A Midterm Progress Report On

Smart Traffic Light System

Submitted in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

ROSHNI	SIMRANJEET KAUR	TANVEER SINGH PUNIAH
URN – 2203593	URN – 2203596	URN – 2203599
CRN – 2115186	CRN – 2115189	CRN – 2115192

UNDER THE GUIDANCE OF

PROF. SATINDERPAL SINGH

(APRIL-2025)

Department of Computer Science and Engineering

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

INDEX

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Introduction to Project

The Smart Traffic Light System aims to revolutionize traffic management by introducing a dynamic, real-time solution that adapts to traffic flow, ensuring optimal traffic control and minimizing congestion. This project leverages cutting-edge technologies to create an intelligent traffic light system that can detect and respond to real-time traffic conditions. By utilizing sensor networks, IoT (Internet of Things) devices, and the Blynk mobile application, the system will analyze traffic data and adjust signal timings accordingly. The project's scope lies in the specialized field of smart transportation systems, where technology is integrated with infrastructure to improve urban mobility. The Smart Traffic Light Project falls under the domain of intelligent transportation systems (ITS), a rapidly growing field that incorporates advanced technologies to enhance the efficiency and safety of transportation networks.

It is particularly relevant in urban areas facing challenges of increasing traffic volumes and limited road infrastructure. The Smart Traffic Light Project aims to revolutionize traffic management by introducing a dynamic, real-time solution that adapts to traffic flow, ensuring optimal traffic control and minimizing congestion. This project integrates advanced technologies like IoT (Internet of Things)**,** sensor networks**,** by providing adaptive traffic management, it enhances urban mobility and addresses the challenges posed by increasing traffic volumes in cities with limited infrastructure. Urban areas worldwide are grappling with escalating traffic congestion, leading to increased travel times, fuel consumption, and environmental pollution. Traditional traffic management systems, often reliant on fixed signal timings, fail to adapt to the dynamic nature of urban traffic, resulting in inefficiencies and heightened commuter frustration. The Smart Traffic Light System aims to revolutionize traffic management by introducing a dynamic, real-time solution that adapts to traffic flow, ensuring optimal traffic control and minimizing congestion.

This project leverages cutting-edge technologies to create an intelligent traffic light system that can detect and respond to real-time traffic conditions. By utilizing sensor networks, IoT (Internet of Things) devices, and the Blynk mobile application, the system will analyze traffic data and adjust signal timings accordingly. To fulfill the rising demands of the populace, alternative energy sources are in high demand. There are two ways to go about it: first, identify new energy sources, and second, cut back on how much energy you use from the currently accessible supplies. The second goal, which is to reduce energy use, is covered in this essay. Basically, this is a research of streetlights with microcontroller-based control lighting.

LED light, a motion sensor, and communication network make up the major components of the smart street light system. When cars arrive, the lights come on, and when no one is there, they go off. It will be challenging for pedestrians and car drivers to differentiate between smart street lighting and the standard street lights, as all street lights switch on prior to their arrival. About 30% of the energy used in cities is used for public lighting, including that in streets, tunnels, ports, and other public spaces. Cities require energy-saving smart lighting systems based on economic and environmental considerations. IoT based street lighting system ensures various metrics of the system like automatic switching of lights from ON state to OFF state or vice versa based on the condition provided for the system. For example, an IoT based system which comprises sensors, actuators, relays, IoT hardware and software with some networking and communication protocols enable the automatic street light monitoring and controlling system to function properly.

This IoT based system will be installed on each pole for controlling the lights and monitoring the street lights energy consumption on IoT based smart street light monitoring and controlling system is demonstrated in which different components of IoT like IoT devices, Gateways, Cloud, web browser, big data, sensors play essential role. All the smart systems are not intelligent but all the intelligent systems are smart it is because intelligent systems work on larger amount of Smart Street Light Monitoring and Controlling System data means there will be good decisions whereas poor amount of data means there would be wrong predictions. So, for making intelligent systems more and more reliable big data is essential requirement. Street Lights are having very high contribution for providing the safety of transportation system and smart cities development. The existing street lighting system uses old techniques and it is facing so many problems like: Existing Street lighting systems are needed to be Turned ON and OFF manually. It has a high-power consumption and their maintenance is also quite expensive. More manpower is required to handle the functioning of the existing street light system. There is a growing demand of IoT based Smart intelligent systems for street light monitoring and controlling operations. Recently, new technologies evolved for smart city development, smart healthcare, smart agriculture, smart transportation system are also exploring for smart street lighting systems for cities. These technologies can resolve the challenges faced by existing systems such as with the help of IoT, street lights can be switched ON and OFF automatically.

Energy Efficiency and Cost Reduction in IoT-Based Street Lighting:

Maintenance of street lights using IoT is quite less which leads to cost reduction. Power consumption is quite low in these street lights using IoT which also leads to energy conservation. No large manpower is required to maintain these street lights using IoT technologies. Monitoring the usage of street lights using IoT system is quite easy. Nowadays, it is observed that the sodium lamps are replaced with LED lights in streets of a city because one of the major factors is power consumption which is less and cost is another issue compared to sodium lamps. Further LED lights are eco-friendly and avoids greenhouse gas emission.

IoT-Based Smart Prototypes for Street Light Monitoring and Energy Management:

Our proposed street light monitoring and controlling system can conserve fair amount of energy we can also monitor it on the web browser. The major contribution of this research work is as follows: Firstly, aprototype is developed for monitoring and controlling of street lights using IoT platforms. Along with this, another prototype is proposed for monitoring electricity consumption at each pole and sending this electricity consumption information to IoT platform, i.e., on web browser. With the first prototype the street lights will TURNED ON/OFF depending on the presence of a moving object. With the implantation of this prototype, nearly 60-70 percent of energy saving can be achieved for a day which is demonstrated in discussion section thoroughly. The developed prototype 2 can be installed at each pole to measure electricity consumption or it could be separately installed for remote monitoring of electricity consumption of home appliances. The integration of both the prototypes can be utilized to collect real-time data of electricity consumption and with large amount of data by applying machine leaning algorithms future electricity consumption demands can be predicted and other controlling actions can be taken with the analysis of the real time data. This work presents an innovative approach to improving the efficiency and monitoring capabilities of street lighting systems using IoT technology.

Real-time adaptability: Unlike traditional systems, the proposed solution can adjust signal timings instantaneously based on current traffic conditions.

IoT integration: Utilizing IoT devices allows for seamless data collection and communication between system components.

Remote monitoring and control: The Blynk mobile application provides authorized users with the ability to oversee and adjust traffic signals remotely.

Cost-effectiveness: By leveraging affordable hardware and open-source software, the system offers a financially viable solution for cities with limited budgets.

Scalability: The modular design enables easy expansion to accommodate additional intersections and adapt to growing urban areas.

Project Category:

This project falls under the category of Intelligent Transportation Systems (ITS)**,** which utilizes modern technologies to enhance the efficiency and safety of transportation networks. The system uses real-time traffic data**,** sensor networks, and wireless communication to optimize traffic flow, making it particularly useful for urban areas facing heavy congestion.

Objectives:

1. To design an IoT-based smart traffic light system to manage traffic.
    - The system will utilize Node MCU ESP8266, an IoT-enabled microcontroller, to facilitate decentralized and real-time traffic management.
    - The ESP8266 will enable wireless communication between traffic signals, ensuring adaptive and automated traffic control.
2. To design a mobile application to control lights remotely.

- The system will integrate with the Blynk mobile application, allowing users or authorities to monitor and control traffic lights remotely.
- The app will provide a user-friendly interface for real-time traffic light management and data visualization.

3. To enable seamless wireless communication for efficient and responsive control.
    - The traffic lights will communicate through a Wi-Fi-based IoT network, allowing real-time data exchange between signals.
    - This wireless connectivity ensures the system can dynamically adjust signal timings based on live traffic conditions, improving overall traffic flow.

CHAPTER 2: SYSTEM REQUIREMENTS

2.1 Hardware Requirements

Microcontroller: NodeMCU ESP8266 (Wi-Fi Enabled)

- Acts as the brain of the system, processing sensor data and controlling LED lights.
- Wi-Fi enabled, allowing communication with the Blynk app for remote control.
- Supports multiple GPIO pins for connecting sensors and LEDs.

LEDs: Red, Yellow, Green (Traffic Light Representation)

- Red LED: Indicates stop.
- Yellow LED: Warns vehicles to slow down.
- Green LED: Signals vehicles to go.

Sensors: IR Sensors or Ultrasonic Sensors for Traffic Detection

- Infrared (IR) Sensors: Detect vehicle presence based on infrared light reflection.
- Ultrasonic Sensors: Measure distance to detect traffic density and adjust signal timing accordingly.

Power Supply: 5V DC Adapter

- Provides power to the NodeMCU ESP8266 and connected components.
- Ensures stable operation of traffic light LEDs and sensors.

Resistors: 220Ω for LED Control

- Limits current flow to prevent LED burnout.
- Ensures proper brightness for visibility.

Wi-Fi Module: Integrated with ESP8266

- Allows real-time data communication between the traffic light system and the Blynk app.
- Enables wireless monitoring and control of traffic lights.

Breadboard & Wires: For Circuit Connections

- Used to connect and test hardware components.
- Facilitates easy prototyping and modifications without soldering.

2.2 Software Requirements

Arduino IDE: Code Development and Programming for Node MCU

- Used to write, compile, and upload code to the ESP8266 microcontroller.
- Provides debugging tools for testing the traffic light system.

Blynk App: Remote Control and Monitoring of Traffic Lights

- A cloud-based mobile application that connects with Node MCU over Wi-Fi.
- Allows users (e.g., traffic authorities) to remotely monitor and override traffic signals if needed.

ESP8266WiFi Library: Enables Wi-Fi Communication

- Allows the microcontroller to connect to the internet.
- Facilitates real-time data exchange with the Blynk cloud server.

Blynk Library: Integrates NodeMCU with the Blynk Mobile Application

- Provides an interface to send and receive data between ESP8266 and the Blynk app.

- Allows users to control and monitor traffic lights remotely.

How These Components Work Together

- The microcontroller (NodeMCU ESP8266) receives traffic data from IR/ultrasonic sensors.
- The system processes the data and adjusts traffic light timing dynamically.
- The Wi-Fi module enables communication with the Blynk app, allowing remote monitoring and manual control.
- The Arduino IDE is used for developing and uploading code to the NodeMCU.
- The Blynk app acts as the user interface, displaying traffic light status and allowing remote modifications.

CHAPTER 3: SOFTWARE REQUIREMENT ANALYSIS

3.1 Problem Definition

The increasing traffic congestion in urban areas necessitates an intelligent system that dynamically manages traffic lights based on real-time traffic flow. Traditional traffic light systems operate on fixed timing mechanisms, leading to inefficiencies such as prolonged waiting times, unnecessary stops, and increased fuel consumption. The Smart Traffic Light System aims to address these challenges by implementing an IoT-based solution that adapts to traffic density, enhances road safety, and allows remote control via a mobile application. Traffic congestion and inefficient traffic management in urban areas lead to significant delays, increased fuel consumption, and environmental pollution. Traditional traffic light systems operate on fixed schedules, which often fail to adapt to real-time traffic conditions, resulting in prolonged waiting times and inefficient signal transitions. Additionally, manually adjusting traffic lights is challenging due to the lack of a centralized control system, making it difficult to optimize schedules based on actual road conditions.To address these challenges, the Smart Traffic Light System is designed to integrate 5mm 12 LED lights with the Blynk mobile application, enabling real-time monitoring, manual control, and automated scheduling of traffic signals.

One of the key problems this system resolves is the delay in schedule timing adjustments, which affects the efficiency of traffic flow. By leveraging NodeMCU ESP8266, the system allows seamless switching of LED signals based on predefined schedules while also providing the flexibility to make real-time adjustments via the Blynk app.The Blynk application serves as the primary interface, allowing users to override the system, manually change signals using button controls, and automate light transitions based on set time intervals. This ensures that traffic signals remain responsive to changing traffic patterns, ultimately improving urban mobility, reducing congestion, and enhancing road safety. The implementation of IoT in traffic management systems has been extensively reviewed in the literature. For instance, the paper titled "A Review of IoT Application in a Smart Traffic Management System" discusses various IoT applications aimed at improving traffic flow and reducing congestion. The paper highlights the potential of IoT-based systems to enhance traffic management through real-time data collection and analysis, leading to more informed decision-making processes. By leveraging IoT technologies, the Smart Traffic Light System can collect and analyze traffic data in real-time, facilitating dynamic adjustments to signal timings. This capability not only optimizes traffic flow but also contributes to reduced fuel consumption and lower emissions, aligning with broader environmental objectives.

3.2 System Modules and Their Functionalities

To effectively implement the Smart Traffic Light System, the project is divided into several functional modules. Each module plays a crucial role in ensuring seamless operation, remote accessibility, and adaptive traffic control. The system integrates 5mm 12 LED lights, a NodeMCU ESP8266, and the Blynk mobile application to create an intelligent traffic management system.

1. Traffic Light Control Module

- This module controls the LED-based traffic signals using a NodeMCU ESP8266.
- The system includes 5mm 12 LED lights (Red, Yellow, Green) that simulate real-world traffic signals.
- The NodeMCU microcontroller regulates the lights based on predefined schedules and real-time inputs received from the Blynk app.
- Ensures smooth operation by switching lights in a controlled sequence, regulating traffic flow effectively.

2. Blynk Integration Module

- The Blynk mobile application is used for remote monitoring and control.
- It includes a button interface for manually toggling between Red, Yellow, and Green signals.
- Implements a schedule timing feature, allowing automatic LED transitions at predefined intervals.
- Traffic authorities can override signals in real-time, ensuring adaptive

control.

3. **Wireless Communication Module**

- Uses the ESP8266 Wi-Fi module for real-time data transmission between the traffic light system and the Blynk app.
- Sends and receives commands to modify LED sequences based on user input or schedule settings.
- Ensures seamless operation, allowing quick adjustments based on traffic density.

4. **Timing and Scheduling Module**

- Defines a preset schedule for LED switching based on typical traffic patterns.
- Enables traffic authorities to customize timing via the Blynk app for better traffic management.
- Optimizes traffic flow by adapting to peak and off-peak hours, ensuring efficient road usage.

5. **Power Management Module**

- Ensures a stable power supply to the NodeMCU, LEDs, and sensors for uninterrupted operation.
- Uses low-power LED components to reduce energy consumption.
- Supports battery backup, ensuring functionality even during power failures.

6. LED Connection Using Micro Wire

The traffic light system is connected to the laptop via micro USB and micro wires to facilitate programming and data transmission. The logic behind the connection is:

1. Micro USB Connection – The NodeMCU is connected to the laptop via a micro USB cable, allowing for code uploads from the Arduino IDE.
2. LED Wiring –

- Red LED → GPIO Pin X
- Yellow LED → GPIO Pin Y
- Green LED → GPIO Pin Z
- Common Ground (GND) for all LEDs

3. Power Supply – The NodeMCU receives power from the laptop during programming and from an external 5V DC adapter during regular operation.
4. Blynk App Integration – The ESP8266 connects to Wi-Fi, enabling remote control via Blynk.

CHAPTER 4: SOFTWARE DESIGN

The Smart Traffic Light System project is designed to provide an intelligent and efficient traffic management solution. Built using modern technologies, the system aims to streamline vehicular movement, reduce congestion, and enhance road safety by incorporating essential features tailored to real-time traffic conditions.

This intelligent traffic management system is a step towards a smarter, safer, and more efficient urban transport network.

The **NodeMCU ESP8266 Flow Diagram** for your **Smart Traffic Light System** will illustrate how the microcontroller interacts with sensors, traffic lights, and a cloud-based system. Below is a structured breakdown of the flow:

Flow Diagram Breakdown

1. **Power On & Initialization**
   - The ESP8266 initializes and connects to the Wi-Fi network.
   - All connected sensors (IR sensors, cameras, ultrasonic, or RFID) are initialized.
2. **Traffic Data Collection**
   - Sensors and cameras collect real-time traffic data.
   - IR/ultrasonic sensors count vehicles at the intersection.
   - Cameras provide live feeds for image processing.
3. **Data Processing**
   - The ESP8266 processes sensor input locally or sends data to a cloud server.
   - If cloud processing is used, data is transmitted via MQTT or HTTP requests.
4. **Decision Making**
   - The microcontroller determines the optimal signal timing based on traffic density.
   - If an emergency vehicle is detected (RFID, siren detection, etc.), priority is given.
   - Pedestrian crossing requests are processed.

5. **Traffic Light Control**

   - The ESP8266 controls LED traffic lights according to the processed data.
   - Signal duration is dynamically adjusted to optimize traffic flow.

6. **Remote Monitoring & Alerts**

   - Traffic data is sent to a dashboard for remote monitoring.
   - Alerts are triggered for anomalies like traffic jams or sensor failures.

7. **Continuous Loop**

   - The process repeats continuously for real-time traffic optimization.

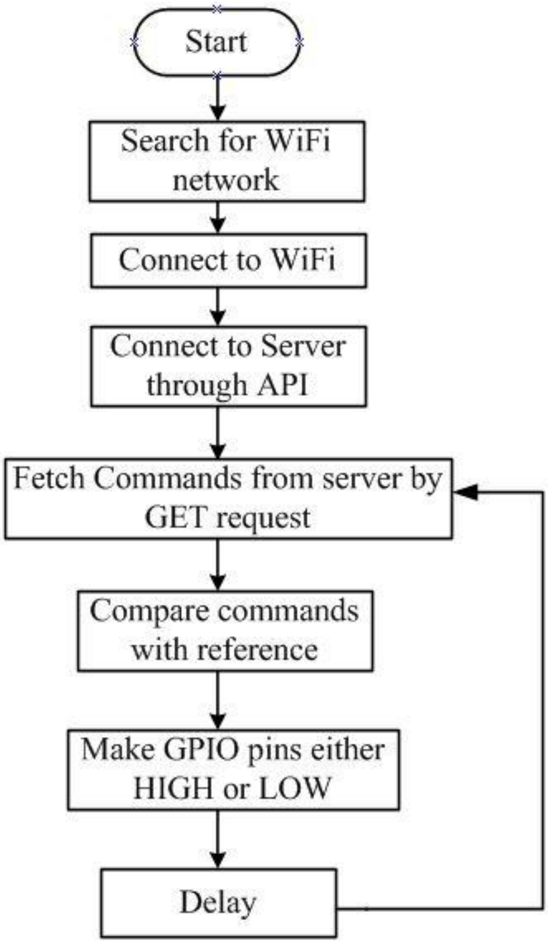

Figure 4.1 Node Mcu Workflow Diagram

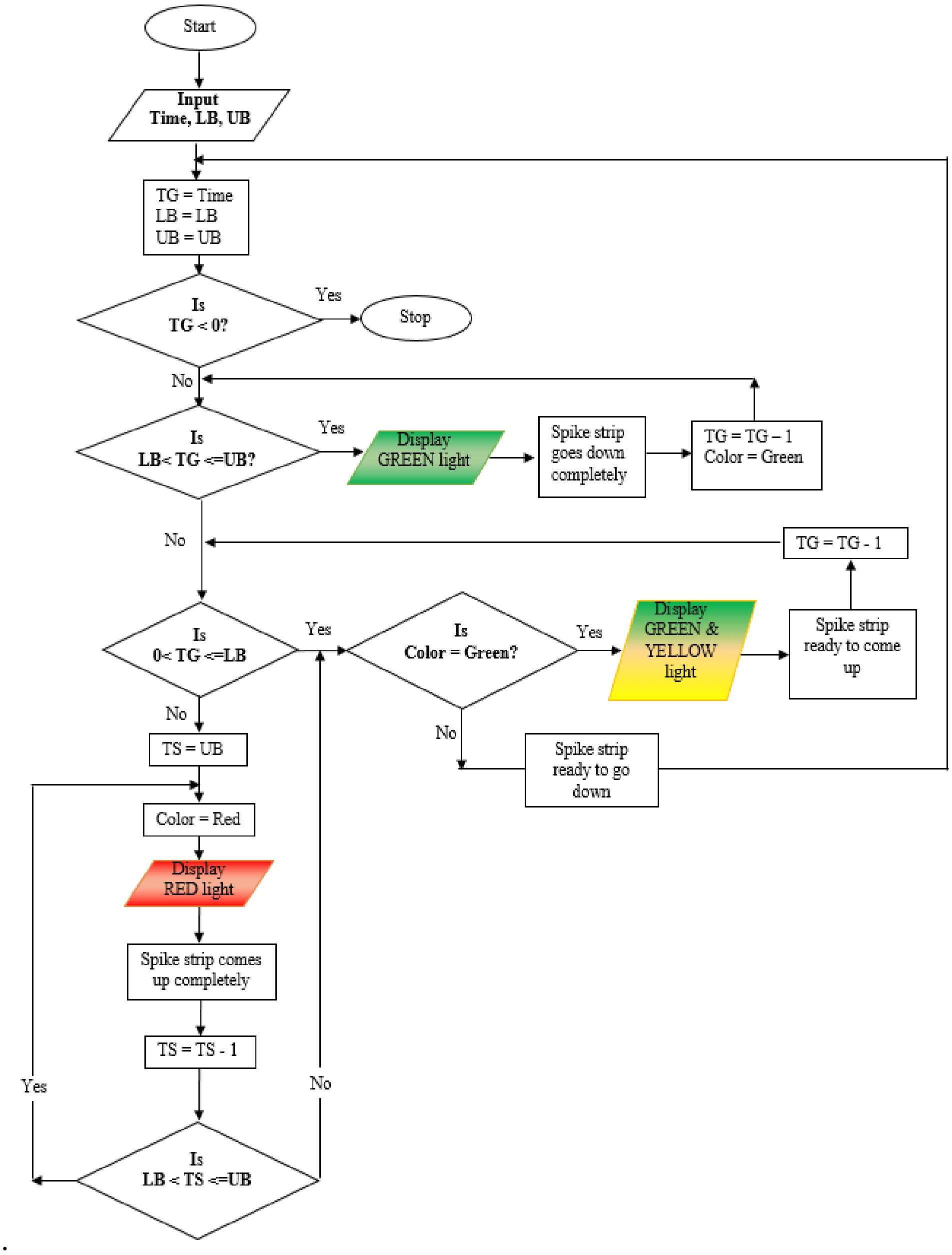

Figure 4.2 Flowchart Diagram of Traffic Light System

1. System Architecture

The system consists of:

- **Hardware Components:** NodeMCU ESP8266, Arduino, LEDs (representing traffic lights), and a Breadboard.
- **Software & Communication:**
  - **Arduino IDE:** To program the Arduino controlling the lights.
  - **Blynk IoT App:** To monitor and control the traffic signals remotely.

    - **Wi-Fi Communication:** NodeMCU connects with Blynk via Wi-Fi for real-time operation.

2. Traffic Light Control Logic

- The **Arduino runs the core logic**, executing predefined traffic light sequences.
- The **NodeMCU ESP8266 communicates with Blynk**, allowing remote control of the lights.
- The **Blynk app acts as the user interface**, enabling administrators to manually switch lights ON/OFF or change their sequence.
- The system ensures a smooth **Red-Yellow-Green** transition for all 12 lights while avoiding conflicts.

Traffic Light Timing Rules:

- **Normal Operation Mode:**
    - Green → 30s, Yellow → 5s, Red → 30s
- **Manual Override Mode (via Blynk):**
    - Admin can manually control each light from the app.
- **Emergency Mode:**
    - If required, all signals can be turned red to stop traffic completely.

3. User Interface with Blynk IoT App

The Blynk IoT app serves as the remote interface for managing the traffic lights.

- **Virtual Buttons:**
    - Each button in the app corresponds to a traffic light, allowing ON/OFF control.
- **Status Monitoring:**
    - The app provides real-time feedback on the status of each light (Red/Yellow/Green).
- **Customization & Scheduling:**
    - The admin can modify signal timings or schedule automatic operations.

4. Wi-Fi & Communication Flow

- **NodeMCU connects to Wi-Fi** and syncs with the Blynk cloud server.
- **Blynk app sends control commands** to NodeMCU, which relays them to Arduino.
- **Arduino executes the commands**, turning the traffic lights ON/OFF accordingly.
- The system ensures **low-latency operation** and **real-time synchronization** of all signals.

5. Traffic Management Automation

- The system runs in **two modes**:
    1. **Automated Mode:** The Arduino follows a predefined timing sequence.
    2. **Manual Mode:** Admins can override traffic signals via the Blynk app.
- **Failsafe Mechanisms:** If Wi-Fi disconnects, the Arduino continues running the last programmed sequence.

The Smart Traffic Light System leverages Arduino, NodeMCU ESP8266, and Blynk IoT for efficient, remotely controlled traffic signal management. The integration of IoT ensures real-time manual and automated operation while maintaining traffic flow and safety.

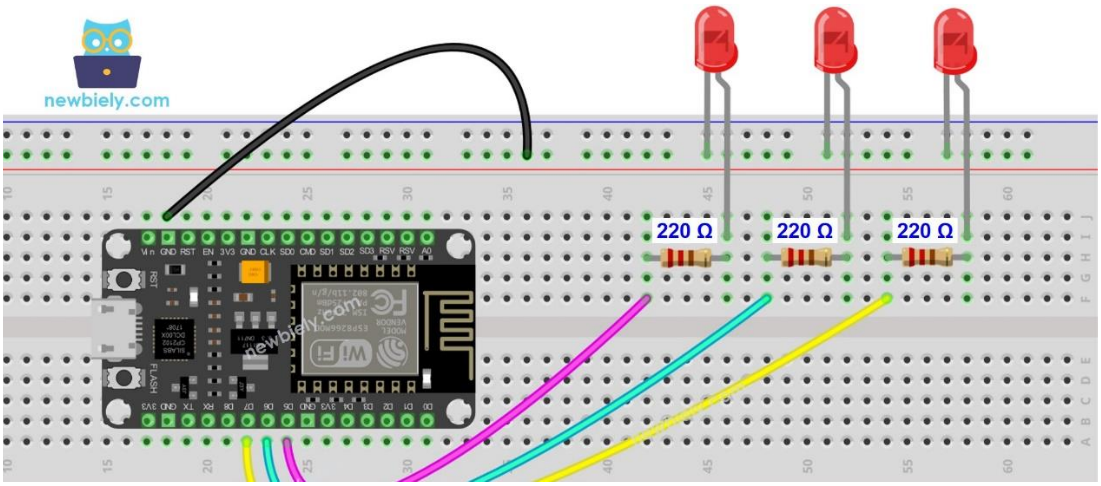

Figure 4.3 Pin Diagram of blinking Lights

CHAPTER 5: TESTING MODULE

To ensure the reliability and efficiency of our **Smart Traffic Light System**, various testing methodologies are applied, including **unit testing, integration testing, end-to-end testing, and regression testing**. Since our project is based on **Arduino, NodeMCU ESP8266, and the Blynk IoT App**, testing is focused on verifying the hardware-software interaction, light switching logic, and remote control functionality.

1. Unit Testing

Unit testing involves testing **individual components** of the system to validate their behavior.

In our project:

- The **Arduino code** is tested to ensure the correct timing and sequence of traffic lights (Green → Yellow → Red).
- The **NodeMCU ESP8266 Wi-Fi module** is tested for connectivity with the **Blynk IoT App** to ensure it receives and executes commands properly.
- The **Blynk App interface** is checked to confirm that it correctly sends manual light control signals.
- Testing tools like **Arduino Serial Monitor** and **Blynk Debug Console** are used to track responses and detect errors.

2. Integration Testing

Integration testing focuses on verifying the interaction between different modules:

- Ensuring that the **Arduino code correctly interacts** with the LED traffic lights.
- Testing the **Wi-Fi communication** between NodeMCU and the **Blynk App** to validate remote control functionality.
- Checking whether **manual and automatic modes switch properly** without interference.
- Simulation tools like **Proteus and Fritzing** can be used to simulate and validate component integration before physical implementation.

3. End-to-End Testing

End-to-end testing ensures the **entire system** functions as expected under real-world conditions:

- The **traffic light sequence runs smoothly** in a continuous loop without unexpected delays.

- The **Blynk App successfully overrides the automatic sequence** and manually controls traffic lights.
- In case of Wi-Fi disconnection, the system **continues running on predefined Arduino logic** without failure.

This testing is conducted by **deploying the system on a breadboard setup** and observing real-time behavior.

4. Regression Testing

Regression testing ensures that new updates or changes do not cause unintended failures:

- If modifications are made to the **timing logic** or **Blynk integration**, the **entire system is retested** to ensure consistency.
- Any **new features** (such as emergency mode) are tested to ensure they **do not interfere** with the existing sequence.

By continuously testing and refining the system, we ensure that our Smart Traffic Light System operates reliably, efficiently, and meets real-world traffic control needs.

CHAPTER 6: PERFORMANCE OF THE PROJECT DEVELOPED

The **Smart Traffic Light System** demonstrates excellent performance by effectively managing a **four-directional traffic signal system** using **Wi-Fi-based control**. With the integration of **NodeMCU ESP8266, Arduino, and the Blynk IoT App**, the system ensures seamless automation while also allowing remote manual control, making it both **efficient and adaptable** to different traffic conditions.

Key Performance Highlights

1. **Real-Time Responsiveness**
   - The system maintains **precise and consistent timing** for traffic light transitions (Green → Yellow → Red), ensuring smooth operation.
   - The **Wi-Fi-based communication** enables immediate responses when switching between manual and automatic modes.
   - The **Blynk App interface** allows instant control of traffic signals, ensuring quick adaptability during emergencies.
2. **Reliable Execution and Synchronization**
   - The traffic lights for **all four directions operate in a synchronized manner**, preventing conflicts in signal switching.
   - The programmed logic ensures that **no two opposite directions receive a green signal simultaneously**, reducing the risk of accidents.
   - In case of **Wi-Fi disconnection**, the system continues to function in its **predefined automatic mode** without disruption.
3. **Remote Accessibility and IoT Integration**
   - The **Blynk App provides real-time monitoring** and control from any location with internet access.
   - Administrators can **modify light timing settings remotely**, making it adaptable for varying traffic conditions.

- The system can be integrated with **smart city infrastructure**, enabling future scalability and automation improvements.

4. **Energy Efficiency and Cost-Effectiveness**
    - Using **low-power components like NodeMCU ESP8266 and LED lights**, the system operates with **minimal power consumption**.
    - The **breadboard prototype** ensures cost-effective testing and deployment before implementing the system on a larger scale.
    - The system eliminates the need for expensive hardware components like **dedicated controllers or complex wiring**, reducing maintenance costs.
5. **Scalability and Future Enhancements**
    - The system can be **expanded to control additional traffic intersections** by integrating multiple NodeMCU modules.
    - Additional features such as **emergency vehicle priority control** and **pedestrian crossing management** can be added for enhanced functionality.
    - The use of **cloud storage and data analytics** in future updates can help track traffic patterns and optimize signal timing dynamically.

The Smart Traffic Light System offers a highly efficient, scalable, and reliable solution for managing urban traffic flow. By integrating IoT-based automation with manual override functionality, it ensures seamless operation, energy efficiency, and real-time adaptability to varying traffic conditions. This makes it a cost-effective and future-ready alternative to traditional traffic management systems.

CHAPTER 7: OUTPUT SCREENS

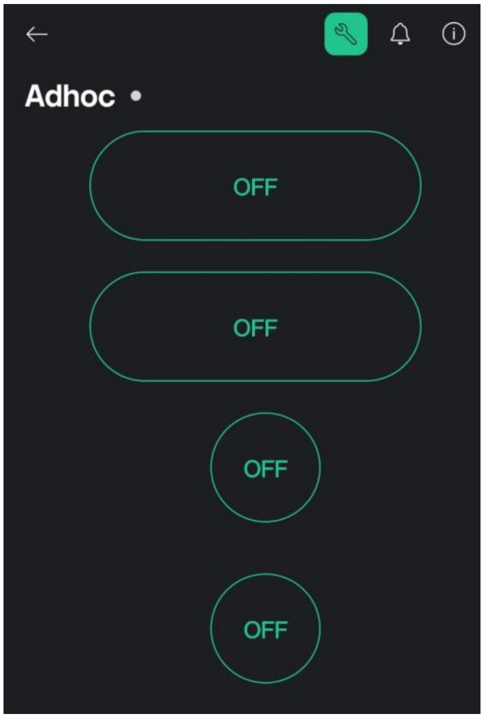

Figure 7.1: Traffic Light Controller IOT App

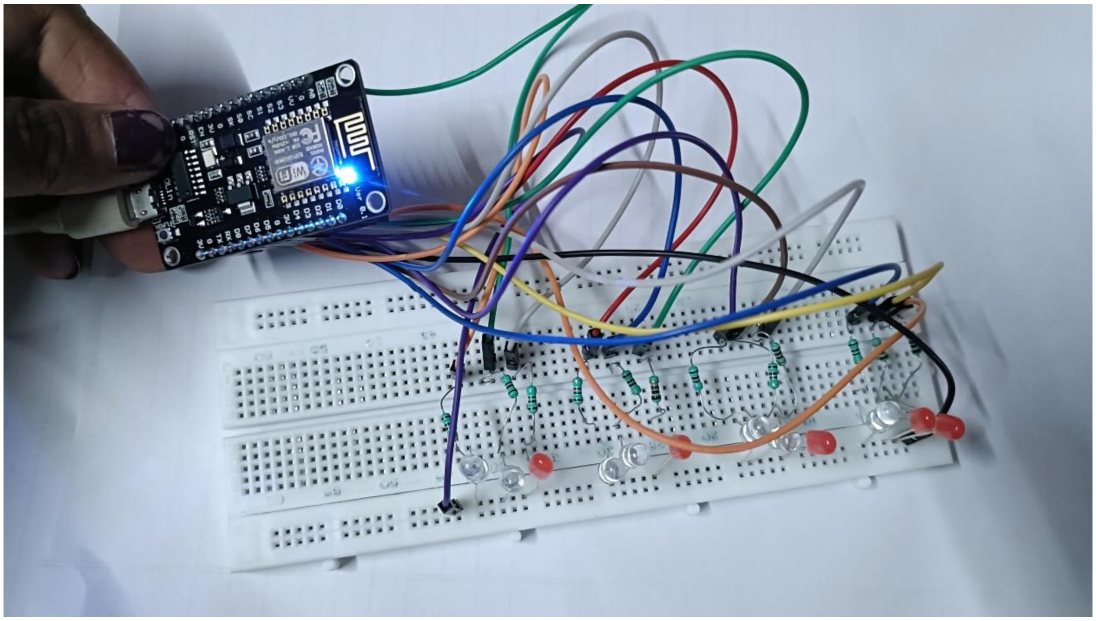

Figure 7.2: Initial Circuit Diagram

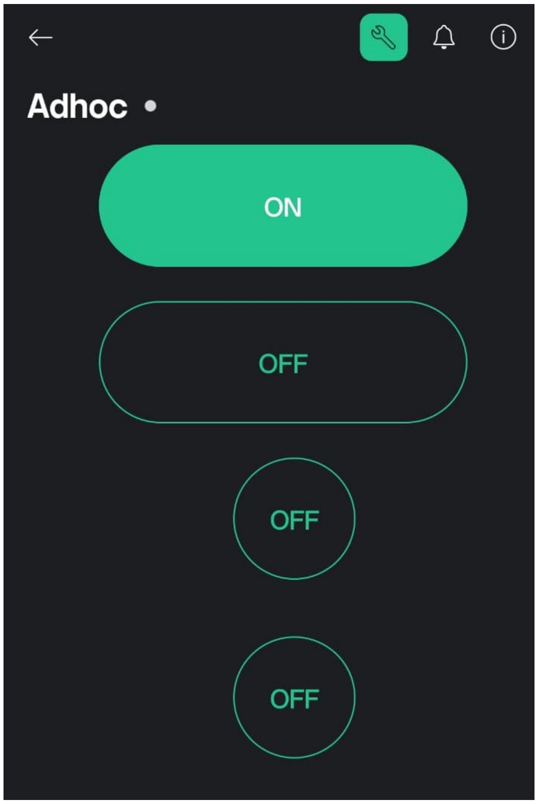

Figure 7.3: Activating Lane1 Flow

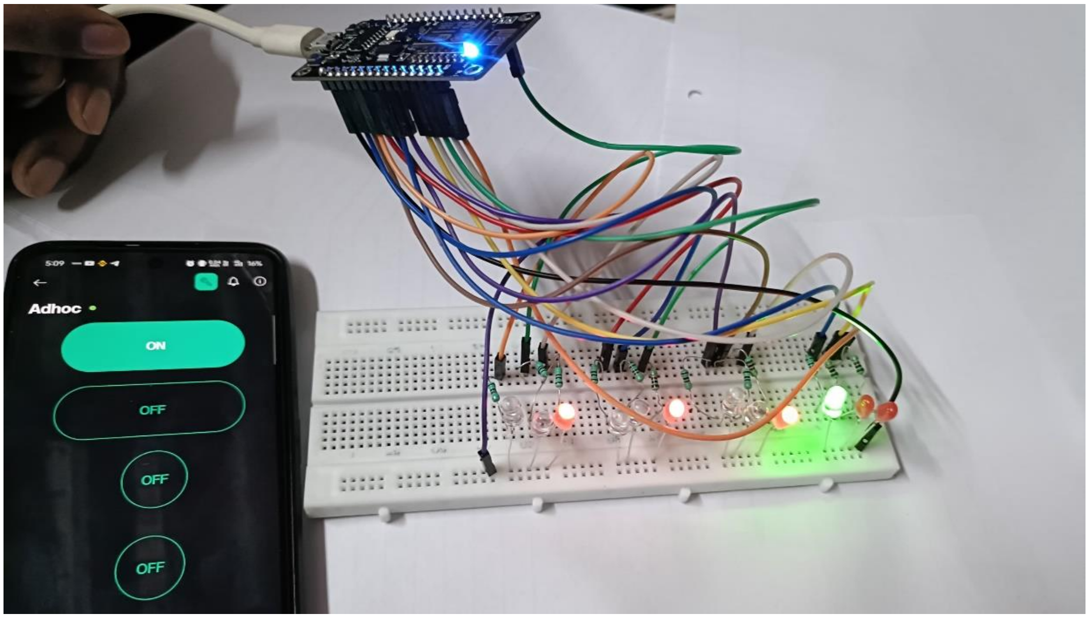

Figure 7.4: Green Light on Lane1 and red at others.

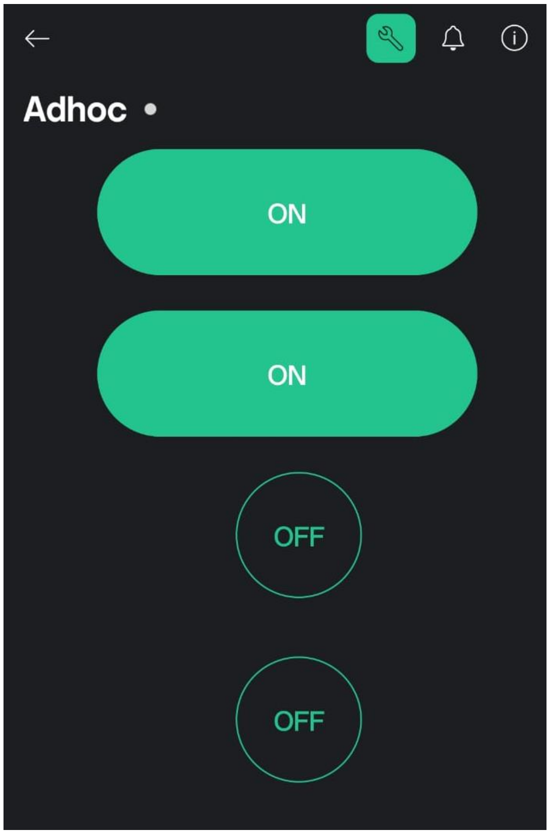

Figure 7.5: Activating Lane2 flow

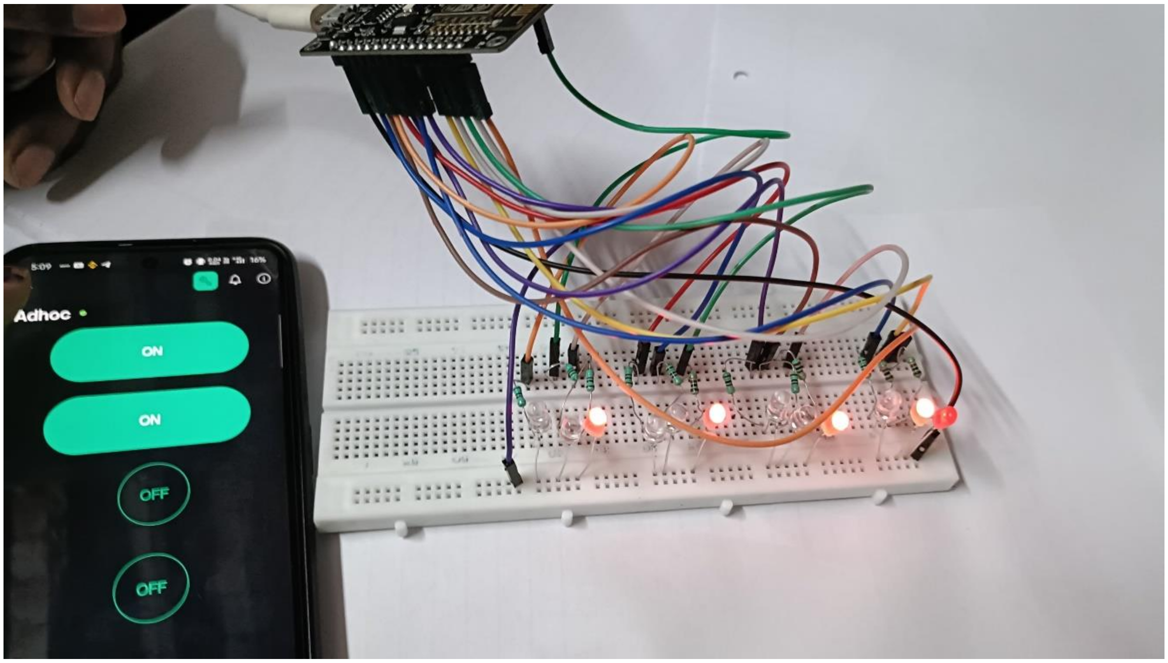

Figure 7.6: Showing Yellow Light at Lane1 and red at others.

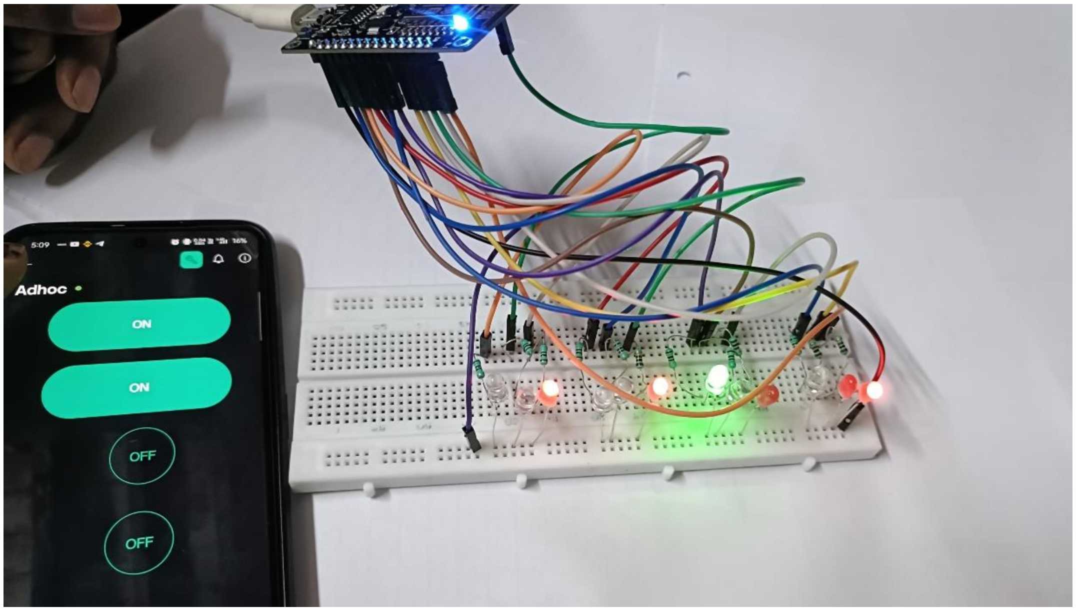

Figure 7.7: Showing a Green light at Lane2 and red at others.

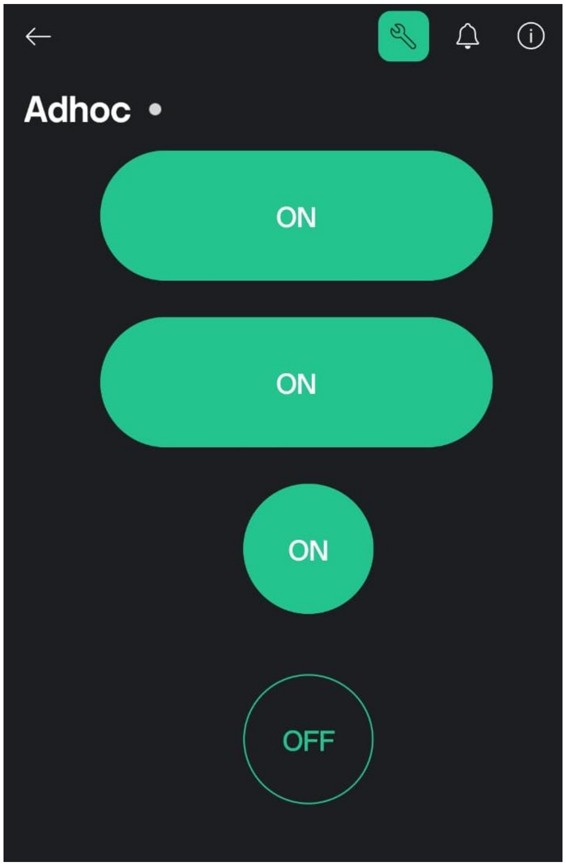

Figure 7.8: Activating Lane3 flow

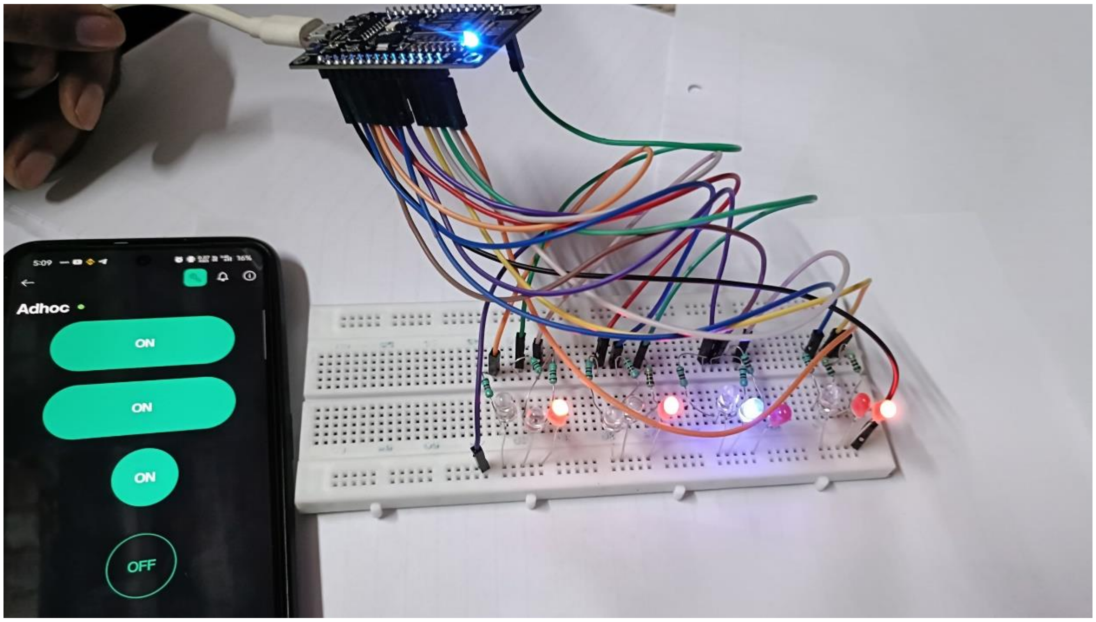

Figure 7.9: Showing Yellow Light at Lane2 and red at others.

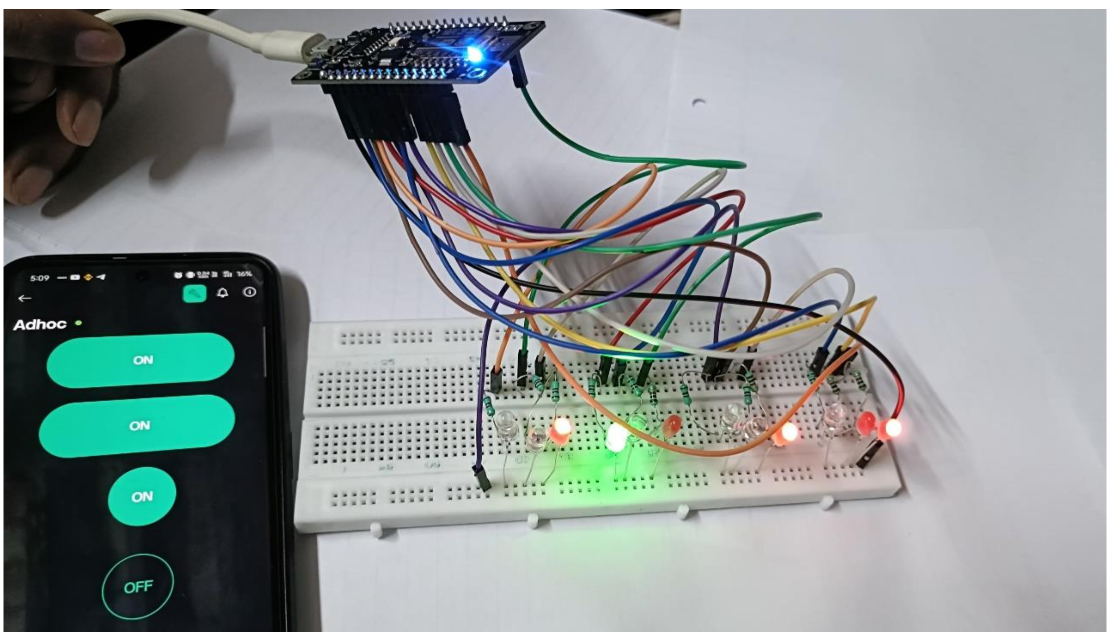

Figure 7.10: Showing a Green light at Lane3 and red at others.

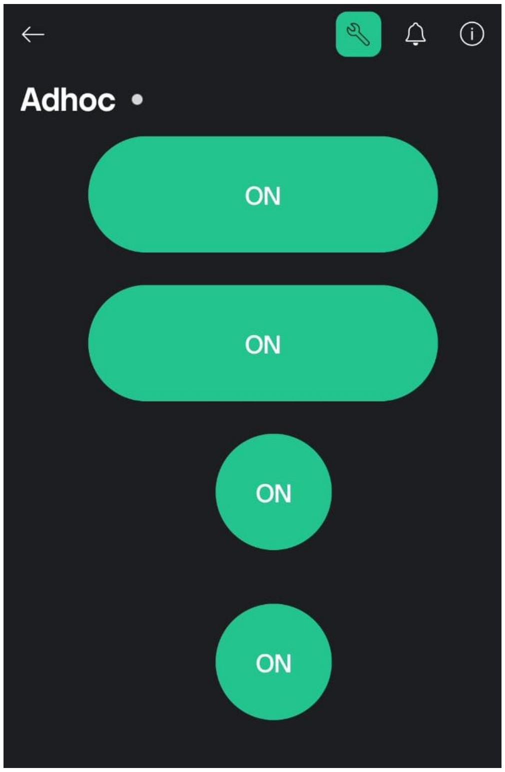

Figure 7.11: Activating Lane4 flow

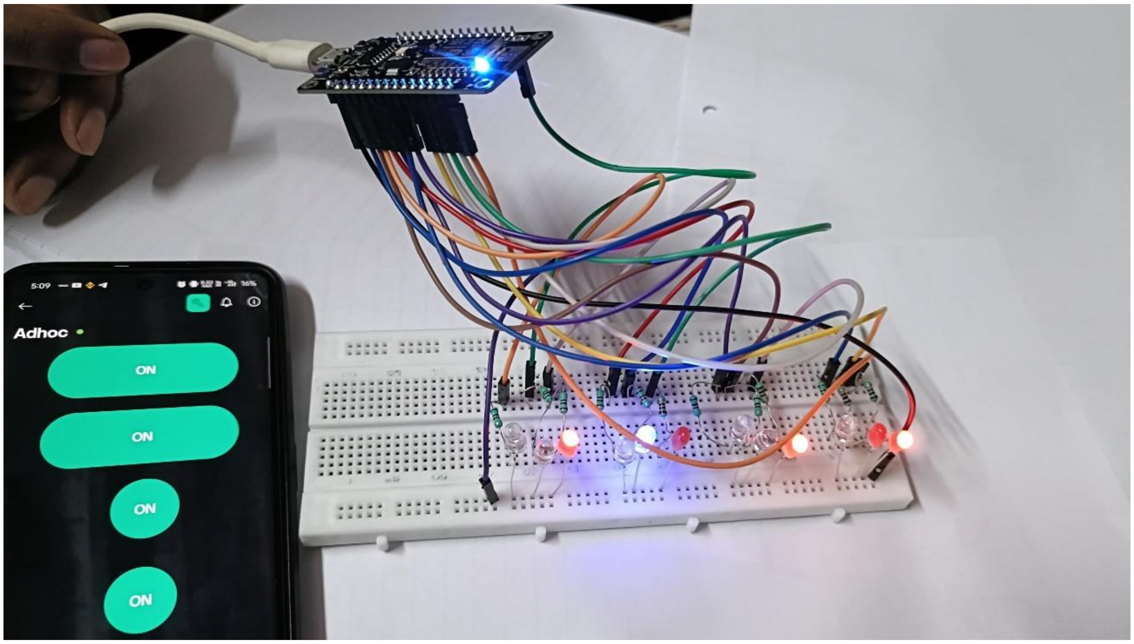

Figure 7.12: Showing Yellow Light at Lane3 and red at others.

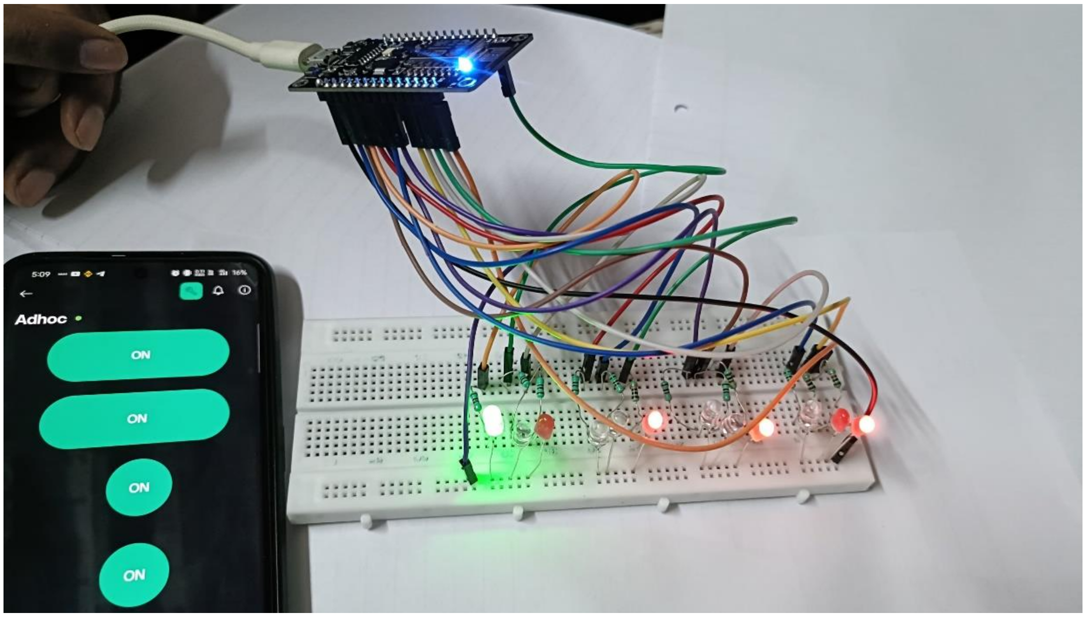

Figure 7.13: Showing a Green light at Lane4 and red at others.

CHAPTER 8: REFERENCES

[1] AF Ahamed, Y Sukhi, S Anita, H Charulatha… - AIP Conference …, 2025 - pubs.aip.org" *IEEE Internet of Things Journal*, vol. 4, pp. 1125–1142, 2025

[2] Gough, M. B., et al., "Preserving privacy of smart meter data in a smart grid environment," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 1, pp. 707-718, 2021.

[3] Dizon, E., and Pranggono, B., "Smart streetlights in Smart City: a case study of Sheffield," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-16, 2021.

[4] Avancini, D. B., et al., "A new IoT-based smart energy meter for smart grids," *International Journal of Energy Research*, vol. 45, no. 1, pp. 189-202, 2021.

[5] Bingöl, E., Kuzlu, M., and Pipattanasompom, M. A., "LoRa-based Smart Streetlighting system for Smart Cities," *7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, pp. 66–70, Apr. 2019.

[6] Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., and Zhao, W., "A survey on Internet of Things: architecture, enabling technologies, security and privacy, and applications," *IEEE Internet of Things Journal*, vol. 4, pp. 1125–1142, 2020.