|  |
| --- |
|  |
| **Submitted by,** |
|  |
| ABINAYA M - 812921106001  SUSITHRA S - 812921106034 |
| DHANALAKSHMI S - 812921106007 |
| GOMATHI S - 812921106012 |
|  |
|  |

**SMART TRAFFIC MANAGEMENT**



TABLE OF CONTENTS

* Abstract
* Objectives
* Introduction
* Block diagram
* Innovation
* Prevention methods
* Components
* Radio signal detector
* Radio waves transmitter
* Ultra-sonic sensor/Hall Effect sensor
* Raspberry Pi
* Python programming
* Light Emitting Diode
* Program
* Output
* Application
* Advantages
* Disadvantages
* Conclusion

ABSTRACT

Traffic congestion is a major problem in many cities of India along with other countries. Failure of signals, poor law enforcement and bad traffic management has lead to traffic congestion. One of the major problems with Indian cities is that the existing infrastructure cannot be expanded more, and thus the only option available is better management of the traffic. Traffic congestion has a negative impact on economy, the environment and the overall quality of life. Hence it is high time to effectively manage the traffic congestion problem. There are various methods available for traffic management such as video data analysis, infrared sensors, inductive loop detection, wireless sensor network, etc. All these methods are effective methods of smart traffic management. But the problem with these systems is that the installation time, the cost incurred for the installation and maintenance of the system is very high. Hence a new technology called Radio Frequency Identification (RFID) is introduced which can be coupled with the existing signaling system that can act as a key to smart traffic management in real time. This new technology which will require less time for installation with lesser costs as compared to other methods of traffic congestion management. Use of this new technology will lead to reduced traffic congestion. Bottlenecks will be detected early and hence early preventive measures can be taken thus saving time and money of the driver.

Objective

Smart Traffic Management Systems are technology solutions that municipalities can integrate into their traffic cabinets and intersections today for fast, cost-effective improvements in safety and traffic flow on their city streets. What's more, deploying these systems today, or upgrading your city's existing Intelligent Transportation Systems (ITS) infrastructure can create huge efficiencies and cost savings, while massively improving system reliability, all of which have excellent ROI.

These systems utilize sensors, cameras, cellular routers and automation to monitor and automatically direct traffic and reduce congestion. The right technology solution can be scaled to any size and painlessly upgraded at any time. Simultaneously, these technology solutions prepare Smart Cities for coming technology evolutions, including Connected Vehicle and the full deployment of 5G networks.

Budgets for public infrastructure are always tight, and constructing roads and bridges is always expensive. Smart Traffic Management Systems help municipal and regional transportation departments to cope with the situation — quickly and cost-effectively. Integrating smart traffic technology helps them affordably get better performance from their existing .

Introduction

A smart traffic management system utilizing cameradata, communication and automated 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 vehicles 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 a longer time interval.

Traffic congestion on road networks is nothing but slower speeds, increased trip time and increased queuing of the vehicles. When the number of vehicles exceeds the capacity of the road, traffic congestion occurs. In the metropolitan cities of India traffic congestion is a major problem. Traffic congestion is caused when the demand exceeds the available road capacity. This is known as saturation [1]. Individual incidents such as accidents or sudden braking of a car in a smooth flow of heavy traffic have rippling effects and cause traffic jams [2]. There are even severe security problems in traffic system due to anti social elements which also leads to stagnation of traffic at one place. In country like India, there is an annual loss of Rs 60,000 crores due to congestion (including fuel wastage). Congestion in India has also led to slow speeds of freight vehicles, and increased waiting time at checkpoints and toll plazas [3].

The average speed of vehicles on key corridors like Mumbai-Chennai, Delhi-Chennai is less than 20kmph, while it is mere 21.35kmph on Delhi-Mumbai stretch. As per the transport corporation of India and IIM, India’s freight volume is increasing annually at a rate of 9.08% and that of vehicles at 10.76%, but that of road is only by 4.01%. This has resulted in reduced road space in accordance with the number of total vehicles [3].The average fuel mileage in India is only 3.96kmpl. The major reason for this is traffic congestion [3].India is the 2nd most populated country after China in Asia, thus with increase in population, the number of vehicles also increase [4].

The economic growth has certainly has had an impact on urban traffic. As the income rises, more and more people begin to go for cars rather than two wheelers [5].Hence there is a need to manage traffic in a smart way as the management of traffic with the conventional way such as the signaling system is not having a major effect in curbing congestion of vehicular traffic.

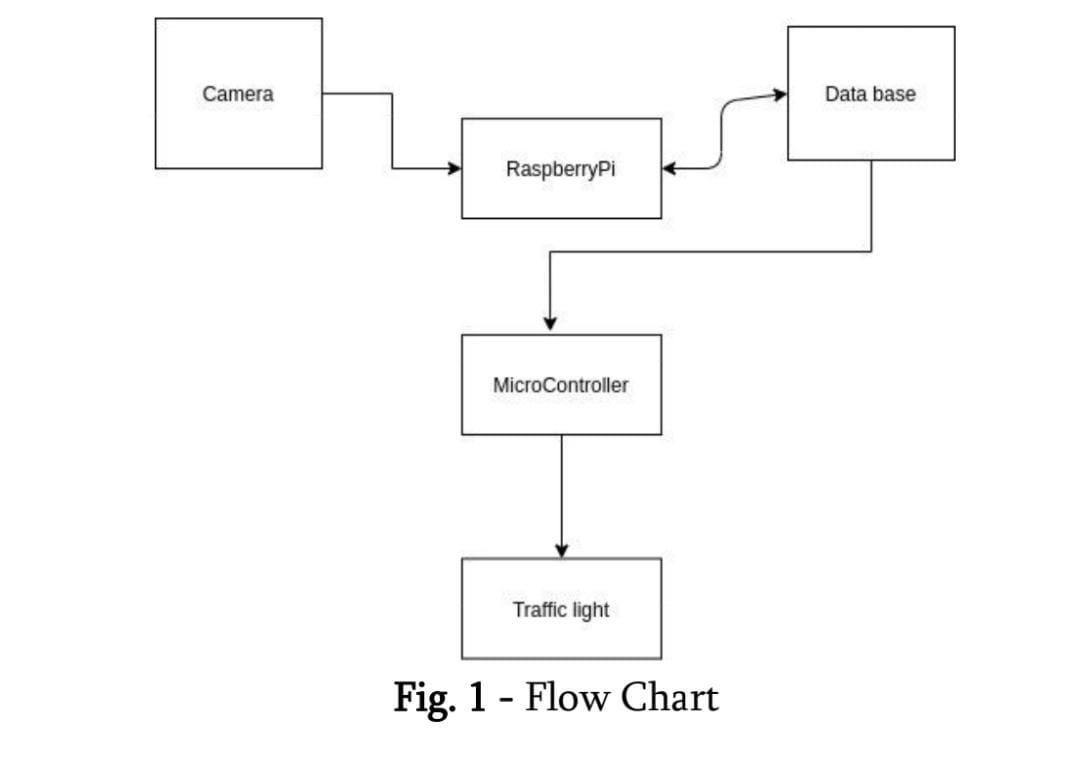
# System design

1) Raspberry Pi

2) LED lights which are used for the purpose of signaling.

3) Traffic cameras which are used for monitoring traffic.

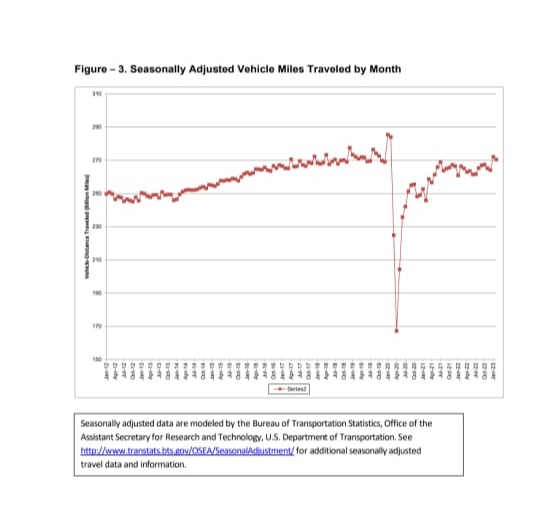
4) Node MCU Microcontroller

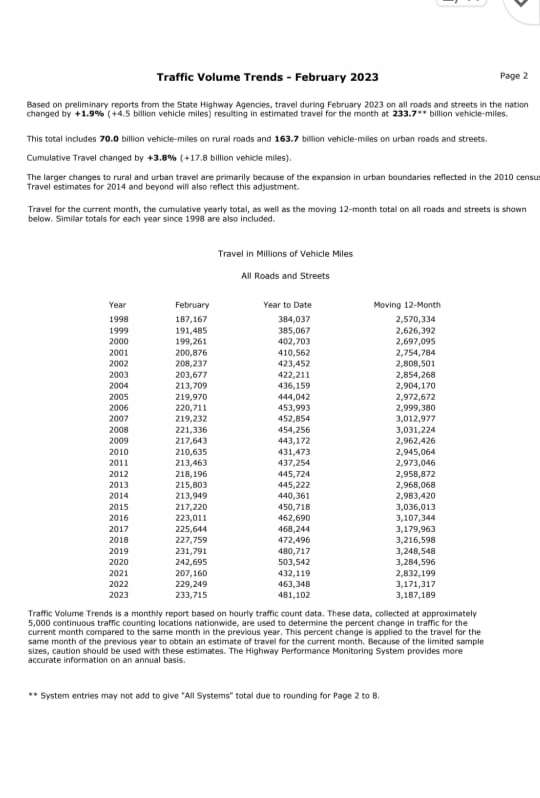
Block diagram

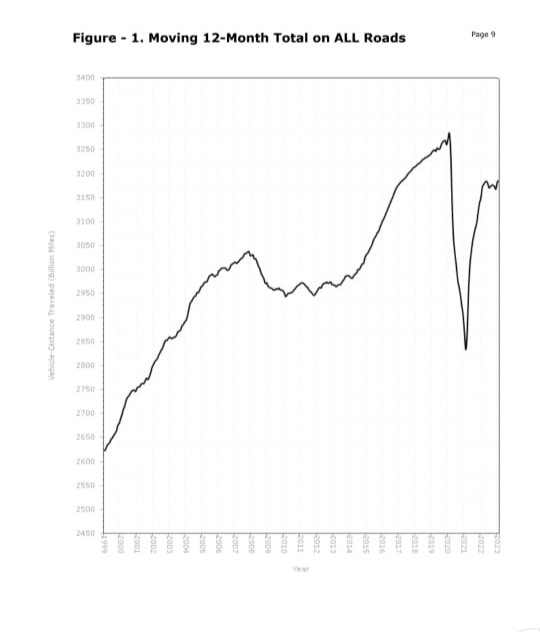
Innovation:-

Travel on all roads and streets changed by +1.9% (+4.5 billion vehicle miles) for February 2023 as compared with February 2022. Travel for the month is estimated to be 233.7 billion vehicle miles,

The seasonally adjusted vehicle miles traveled for February 2023 is 270.5 billion miles, a +1.5% (+4.1 billion vehicle miles) change over February 2022. It also represents a -0.6% change (-1.5 billion vehicle miles) compared with January 2023,Cumulative Travel for 2023 changed by +3.8% (+17.8 billion vehicle miles).







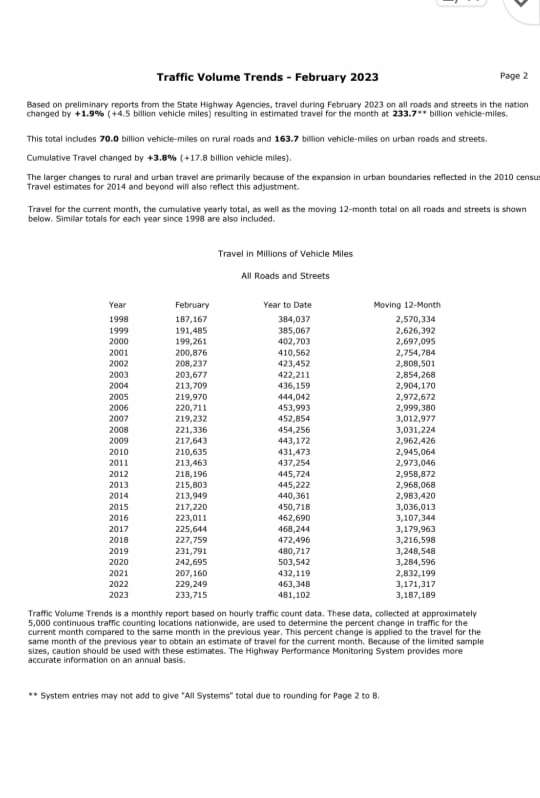
Prevention methods:-

**Use Of Autonomous Or Self-Driving Vehicles:-**

Utilizing self-driven cars can reduce traffic congestion to some level. Generally, traffic congestion is caused by human error, road rage, and the desire to reach early at someplace.

**Taxing Vehicles parking:-**

The growth in population and urbanization may have some cons in disguise that may lead to traffic congestion. It has become an urban cult in that every person likes to own a private vehicle and travel to work through it. A huge segment of the masses mostly avoids public transport systems and prefers to travel privately.



**Automated Traffic Light System:-**

Another smart way through which traffic jams can be prevented is an automated traffic light system. With IoT in transportation, the working of traffic light systems can be enhanced. With data-driven features, traffic signals can be made adaptive and automated.

The National Highway Traffic Safety Administration today released its early estimates of traffic fatalities for the first half of 2023, estimating that traffic fatalities declined for the fifth straight quarter. An estimated 19,515 people died in motor vehicle traffic crashes, representing a decrease of about 3.3% as compared to 20,190 fatalities in the first half of 2022. Fatalities declined in both the first and second quarters of 2023.

Continuing the trend identified in the first quarter estimates released in June, preliminary data shows vehicle miles traveled in the first half of 2023 increased by about 35.1 billion miles, roughly 2.3% higher than the same time last year. More miles driven combined with fewer traffic deaths resulted in a fatality rate of 1.24 fatalities per 100 million VMT, down from the projected rate of 1.31 fatalities per 100 million VMT in the first half of 2023.

"After spiking during the pandemic, traffic deaths are continuing to slowly come down—but we still have a long way to go,” U.S. Transportation Secretary Pete Buttigieg said. “Safety has always been the core mission of this Department, and thanks to President Biden, we are delivering unprecedented resources to communities across the country to make their streets safer.”

“While we are encouraged to see traffic fatalities continue to decline from the height of the pandemic, there’s still significantly more work to be done,” NHTSA Acting Administrator Ann Carlson said. “NHTSA is addressing traffic safety in many ways, including new rulemakings for lifesaving vehicle technologies and increased Bipartisan Infrastructure Law funding for state highway safety offices. We will continue to work with our safety partners to meet the collective goal of zero fatalities.”

NHTSA estimates a decrease in fatalities in 29 states, while 21 states, Puerto Rico, and the District of Columbia, are projected to have experienced increases.

NHTSA has announced several safety initiatives aimed at reducing traffic deaths, including proposed rulemakings to require automatic emergency braking systems in passenger cars, light trucks and heavy vehicles.

## SYSTEM IMPLEMENTATION

Steps in the proposed system for controlling traffic light 1:

1. Camera: Continuously record traffic video.

2. Read Image: Read frames of the traffic image.

3. Grayscale Image Conversion: It converts color image to grayscale image. This method is based on different color transforms. According to the R, G, B value in the image, it calculates the value of grayscales and converts the image into a grayscale image.

4. Image Binarization: Grayscale image is converted into black and white image.

5. Traffic Signal Control: Based on vehicle count signal timings are changed and the respective LED glows.

Steps for controlling traffic light 2:

1. Initialize System.

2. Configure ESP 8266 module for multi access point through AT commands.

3. Connect WI-FI module to WI-FI network.

4. Start UDP local port in WI-FI module.

5. Establish UDP connection to Raspberry pi.

6. Wait for data.

7. Change traffic light signal 2 depending upon their received data from raspberry Pi.

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

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 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 sign can be prepared by the tuner.

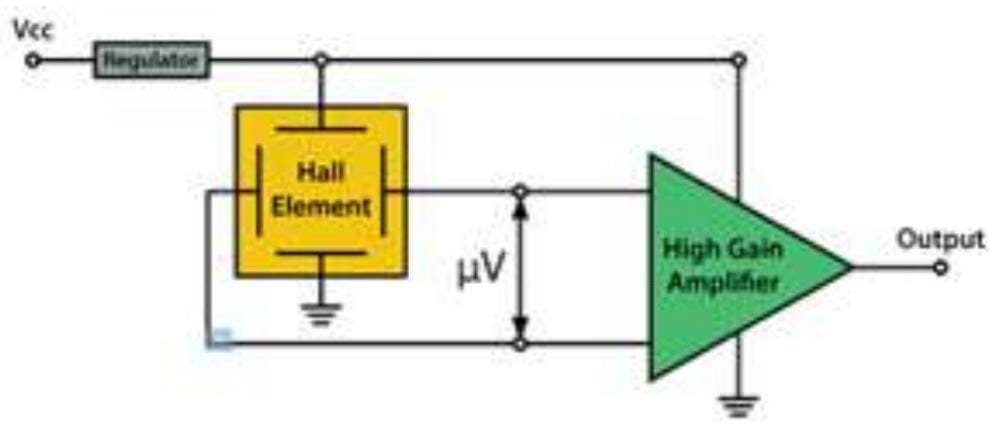
•Tuner: A circuit that can pull back sign of a specific recurrence from a blend of sign of various

frequencies.

• Detector: This part is answerable for isolating the sound data from the transporter wave. For AM (Amplitude balance) flag, 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

## Hall Effect Sensor

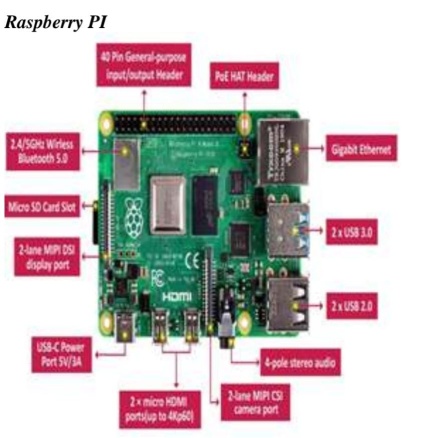
 The Hall Effect is the most common method of measuring magnetic field and the Hall Effect sensors are very popular and have many contemporary applications. For example, they can be found in vehicles as wheel speed sensors as well as crankshaft or camshaft position sensors.The basic Hall Element of the Hall Effect magnetic sensors mostly provides very small voltage of only a few micro volts per Gauss, so therefore, these devices are usually manufactured with built-in high gain amplifiers.

## Raspberry PI

The Raspberry Pi is a small sized personal computer (PC) which is structured and fabricated by the Raspberry Pi Foundation (a non-benefit association) which is dedicated to making PCs and programming guidelines as effectively open as conceivable to the intended interest group. Software engineers over the world have taken the modest stage for ventures which are from reproducing setting up modest however amazing home media gadgets.

## Advantages

•It is a solitary board PC

•It is very cost effective.

## Light-Emitting Diode

A light-producing diode is a semiconductor which has a light source that conveys light when current is permitted to move through it. Electrons in the semiconductor join with the electron gaps, along these lines discharging vitality as photons. LEDs have numerous points of interest over other radiant light sources.

They are:

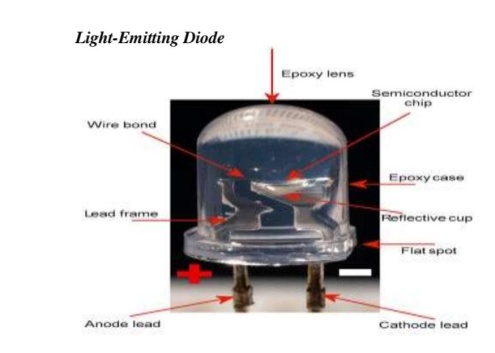
•Lower vitality utilization

• Longer lifetime

•Improved strength

•Small in size

• Faster rate of exchanging



Program:

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)

GPIO.setmode(GPIO.BOARD)

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):

b3=0

print(" magnet 13")

print(" not detected")

print("------------------------------

if(GPIO.input(hallpin21)==True):

c1=1

print(" magnet 21")

print(" detected")

if(GPIO.input(hallpin21)==False):

c1=0

print("magnet 21")

print("not detected")

if(GPIO.input(hallpin22)==True):

c2=1

print("magnet 22")

print("detected")

if(GPIO.input(hallpin22)==False):

c2=0

print(" magnet 22")

print("not detected")

if(GPIO.input(hallpin23)==True):

c3=1

print("magnet 23")

print(" detected")

if(GPIO.input(hallpin23)==False):

c3=0

print("magnet 23")

print("not detected")

print("-------------------------------")

if(GPIO.input(hallpin31)==True):

d1=1

print("

magnet 31")

print("

detected")

if(GPIO.input(hallpin31)==False):

d1=0

print("

magnet 31")

print("

not detected")

if(GPIO.input(hallpin32)==True):

d2=1

print("

magnet 32")

print("

detected")

if(GPIO.input(hallpin32)==False):

d2=0

print("

magnet 32")

print("

not detected")

if(GPIO.input(hallpin33)==True):

d3=1

print("

magnet 33")

print("

detected")

if(GPIO.input(hallpin33)==False):

d3=0

print("

magnet 33")

print("

not detected")

sum1=a1+a2+a3

sum2=b1+b2+b3

sum3=c1+c2+c3

sum4=d1+d2+d3

print(sum1)

print(sum2)

print(sum3)

print(sum4)

f1=0

f2=0

f3=0

f4=0

if(f1==1)and(f2==1)and(f3==1)and(f4==1):

f1=0

f2=0

f3=0

f4=0

if(f1==0):

if(sum1>sum2)and(sum1>sum3)and(sum1>sum4):

GPIO.output(LED1, GPIO.HIGH)

GPIO.output(LED12, GPIO.HIGH)

GPIO.output(LED22, GPIO.HIGH)

GPIO.output(LED32, GPIO.HIGH)

sleep(15)

GPIO.output(LED1, GPIO.LOW)

GPIO.output(LED12, GPIO.LOW)

GPIO.output(LED22, GPIO.LOW)

GPIO.output(LED32, GPIO.LOW)

f1=1

if(f2==0):

if(sum2>sum1)and(sum2>sum3)and(sum2>sum4):

GPIO.output(LED11, GPIO.HIGH)

GPIO.output(LED2, GPIO.HIGH)

GPIO.output(LED22, GPIO.HIGH)

GPIO.output(LED32, GPIO.HIGH)

sleep(15)

GPIO.output(LED11, GPIO.LOW)

GPIO.output(LED2, GPIO.LOW)

GPIO.output(LED22, GPIO.LOW)

GPIO.output(LED32, GPIO.LOW)

f2=1

if(f3==0):

if(sum3>sum1)and(sum3>sum2)and(sum3>sum4):

GPIO.output(LED21, GPIO.HIGH)

GPIO.output(LED2, GPIO.HIGH)

GPIO.output(LED12, GPIO.HIGH)

GPIO.output(LED32, GPIO.HIGH)

sleep(15)

GPIO.output(LED21, GPIO.LOW)

GPIO.output(LED2, GPIO.LOW)

GPIO.output(LED12, GPIO.LOW)

GPIO.output(LED32, GPIO.LOW)

f3=1

if(f4==0):

if(sum4>sum1)and(sum4>sum2)and(sum4>sum3):

GPIO.output(LED31, GPIO.HIGH)

GPIO.output(LED2, GPIO.HIGH)

GPIO.output(LED12, GPIO.HIGH)

GPIO.output(LED22, GPIO.HIGH)

sleep(15)

GPIO.output(LED31, GPIO.LOW)

GPIO.output(LED2, GPIO.LOW)

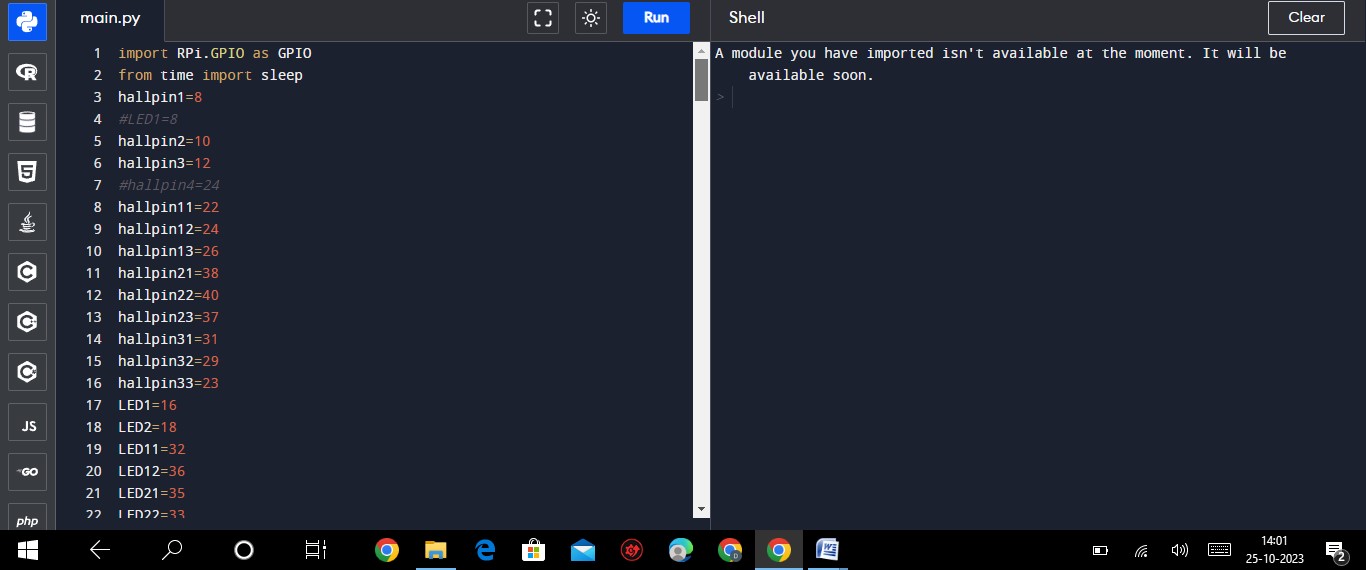
GPIO.output(LED12, GPIO.LOW)

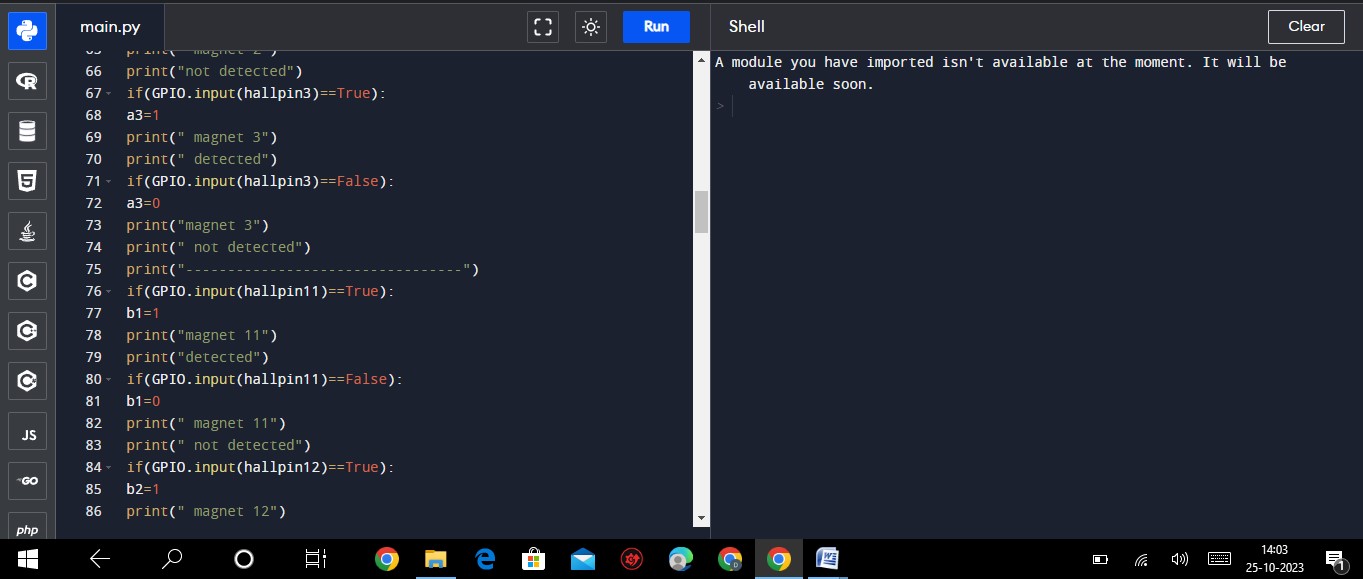
GPIO.output(LED22, GPIO.LOW)

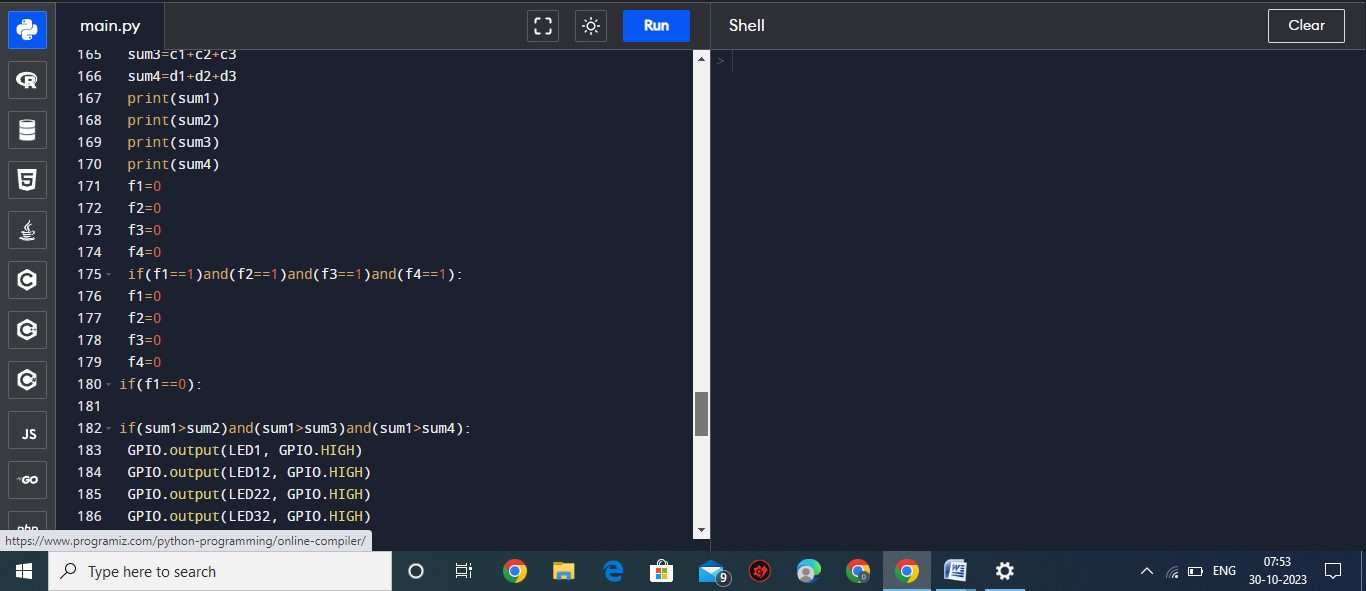
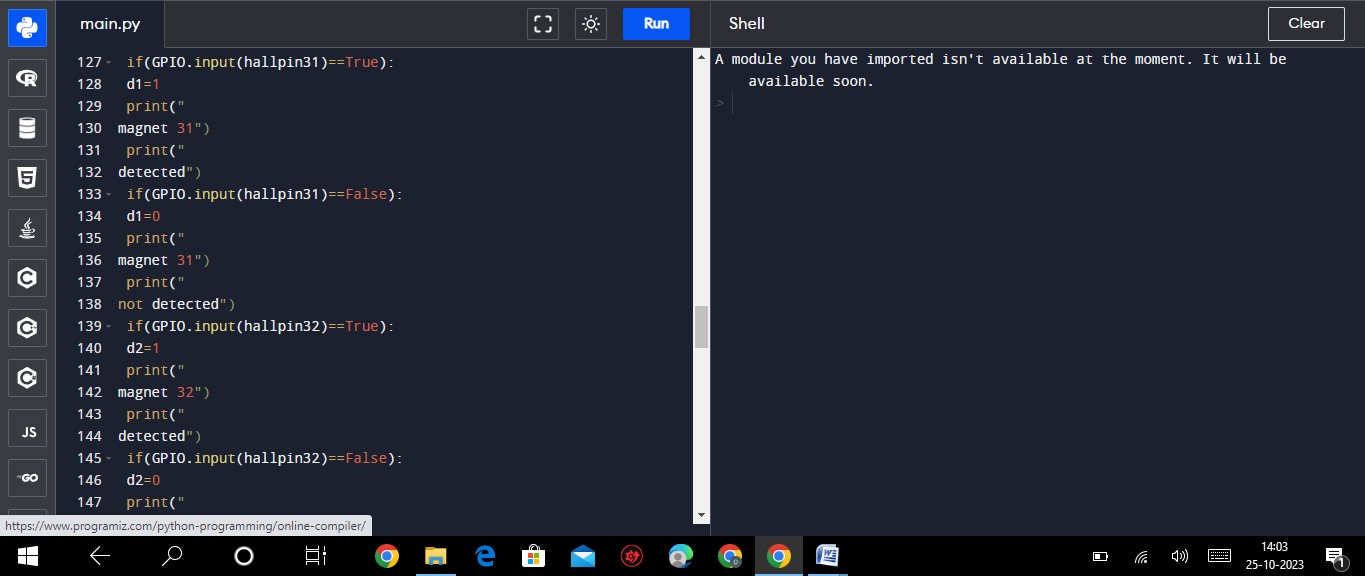
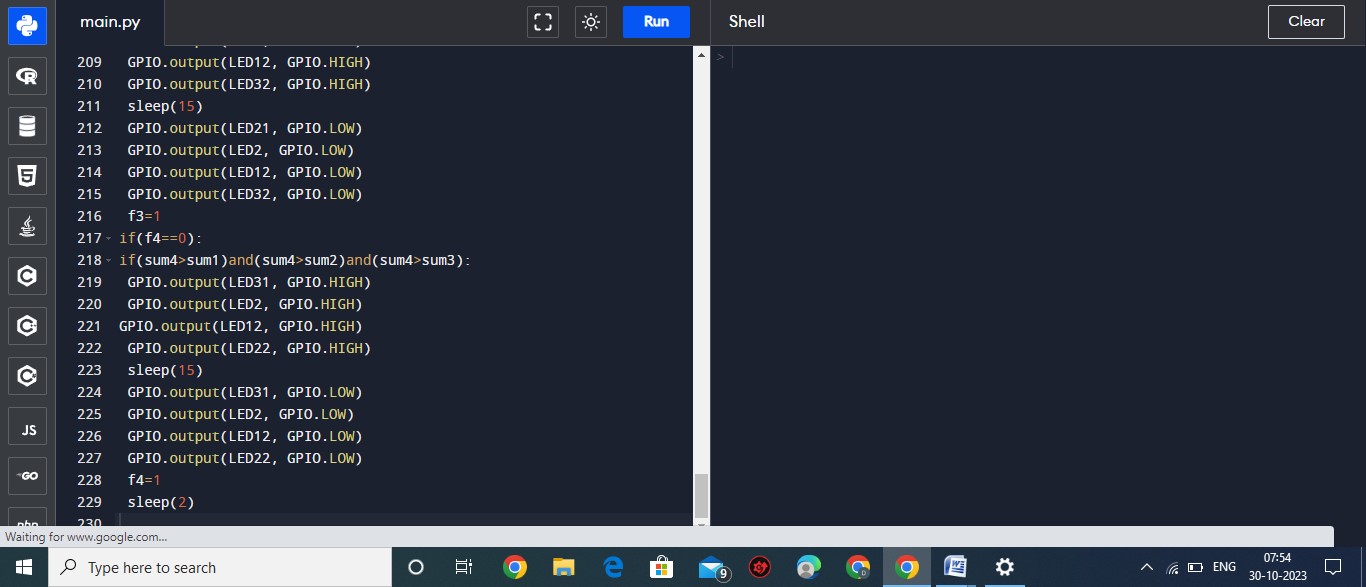
f4=1

sleep(2)

Program







Output

Moderate traffic. Alternating signals.

High traffic. Red signal for the main road.

Low traffic. Green signal for the main road. Moderate traffic. Alternating signals.

Moderate traffic. Alternating signals.

High traffic. Red signal for the main road.

High traffic. Red signal for the main road.

High traffic. Red signal for the main road.

Application

Detection and Management of traffic Congestion

In addition to the earlier method of traffic congestion detection, one more method can be used. A server can be maintained which can receive certain crucial data calculated by the Controller of the signals. The main aim is to implement a system that would trace the travel time of individual cars as they pass the roadside controllers and compute an average trip time using a rule-based system to decide whether the area is congested or uncongested. If congestion is sensed then system would control traffic signals / generate automatic re-routing messages to selected approaching vehicles.

Automatic detection of speed limit

Violation We can use this technique to calculate the speed of a motorist and to detect if he violates the prescribed/set speed limit. If the motorist violates the rule, a warning message will be sent to the motorist via audio and/or video interface and penalty will be calculated in the server and billed monthly to the vehicle owner [14].

Automatic Billing of Core Area / Toll

Charges Automatic toll collection and automatic ―core area charge‖ collections are also done using the same framework. Controller unit will be placed at toll-booth and along the motor able roads around the core area which will detect each individual vehicle uniquely within its zone by capturing their device ids and will keep records of the time during which the vehicle was seen by those Controllers within its reading zone. This information will be sent to a main server. Accordingly the main server will calculate the charges and raise bills against the vehicle ids [

Advantages of a Smart Traffic Management System

**Cleaner, greener, safer, and more accessible roads are a few benefits of implementing IoT and intelligent technology.**

It helps with the following:

* Reducing traffic jams and accidents on the streets
* Ensuring immediate clearance for emergency vehicles
* Facilitating safer and shorter commute times
* Reducing congestion & energy consumption at intersections
* Offering significant productivity benefits with real-time monitoring of crucial infrastructures
* Reducing operating costs with efficient traffic management processes
* Ensuring compliance with the regulations for reducing the carbon footprint
* Saving billions of gallons of fuel wasted every year
* Accurate tracking & quick recovery of lost and stolen vehicles

Disadvantage

* Very costly-

Area traffic control is a veryadvanced traffic control strategy it involvevery advanced technology and highly skilledpersons to operate the system to control thetraffic which makes it very costly.

* Very complex-

Area traffic control system isa very big system which includes manyunites in it like Vehicle Detectors,Intersection Controller, Communicationetwork, Application !oftware, Central"#egional$ Control !ystem. These unit is use to perform different%different task for thesystem.

•Suitable only for lane followingtraffic-

In area traffic control system we usevehicle detector to collect the data to find theactual flow and to get signal timing accordingto the present condition of traffic. Thesevehicle detectors detect the vehicle on thebasis of lane. 'or e&ample we are collectingdata for tow lane road then the detectors willable to detect the vehicle which will comefrom their respective lane and the vehiclewhich is using space other than these twolanes cannot be detected. !o data will not beaccurate. !o we can say that it will give bestresult only for lane following traffic.

Conclusion

Smart Traffic Management System has been developed by using multiple features of hardware components in IoT. Traffic optimization is achieved using IoTplatform for efficient utilizing allocating varying time to all traffic signal according to available vehicles count in road path.

The proposed work focuses on Smart Traffic management System using RFID which will eliminate the drawbacks of the existing system such as high implementation cost, dependency on the environmental conditions, etc. The proposed system aims at effective management of traffic congestion. It is also cost effective than the existing system. Furthermore, the study presents the problems in metropolitan areas all over the world caused by congestions and the related sources. Congestions developed to a problem, which affects economies worldwide. Particularly metropolitan areas are worst hit under these conditions. Congestions have a negative impact on the financial situation of a country, on the environment and hence the overall quality of life. The proposed system can be enhanced by using any other powerful communication network other than GSM.