**Praktik Pembuatan Akun Wokwi dan Github**

*Pelangi Anggel*

Fakultas Vokasi , Universitas Brawijaya

Email: pelangianggel283@gmail.com

**Abstract**

This practice aims to understand the process of simulating and implementing an environmental menitoring system using the ESP32 microcontroller and a temperature-humadity sensor. The ESP32 function as a microcontroller with bulit-in WI-FI connectivity, enabling real-time data transmission, while the DHT22 sensor is used to measure temperature and humadity. The simulation results demonstrate the effectivesess of using ESP32 and IoT-based paltforms for remote environmental monitoring. This practice highlights the importance of integrating microcontroller based sensor system with cloud computing to enhance automation and real time data acquisition.

*Keywords—ESP32,, GitHub, IoT, Sensor, Arduino*

**1. Introduction**

**1.1 Background**

ESP32 and IoT platforms play a curcial role in the development of smart environmental monitoring system. The ESP32 is a microcontroller that supports WI-Fi communicatio, making it deal for real-time data transmission. Meanwhile, sensors susch as the DHT22 provide accurate temperature and humadity measurements, which are essential for application in agricultre, insdustrial automation, and smart homes. Byintegrating these technologies with cloud-based platforms, users can acces and analyze environmental data remotely. Understanding the proces of configuring and programming the ESP32 for sensor integration is a fundemental step for students and developers in the field of IoT.

**1.2 Objectives**

This experiment aims to introduce participants to the process of configuring and setting up an ESP32-based environmental monitoring system. The main objectivities include integrating the temperature humadity sensor with the ESP32, programming the microcontroller using the Arduino IDE, and transmitting real time data to a cloud-based platform. This practice servers as a foundational step toward developing more complex IoT applications, such as automated climate control system and smart famrming solutions.

**2. Methodology**

**2.1 Tools & Materials**

* **Microcontroller**: Virtual Arduino Uno (via Wokwi)
* **Software**: Wokwi (https://wokwi.com), GitHub (https://github.com), Vs Code, PlatformIO IDE,
* **Internet Access**
  1. **Implementation Steps**

1. Open wokwi.com, and Choose ESP 32 to create a diagram like the one in the module.
2. Open the Vscode. Create new project in PlatformIO IDE.
3. Then compile the c++ code in main.cpp.
4. After the compiling process is successful. there are 2 important files that will be used in the simulation process. namely the firmware.bin and firmware.elf files.
5. Copy the relative path of each file into the wokwi.toml file.
6. Then create a diagram.json file and copy and paste from the json diagram on the wokwi.com platform.
7. Before starting the simulation, request a new license by running the > Wokwi command: Request a New License.
8. The last step is to run the simulation by typing the command

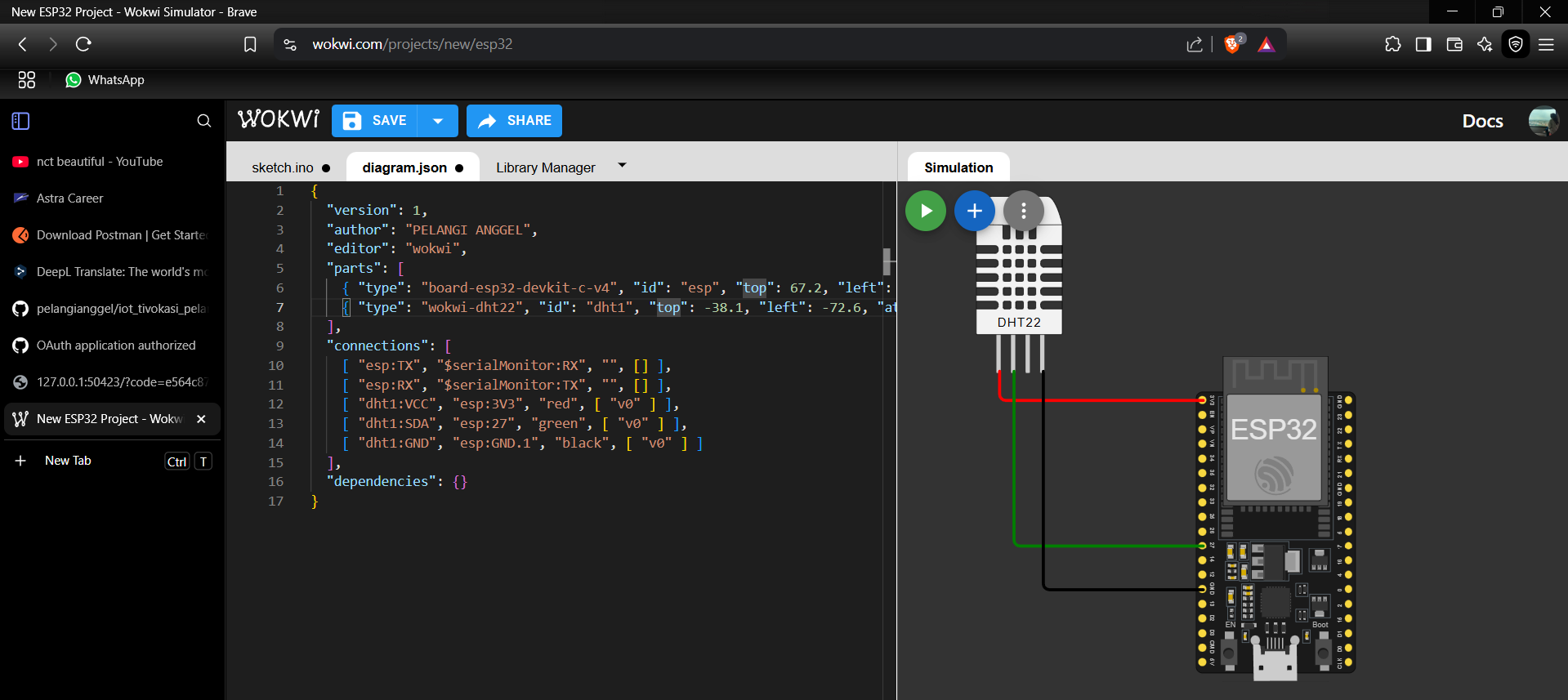
> Wokwi: Start Simulator

**3. Results and Discussion**

**3.1 Experimental Results**

|  |  |  |
| --- | --- | --- |
| **Platform** | **Completed Task** | **Key Outcome** |
| ESP32 | Reading and transmitting temperature and humadity data | Temperature and humadity data were successfully measured and tranmitted in real time to the cloud platform. |
| VS Code | Programming and uploading code to ESP32 | The code was siccsefully written, compiled, and uploades to the ESP32 microcontroller. |

Wokwi screenshot result:



VsCode screenshot result:

