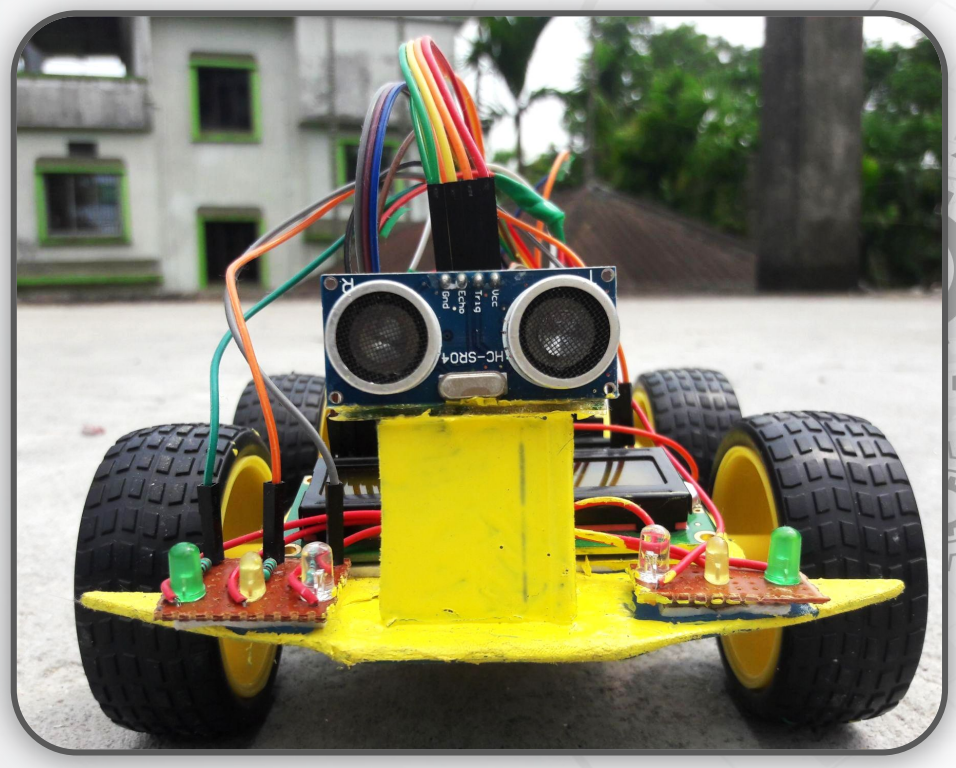
OBSTACLE DETECTION AND SPEED CONTROL OF VEHICLE IN MOTION



Presented by-

Group 6

- Avishek Biswas
- Utpalendu Barman
- Sourav Ghosh
- Akash Debnath
- Md Imran Hasan
- Saswata Nath
- Abhijit Sannyasi
- Arghyadev Sarkar

Under the supervision of

Prof. Atanu Maji

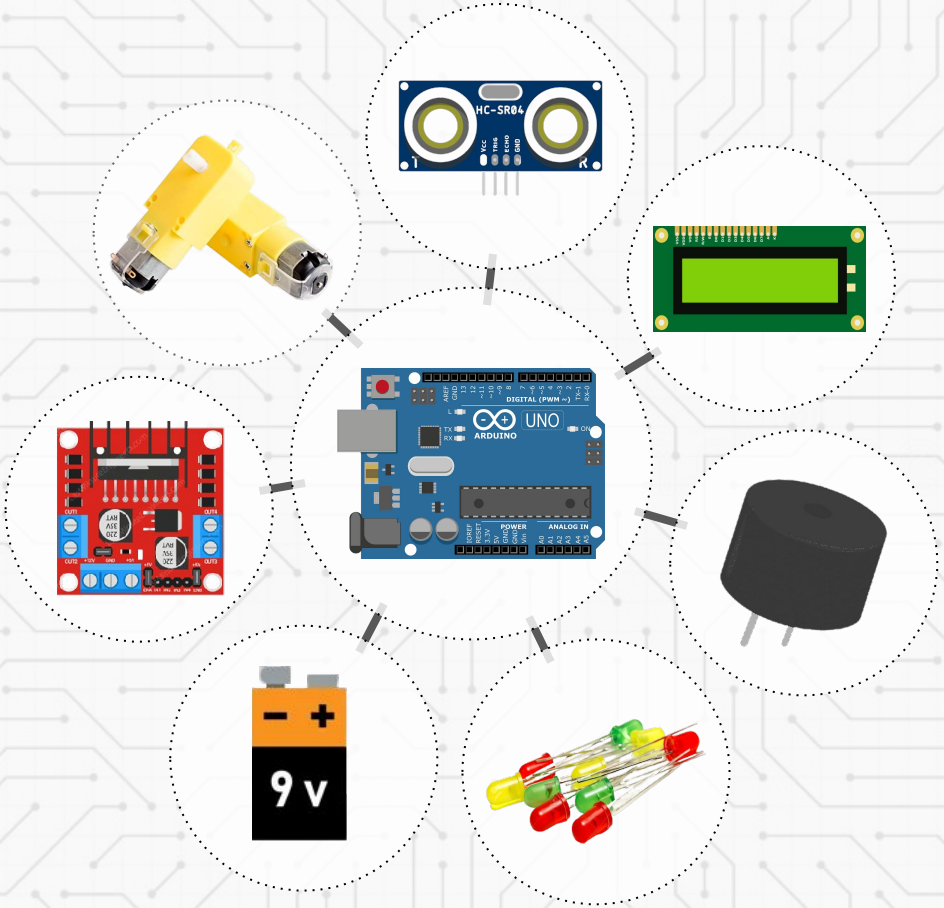
Department of Electrical Engineering

COOCH BEHAR ENGINEERING COLLEGE

INTRODUCTION

Various studies worldwide have concluded that numerous fatal accidents occur due to the negligence of driver. The reasons behind these are manifold.

In order to avoid such accidents, an idea has been brought out where the obstacles in front of a moving vehicle are detected by sensors, and the speed of the vehicles are reduced to an extent so that the accidents can be avoided.



Equipments Used

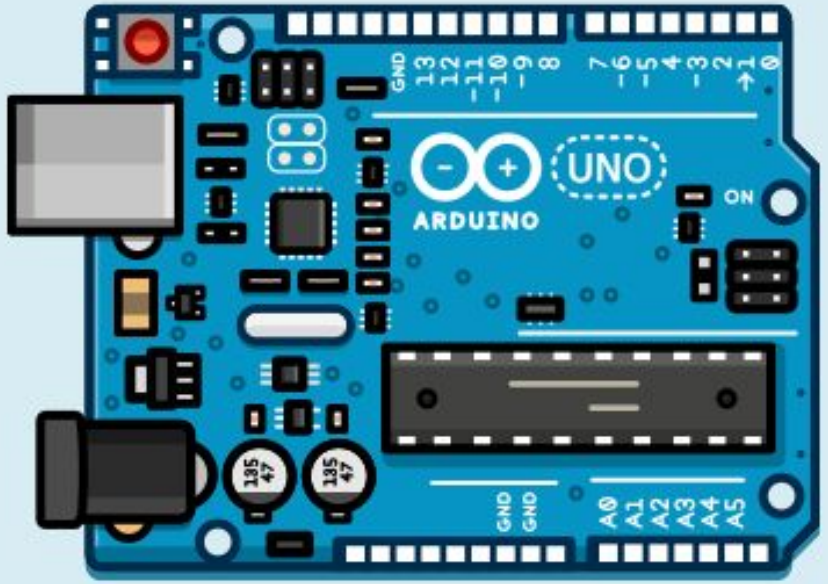
- ★ Arduino UNO R3
- ★ L298N Motor Driver
- ★ Gear Motors
- ★ Ultrasonic Sensor
- ★ 16 × 2 LCD
- ★ Buzzer
- ★ LED
- ★ Battery

- ★ Resistors
- ★ Jumper Wires

Arduino UNO R3

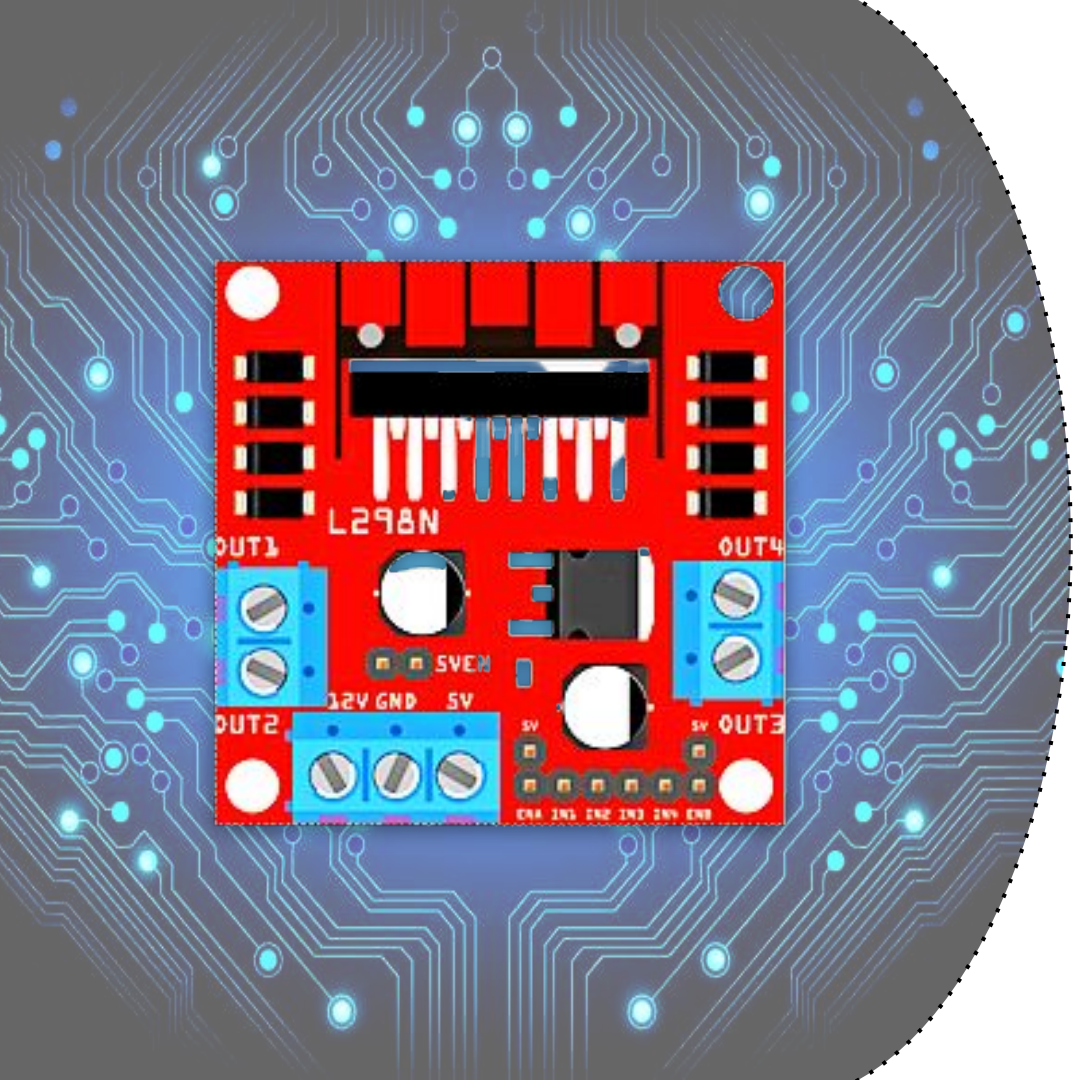
- Microcontroller board
- Operating voltage - 5V
- 14 Digital i/o pins
- 6 Analog i/o pins
- 6 PWM i/o pins
- Pin no. 13 - LED pin

- Program - Arduino IDE
- Flash memory - 32KB



Why Arduino UNO Rev3 is preferred here?

- ★ Compatibility over other models.
- ★ Light architecture and less interface.
- ★ Ease of performing desktop prototyping.
- ★ Ease of availability and cheaper.



L298N MOTOR DRIVER

- There are three sections-
  1. Power Section.
  2. Logical Input Section.
  3. Output section.
- Max. supply voltage - 35V
- Max. output Dc current - 4A

GEAR MOTOR

Gear motors are preferred over conventional motors, as the **momentum** can be easily controlled here.

- There are two parts in the Gear motor:
  - The DC motor.
  - The Gears.

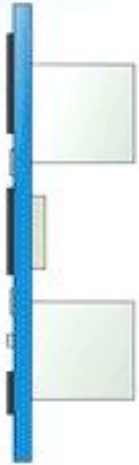
HIGH TORQUE AND , LESS MOTION

HC-SR04 Ultrasonic Sensor

TRIG port triggers ultrasound pulse,

ECHO port receives the same pulse after it gets reflected by the object.

Duration is gained from sensor, and **Distance** is calculated from that.



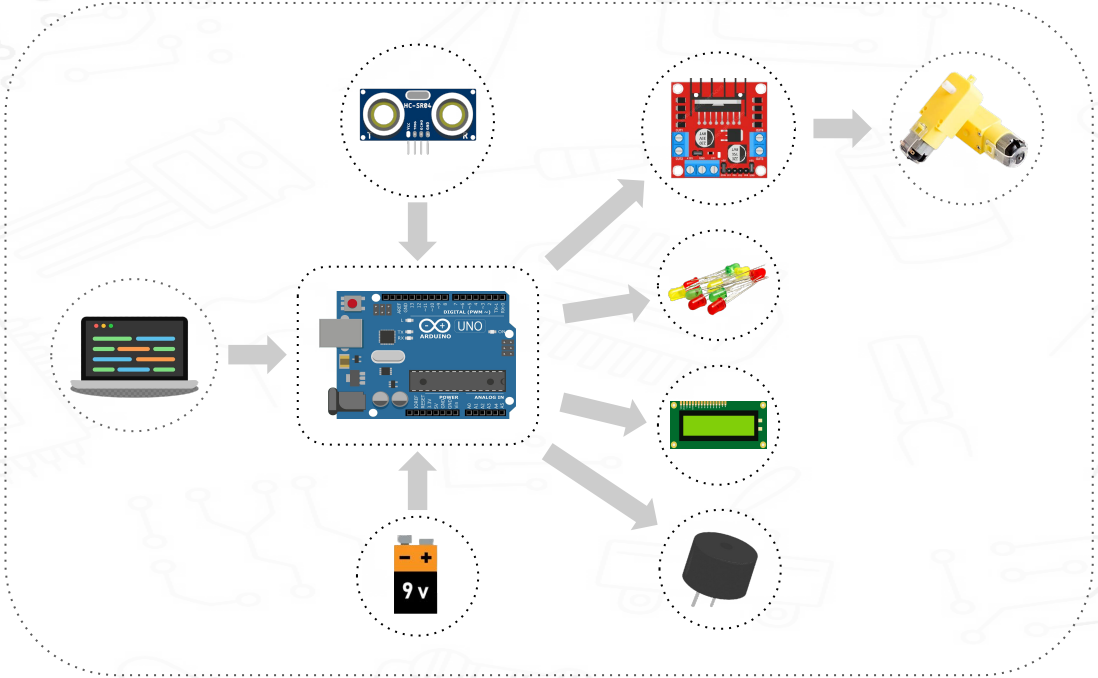
16 × 2 LCD Display

- Column count – 16
- Row count – 2

The distance between the obstacle and the car is displayed continuously in the LCD.

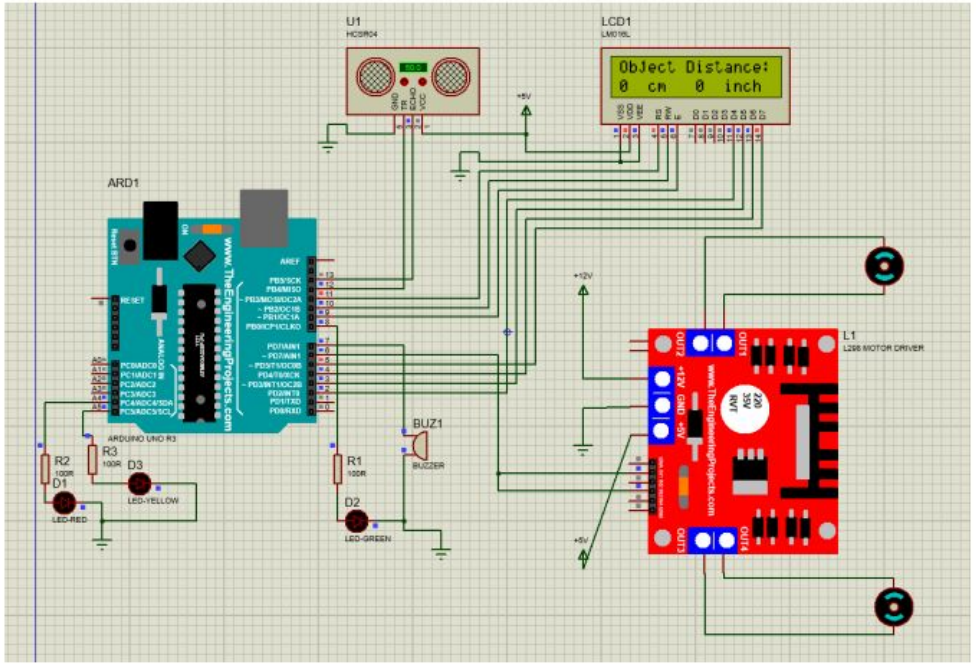
Working Principle

The Arduino code is provided to the microcontroller & the sensor output is fed to the same.



- The Motor driver input comes from the microcontroller based on the code.
- The Gear motors function according to the driver output.
- The other equipments, as LEDs, Buzzer, Display works based on the code given to the Arduino board.
- The +9V batteries provide the power to the entire circuit.

Circuit Diagram

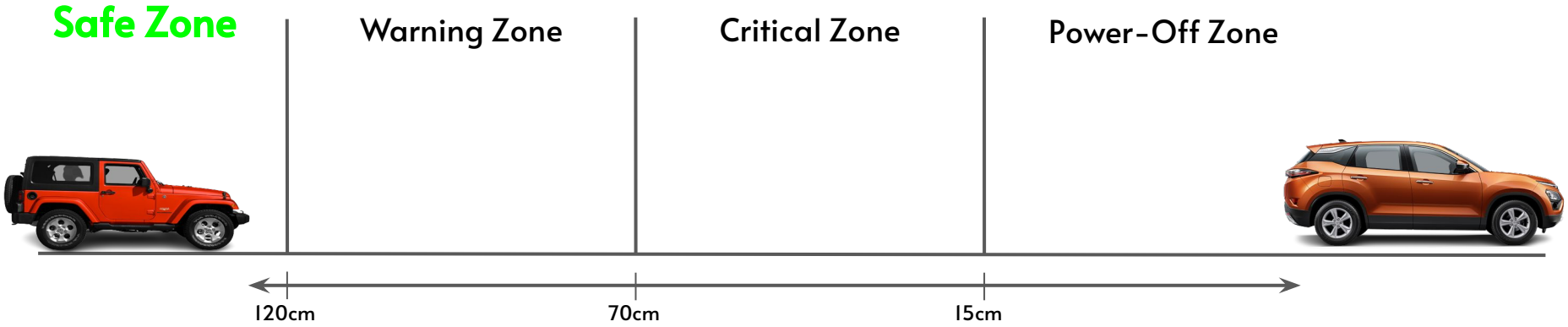


The circuit diagram is prepared in the simulation environment, with the help of Proteus Professional 8

All the connections among the equipments used, are presented in the diagram carefully, and keeping similar to the physical model.

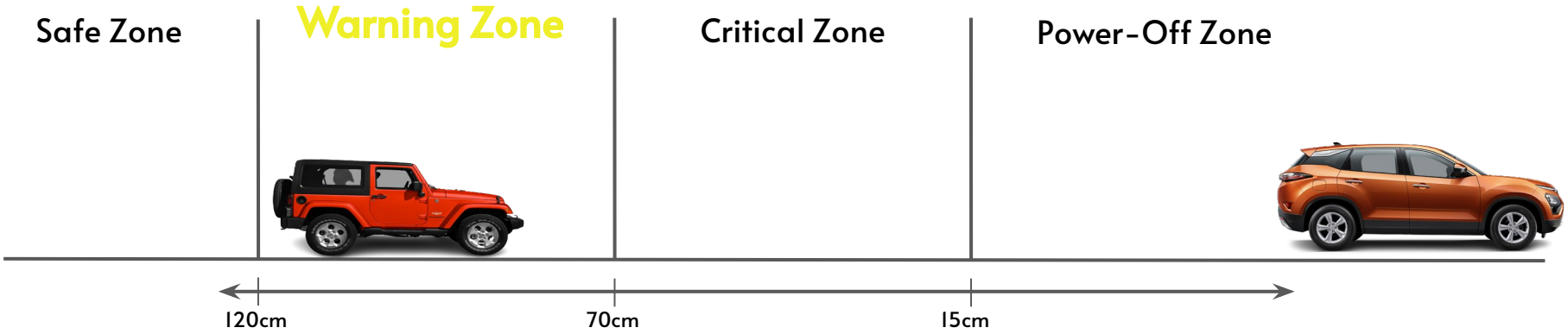
Safe Zone

- ★ The distance is - More than 120cm.
- ★ The car runs without any reduction in the speed.
- ★ Only Green LED remains turned ON.
- ★ No alarm is given to the driver.



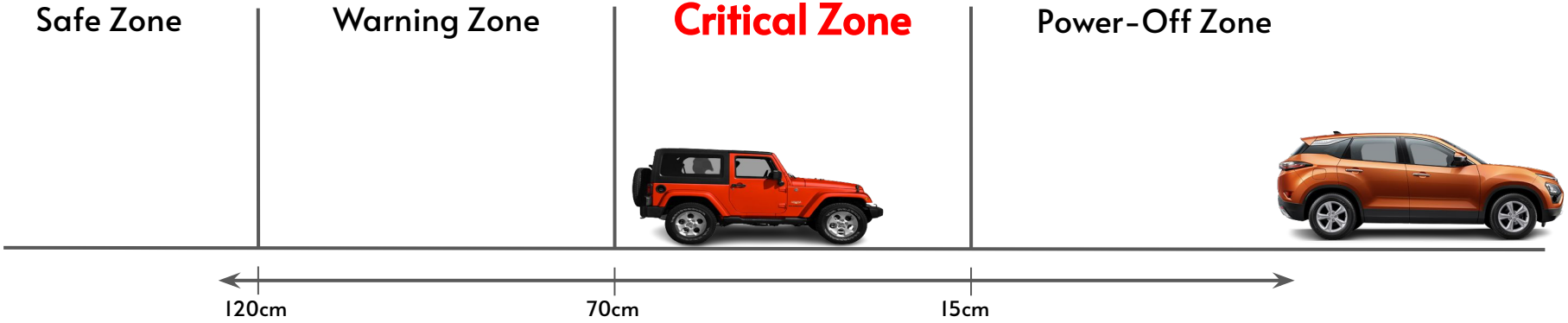
Warning Zone

- ★ The distance is - Between 120cm & 70cm.
- ★ The car runs without any reduction in the speed.
- ★ Only Yellow LED Blinking.
- ★ Alarm is Beeping to warn the driver.



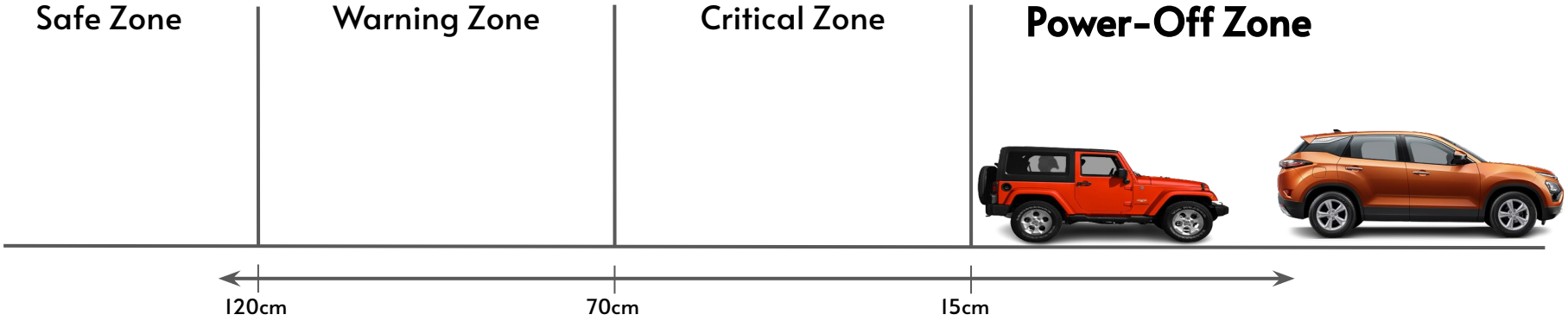
Critical Zone

- ★ The distance is - Between 70cm & 15cm.
- ★ Speed of the car reduces to a certain extent.
- ★ Only Red LED remains turned ON.
- ★ Alarm continuously Beeping.

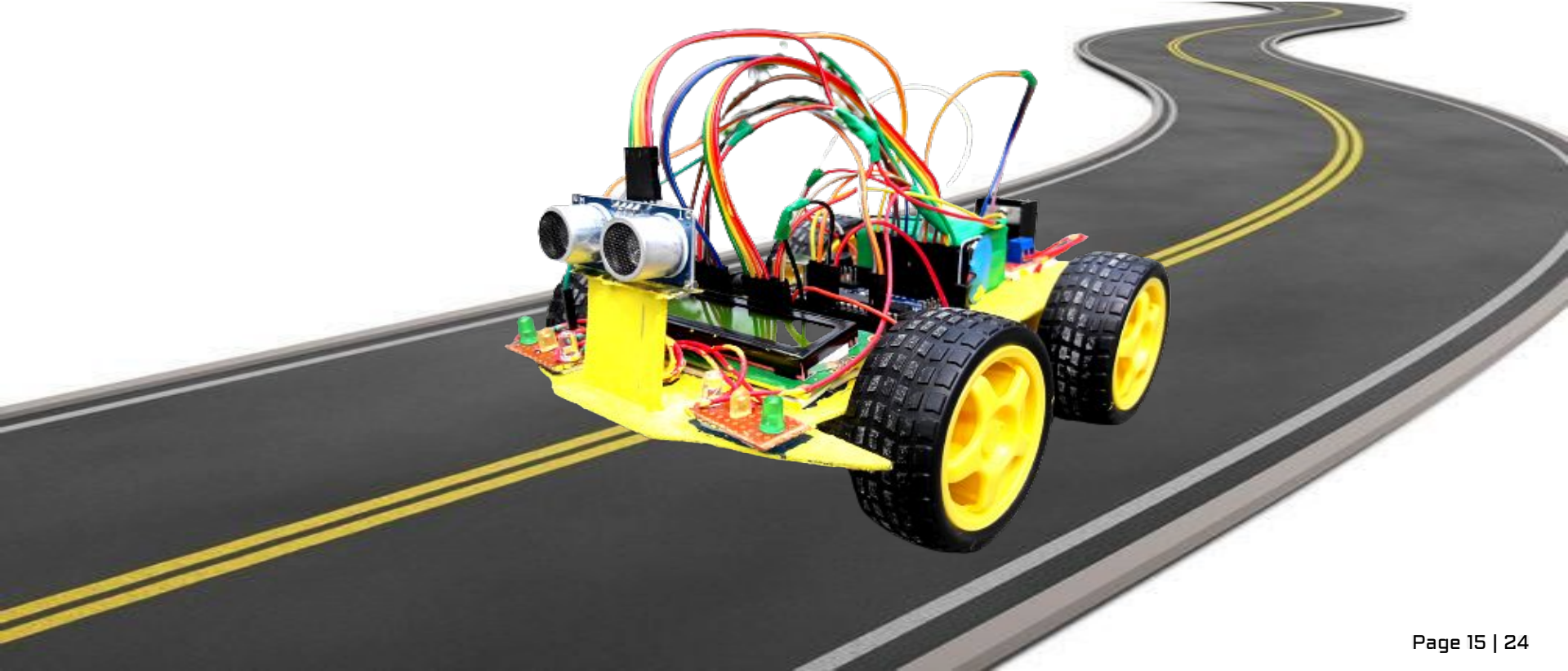


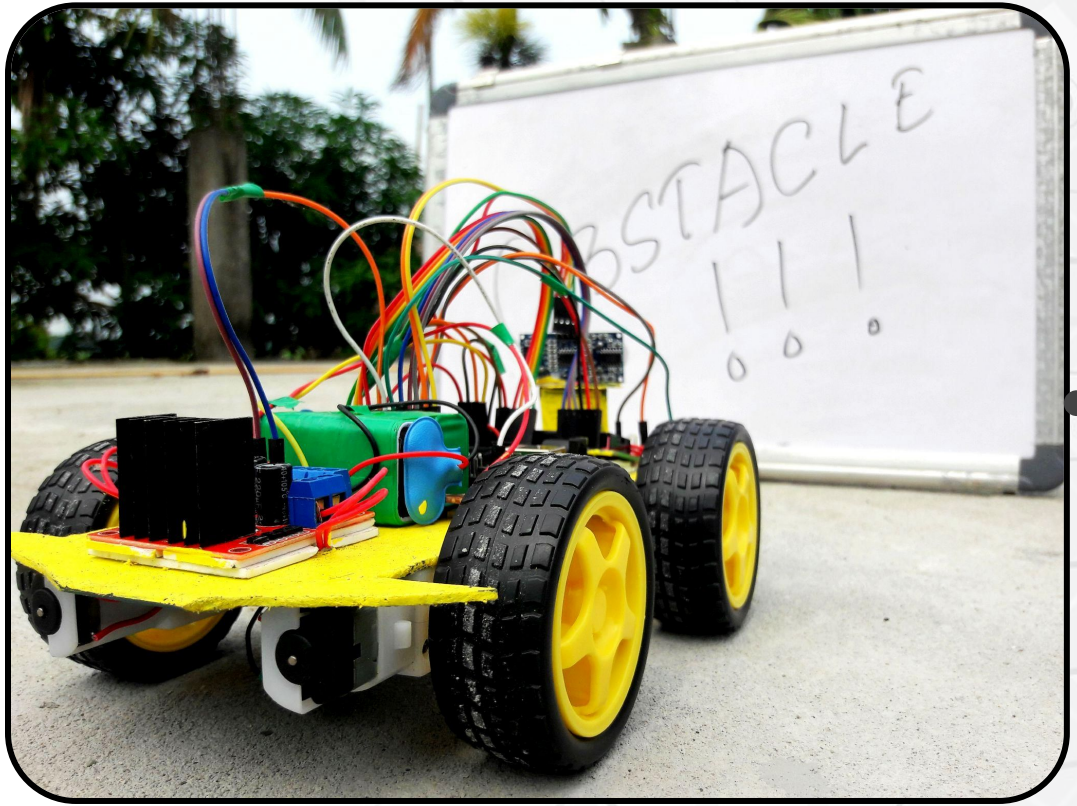
Power-Off Zone

- ★ The distance is - Less than 15cm
- ★ The car will stop.
- ★ All LEDs and Buzzer Off

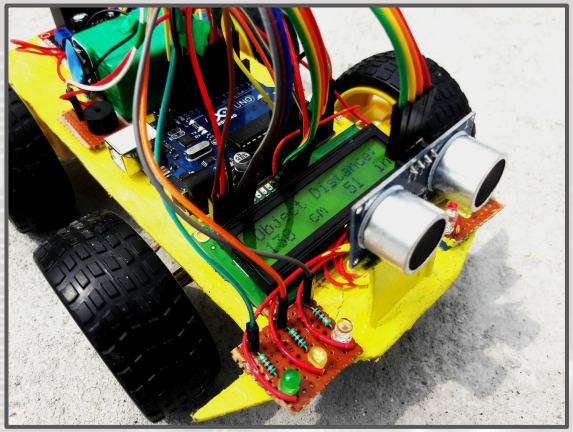


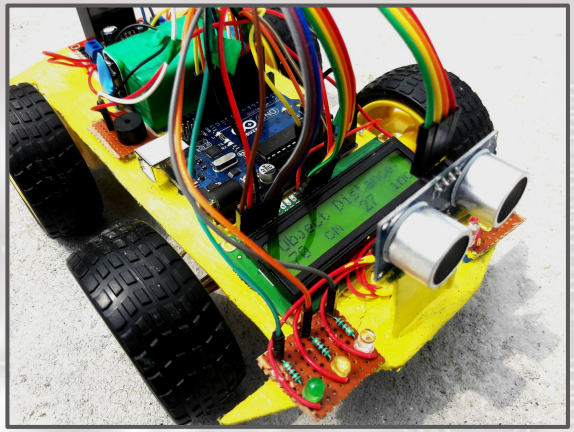
Result and Analysis

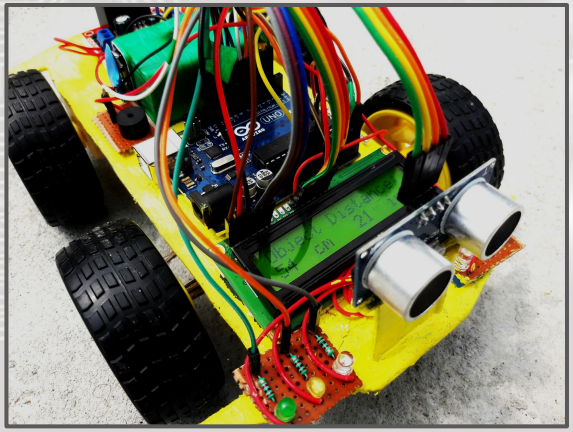


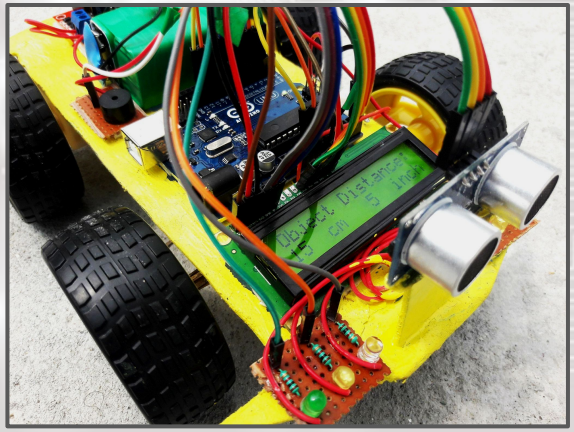


The obstacles in front of the vehicle are detected continuously by the Ultrasonic Sensor used in the model, and the distance of the object is displayed continuously as well









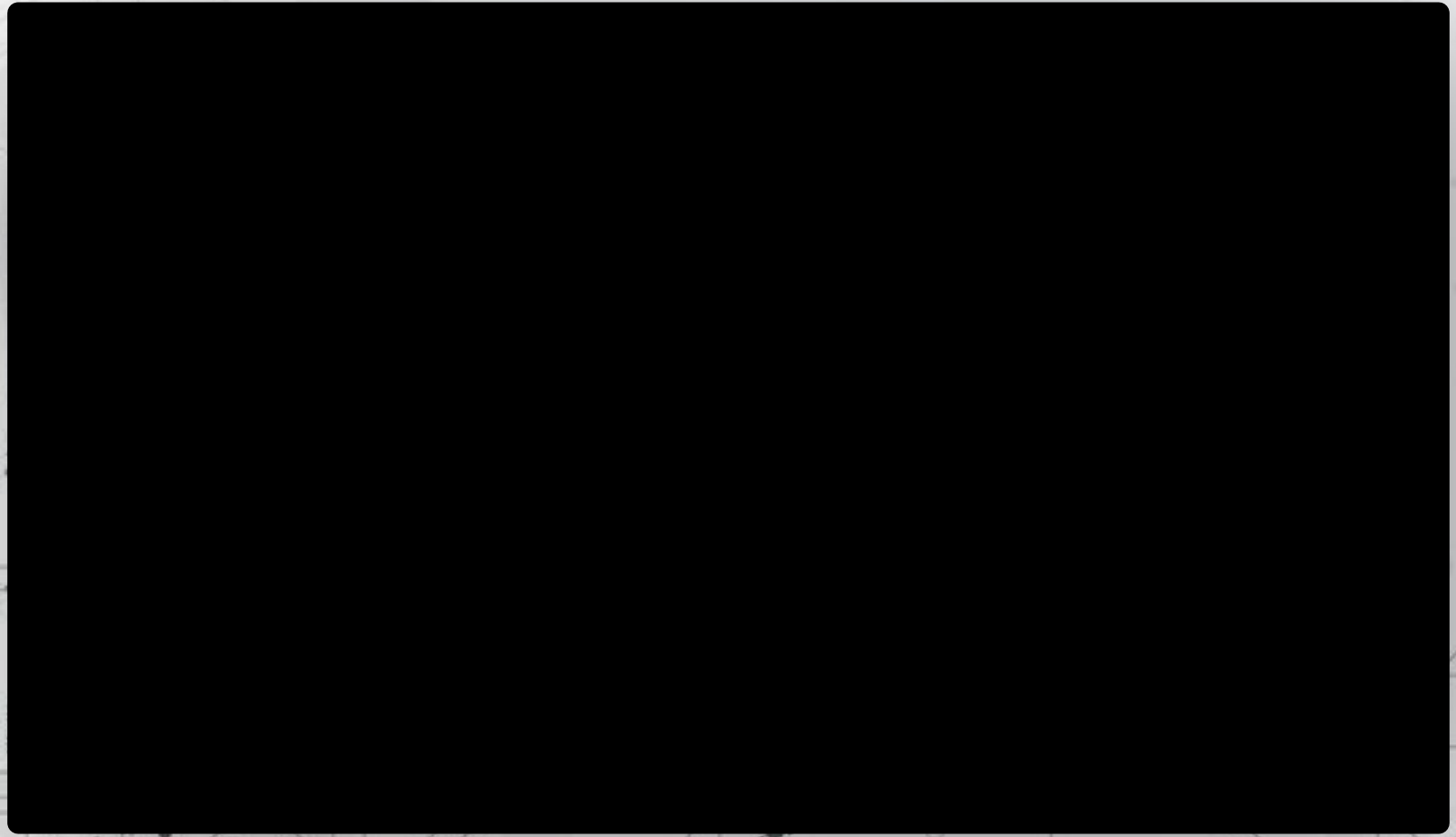
The distance of the obstacle are displayed in the LCD for four different scenarios:

Top-Left:
Safe Zone

Top-Right:
Warning Zone

Bottom-Left:
Critical Zone

Bottom-Right:
Power-Cut Zone



Arduino Code

Declaring Variables

```
#include<LiquidCrystal.h>
LiquidCrystal LCD(11, 10, 9, 2, 3, 4, 5);

int trigPin = 12;
int echoPin = 13;
int M = 6;
int buzzer = 7;
int green = 8;
int val1 = 255;
int val2 = 100;
int val3 = 0;
int red = A4;
int yellow = A5;

long duration, distance, inch;
```

Providing functions to the variables

```
void setup() {

  pinMode(M, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(green, OUTPUT);
  pinMode(red, OUTPUT);
  pinMode(yellow, OUTPUT);

  LCD.begin(16,2);
  LCD.setCursor(0,0);
  LCD.print("Object Distance:");
}
```

Code for sensor operations

```
void loop() {

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  inch = (distance/2.54);
```

Code for LCD operations

```
LCD.setCursor(0,1);
LCD.print("      ");
LCD.setCursor(0,1);
LCD.print(distance);
LCD.print("  cm");
LCD.print("    ");
LCD.print(inch);
LCD.print("  inch");
delay(250);
```

Code for Warning Zone

```
if(distance <= 120 && distance > 70){
  analogWrite(M, val1);
  digitalWrite(yellow, HIGH);
  digitalWrite(green, LOW);
  delay(45);
  digitalWrite(green, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(red, LOW);

  tone(buzzer, 1000);
  delay(200);
  noTone(buzzer);
  delay(200);
  }
```

Arduino Code

Code for Critical Zone

```
else if(distance <= 70){
  analogWrite(M, val2);
  digitalWrite(yellow, LOW);
  digitalWrite(red, HIGH);
  digitalWrite(green, LOW);
  digitalWrite(buzzer, HIGH);
  }
```

Code for Power-Off Zone

```
else if (distance <= 15){
 analogWrite(M, val3);
 digitalWrite(buzzer, LOW);
 digitalWrite(green, LOW);
 digitalWrite(yellow, LOW);
 digitalWrite(red, LOW);
 }
```

Code for Safe Zone

```
else{
 analogWrite(M, val1);
 digitalWrite(buzzer, LOW);
 digitalWrite(green, HIGH);
 digitalWrite(yellow, LOW);
 digitalWrite(red, LOW);
 }
```

FUTURE SCOPE

FUTURE SCOPE

★ We can add voice commands and other automations like this.

★ We can add customizability to the users, where the distance can be changed by them.

★ Automatic emergency call to kinsmen if accident happens.

Thank You