

IOT BASED SMART TRANSPORTATION SYSTEM

Submitted in partial fulfilment of the requirements for the award of
Bachelor of Engineering Degree in Electronics and Communication Engineering

By

VADAKOPPULA SATYANARAYANA (Reg. No. 39130490)

DEVARALA VENGALARAO YADAV (Reg. No. 39130110)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING**

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119

APRIL- 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **VADAKOPPULA SATYANARAYANA (Reg. no. 39130490)** and **DEVARALA VENGALARAO YADAV (Reg. no. 39130110)** who carried out the project entitled "**IOT BASED SMART TRANSPORTATION SYSTEM**" under my supervision from June 2022 to November 2022.

Internal Guide

Ms.T. Gomathi, M.Tech., (Ph.D).

Head of the Department

Dr. T. Ravi, M.E., Ph.D.

Submitted for Viva voce Examination held on 26.04.23

Internal Examiner

External Examiner

DECLARATION

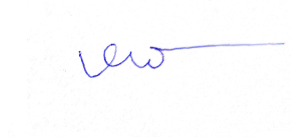
We, **VADAKOPPULA SATYANARAYANA (Reg. No. 39130490)** and **DEVARALA VENGALARAO YADAV(Reg. No. 39130110)** hereby declare that the Project Report entitled “IOT BASED SMART TRANSPORTATION SYSTEM” done by us under the guidance of **Ms.T.Gomathi,M.Tech.,(Ph.D).**, Assistant Professor, Department of Electronics and Communication Engineering, at SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in ELECTRONICS AND COMMUNICATION.

DATE: 26.04.23

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

1.



2.



ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey our thanks to **Dr. N.M. Nandhitha, M.E., Ph.D.**, Dean, School of Electrical and Electronics Engineering, and **Dr. T. RAVI, M.E., Ph.D.**, Head of the Department, Dept. of Electronics and Communication for providing us necessary support and details at the right time during the progressive reviews.

We would like to express my sincere and deep sense of gratitude to our Project Guide **Ms.T. Gomathi ,M.Tech.,(Ph.D).**, Assistant Professor, Department of Electronics and Communication Engineering, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication who were helpful in many ways for the completion of the project.

ABSTRACT

From the beginning of civilizations, transportation has been one of the most important requirements for humans. Over the years, it has been evolved to modern transportation systems such as road, train, and air transportation. With the development of technology, intelligent transportation systems have been enriched with Information and Communications Technology (ICT). Nowadays, smart city concept that integrates ICT and Internet-of-Things (IoT) have been appeared to optimize the efficiency of city operations and services. Recently, several IoT-based smart applications for smart cities have been developed. Among these applications, smart services for transportation are highly required to ease the issues especially regarding to road safety. In this context, this study presents a literature review that elaborates the existing IoT-based smart transportation systems especially in terms of road safety. In this way, the current state of IoT-based smart transportation systems for safer roads are provided. Then, the current research efforts undertaken by the authors to provide IoT-based safe smart traffic system are briefly introduced. It is emphasized that road safety can be improved using Vehicle-to-Infrastructure (V2I) communication technologies via the cloud (Infrastructure-to-Cloud — I2C). Therefore, it is believed that this study offers useful information to researchers for developing safer roads in smart cities.

TABLE OF CONTENTS

Chapter No	TITLE		Page No
	ABSTRACT		V
	LIST OF FIGURES		VIII
1	INTRODUCTION		1
2	LITERATURE SURVEY		
	2.1	Inferences from Literature Survey	3
	2.2	Open problems in Existing System	5
3	REQUIREMENT ANALYSIS		
	3.1	Feasibility Studies/Risk Analysis of the Project	6
	3.2	Hardware and Software requirements and Specification Documents	7
4	DESCRIPTION OF PROPOSED SYSTEM		
	4.1	Selected Methodology or Process model	19
	4.2	Architecture of overall design of proposed system	19
	4.3	Description of Software for Implementation and Testing plan of the Proposed system	20
	4.4	Project Management Plan	22
5	IMPLEMENTATION DETAILS		
	5.1	Development and Deployment	24
	5.2	Algorithms and Flowchart	26
6	RESULTS AND DISCUSSIONS		27
7	CONCLUSION		

	7.1	Conclusion	31
	7.2	Future work	32
	7.3	Research Issues	33
	7.4	Implementation Issues	35
	REFERENCES		37
	SOURCE CODE		39

LIST OF FIGURES

Figure No.	TITLE	Page No.
3.1	Voltage Regulator	7
3.2	Adaptor	8
3.3	Capacitor	9
3.4	Pin Diagram for Capacitor	10
3.5	Resistor	11
3.6	LED	11
3.7	LED Symbol	12
3.8	NODEMCU	12
3.9	Pin Diagram for NODEMCU	14
3.10	LCD	14
3.11	Pin Diagram for LCD	15
3.12	GPS Tracking System	17
3.13	Vehicle with Tracking System	18
4.1	Workflow Model of the Existing System	19
4.2	Connection between Arduino nano and PC	21
4.3	Selecting the board on Arduino IDE	21
4.4	Selecting the Port on Arduino IDE	22
4.5	Block Diagram	23
5.1	Connections of the Circuit	24
5.2	GPS Connection	25
6.1	Final Output	27
6.2	Displaying the output in LCD	28

CHAPTER-1

INTRODUCTION

In current days, the safety of private and civic vehicles is the main cause for anxiety. A tracking system can be installed in a vehicle to confirm safety while traveling. Besides, policemen can track the signal released by the tracking system to locate a stolen vehicle. The Telegraph informed that the rate of stolen vehicle was 300,000 in 2002, but in 2014 it fell under 100,000 due to implementation of security systems (Rahman et al. 2016). School transport is one of the most important transportations that needs safety in its top priorities. There are millions of children riding the school bus in European Union and 2 United States, as well as in many cities all over the world daily.

However, children (especially elementary school children) are vulnerable and cannot defend or protect themselves from any attack (Zambia et al. 2015). According to a recent American statistic, from 2003 to 2012, 119 school pedestrians in United States with age under 19 years have died in accidents involving school buses; 65% were crushed through school transportation, 5% working as school transport automobiles and 30% by other vehicles elaborated in accidents (Fatalities and Group 2014). The objective of Intelligent Transportation Systems (ITS) is to create a smart and secure transportation system, taking into account all stakeholders, including government establishments, parents, as well a's university (Tomas et al. 2013). Vehicle tracking by using satellite positioning systems, such as Global Positioning System (GPS) or the Russia's version of GPS called (GLONASS), it can be highly beneficial, especially during traffic jams or adverse weather environments (Tomas et al. 2013) [5].

Today, GPS finds wide applicability in a number of applications, commonly in the navigation area, tracking etc. (Kumar n.d.) (Academy 2010). Assisted GPS (A-GPS) systems help in increasing the sensitivity of the GPS reception and decreasing time-to-first-fix (TTFF). The central idea behind A-GPS is to deliver assistance information to the GPS receiver over a wireless network. Using this information, moving devices (like mobiles, vehicles) equipped with A-GPS can calculate

locations faster and in much more challenging situations. An estimate of the location will be calculated at a server by using the feedback information received from the A-GPS receiver. Based on where the position calculations are conducted, A-GPS methods are categorized as either mobile-based or network based (Alliance 2007).

Nowadays, there exist several systems that are used to track the location of school bus. In (Dinkar and Shaikh 2011), the authors demonstrate the progress of an Internet of Things (IoT) based bus monitoring system. They show that, when the localization and speed of the sensors are adjusted, the system helps many stakeholders (parents, the government, the school and many other authorities) save real time path of the school bus performance, which result in a better controlled bus. In (Saranya et al.2013), a tracking system with a child module is proposed that is to be carried by children. The system transmits the tracking data to a database and a mobile device.

This system has the inherent disadvantage that it may not be appropriate for children to carry the module due to security reasons and that broad-scale distribution of the modules may be quite expensive. The work presented in (Mori et al. 2011) investigates a tracking system that employs android terminals.

CHAPTER-2

LITERATURE SURVEY

2.1 INFERENCES FROM LITERATURE SURVEY

Tracking systems were first developed for the shipping industry because they wanted to determine where each vehicle was at any given time. Passive systems were developed in the beginning to fulfill these requirements. For the applications which require real time location information of the vehicle, these systems can't be employed because they save the location information in the internal storage and location information can only be accessed when vehicle is available. To achieve Automatic Vehicle Location system that can transmit the location information in real time, Active systems are developed. Real time vehicular tracking system incorporates a hardware device installed in the vehicle (In-Vehicle Unit) and a remote Tracking server. The information is transmitted to Tracking server using GSM/GPRS modem on GSM network by using SMS or using direct TCP/IP connection with Tracking server through GPRS. Tracking server also has GSM/GPRS modem that receives vehicle location information via GSM network and stores this information in database. This information is available to authorized users of the system via website over the internet.

Authors: Prathmesh Chavan, Ganesh shedage, Harshalal Vyavahare, Akash Kharat

Global Positioning Systems (GPS) are used for various applications. This paper mainly focuses on developing advancement in GPS based vehicle navigation system using Android apps. Now a day management of buses is one of the main problems with college transport. Their management involves a lot of problems to the staff. Moreover, while dealing with the management of college transport, one should also keep in mind the cost of operation. This paper gives a possible, efficient and low-cost solution to the above problem. This system is helps to locate the routes of the buses by which they are travelling and displaying the current position of bus on Android apps so that it helps in tracking the bus time.

Authors: SeokJu Lee, Girma Tewolde, Jaerock Kwon An efficient vehicle tracking system is designed and implemented for tracking the movement of any equipped vehicle from any location at any time. The proposed system made good use of a popular technology that combines a Smartphone application with a microcontroller. This will be easy to make and inexpensive compared to others. The designed in-vehicle device works using

Global Positioning System (GPS) and Global system for mobile communication / General Packet Radio Service (GSM/GPRS) technology that is one of the most common ways for vehicle tracking. The device is embedded inside a vehicle whose position is to be determined and tracked in real-time. A microcontroller is used to control the GPS and GSM/GPRS modules. The vehicle tracking system uses the GPS module to get geographic coordinates at regular time intervals. The GSM/GPRS module is used to transmit and update the vehicle location to a database. A Smartphone application is also developed for continuously monitoring the vehicle location. The Google Maps API is used to display the vehicle on the map in the Smartphone application. Thus, users will be able to continuously monitor a moving vehicle on demand using the Smartphone application and determine the estimated distance and time for the vehicle to arrive at a given destination. In order to show the feasibility and effectiveness of the system, this paper presents experimental results of the vehicle tracking system and some experiences on practical implementation.

Authors: N. Hounsell B. Shrestha, C. D'Souza

Automatic Vehicle Location (AVL) systems are being used increasingly in public transport operations in towns and cities in the UK, providing applications including real time passenger information, bus fleet management and bus priority at traffic signals. AVL systems also usually provide a wealth of information on bus operational performance which can be used off-line for a range of analytical purposes, such as for improved scheduling or the identification of problem locations in the network. This paper discusses the use of AVL data for performance monitoring through a Case study of the iBus system in London. iBus is one of the largest bus-based AVL systems in the world, covering 8000 buses on some 700 bus routes, with 2500 bus stops showing real-time bus arrival times. One of its applications is bus priority at traffic signals, with facilities currently operational at

some 2000 traffic signal installations. This paper concludes with a discussion of the limitations and statistical considerations needed with data of this sort as well as its advantages

2.2 OPEN PROBLEM IN EXISTING SYSTEM

2.2.1. The Product is not for Real time purpose

Traffic Control System for Emergency Vehicles which is in the existing methodology is generated in the software simulation platform called LabView. Which is not more effective for the real time application. Yet it's not as simple as it looks to put this innovative technology into practice. While making a traffic control system, developer must deal with several issues. The price of creating a traffic control system is expensive. The process might take a while and the necessary supplies and equipment are not appropriate in this existing method.

2.2.2. The Microcontrollers and Processors used are High End and Complex

Programming microcontrollers or microprocessors is challenging for a variety of reasons. Secondly, it might be challenging to know how to configure them because they frequently include a lot of peripherals and I/O pins. Second, it might be challenging to decide which one to employ for a given activity because they frequently offer a wide variety of skills and features. Ultimately, they frequently need some programming knowledge in order to reach their full potential. As a result, many non-programmers find it challenging to begin working with microcontrollers or microprocessors. One of the most significant problems with the current approach to using traffic control system is that the sensors have not been used correctly.

2.2.3. Unrealistic and Adaptable

The current approach of using a traffic control system has a number of issues that prevent it from being adaptable or workable. The existing method, for starters, necessitates the purchase of pricey gear, including an LCD screen. Second, installing and setting up the software needed to run a traffic control system is challenging.

CHAPTER-3

REQUIREMENT ANALYSIS

3.1 FEASABILITY STUDIES/RISK ANALYSIS OF THE PROJECT

- **Technical Feasibility:** Assess the technical feasibility of implementing IoT-based smart transportation systems, including the availability and reliability of IoT infrastructure, communication networks, sensors, and devices. Evaluate the compatibility and interoperability of different systems and technologies, and identify any technical challenges or limitations that may affect the implementation.
- **Economic Feasibility:** Evaluate the economic feasibility of implementing IoT-based smart transportation systems, including the cost of deploying and maintaining IoT infrastructure, sensors, devices, and communication networks. Assess the potential return on investment (ROI) and cost-benefit analysis, considering factors such as operational cost savings, revenue generation opportunities, and potential funding sources.
- **Operational Feasibility:** Assess the operational feasibility of implementing IoT-based smart transportation systems, including the impact on existing transportation operations, processes, and workflows. Evaluate the ability to integrate IoT technologies into current systems and processes, and identify any operational challenges, risks, or disruptions that may arise during implementation.
- **Financial and Economic Risks:** Assess the financial and economic risks associated with implementing IoT-based smart transportation systems, including potential cost overruns, funding uncertainties, revenue generation challenges, and economic viability of the project. Consider potential risks to the financial sustainability of the system and assess the measures in place to manage these risks.
- **User Acceptance Risks:** Assess the risks associated with user acceptance and adoption of IoT-based smart transportation systems. Consider potential resistance to change, user behavior changes, and user satisfaction, and assess the measures in place to promote user acceptance and adoption of the system.

3.2 HARDWARE AND SOFTWARE REQUIREMENTS AND SPECIFICATION DOCUMENTS

3.2.1 VOLTAGE REGULATOR

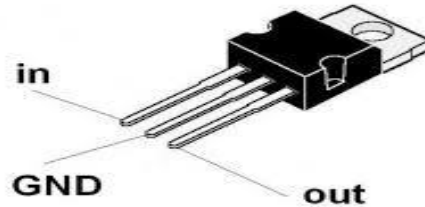


Fig.3.1 Voltage Regulator

The function of a voltage regulator (as shown fig :3.1) is to maintain a constant DC voltage at the output irrespective of voltage fluctuations at the input and (or) variations in the load current. In other words, voltage regulator produces a regulated DC output voltage.

Voltage regulators are also available in Integrated Circuits (IC) forms. These are called as voltage regulator ICs.

I. *FIXED VOLTAGE REGULATOR*

A fixed voltage regulator produces a fixed DC output voltage, which is either positive or negative. In other words, some fixed voltage regulators produce positive fixed DC voltage values, while others produce negative fixed DC voltage values.

78xx voltage regulator ICs produce positive fixed DC voltage values, whereas, 79xx voltage regulator ICs produce negative fixed DC voltage values.

The following points are to be noted while working with 78xx and 79xx voltage regulator ICs –

- “xx” corresponds to a two-digit number and represents the amount (magnitude) of voltage that voltage regulator IC produces.
- Both 78xx and 79xx voltage regulator ICs have 3 pins each and the third pin is used for collecting the output from them.
- The purpose of the first and second pins of these two types of ICs is different

–

- The first and second pins of 78xx voltage regulator ICs are used for connecting the input and ground respectively.
- The first and second pins of 79xx voltage regulator ICs are used for connecting the ground and input respectively.

3.2.2 ADAPTOR



Fig.3.2 Adaptor

12 Volt 1 Amp Power Adapter(as shown in fig:3.2) takes an AC INPUT of 100-240V and gives 12V 1A DC output

12 Volt DC 1 Amp power supply is suitable for powering a wide range of applications including CCTV cameras, wireless routers, Robotics and DIY kits.

Specifications: -

- Input - 100-240 VAC 50/60Hz
- Category - Switch Mode Power Adaptor (SMPS)
- Output Type - DC
- Output - 12Volts 1Amp

3.2.3 CAPACITOR



Fig.3.3 Capacitor

An electrolytic capacitor (as shown in fig:3.3) definition is, it is a polarized capacitor whose anode has a higher or more positive voltage than the Cathode. As the name suggests it is a polarized capacitor and an electrolytic capacitor function is, it uses an electrolyte to operate with a higher or more positive voltage on the anode than the cathode. Therefore, the anode terminal is denoted with a positive sign, while the cathode with a negative sign. Applying a reverse polarity voltage of 1 to 1.5 volts may destroy the capacitor and dielectric and the result is hazardous, leading to an explosion or fire.

An electrolytic capacitor uses an electrolyte, in the form of solid, liquid or gel – serves as cathode or negative plate to achieve much higher capacitance per unit volume. On the other hand, a positive plate or anode made of metal acts as an insulating oxide layer formed through anodization. This allows an oxide layer to work as the dielectric of the capacitor.



Fig.3.4 Capacitors pin diagram

3.2.4 RESISTOR

The term resistor (as shown in fig:3.5) refers to a device that acts as a two-terminal passive electrical component that is used to limit or regulate the flow of electric current in electrical circuits. And it also allows us to introduce a controlled amount of resistance into an electrical circuit. The most important and commonly used components in an electronic circuit are resistors.

A resistor's main job is to reduce current flow and lower voltage in a specific section of the circuit. It's made up of copper wires that are wrapped around a ceramic rod and coated with insulating paint.

The basic idea is known to all about how electricity flows through an electronic circuit. Here, two categories can be identified which are conductors and insulators. Insulators do not allow the flow of electrons, but the conductor does. However, the resistor determines the amount of electricity that is allowed to pass through them. The total voltage passes through when it is passed through a conductor like the metal; by introducing the resistors, the amount of voltage and current can be controlled.

The ease at which the electrons will allow the electricity to flow through it is known as resistance.

An insulator has better resistance than the conductor, and the term resistance is defined as the electrical quantity used by the resistor to control the flow of electrons.



Fig.3.5 Resistor

3.2.5 LED

A light-emitting diode (LED) (as shown in fig:3.6) is a semiconductor device that emits light when an electric current flows through it. When current passes through an LED, the electrons recombine with holes emitting light in the process. LEDs allow the current to flow in the forward direction and blocks the current in the reverse direction

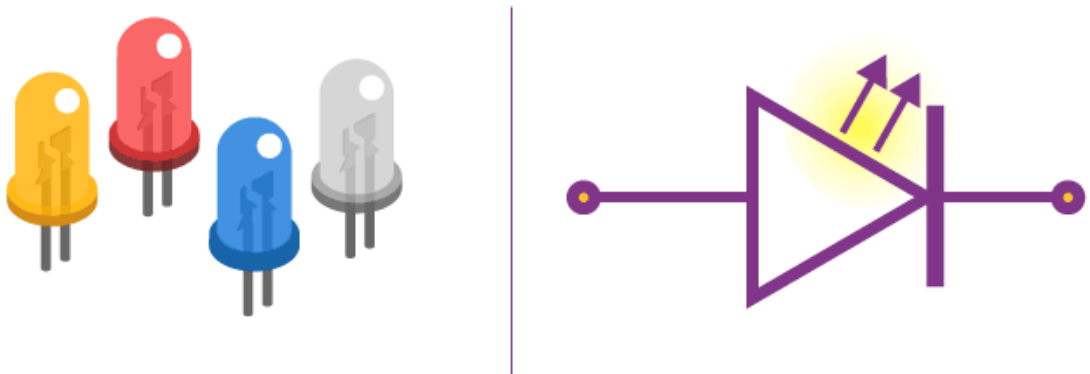


Fig.3.6 LED

Light-emitting diodes are heavily doped p-n junctions. Based on the semiconductor material used and the amount of doping, an LED will emit coloured light at a

particular spectral wavelength when forward biased. As shown in the figure, an LED is encapsulated with a transparent cover so that emitted light can come out.

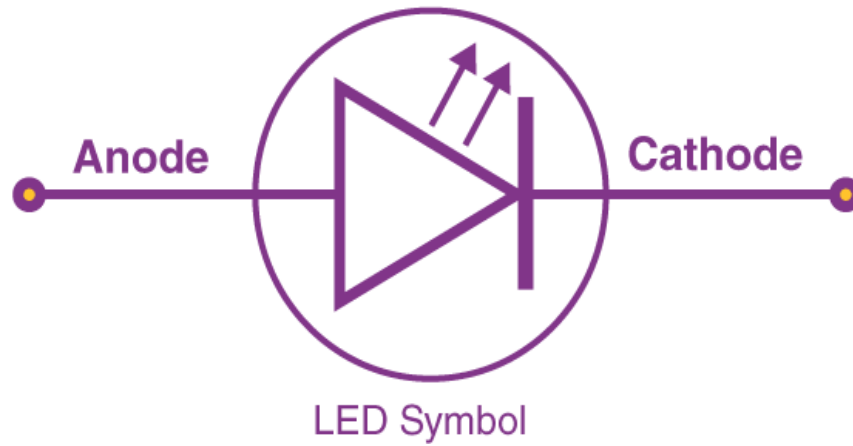


Fig.3.7 LED Symbol

3.2.6 NODEMCU



Fig.3.8 Nodemcu

The NodeMCU (as shown in fig:3.8) (Node MicroController Unit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and

manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK. That makes it an excellent choice for Internet of Things (IoT) projects of all kinds.

However, as a chip, the ESP8266 is also hard to access and use. You must solder wires, with the appropriate analog voltage, to its pins for the simplest tasks such as powering it on or sending a keystroke to the “computer” on the chip. You also have to program it in low-level machine instructions that can be interpreted by the chip hardware. This level of integration is not a problem using the ESP8266 as an embedded controller chip in mass-produced electronics. It is a huge burden for hobbyists, hackers, or students who want to experiment with it in their own IoT projects.

But, what about Arduino? The Arduino project created an open-source hardware design and software SDK for their versatile IoT controller. Similar to NodeMCU, the Arduino hardware is a microcontroller board with a USB connector, LED lights, and standard data pins. It also defines standard interfaces to interact with sensors or other boards. But unlike NodeMCU, the Arduino board can have different types of CPU chips (typically an ARM or Intel x86 chip) with memory chips, and a variety of programming environments. There is an Arduino reference design for the ESP8266 chip as well. However, the flexibility of Arduino also means significant variations across different vendors. For example, most Arduino boards do not have WIFI capabilities, and some even have a serial data port instead of a USB port.

I. SPECIFICATIONS

- Microcontroller: Ten silica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB

- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT project

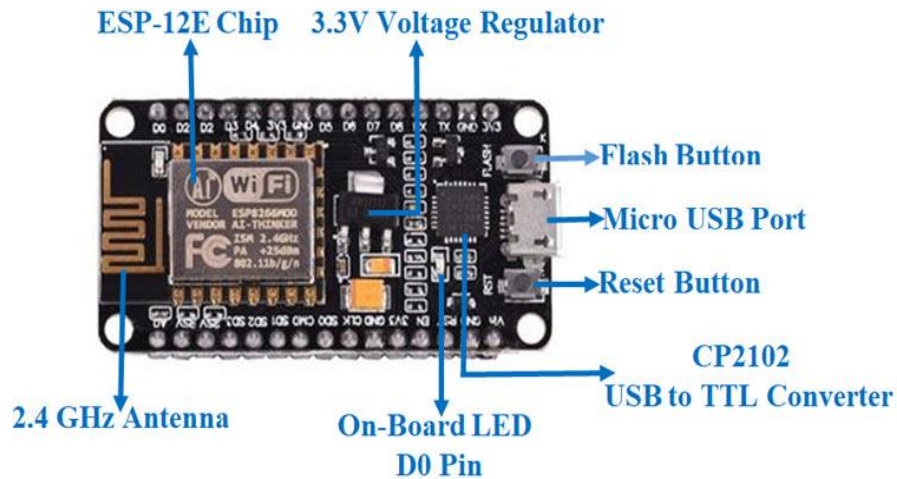


Fig.3.9 pin diagram for nodemcu

3.2.7 LCD

Liquid Crystal Display (LCD) is widely used in various electronics applications. It is commonly used in various systems to show different status and parameters. LCD16x2 has 2 lines with 16 characters in each line. Each character is made up of a 5x8 (column x row) pixel matrix



Fig.3.10 LCD

This LCD has two registers, namely, Command and Data. Command registers stores various commands given to the display. Data register stores data to be displayed. The process of controlling the display involves putting the data that form the image of what you want to display into the data registers, then putting instructions in the instruction register. In your Arduino project LCD simplifies this for you so you don't need to know the low-level instructions. Contrast of the display can be adjusted by adjusting the potentiometer to be connected across VEE pin.

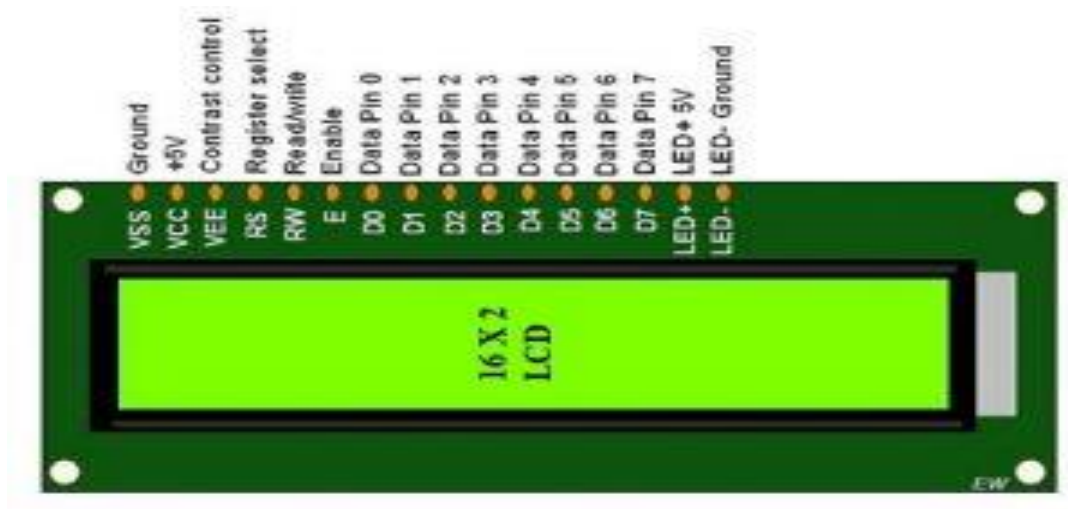


Fig.3.11 Pin Diagram for LCD

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.

- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.

I. APPLICATIONS

- The liquid crystal displays (LCDs) are used in aircraft cockpit displays.
- It is used as a display screen in calculators.
- For displaying images used in digital cameras.
- The television is main applications of LCD.
- Mostly the computer monitor is made up of LCDs.
- It is used in instruments panel where all the lab instruments use LCD screens for display.
- The LCDs are commonly used in all the digital wrist watches for displaying time

3.2.8 GPS MODULE

The Global Positioning System (GPS) is a satellite-based navigation system which was developed by the U.S. Department of Defense (DoD) in the early 1970s. Initially, GPS was developed as a military system to fulfill U.S. military needs. However, it was later made available to civilians, and is now a dual-use system that can be accessed by both military and civilian users.

GPS provides continuous positioning and timing information, anywhere in the world under any weather conditions. Because it serves an unlimited number of users beside being used for security reasons, GPS is a one-way-ranging (passive) system. That's users can only receive the satellite signals. This chapter introduces the GPS system, its components, and its basic idea.

GPS nominally consists of constellation of 24 operational satellites. This constellation known as the initial operational capability (IOC), was completed in July

1993. The official IOC announcement, however, was made on December 8, 1993. To ensure continuous worldwide GPS coverage.

The main functionality of a GPS-based tracking system comes from the use of the Global Navigation Satellite System (GNSS) network. This network of satellites emits microwave signals which are sent through to a variety of GPS devices. These devices can be found in vehicles or smartphones themselves for example. The information that is broadcast from the vehicle are things like location, vehicle speed, direction, etc.

In regards to vehicle tracking, or vehicle communication technology.

In this article, we'll look at what each of these four components does, how they work together, and ultimately the data GPS provides. Once you know and understand the process, you can begin to think about ways in which you can take advantage of GPS to enhance your fleet operations.

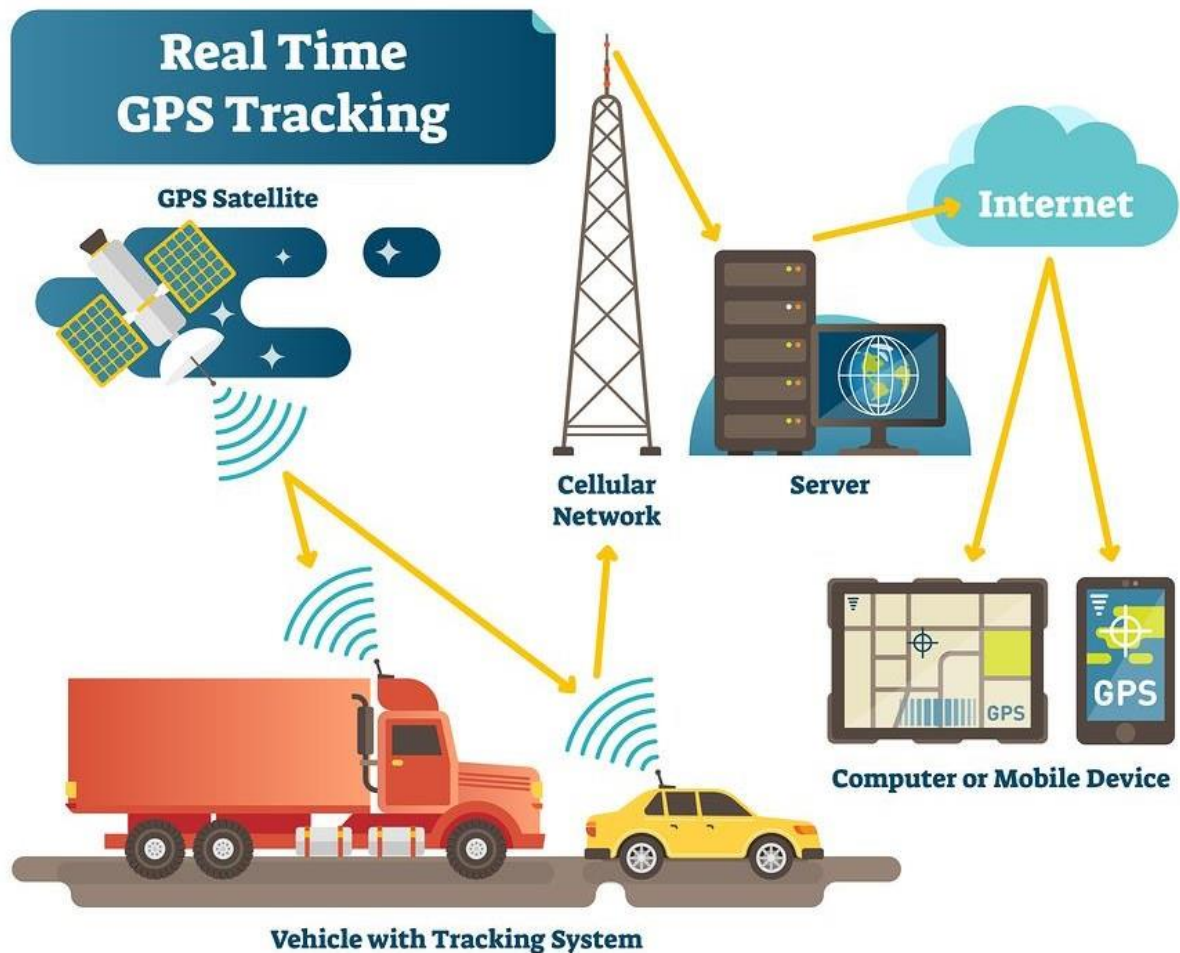


Fig.3.12 GPS Tracking System

The tracking device is installed into a vehicle (or piece of equipment or asset) to gather all sorts of information including speed, idle time, diagnostics, etc. It uses Global Positioning Systems (GPS satellites) to know the vehicles or equipment's location at all times. The information that is gathered from the vehicle is then stored on the device inside. The data is then transmitted by using a wireless, or cellular network through providers like AT&T and Verizon. It travels over one of these

cellular networks back to a server. The server acts as the “cloud” that allows you to access the information no matter where you are while using computer, tablet, or smartphone. The most basic information that is reported from the vehicle is real-time location tracking of your vehicles and equipment. The location data is displayed on a map in near real-time. Providers then offer a myriad of ways in which data can be accessed such as breadcrumb trails of where the vehicle has traveled, how fast it traveled on the way to a location, and more. All the complexity of gathering the data is done in the background, so all you need to worry about is logging in to setup automated reports and alerts of the information that is important and relevant to you.

Fig.3.13 Vehicle with Tracking system



CHAPTER 4

DISCRIPTION OF THE PROPOSED SYSTEM

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

The methodology for developing an IoT-based smart transportation system.

Select IoT Devices and Sensors: Choose the appropriate IoT devices and sensors to collect data on the transportation system. These could include GPS sensors.

Determine the Data Collection Method: Decide on the method of data collection and transmission, such as using a wireless network or satellite communication.

Design a data analytics platform to process and analyze the data collected from the sensors. This platform could include machine learning algorithms, predictive analytics, and data visualization tools.

Test the IoT-based smart transportation system to ensure that it functions correctly and meets the needs of the transportation system.

In this project we are going to track objects, by manipulating NODEMCU that collects data from the GPS model using the Global Navigation Satellite System, by getting these data from the web server, the received data from the NODEMCU and show the track process on the desktop, smart mobile or any other internet depending device.

4.2 ARCHITECTURE OR OVERALL DESIGN OF PROPOSED SYSTEM

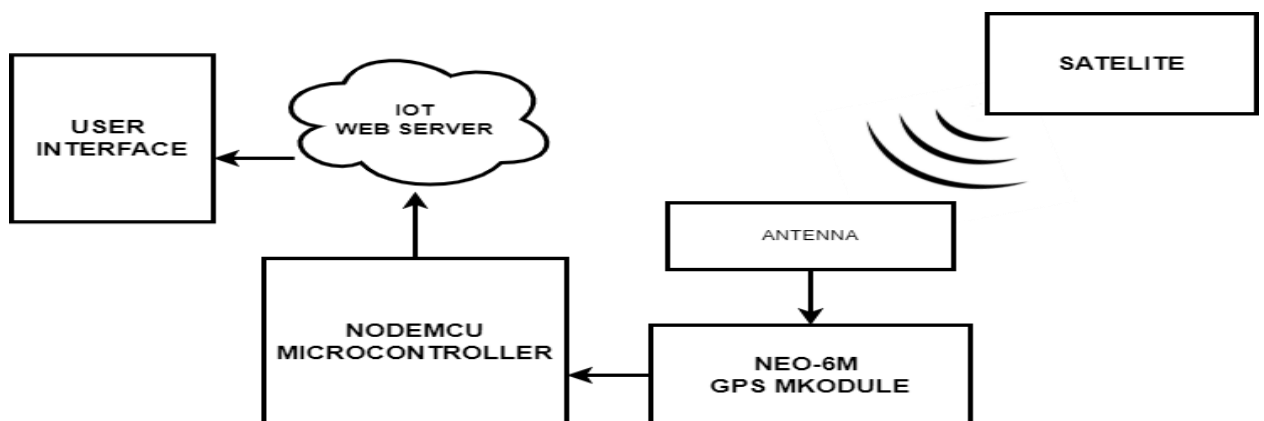


Fig.4.1 Workflow model of the existing system

These devices and sensors are used to collect data on various aspects of the transportation system, including vehicle location, speed, fuel consumption, and environmental conditions

A communication network is used to transmit the data collected by the IoT devices and sensors to the data analytics platform. This network could be a wireless network, cellular network, or satellite communication. By combining these components, an IoT-based smart transportation system can be developed that can help improve the efficiency, safety, and sustainability of the transportation infrastructure while reducing costs and improving the overall experience for users.

4.3 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED SYSTEM

The Arduino Integrated Development Environment or Arduino Software (IDE) contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension (.ino). The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor. The development of electronics is now easier thanks to Arduino software (IDE), and Arduino boards (hardware). This set help to build digital and interactive devices with the help of other components.

The Arduino software (IDE) is an open-source software, which is used to programme the Arduino boards, and is an integrated development environment, developed by arduino.cc. Allow to write and upload code to Arduino boards. And it consists of many libraries and a set of examples of mini projects. Arduino software (IDE) is compatible with different operating systems (Windows, Linux, Mac OS X), and supports the programming languages (C/C++) The Arduino software is easy to use for beginners, or advanced users. It uses to get started with electronics programming and robotics, and build interactive prototypes. So, Arduino software is a tool to develop new things. and create new electronic projects, by Anyone (children, hobbyists, engineers, programmers, ... etc).

- Connect your Arduino board into your computer by using the USB type a to bcable.

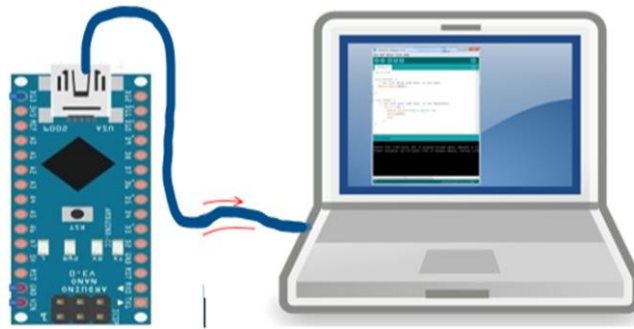


Fig.4.2 Connection between Arduino Nano and PC

- Open the Arduino IDE and go to Tools > Board > then select the Arduino Nano.

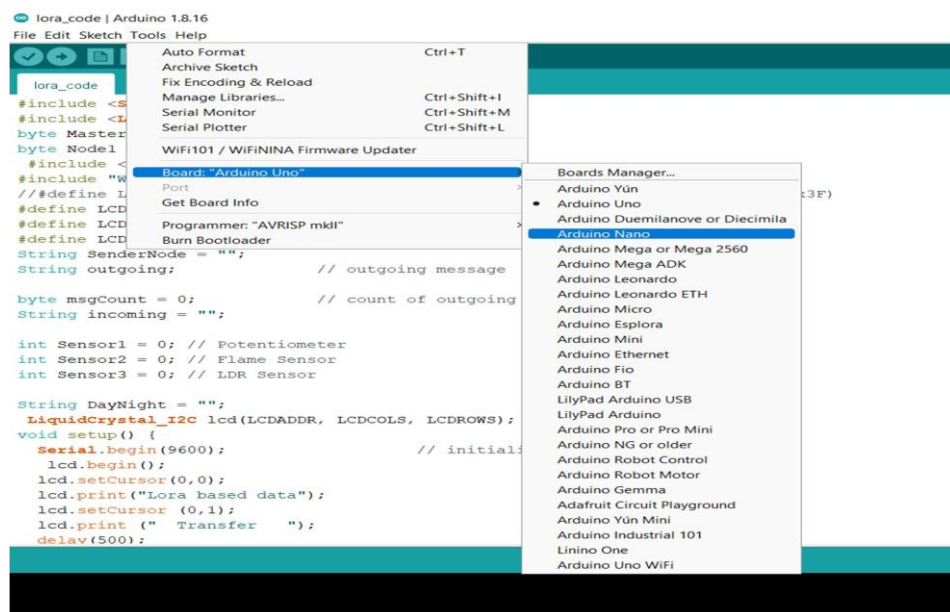


Fig.4.3 Selecting the board on Arduino IDE

- Now Open the Arduino IDE and go to Tools > port > then select the COM8(Arduino Nano).

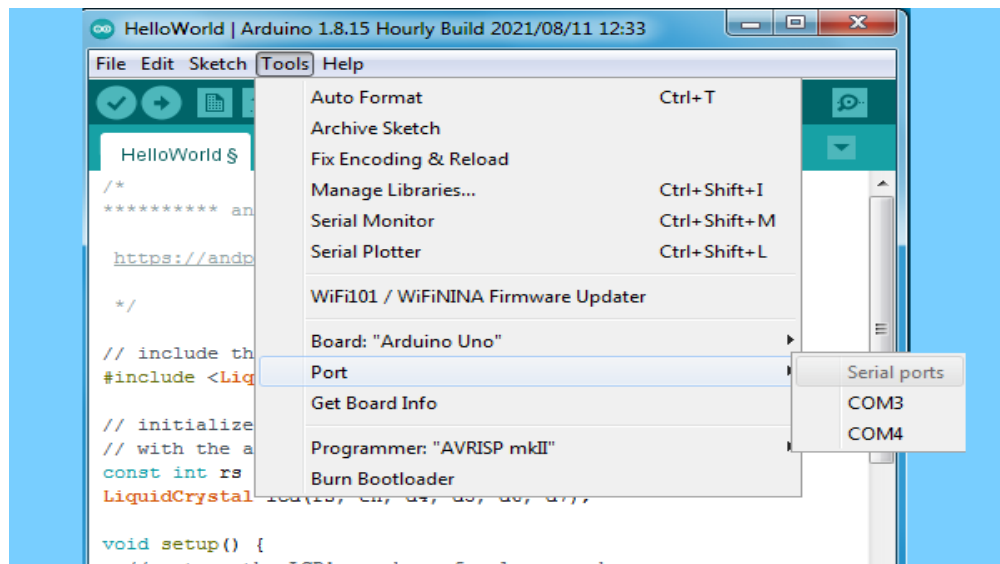


Fig.4.4 Selecting the port on Arduino IDE

- Now, write program on Arduino IDE.

Step 3: Testing

- After the Arduino IDE finishes the programming, you can now test the project. Place the sensor in water this should activate the buzzer, LED should turn ON and gate should be opened automatically.
- Now remove the sensor from water LED and buzzer should get turned off and also the dam gate should be closed automatically.
- If nothing is working redo the above steps and make sure the steps are followed correctly.

4.4 PROJECT MANAGEMENT PLAN

Developing a project management plan for an IoT-based smart transportation system requires a comprehensive approach that takes into account the unique aspects of the project. Here are the key components of a project management plan for an IoT-based smart transportation system:

- **Project objectives:** Clearly define the project objectives, including the scope, deliverables, and success criteria.
- **Project schedule:** Develop a detailed project schedule that includes all project activities, milestones, and deadlines.
- **Resource allocation:** Identify the resources required for the project, including

personnel, equipment, and budget.

- Risk management plan: Develop a plan to identify, assess, and mitigate potential risks associated with the project.
- Quality management plan: Develop a plan to ensure that the project deliverables meet the required quality standards.
- Communication plan: Develop a plan to ensure effective communication among all stakeholders involved in the project.
- Change management plan: Develop a plan to manage changes to the project scope, schedule, and budget.
- Project monitoring and evaluation plan: Develop a plan to monitor and evaluate project progress and ensure that it remains aligned with the project objectives.
- Project closure plan: Develop a plan to close the project and ensure that all project deliverables have been met.

Overall, a project management plan for an IoT-based smart transportation system should be comprehensive, flexible, and responsive to changing project needs. It should also be regularly reviewed and updated throughout the project lifecycle to ensure that it remains aligned with the project objectives

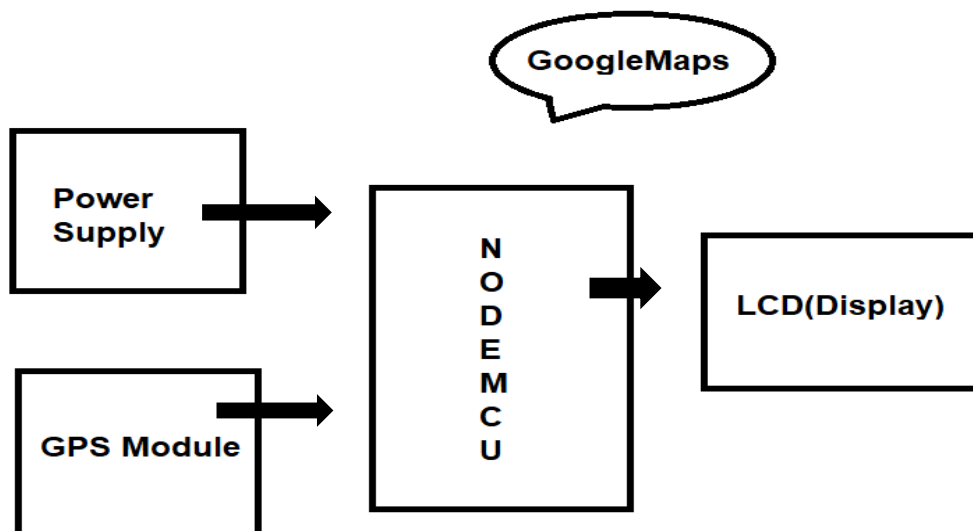


Fig.4.5 Block Diagram

CHAPTER-5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT

The GPS module will be powered with 3.3V power supply. It communicates with the ESP32 via serial communication, using the TX and RX pins available on the 4pins header. The following image shows the schematic diagram of the circuit used in this tutorial.

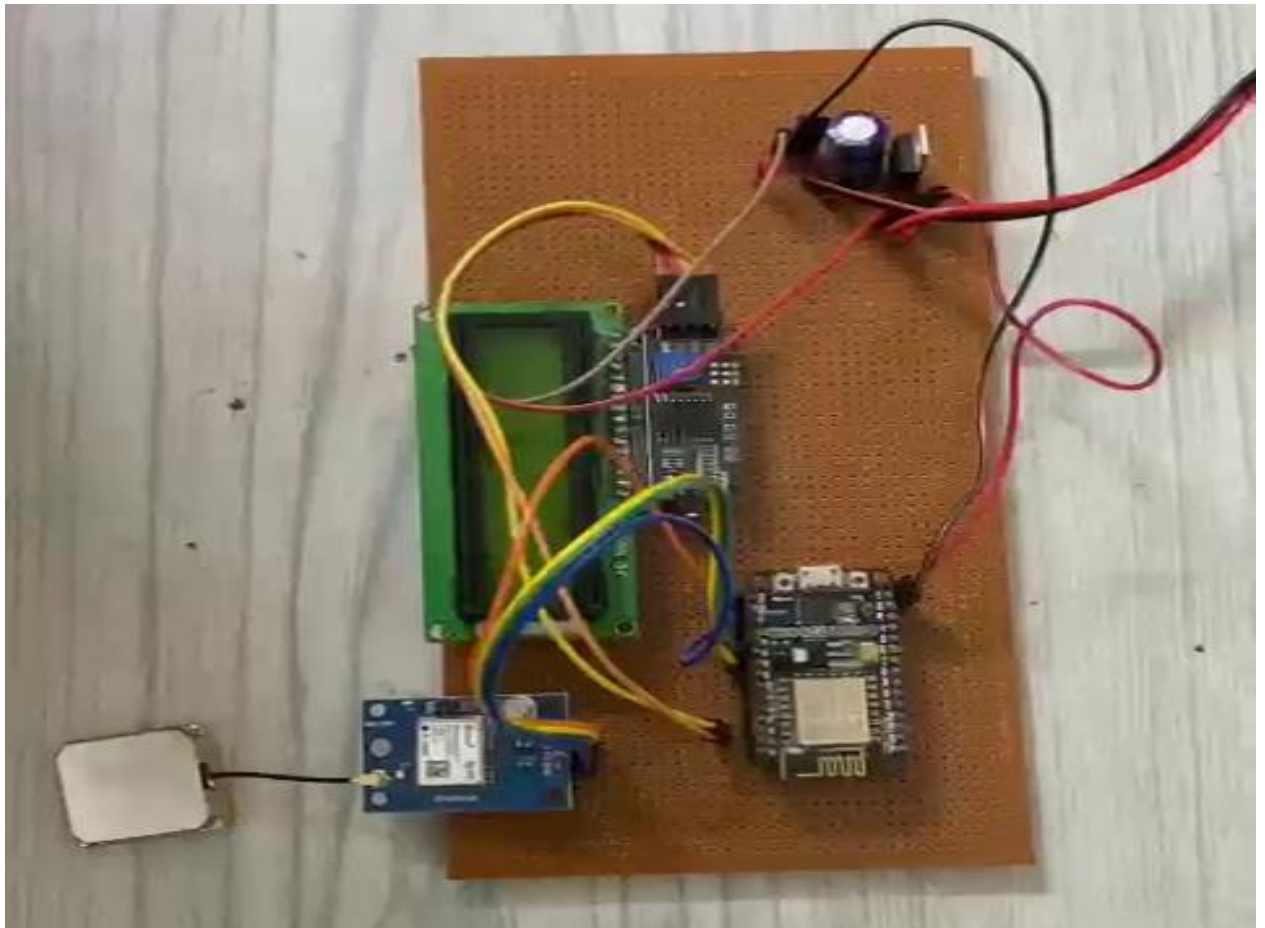


Fig.5.1 Connections of the circuit

- Connect the VCC pin of the GPS module to the ESP32 3.3V pin.
- Connect the GPS ground pin to the ESP32 ground pin.
- Now, connect the RX pin of the GPS module to the D4 pin of the ESP32. Similarly, connect the TX pin of the GPS module to the D3 pin of the ESP32 (D4 and D3 pins of the ESP32 are defined in the software).
- Last, connect your ESP32 to the computer through a USB cable. The ESP32 will be powered from the USB 5v.

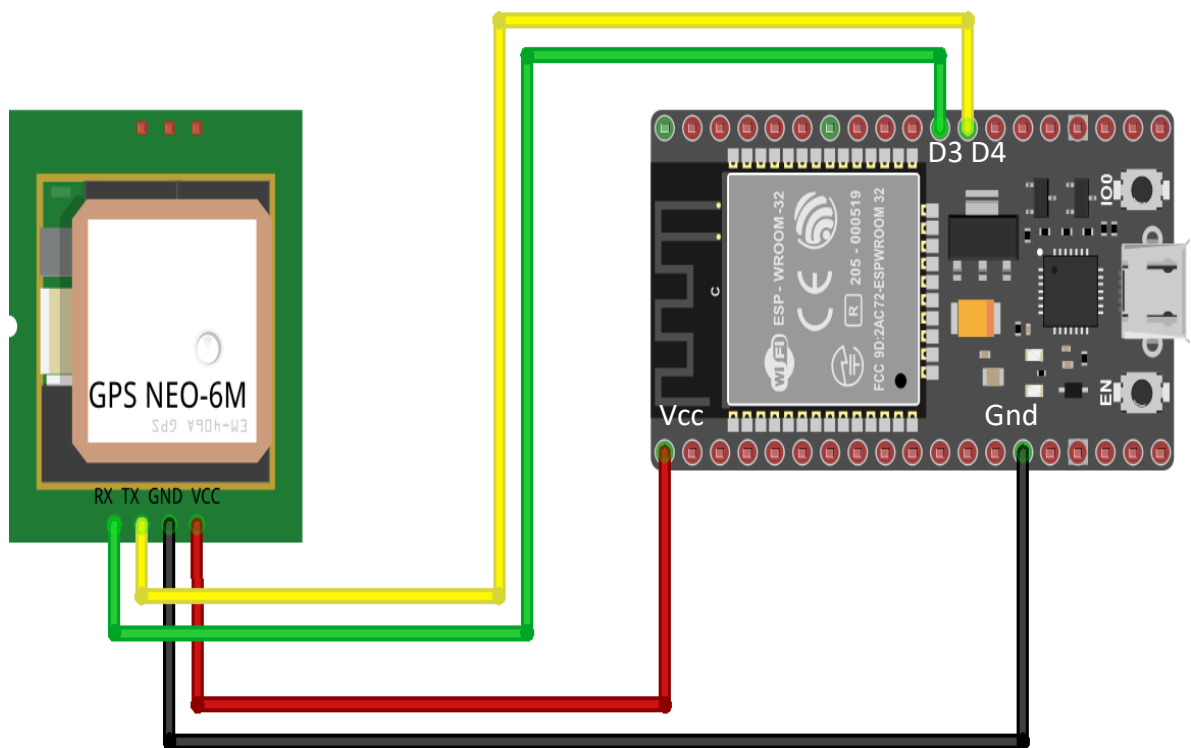
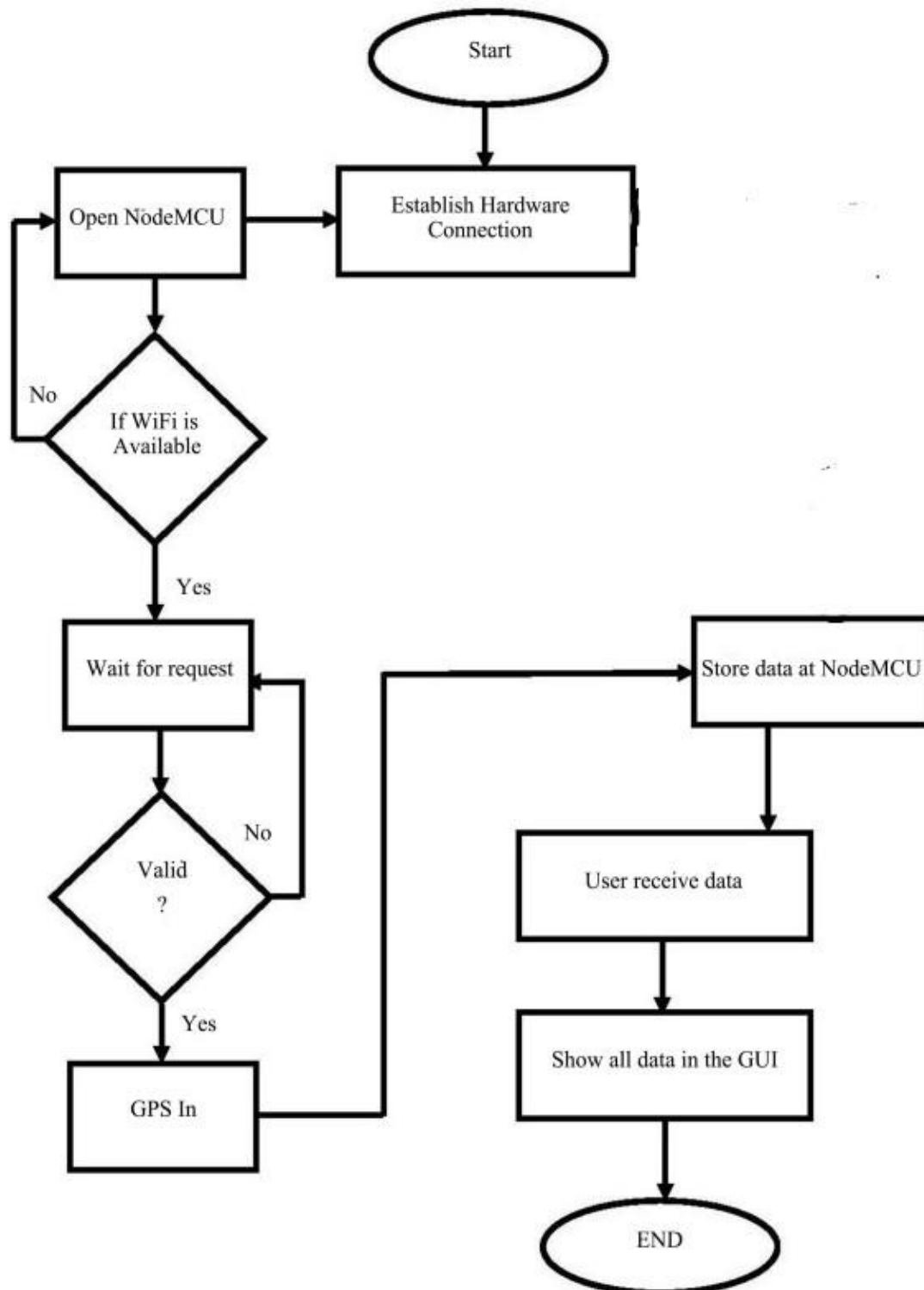


Fig 5.2 GPS Connection

5.2 ALGORITHMS AND FLOWCHART



CHAPTER-6

RESULTS AND DISCUSSIONS

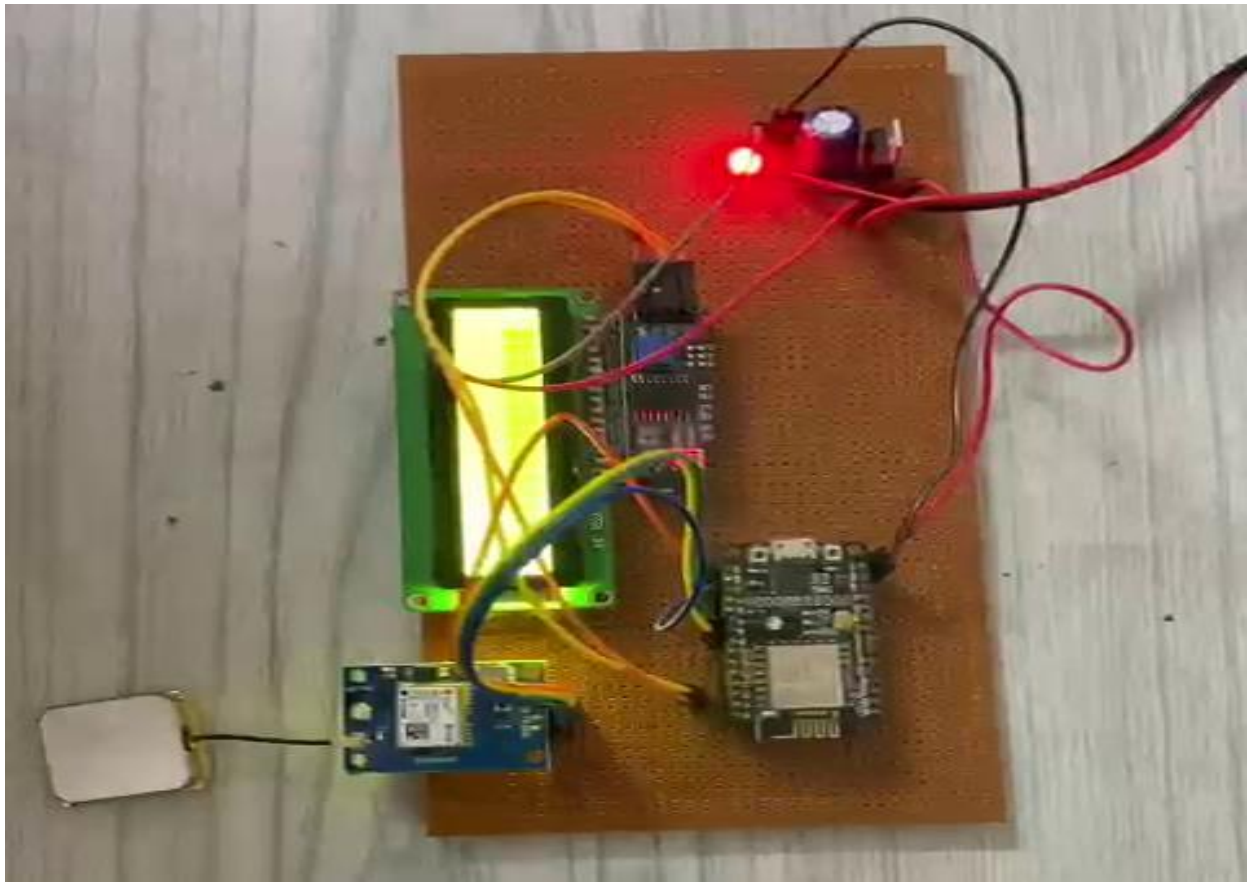


Fig 6.1 The Final Output

This is the final circuit that has been connected according to the circuit diagram and the main components that has been connected in this are GPS , Ardunio, PCB these connected to LCD

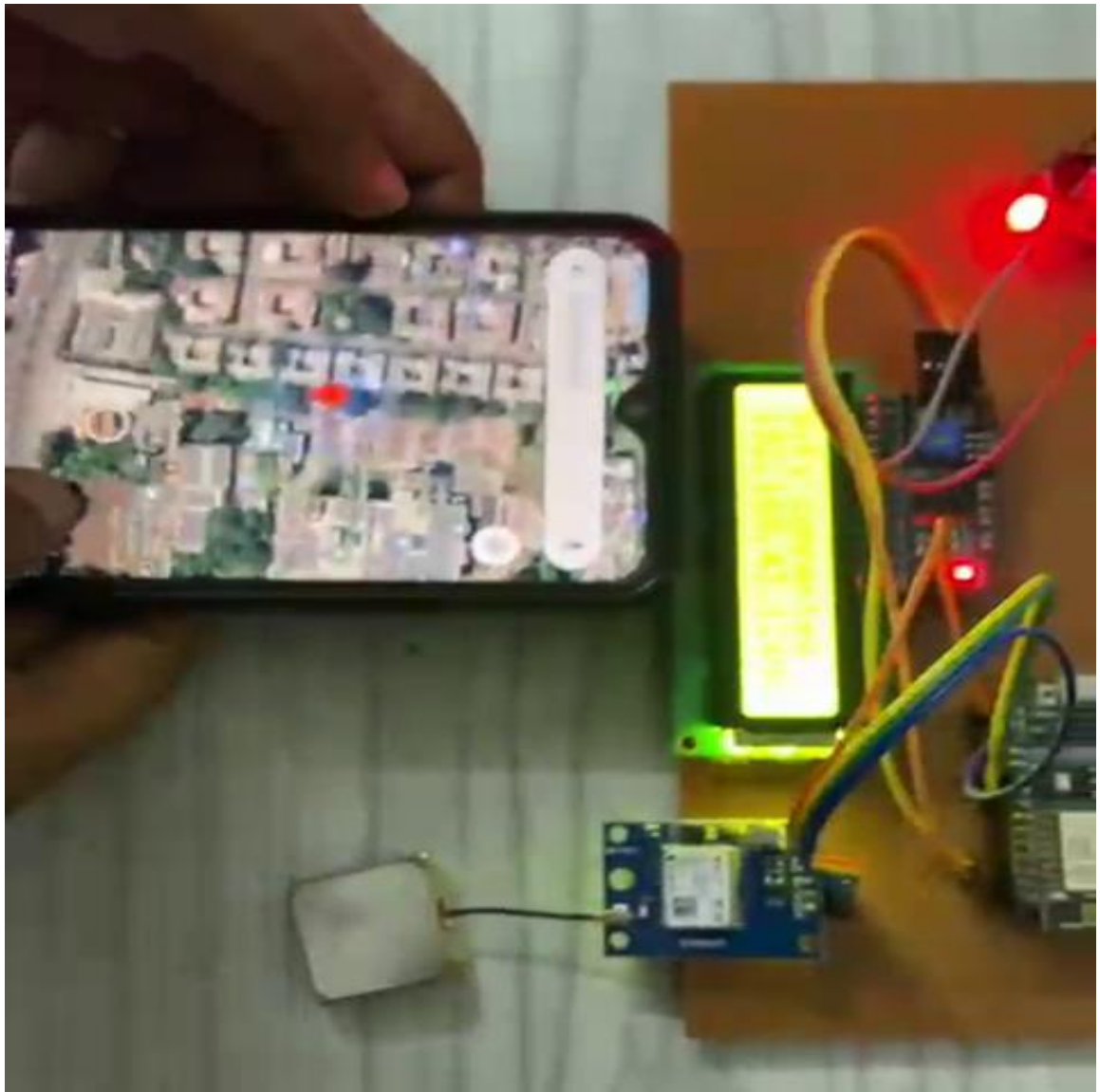


Fig.6.2 Displaying Screen

The result came out after applying these steps

- The code was uploaded on the NODEMCU port using Arduino IDE platform.
- Then the IP address showed up on the serial monitor with the status of the Wi-Fi if it's connected to the server or not.
- The device been Moved from one place to another while it's blinking.
- After some of time of collecting data on the device by the satellite.





We were able to track a car and get movement on the map. IoT tracking system both in case of personal as well as business purpose improves safety and security, performance monitoring and increases productivity. So in the coming years, it is going to play a major role in our day-to-day living. it's because of the module probability and antennas receive signal, strength and multipath.

CHAPTER-7

CONCLUSION

7.1 CONCLUSION

- In this research, we developed and tested a vehicle tracking system to track a moving or stationary vehicle in real-time. The proposed system uses the help of Assisted GPS in the crowded area to track and locate the transportation vehicle when the Autonomous GPS fails or is highly late. Three different areas are chosen to be tested for our system, 14 downtowns of Dallas-Texas, downtown of Little Rock-Arkansas and Petite Jean Mountain in Arkansas.
- The first part of the system that uses Autonomous GPS needs much longer time to start collecting the vehicle information due to the limitations of the GPS system. While Assisted GPS requires only few seconds to calculate the route accurately. Downtown of Dallas-Texas was the slowest region in calculating the route path, while downtown of Little Rock-Arkansas was the fastest case. Both systems have no differences in calculating routing path measurements.
- An in-vehicle device, hardware components are used for the vehicle tracking system composed of a Raspberry pi platform with 4G shield embedded with GPS sensor and a connection bridge. These components are to be connected to the hardware system and then to the cloud to obtain the vehicle's location/speed information and transmit it to a server through 4G LTE network. IoT platform (Ubidots) has been used which is a system that can be programmed and consequently customized by outside developers/users.
- Due to its flexibility, the platform can be adapted to countless needs and niches that the platform's original developers could not have possibly contemplated. The system can validate its operational performance to track a vehicle's location anytime from anywhere.

7.2 FUTURE DEVELOPMENTS

The field of IoT-based smart transportation systems is continuously evolving, and there are several potential future developments that could shape the direction of this technology. Some of the key future developments in IoT-based smart transportation systems may include:

- **5G and Beyond:** The deployment of 5G and beyond networks is expected to revolutionize the IoT landscape, including smart transportation systems. The high-speed and low-latency capabilities of 5G and beyond networks can enable real-time data processing and communication between devices, leading to more advanced and efficient transportation systems. This could enable applications such as autonomous vehicles, remote vehicle control, and enhanced real-time traffic management.
- **Edge Computing:** Edge computing, where data processing and analysis are done closer to the source of data rather than sending all data to a centralized cloud, is gaining traction in IoT-based systems. In smart transportation systems, edge computing can enable real-time data processing, analysis, and decision-making at the edge of the network, leading to reduced latency, improved data privacy, and enhanced system efficiency.
- **Multi-modal Transportation:** IoT-based smart transportation systems can enable seamless integration and coordination among different modes of transportation, such as public transportation, private vehicles, bicycles, and pedestrian traffic. Future developments may focus on creating integrated and multimodal transportation systems that provide users with personalized and optimized travel recommendations, real-time information on different modes of transportation, and seamless interconnectivity among different transportation options.
- **Sustainable Transportation:** With increasing concerns about environmental sustainability, future developments in IoT-based smart transportation systems may prioritize promoting sustainable transportation options, such as electric vehicles, carpooling, and public transportation. This may include the integration of charging infrastructure for electric vehicles, incentivizing sustainable

transportation choices through data-driven approaches, and optimizing transportation resources to reduce emissions and minimize environmental impact.

➤ **User Experience and Services:** Future developments in IoT-based smart transportation systems may prioritize enhancing the overall user experience and services. This may include the development of user-centric applications, such as mobile apps and smart city platforms, that provide personalized travel recommendations, real-time information on traffic conditions and public transportation schedules, and seamless and convenient user interactions with the transportation system.

In conclusion, the future developments in IoT-based smart transportation systems are expected to be driven by advancements in technologies such as 5G, edge computing, AI/ML, blockchain, and V2X communication, with a focus on sustainable transportation, user experience, and seamless integration among different modes of transportation. These developments have the potential to transform the transportation industry and enable more efficient, safe,

7.3 RESEARCH ISSUES

While IoT-based smart transportation systems offer numerous benefits, there are several research issues that need to be addressed for their successful implementation. Some of the key research issues in IoT-based smart transportation systems may include:

➤ **Data Security and Privacy:** The vast amount of data generated by IoT devices in smart transportation systems, such as vehicle data, traffic data, and user data, raises concerns about data security and privacy. Ensuring the confidentiality, integrity, and availability of data, as well as protecting the privacy of users and their personal information, is critical in the design and operation of IoT-based smart transportation systems.

➤ **Interoperability and Standardization:** Interoperability and standardization are important challenges in IoT-based smart transportation systems, as different

devices, sensors, and systems may use different communication protocols, data formats, and interfaces. Achieving seamless interoperability and standardization among various components of the transportation ecosystem, such as vehicles, infrastructure, and back-end systems, is crucial for efficient data exchange, decision-making, and system integration.

➤ **Scalability and Reliability:** IoT-based smart transportation systems can generate massive amounts of data, and the scalability and reliability of the underlying infrastructure are critical for handling such data volumes. Ensuring the scalability and reliability of IoT-based smart transportation systems, including the communication networks, data storage, and processing capabilities, is essential to support real-time data processing, analysis, and decision-making for efficient transportation operations.

➤ **Energy Efficiency:** Many IoT devices in smart transportation systems are battery-powered, such as sensors in vehicles, traffic lights, and roadside infrastructure. Energy efficiency is a critical research issue in IoT-based smart transportation systems, as optimizing the energy consumption of these devices can extend their battery life, reduce maintenance costs, and enhance overall system sustainability.

➤ **System Resilience and Robustness:** IoT-based smart transportation systems are susceptible to various types of disruptions, such as network failures, cyber-attacks, and system failures. Ensuring the resilience and robustness of the system to these disruptions, including fault tolerance, redundancy, and recovery mechanisms, is crucial for maintaining the integrity, availability, and reliability of the transportation system.

7.4 IMPLEMENTATION ISSUES

Implementing IoT-based smart transportation systems can present several challenges that need to be carefully addressed for successful deployment. Some of the key implementation issues in IoT-based smart transportation systems may include:

- **Infrastructure and Connectivity:** Establishing the required infrastructure and ensuring reliable connectivity is crucial for the operation of IoT-based smart transportation systems. This includes setting up IoT devices, sensors, communication networks, and data storage facilities in a distributed and coordinated manner. Implementing robust and scalable communication networks, such as 5G, and addressing challenges related to coverage, latency, and reliability, is essential for enabling seamless data exchange and real-time communication among various components of the transportation ecosystem.
- **Cost and Investment:** Implementing IoT-based smart transportation systems can involve significant costs, including the deployment of IoT devices, communication networks, data storage, and analytics platforms. Cost considerations, including initial investments, ongoing operational costs, and potential returns on investment, need to be carefully evaluated to ensure the feasibility and sustainability of the system. Identifying funding sources, developing business models, and assessing the economic viability of the system are important implementation issues that need to be addressed.
- **Security and Privacy:** Ensuring the security and privacy of data and systems in IoT-based smart transportation systems is a key implementation issue. Protecting against cyber-attacks, unauthorized access, data breaches, and privacy violations is crucial for maintaining the integrity, confidentiality, and availability of data, as well as ensuring user trust and confidence in the system. Implementing robust security measures, such as encryption, authentication, access control, and intrusion detection, is essential to safeguard the system against security threats.
- **Regulatory and Legal Compliance:** IoT-based smart transportation systems may need to comply with regulatory and legal requirements, such as data privacy, cybersecurity, and transportation regulations. Understanding and adhering to the

applicable regulations, standards, and guidelines, and obtaining necessary approvals and permits, are important implementation issues that need to be addressed to ensure compliance with legal and regulatory frameworks.

➤ **User Adoption and Acceptance:** The successful implementation of IoT-based smart transportation systems relies on user adoption and acceptance. User acceptance, behavior, and perceptions play a crucial role in the adoption and usage of the system. Addressing user concerns, providing user-friendly interfaces, ensuring transparency in data handling, and fostering user trust are important implementation issues that need to be considered to ensure user adoption and acceptance of the system.

In conclusion, implementing IoT-based smart transportation systems can present several challenges that need to be carefully addressed for successful

REFERENCES

- [1] for Buses in City of GPS.”: 91–99. Al-Lawati, Anwaar et al. 2015. “RFID-Based System for School Children Transportation Safety Enhancement.” 2015 IEEE 8th GCAcademy, World. 2010. “Real Time Web-Based Vehicle Tracking Using Conference and Exhibition, GCCCE 2015: 1–4.
- [2] Catherine Shu. 2013. “Guardian Uses Bluetooth Low Energy Tech to Keep Your Child SafeTechCrunch.” <https://techcrunch.com/2013/10/09/guardian-uses-bluetoothlow-energy-tech-to-keep-your-child-safe/> (May 13, 2018)
- [3] School Buses.” <https://newatlas.com/kidtrack-biometric-school-bus-scanner/267Ben-Coxworth>. 2013. “Kidtrack Biometric System Keeps Track of Kids on 23/ (May 13,2018)
- [4] Clarke, R., Bowen, t. and Head, J., “Mass Deployment of Bus Priority Using Real-Time Passenger Information Systems”, Proc. European Transport conference, Leeuwenhorst, Netherlands, (2007).
- [5] Glasgow City Council, (2009). Online, accessed March 2009. Downloadable from <http://www.glasgow.gov.uk/NR/rdonlyres/160FA4CBACC3-4F35-922D471C3502617D/0/TheBusInformationandSignallingSystem.pdf>.
- [6] Hardy, N., “iBus Benefits Realisation Workstream: Method and Progress to Date”, Proc. 16th ITS World Congress, Stockholm, (2009).
- [7] Helsinki Urban Traffic Control, “Helsinki Public Transport Signal Priority and Passenger Information”. Online, accessed March 2009. Downloadable from http://www.hel2.fi/liikenteenohjaus/eng/pt_telematics.asp.
- [8] Hill R., Bretherton D. and Maxwell A., “Real time Passenger Information and Bus Priority System in Cardiff – Development and Bus Priority Trial”, Proc. 8th World Congress on Intelligent Transport Systems, Sydney, (2001)
- [9] Hounsell, N.B., McLeod, F.N. and Shrestha, B.P., “Bus Priority at Traffic Signals: Investigating the Options” Proc. 12th International Conference on Road Traffic Information and Control, IEE London, (2004).

- [10] Hounsell, N. B., Shrestha, B. P., AVL based Bus Priority at Traffic Signals: A Review and Case Study of Architectures. *Eur. J. Transp. Infrastruct. Res.*, 5, (1), pp. 13-29, (2005).
- [11] Hounsell, N. B., Shrestha, B. P., McLeod, F. N., Palmer, S., Bowen, T., Head J. R., "Using GPS for Bus Priority in London: Traffic Signals Close to Bus Stops", *IET Intell. Trans. Syst.*, 2007, 1, (2), pp.131-137, (2006).
- [12] Jensen M. and Birk J., Transit Signal Priority (TSP) Aalborg. *Proc. European ITS Congress, Aalborg*, (20

SOURCE CODE

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include<LiquidCrystal_I2C.h>
#include<Wire.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
TinyGPSPlus gps; // The TinyGPS++ object
//rx=12 tx=13
SoftwareSerial ss(13, 12); // The serial connection to the GPS device
const char* ssid = "havi"; //Replace with ssid
const char* password = "12345"; //replace with password
float latitude , longitude;
int year , month , date, hour , minute , second;
String date_str , time_str , lat_str , lng_str;
int pm;
WiFiServer server(80);
void setup()
{
    Serial.begin(115200);
    lcd.backlight();
    ss.begin(9600);
    lcd.init();
    lcd.setCursor(0, 0);
    lcd.print("  IOT THEFT    ");
    lcd.setCursor(0, 1);
    lcd.print("VEHICLE TRACKING ");
    delay(2000);
    lcd.clear();
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);
```

```

WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED)
{
    delay(500);
    Serial.print(".");
    lcd.setCursor(0, 0);
    lcd.print("WiFi connecting... ");
}
Serial.println("");
Serial.println("WiFi connected");
lcd.setCursor(0, 0);
lcd.print("WiFi connected ");
lcd.setCursor(0, 1);
lcd.print(WiFi.localIP());
server.begin();
Serial.println("Server started");
// Print the IP address
Serial.println(WiFi.localIP());
}

void loop()
{
    while (ss.available() > 0)
        if (gps.encode(ss.read()))
        {
            if (gps.location.isValid())
            {
                latitude = gps.location.lat();
                lat_str = String(latitude , 6);
                longitude = gps.location.lng();
                lng_str = String(longitude , 6);
            }
            if (gps.date.isValid())
            {

```

```

date_str = "";
date = gps.date.day();
month = gps.date.month();
year = gps.date.year();
if (date < 10)
    date_str = '0';
date_str += String(date);
date_str += " / ";
if (month < 10)
    date_str += '0';
date_str += String(month);
date_str += " / ";
if (year < 10)
    date_str += '0';
date_str += String(year);
}
if (gps.time.isValid())
{
    time_str = "";
    hour = gps.time.hour();
    minute = gps.time.minute();
    second = gps.time.second();
    minute = (minute + 30);
    if (minute > 59)
    {
        minute = minute - 60;
        hour = hour + 1;
    }
    hour = (hour + 5) ;
    if (hour > 23)
        hour = hour - 24;
    if (hour >= 12)
        pm = 1;

```

```

else
    pm = 0;
hour = hour % 12;
if (hour < 10)
    time_str = '0';
time_str += String(hour);
time_str += " : ";
if (minute < 10)
    time_str += '0';
time_str += String(minute);
time_str += " : ";
if (second < 10)
    time_str += '0';
time_str += String(second);
if (pm == 1)
    time_str += " PM ";
else
    time_str += " AM ";
}
}

```

// Check if a client has connected

```
WiFiClient client = server.available();
```

```
if (!client)
```

```
{
    return;
}
```

// Prepare the response

```
String s = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n <!DOCTYPE
html> <html> <head> <title>GPS Interfacing with NodeMCU</title> <style>";
s += "a:link {background-color: YELLOW;text-decoration: none;}";
```

```

s += "table, th, td {border: 1px solid black;} </style> </head> <body> <h1
style="";
s += "font-size:300%;";
s += " ALIGN=CENTER> GPS Interfacing with NodeMCU</h1>";
s += "<p ALIGN=CENTER style='\"font-size:150%;\"'";
s += "> <b>Location Details</b></p> <table ALIGN=CENTER style="";
s += "width:50%";
s += "> <tr> <th>Latitude</th>";
s += "<td ALIGN=CENTER >";
s += lat_str;
s += "</td> </tr> <tr> <th>Longitude</th> <td ALIGN=CENTER >";
s += lng_str;
s += "</td> </tr> <tr> <th>Date</th> <td ALIGN=CENTER >";
s += date_str;
s += "</td></tr> <tr> <th>Time</th> <td ALIGN=CENTER >";
s += time_str;
s += "</td> </tr> </table> ";

if (gps.location.isValid())
{
s += "<p align=center><a style='\"color:RED;font-size:125%;\"'
href='\"http://maps.google.com/maps?&z=15&mrt=yp&t=k&q="";
s += lat_str;
s += "+";
s += lng_str;
s += "\"\" target='\"_top\"\">Click here!</a> To check the location in Google
maps.</p>";
}
s += "</body> </html> \n";
client.print(s);
delay(100);
}

```

