Abstract *— Traffic is a major obstacle faced by all metropolitan cities; this is due to the exponential increase in the number of vehicles on the road but the infrastructure for road transportation remains the same. The average travelling time consumption by people is increasing year by year for the same given way points. Most of the cities still rely on conventional traffic signaling which is controlled manually or time based. This conventional system used around the world is not efficient as it lacks useful data from reliable real-time sources to clear the way for emergency vehicles (ambulances, firetrucks and police vehicles) during heavy traffic conditions. Manual interaction involves manual errors, which leads to more fuel consumption and leads to health issues. Due to this Traffic flowing to a junction from all the directions at a given time is unequal. A smart traffic management is a system, where traffic is controlled by the management system, which controls the traffic lights in accordance with the real time situation of traffic moving from all different directions in a junction. This real time data is collected either from google maps (future work) or from various sensors placed at equal intervals of distance at a junction. This data is collected and brought to a control system which autonomously calculates the optimum time for the release of the green signal.*

We are aiming to solve the issue of traffic by efficiently controlling the signaled intersections in cities by presenting present an algorithm based on comparative real time data analysis using IOT [2]. Considering this review, we identify a range of important possibilities for contributions to traffic management, detection technology and flexible optimization techniques that use various kinds of automated learning.

Keywords *— smart traffic, IOT, Emergency vehicle, autonomous management, google maps.*

I. 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 on 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 administration 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 interims 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 the 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 kilo meter 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

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 positive 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 micro volts per Gauss, so therefore, these devices are usually manufactured with built-in high gain amplifiers.

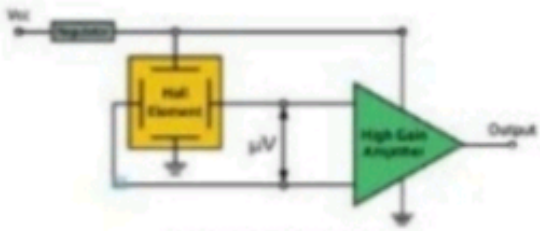

Fig.2-Hall effect sensor

There are two types of Hall Effect sensors, one providing analog and the other digital output. The 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 where as 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

Fig.3-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.

Despite the fact that the first aim of the Raspberry Pi venture was to PCs with programming choices under the control of understudies, 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. Advantages

•It is a solitary board PC

•It is very cost effective.

•A completely fledged working framework and it very well may be utilized as an everyday PC.

•The availability of GPIO (General Purpose Input Output pins) recognizes a Raspberry Pi from conventional PCs. These pins can be associated with sensors and other outside parts which help in interfacing with them automatically a language, for example, Python or Java. This enables you to manufacture and model or any Internet of Things gadgets that can detect this present reality

•Newer models have Wi-Fi and Bluetooth options worked in. This permit bringing ventures into remote mode effectively.

Light-Emitting Diode

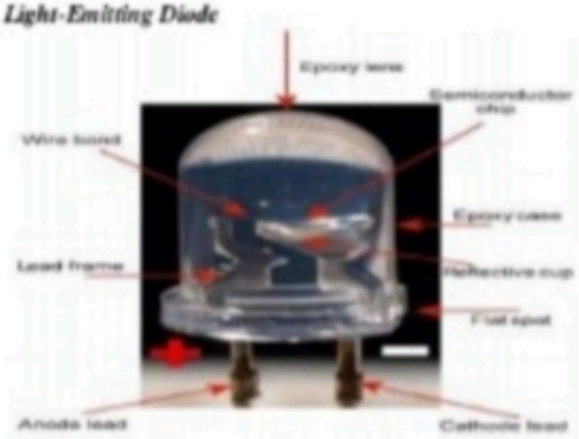

Fig. 4- Light emitting diode

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.

The shade of the light (comparing to the vitality obtained by the photons) is found by the vitality required for electrons to cross the vitality hole of the semiconductor. White light is gotten by utilizing a few different semiconductors. 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

In a light discharging diode, the recombination of electrons and electron openings in a semiconductor creates light (or infrared radiation). This procedure is designated "electroluminescence". The wavelength of the light is relied upon the hole of the vitality band in the kind of semiconductors utilized. Since these materials have a high list of refraction, plan highlights of the gadgets, for example, extraordinary optical coatings are important to emanate light effectively with low wastage.

III. METHODOLOGY

Basic flowchart[5]

- This is the basic ideology behind controlling the timing of traffic lights with respect to present time traffic conditions.
- 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 terminating previous loop and changing the signal immediately.

Fig. 5- Process flow chart

Complex flow chart

Fig. 6- data collection and priority line up

Proposed system

This traffic management control system consists of hall effect sensor, 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

when contrasted with the vast majority of different dialects and stages. Extensive Support Libraries: Python gives a tremendous default library which comprises territories like web conventions, string tasks, web administrations apparatuses and working framework (OS) interfaces. Open Source and Community Development: Python language is an open source programming language which implies that it is allowed to utilize and convey, including for business and business purposes.

User-accommodating Data Structures:

This programming language has a worked in rundown and word reference information structures which can be utilized to manufacture quick runtime information structures. Further it likewise gives the alternative of extremely spirited and a significant level information composing which confines the length of help code that is fundamental.

Snippet of 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("-----------------------------")
```