

# LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

## Praktik Simulasi Lampu Lalu Lintas (Traffic Light) Menggunakan Platform Wokwi dan Visual Studio Code

*Andita Mayla Tifanny*

*Fakultas Vokasi, Universitas Brawijaya*

*Email: [fanfannyy69@gmail.com](mailto:fanfannyy69@gmail.com)*

### **Abstract**

This experiment aims to simulate a traffic light system using the Wokwi platform and Visual Studio Code. The practice is carried out by designing program codes and traffic system diagrams virtually on the Wokwi platform, then testing and simulating using the C/C++ programming language on Visual Studio Code. The results of the experiment show that Wokwi is used in editing program codes and diagrams flexibly, while Visual Studio Code can effectively help to simulate the traffic light system. In conclusion, the combination of these two platforms can facilitate the design, testing, and development of microcontroller-based systems efficiently.

**Keywords**—*Traffic Light, Wokwi, Visual Studio Code, IoT Simulation, Microcontroller*

### **Abstrak**

Eksperimen ini bertujuan untuk mensimulasikan sistem lampu lalu lintas (traffic light) menggunakan platform Wokwi dan Visual Studio Code. Praktik dilakukan dengan merancang kode program dan diagram sistem lalu lintas secara virtual pada platform Wokwi, kemudian menguji dan mensimulasikan menggunakan bahasa pemrograman C/C++ pada Visual Studio Code. Hasil eksperimen menunjukkan bahwa Wokwi digunakan dalam pengeditan kode program dan diagram secara fleksibel, sementara Visual Studio Code secara efektif dapat membantu untuk mensimulasikan sistem traffic light. Kesimpulannya, kombinasi kedua platform ini dapat mempermudah perancangan, pengujian, dan pengembangan sistem berbasis mikrokontroler secara efisien.

**Kata Kunci**—*Lampu Lalu Lintas, Wokwi, Visual Studio Code, Simulasi IoT, Mikrokontroler*

## **1. Introduction (Pendahuluan)**

### **1.1 Latar belakang**

Internet of Things (IoT) semakin berkembang dan diterapkan dalam berbagai aspek kehidupan pada masa kini. Salah satu contoh penerapannya adalah sistem lampu lalu lintas (traffic light) berbasis mikrokontroler yang dapat

disimulasikan secara virtual. Simulasi ini dilakukan dengan menguji kode program tanpa memerlukan perangkat keras, sehingga lebih efisien dalam hal waktu dan biaya.

Praktikum ini menggunakan platform Wokwi dan Visual Studio Code untuk mensimulasikan sistem lampu lalu lintas berbasis mikrokontroler. Wokwi digunakan sebagai editor utama kode program dan diagram, sementara Visual Studio Code membantu untuk mensimulasikan sistem traffic light. Melalui kombinasi kedua platform ini, pengembangan sistem berbasis IoT menjadi lebih mudah, fleksibel, dan efisien.

## **1.2 Tujuan eksperimen**

Tujuan dari eksperimen ini adalah:

1. Merancang kode program dan diagram sistem lalu lintas secara virtual pada platform Wokwi.
2. Mensimulasikan sistem lampu lalu lintas menggunakan platform Visual Studio Code.
3. Mengevaluasi efektivitas penggunaan Wokwi dan Visual Studio Code dalam pengembangan dan simulasi sistem IoT.

## **2. Methodology (Metodologi)**

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

Wokwi, Visual Studio Code, dan web browser (Google Chrome, Mozilla Firefox)

### **2.2 Implementation Steps (Langkah Implementasi)**

#### **1. Perancangan kode program dan diagram di Wokwi**

- Membuka situs resmi [Wokwi](https://wokwi.com) dan membuat proyek baru.
- Merancang diagram dengan menambahkan komponen 3 LED berwarna merah, kuning, dan hijau.
- Menghubungkan 3 LED berwarna merah, kuning, dan hijau dengan mikrokontroler ESP32.
- Menulis logika kode program agar 3 LED berwarna merah, kuning, dan hijau bisa menyala secara bergantian.

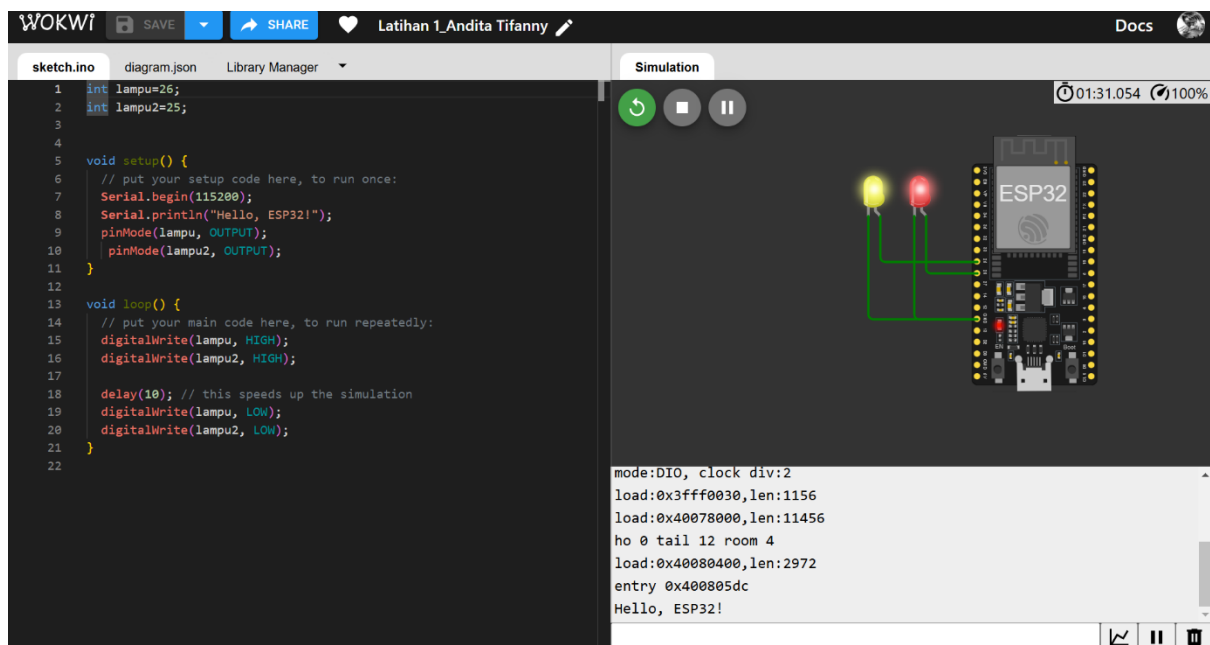
#### **2. Mensimulasikan sistem lalu lintas menggunakan Visual Studio Code**

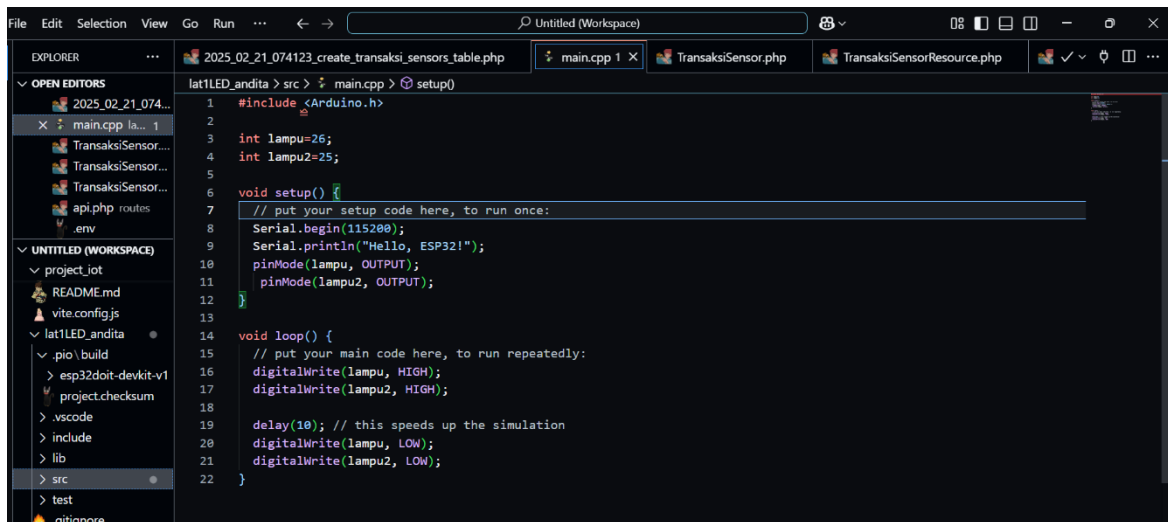
- Membuka Visual Studio Code dan menginstal Wokwi Simulator dan PlatformIO.
- Membuat proyek baru di PlatformIO.
- Menyalin kode program yang telah dibuat pada platform Wokwi ke file main.cpp.
- Melakukan kompilasi kode program main.cpp dengan menekan tombol centang di pojok kanan atas.

- Membuat file dengan nama wokwi.toml.
- Setelah kompilasi berhasil, menyalin relative path file firmware.bin dan firmware.elf ke dalam file wokwi.toml.
- Membuat file diagram.json dan menyalin kode diagram.json pada Wokwi.
- Sebelum memulai simulasi, harus menjalankan perintah > Wokwi: Request a New License.
- Terakhir, melakukan simulasi sistem lampu lalu lintas (traffic light) dengan cara mengetik command > Wokwi: Start Simulator.

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

#### 3.1 Experimental Results (Hasil Eksperimen)





The screenshot shows the Visual Studio Code editor with a workspace named 'lat1LED\_andita'. The Explorer sidebar on the left shows the project structure, including files like 'main.cpp', 'TransaksiSensor.php', and 'TransaksiSensorResource.php'. The main editor window displays the 'main.cpp' file, which is an Arduino sketch. The code includes the Arduino.h header, defines two pins (lampu=26 and lampu2=25), and contains setup() and loop() functions. The setup() function initializes the serial port at 115200 baud and prints 'Hello, ESP32!'. The loop() function writes HIGH to both pins, delays for 10ms, and then writes LOW to both pins.

```
1 #include <Arduino.h>
2
3 int lampu=26;
4 int lampu2=25;
5
6 void setup() {
7     // put your setup code here, to run once:
8     Serial.begin(115200);
9     Serial.println("Hello, ESP32!");
10    pinMode(lampu, OUTPUT);
11    pinMode(lampu2, OUTPUT);
12 }
13
14 void loop() {
15     // put your main code here, to run repeatedly:
16     digitalWrite(lampu, HIGH);
17     digitalWrite(lampu2, HIGH);
18
19     delay(10); // this speeds up the simulation
20     digitalWrite(lampu, LOW);
21     digitalWrite(lampu2, LOW);
22 }
```



The screenshot shows the Visual Studio Code editor with the same workspace. The Explorer sidebar now shows 'diagram.json' as the active file. The main editor window displays the content of 'diagram.json', which is a JSON object representing a circuit diagram. It includes a version, author, editor, and a list of parts. The parts list includes an ESP32 board and two LEDs (one red, one yellow). The connections list defines the wiring between the board pins and the LEDs.

```
1 {
2     "version": 1,
3     "author": "Andita Mayla Tifanny",
4     "editor": "wokwi",
5     "parts": [
6         { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {} },
7         { "type": "wokwi-led", "id": "led1", "top": 25.2, "left": -63.4, "attrs": { "color": "red" } },
8         {
9             "type": "wokwi-led",
10            "id": "led2",
11            "top": 25.2,
12            "left": -101.8,
13            "attrs": { "color": "yellow" }
14        }
15    ],
16    "connections": [
17        [ "esp:TX", "$serialMonitor:RX", "", [] ],
18        [ "esp:RX", "$serialMonitor:TX", "", [] ],
19        [ "led1:A", "esp:26", "green", [ "v48", "h38.4" ] ],
20        [ "led1:C", "esp:GND.1", "green", [ "v86.4", "h58" ] ],
21        [ "led2:A", "esp:25", "green", [ "v38.4", "h86.4" ] ],
22        [ "led2:C", "esp:GND.1", "green", [ "v86.4", "h86.8" ] ]
23    ],
24    "dependencies": {}
25 }
```