Project Report

SUBMITTED TO:

Shams Shahriar.
Lecturer.
Daffodil International University.

SUBMITTED BY:

Group Name: “Delta”

Group Members:

1. Md. Anisur Rahman Rony (162-15- 7880)
2. Meherin Amir (162-15-7922)
3. Nujhat Tabassum Amithy (162-15-7750)
4. Abdullah Al Foysal (162-15-7683)

Project Name: “Obstacle Avoiding Robot Car”
Course Code: CSE 423.
Course Title: Embedded System.
Section: E.
Department of Computer Science and Engineering.

Project Overview

We are going to make an obstacle avoiding robot car using Arduino Uno. Our robot will detect obstacle by measuring the distance between the robot and sensor using Sonar Sensor. If an obstacle is found it will move slowly and stop. Then it will move another direction. A DC motor controller is used to control our geared motors of robot car. The motor controller is used to change the speed and direction of robot. The main components of our robot are:

1. Arduino Uno R3

2. Dual DC Motor Controller

3. Sonar Sensor

4. Directional Wheel

5. DC Geared Motors

6. Rechargeable Battery Unit

7. Robot Chassis

After the successful implementation of making obstacle avoiding robot car, we have another plan to add an actuator to perform another task. In this case, our plan is to add powerful servo motor as an actuator which will remove the obstacle from in front of it or it will hit it using a hammer attached with that servo motor. In extension another plan is to use a dc motor to cut the obstacle. Or the actuator will be used to perform another task. This is our future plan regarding our robot.

Previous Works

There are several similar works related our project.

In the first link given below, they used four geared motors for wheels of robot car, lithium ion battery for power supply. They used soldiering iron and glue gun for joining wires and different parts of robot. Plastic board was used to make the main structure of robot. One Arduino was kept over the board and Motor Driver Shield was connected with it. Wires of geared motors were connected with Motor Driver Shield. One servo motor was used to moving the sensor and that sensor was used for detecting obstacle by measuring distance between the robot and the sensor. Sensor was attached with servo motor. Finally, all the connections were given among the Arduino, motor driver, sensors and battery by different types of wires such as male to male, male to female and female to female wires. Different types of nuts were also used besides glue gun for joining parts. Then the code was uploaded to the robot for giving it all the instructions. And then finally the obstacle avoiding robot was made.

<u>They used below components for making their robot:</u>

1) Arduino Uno

2) Motor Driver Shield

3) Wheels

4) TT Gear Motor

5) Servo Motor

6) Ultrasonic Sensor

6) 18650 Li-on Battery

7) 18650 Battery Holder

8) Male and Female Jumper wire

9) Acrylic Sheet

10) DC Power Switch

Almost same works were done in second link.

In the third link only two geared motors were used instead of four motors and a bread board was also used. Two 9 Volt batteries were as power supply.

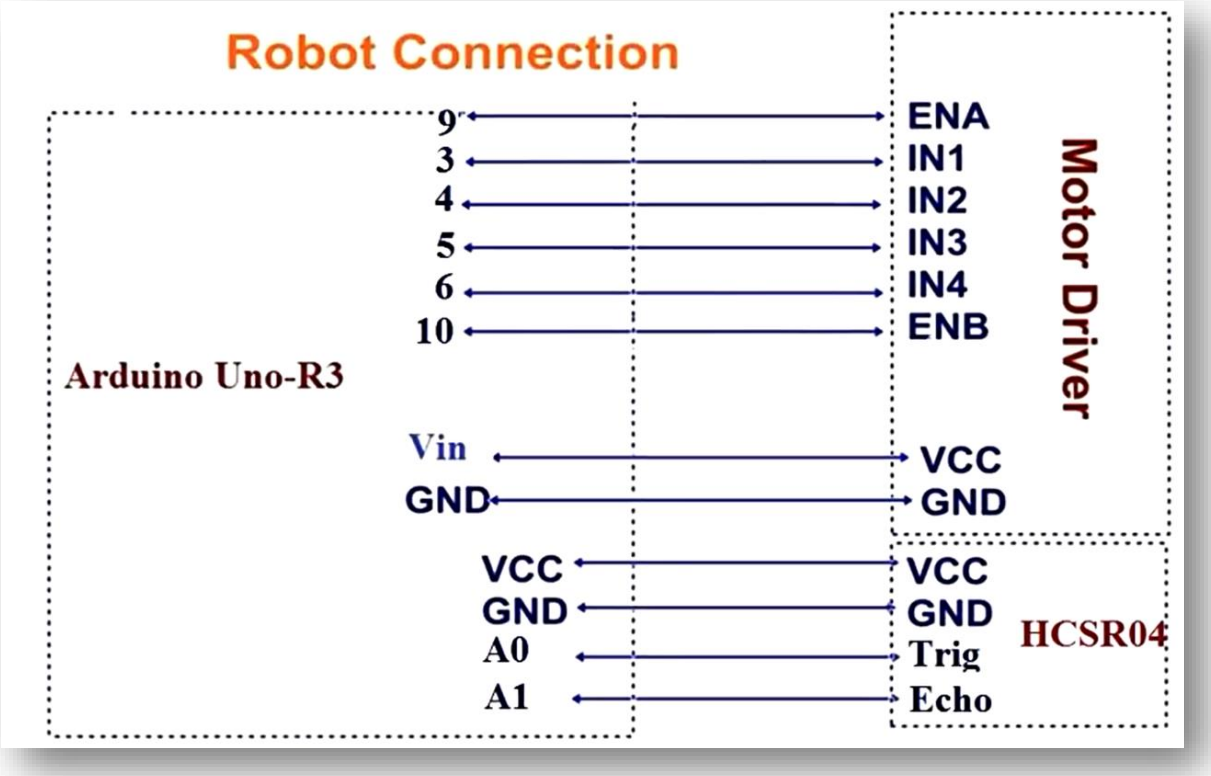
There are no differences between third and four links except the design and size of wheels.

In the link five the graphical representation of connection was explained briefly.

Serial	Links of Previous works related to our project
1.	https://www.youtube.com/watch?v=1n_KjpMfVT0
2.	https://www.youtube.com/watch?v=h-B42_HXL00
3.	https://www.youtube.com/watch?v=tXsP9STxdBc
4.	https://www.youtube.com/watch?v=d-d83QtFcnk
5.	https://www.youtube.com/watch?v=-RKXQrULcq4

Connection Diagram of Project

Connection diagram of our robot project is given below. This is actually the diagram of connections among Arduino, sensor and motor driver.



Project Code

The code which was used in our project is given below:

#define trigPin A0 //Assin PIN A0 as trigPin (Connect ARDUINO UNO "A0" pin with Ultrasonic Sonar Sensor "TRIG" Pin)

#define echoPin A1 //Assin PIN A1 as echoPin (Connect ARDUINO UNO "A1" pin with Ultrasonic Sonar Sensor "ECHO" Pin)

#define MotorA_IN1 3 //Assin PIN 4 as MotorA_IN1 (Connect ARDUINO UNO "3" pin with L293D Motor Driver "IN1" Pin)

#define MotorA_IN2 4 //Assin PIN 3 as MotorA_IN2 (Connect ARDUINO UNO "4" pin with L293D Motor Driver "IN2" Pin)

```
#define MotorB_IN3 5      //Assin PIN 7 as MotorB_IN3 (Connect ARDUINO UNO "5" pin with L293D Motor Driver "IN3" Pin)

#define MotorB_IN4 6      //Assin PIN 8 as MotorB_IN4 (Connect ARDUINO UNO "6" pin with L293D Motor Driver "IN4" Pin)

#define MotorA_PWM 9      //Assin PIN 5 as MotorA_PWM (Connect ARDUINO UNO "9" pin with L293D Motor Driver "ENA" Pin)

#define MotorB_PWM 10     //Assin PIN 6 as MotorB_PWM (Connect ARDUINO UNO "10" pin with L293D Motor Driver "ENB" Pin)

void setup() {

  pinMode(MotorA_IN1, OUTPUT);      //Declare "MotorA_IN1" as "Output Pin".

  pinMode(MotorA_IN2, OUTPUT);      //Declare "MotorA_IN2" as "Output Pin".

  pinMode(MotorB_IN3, OUTPUT);      //Declare "MotorB_IN3" as "Output Pin".

  pinMode(MotorB_IN4, OUTPUT);      //Declare "MotorB_IN4" as "Output Pin".

  pinMode(MotorA_PWM, OUTPUT);      //Declare "MotorA_PWM" as "Output Pin".

  pinMode(MotorB_PWM, OUTPUT);      //Declare "MotorA_PWM" as "Output Pin".

  pinMode(trigPin, OUTPUT);      //Declare "trigPin" as "Output Pin".

  pinMode(echoPin, INPUT);      //Declare "echoPin" as "Input Pin".

}


int search(void)

{

  float duration = 0.00;      //Float type variable declaration

  float CM = 0.00;

  digitalWrite(trigPin, LOW);     //Trig_pin uotput as 0V (Logic Low-Level)

  delayMicroseconds(2);     //Delay for 2 us

  //Send 10us High Pulse to Ultra-sonic Sonar Sensor "trigpin"

  digitalWrite(trigPin, HIGH);      //Trig-pin output as 5V (Logic High-Level)

  delayMicroseconds(10);     //Delay for 10 us
  digitalWrite(trigPin, LOW);     //Trig_pin uotput as 0V (Logic Low-Level)
  duration = pulseIn(echoPin, HIGH);      //Start counting time, upto again "echoPin" back to Logical "High-Level" and puting the "time" into a variable called "duration"
  CM = (duration / 58.82);    //Convert distance into CM.
  return CM;
}


void loop() {
  float distance = 0.00;
  float RobotSpeed = 0.00;
  //Measuring the distance in CM
  distance = search();
  if ((distance < 40))    //If obstaclw found in 40 CM.  {
    RobotSpeed = 100;     //Speed Down
    analogWrite(MotorA_PWM, RobotSpeed);      //Update speed in MOTORA Output Terminal
    analogWrite(MotorB_PWM, RobotSpeed);      //Update speed in MOTORB Output Terminal
    RobotStop();      //Robot Stop
    delay(10);
    RobotBackward();      //Robot Run Backward Direction
    delay(400);
    RobotStop();      //Robot Stop
    delay(10);
    distance = search();      //Check obstacle again
    int a = 250;
    int b = 250;
    if (distance < 30)  //30cm
    {
      RobotRight();     //Robot Turn into Right Direction
      a = a + 50;
      delay(a);
      distance = search();      //Check obstacle again
    }
    else
    {
      b = b + 50;
      RobotLeft();     //Robot Turn into Left Direction
      delay(b);
      distance = search();      //Check obstacle again
    }
  }
  else if ((distance >= 40) && (distance <= 70))
  {
    RobotSpeed = 150;     //Speed Increase Slightly
    analogWrite(MotorA_PWM, RobotSpeed);      //Update speed in MOTORA Output
Terminal
    analogWrite(MotorB_PWM, RobotSpeed);      //Update speed in MOTORB Output
Terminal
    RobotBackward();
  }
  else
  {
    RobotSpeed = 255;     //Speed increase to full speed
    analogWrite(MotorA_PWM, RobotSpeed);      //Update speed in MOTORA Output
Terminal
```

```
    analogWrite(MotorB_PWM, RobotSpeed);      //Update speed in MOTORB Output Terminal

    RobotForward();      //Robot Move to Forward Direction

  }

}


void RobotForward()

{

  digitalWrite(MotorA_IN1, HIGH);

  digitalWrite(MotorA_IN2, LOW);

  digitalWrite(MotorB_IN3, HIGH);

  digitalWrite(MotorB_IN4, LOW);

}


void RobotBackward()

{

  digitalWrite(MotorA_IN1, LOW);

  digitalWrite(MotorA_IN2, HIGH);

  digitalWrite(MotorB_IN3, LOW);

  digitalWrite(MotorB_IN4, HIGH);

}


void RobotLeft()

{

  digitalWrite(MotorA_IN1, LOW);

  digitalWrite(MotorA_IN2, HIGH);

  digitalWrite(MotorB_IN3, HIGH);
  digitalWrite(MotorB_IN4, LOW);
}

void RobotRight()
{
  digitalWrite(MotorA_IN1, HIGH);
  digitalWrite(MotorA_IN2, LOW);
  digitalWrite(MotorB_IN3, LOW);
  digitalWrite(MotorB_IN4, HIGH);
}

void RobotStop()
{
  digitalWrite(MotorA_IN1, LOW);
  digitalWrite(MotorA_IN2, LOW);
  digitalWrite(MotorB_IN3, LOW);
  digitalWrite(MotorB_IN4, LOW);
}
```

END