NAME- ROHAN SAINI

BRANCH-CST

COURSE-B.TECH

UNIVERSITY ROLL.NO-2017537

UNIVERSITY ID-20021427

OBSTACLE AVOIDING ROBOT CAR USING ARDINO

OBJECTIVE-To build an obstacle avoiding car using ardino.

INTRODUCTION: A robot that avoids the obstacle which comes in its path. Obstacle avoidance robot is design to allow robot to navigate in unknown environment by avoiding collisions. Obstacle avoiding robot senses obstacles in the path, avoids it and resumes its running. We have used sensors to achieve this objective. We have used four D.C.MOTORS i.e battery operated motors. The main component behind this robot is ArduinoUNO R3 microcontroller which is a brain of this robot.

MATERIAL REQUIRED:

-Hardware

Chasis

Ultrasonic Sensor

Arduino UNO

H-Bridge Motor Driver L293D

Jumper Wires

Batteries

Servo motor

-Software

Arduino IDE

ARDINO CODE:

```
#include<AFMotor.h>
#include<Servo.h>
int large_find(int []);
AF_DCMotor rightBack(2);                    //Create an object to control each motor
AF_DCMotor rightFront(4);
AF_DCMotor leftFront(3);
AF_DCMotor leftBack(1);
Servo servoLook;
byte trig=2;
byte echo=13;
byte motorSpeed=200;
byte maxDist = 150;
int motorOffset = 10;                       //Factor to account for one side being more powerful
int turnSpeed = 50;
float timeOut=2*(maxDist+10)/100/340*1000000;
```

```
void setup() {
  rightBack.setSpeed(motorSpeed);              //Set the motors to the motor speed
  rightFront.setSpeed(motorSpeed);
  leftFront.setSpeed(motorSpeed);
  leftBack.setSpeed(motorSpeed);
  rightBack.run(RELEASE);                      //Ensure all motors are stopped
  rightFront.run(RELEASE);
  leftFront.run(RELEASE);
  leftBack.run(RELEASE);                      //Assign the servo pin
  pinMode(trig,OUTPUT);                        //Assign ultrasonic sensor pin modes
  pinMode(echo,INPUT);
  servoLook.attach(10);
  Serial.begin(9600);
}

void loop() {
  servoLook.write(90);
  rightFront.run(FORWARD);
  leftFront.run(FORWARD);
  rightBack.run(FORWARD);
  leftBack.run(FORWARD);                       //Create a variable to store the pulse travel time
  int distance;
  int distances[4]={0,0,0,0};
  distance=get_distance();                    //Create a variable to store the calculated distance
  if(distance<=15)
  {
    rightBack.run(RELEASE);                    //Ensure all motors are stopped
    rightFront.run(RELEASE);
    leftFront.run(RELEASE);
```

```
leftBack.run(RELEASE);
delay(200);
rightBack.run(BACKWARD);
rightFront.run(BACKWARD);
leftFront.run(BACKWARD);
leftBack.run(BACKWARD);
delay(500);
rightBack.run(RELEASE);                    //Ensure all motors are stopped
rightFront.run(RELEASE);
leftFront.run(RELEASE);
leftBack.run(RELEASE);
servoLook.write(180);                      //left
delay(750);
distances[0]=get_distance();               //left
servoLook.write(135);                      //left-right
delay(400);
distances[1]=get_distance();               //left-right
servoLook.write(45);                      //right-left
delay(750);
distances[2]=get_distance();               //right-left
servoLook.write(0);                        //right side
delay(400);
distances[3]=get_distance();               //right
int lg=large_find(distances);
if(lg==0)                      //condition for turn left side
{
  turn_left(700);
}
else if(lg==1)                   //condition for turn right
```

```
   {
     turn_left(350);
   }
   else if(lg==2)
   {
     turn_right(350);
   }
   else if(lg==3)
   {
     turn_right(700);
   }
  }
}
void turn_left(int duration)
{
  rightBack.setSpeed(motorSpeed+turnSpeed);              //Set the motors to the motor speed
  rightFront.setSpeed(motorSpeed+turnSpeed);
  leftFront.setSpeed(motorSpeed);
  leftBack.setSpeed(motorSpeed);
  rightBack.run(FORWARD);
  rightFront.run(FORWARD);
  leftFront.run(BACKWARD);
  leftBack.run(BACKWARD);
  delay(duration);
  rightBack.setSpeed(motorSpeed);                    //Set the motors to the motor speed
  rightFront.setSpeed(motorSpeed);
  leftFront.setSpeed(motorSpeed);
  leftBack.setSpeed(motorSpeed);
  rightBack.run(RELEASE);
```

```
  rightFront.run(RELEASE);
  leftFront.run(RELEASE);
  leftBack.run(RELEASE);
}
void turn_right(int duration)
{
  rightBack.setSpeed(motorSpeed);              //Set the motors to the motor speed
  rightFront.setSpeed(motorSpeed);
  leftFront.setSpeed(motorSpeed+turnSpeed);
  leftBack.setSpeed(motorSpeed+turnSpeed);
  rightBack.run(BACKWARD);
  rightFront.run(BACKWARD);
  leftFront.run(FORWARD);
  leftBack.run(FORWARD);
  delay(duration);
  rightBack.setSpeed(motorSpeed);                    //Set the motors to the motor speed
  rightFront.setSpeed(motorSpeed);
  leftFront.setSpeed(motorSpeed);
  leftBack.setSpeed(motorSpeed);
  rightBack.run(RELEASE);
  rightFront.run(RELEASE);
  leftFront.run(RELEASE);
  leftBack.run(RELEASE);
}
int get_distance()
{
  int distances;
  unsigned long pulseTime;
    digitalWrite(trig, HIGH);                  //Generate a 10 microsecond pulse
```

```
    delayMicroseconds(10);
    digitalWrite(trig, LOW);
    pulseTime = pulseIn(echo, HIGH, timeOut);
    distances = (float)pulseTime * 340 / 2 / 10000;
    return distances;
}
int large_find(int distances[])
{
  int lg=0;
  int lg_dis=distances[0];
  int i=0;
  while(i<4)
  {
    if(distances[i]>lg_dis)
    {
      lg_dis=distances[i];
      lg=i;
    }
    i++;
  }
  return lg;
}
```

ARDUINO UNO R3:

Arduino is an open-source platform used for building electronics projects. Arduino consists of roboth a physical programmable circuit board and IDE that runs on your computer, used to write and upload computer code to the physical board. The Arduino IDE uses a simplified version of C++, making it easier to learn to program.

MOTOR DRIVER:

Driver section consists of Motor driver and four DC motors. Motor driver is used for driving motors because Arduino does not supply enough voltage and current to motor. Arduino sends commands to this motor driver and then it drives motors in any direction as we want. Working of obstacle avoidance robot is very interesting. Then Arduino drives the motor according to sensors' output. L293D motor driver is used. L293D can rotate the motor in the forward and reverse direction.

Sensor section:

This section contains Ultrasonic sensors. These section is used to sense the particular obstacle which comes in between its path.

Control section:

Control section consists of Arduino UNO R3 which is used for controlling whole process of obstacle avoidance. Arduino reads every pin from each component and acts accordingly.

APPLICATION:

This paper is all about Obstacle Avoidance Robot using Arduino which avoids obstacles which it encounters. In future this project can be enhance by connecting Bluetooth module and a camera so that the user can see the detected obstacle on his screen by sitting at just one place