesta scoate un sunet de avertizare si isi schimba directia de mers astfel incat sa evite acel obiect. Pentru a se deplasa, robotul se foloseste de motorul DC pe care il are la dispozitie si care este apelat de catre functia StartMotor(). Atunci cand intalneste un obiect se apeleaza functia goLeft(), functie care schimba directia si viteza de mers a rotilor astfel incat robotul sa poata evita cu succes obiectul. Pentru a stii tot timpul daca se indreapta spre un obiect sau nu, robotul se foloseste de sonarul din dotare si cu ajutorul functiei checkDistance() care se apeleaza cu ajutorul timer-ului la fiecare 1 secunda si care converteste datele primite de la sonar. Cu ajutorul buzzer-ului conectat la robot, acesta anunta sonor prezenta unui obiect in directia lui de mers pana il evita .