Ahsanullah University of Science and Technology

Department of Computer Science and Enineering

CSE3216: Microcontroller Based System Design Lab

Spring 2020

Final Report

Traffic & Railway Control System

Lab Section: B1

Group: G3

Submitted by

Jarin Sultana	170204068
Rafsan Habib	170204069
Alam Khan	170204084
Nafisa Hossain	170204112

Submitted To

Ms. Afsana Ahmed Munia
Assistant Professor
CSE, AUST

Nibir Chandra Mandal
Lecturer
CSE, AUST

Date of Submission: 5 March, 2021

Contents

1 Objective

The main objective of our project is to design a traffic control system for a smart city where the timing of signals will change automatically.

Traffic congestion is a severe problem in Dhaka. So it is time to shift from manual mode to an automated system with decision-making capabilities.

Present-day traffic signaling systems are fixed time based which may render inefficient if one lane is operational than the others. To optimize this problem we have made a framework for an intelligent traffic control system. Sometimes higher traffic presence at one side of the junction demands longer green time as compared to the standard allotted time. We, therefore, propose here a mechanism in which the time period of green light and red light is assigned on the basis of the presence of the traffic present at that time. This can be achieved by using an ultrasonic sensor. The sensors which are present on the sides of the road will detect the presence of the vehicles and send the information to the microcontroller(Arduino) where it will decide how long a flank will be open or when to change over the signal lights.

The railway is one of the most important mediums of transportation. But in this modern age of technology, we still rely on manpower to regulate rail control. We should convert the regulation of rail from manpower to digitized control as manpower has limited capacity and prone to commit an error. We are going to use a system that uses sensors to detect trains and trigger the barrier using automation to restrict the movement of vehicles.

2 Social Values

Traffic jams are like a burden for us. It consumes our precious time, which is not exchangeable with others. We pass our time as lazy during the traffic jam condition. Students may get more education or pleasure, the scientist may get more time for research, doctors may get more chances to give patients patience, and ordinary people may get more time to realize their lives. Increasing traffic congestion does impose costs not only on travelers but also on the whole economic activities and finally affects national income. It has been difficult to develop and apply empirical measures of the extent of those economic costs. Traffic jams also cause stress for both the drivers and passengers which might lead to unwelcome events such as accidents which might lead to the early death of any driver/passenger/passerby.

Moreover, Railway accidents are also a major issue in Bangladesh nowadays. Every year we lose a lot of lives due to railway accidents. Human errors are accountable for 72% of rail accidents, according to Bangladesh Railway(19 December 2020: Dhaka Tribune). By using our project we can minimize this rate of death.

3 Required Components

The following parts and tools are required for building this project:

- Arduino Mega 2560: It is required for interact with surrounding and controlling the electronics.
- 10 X HC-SR04 Ultrasonic Sensors: It is required for measuring obstacle distance from the component.
- 6 X Red LEDs: It is used to stop the vehicle
- 6 X Green LEDs: It is used to slow down the vehicle
- 6 X Yellow LEDs: It is used to give the vehicles green signal.
- LCD: It is used for showing which signal or train line is on.

- Buzzer: It is used to indicate that train is coming.
- 20 X 220 Ohm Resistors: It is used to control the current flow.
- 12-volt Battery: It is required for giving power to whole system.
- Breadboards: It is required for creating electrical connections between the components.
- Jumper Cables: It is required for connected the components.

4 Design

The circuit diagram is given below-

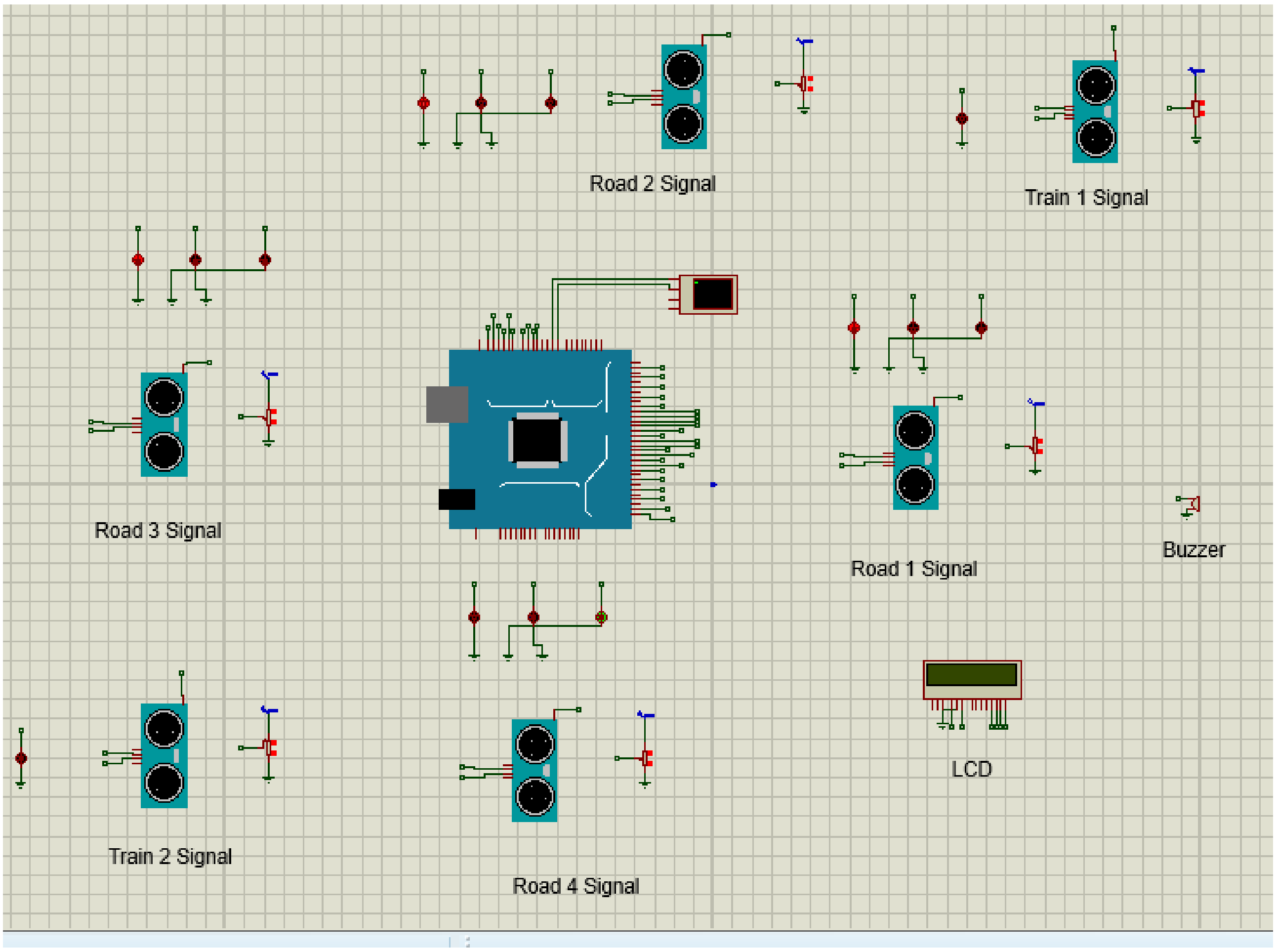


Figure 1: Setup of Traffic & Railway Control System

5 Working Procedure

In Road System,

- If there is traffic at all the signals, then the system will work normally by controlling the signals one by one.
- If there is no traffic near a signal, then the system will skip this signal and will move on to the next one. For example, if there is no vehicle at signal 2, 3 and currently the system is allowing vehicles at signal 1 to pass. Then after signal 1, the system will move on to signal 4 skipping signal 2 and 3.
- If there is no traffic at all the 4 signals, then also the system will work normally by controlling the signals one by one.

In Railway System,

- If there is a train coming from any side, then the system will restrict all the roads.
- If there is no train, then the roads will run according to the previously described road system.

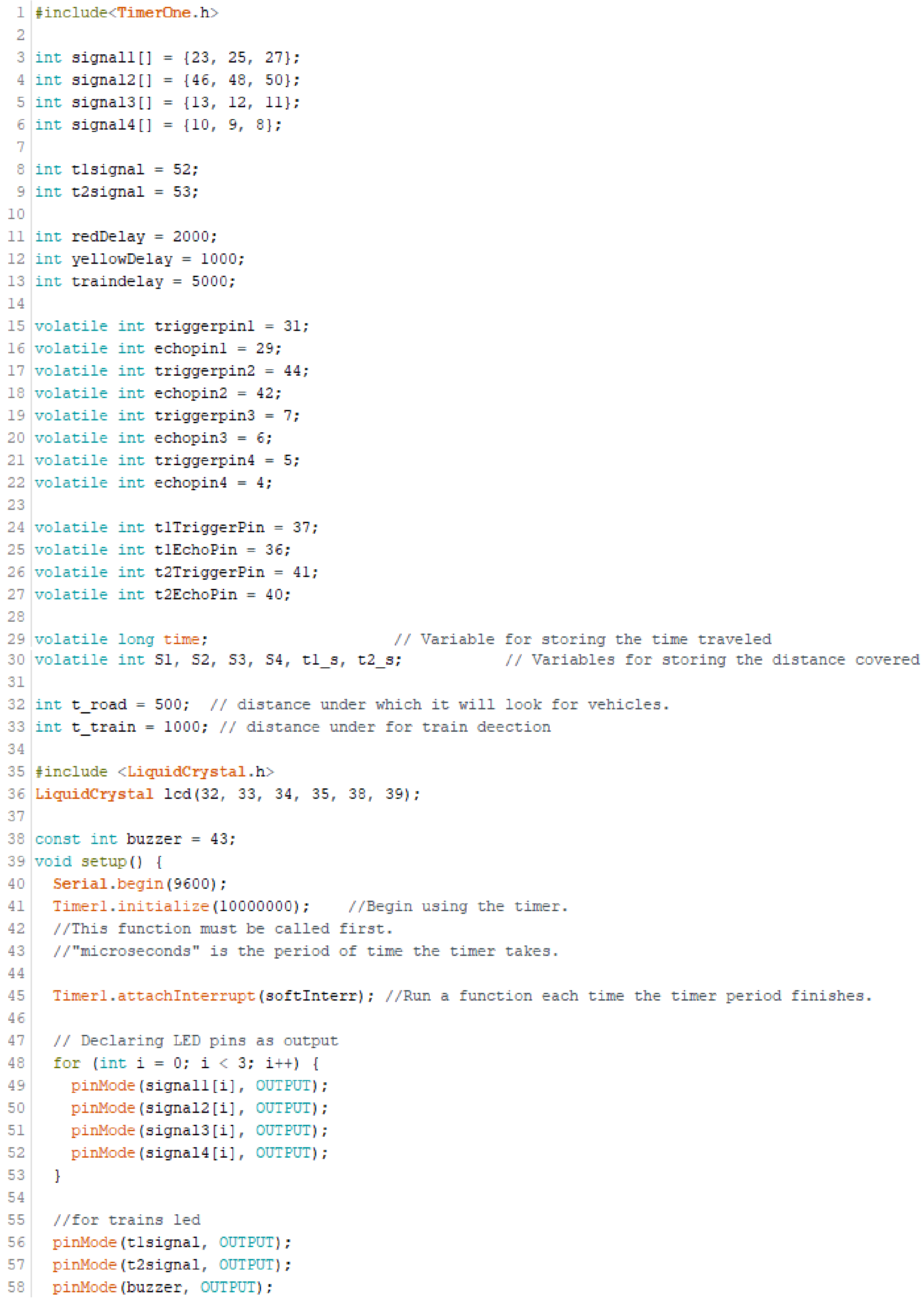
Users can see all blocked and free roads by following red, yellow, and green LEDs.

6 Budget

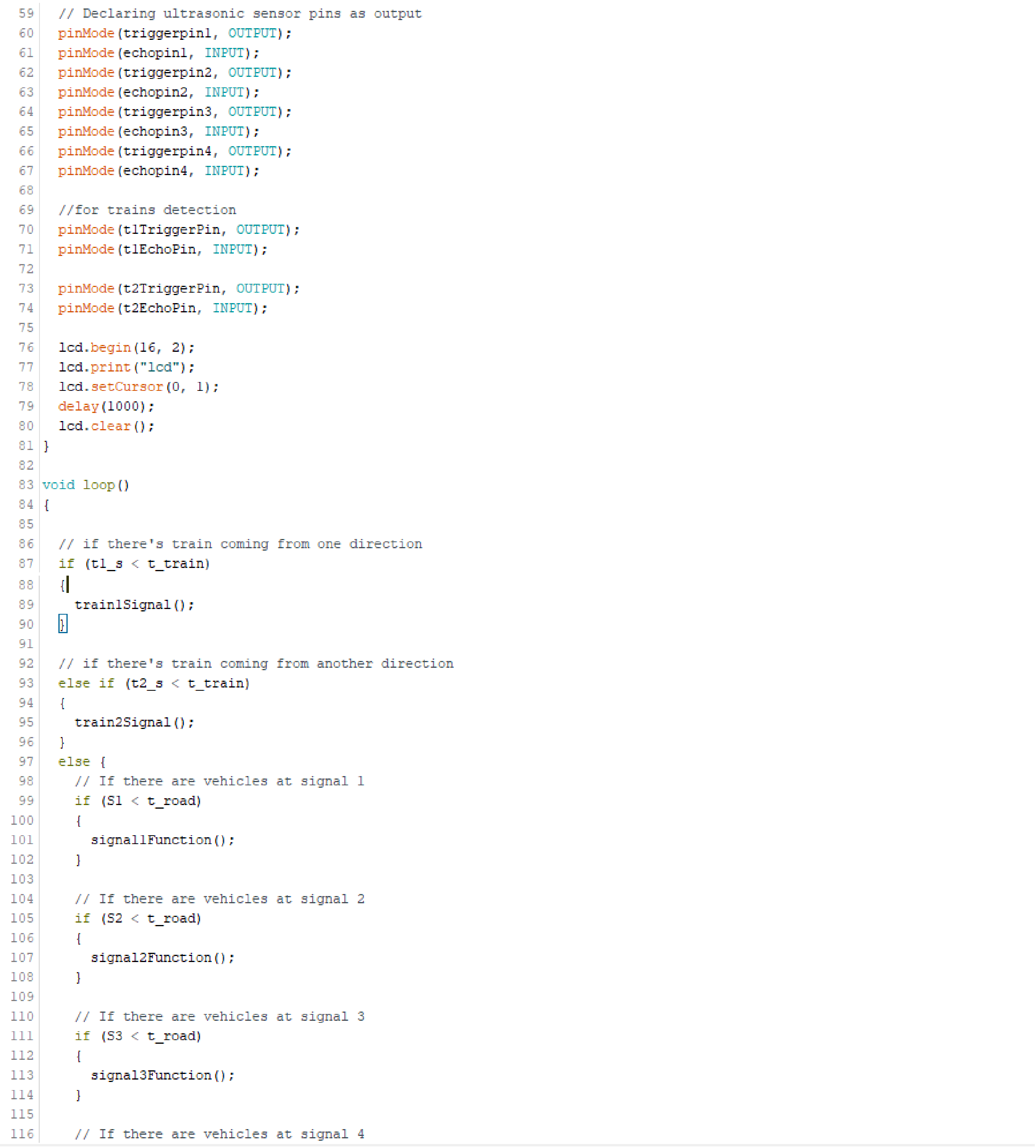
Equipment	**Quantity**	**Budget(Tk)**
Arduino Mega	1	920
HC-SR04 Ultrasonic Sensors	10	1200
LEDs	10	10
Resistors	20	50
Jumper Cables	As Required	100
Breadboards	3	120
12-volt Battery	2	80
Cardboard	1	80
Cutter	1	50
LCD	1	50
Buzzer	1	50
Toy Car	2	200
Total		**2,810**

This is still the estimated budget. As the real project could not be completed, the real budget remains unknown but most probably will not vary by much.

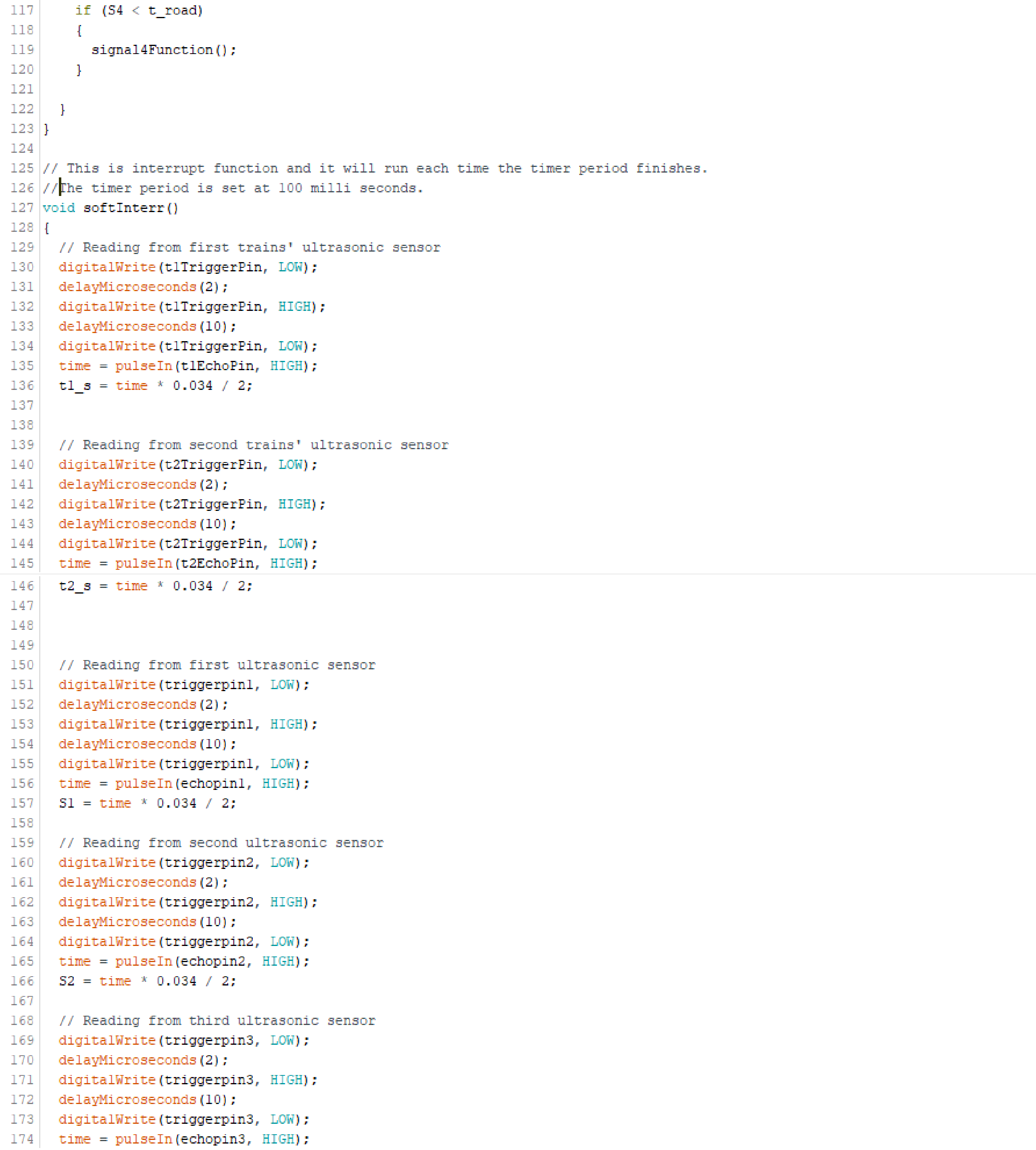
7 Code



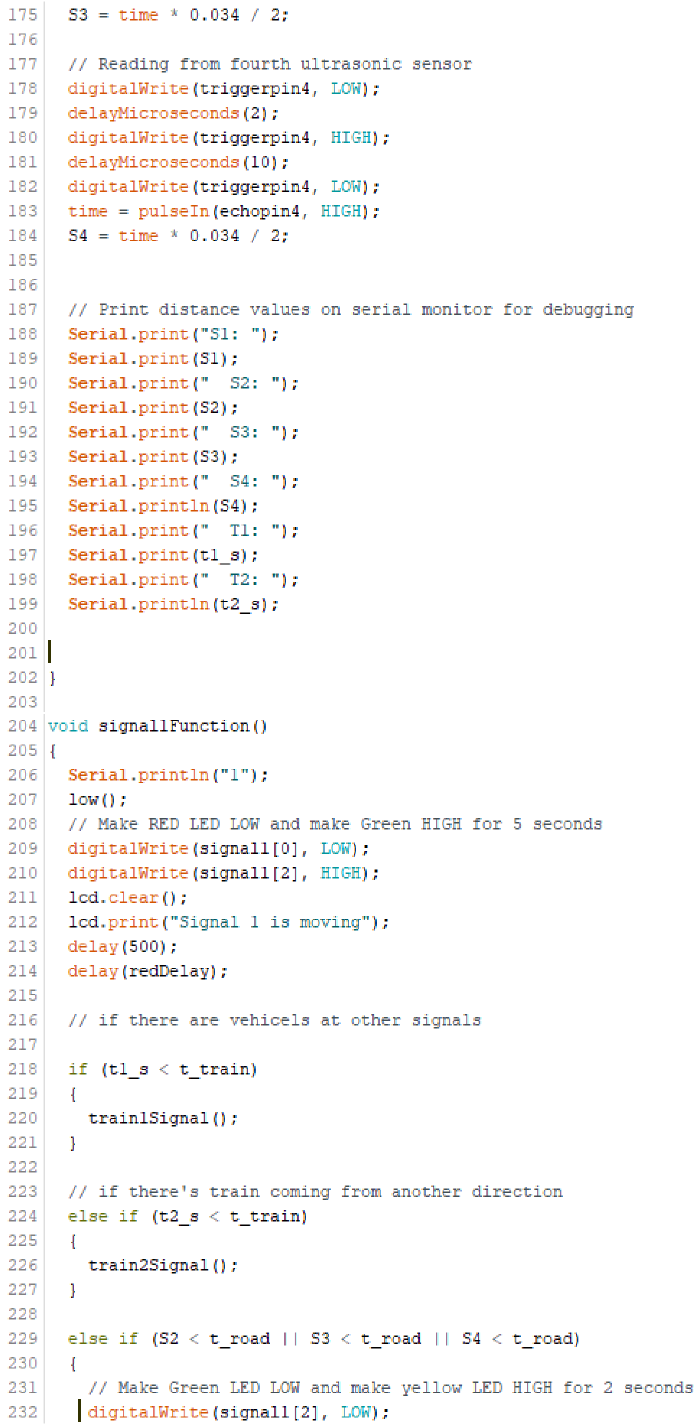
```
#include<TimerOne.h>

int signal1[] = {23, 25, 27};
int signal2[] = {46, 48, 50};
int signal3[] = {13, 12, 11};
int signal4[] = {10, 9, 8};

int t1signal = 52;
int t2signal = 53;

int redDelay = 2000;
int yellowDelay = 1000;
int traindelay = 5000;

volatile int triggerpin1 = 31;
volatile int echopin1 = 29;
volatile int triggerpin2 = 44;
volatile int echopin2 = 42;
volatile int triggerpin3 = 7;
volatile int echopin3 = 6;
volatile int triggerpin4 = 5;
volatile int echopin4 = 4;

volatile int t1TriggerPin = 37;
volatile int t1EchoPin = 36;
volatile int t2TriggerPin = 41;
volatile int t2EchoPin = 40;

volatile long time;                     // Variable for storing the time traveled
volatile int S1, S2, S3, S4, t1_s, t2_s;           // Variables for storing the distance covered

int t_road = 500;  // distance under which it will look for vehicles.
int t_train = 1000; // distance under for train deection

#include <LiquidCrystal.h>
LiquidCrystal lcd(32, 33, 34, 35, 38, 39);

const int buzzer = 43;
void setup() {
  Serial.begin(9600);
  Timer1.initialize(10000000);    //Begin using the timer.
  //This function must be called first.
  //"microseconds" is the period of time the timer takes.

  Timer1.attachInterrupt(softInterr); //Run a function each time the timer period finishes.

  // Declaring LED pins as output
  for (int i = 0; i < 3; i++) {
    pinMode(signal1[i], OUTPUT);
    pinMode(signal2[i], OUTPUT);
    pinMode(signal3[i], OUTPUT);
    pinMode(signal4[i], OUTPUT);
  }

  //for trains led
  pinMode(t1signal, OUTPUT);
  pinMode(t2signal, OUTPUT);
  pinMode(buzzer, OUTPUT);
```



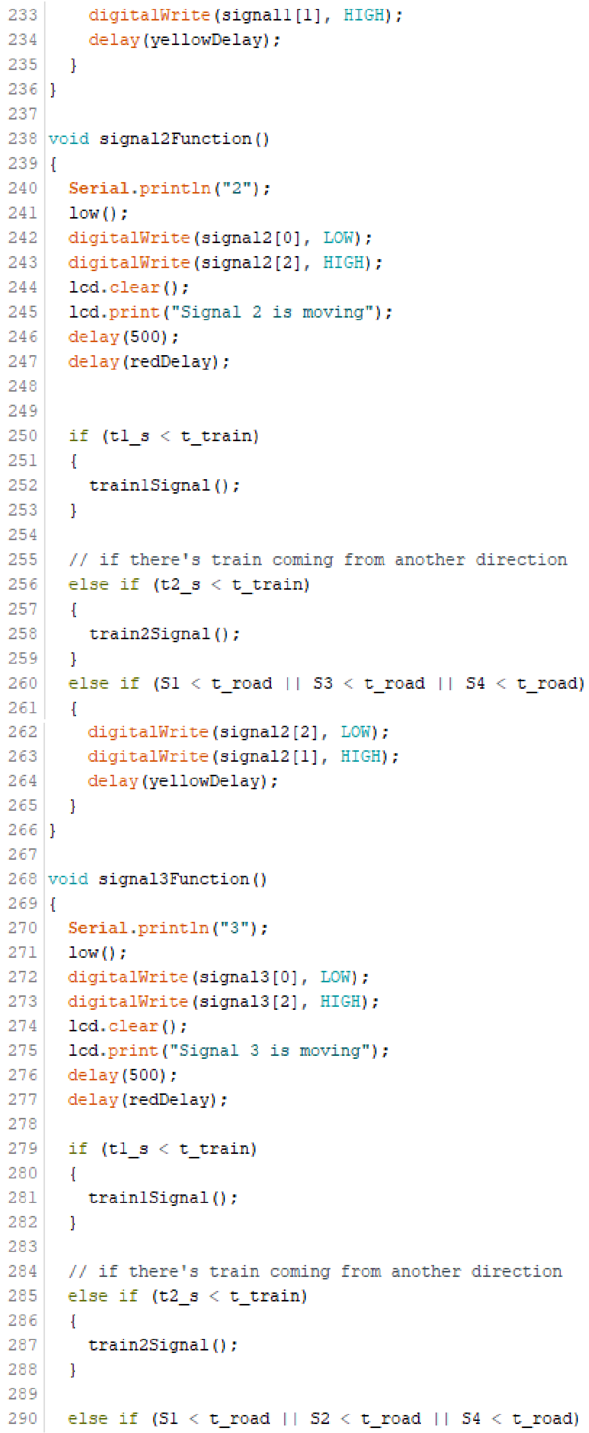
```
  // Declaring ultrasonic sensor pins as output
  pinMode(triggerpin1, OUTPUT);
  pinMode(echopin1, INPUT);
  pinMode(triggerpin2, OUTPUT);
  pinMode(echopin2, INPUT);
  pinMode(triggerpin3, OUTPUT);
  pinMode(echopin3, INPUT);
  pinMode(triggerpin4, OUTPUT);
  pinMode(echopin4, INPUT);

  //for trains detection
  pinMode(t1TriggerPin, OUTPUT);
  pinMode(t1EchoPin, INPUT);

  pinMode(t2TriggerPin, OUTPUT);
  pinMode(t2EchoPin, INPUT);

  lcd.begin(16, 2);
  lcd.print("lcd");
  lcd.setCursor(0, 1);
  delay(1000);
  lcd.clear();
}

void loop()
{

  // if there's train coming from one direction
  if (t1_s < t_train)
  {
    train1Signal();
  }

  // if there's train coming from another direction
  else if (t2_s < t_train)
  {
    train2Signal();
  }
  else {
    // If there are vehicles at signal 1
    if (S1 < t_road)
    {
      signal1Function();
    }

    // If there are vehicles at signal 2
    if (S2 < t_road)
    {
      signal2Function();
    }

    // If there are vehicles at signal 3
    if (S3 < t_road)
    {
      signal3Function();
    }

    // If there are vehicles at signal 4
```



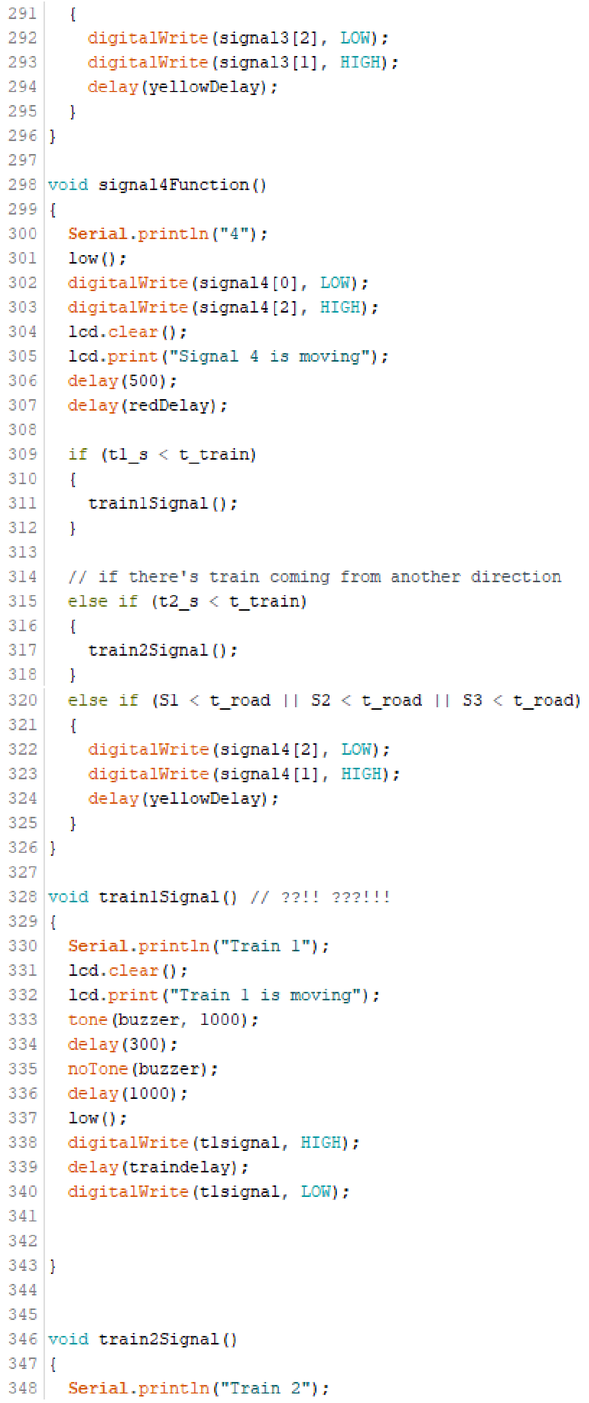
```
    if (S4 < t_road)
    {
      signal4Function();
    }

  }
}

// This is interrupt function and it will run each time the timer period finishes.
//The timer period is set at 100 milli seconds.
void softInterr()
{
  // Reading from first trains' ultrasonic sensor
  digitalWrite(t1TriggerPin, LOW);
  delayMicroseconds(2);
  digitalWrite(t1TriggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(t1TriggerPin, LOW);
  time = pulseIn(t1EchoPin, HIGH);
  t1_s = time * 0.034 / 2;


  // Reading from second trains' ultrasonic sensor
  digitalWrite(t2TriggerPin, LOW);
  delayMicroseconds(2);
  digitalWrite(t2TriggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(t2TriggerPin, LOW);
  time = pulseIn(t2EchoPin, HIGH);
  t2_s = time * 0.034 / 2;



  // Reading from first ultrasonic sensor
  digitalWrite(triggerpin1, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerpin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerpin1, LOW);
  time = pulseIn(echopin1, HIGH);
  S1 = time * 0.034 / 2;

  // Reading from second ultrasonic sensor
  digitalWrite(triggerpin2, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerpin2, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerpin2, LOW);
  time = pulseIn(echopin2, HIGH);
  S2 = time * 0.034 / 2;

  // Reading from third ultrasonic sensor
  digitalWrite(triggerpin3, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerpin3, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerpin3, LOW);
  time = pulseIn(echopin3, HIGH);
```



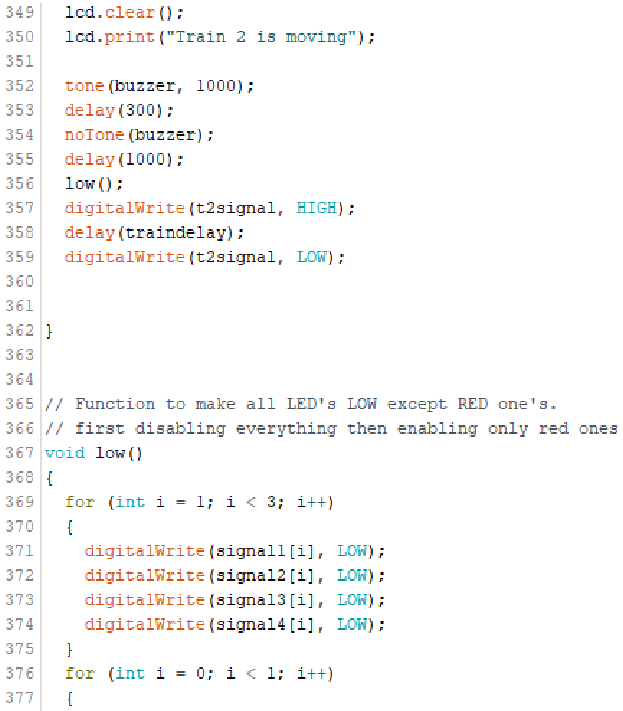
```
  S3 = time * 0.034 / 2;

  // Reading from fourth ultrasonic sensor
  digitalWrite(triggerpin4, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerpin4, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerpin4, LOW);
  time = pulseIn(echopin4, HIGH);
  S4 = time * 0.034 / 2;


  // Print distance values on serial monitor for debugging
  Serial.print("S1: ");
  Serial.print(S1);
  Serial.print("  S2: ");
  Serial.print(S2);
  Serial.print("  S3: ");
  Serial.print(S3);
  Serial.print("  S4: ");
  Serial.println(S4);
  Serial.print("  T1: ");
  Serial.print(t1_s);
  Serial.print("  T2: ");
  Serial.println(t2_s);


}

void signal1Function()
{
  Serial.println("1");
  low();
  // Make RED LED LOW and make Green HIGH for 5 seconds
  digitalWrite(signal1[0], LOW);
  digitalWrite(signal1[2], HIGH);
  lcd.clear();
  lcd.print("Signal 1 is moving");
  delay(500);
  delay(redDelay);

  // if there are vehicels at other signals

  if (t1_s < t_train)
  {
    train1Signal();
  }

  // if there's train coming from another direction
  else if (t2_s < t_train)
  {
    train2Signal();
  }

  else if (S2 < t_road || S3 < t_road || S4 < t_road)
  {
    // Make Green LED LOW and make yellow LED HIGH for 2 seconds
    digitalWrite(signal1[2], LOW);
```



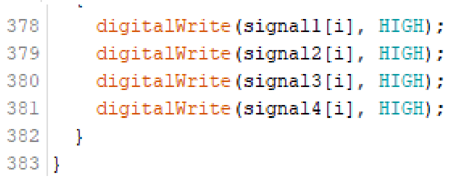
```
    digitalWrite(signal1[1], HIGH);
    delay(yellowDelay);
  }
}

void signal2Function()
{
  Serial.println("2");
  low();
  digitalWrite(signal2[0], LOW);
  digitalWrite(signal2[2], HIGH);
  lcd.clear();
  lcd.print("Signal 2 is moving");
  delay(500);
  delay(redDelay);


  if (t1_s < t_train)
  {
    train1Signal();
  }

  // if there's train coming from another direction
  else if (t2_s < t_train)
  {
    train2Signal();
  }
  else if (S1 < t_road || S3 < t_road || S4 < t_road)
  {
    digitalWrite(signal2[2], LOW);
    digitalWrite(signal2[1], HIGH);
    delay(yellowDelay);
  }
}

void signal3Function()
{
  Serial.println("3");
  low();
  digitalWrite(signal3[0], LOW);
  digitalWrite(signal3[2], HIGH);
  lcd.clear();
  lcd.print("Signal 3 is moving");
  delay(500);
  delay(redDelay);

  if (t1_s < t_train)
  {
    train1Signal();
  }

  // if there's train coming from another direction
  else if (t2_s < t_train)
  {
    train2Signal();
  }

  else if (S1 < t_road || S2 < t_road || S4 < t_road)
```



```
  {
    digitalWrite(signal3[2], LOW);
    digitalWrite(signal3[1], HIGH);
    delay(yellowDelay);
  }
}

void signal4Function()
{
  Serial.println("4");
  low();
  digitalWrite(signal4[0], LOW);
  digitalWrite(signal4[2], HIGH);
  lcd.clear();
  lcd.print("Signal 4 is moving");
  delay(500);
  delay(redDelay);

  if (t1_s < t_train)
  {
    train1Signal();
  }

  // if there's train coming from another direction
  else if (t2_s < t_train)
  {
    train2Signal();
  }
  else if (S1 < t_road || S2 < t_road || S3 < t_road)
  {
    digitalWrite(signal4[2], LOW);
    digitalWrite(signal4[1], HIGH);
    delay(yellowDelay);
  }
}

void train1Signal() // ??!! ???!!!
{
  Serial.println("Train 1");
  lcd.clear();
  lcd.print("Train 1 is moving");
  tone(buzzer, 1000);
  delay(300);
  noTone(buzzer);
  delay(1000);
  low();
  digitalWrite(t1signal, HIGH);
  delay(traindelay);
  digitalWrite(t1signal, LOW);


}


void train2Signal()
{
  Serial.println("Train 2");
```



```
  lcd.clear();
  lcd.print("Train 2 is moving");

  tone(buzzer, 1000);
  delay(300);
  noTone(buzzer);
  delay(1000);
  low();
  digitalWrite(t2signal, HIGH);
  delay(traindelay);
  digitalWrite(t2signal, LOW);


}


// Function to make all LED's LOW except RED one's.
// first disabling everything then enabling only red ones
void low()
{
  for (int i = 1; i < 3; i++)
  {
    digitalWrite(signal1[i], LOW);
    digitalWrite(signal2[i], LOW);
    digitalWrite(signal3[i], LOW);
    digitalWrite(signal4[i], LOW);
  }
  for (int i = 0; i < 1; i++)
  {
```



```
    digitalWrite(signal1[i], HIGH);
    digitalWrite(signal2[i], HIGH);
    digitalWrite(signal3[i], HIGH);
    digitalWrite(signal4[i], HIGH);
  }
}
```

8 Members Contribution

8.1 Contribution of Jarin Sultana (ID: 170204068)

All the code and design of road signal 1 is done by this ID. In this part, the control of the LEDs and the values calculated from Ultrasonic Sensors are handled here accordingly.The Ultrasonic Sensors, LEDs used for this signal purpose is designed by this ID.

8.2 Contribution of Rafsan Habib (ID: 170204069)

All the code and design of train signals and road signal 4 is done by this ID. In this part, the control of the LEDs and the values calculated from Ultrasonic Sensors are handled here accordingly.The Ultrasonic Sensors, LEDs used for this signal purpose is designed by this ID.

8.3 Contribution of Alam Khan (ID: 170204084)

All the code and design of road signal 2 is done by this ID. In this part, the control of the LEDs and the values calculated from Ultrasonic Sensors are handled here accordingly.The Ultrasonic Sensors, LEDs used for this signal purpose is designed by this ID.

8.4 Contribution of Nafisa Hossain (ID: 170204112)

All the code and design of road signal 3 is done by this ID. In this part, the control of the LEDs and the values calculated from Ultrasonic Sensors are handled here accordingly.The Ultrasonic Sensors, LEDs used for this signal purpose is designed by this ID.

9 Difficulties

- Searching for libraries of different components to be used by Proteus
- Making simulation as close to real world scenario as possible to ensure that the simulation may come handy when implementing in real life.

10 Future Work

We will update this system when a vehicle tries to move even during a red signal, it will turn on an alarm to warn the driver of the vehicle and will send the alert to the traffic warden with the picture.

11 Conclusion

Though the prototype model worked very efficiently with remarkable outputs, the real-life situation is going to be way more challenging and demanding. We hope that our system will bring about some changes in the current traffic management system and make our life easier than before.