SMART TRAFFIC MANAGEMENT SYSTEM

A PROJECT REPORT

for

INTERNET OF THINGS (ITA3009)

in

B.C.A

by

MOHAMMED ZYAAN C(22BCA0269)

Under the Guidance of

BALAJI E

Assistant Professor (Senior Grade), SCORE

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science Engineering and Information Systems

TABLE OF CONTENTS

CONCLUSION

ABSTRACT

Traffic management is one of the biggest infrastructure hurdles faced by developing countries today. Developed countries and smart cities are already using IoT and to their advantage to minimize issues related to traffic. The culture of the car has been cultivated speedily among people in all types of nations. In most cities, it is common for people to prefer riding their own vehicles no matter how good or bad the public transportation is or considering how much time and money is it going to take for them to reach their destination. This research aims to monitor the traffic condition and to control the traffic light.This system can reduce not only traffic congestion but also waiting time. This system is designed to be implemented in places nearing the junctions. Thisresearch is based on the effective use of Internet of Things (IoT). This system will display the traffic conditions in the website and the traffic light can be controlled from this website. This system has been implemented by using esp8266,ultrasonic sensor and arduino. Esp8266 nodemcu which is IoT device is used to transmit the traffic information to the website which is connected with this device. Ultrasonic sensors are placed on each road to sense the presence and absence of vehicles. Traffic information is received from these sensors. Traffic light prototype is built by using an arduino UNO. This traffic light can be controlled from the website. The system will display the traffic states in the website that can guide the drivers to select the right way and avoid traffic congestions.

INTRODUCTION:

In a world with growing population and increasing transportation needs, the cities need an intelligent traffic management system (ITS) which works based on traffic concentration on different lanes. This system will be created to reduce waiting time at lanes and let the citizens travel to their homes, offices, schools and other destinations more quickly. The problem of unnecessarily waiting at signal, while other roads are empty is the cause of delay for people to reach their destinations. This problem is addressed here. The citizens will not be stuck in traffic for too long. There is no live traffic monitoring developed for the citizens to be able to view the traffic status from anywhere. This system aims to provide the users with facility (maybe a website or an app) to view the actual traffic density and traffic status, like is there is no traffic, light traffic or very high traffic. This will help the people to decide on their path from whichever location they are current at. This will help them to avoid traffic and or pass through light traffic in order to get to their destinations. An emergency service has a very crucial role in people's life. In India, people always complain about the not getting ambulance, police, fire brigades and other lifesaving services on time. This system addresses this issue and gives the traffic control center people and authorities with a facility to manually control the traffic so that the citizens can get easy and fast access to the emergency services be it medical necessity, crime prevention measure or rescue services.

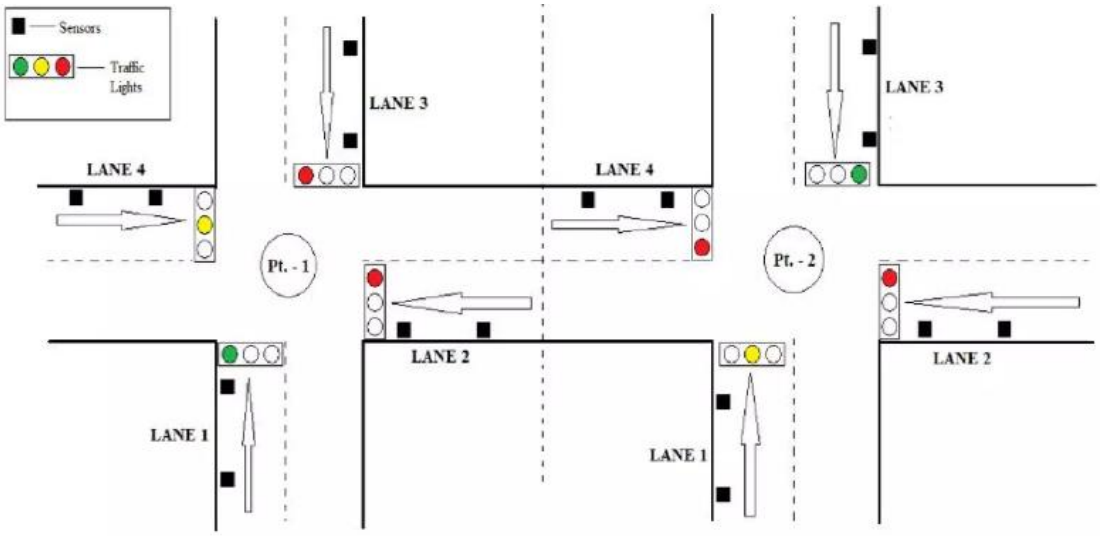
SCOPE OF THE PROJECT:

smart traffic management system utilizing sensor data, communication and auto- mated algorithms is to be developed to

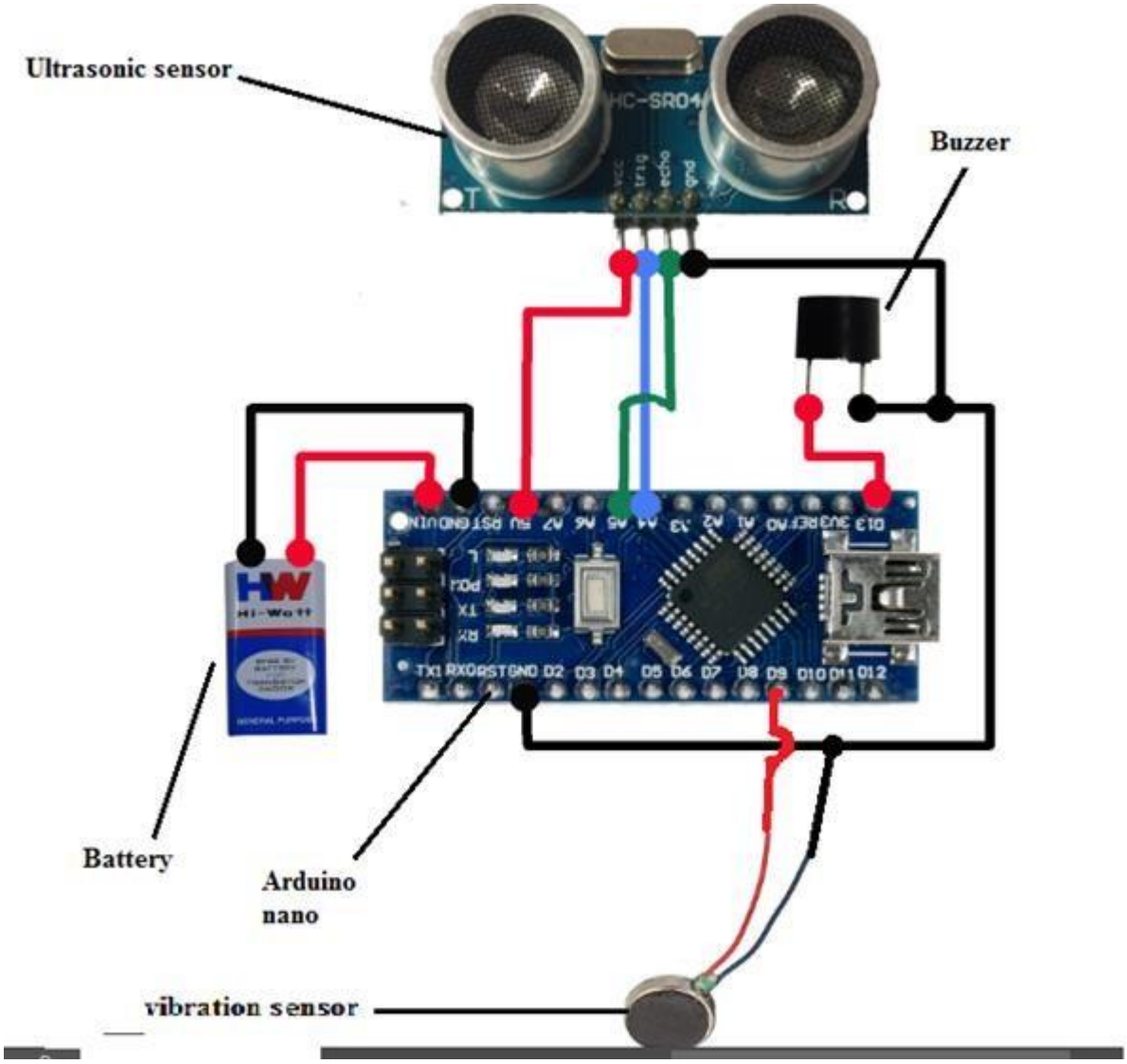
keep traffic flowing more smoothly. The aim is to optimally control the duration of green or red light for a specific traffic light at an intersection. The traffic signals should not flash the same stretch of green or red all the time, but should depend on the number of cars present. When traffic is heavy in one direction, the green lights should stay on longer; less traffic should mean the red lights should be on for longer time interval. This solution is expected to eliminate inefficiencies at intersections and minimize the cost of commuting and pollution.

PROJECT DESCRIPTION

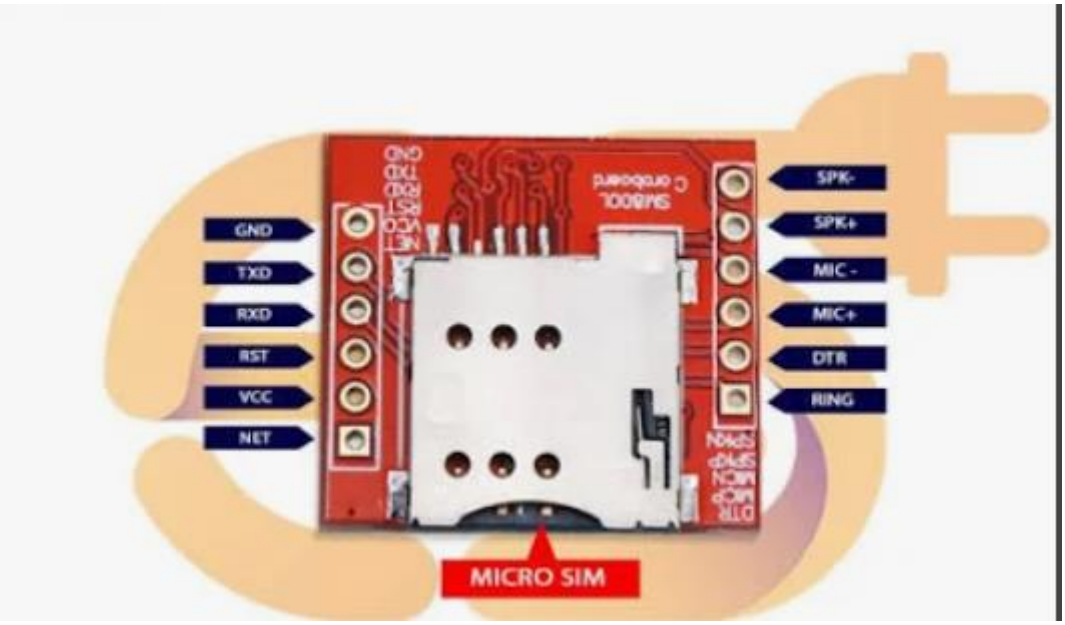
1. **Circuit DIAGRAM smart traffic management:**



2 ultra sonic sensor:



3.GSm sim module:



2. Concept Explanation:

A smart traffic management system utilizing sensor data, communication and auto- mated algorithms is to be developed to keep traffic flowing more smoothly. The aim is to optimally control the duration of green or red light for a specific traffic light at an intersection. The traffic signals should not flash the same stretch of green or red all the time, but should depend on the number of cars present. When traffic is heavy in one direction, the green lights should stay on longer; less traffic should mean the red lights should be on for longer time interval. This solution is expected to eliminate inefficiencies at intersections and minimize the cost of commuting and pollution

HARDWARE DESCRIPTION

COMPONENTS

ARDUINO NANO WITH CABLE

Arduino Nano is a type of microcontroller board, and it is designed by Arduino.cc. It can be built with a microcontroller like Atmega328. This microcontroller is also used in the Arduino UNO.

It is a small size board and also flexible with a wide variety of applications.

This board has many functions and features like an Arduino Demilunes board. However, this Nano board is different in packaging. It doesn't have any DC jack so that the power supply can be given using a small USB port otherwise straightly connected to the pins like VCC & GND. This board can be supplied with 6 to 20volts using a mini-USB port on the board.

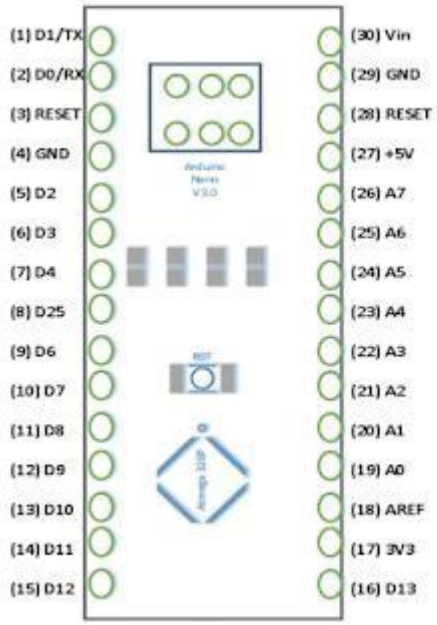
The communication of an Arduino Nano board can be done using different sources like using an additional Arduino board, a computer, otherwise using microcontrollers. The microcontroller using in Nano board (ATmega328) offers the serial communication (UART TTL). This can be accessible at digital pins like TX, and RX. The Arduino software comprises of a serial monitor to allow easy textual information to transmit and receive from the board.

The programming of an Arduino nano can be done using the Arduino software. Click the Tools option and select the nano board. Microcontroller ATmega328 over the Nano board comes with preprogrammed with a boot loader. This boot loader lets to upload new code without using an exterior hardware programmer. The communication of this can be done with the STK500 protocol.

ARDUINO NANO DIAGRAM

Arduino Nano Pinout

Arduino nano pin configuration is shown below, and each pin functionality is discussed below.



Power Pin (Vin, 3.3V, 5V, GND):

- Vin is the input voltage of the board, and it is used when an external power source is used from 7V to 12V.
- 5V is the regulated power supply voltage of the nano board and it is used to give the supply to the board as well as components.
- 3.3V is the minimum voltage which is generated from the voltage regulator on the board.
- GND is the ground pin of the board.

RST Pin (Reset): This pin is used to reset the microcontroller.

Analog Pins (A0-A7): These pins are used to calculate the analog voltage of the board within the range of 0V to 5V.

I/O Pins (Digital Pins from D0 – D13): These pins are used as an i/p otherwise o/p pins. 0V & 5V

Serial Pins (Tx, Rx): These pins are used to transmit & receive TTL serial data.

External Interrupts (2, 3): These pins are used to activate an interrupt.

PWM (3, 5, 6, 9, 11): These pins are used to provide 8-bit of PWM output.

ULTRASONIC SENSOR

An ultrasonic sensor is an instrument that measures the distance to an object using ultrasonic sound waves. An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity.

The ultrasonic sensor works on the principle of SONAR and RADAR system which is used to determine the distance to an object. An ultrasonic sensor generates the high-frequency sound (ultrasound) waves. When this ultrasound hits the object, it reflects as echo which is sensed by the receiver. Ultrasonic sensors work by emitting sound waves at a frequency too high for humans to hear.

They then wait for the sound to be reflected back, calculating distance based on the time required. This is similar to how radar measures the time it takes a radio wave to return after hitting an object.

While some sensors use a separate sound emitter and receiver, it’s also possible to combine these into one.

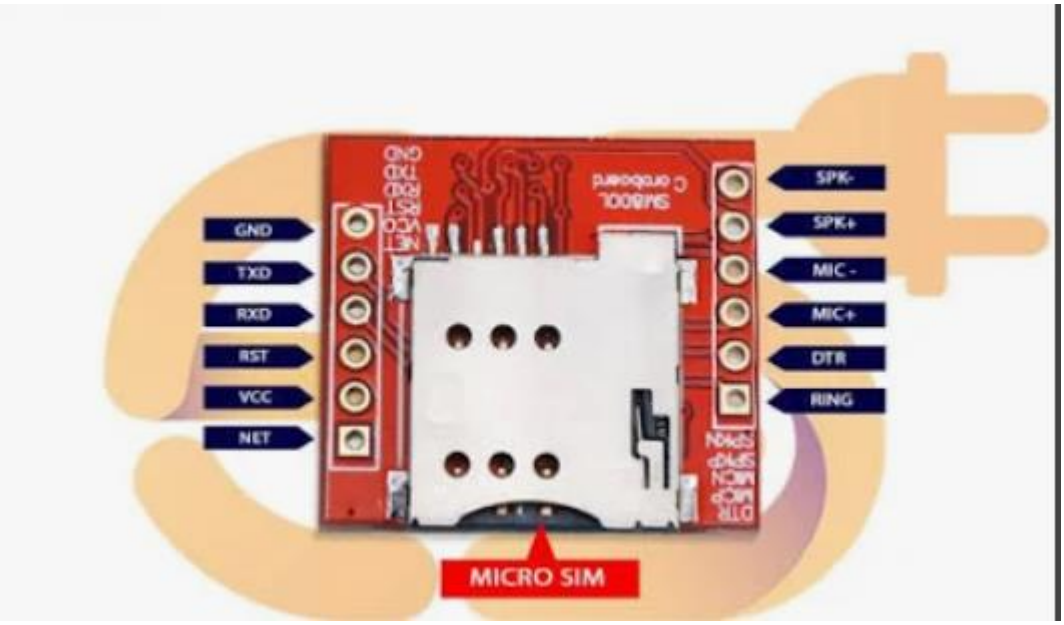
For ultrasonic sensing, the most widely used range is 40 to 70 kHz. The frequency determines range and resolution; the lower frequencies produce the greatest sensing range. At 58 kHz, a commonly used frequency, the measurement resolution is one centimeter (cm), and range is up to 11 meters having an ultrasonic element alternate between emitting and receiving signals. This type of sensor can be manufactured in a smaller package than with separate elements, which is convenient for applications where size is at a premium.

Ultrasonic sensors are used around the world, indoors and outdoors in the harshest conditions, for a variety of applications. Our ultrasonic sensors, made with piezoelectric crystals, use high frequency sound waves to resonate a desired frequency and convert electric energy into acoustic energy, and vice versa.

GSM SIM MODULE:

A GSM module or a GPRS module is a chip or circuit that will be used to establish communication between a mobile device or a computing machine and a GSM or GPRS system. The modem (modulator-demodulator) is a critical part here These modules consist of a GSM module or GPRS modem powered by a power supply circuit and communication interfaces (like RS-232, USB 2.0, and others) for computers.

A GSM modem can be a dedicated modem device with a serial, USB, or Bluetooth connection, or it can be a mobile phone that provides GSM modem capabilities.



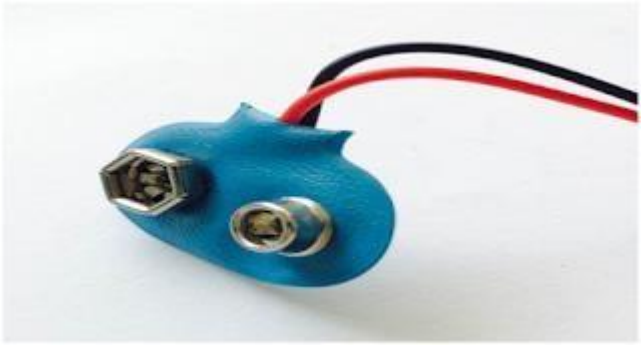
BATTERY CAP

A battery assembled cap, a cylindrical battery with the cap and a method for making the same. ... The vent cap is attached to the battery cover by a hinge connection which allows for play between the vent cap and the battery cover, and which allows for rotation of the vent cap.

A battery holder is one or more compartments or chambers for holding a battery.

For dry cells, the holder must also make electrical contact with the battery terminals. For wet cells, cables are often connected to the battery terminals, as is found in automobiles or emergency lighting equipment The purpose of the vent caps is to allow for the escape of gases formed when the battery is charging. In addition, the vent caps allow water and acid levels of the battery to be checked during maintenance. Lead-acid batteries can produce explosive mixtures of hydrogen and oxygen gases when they are being charged.
Battery can play an important role in achieving the target of universal access to clean, reliable and affordable electricity services. Battery is an energy storage device consisting of

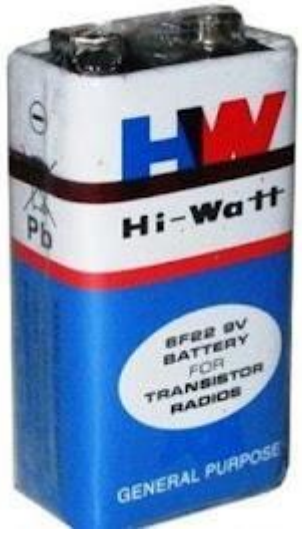
two or more electrochemical cells that convert stored chemical energy into electrical energy and used as a source of power.



9-VOLT BATTERY

A battery is an electrochemical device that produces a voltage potential when placing metals of different affinities into an acid solution (electrolyte).
The open circuit voltage (OCV) that develops as part of an electrochemical reaction varies with the metals and electrolyte used.
A battery is a device consisting of one or more electrochemical cells with external connections for powering electrical devices such as flashlights, mobile phones, and electric cars. When a battery is supplying electric power, its positive terminal is the cathode, and its negative terminal is the anode.

A battery is a device consisting of one or more electrochemical cells with external connections for powering electrical devices such as flashlights, mobile phones and electric cars. When a battery is supplying electric power, its positive terminal is the cathode, and its negative terminal is the anode. The terminal marked negative is the source of electrons that will flow through an external electric circuit to the positive terminal. When a battery is connected to an external electric load, a redox reaction converts high-energy reactants to lower-energy products, and the free energy difference is delivered to the external circuit as electrical energy.



SOLDERING WIRE

Solder is a fusible alloy used to join less fusible metals or wires, etc. ... Solder wire is comprised of different alloys, or of pure tin. Each metal requires a certain type of soldering wire to create strong bonds, because the combinations of metals that comprise soldering wire melt at different temperatures.

3. **LEDs**: LEDs are used for the purpose of signaling according to the traffic condition.

CONNECTING WIRE

Connecting wires allows an electrical current to travel from one point on a circuit to another because electricity needs a medium through which it can move. Most of the connecting wires are made up of copper or aluminum. Copper is cheap and good conductivity.

The Function of Connecting Wires. Connecting wires provide a medium to an electrical current so that they can travel from one point on a circuit to another. In the case of computers, wires are embedded into circuit boards to carry pulses of electricity.

WIRING INSTRUCTIONS

Ground of buzzer and vibration sensor to GND of Arduino

+ve of middle leg of switch to Arduino pin 5

+ve of Buzzer to first leg of switch

+ve of Vibration motor to third leg of switch

ULTRA-SONIC SENSOR PIN

Ultrasonic sensor pin VCC - Arduino pin VCC

Ultrasonic sensor pin GND - Arduino pin GND

Ultrasonic sensor pin Trig - Arduino pin 12

Ultrasonic sensor pin Echo - Arduino PIN 1

MAKING THE MODULES

1. First cut the board in 5 X 3 cm dimension and solder the female headers for the Arduino to the board.
2. Then solder the buzzer.

3. Then connect the vibrating motor using the glue gun and solder wires to it.
4. Then connect the LED.
5. Then connect the switch.
6. Then connect header pins for ultrasonic sensors and for battery input.
7. Then solder everything as shown in the circuit diagram.
8. Now connect the Arduino and ultrasonic sensor to the board.

WORKING OPERATION

The vehicle concentration or density will be detected by the IR LEDs and photodiodes. The controller will take the data from IR communication taking place in between IR Transmitter and IR Receiver. The microcontroller arduino nano will process this received data and show the signals through red, yellow and green LED's along with that controller will also ensure the delivery of the sensor data to users in form of a user friendly website where the people could watch the live traffic status.

For the manual control operation, the authorized person will send the request to the server with the login data i.e. the username and password. The server will match the user entered data with the pre-stored values, after successful

matching the server will open the page from which the user can set the direction in which he/she wants to display the green signal light.

The user after entering on the control page will be asked which mode of operation he/she would like to keep for the traffic. There are two options being provided one manual and the other is automatic. First, he/she have to select the manual mode and then as the user will click on opening a specific lane i.e. turning the green signal for that specific lane, the microcontroller will stop taking the data from IR Communication and will open the requested lane by showing the green signal in twenty seconds. When the authorized person is done its work, he/she will again have to set the control to automatic mode for the proper functioning of traffic. Then he/she can log out by clicking on the log out button, the server will redirect to the monitoring page.

The circuit has been designed and configured with the help of Proteus Design Suite. The LEDs; red, yellow and green, are connected in the matrix of 4× 3 where

4 represent the number of lanes and 3 represent respective LEDs. For the purpose of simulation and circuit design torch ldr is used as input collecting device. In the implementation of hardware the torch ldr's will be replaced by IR sensor (transmitter and receiver). The controller is also connected with virtual terminal which is acting as an IOT server. Just as how the user sends the request to server when once connected to web, we are also sending the request to virtual terminal and the virtual terminal it sends the response to user by displaying the messages on terminal.

ARDUINO PROGRAMMING

```
//Mohammed Zyaan C//////////////////////

////cmdzyaan27@gmail.com//////////////

//8072372498////////////////////////////

///vellore//////////////////////////////

#include<TimerOne.h>

////////////////////////////////////////////////////////////////
/

#include <SoftwareSerial.h>


//--------------------------------------------------------------------------------

//Alert reciever's phone number with country code

const String PHONE_1 = "+918072372498";

const String PHONE_2 = ""; //optional

const String PHONE_3 = ""; //optional
```

```
//--------------------------------------------------------------------------------


//--------------------------------------------------------------------------------
#define rxPin 2
#define txPin 3
SoftwareSerial sim800L(rxPin,txPin);
#define buzzer_pin A4
////////////////////////////////
nt signal1[] = {13, 12, 11};
int signal2[] = {10, 9, 8};
int signal3[] = {7, 6, A5};
int redDelay = 2000;
int yellowDelay = 2000;


volatile int triggerpin1 = A0;
volatile int echopin1 = A1;
volatile int triggerpin2 = A2;
volatile int echopin2 = A3;
volatile int triggerpin3 =4;
volatile int echopin3 = 5;
```

```
int count =0;

volatile long time;                // Variable for storing the time traveled

volatile int S1, S2 ,S3;           // Variables for storing the distance covered


int t = 5;  // distance under which it will look for vehicles.


void setup(){

  Serial.begin(115200);

  sim800L.begin(9600);

  Timer1.initialize(100000);  //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<2; i++){

for(int i=0; i<2; i++){

    pinMode(signal1[i], OUTPUT);

    pinMode(signal2[i], 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);

////////////////////////////////////////////////////////////////
///
 Serial.println("Initializing...");
 //Once the handshake test is successful, it will back to OK
 sim800L.println("AT");
 delay(1000);
 sim800L.println("AT+CMGF=1");
 delay(1000);

////////////////////////////////////////////////////////////////
/
}
```

```
void loop()
{
  signal01Function();
  signal02Function();
  while(sim800L.available())
{
    signal01Function();
    signal02Function();
    delay(100);
    if ((S1 < t ||(S2 < t)) && count == 0)
      {
      Serial.println(sim800L.readString());
      digitalWrite(buzzer_pin,HIGH);
      Serial.println("trafic detected...! take action immediately.");
      send_multi_sms();
      count=1;

  }
 else if(S1 > t && S2 > t && count == 1)
 {
```

```
  delay(10);
  Serial.println("count");
  Serial.println(count);
  count=0;
  }
}
}
{
  // 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 second ultrasonic sensor
 digitalWrite(triggerpin3, LOW);
 delayMicroseconds(2);
 digitalWrite(triggerpin3, HIGH);
 delayMicroseconds(10);
digitalWrite(triggerpin3, LOW);
 time = pulseIn(echopin3, HIGH);
 S3= time*0.034/2;

 // Print distance values on serial monitor for debugging
/*Serial.println("S1=");
  Serial.println(S1);
 Serial.println("S2=");
   Serial.println(S2);
 Serial.println("S3=");
```

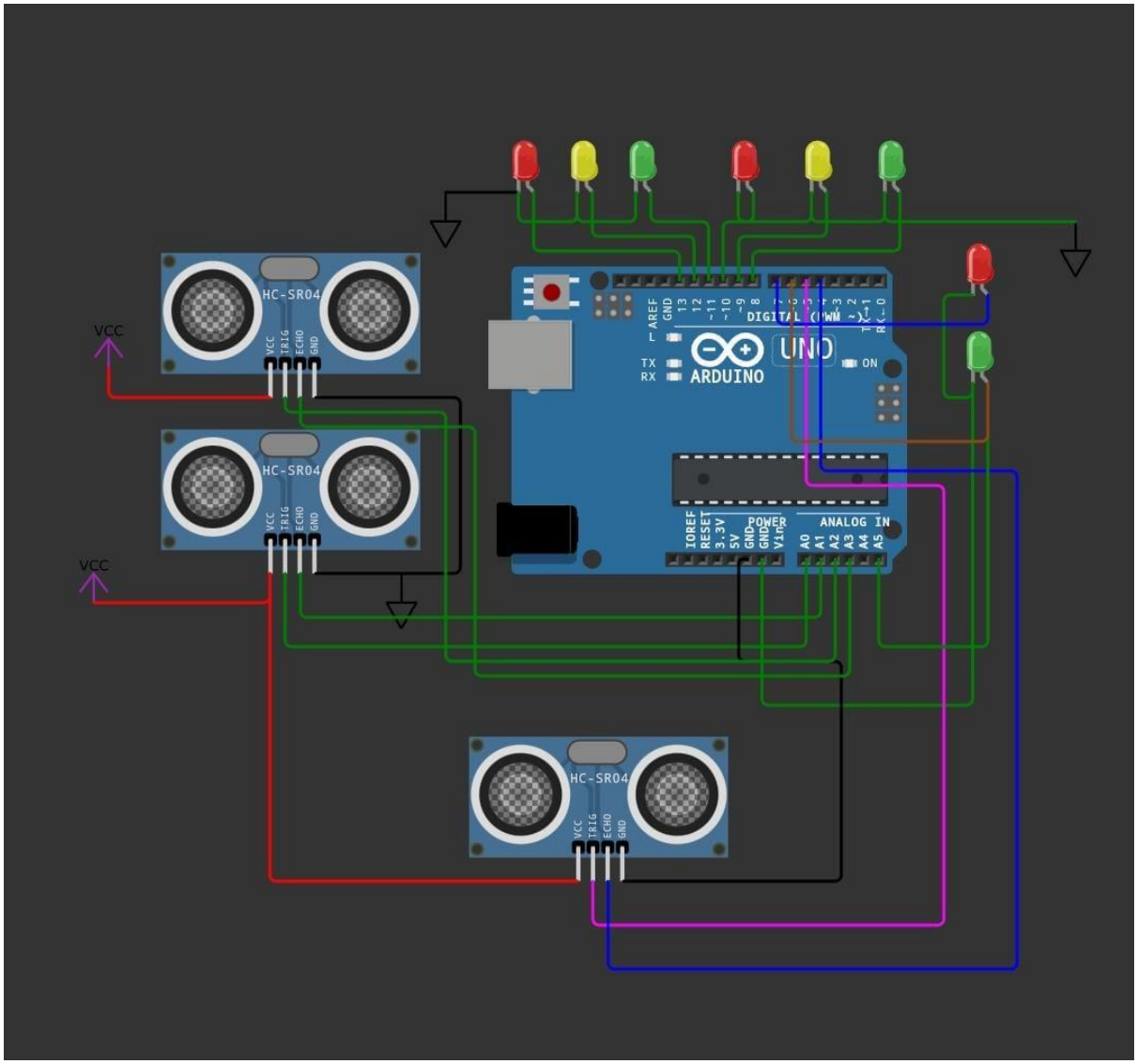
```
   Serial.println(S3);
 */
}

void zeebra_level ()
{
for(int i=0;i<10;++i)
  {
   delay(300);
   if (S3 < 20)
  {i=0;}
  }
}
void signal01Function()
{
void(* resetFunc) (void) = 0; //declare reset function @ address 0

  Serial.println("01");
  //low();
  digitalWrite(signal1[0], LOW);
```

```
  digitalWrite(signal1[2], HIGH);
  digitalWrite(signal2[0], LOW);
  digitalWrite(signal2[2], HIGH);
  delay(5000);
  digitalWrite(signal1[2], LOW);
  digitalWrite(signal1[1], HIGH);
  digitalWrite(signal2[2], LOW);
  digitalWrite(signal2[1], HIGH);
  delay(1000);
  digitalWrite(signal1[1], LOW);
  digitalWrite(signal1[0], HIGH);
  digitalWrite(signal2[1], LOW);
  digitalWrite(signal2[0], HIGH);
 }

void signal02Function()
{
  Serial.println("02");
  //low();
  digitalWrite(signal3[0], LOW);
```

```
  digitalWrite(signal3[2], HIGH);
zeebra_level ();
  digitalWrite(signal3[2], LOW);
  digitalWrite(signal3[1], HIGH);
  delay(1000);
  digitalWrite(signal3[1], LOW);
  digitalWrite(signal3[0], HIGH);
}

// Function to make all LED's LOW except RED one's.
void send_multi_sms()
{
  if(PHONE_1 != ""){
    Serial.print("Phone 1: ");
    send_sms(" trafic Detected..! take action immediately.", PHONE_1);
  }
}
//-------------------------
void send_sms(String text, String phone)
```

```
{
   Serial.println("sending sms....");
   delay(50);
   sim800L.print("AT+CMGF=1\r");
   delay(1000);
   sim800L.print("AT+CMGS=\""+phone+"\"\r");
   delay(1000);
   sim800L.print(text);
   delay(100);
sim800L.write(0x1A);  //ascii code for ctrl-26
//Serial2.println((char)26); //ascii code for ctrl-26
   delay(3000);
   count++;
   Serial.println("count");
   Serial.println(count);
}
```

RESULT



THE presented system is designed and configured for the use of the blind and visually disabled people. This device can handle several states that the visually impaired people face. This device responds to the user in all the circumstances which is faced by the blind people with the help of the use of the Ultrasonic sensors and the Arduino Board.

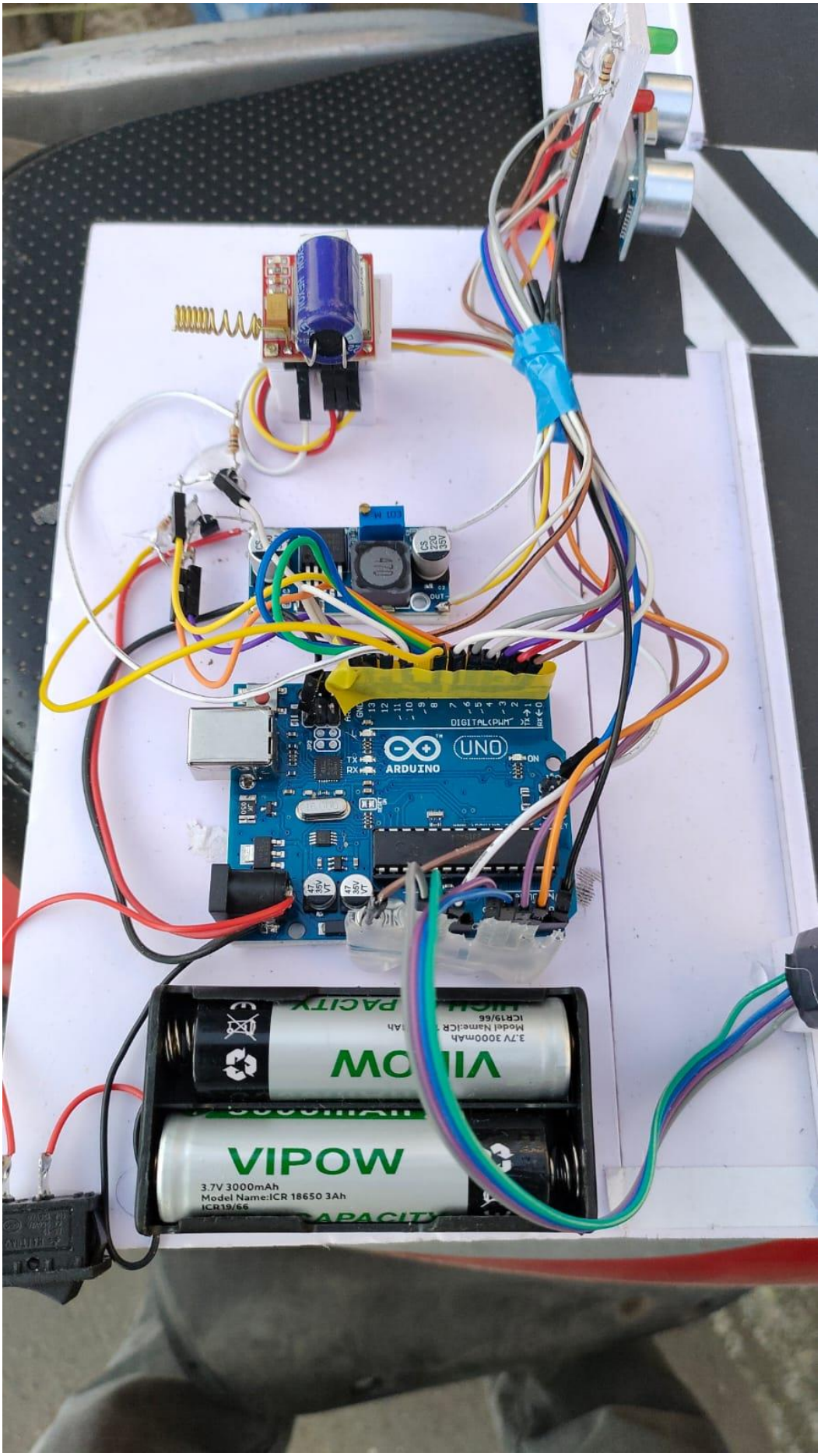
Case 1: When the obstacle or the object is in the left it will tell the user that:
The obstacle is in left.
Case 2: When the obstacle is in right it will say: The obstacle in right. Case 3: When the obstacle is in front, the device will say: the

obstacle is in front. Similarly for all the directions like left, right, back etc. it will notify the user wearing it.

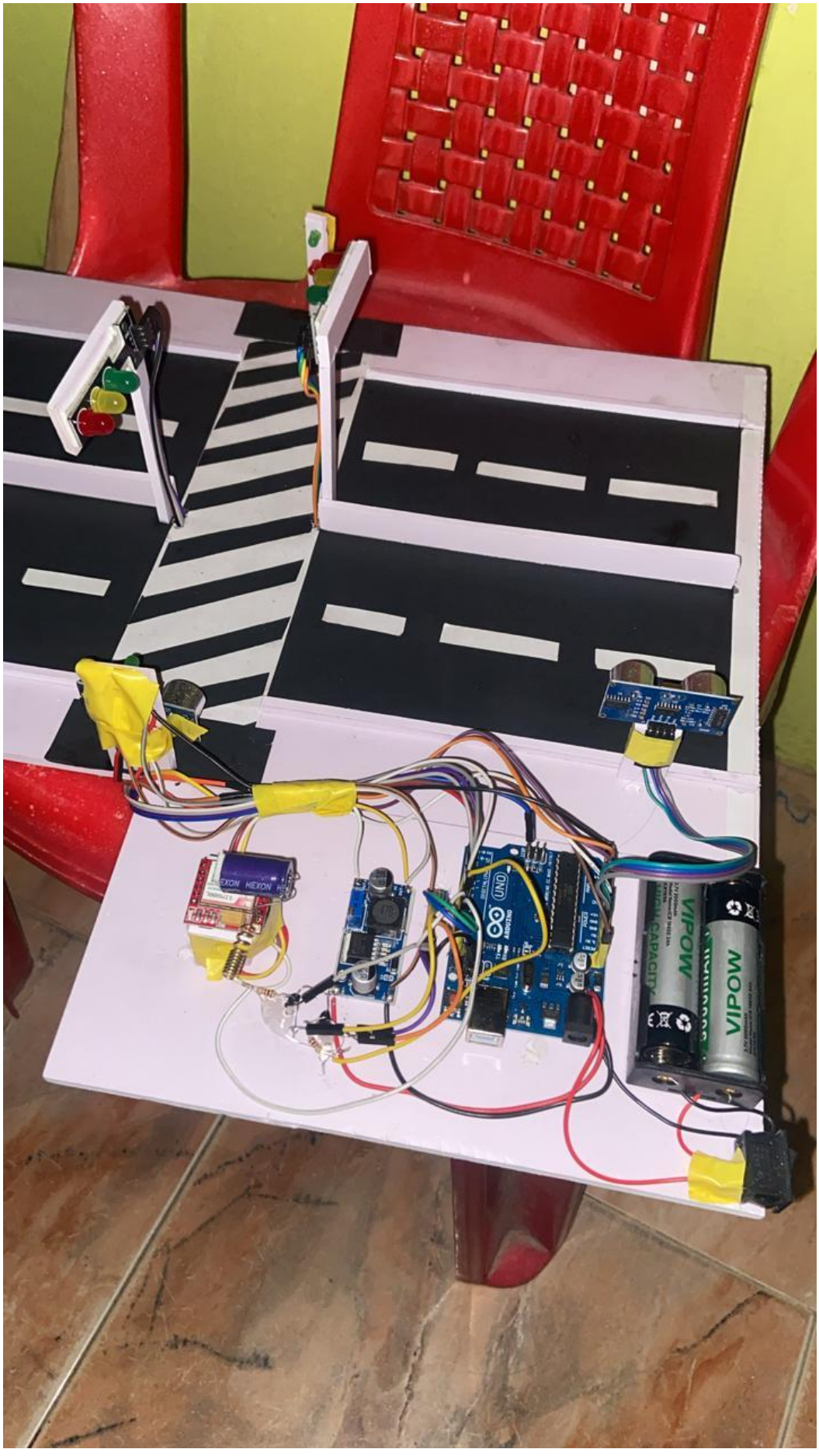
CONCLUSION:

Traffic management is one of the biggest infrastructure hurdles faced by developing countries like India today. There is an exigent need of efficient traffic management system in our country, as almost every indian waste their precious time getting stuck in traffic. To reduce this congestion and unwanted time delay in traffic, an advanced system is designed here in this project. With field application of the IOT technology, the maddening chaos of traffic can be effectively channelized by distributing the time slots based on the merit of the vehicle load in certain lanes of multi junction crossing. We have successfully implemented the prototype at laboratory scale with remarkable outcomes.

With this project, the idea is spread to set green signals in accordance with the type of traffic concentration; this would save the time of common people and some of health issues like headaches because of the noises being generated on roads in reople keep on banging their cars and other vehicles horns. This projec emphasizes the importance of IoT based congestion control, live traffic monitoring as well as the controlling o the traffic manually.


ARDUINO
UNO
VIPOW
3.7V 3000mAh
Model Name:ICR 18650 3Ah
ICR19/66

Full pictorial presentation:


VIPOW
VIPOW