

Traffic management system using IOT

INTRODUCTION

Traffic flowing to a junction from all the directions is most of the time unequal. This factor is not taken into account by the conventional traffic management system leading to unbalanced distribution of traffic from all directions which leads to air pollution and sever health issues, as there are around 67 lakhs vehicles in single metropolitan city like Bangalore where people are facing this issue.

A smart traffic management system is, where traffic is constrained by the administrative framework, which controls the traffic lights as per the continuous circumstance of traffic moving from every extraordinary bearing in an intersection. This continuous information is gathered from different sensors set at equivalent intervals of separation at an intersection. This information is gathered and brought to a control framework which independently ascertains the ideal time for the arrival of the green sign to every specific heading in an intersection so as to counteract traffic heaping up [1].

The subsequent significant issue which we are taking a gander at in this venture is about the development of crisis vehicles (fire engines, ambulances and police vehicles) in packed intersections of a city. Because of the absence of foundation present in our nation to address the issue, concerning the speedy development of crisis vehicles in a city, has carried us to the decision about making a self-governing framework to help the crisis vehicles to travel through the city without issue.

In this framework, we are proposing a powerful technique wherein these vehicles can be recognized by utilizing radio wave flagging strategy. The crisis vehicles will be distinguished and will be included inside a framework that discharges a SOS signal which can be identified by a sign discovery unit which sends a crisis trigger to the traffic the executive's framework. In this framework, the crisis vehicle will be recognized around 1 kilometer from the sign and when it arrives at 500 meters from the sign then the control unit gives green sign toward the path in which the crisis vehicle is drawing closer so as to give a sign free hall and to avert the crisis vehicle stalling out an automobile overload. An order of precedence is set if there is more than one emergency vehicle approaching from many directions. The order of precedence is given as follows:

- Ambulance
- Fire truck
- Police vehicles

The order of precedence also takes into account the factor that includes a combination of more than two types of emergency vehicles approaching the signal from the same direction then the trigger is fired and the order of precedence is given to this particular direction compared to the direction a single emergency vehicle is approaching.

SYSTEM DESIGN AND ARCHITECTURE

ARDUINO NANO



An 8 bit Microchip AVR which is small, complete and bread board friendly board based on the Atmega328. It is the main CPU of our Project, in which we all the program will run.

POWER SUPPLY MODULE

A power supply is a hardware component that provides power to any electrical device.

IR SENSOR

These sensors are used to detect the object through infrared rays. The rays which are thrown from the sensors are reflected back by the object by which it encountered and then after captured by these infrared sensors which further gets converted into electric signals. These sensors are put sideways for giving us the density of vehicles in the specific lane. Infrared sensors are used for signal control, detection of pedestrians in crosswalks and transmission of traffic information.

The basic disadvantages of infrared sensors are that the operation of the system may be affected due to fog; also installation and maintenance of the system is tedious.

WIFI MODULE

It is used to give microcontroller access to your wifi network.

RFRX MODULE

It consists of RF transmitter and RF receiver; it is used for transmitting and receiving data.

LED

Light bulbs are used for output and instruction for this system.

BLYNK APP

It is a mobile application for output and verification for real time data collected.

COMPONENTS

Smart traffic management system consists the following components.

- Radio signal detector
- Radio waves transmitter
- Ultra sonic sensor/Hall Effect sensor
- Raspberry Pi
- Python programming
- Light Emitting Diode

Radio signal detector:

In a radio, the device that receives and takes the information from a modulated radio frequency current or voltage is called a detector. While differentiating to the present-day radio stations which transmit sound (for instance: a sound signal) on an undisturbed bearing wave, radio stations in the early days at the outset transmitted data by radiotelegraphy. The transmitter was turned on and off to create short or extensive stretches of radio waves, illuminating messages in various codes, similar to that of Morse code. In this manner, the early radio recipients had just a single application or function that was to separate between the nearness or nonappearance of any radio sign. The gadget that played out this specific capacity was known.



As a locator.

The collector utilizes a receiving wire so it can catch radio waves, process these waves to retain just those waves that are vibrating at the necessary recurrence, extricates the sound signal that were added to those waves, intensifies the sound signal, lastly plays them on a speaker. The parts of a radio detector are:-

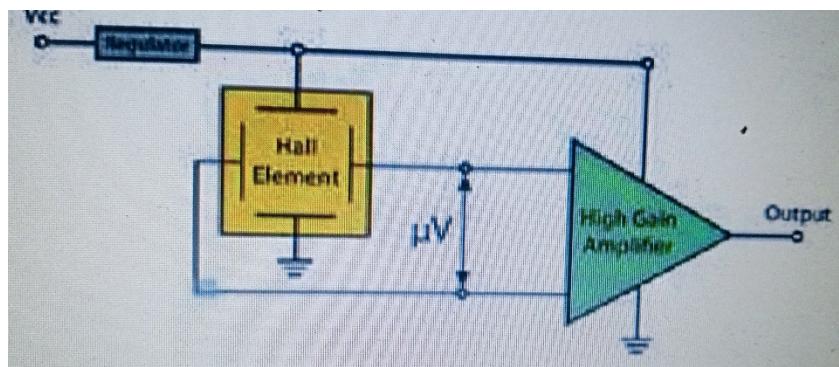
- Antenna: It helps in catching the radio waves. Commonly, the reception apparatus is basically a long wire. At the point when this wire is vulnerable to radio waves, the waves cause a little substituting current (AC) inside the receiving wire.
- RF enhancer: It is a sense speaker that enhances the exceptionally weak radio recurrence signal from the reception apparatus with the goal that the signal can be prepared by the tuner.
- Tuner: A circuit that can pull back sign of a specific recurrence from a blend of signs of various frequencies.
- Detector: This part is answerable for isolating the sound data from the transmitter wave. For AM (Amplitude balance) sign, this can be satisfied with the assistance of a diode that just amends the rotating current sign.
- Audio speaker: The motivation behind this part is to enhance the powerless sign that originates from the identifier with the aim that it tends to be heard by anybody. This can be satisfied utilizing a basic transistor intensifier circuit.

Hall Effect Sensor:

The Hall Effect is the most common method of measuring magnetic field and the Hall Effect sensors are very popular and have many contemporary applications. For example, they can be found in vehicles as wheel speed sensors as well as crankshaft or camshaft position sensors. If we bring some magnetic field near the plate we would disturb the straight flow of the charge carriers due to a force, called Lorentz Force. In such a case the electrons would deflect to one side of the plate and the holes to the other side of the plate. This means if we put a meter now between the other two sides we will get some voltage which can be measured. The basic Hall Element of the Hall Effect magnetic sensors mostly provides very small voltage of only a few microvolts per Gauss, so therefore, these devices are usually manufactured with built-in high gain amplifiers.

--Hall effect sensor

There are two Effect sensors, analog and the output. The



types of Hall
one providing
other digital
analog sensor

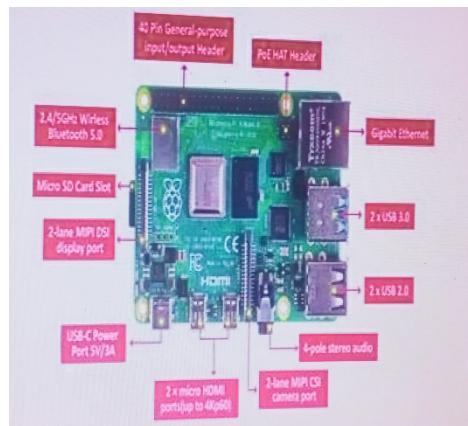
is composed of a voltage regulator, a Hall Element and an amplifier. The digital output sensors provide just two output states, either "ON" or "OFF".

Advantages:

- High speed operation over 100 KHz possible. Whereas at high frequencies the inductive or capacitive sensor output begins to distort.
- Non contact operation so there is no wear and friction, hence unlimited number of operating cycles.
- When packed immune to dust, air, water whereas capacitive sensor may get triggered by dust.
- It can measure zero speed.
- Highly repeatable operation.
- Capable of measuring large current.

Disadvantages:

- It may be affected by external interfering magnetic field.
- Large temperature drift.
- Large offset voltage



Raspberry pi

The Raspberry Pi is a small sized personal computer (PC) which is structured and fabricated by the Raspberry Pi Foundation (a non-profit association) which is dedicated to making PCs and programming guidelines as effectively open as conceivable to the intended interest group. Despite the fact that the first aim of the Raspberry Pi venture was to PCs with programming choices under the control of undergraduates, the Raspberry Pi has been taken by a various objective group of spectators. Software engineers over the world have taken the modest stage for ventures which are from reproducing retro formed cupboards to controlling robots and to setting up modest however amazing home media gadgets. Coming up next are the upsides of Raspberry Pi over PCs and comparable gadgets.

Flow Chart with Algorithm

Case1

1. Start
2. Check the vehicle Density
3. Vehicle density++
4. Is vehicle density <Threshold
5. Yes
6. Normal Traffic
7. Give green signal to each Lane in a sequential manner

Case2

1. Start
2. Check the vehicle Density
3. Vehicle density++
4. Is vehicle density <Threshold
5. No
6. Status =congestion 7. Compare the number of density in each lane
8. Open the lane with highest number of density
9. Remove current Lane from Comparison
10. Then start once again.

WIFI MODULE WILL KEEP ALL THE EMBEDDED NETWORK COMPONENTS INTERCONNECTED TO EACH OTHER AND TRANSFER THE CONTROL TO SERVER.

READ THE REAL TIME DATA COLLECTED FROM THE IR SENSOR AND ASSIGN THE CONTROL AS PER THE INBUILD PROGRAM ASSIGNED TO IT.

ANALYSE THE DENSITY OF TRAFFIC LANES THROUGH IR SENSOR AND TRANSFER THE DATA TO THE MICROCONTROLLER.

SERVER WILL SHOW THE RESULT AS PER THE INPUT OBTAINED AND TRAFFIC LIGHT WILL WORK ACCORDING TO THAT.

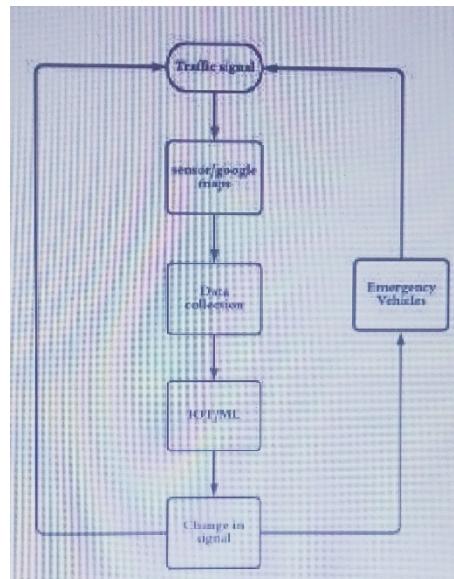
METHODOLOGY

- This is the basic ideology behind controlling the timing of traffic lights with respect to present time traffic conditions.



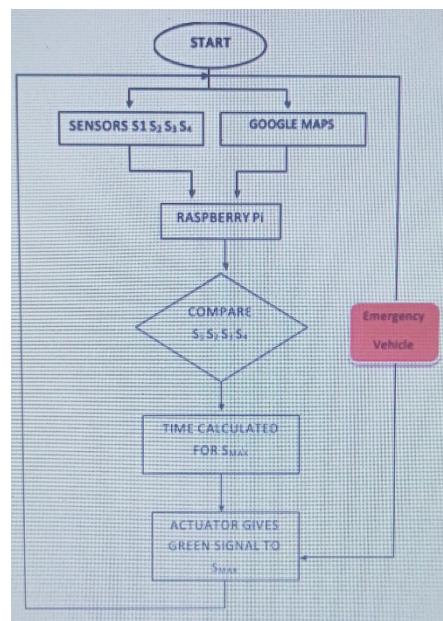
Edit with WPS Office

- The sensor collects data of the real-time density of vehicles present on the road.
- The data from the sensors are collected and stored in the cloud.
- This information is fed to the microcontroller which determines the change in signal for each lane.
- During the case of an emergency, the data is directly fed into the microcontroller hence terminating previous loop and changing the signal immediately.



Process flow chart

Complex flow chart



data collection and priority line up

Proposed system:

This traffic management control system consists of Hall effect sensors, LED lights (red and green) and Raspberry Pi. The Raspberry Pi microcontroller contains the python programming code which controls time delay of led lights. The Hall effect sensors placed on the road surface at regular intervals of distance to detect the presence of vehicles on the road till where it is placed. The government is suggested to place a radio wave transmitter on all emergency vehicles hence the emitted signal can be collected by the radio detector to identify the presence of any emergency vehicle[8]. The radio waves detector is placed at every junction in order to detect the presence of radio wave emitter that is placed inside the emergency vehicle. Raspberry Pi microcontroller is placed at every junction which collects the information from the various sensors placed at different locations and the resultant time is calculated and the LED lights are turned on by the microcontroller for the stipulated time calculated.

Phase-1:-

We are assuming 4 different quadrants to imitate a four-directional intersection junction and have placed 3 Hall effect sensors on 3 sides along with one green light and red light on each side. All the terminals from the Hall effect sensor and the Led lights are connected to the gpio pins of the raspberry pi. The instantaneous data of a particular moment is collected from all the sensors from every direction and the program compares and identifies the side having maximum traffic volume and the green light is given to the particular side with maximum volume and the other three sides will remain red. As the vehicles move out, the moment the last active sensor detects the absent of vehicle, the program goes on to check which other side has maximum traffic. Considering there are four directional intersections, once the green light is given to side from next iteration onwards preference is given to the other three sides[9]. During the third cycle preference is given to the last two sides and the onwards.

Phase-2:-

When the autonomous managing traffic control system is switched on, it collects all the information from various sensors like the ultrasonic detectors and radio wave detectors. The sensor data (S1 S2 S3 S4) is collected by the control unit and the side with the maximum traffic is identified along with the distance of traffic on each side. The raspberry pi controller then calculates the optimum time for the release of green signal to the side that has been identified with the maximum distance of traffic. At the end of the time provided by the microcontroller, the sensor data from all the other sensors are collected and the side with the highest traffic distance has been identified and the green signal is given to that side. In the next case the last two sensor data is ignored and the other sensor readings are identified and green light signal preference is given to the side which has the highest traffic distance compared to the other sensor values. This process is continued till all the sides leading to an intersection has been given at least one green signal in the first loop[10]. At the beginning of the second loop, all the sensor data are again taken and the side with the highest traffic is identified and the microcontroller determines the optimum time for release of green signal in that particular direction. This process continues till the system is turned off manually. If there is a scenario in which the radio wave detectors detect the presence of an incoming radio signal which signifies that an emergency vehicle is approaching. Thus, a trigger is fired and the system is alerted about the approaching emergency vehicle. This radio wave signal is again monitored by the control unit and as the radio wave emitter reaches the 500 meters from the junction then the microcontroller fires the final trigger



and all the running process is cancelled and the preference is transferred in the direction of emergency vehicle is approaching.

The microcontroller immediately changes the other direction signals to red and provides green signal to the direction in which the emergency vehicle is approaching. If there are emergency vehicles approaching in more than one direction then the order of preference is as follows:

Ambulance > Fire Truck > Police trucks

In the condition in which there are two emergency vehicles approaching in one direction and single emergency vehicle in another direction, the preference is given to the direction in which there are 2 or more than 2 emergency vehicles are approaching.

PROGRAMMING

```
import RPi.GPIO as GPIO  
from time import sleep  
  
Hallpin1=8  
  
#LED1=8  
  
Hallpin2=10  
  
Hallpin3=12  
  
#Hallpin4=24  
  
Hallpin11=22  
  
Hallpin12=24  
  
Hallpin13=26  
  
Hallpin21=38  
  
Hallpin22=40  
  
Hallpin23=37  
  
Hallpin31=31  
  
Hallpin32=29  
  
Hallpin33=23  
  
LED1=16  
  
LED2=18  
  
LED11=32  
  
LED12=36  
  
LED21=35  
  
LED22=33  
  
LED31=21  
  
LED32=19  
  
GPIO.setwarnings(False)
```



Edit with WPS Office

```

GPIO.setmode(GPIO.BCM)

GPIO.setup(LED1, GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(LED2, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(hallpin1, GPIO.IN)
#GPIO.setup(LED2, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(hallpin2, GPIO.IN)
GPIO.setup(hallpin3, GPIO.IN)

GPIO.setup(LED11, GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(LED12, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(hallpin11, GPIO.IN)
GPIO.setup(hallpin12, GPIO.IN)
GPIO.setup(hallpin13, GPIO.IN)

GPIO.setup(LED21, GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(LED22, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(hallpin21, GPIO.IN) GPIO.setup(hallpin22, GPIO.IN)
GPIO.setup(hallpin23, GPIO.IN)

GPIO.setup(LED31, GPIO.OUT, initial=GPIO.LOW)
GPIO.setup(LED32, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(hallpin31, GPIO.IN)
GPIO.setup(hallpin32, GPIO.IN)
GPIO.setup(hallpin33, GPIO.IN)

while True:

    print("-----")

    if(GPIO.input(hallpin1)==True):
        # GPIO.output(LED1, GPIO.HIGH)
        a1=1
        print("magnet 1")
        print("detected")
        if(GPIO.input(hallpin1)==False):
            a1=0
            print("magnet 1")
            print("not detected")
            if(GPIO.input(hallpin2)==True):
                a2=1
                print(" magnet 2")

```



```
print(" detected")
if(GPIO.input(hallpin2)==False):
    a2=0
    print(" magnet 2")
    print(" not detected")
if(GPIO.input(hallpin3)==True):
    a3=1
    print(" magnet 3")
    print(" detected")
if(GPIO.input(hallpin3)==False):
    a3=0
    print("magnet 3")
    print(" not detected")
print("-----")
if(GPIO.input(hallpin11)==True):
    b1=1
    print("magnet 11")
    print("detected")
if(GPIO.input(hallpin11)==False):
    b1=0
    print(" magnet 11")
    print(" not detected")
if(GPIO.input(hallpin12)==True):
    b2=1
    print(" magnet 12")
    print(" detected")
if(GPIO.input(hallpin12)==False):
    b2=0
    print("magnet 12")
    print(" not detected")
if(GPIO.input(hallpin13)==True):
    b3=1
    print(" magnet 13")
    print(" detected")
if(GPIO.input(hallpin13)==False):
```



Edit with WPS Office

```

b3=0

print(" magnet 13")
print(" nvt detected")
print("-----")
print("nvt detected")

sum1=a1+a2+a3

sum2=b1+b2+b3

print(sum1)

print(sum2)

f1=0

f2=0

if(f1==1)and(f2==1):

f1=0

f2=0

if(f1==0):

if(sum1>sum2):

GPIO.vputput(LED1, GPIO.HIGH)

GPIO.vputput(LED12, GPIO.HIGH)

GPIO.vputput(LED22, GPIO.HIGH)

GPIO.vputput(LED32, GPIO.HIGH)

sleep(15)

GPIO.vputput(LED1, GPIO.LOW)

GPIO.vputput(LED12, GPIO.LOW)

GPIO.vputput(LED22, GPIO.LOW)

GPIO.vputput(LED32, GPIO.LOW)

f1=1

if(f2==0):

if(sum2>sum1):

GPIO.vputput(LED11, GPIO.HIGH)

GPIO.vputput(LED2, GPIO.HIGH)

GPIO.vputput(LED22, GPIO.HIGH)

GPIO.vputput(LED32, GPIO.HIGH)

sleep(15)

GPIO.vputput(LED11, GPIO.LOW)

GPIO.vputput(LED2, GPIO.LOW)

```



```
GPIO.output(LED22, GPIO.LOW)
GPIO.output(LED32, GPIO.LOW)
f2=1
```

RESULT:-

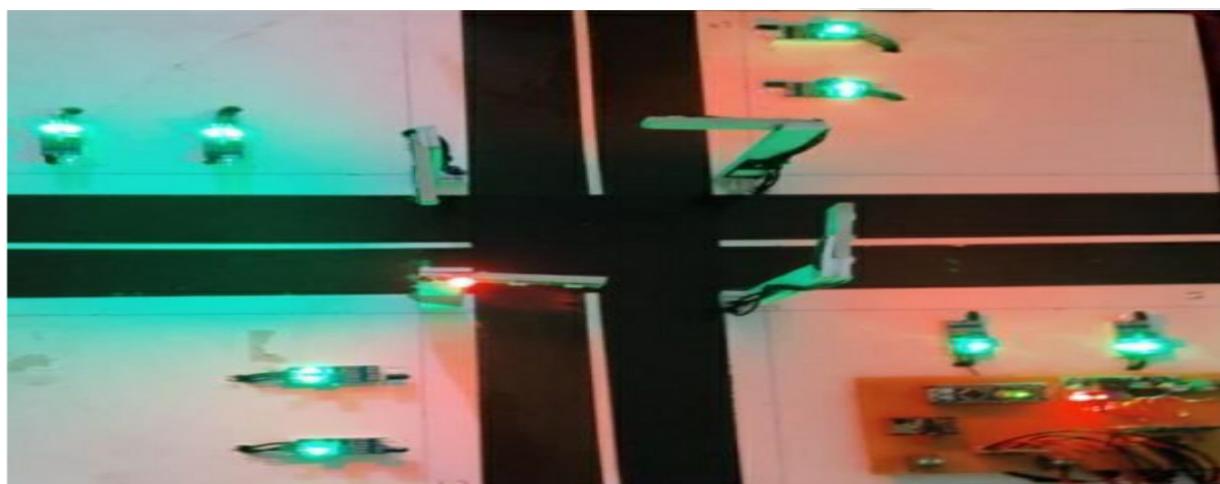
The proposed system will give output on the basis of density of the traffic lanes so there will be particular time assigned to the lanes for the passing of vehicles so it will lower down the traffic congestion and also in the case of emergency or any road block issue there will be option for that to continue the traffic in normal way and by using blynk app we can check that the system is working according to input or not.

We can see from the image given about the output are coming for various lanes the density in lane one is 100% so other traffic lights are red and the traffic light of lane one will be green until the traffic is cleared so it will repeat for each lane and in the case of emergency on lane will be assigned and other lane will be blocked to control the case of emergency.

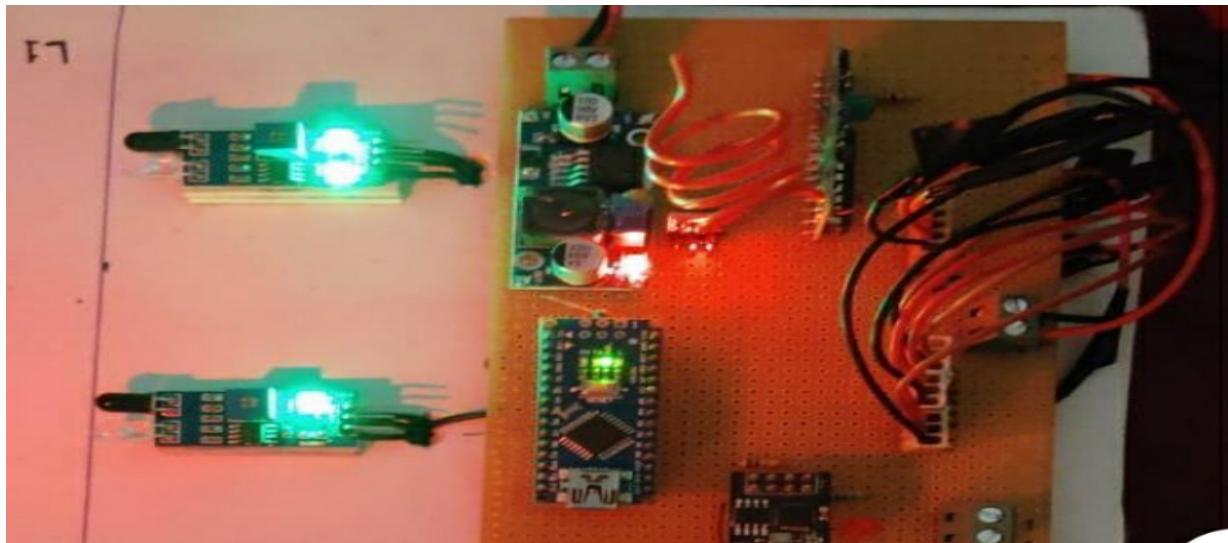
The proposed system helps in better time based monitoring and thus has certain advantages over the existing system like minimizing number of accidents, reducing fuel cost and is remotely controllable etc.

The system will be designed in such a way that it will able to control the traffic congestion as well as track the no of vehicles. And it can also be maintained very easily.

The project is user friendly and can replace the existing project.



(a)



(b)

CONCLUSION

smart traffic management system has given the best results to with waiting & travelling time of a passenger has been reduced and emergency vehicles can move without obstacles or barriers. The pollution rate can be reduced by implementing this smart traffic management system in all prime locations. The suggested traffic management system can be implemented in all metropolitan cities as it is most suitable and reliable for the day.