American International University- Bangladesh

Department of Computer Science

OEL Lab Report Cover Sheet

Course Name	MICROPROCESSOR AND EMBEDDED SYSTEMS
Lab Report No.	10 (OEL)
Lecturer Name	**MD. ALI NOOR**
Semester	SUMMER 2021-22
Submission Date	11/08/2022
Section	O
Group No.	03

Student Name	Student ID	Contribution (out of 100%)
1. EMON, MEHEDI HASAN	19-41118-2	
2. ASIF AHAMMED	19-40020-1	
3. EASINUL ABEDIN SAYEM	19-40291-1	
4. MD REDOY SHEIKH	19-40101-1	
5. MD. NAZIM HASAN	19-40118-1	
6. MOHI UDDIN ANANDO	20-43585-1	

Title: Speed control system based on a obstacle distance.

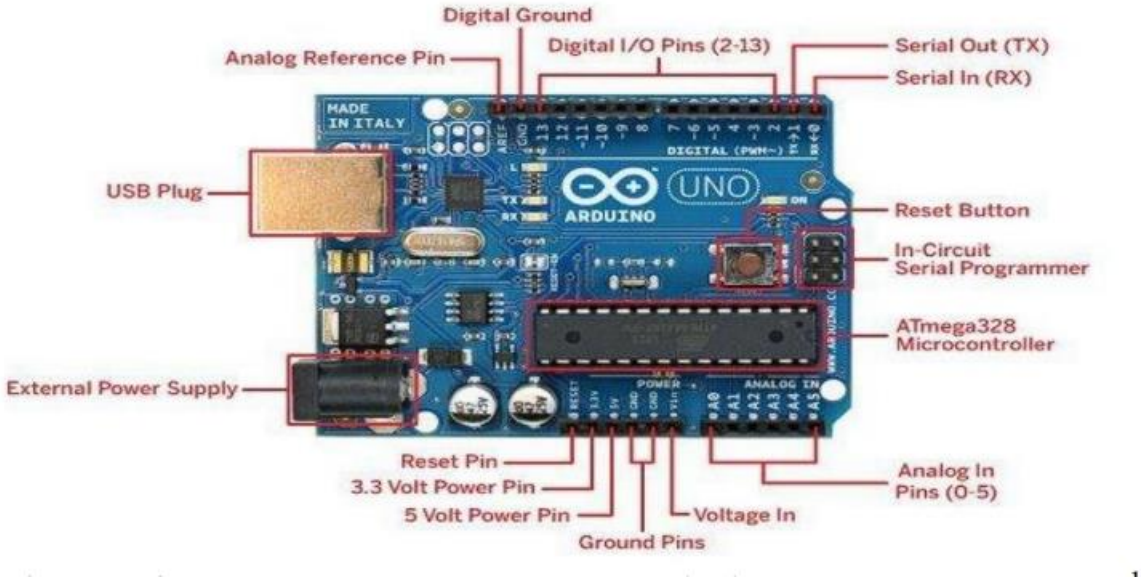
Abstract:

The objective of this experiment is to detect and measure the distance from the obstacle to the object and automatically control the speed based on the distance. When an obstacle is detected within 10 cm, a LED will turn on and the motor will rotate.

Theory and Methodology:

Aurdino is a open-source platform for creating interactive electronic project . It consists both programable microcontroller and a piece of software that runs on your computer.

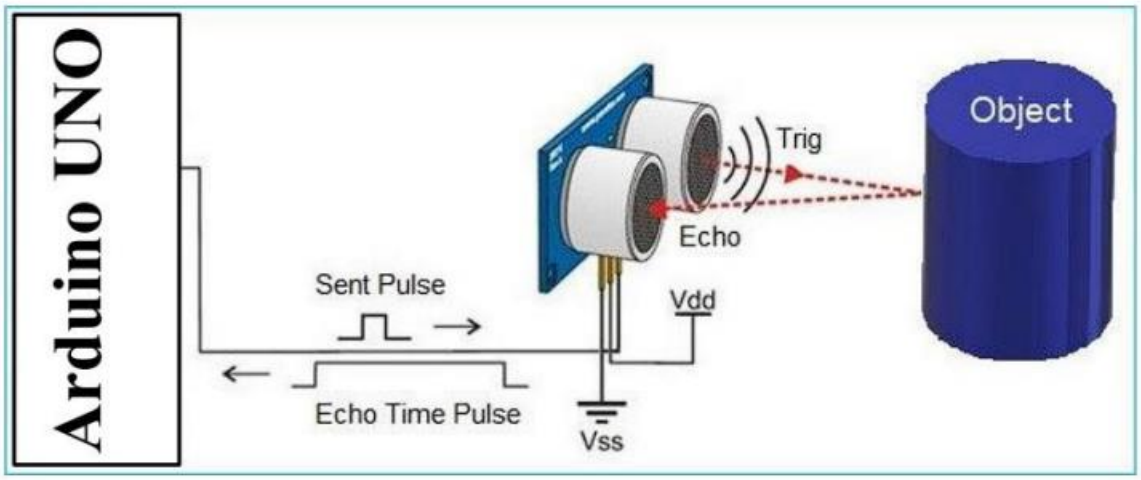
Overview of the Board (Arduino UNO R3)



HCS R04 is an ultrasonic ranging module that consists of a transmitter , receiver and control circuit.It has four pins for VCC,GND,Trigger and Echo.The module automatically sends 40KHz and detects whether there is a pulse signal back. As the pin generate from the Arduino board travel out from the trigger and comes back to the echo.The objective we take half of the distance covered by the trigger will be calculated by the following equation: distance , cm=microseconds /29/2

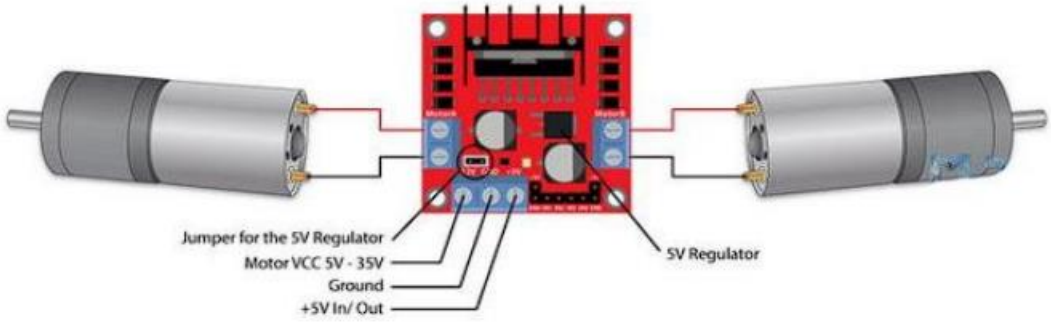
Distance, in=microseconds/74/2

Overview of Sonar Sensor:

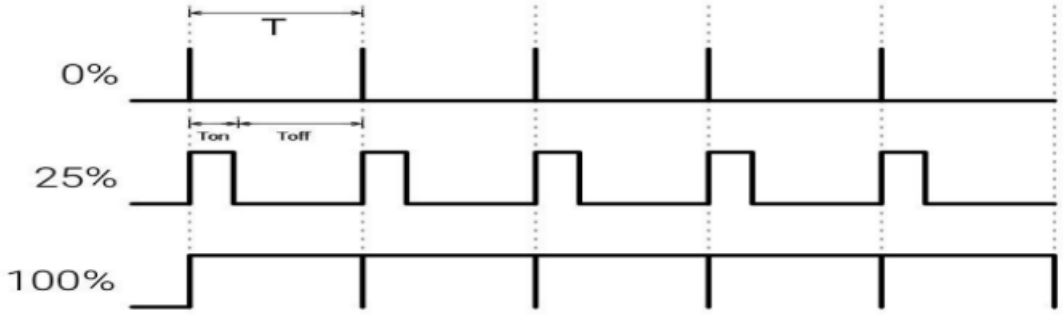


Motor speed varies according to duty cycle.Duty cycle is function of one time period.the duty cycle is commonly expressed as a percentage or ratio. A period is a time it takes for a signal to complete on and off cycle.

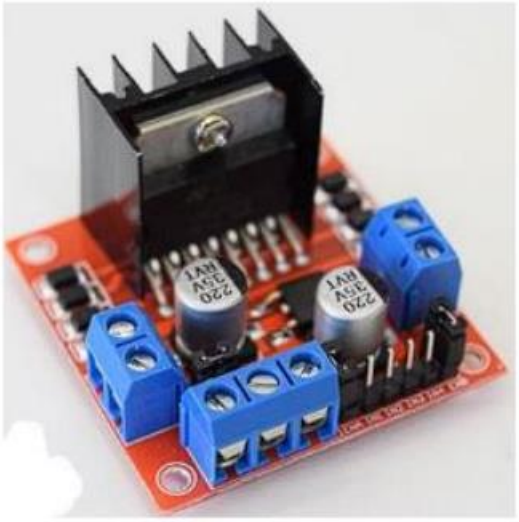
Duty cycle=(Ton/T) X 100%



Suppose the duty is zero and the motor does not run.and when the duty cycle is 100% the motor moves on maximum RPM . the motor running after giving some fixed voltage that is called threshold voltage.



The L298 driver is a dual H-bridge motor driver which allows speed abd direction control of 2 DC motors at the same time.the module has two terminal .The VCC for motor and a 5V pin which can wither be an input output . Nexts are the logic unit. The input 1 and the input 2 pins are used for controlling rotaion direction of the motor.



Apparatus:

01.Arduino Mega

02.LED Indicator

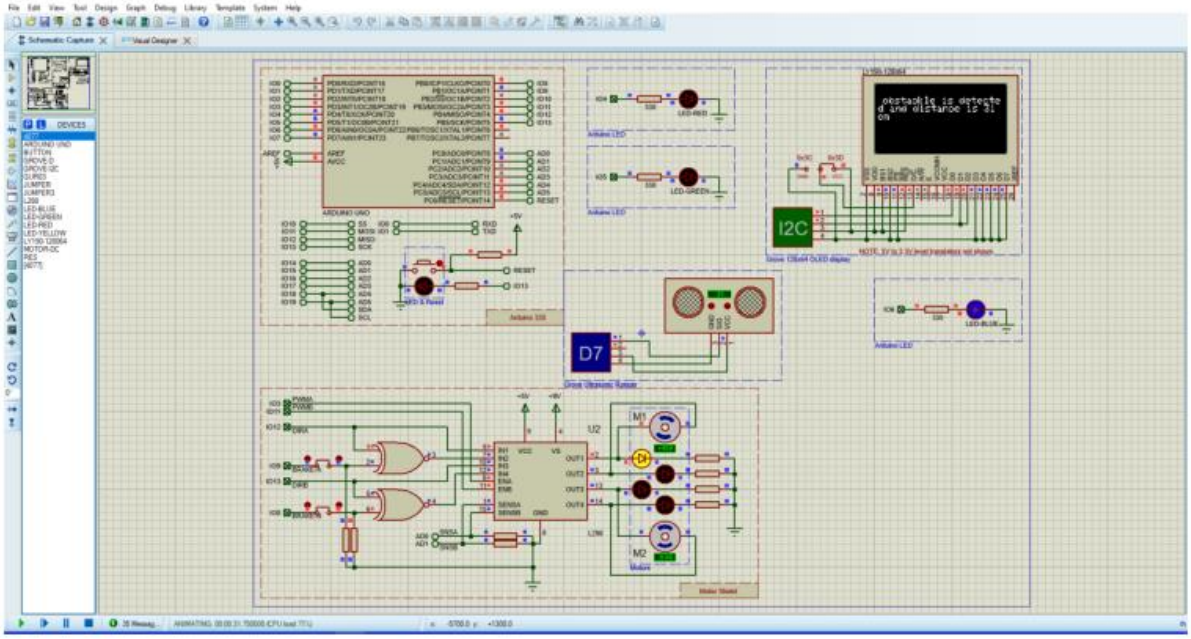
03.Ultrasonic Sensor

04.DC Motors

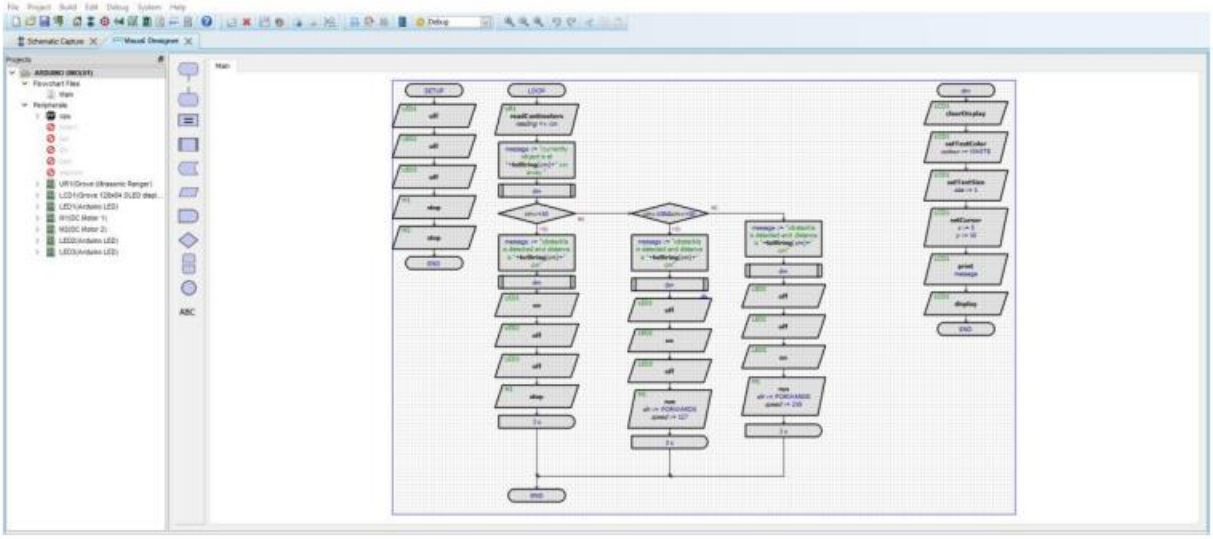
05.Breadboard

06.Jumper wires

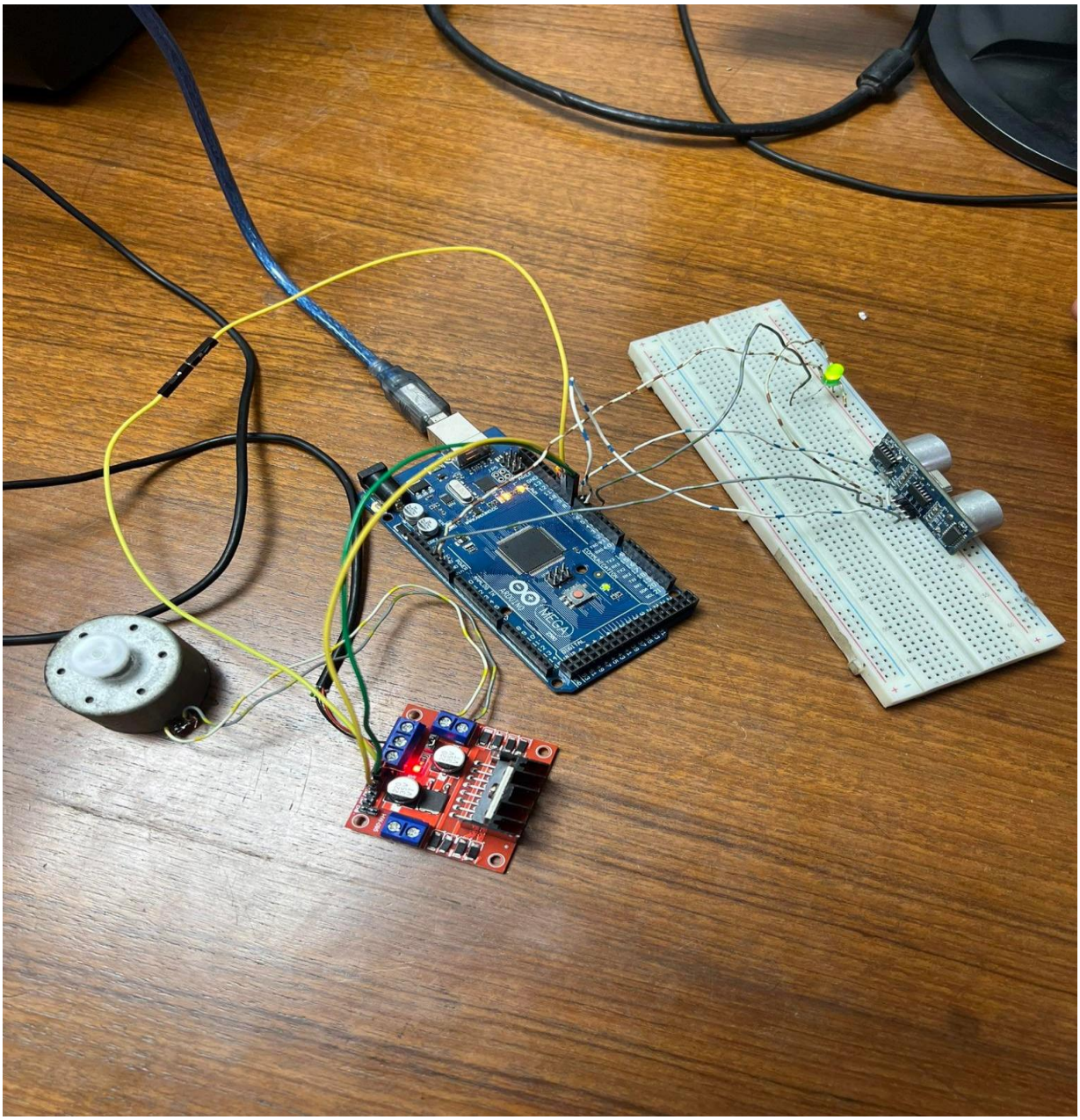
Simulation Setup and Schematic Diagram:

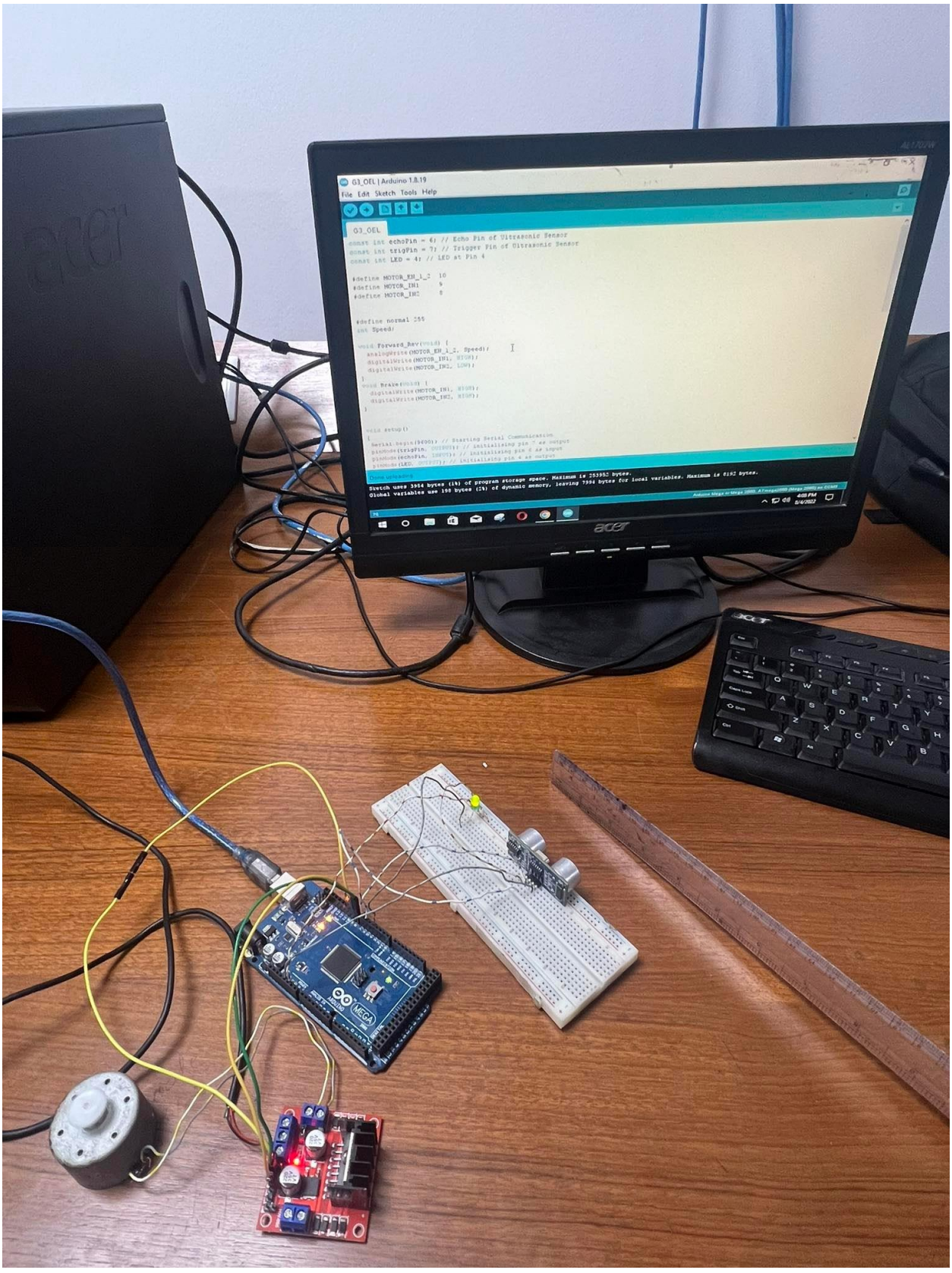


Flowchart:



Hardware Set-Up:




G3_OEL | Arduino 1.8.19
File Edit Sketch Tools Help
G3_OEL
const int echoPin = 6; // Echo Pin of Ultrasonic Sensor
const int trigPin = 7; // Trigger Pin of Ultrasonic Sensor
const int LED = 4; // LED at Pin 4
#define MOTOR_EN_1_2 10
#define MOTOR_IN1 9
#define MOTOR_IN2 8
#define normal 255
int Speed;
void Forward_Rev(void) {
analogWrite(MOTOR_EN_1_2, Speed);
digitalWrite(MOTOR_IN1, HIGH);
digitalWrite(MOTOR_IN2, LOW);
}
void Brake(void) {
digitalWrite(MOTOR_IN1, HIGH);
digitalWrite(MOTOR_IN2, HIGH);
}
void setup()
{
Serial.begin(9600); // Starting Serial Communication
pinMode(trigPin, OUTPUT); // initialising pin 7 as output
pinMode(trigPin, INPUT); // initialising pin 6 as input
pinMode(echoPin, INPUT); // initialising pin 4 as output
pinMode(LED, OUTPUT); // initialising pin 4 as output
Done uploading.
Sketch uses 3954 bytes (1%) of program storage space. Maximum is 253952 bytes.
Global variables use 198 bytes (2%) of dynamic memory, leaving 7994 bytes for local variables. Maximum is 8192 bytes.
4:28 PM
5/4/2022
acer
MEGA

Data Collection table

Distance	Motor Status
10	**Off**
5	**On(high speed)**
7	**On(high speed)**
2	**On(low speed)**
15	**off**

Program/Lab Code:

```
const int echoPin = 6; // Echo Pin of Ultrasonic Sensor
const int trigPin = 7; // Trigger Pin of Ultrasonic Sensor
const int LED = 4; // LED at Pin 4

#define MOTOR_EN_1_2  10
#define MOTOR_IN1     9
#define MOTOR_IN2     8



#define normal 255
int Speed;

void Forward_Rev(void) {
  analogWrite(MOTOR_EN_1_2, Speed);
  digitalWrite(MOTOR_IN1, HIGH);
  digitalWrite(MOTOR_IN2, LOW);
}
void Brake(void) {
  digitalWrite(MOTOR_IN1, HIGH);
  digitalWrite(MOTOR_IN2, HIGH);
```

```
}


void setup()
{
 Serial.begin(9600); // Starting Serial Communication
 pinMode(trigPin, OUTPUT); // initialising pin 7 as output
 pinMode(echoPin, INPUT); // initialising pin 6 as input
 pinMode(LED, OUTPUT); // initialising pin 4 as output

 pinMode(MOTOR_EN_1_2, OUTPUT);
 pinMode(MOTOR_IN1, OUTPUT);
 pinMode(MOTOR_IN2, OUTPUT);
}
void loop()
{
 Speed = normal; // Normal Speed
 long duration, inches, cm;

 digitalWrite(trigPin, LOW);
 delayMicroseconds(2);
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);

 digitalWrite(trigPin, LOW);

 duration = pulseIn(echoPin, HIGH); // using pulsin function to determine total time
 inches = microsecondsToInches(duration); // calling method
 cm = microsecondsToCentimeters(duration); // calling method
 if(cm<10)
```

```
 { Serial.print(inches);
 Serial.print("in, ");
 Serial.print(cm);
 Serial.print("cm");
 Serial.println();
 digitalWrite(LED, HIGH);
 Forward_Rev();
 //delay(500);
 }
 else
 {
  Brake();
  //delay(500);
  digitalWrite(LED, LOW);
 }
 delay(100);
}
long microsecondsToInches(long microseconds) // method to covert microsec to inches
{
return microseconds / 74 / 2;
}
long microsecondsToCentimeters(long microseconds) // method to covert microsec to cm
{
 return microseconds / 29 / 2;
}
```

Discussion:

We learned how to automatically adjust the motor speed based on the obstacle distance as determined by the ultrasonic sensor in this OEL project. We also studied the L298N driver's role in the motor system. We wired an ultrasonic sensor and a motor to the Arduino board for the experiment. The necessary program is then written into the Arduino Mega. Some data were logged into the data collection database during the trial.

Conclusion:

This experiment sought to develop an automatic speed control system based on obstacle distance. We could observe from the data gathering table that the motor's speed varied according to the distance of the obstacles. The motor turned at the proper speed when the distance was short, but it was off when the distance was long. Therefore, it can be determined that the experiment's goal was accomplished.