LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Simulasi Relay, Button & LED**

*Assyfaul Khoiriyah*

Fakultas Vokasi, Universitas Brawijaya

[assyfaassyfa0@gmail.com](mailto:assyfaassyfa0@gmail.com)

**Abstract (Abstrak)**

Internet of Things (IoT) memungkinkan interaksi antar perangkat melalui jaringan. Pada praktikum ini, dilakukan simulasi penggunaan ESP32 untuk mengontrol LED dan modul relay menggunakan push button. System dirancang dengan input dari push button yang mengaktfikan LED dan relay ketika ditekan. Implementasi dilakukan menggunakan wokwi simulayor untuk memvalidasi fungsionalittas. Hasil simulasi menunjukkan bahwa system berfungsi sesuai dengan yang diharapkan, dimana LED dan relay aktif saat tombol ditekan dan nonaktif saat dilepas. Praktikum ini memberikan peahaman dasar tentang pengendalian perangkat elektronik berbasis IoT.

Kata kunci: Internet of Things, ESP32, relay, push button, LED

1. **Introduction (Pendahuluan)**
   1. **Latar Belakang**

Perkembangan teknologi Internet of Things (IoT) telah membawa perubahan yang signifikan dalam cara perangkat elektronik beroperasi dan berinteraksi satu sama lain. IoT memungkinkan perangkat untuk saling terhubung dan berkomunikasi melalui jaringan, sehingga dapat dikendalikan dan dimonitor secara otomatis dari jarak jauh. Teknologi ini telah diterapkan di berbagai bidang, termasuk rumah pintar, industri otomatis, kesehatan, dan transportasi.

Dalam dunia otomasi, mikrokontroler seperti ESP32 memiliki peran penting dalam menghubung berbagai perangkat, seperti sensor dan akuator dalam sebuah ekosistem IoT. Pada praktikum ini, dilakukan simulasi penggunaan ESP32 untuk mengontrol LED dan relay menggunakan push button. Simulasi ini bertujuan untuk memahami bagaimana ESP32 membaca input dari tombol dan mengaktifkan perangkat keluaran berdasarkan kondisi yang diterima

**1.2 Tujuan Eksperimen**

Tujuan dari eksperimen ini adalah:

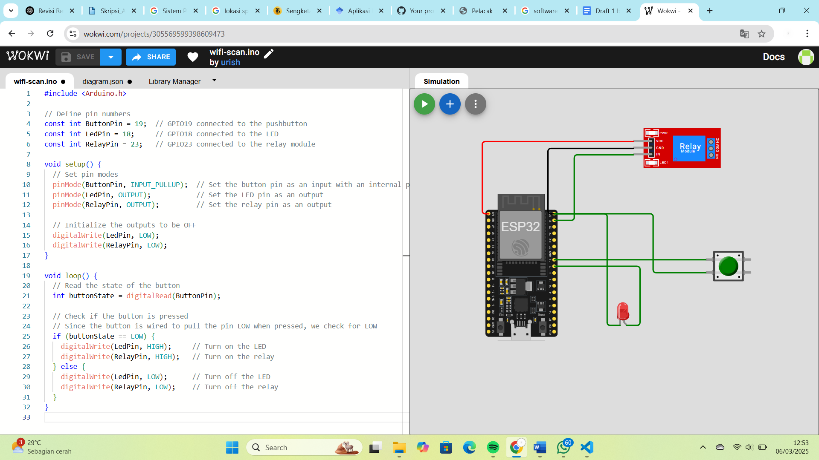
1. Memahami dasar penggunaan esp32 dalam sistem IoT
2. Mengetahui cara kerja push button sebagai input digital
3. Menguji integrasi LED dan relay dalam simulasi berbasis ESP32
4. **Methodology (Metodologi)**

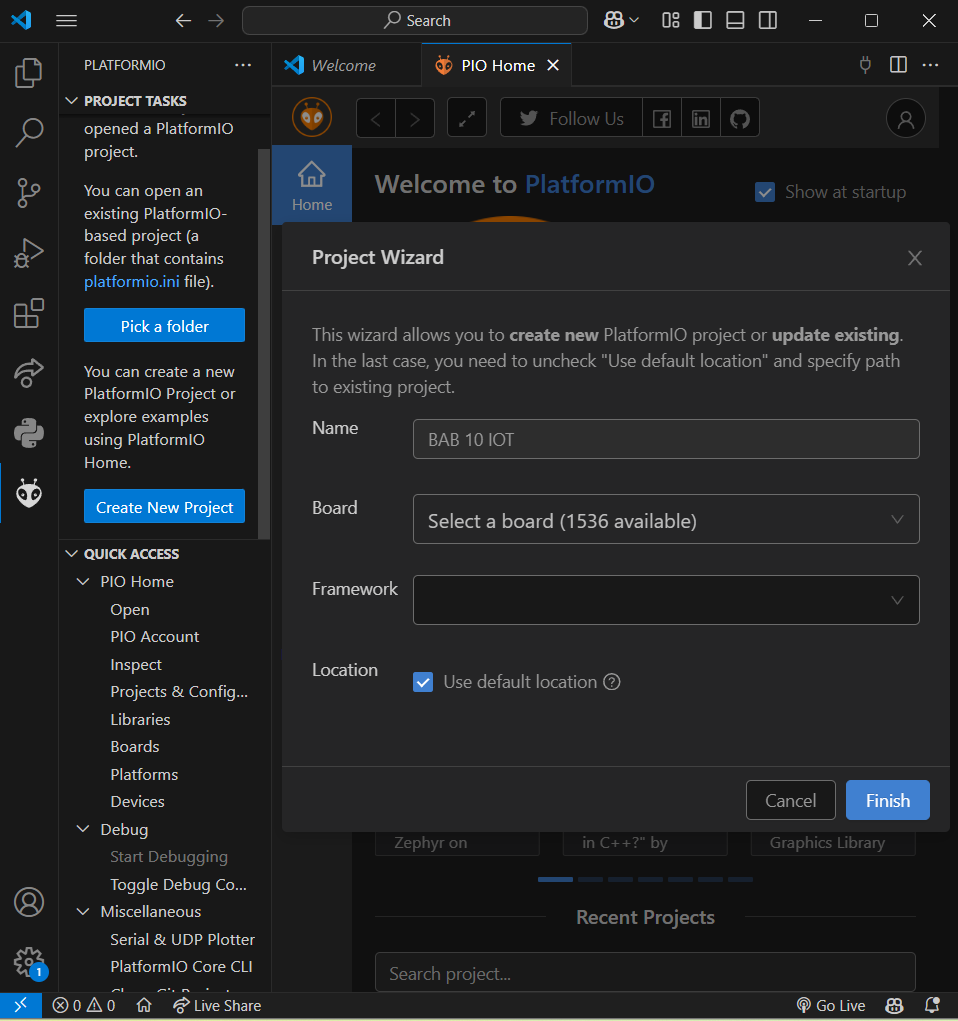
**2.1 Tools & Materials (Alat dan Bahan)**

* ESP32
* Push button
* LED
* Resistor
* Modul relay
* Wokwi simulator

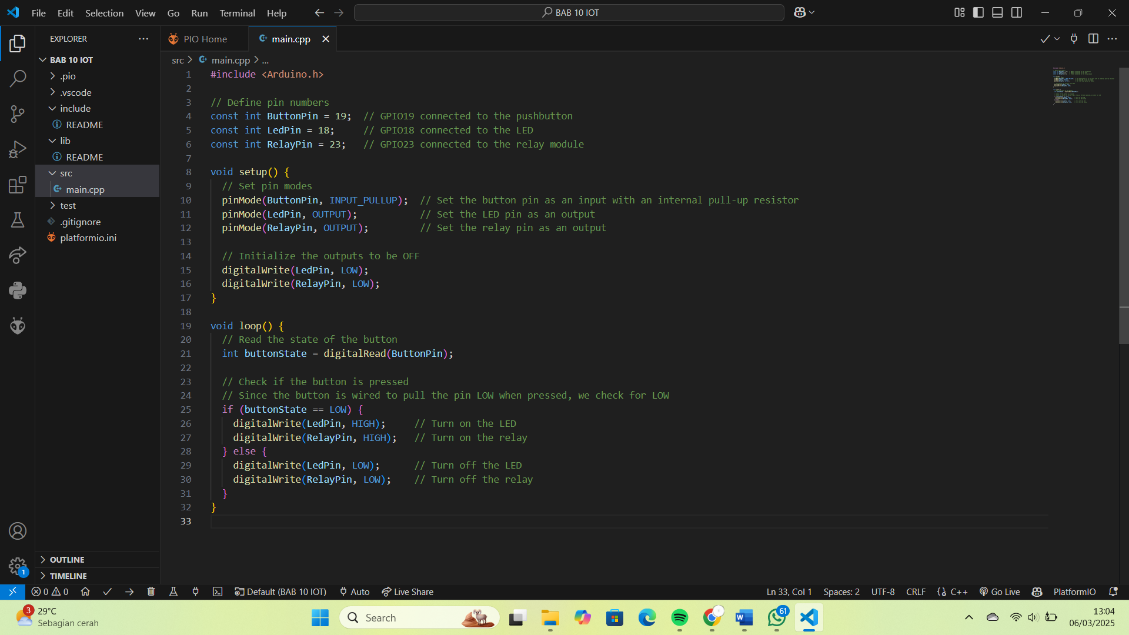
**2.2 Implementation Steps (Langkah Implementasi)**

1. Buka web wokwi dan buka diagramnya
2. Lalu masukkan codenya

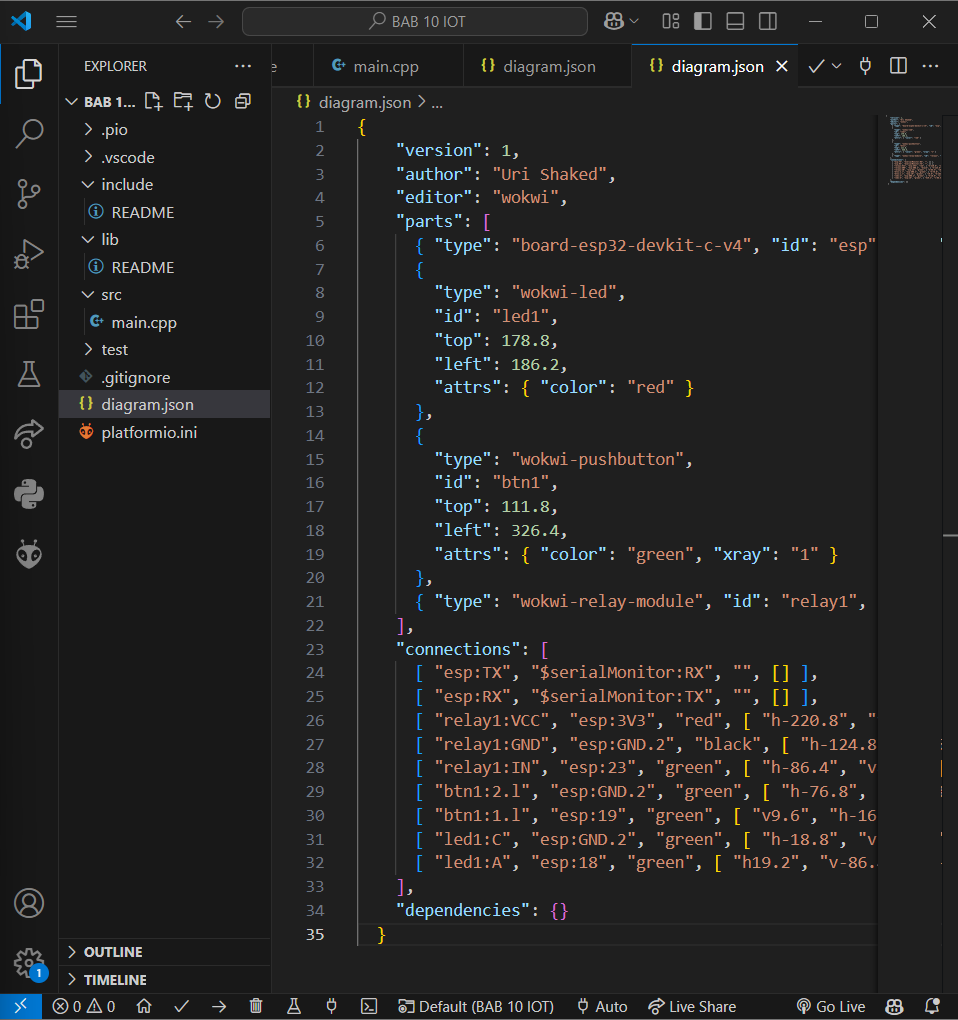


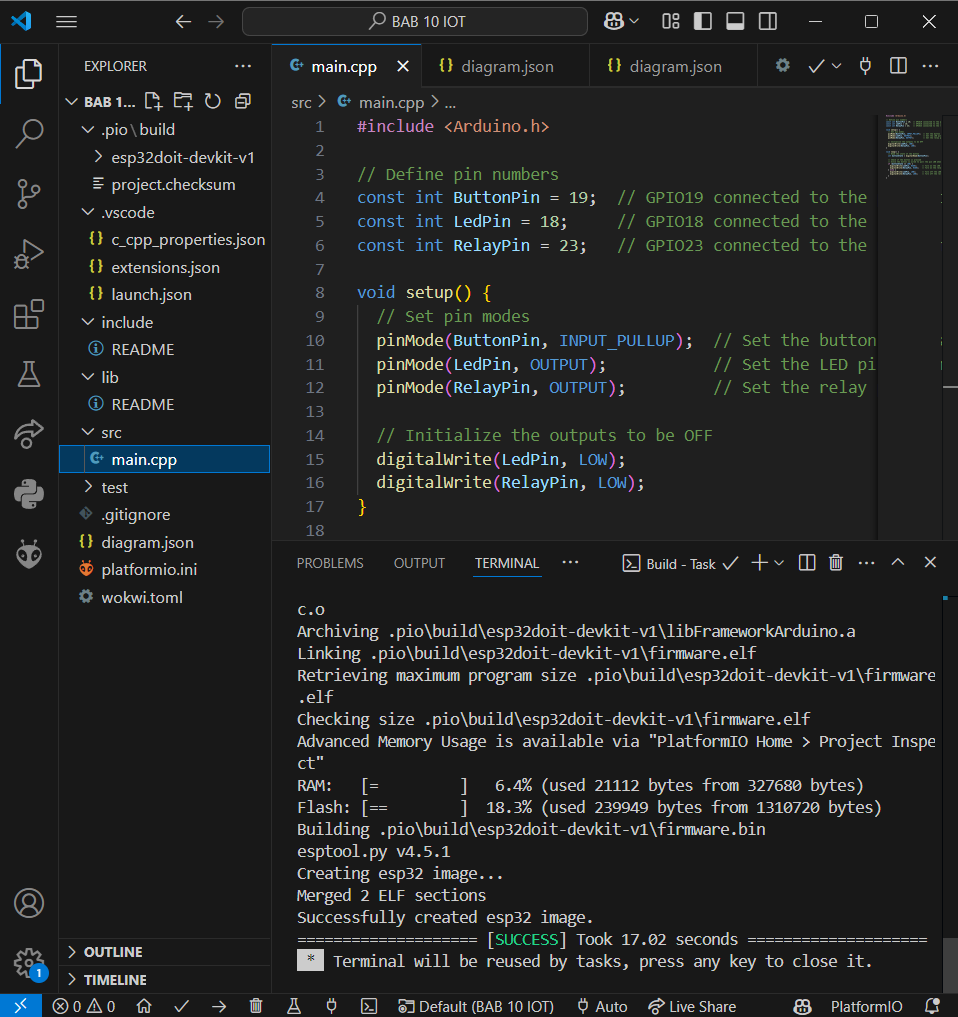
1. Buat project baru pada platform IO

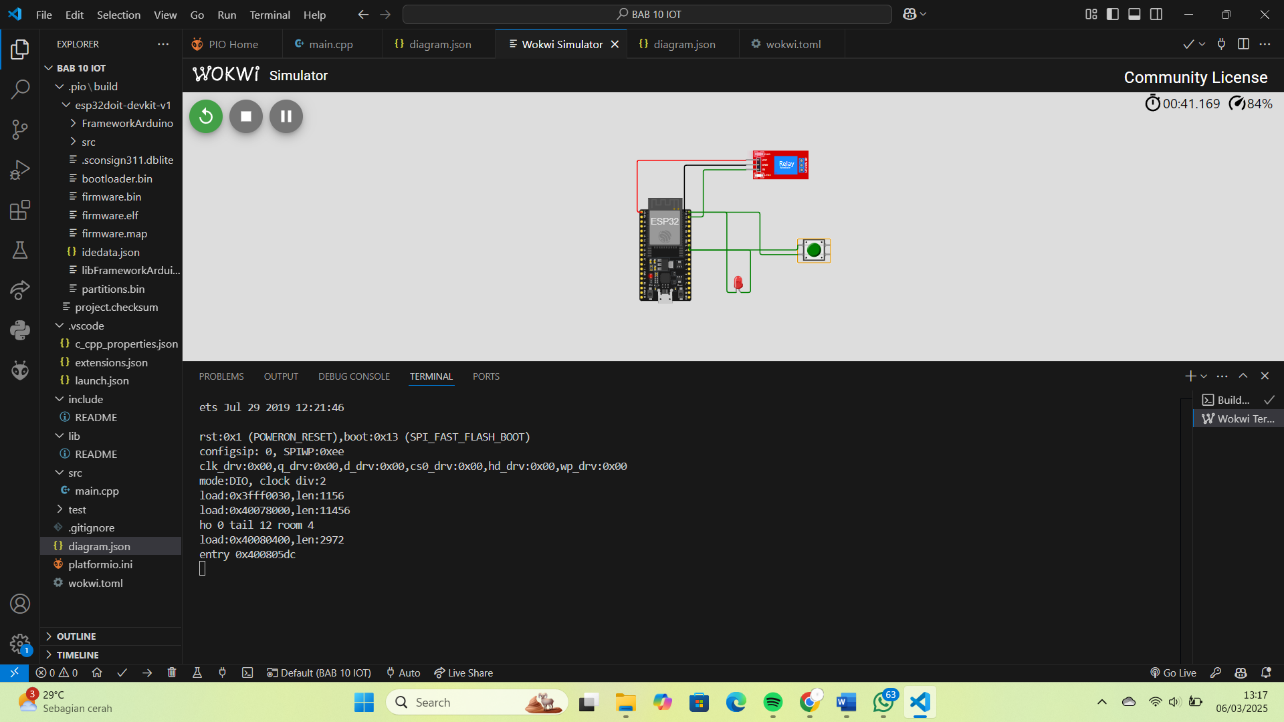
1. Tuliskan code pada file main.cpp nya



1. Lalu edit pada file platfrom.io
2. Buat file diagram.json dan copypaste kan dari diagram.json pada web wokwi.com



1. Buat file wokwi.toml dan masukkan code nya
2. Lalu compile file pada main.cpp
3. Kemudian lakukan request license ke wokwi.com
4. Jalankan simulasi dengan perintah wokwi: start simulator



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

**3.1 Experimental Results (Hasil Eksperimen)**

Dari hasil simulasi yang dilakukan di wokwi, diperoleh bahwa:

* Saat tomtol ditekan, LED menyala dan relay aktif
* Sata tombol dilepas, LED mati dan relay nonaktif
* Sistem bekerja sesuai dengan logika yang dirancang

Eksperimen ini menunjukkan bahwa ESP32 dapat digunakan utuk mengontrol perangkat elektronik dengan input sederhana seperti push button.

1. **Appendix (Lampiran, jika diperlukan)**

#include <Arduino.h>

// Define pin numbers

const int ButtonPin = 19;  // GPIO19 connected to the pushbutton

const int LedPin = 18;     // GPIO18 connected to the LED

const int RelayPin = 23;   // GPIO23 connected to the relay module

void setup() {

  // Set pin modes

  pinMode(ButtonPin, INPUT\_PULLUP);  // Set the button pin as an input with an internal pull-up resistor

  pinMode(LedPin, OUTPUT);           // Set the LED pin as an output

  pinMode(RelayPin, OUTPUT);         // Set the relay pin as an output

  // Initialize the outputs to be OFF

  digitalWrite(LedPin, LOW);

  digitalWrite(RelayPin, LOW);

}

void loop() {

  // Read the state of the button

  int buttonState = digitalRead(ButtonPin);

  // Check if the button is pressed

  // Since the button is wired to pull the pin LOW when pressed, we check for LOW

  if (buttonState == LOW) {

    digitalWrite(LedPin, HIGH);     // Turn on the LED

    digitalWrite(RelayPin, HIGH);   // Turn on the relay

  } else {

    digitalWrite(LedPin, LOW);      // Turn off the LED

    digitalWrite(RelayPin, LOW);    // Turn off the relay

  }

}