LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Real Hardware ESP32 menggunakan LED**

*Ricelvin Candra Hagana*

*Fakultas Vokasi, Universitas Brawijaya*

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

***Abstract***

Praktikum ini bertujuan untuk mempelajari cara kerja mikrokontroler **ESP32** dalam mengontrol perangkat output sederhana berupa **LED** secara langsung dengan menggunakan **bahasa pemrograman Arduino (C++)**. Praktikum dilakukan menggunakan **hardware ESP32**, **breadboard**, dan **komponen LED** untuk mengamati proses digital output secara real-time. Dalam eksperimen ini, mahasiswa mengunggah program dari komputer ke ESP32 menggunakan **Arduino IDE**, lalu mengontrol LED dengan memberikan sinyal HIGH dan LOW pada pin tertentu.

Proses ini mencakup pemahaman konfigurasi pin, delay waktu, serta logika kontrol dasar. Praktikum juga mencakup pengenalan pemrograman dasar pada ESP32, termasuk penggunaan digitalWrite(), pinMode(), dan delay(). Dengan pendekatan hands-on, praktikum ini memberikan pemahaman menyeluruh tentang bagaimana sistem IoT dapat berinteraksi dengan dunia nyata melalui aktuator sederhana seperti LED.

Hasil eksperimen menunjukkan bahwa ESP32 mampu mengendalikan LED sesuai dengan instruksi yang diprogramkan. Praktikum ini menjadi fondasi penting untuk memahami dasar-dasar sistem kendali digital menggunakan mikrokontroler IoT secara langsung.

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

***1.1. Latar Belakang***

ESP32 merupakan salah satu mikrokontroler populer dalam dunia Internet of Things (IoT) karena memiliki kemampuan Wi-Fi dan Bluetooth terintegrasi serta harga yang terjangkau. Salah satu dasar dalam memahami ESP32 adalah belajar bagaimana mengontrol komponen output seperti LED.

Pengendalian LED sangat penting sebagai langkah awal pembelajaran dalam membangun sistem otomasi dan IoT. Meskipun terlihat sederhana, proses ini melibatkan pengaturan pindigital, logika program, serta pemahaman tentang delay dan loop dalam pemrograman mikrokontroler.

Melalui praktik ini, mahasiswa akan memperoleh pengalaman langsung menggunakan perangkat keras dan menguji fungsionalitas dasar ESP32 sebagai pengontrol output dalam sistem IoT. Kegiatan ini juga memperkenalkan penggunaan **Arduino IDE** sebagai platform utama untuk menulis dan mengunggah program ke ESP32.

* 1. ***Tujuan Eksperimen***
* Memahami cara kerja dasar mikrokontroler ESP32.
* Mengontrol LED menggunakan kode program berbasis Arduino.
* Menghubungkan dan menguji rangkaian ESP32 dan LED secara real hardware.
* Mengembangkan kemampuan analisis terhadap sistem digital output sederhana.

1. ***Methodology (Metodologi)***

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

* ESP32 Development Board
* Breadboard
* LED (merah)
* Resistor 220Ω
* Kabel jumper
* Laptop dengan Arduino IDE
* Kabel USB Micro untuk menghubungkan ESP32 ke laptop

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

1. ***Persiapan Alat***
   * Siapkan ESP32, LED, resistor, breadboard, dan kabel jumper.
   * Hubungkan ESP32 ke laptop menggunakan kabel USB.
2. ***Instalasi Board ESP32 di Arduino IDE***
   * Buka Arduino IDE.
   * Tambahkan board ESP32 melalui Board Manager*.*
3. ***Rangkaian LED***
   * Hubungkan pin **GPIO 2** (D2) pada ESP32 ke anoda LED.
   * Hubungkan katoda LED ke resistor 220Ω, lalu ke GND.
4. ***Kode Program***
5. #include <Arduino.h>  // Wajib untuk PlatformIO + ESP32
7. // Deklarasi pin LED
8. int lampu = 18;
9. int lampu2 = 20;
11. void setup() {
12. Serial.begin(115200);  // Inisialisasi komunikasi Serial
13. Serial.println("ESP32 Blinking LED");
15. // Atur pin sebagai OUTPUT
16. pinMode(lampu, OUTPUT);
17. pinMode(lampu2, OUTPUT);
18. }
20. void loop() {
21. // Nyalakan kedua LED
22. digitalWrite(lampu, HIGH);
23. digitalWrite(lampu2, HIGH);
24. Serial.println("LED ON");
26. delay(1000); // Tunggu 1 detik
28. // Matikan kedua LED
29. digitalWrite(lampu, LOW);
30. digitalWrite(lampu2, LOW);
31. Serial.println("LED OFF");
33. delay(1000); // Tunggu 1 detik sebelum mengulang
34. }

*5. Lampiran*

