/\*obstacle avoiding, Bluetooth control, voice control robot car.

https://srituhobby.com

\*/

#include <Servo.h>

#include <AFMotor.h>

#define Echo A0

#define Trig A1

#define motor 10

#define Speed 170

#define spoint 103

char value;

int distance;

int Left;

int Right;

int L = 0;

int R = 0;

int L1 = 0;

int R1 = 0;

Servo servo;

AF\_DCMotor M1(1);

AF\_DCMotor M2(2);

AF\_DCMotor M3(3);

AF\_DCMotor M4(4);

void setup() {

Serial.begin(9600);

pinMode(Trig, OUTPUT);

pinMode(Echo, INPUT);

servo.attach(motor);

M1.setSpeed(Speed);

M2.setSpeed(Speed);

M3.setSpeed(Speed);

M4.setSpeed(Speed);

}

void loop() {

//Obstacle();

//Bluetoothcontrol();

//voicecontrol();

}

void Bluetoothcontrol() {

if (Serial.available() > 0) {

value = Serial.read();

Serial.println(value);

}

if (value == 'F') {

forward();

} else if (value == 'B') {

backward();

} else if (value == 'L') {

left();

} else if (value == 'R') {

right();

} else if (value == 'S') {

Stop();

}

}

void Obstacle() {

distance = ultrasonic();

if (distance <= 12) {

Stop();

backward();

delay(100);

Stop();

L = leftsee();

servo.write(spoint);

delay(800);

R = rightsee();

servo.write(spoint);

if (L < R) {

right();

delay(500);

Stop();

delay(200);

} else if (L > R) {

left();

delay(500);

Stop();

delay(200);

}

} else {

forward();

}

}

void voicecontrol() {

if (Serial.available() > 0) {

value = Serial.read();

Serial.println(value);

if (value == '^') {

forward();

} else if (value == '-') {

backward();

} else if (value == '<') {

L = leftsee();

servo.write(spoint);

if (L >= 10 ) {

left();

delay(500);

Stop();

} else if (L < 10) {

Stop();

}

} else if (value == '>') {

R = rightsee();

servo.write(spoint);

if (R >= 10 ) {

right();

delay(500);

Stop();

} else if (R < 10) {

Stop();

}

} else if (value == '\*') {

Stop();

}

}

}

// Ultrasonic sensor distance reading function

int ultrasonic() {

digitalWrite(Trig, LOW);

delayMicroseconds(4);

digitalWrite(Trig, HIGH);

delayMicroseconds(10);

digitalWrite(Trig, LOW);

long t = pulseIn(Echo, HIGH);

long cm = t / 29 / 2; //time convert distance

return cm;

}

void forward() {

M1.run(FORWARD);

M2.run(FORWARD);

M3.run(FORWARD);

M4.run(FORWARD);

}

void backward() {

M1.run(BACKWARD);

M2.run(BACKWARD);

M3.run(BACKWARD);

M4.run(BACKWARD);

}

void right() {

M1.run(BACKWARD);

M2.run(BACKWARD);

M3.run(FORWARD);

M4.run(FORWARD);

}

void left() {

M1.run(FORWARD);

M2.run(FORWARD);

M3.run(BACKWARD);

M4.run(BACKWARD);

}

void Stop() {

M1.run(RELEASE);

M2.run(RELEASE);

M3.run(RELEASE);

M4.run(RELEASE);

}

int rightsee() {

servo.write(20);

delay(800);

Left = ultrasonic();

return Left;

}

int leftsee() {

servo.write(180);

delay(800);

Right = ultrasonic();

return Right;

}