**LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)**

*Author : Nelly Fananda Melani*

*Fakultas Vokasi, Universitas Brawijaya*

*Email:* [*nellyfananda15@student.ub.ac.id*](mailto:nellyfananda15@student.ub.ac.id)

**Abstract**

This experiment aims to analyze the implementation of a traffic light system based on ESP32 in the context of the Internet of Things (IoT) course. The simulation was developed using Visual Studio Code (VSCode) with the PlatformIO IDE extension and Wokwi Simulator, enabling testing without additional hardware. The system consists of three LEDs—green, yellow, and red—controlled via GPIO pins on the ESP32. The LED lighting sequence follows a predetermined cycle: green light for 5000 ms, yellow light for 3000 ms, and red light for 1000 ms before repeating. The experimental results indicate that this simulation provides fundamental insights into microcontroller programming, electronic device control, and time-based logic implementation in basic automation systems. In addition to serving as an academic exercise, this project has the potential for further development in real-world applications, including as a reference for final projects related to IoT systems and traffic automation.

**Keywords**—*Internet of Things, Traffic Light, ESP32, Automation, Simulation*

**1. Introduction (Pendahuluan)**

Traffic lights are a crucial component of modern transportation systems, regulating vehicle flow and enhancing road safety. These systems operate based on a predetermined cycle, where the red light signals vehicles to stop, the yellow light serves as a warning, and the green light allows traffic to proceed. With advancements in technology, traffic light systems can now be controlled automatically using microcontrollers and integrated with Internet of Things (IoT) concepts to optimize traffic management. In an educational context, understanding these systems can be achieved through simulations using ESP32 as the primary microcontroller. This project aims to simulate the operation of a basic traffic light system using three LEDs—red, yellow, and green—controlled by ESP32. The simulation is developed using Wokwi Simulator within Visual Studio Code (VSCode) with the PlatformIO IDE extension, enabling virtual testing without the need for physical hardware. Through this project, students can gain foundational knowledge in microcontroller programming, electronic device control, and time-based logic implementation in automation systems. Beyond serving as an academic exercise, this simulation can also be a foundation for further development, including research or final projects related to IoT-based traffic management systems.

**1.1 Background of the IoT Practicum (Latar Belakang Praktikum IoT)**

The rapid development of Internet of Things (IoT) technology has led to significant innovations in various fields, including traffic management systems. Smart traffic lights, which integrate IoT-based automation, play a vital role in improving transportation efficiency and reducing congestion. Understanding the fundamentals of microcontroller programming and automated control systems is essential for students who wish to explore IoT applications in real-world scenarios. This practicum focuses on simulating a basic traffic light system using ESP32, a widely used microcontroller in IoT applications. By utilizing Wokwi Simulator within Visual Studio Code (VSCode) and the PlatformIO IDE extension, students can design, implement, and test the system virtually without requiring physical components. This simulation provides an accessible learning experience, allowing students to grasp the core concepts of IoT and automation before moving on to more complex hardware-based implementations. Through this practicum, students will develop skills in microcontroller programming, GPIO control, and time-based logic implementation. Additionally, the project serves as a foundation for understanding how IoT can be applied to traffic management systems, offering insights into potential real-world applications.

**1.2 Experimental Objectives (Tujuan Eksperimen)**

This experiment aims to:

1. Simulate the operation of a basic traffic light system using three LEDs (red, yellow, and green) controlled by ESP32.
2. Implement time-based logic to manage the LED transitions, ensuring the system follows a predefined cycle:
   * Green light: 5000 ms
   * Yellow light: 3000 ms
   * Red light: 1000 ms
3. Utilize Wokwi Simulator as a virtual testing environment, eliminating the need for physical hardware.
4. Enhance students' understanding of IoT concepts, microcontroller programming, and automation principles.
5. Provide a practical foundation for future research or final projects related to IoT-based traffic management systems.

**2. Methodology (Metodologi)**

**2.1 Tools & Materials (Alat dan Bahan)**

This experiment is conducted virtually using the Wokwi Simulator, eliminating the need for physical hardware. The tools and materials used in this project include:

1. Microcontroller: ESP32 (simulated in Wokwi).
2. Output Components:
   * Green LED (indicates "go" signal).
   * Yellow LED (acts as a warning signal).
   * Red LED (indicates "stop" signal).
3. Software & Development Platforms:
   * Visual Studio Code (VSCode) with PlatformIO IDE extension for code development and project management.
   * Wokwi Simulator for virtual ESP32 simulation.
   * C++ Programming Language to control the LED timing on ESP32.
4. Additional Resources:
   * ESP32 documentation and Wokwi reference materials.
   * Developer forums and community discussions for troubleshooting.

**2.2 Implementation Steps (Langkah Implementasi)**

The experiment is implemented through several key steps:

1. Setting Up the Development Environment

* Install Visual Studio Code (VSCode) and add the PlatformIO IDE extension.
* Create a new ESP32 project using PlatformIO.
* Access Wokwi Simulator to design a virtual traffic light system.

1. Designing the Traffic Light System
   * Connect three LEDs (green, yellow, red) to ESP32 GPIO pins virtually in Wokwi.
   * Define the lighting duration for each traffic signal:
     + Green Light: 5000 ms (5 seconds).
     + Yellow Light: 3000 ms (3 seconds).
     + Red Light: 1000 ms (1 second).
2. Writing the Code

The following program, written in C++ using PlatformIO, controls the LED sequence:

#include <Arduino.h>

int red = 13;

int yellow = 14;

int green = 26;

void setup() {

    pinMode(red, OUTPUT);

    digitalWrite(red, LOW);

    pinMode(yellow, OUTPUT);

    digitalWrite(yellow, LOW);

    pinMode(green, OUTPUT);

    digitalWrite(green, LOW);

}

void loop() {

    digitalWrite(red, HIGH);

    delay(1000);

    digitalWrite(red, LOW);

    digitalWrite(yellow, HIGH);

    delay(3000);

    digitalWrite(yellow, LOW);

    digitalWrite(green, HIGH);

    delay(5000);

    digitalWrite(green, LOW);

}

1. Simulation and Testing

* Run the program in Wokwi Simulator.
* Verify that the LEDs switch correctly according to the predefined sequence and duration.
* Identify and fix any coding errors or inconsistencies.

1. Analysis and Improvement

* Debug and refine the code if simulation results deviate from expectations.
* Explore possible improvements, such as adding vehicle detection sensors or IoT-based MQTT communication for real-time traffic control.

**3. Results and Discussion (Hasil dan Pembahasan)**

**3.1 Experimental Results (Hasil Eksperimen)**

The simulation of the traffic light system using ESP32 in Wokwi Simulator was successfully implemented. The results demonstrate the correct operation of the LED sequence, where each light turns on and off according to the predefined timing:

* Green Light: 5000 ms (5 seconds)
* Yellow Light: 3000 ms (3 seconds)
* Red Light: 1000 ms (1 second)

**3.2 Simulation Screenshots (Tangkapan Layar Simulasi)**

Below are the screenshots taken from Wokwi Simulator during the experiment, showing the LED states in different phases of the cycle:

|  |  |
| --- | --- |
| **Traffic Light Phase** | **LED State (Simulation Output)** |
| Green Light (Go) |  |
| Yellow Light (Warning) |  |
| Red Light (Stop) |  |

The LED transitions were successfully simulated and followed the expected cycle, as shown in the screenshots above.

**3.3 Performance Evaluation (Evaluasi Kerja)**

The experiment confirmed the following observations:

1. Accurate Timing Control: The LEDs switched at precise intervals, demonstrating effective implementation of time-based logic.
2. Stable Microcontroller Operation: ESP32 successfully executed the programmed sequence without errors.
3. Simulation Efficiency: The use of Wokwi Simulator allowed for quick debugging and testing without requiring physical hardware.

**3.4 Discussion (Pembahasan)**

The experiment highlights the fundamental principles of microcontroller-based traffic light control. Several key takeaways can be drawn from this simulation:

* Understanding GPIO Control: The experiment reinforced the use of ESP32 GPIO pins to control external components such as LEDs.
* Programming Timing Logic: Implementing delays for different signal durations helped develop time-based automation logic.
* Simulation vs. Real-World Implementation:
  + In a real-world scenario, resistors and transistors would be required to regulate current flow to LEDs.
  + Additional components, such as sensors, could be integrated to optimize traffic flow dynamically.

**4. Appendix (Lampiran)**



  