**Praktik Pembuatan Akun Wokwi dan Github**

**Disusun Untuk Memenuhi Tugas Mata Kuliah :**

**Internet Of Things**

**Dosen Pengampu : Rachmad Andri Atmoko, S.ST, M.T, MCF**



**Disusun Oleh :**

1. Rifqi Athallah G.R(233140701111008)

PROGRAM STUDI TEKNOLOGI INFORMASI(

FAKULTAS VOKASI

UNIVERSITAS BRAWIJAYA

2025

**Abstract**

Percobaan ini bertujuan untuk memahami proses pembuatan akun *GitHub* dan *Wokwi* serta mengintegrasikan keduanya dalam simulasi *Internet of Things (IoT)*. Langkah pertama dalam percobaan ini adalah membuat akun *GitHub* pribadi dan menggunakannya untuk *sign up* pada *Wokwi*. Setelah berhasil masuk, pengguna dapat mencoba berbagai *template* simulasi yang tersedia di *Wokwi* serta melakukan eksperimen dengan menambahkan komponen yang diperlukan.

Setelah melakukan percobaan simulasi di situs *Wokwi*, langkah selanjutnya adalah mengunduh file konfigurasi **diagram.json** dan membukanya di *Visual Studio Code (VS Code)*. Dalam implementasi ini, tahap awal yang dilakukan adalah menginstal ekstensi *PlatformIO* dan *Wokwi Simulator* pada *VS Code*. Setelah itu, pengguna membuat proyek baru di *PlatformIO* dengan menentukan nama proyek, memilih *board* **DOIT ESP32 Devkit V1**, serta menggunakan *framework* **Arduino** untuk mengembangkan sistem berbasis mikrokontroler.

Hasil dari percobaan ini menunjukkan bahwa proses pembuatan akun *GitHub* dan *Wokwi* dapat dilakukan dengan mudah serta memungkinkan pengguna untuk melakukan simulasi proyek *IoT* secara efektif. Selain itu, percobaan ini juga membuktikan bahwa integrasi antara *Wokwi* dan *VS Code* melalui *PlatformIO* dapat mendukung pengembangan serta pengujian perangkat lunak berbasis ESP32 secara efisien sebelum diterapkan pada perangkat keras sebenarnya.

# **BAB I**

# **PENDAHULUAN**

## **1.1** **Latar Belakang**

## Dalam era digital saat ini, *Internet of Things (IoT)* menjadi salah satu teknologi yang berkembang pesat dan banyak diterapkan dalam berbagai bidang, termasuk otomatisasi, pemantauan, dan kontrol sistem berbasis mikrokontroler. Untuk mendukung pengembangan dan simulasi sistem *IoT*, diperlukan platform yang dapat mempermudah proses perancangan dan pengujian sebelum implementasi pada perangkat keras sebenarnya. Salah satu platform yang sering digunakan adalah *Wokwi*, yang memungkinkan simulasi berbasis web untuk berbagai perangkat mikrokontroler seperti ESP32 dan Arduino. Selain itu, *GitHub* juga menjadi bagian penting dalam pengelolaan kode sumber serta kolaborasi dalam pengembangan perangkat lunak.

Percobaan ini dilakukan untuk memahami proses pembuatan akun *GitHub* dan *Wokwi*, serta bagaimana kedua platform ini dapat digunakan dalam pengembangan dan simulasi proyek *IoT*. Langkah pertama dalam percobaan ini adalah membuat akun *GitHub* pribadi dan menggunakannya untuk *sign up* pada *Wokwi*. Setelah berhasil masuk, pengguna dapat mengeksplorasi berbagai *template* simulasi yang tersedia serta melakukan eksperimen dengan menambahkan komponen yang diperlukan.

Selain itu, percobaan ini juga bertujuan untuk mengintegrasikan *Wokwi* dengan *Visual Studio Code (VS Code)* sebagai lingkungan pengembangan utama. Dengan menginstal ekstensi *PlatformIO* dan *Wokwi Simulator* di *VS Code*, pengguna dapat mengembangkan dan menguji proyek berbasis ESP32 secara lebih fleksibel. Pembuatan proyek baru di *PlatformIO* dengan memilih *board* **DOIT ESP32 Devkit V1** dan menggunakan *framework* **Arduino** menjadi langkah awal dalam memahami bagaimana sistem *IoT* dapat dikembangkan dan diuji secara virtual sebelum diterapkan pada perangkat fisik.

Melalui percobaan ini, diharapkan peserta dapat memahami proses pengelolaan proyek perangkat lunak berbasis *IoT*, mulai dari pembuatan akun, simulasi sistem, hingga pengujian kode dalam lingkungan pengembangan yang lebih kompleks. Dengan adanya simulasi ini, pengembangan proyek *IoT* dapat dilakukan dengan lebih efisien, tanpa harus langsung menggunakan perangkat keras yang sebenarnya.

## **1.2 Tujuan eksperimen**

## Tujuan dari eksperimen ini adalah untuk mencoba simulasi *Internet of Things (IoT)* menggunakan platform *Wokwi* serta mengeksplorasi berbagai variasi penambahan komponen sesuai kebutuhan. Selain itu, eksperimen ini juga bertujuan untuk mengintegrasikan konfigurasi sistem di *Visual Studio Code (VS Code)* dengan menerapkan logika pemrograman yang telah dirancang. Melalui percobaan ini, diharapkan pengguna dapat memahami bagaimana cara menghubungkan simulasi perangkat keras dengan kode program serta menguji fungsionalitasnya sebelum diterapkan pada perangkat fisik yang sebenarnya.

# 

# **BAB II**

**Metodologi**

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

* Akun Github
* Website Wikwok
* Diagram,json
* Extension PlatformIO & Wokwi pada Vscode
* Visual studio code (Vscode)

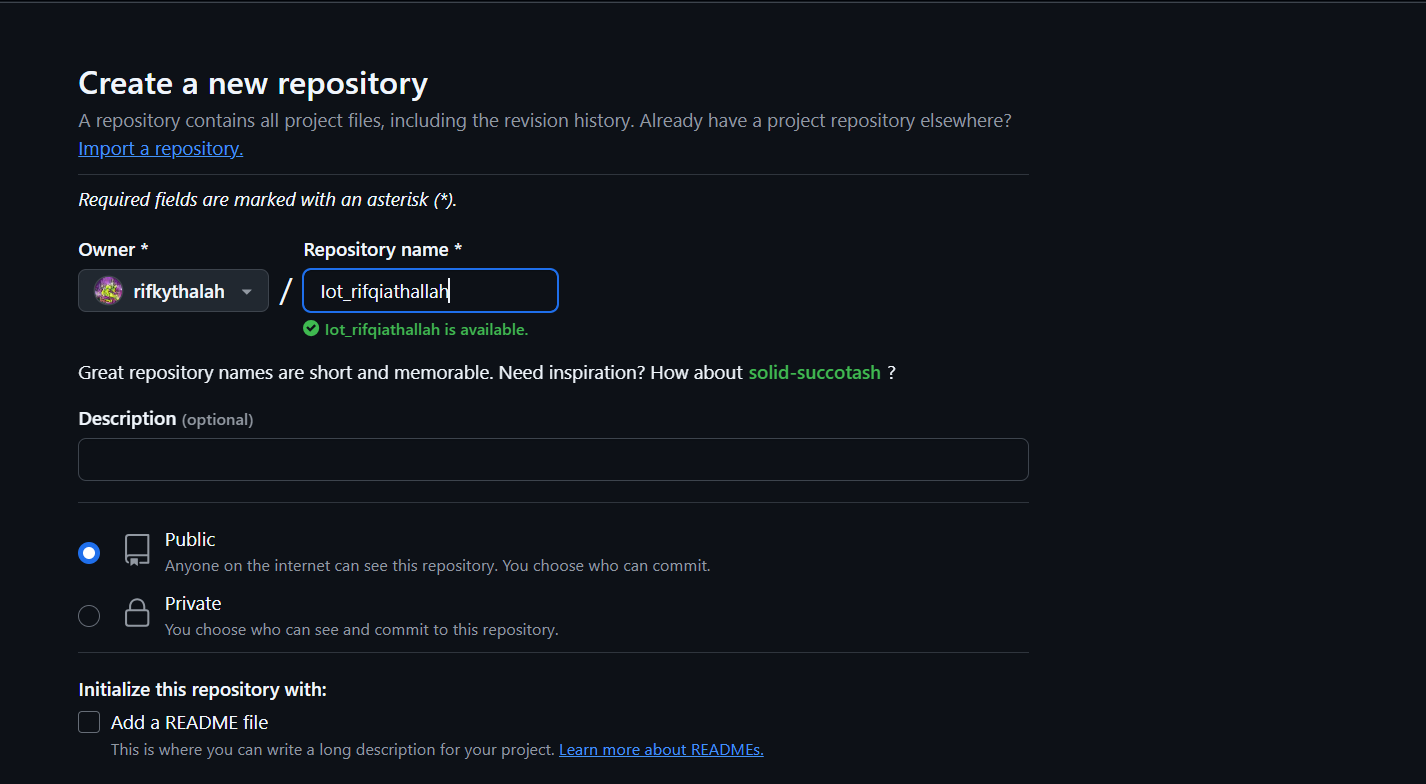
**2.2 Implementation Steps**

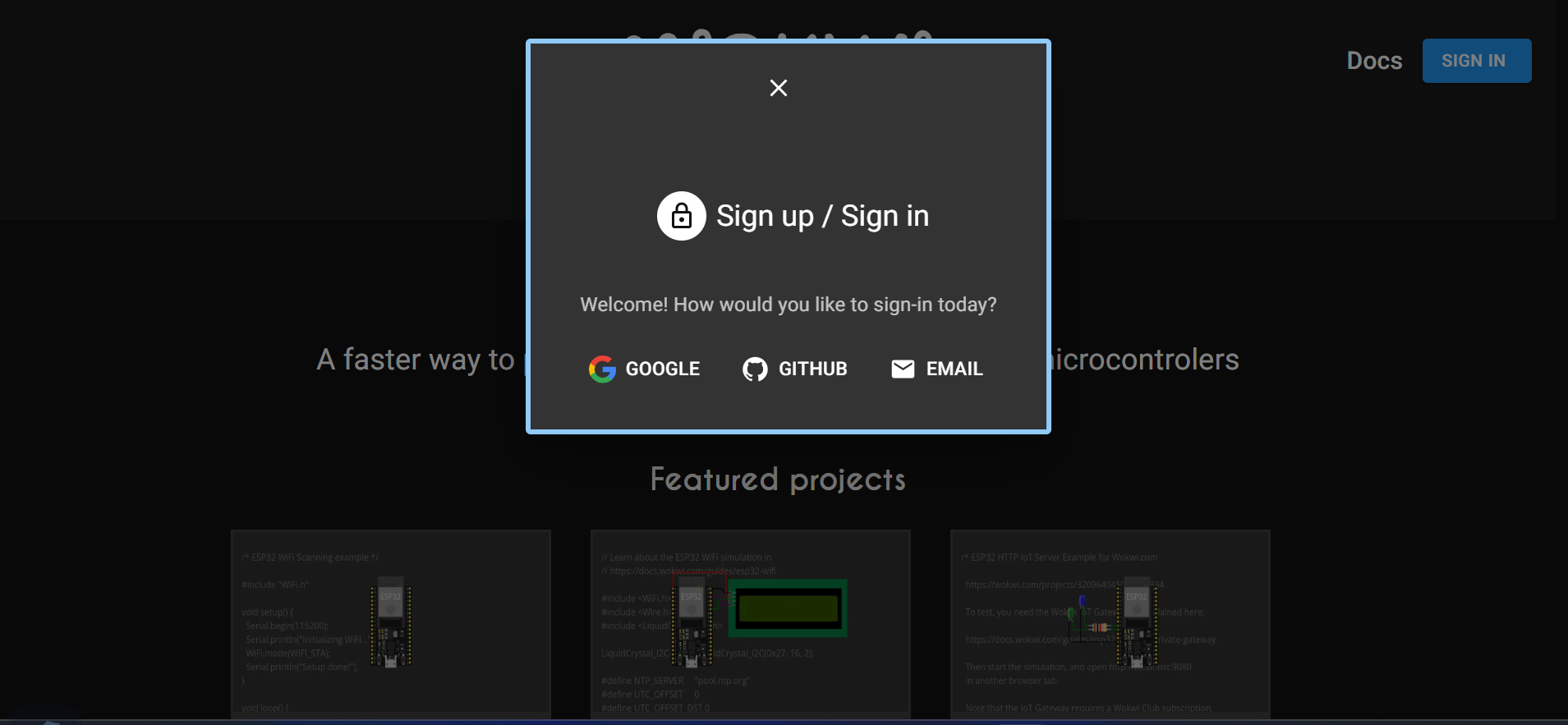
1. Pembuatan Akun dan Simulasi di Wokwi
   * Membuat akun *GitHub* pribadi dan menggunakannya untuk *sign up* di *Wokwi*.
   * Mengeksplorasi berbagai *template* yang tersedia pada *Wokwi* dan memilih proyek simulasi yang sesuai.
   * Menambahkan komponen tambahan (*parts*) ke dalam simulasi sesuai kebutuhan.
2. Konfigurasi dan Pengembangan di Visual Studio Code (VS Code)
   * Menginstal ekstensi *PlatformIO* dan *Wokwi Simulator* di *VS Code*.
   * Mengunduh file diagram.json dari *Wokwi* untuk digunakan dalam pengembangan lebih lanjut.
   * Membuat proyek baru di *PlatformIO* dengan menentukan *board* DOIT ESP32 Devkit V1 dan menggunakan *framework* Arduino.
   * Menulis kode logika yang diperlukan untuk mengontrol perangkat dalam proyek *IoT*.
3. Pengujian dan Validasi
   * Menjalankan simulasi proyek di *Wokwi* untuk mengamati fungsionalitas sistem.
   * Melakukan debugging dan perbaikan kode jika terdapat kesalahan atau ketidaksesuaian dalam sistem.
   * Menguji integrasi kode dengan perangkat keras virtual dan memastikan semua komponen bekerja sebagaimana mestinya.

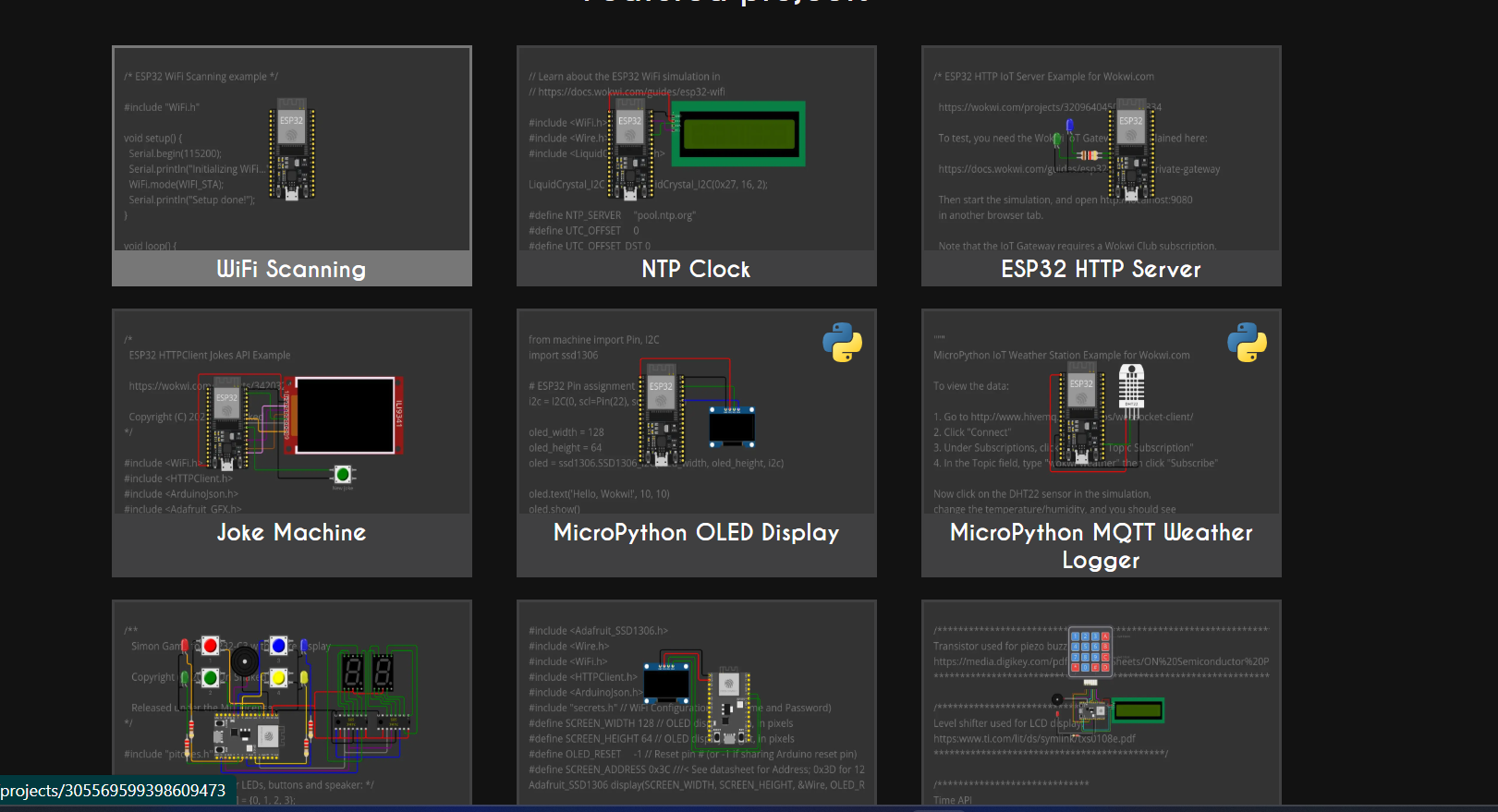
# **BAB III**

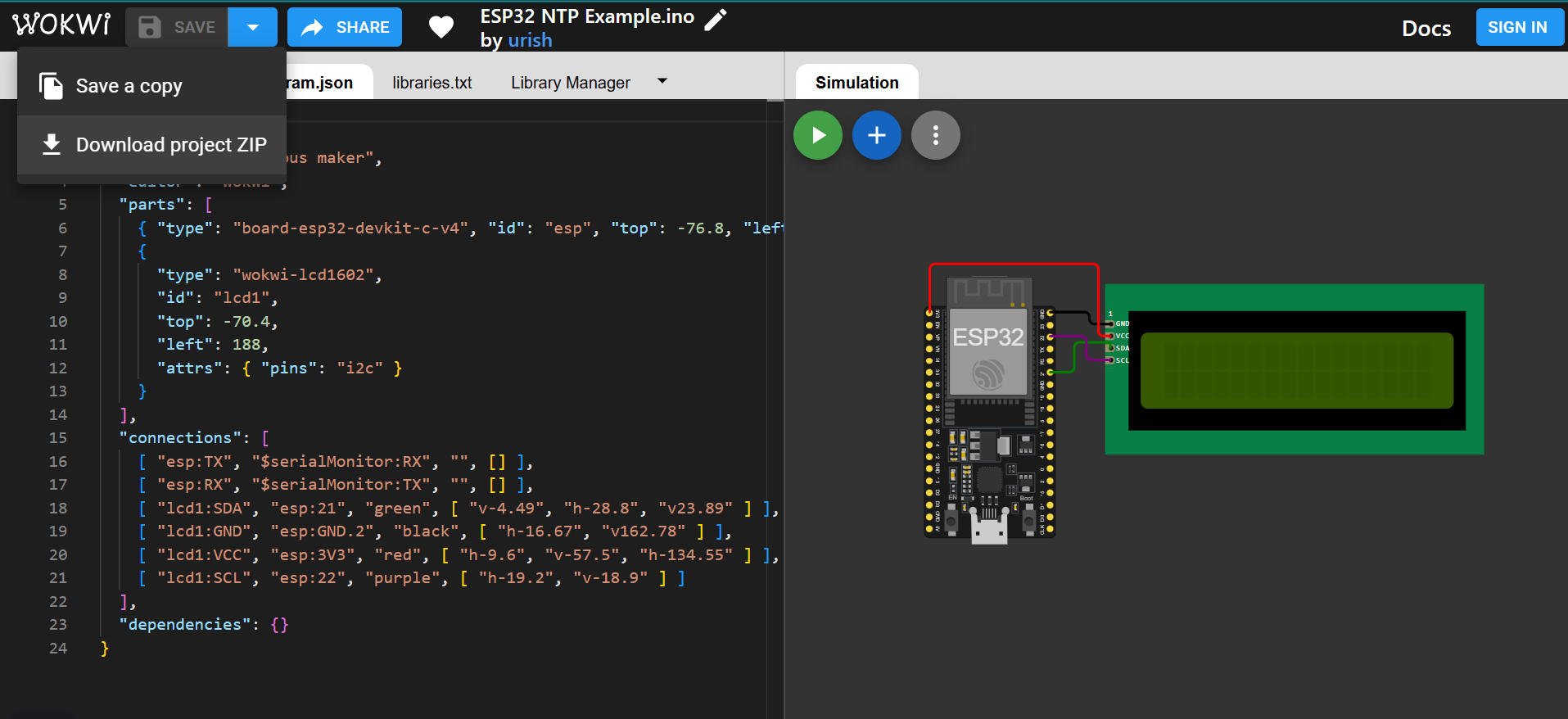
## **Results and Discussion**

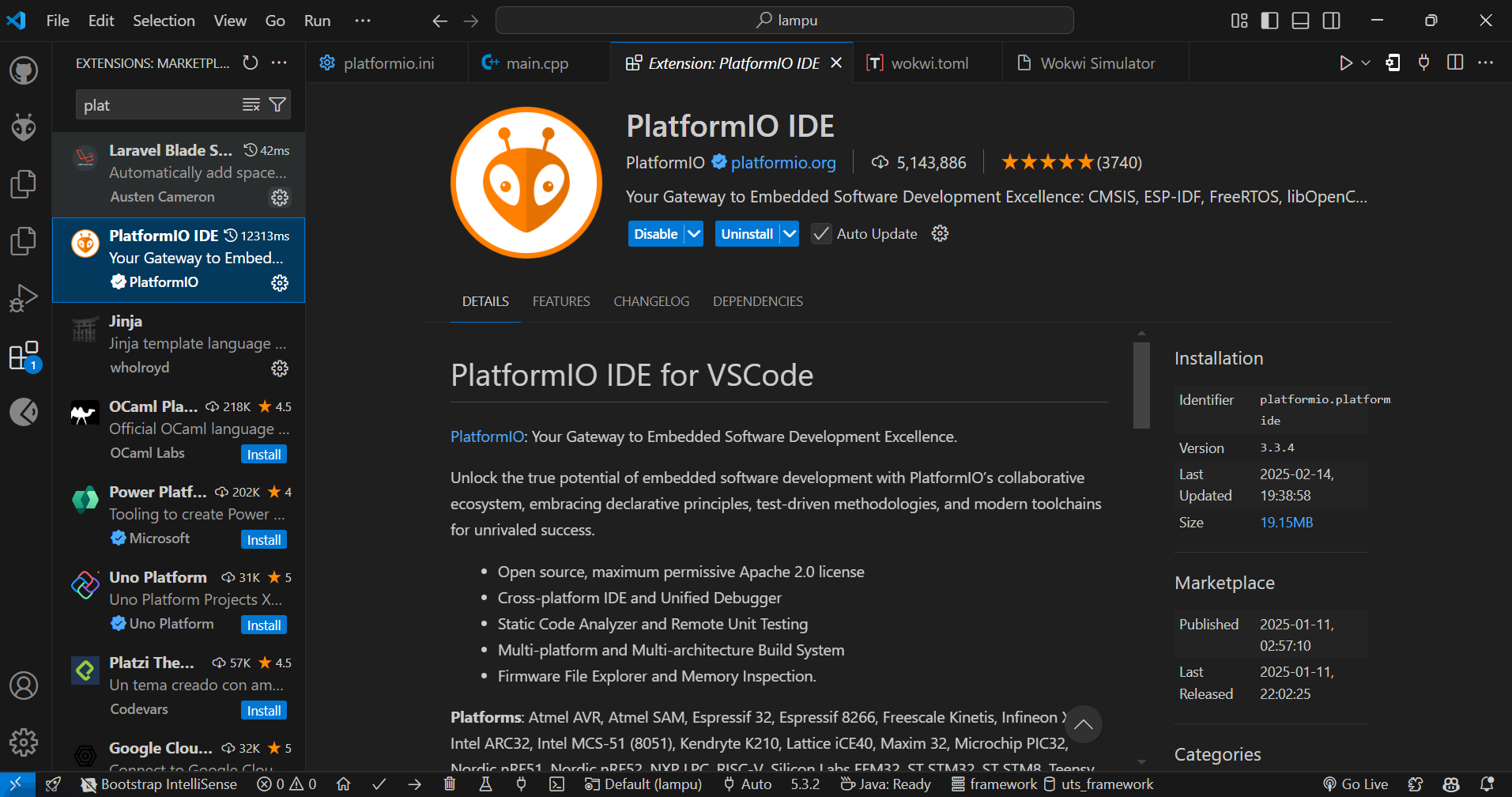
**3.1 Experimental Results**

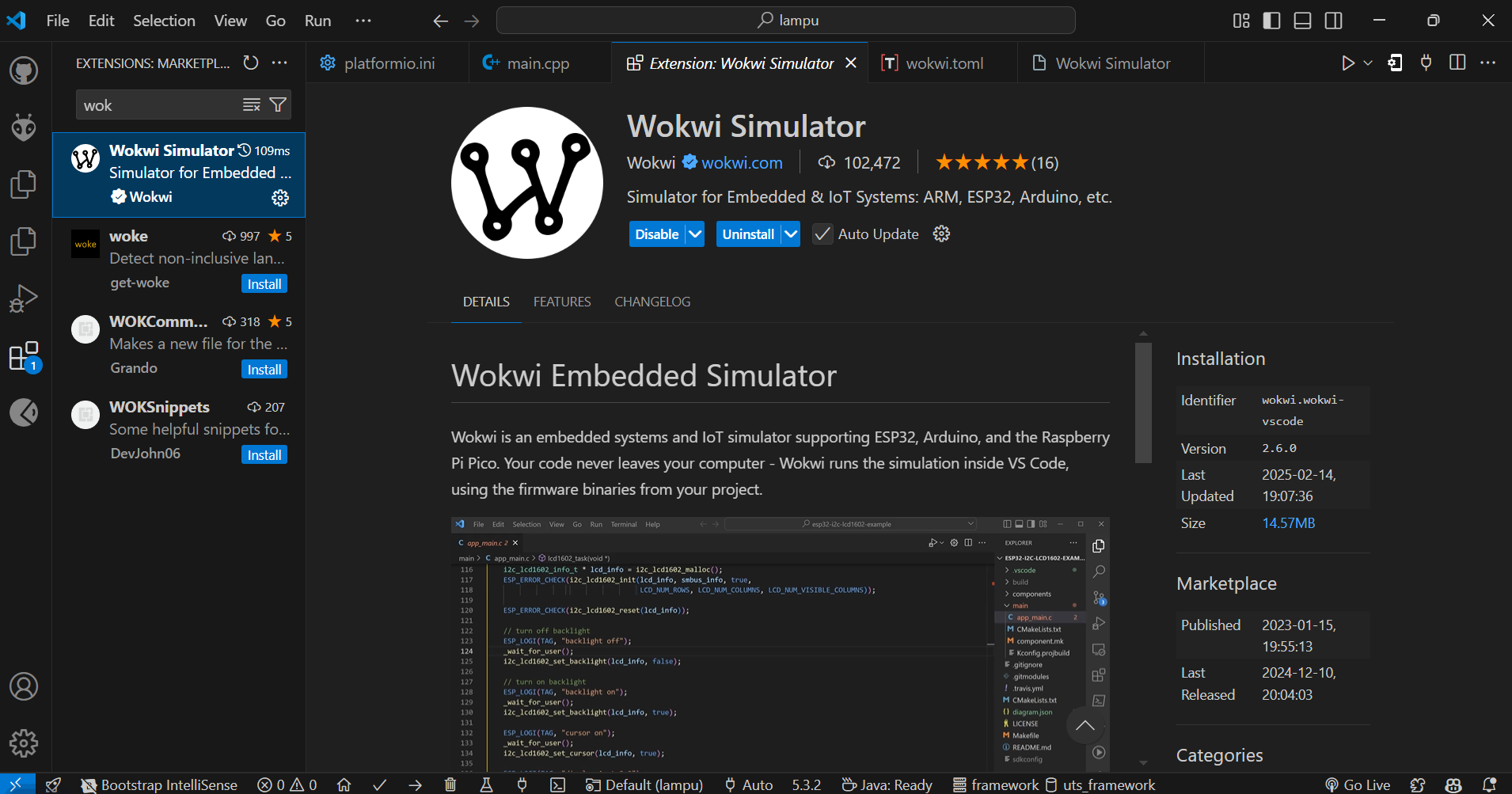
****

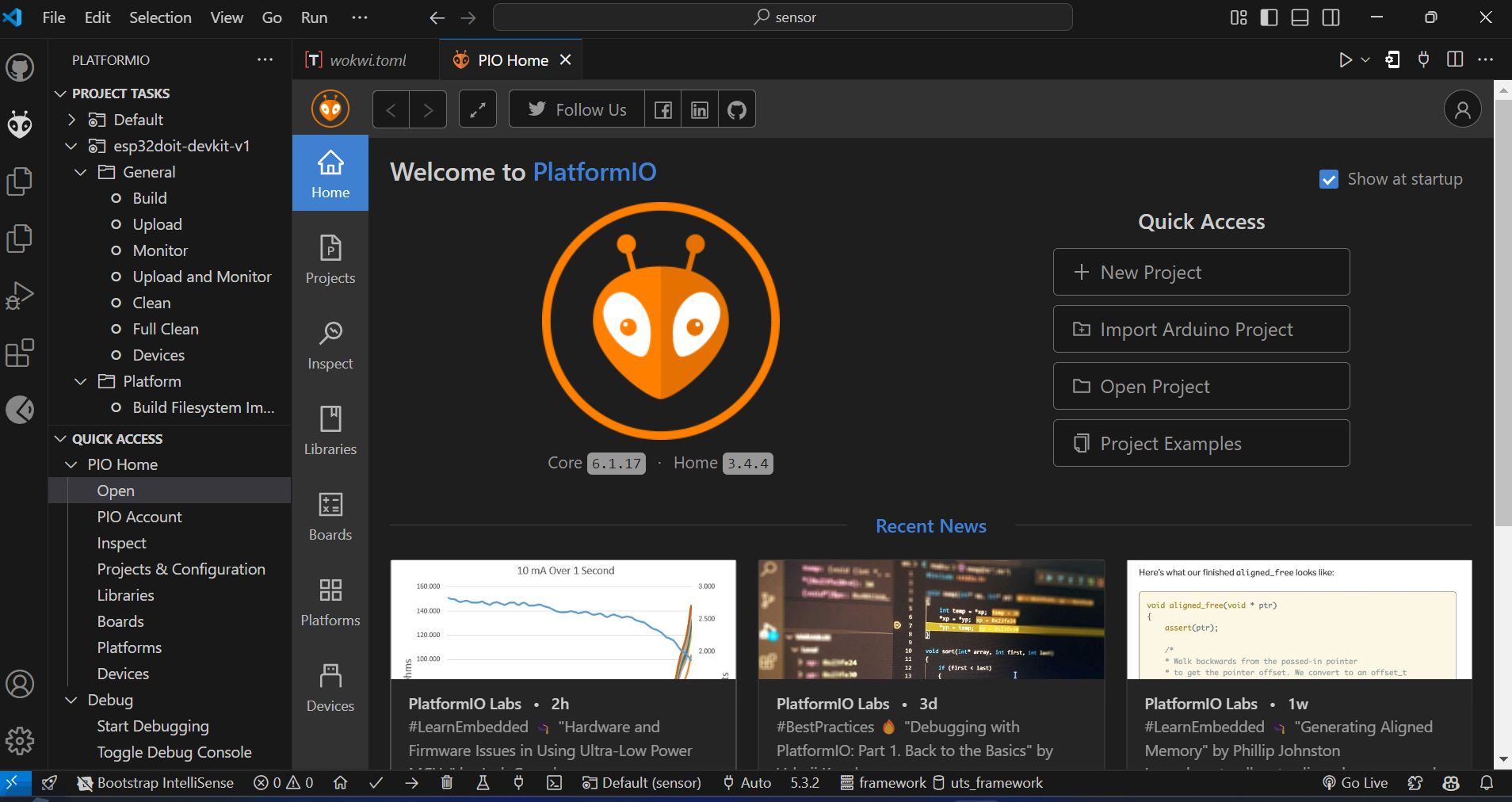
****

****

****





****