LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Real Hardware ESP32**

*Adellia Eka Putri – 233140701111026*

*Fakultas Vokasi, Universitas Brawijaya*

*Email: adelliaeka10@gmail.com*

**Abstract** (Abstrak)

Praktik real hardware ESP32 dengan integrasi modul Silicon Labs menawarkan solusi efisien dalam pengembangan sistem embedded dan IoT. Penelitian ini berfokus pada implementasi ESP32 bersama komponen Silicon Labs seperti sensor dan transceiver untuk aplikasi akuisisi data, kontrol perangkat, dan komunikasi nirkabel. Metode yang digunakan meliputi pemrograman ESP32 melalui Arduino IDE tanpa menggunakan ESP-IDF, serta pengujian interoperabilitasnya dengan chip Silicon Labs. Hasil eksperimen menunjukkan bahwa kombinasi ESP32 dan Silicon Labs mampu memberikan performa yang stabil dalam transmisi data, efisiensi daya yang baik, serta respons cepat dalam aplikasi real-time. Namun, tantangan seperti sinkronisasi antarmuka komunikasi dan optimasi firmware perlu diperhatikan. Studi ini memberikan panduan praktis bagi pengembang dalam memadukan ESP32 dengan solusi Silicon Labs untuk sistem IoT yang andal dan hemat energi.

*Kata kunci— ESP32, Silicon Labs, IoT, hardware, Arduino IDE.*

1. **Pendahuluan**
2. **Latar Belakang**

Perkembangan teknologi Internet of Things (IoT) semakin pesat, mendorong kebutuhan akan perangkat embedded yang hemat daya, andal, dan terintegrasi dengan berbagai komponen elektronik. ESP32, sebagai salah satu mikrokontroler populer, banyak digunakan karena kemampuannya dalam konektivitas Wi-Fi dan Bluetooth serta dukungan pemrograman yang fleksibel. Namun, dalam beberapa aplikasi IoT, ESP32 perlu diintegrasikan dengan komponen eksternal seperti sensor, transceiver, atau modul manajemen daya dari produsen lain, salah satunya Silicon Labs. Silicon Labs dikenal dengan solusi komunikasi nirkabel dan sensor yang efisien, sehingga kombinasi antara ESP32 dan chip Silicon Labs dapat menjadi solusi optimal untuk sistem IoT yang lebih kompleks.

Penggunaan Arduino IDE sebagai platform pengembangan memudahkan prototipe tanpa perlu menggunakan framework seperti ESP-IDF, sehingga cocok untuk pengembang pemula maupun profesional. Namun, integrasi antara ESP32 dan Silicon Labs memerlukan analisis mendalam terkait kompatibilitas hardware, manajemen komunikasi (SPI, I2C, UART), serta optimasi konsumsi daya. Oleh karena itu, penelitian ini dilakukan untuk mengevaluasi praktik nyata penggabungan ESP32 dengan komponen Silicon Labs dalam berbagai skenario IoT.

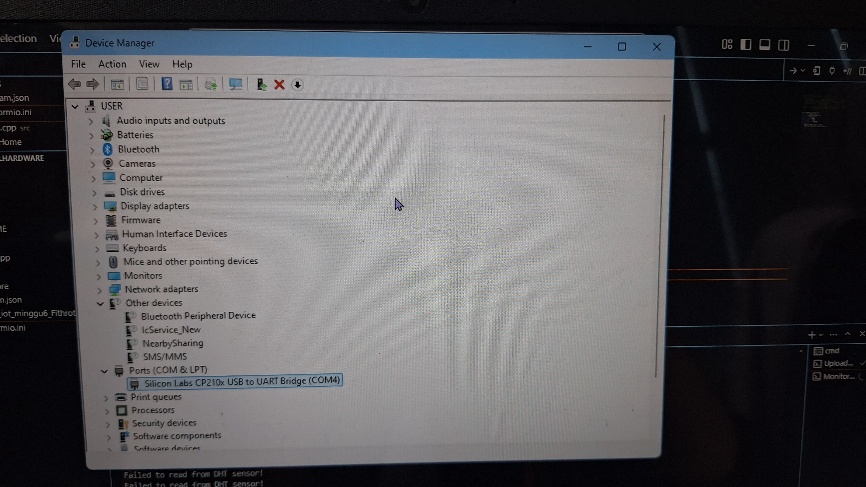
1. **Tujuan Eksperimen**

Praktikum ini memiliki beberapa tujuan utama:

1. Mengimplementasikan integrasi ESP32 dengan modul Silicon Labs
2. Menganalisis kinerja system
3. Mengidentifikasi tantangan teknis seperti sinkronisasi antarmuka, manajemen daya.
4. **Metodologi**
5. **Alat dan Bahan**

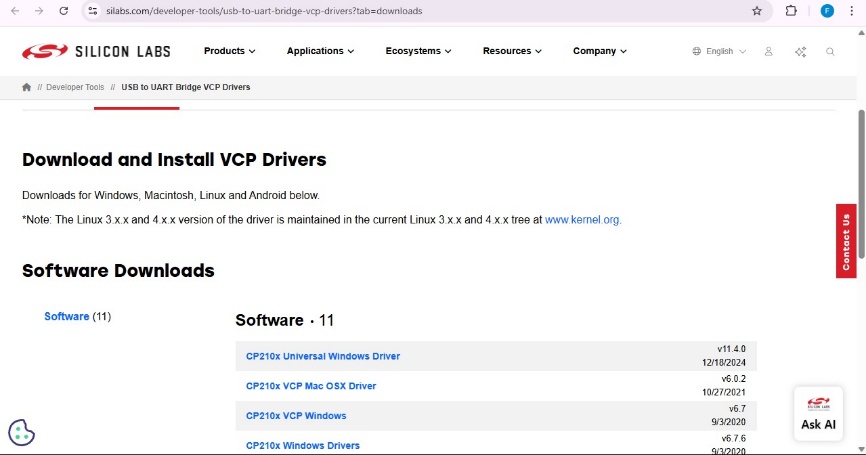
ESP32, Breadboard, Kabel jumper, Lampu LED, Kabel micro USB.

1. **Langkah Implementasi**
2. Pastikan Hardware ESP32 dikenali oleh computer

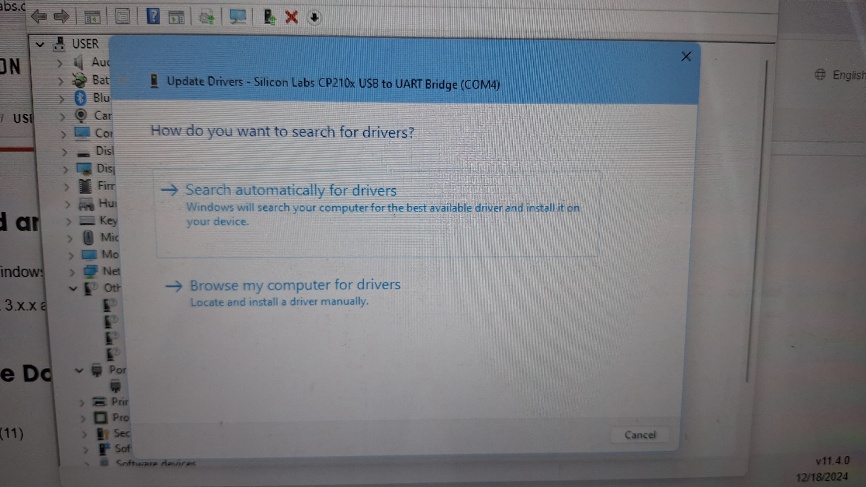


1. Perhatikan bagian Ports (COM & LPT) harus muncul **Silicon Labs…..** Jika belum muncul harus dilakukan proses instalasi driver secara manual mengikuti tutorial sebagai berikut :

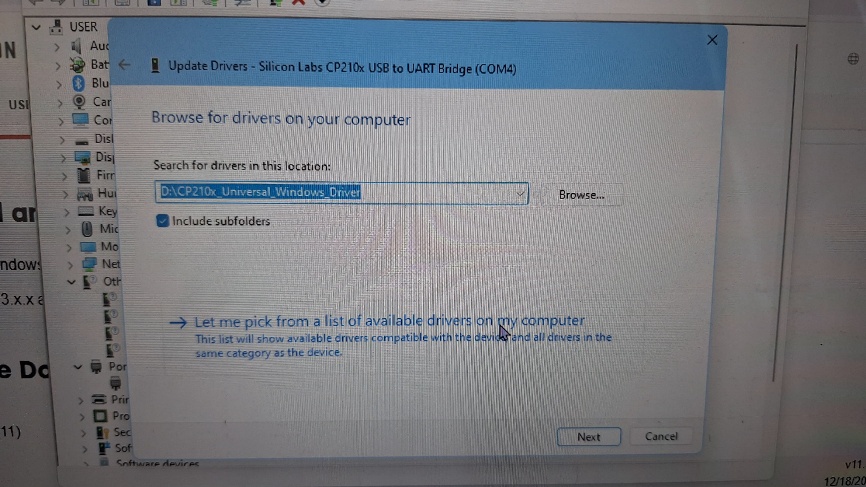
Download Driver Silicon Labs CP210x di alamat : <https://www.silabs.com/developer-tools/usb-to-uart-bridge0-vcp-drivers?tab=downloads>



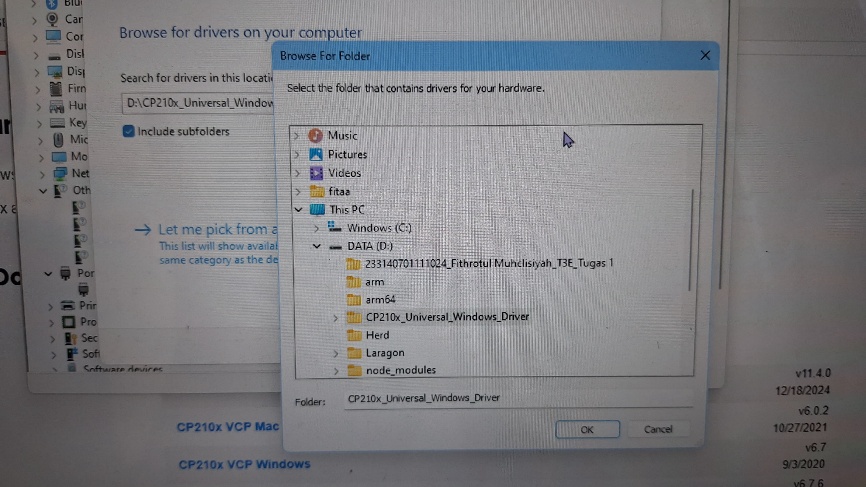
1. Klik kanan Update Driver
2. Pilih Browse my computer for drivers



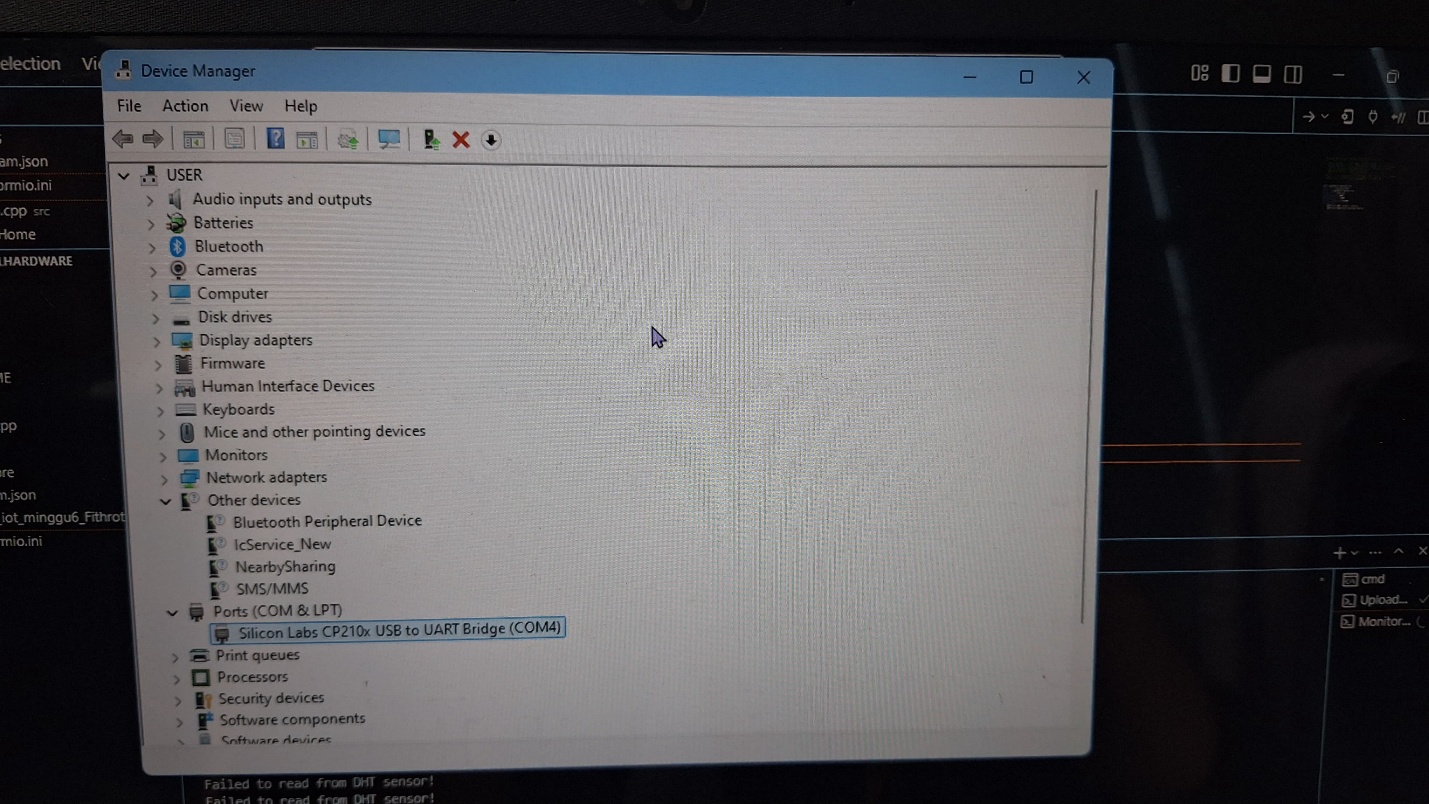
1. Pilih ***Let me pick from a list of available drivers on my computer***



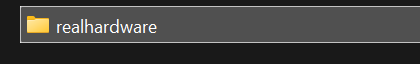
1. Klik menu **Browse**, arahkan ke folder driver yang telah di download (harus di extract folder)



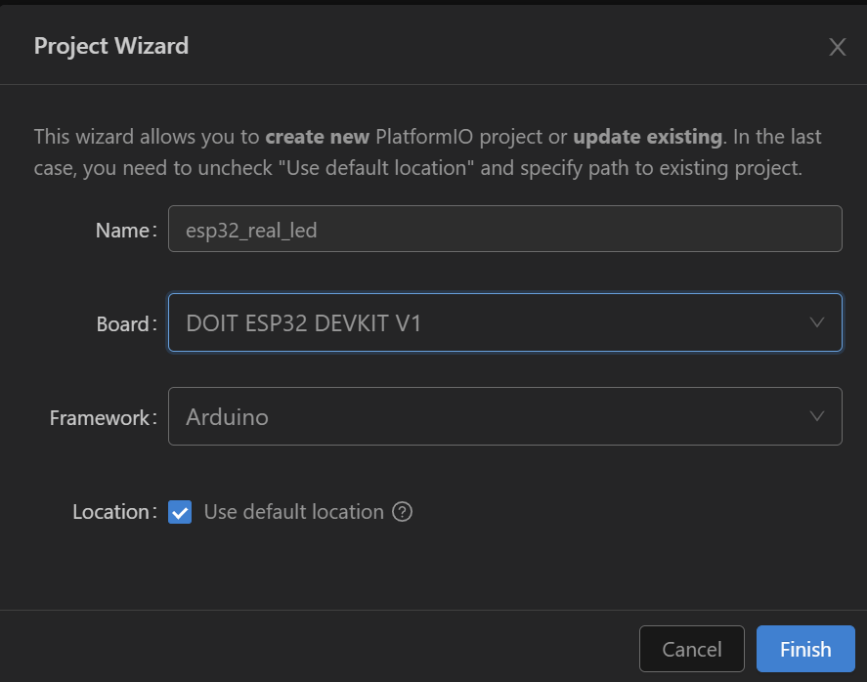
1. Setelah dipilih, klik next dan proses instalasi akan berjalan. Pastikan tampilan device manager tidak menunjukkan masalah dan hardware ESP32 terkenali dengan baik seperti tampilan berikut :



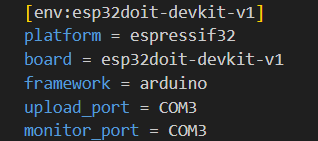
1. Buat folder baru dan buka di vscode



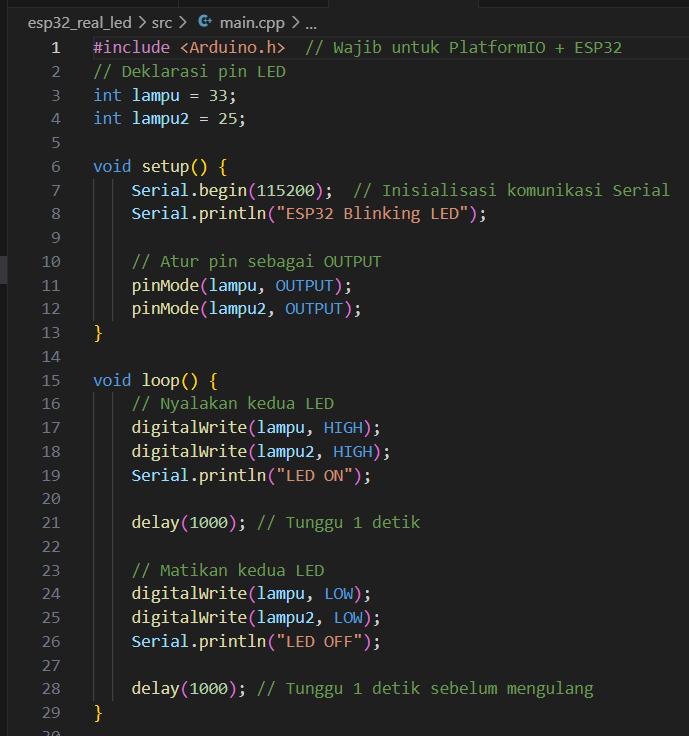
1. Pastikan device ESP32 muncul pada platform.io
2. Buat project baru di platformio



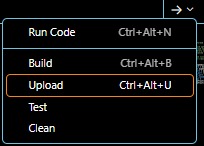
1. Ubah file platformio.ini  dan modifikasi sebagai berikut



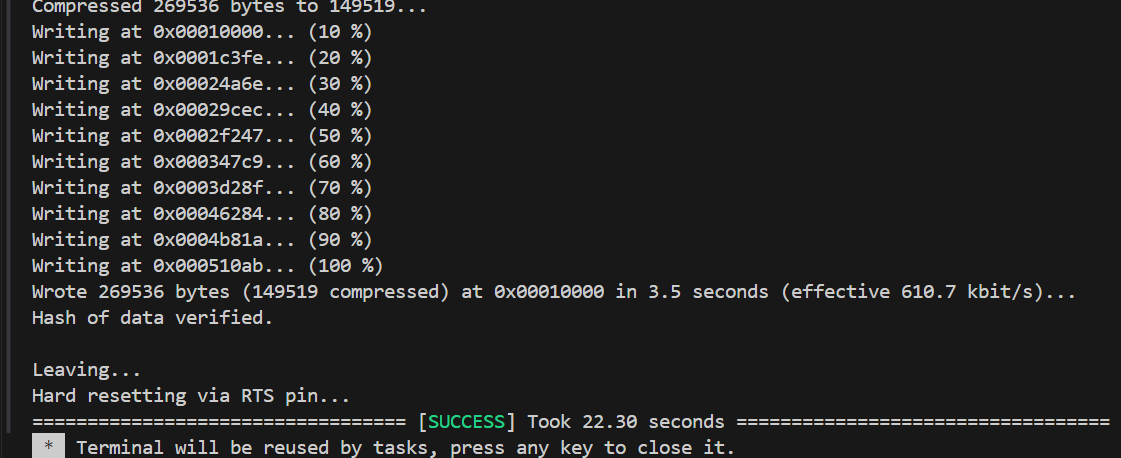
1. Kemudian pada file main.cpp   masukkan koding lampu LED yang telah dibuat



1. Kemudian lakukan Upload pada menu **Upload**



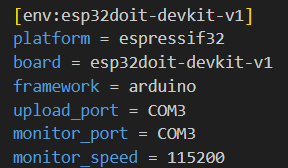
1. Proses compiling dan upload akan berjalan dan pastikan berhasil seperti tampilan berikut



1. Pada latihan berikutnya perlu dilakukan upload program untuk mengecek apakah hardware ESP32 dapat terhubung ke Access Point WIFI disekitar. Untuk melakukannya perlu melakukan perubahan koding pada file main.cpp seperti berikut ini :



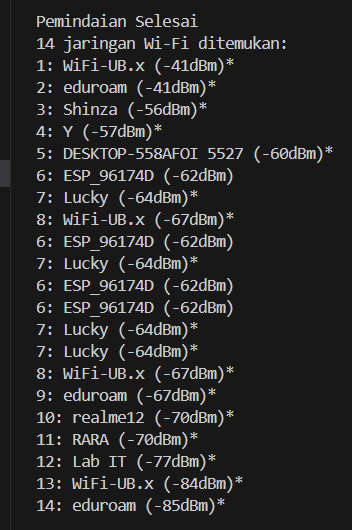
1. Lakukan proses upload, Ubah kembali file platformio.ini sebagai berikut



1. Kemudian klik tombol serial monitor



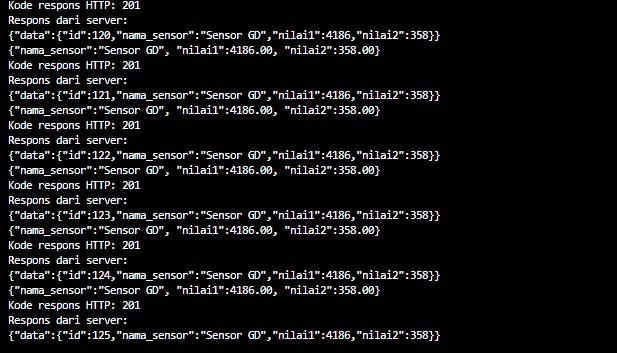
1. Pastikan tampilan serial monitor menunjukkan nama Access Point WIFI disekitar berikut juga dengan keterangan kekuatan sinyal seperti tampilan berikut ini

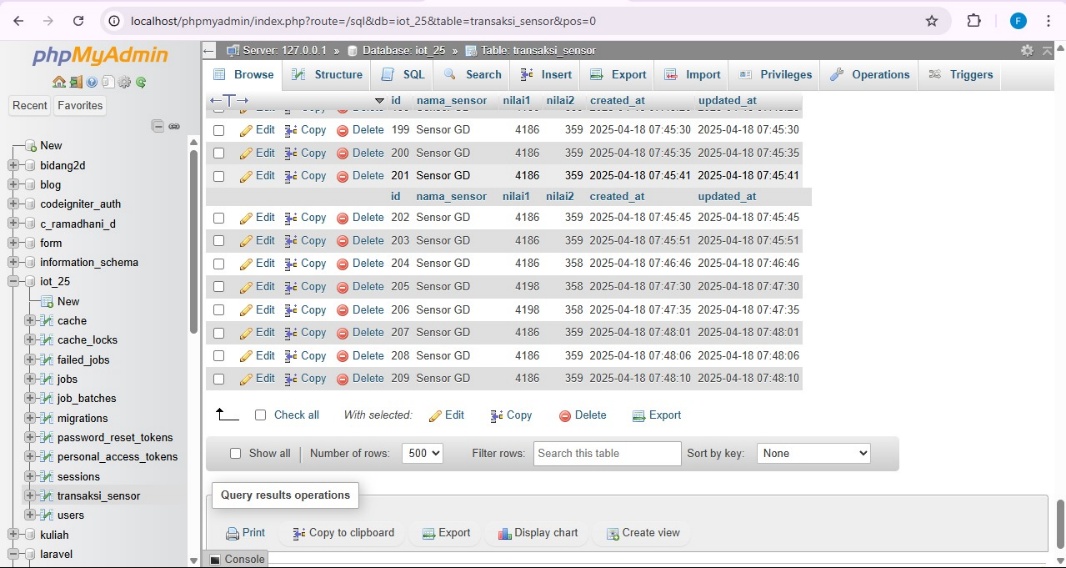


1. Jalankan API laravel kembali dengan perintah **php artisan serve --host=0.0.0.0 --port=8000**
2. Kemudian jalankan NGROK **ngrok http --scheme=http 8000**
3. Lakukan modifikasi file main.cpp sebagai berikut
4. Kemudian lakukan modifikasi kembali pada file platformio.ini dengan tambahan baris sebagai berikut

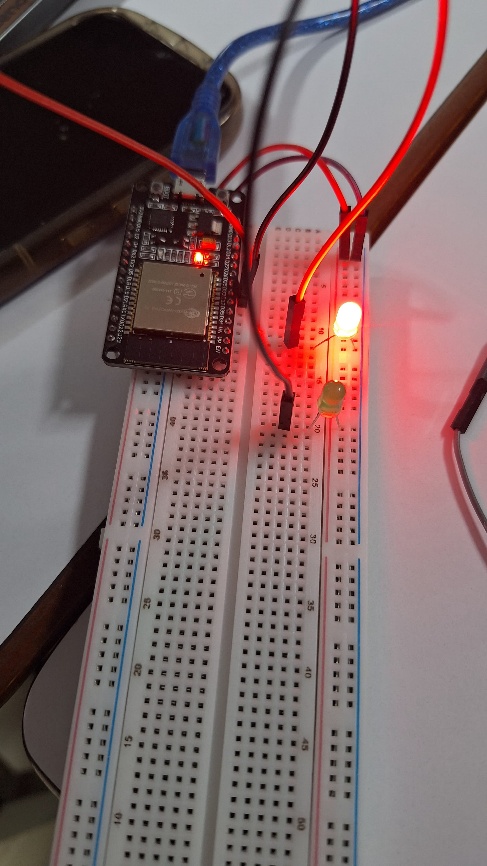


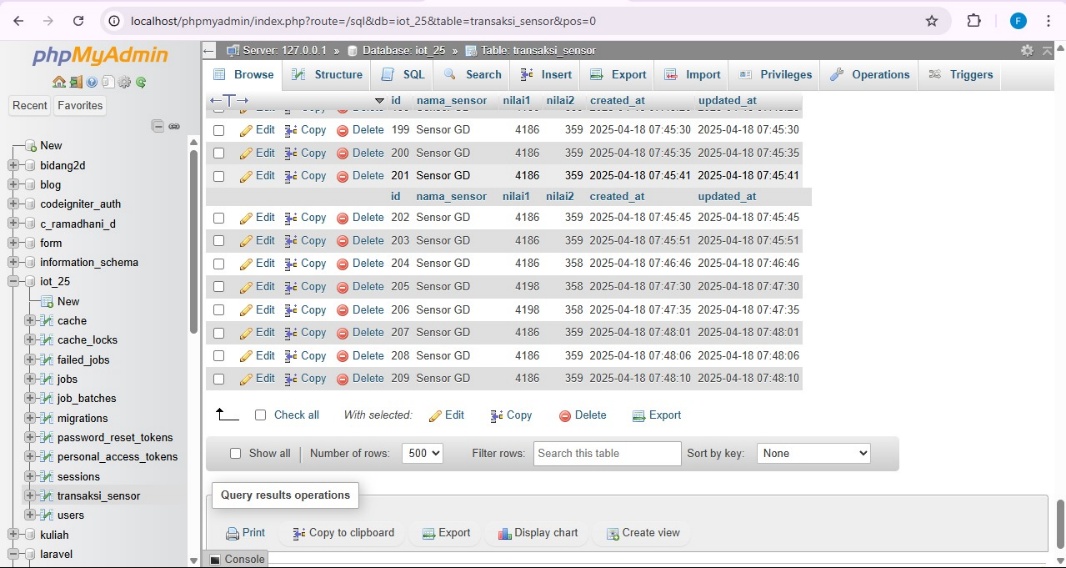
1. Lakukan proses upload. Kemudian jalankan simulasi. Pastikan data yang dikirim dari hardware ESP32 dapat masuk ke database.





1. **Hasil dan Pembahasan**
2. **Hasil Eksperimen**





1. **Lampiran (jika diperlukan)**

