A

PROJECT REPORT ON

<u>Hindrance Avoiding Robot Car</u>

By

GRANTHY TEJASH DESAI (CE-24) (17CEUON045)
AYUSHI WALIA (CE-08) (17CEUON059)

BTech CE Semester-VI
Subject: System Design Practice

Guided by:
Prof. Pandav K. Patel
Assistant Professor
Dept. of Comp. Engg.

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

CERTIFICATE

This is to certify that the practical / term work carried out in the subject of **System Design Practice** and recorded in this journal is the bonafide work of

GRANTHY TEJASH DESAI(CE-24) (17CEUON045)
AYUSHI WALIA (CE-08) (17CEUON059)

of B. Tech semester **VI** in the branch of **Computer Engineering** during the academic year **2019-2020**.

Guided by	HOD
Prof. Pandav K. Patel	**Dr. C. K. Bhensdadia**,
Assistant Professor,	Head,
Dept. of Computer Engg.,	Dept. of Computer Engg.,
Faculty of Technology	Faculty of Technology
Dharmsinh Desai University, Nadiad	Dharmsinh Desai University, Nadiad

Table of Content

1. Abstract

In this work, we have presented a robot, which is compact, autonomous and fully functional.

This robot car is built to sense any obstacle in its path, to avoid it and resume its running involving the pre-computation of an obstacle free path. Ultrasonic sensors were adapted to implement a real-time obstacle avoidance system for wheeled robots, so that the robot can continually detect surroundings, avoid obstacles, and move toward the target area.

This model has tremendous applications in robot vacuum cleaners, avoiding concealed paths, parking systems, emergency rescue etc.

In conclusion, through this project, we aim to construct a model of a smart car that is beneficial to the problems of the present generation.

2. Introduction

Problems addressed

1. Need for a system which can detect obstacles and move in a pre-computed path.

2. Need for the detection of obstacles that can appear suddenly.

3. Need to minimize human risk regarding the upper limit of a human eye.

4. Need to assist the physically handicapped by incorporating cutting edge technologies in wheelchairs.

5. Need for automated vacuum cleaners that can work without human effort.

6. Need for easy parking systems to avoid dents and accidents.

Solutions offered

1. Robot detects obstacles and avoids them in the trajectory

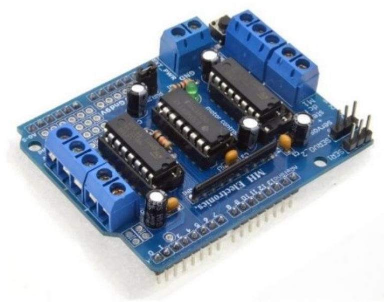
2. Robot uses an Ultrasonic Sensor which is capable of detecting obstacles which may appear suddenly, for instance, an animal in front of a vehicle.

3. Robot can operate in the environment without much interference.

4. Robot can detect very minute details, which the human eye may neglect.

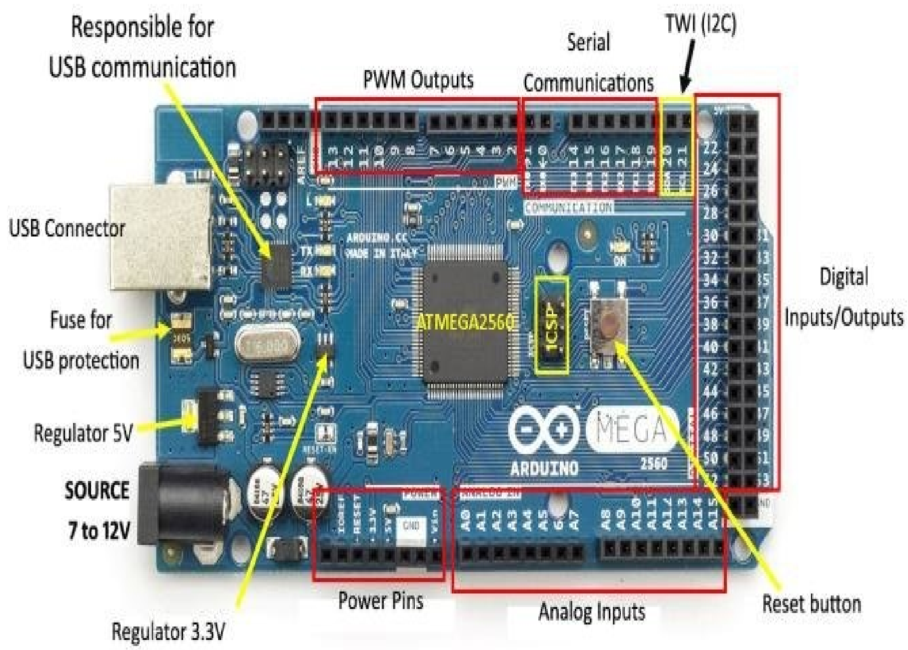
5. Robot can map various topographies and terrains which humans cannot.

3. Components Used

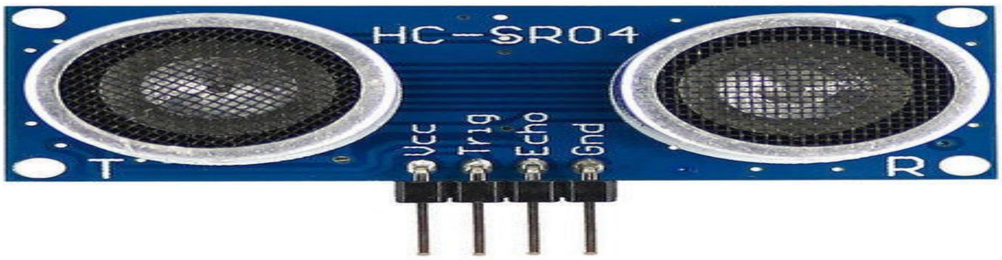


TT-motor Gear

Motor Driver



Arduino Mega



Ultrasonic Sensor

Jumper Wires ,Wheels, Motor

4. Features of the components used

1. Arduino Mega

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 .It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The power pins are as follows:

• **VIN**. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

● **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator.

● **3V3**. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

● **GND**. Ground pins.

Technical Specifications

Microcontroller	ATmega2560
Operating Voltage	5v
Input voltage(recommended)	7-12v
Input Voltage (limits)	6-20v
Digital I/O Pins	54
Analog Input pins	16
DC Current per I/O Pin	40 mA
DC current for 3.3v Pin	50 mA
Flash memory	256 KB
SRAM	8KB
EEPROM	4KB
Clock Speed	16 MHz

2. Ultrasonic Sensor

The Ultrasonic Sensor sends out a high-frequency sound pulse and then times how long it takes for the echo of the sound to reflect back. The sensor has two openings on its front.

- Tiny speaker to transmit opening ultrasonic waves
- Microphone to receive the ultrasonic waves

The ultrasonic sensor calculates distances by -

The speed of sound is approximately 341 meters per second in air. The ultrasonic sensor uses this information along with the time difference between sending and receiving the sound pulse to determine the distance to an object. It uses the following mathematical equation:

$$\textbf{Distance} = \frac{Time\ x\ Speed\ of\ Sound}{2}$$

Time = the time between when an ultrasonic wave is transmitted and when it is received

We divide this number by 2 since the sound wave has to travel to the object and back. The limitations of an ultrasonic sensor is some objects might not be detected it which are shaped or positioned in such a way that the sound wave bounces off the object, but are deflected away from the sensor.

It is also possible for the object to be too small to reflect enough of the sound wave back to the sensor to be detected.

Objects can absorb the sound wave all together (cloth, carpeting, etc), which means that there is no way for the sensor to detect them accurately.

These are important factors to consider when designing and programming a robot using an ultrasonic sensor.

Testing the components

The Arduino has two different types of input pins, those being analog and digital. We will focus on the Digital Input Pins.

To check the working of our ultrasonic sensor

- The serial monitor allows the computer to connect serially with the Arduino.

- It takes data that the Arduino is receiving from sensors and other devices and displays it in real-time on the computer.

- Having this ability is invaluable to debug the code and understand what number values the chip is actually receiving.

- Navigate to File --> Examples --> 1.Basics --> AnalogReadSerial.

- Click the button to engage the serial monitor which looks like a magnifying glass. We can now see the numbers being read by the analog pin in the serial monitor. When we turn the knob the numbers will increase and decrease. The numbers will be between the range of 0 and 1023. The reason for this is that the analog pin is converting a voltage between 0 and 5V to a discreet number. Digital inputs may be used as the basis for countless digital communication protocols.

- By creating a 5V (HIGH) pulse or 0V (LOW) pulse, we can create a binary signal, the basis of all computing.

- This is useful for talking to digital sensors like a PING ultrasonic sensor, or communicating with other devices.

- When the switch is pressed, the electrical connections in the switch has less resistance than the resistor, and the electricity no longer connects to ground. Instead, electricity flows between 5V and the digital pin.

- This is because electricity always chooses the path of least resistance.

Shields

●Shields are the expansion adapter boards that plug in over top of the Arduino Uno and gives it special functions.

● Since the Arduino is open hardware, anyone who has the inclination is free to make an Arduino shield for whatever task they wish to accomplish.

● Hence, there are countless number of Arduino shields available, out of which the official ones are -

● Wireless SD Shield
● Ethernet Shield
● Motor Shield

Motor Shield

The Arduino Motor Shield allows the user to easily control motor direction and speed using an Arduino.
Arduino pins are straightforward and hence it makes it very simple to incorporate a motor into a project. It also allows you to be able to power a motor with a separate power supply of up to 12V.

● The motor shield has 2 channels, which allows for the control of two DC motors, or 1 stepper motor.

● An external power supply, the motor shield can safely supply up to 12V and 2A per motor channel.

● There are pins on the Arduino that are always in use by the shield.

● By addressing these pins one can select a motor channel to initiate, specify the motor direction (polarity), set motor speed (PWM), stop and start the motor, and monitor the current absorption of each channel.

The pin breakdown is as follows:

Function	**Channel A**	**Channel B**
Direction	Digital 12	Digital 13
Speed (PWM)	Digital 3	Digital 11
Brake	Digital 9	Digital 8
Current Sensing	Analog 0	Analog 1

Procedure

Getting the hardware ready

● Switch it on and off and then look in to making the LED on the left of Arduino with the letter L next to it blink on and off for 2 seconds at a time.

● The Arduino Uno requires a male USB A to male USB B cable. Plug the USB cable in to the Arduino and your computer, the LED will start blinking. It is the default program stored on the chip.

● The USB cable powers the device. Arduinos can also run standalone by using a power supply in the bottom left of the board.

● Once programming is done it does not require to be constantly connected to a machine we can opt to power it separately. This is entirely dependant on the use case and circumstances we want to use the device in.

Getting the software ready

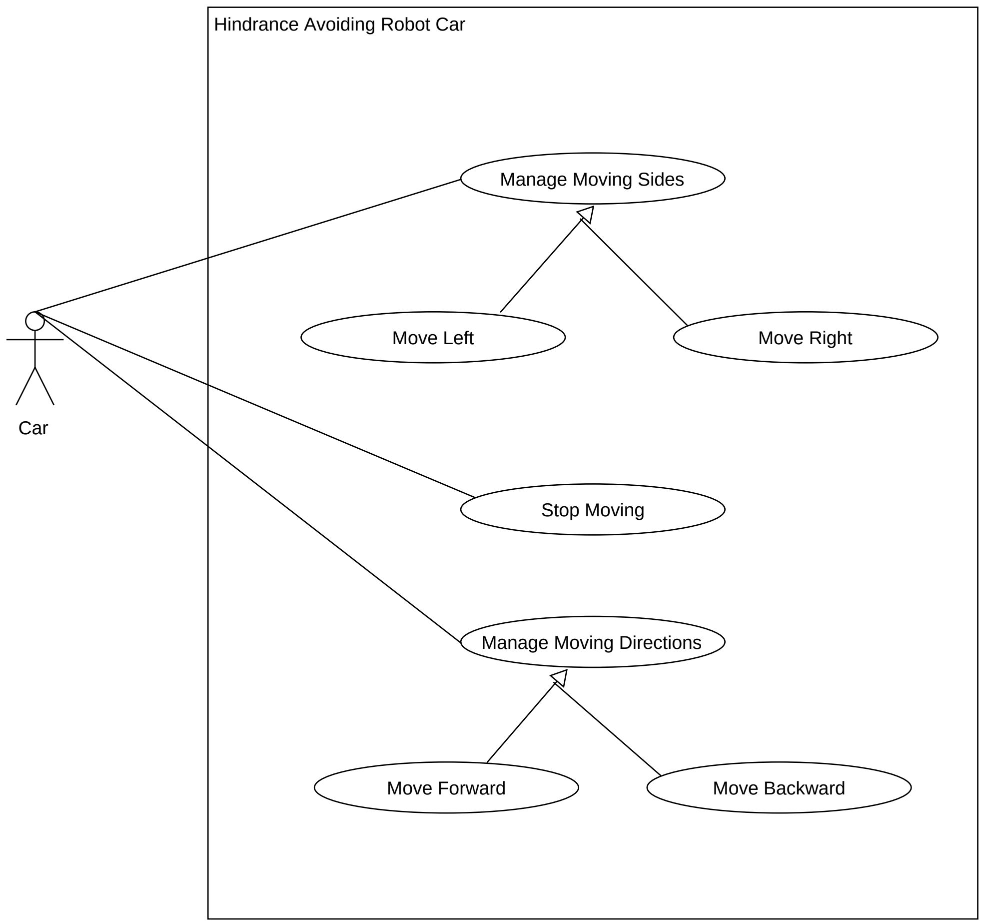
● Download and install the Arduino IDE (integrated development environment).

● To set the board, go to the following:

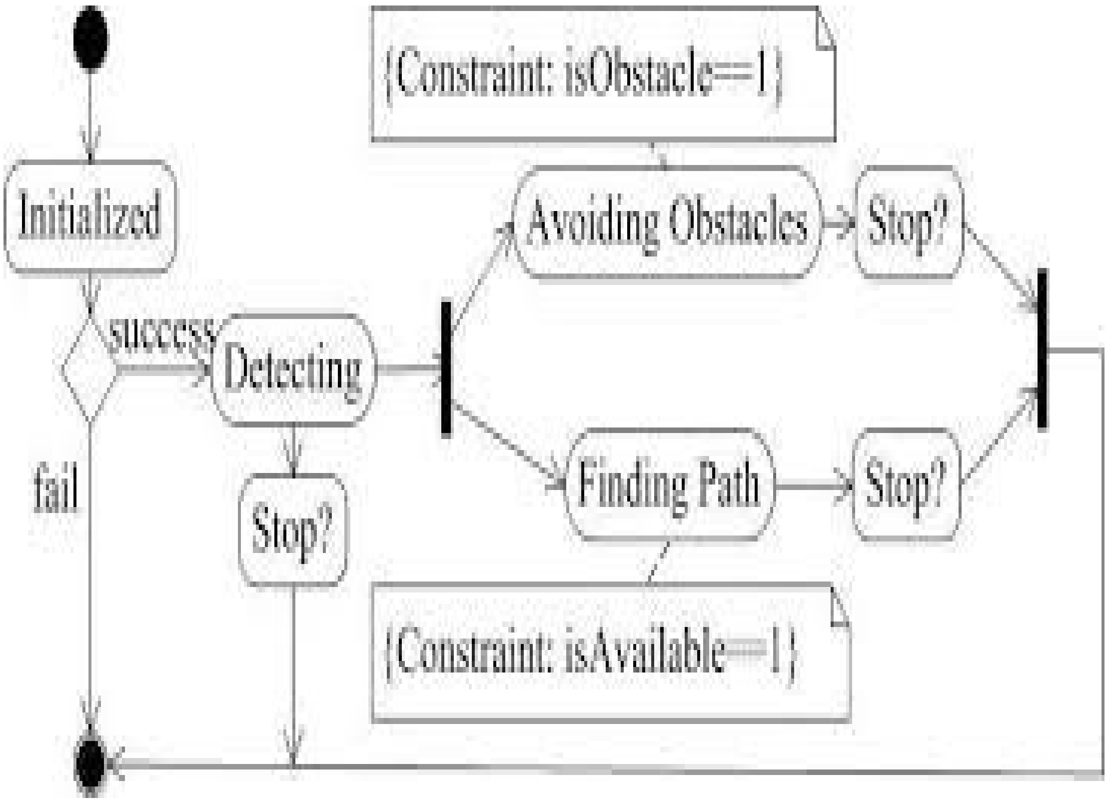
Tools-Boards
Select Arduino Mega

5. Designs

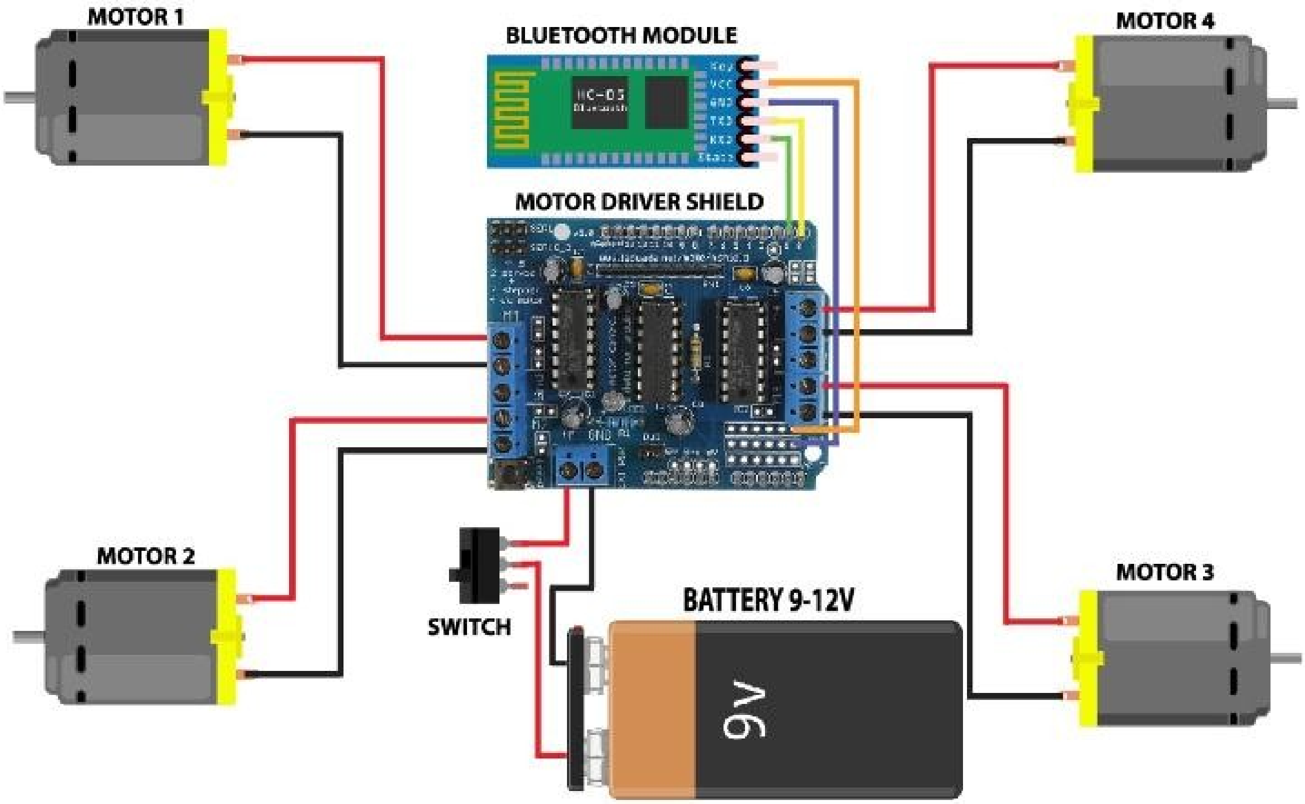
Use-Case Design:



Activity Diagram:



Circuit Diagram:



6. Applications:

• The device has application in surveying different landscapes and mapping them. It can also be used in commercial devices like:

Automated lawn mover
Smart room cleaner

• Hindrance avoiding robot car can be used in almost all mobile robot navigation systems.

• They can also be used in dangerous environments, where human penetration could be fatal.

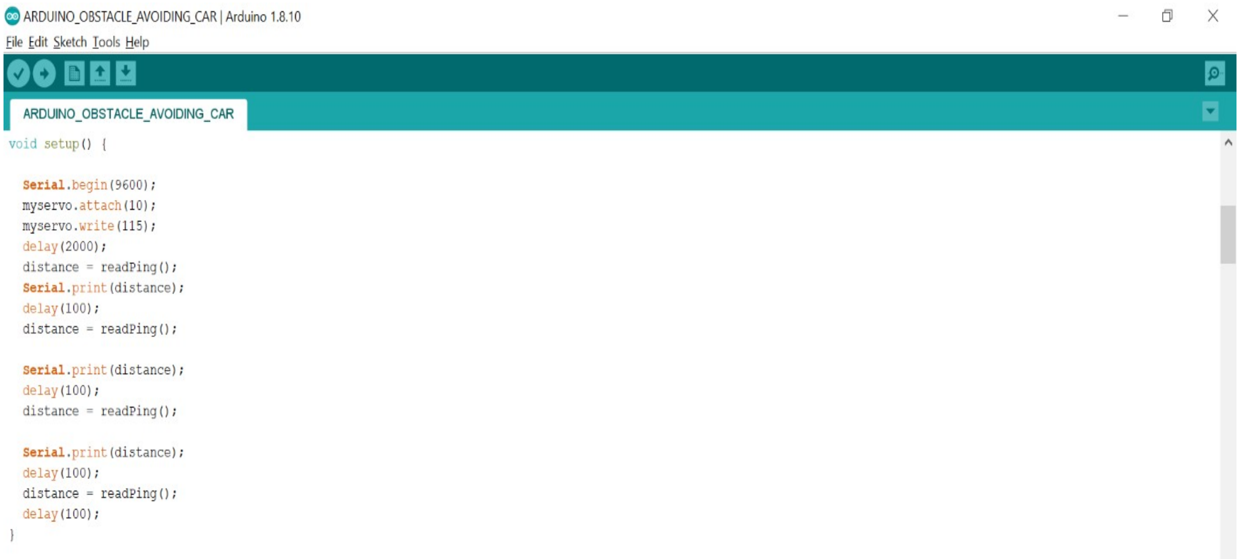
• Unmanned vehicle driving(Self Driving car).

• Mining Vehicle that uses Hindrance Detection.

7. Screenshots:

CODE:

To setup the project first



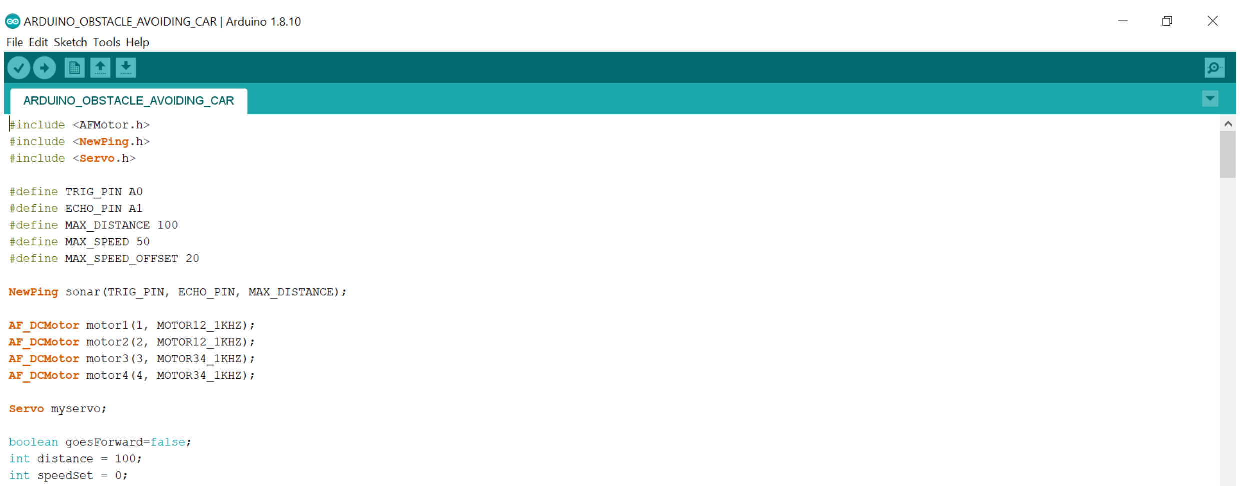
ARDUINO_OBSTACLE_AVOIDING_CAR | Arduino 1.8.10

File Edit Sketch Tools Help

ARDUINO_OBSTACLE_AVOIDING_CAR

```
void setup() {

  Serial.begin(9600);
  myservo.attach(10);
  myservo.write(115);
  delay(2000);
  distance = readPing();
  Serial.print(distance);
  delay(100);
  distance = readPing();

  Serial.print(distance);
  delay(100);
  distance = readPing();

  Serial.print(distance);
  delay(100);
  distance = readPing();
  delay(100);
}
```

The initialization part



ARDUINO_OBSTACLE_AVOIDING_CAR | Arduino 1.8.10

File Edit Sketch Tools Help

ARDUINO_OBSTACLE_AVOIDING_CAR

```
#include <AFMotor.h>
#include <NewPing.h>
#include <Servo.h>

#define TRIG_PIN A0
#define ECHO_PIN A1
#define MAX_DISTANCE 100
#define MAX_SPEED 50
#define MAX_SPEED_OFFSET 20

NewPing sonar(TRIG_PIN, ECHO_PIN, MAX_DISTANCE);

AF_DCMotor motor1(1, MOTOR12_1KHZ);
AF_DCMotor motor2(2, MOTOR12_1KHZ);
AF_DCMotor motor3(3, MOTOR34_1KHZ);
AF_DCMotor motor4(4, MOTOR34_1KHZ);

Servo myservo;

boolean goesForward=false;
int distance = 100;
int speedSet = 0;
```

To see if there are objects on either sides

```
}

int lookRight()
{

  Serial.write("look right");
    myservo.write(50);
    delay(500);
    int distance = readPing();
    delay(100);
    myservo.write(115);
    return distance;
}
int lookLeft()
{

  Serial.write("look Left");
    myservo.write(170);
    delay(500);
```

ARDUINO_OBSTACLE_AVOIDING_CAR | Arduino 1.8.10

File Edit Sketch Tools Help

ARDUINO_OBSTACLE_AVOIDING_CAR §

```
int lookLeft()
{

  Serial.write("look Left");
    myservo.write(170);
    delay(500);
    int distance = readPing();
    delay(100);
    myservo.write(115);
    return distance;
    delay(100);
}

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

To stop the car or to move it in either directions.

ARDUINO_OBSTACLE_AVOIDING_CAR | Arduino 1.8.10

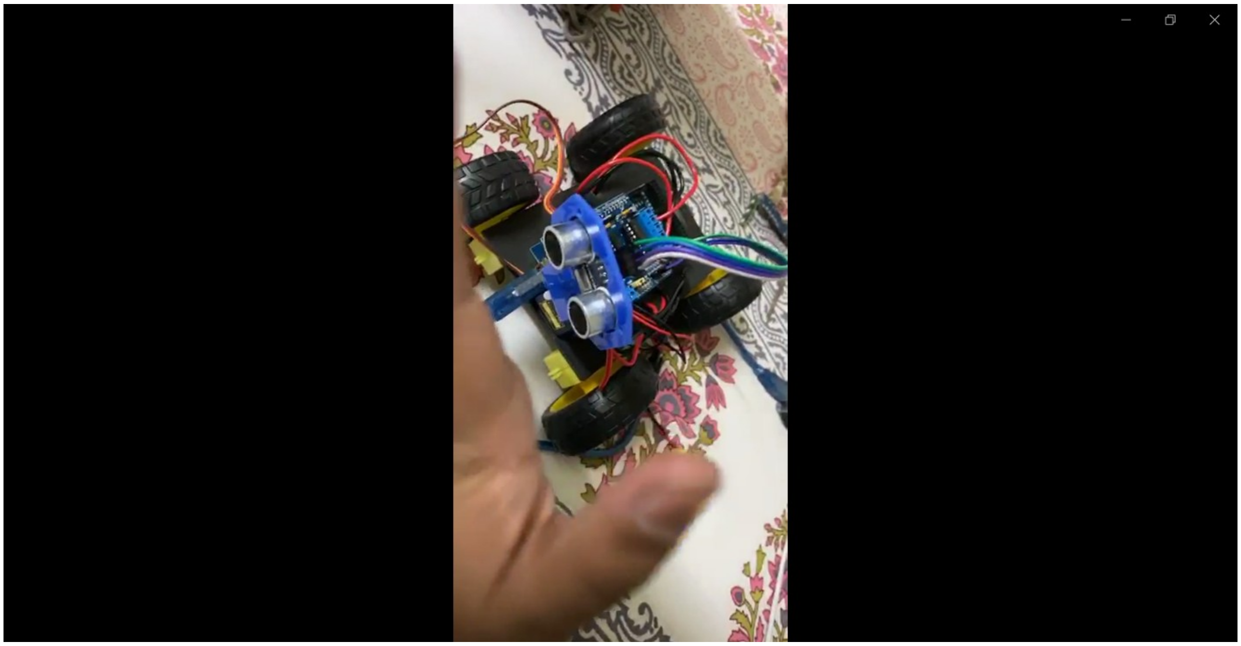
File Edit Sketch Tools Help

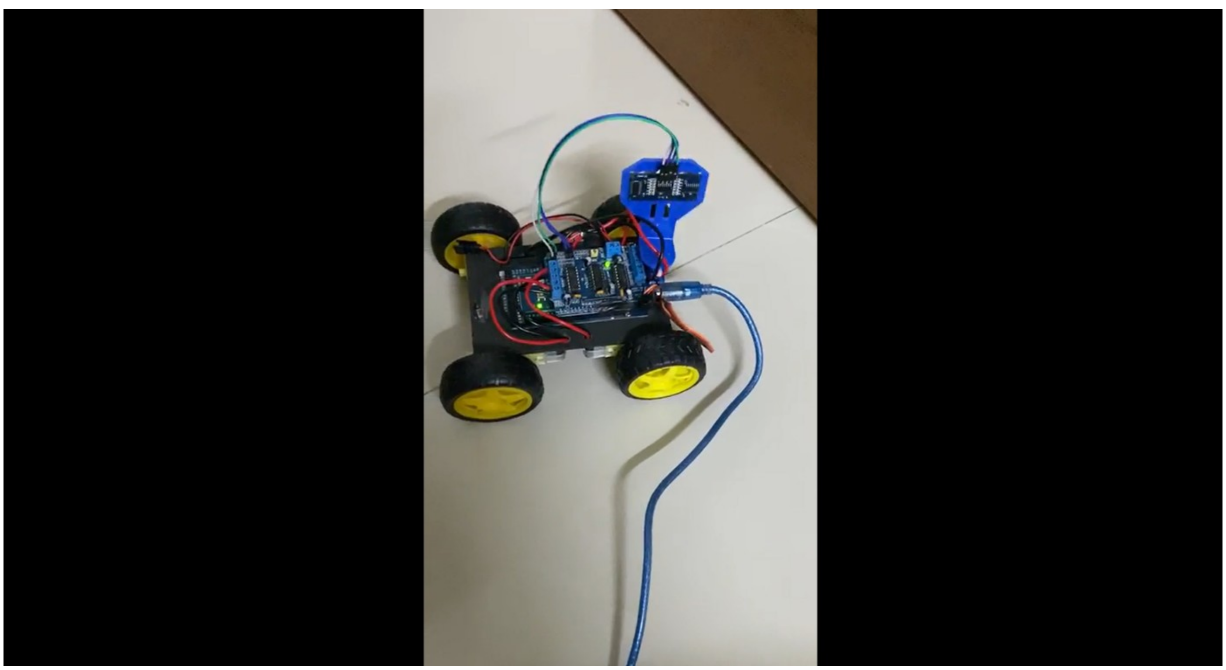
ARDUINO_OBSTACLE_AVOIDING_CAR §

```
void moveStop() {

  Serial.write("car stop");
  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);
  }

void moveForward() {

  Serial.write("move Forward");
 if(!goesForward)
  {
    goesForward=true;
    motor1.run(FORWARD);
    motor2.run(FORWARD);
    motor3.run(FORWARD);
    motor4.run(FORWARD);
   for (speedSet = 0; speedSet < MAX_SPEED; speedSet +=2)
   {
    motor1.setSpeed(speedSet);
    motor2.setSpeed(speedSet);
    motor3.setSpeed(speedSet);
    motor4.setSpeed(speedSet);
```

```
    motor4.setSpeed(speedSet);
    delay(5);
   }
  }
}

void moveBackward() {

  Serial.write("Move Backword");
    goesForward=false;
    motor1.run(BACKWARD);
    motor2.run(BACKWARD);
    motor3.run(BACKWARD);
    motor4.run(BACKWARD);
  for (speedSet = 0; speedSet < MAX_SPEED; speedSet +=2)
  {
    motor1.setSpeed(speedSet);
    motor2.setSpeed(speedSet);
    motor3.setSpeed(speedSet);
    motor4.setSpeed(speedSet);
    delay(5);
  }
```

8. Image of the car:





9. Videos of the working car:

https://drive.google.com/drive/folders/1mCKMWbzy5ejVtpyhhR-DoZ2K7QIfqp4U?usp=sharing

https://drive.google.com/open?id=1PM6x4ZbPIdom8ToG4SywVMPo2_25uZYv

https://drive.google.com/open?id=1CVIBzrC_uKT8AtyGD0QReS2RSIB_7ZdH

10. Cost Analysis:

S.NO.	**Component Name**	**Cost(in INR)**
1.	Arduino Mega	500
2	Motor Shield	230
1.	DC Motors X2	210
2.	Jumper Wires	90
3.	Ultrasonic Sensor	120
4.	Wires	50
	Total	**1200**

The cost of batteries, glue, sheet is extra and has not been included in the table above.

11. Conclusion:

The goal of our project is to create an autonomous robot which intelligently detects the obstacle in its path and navigate according to the actions we set for it.

Future Extensions:

There are few extensions which can be used in future:

- In future we are going to add facilities like the car can be monitored by other controls like:
  Voice control or bluetooth control or by a remote controller.
- Also according to location in which mode it shall be driven that is sports mode or traffic mode.
- IR sensors, Lidar can be added in future.
- Can be made for physically challenged People as they are dependent on people to take them wherever they want.

12. Bibliography:

www.google.com

www.tutorialspoint.com

www.geeksforgeeks.com