# **LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)**

**Fakultas Vokasi, Universitas Brawijaya**

**Praktek Simulasi ESP32 & Sensor Suhu Kelembaban**

*Alvaro Zeka Ricardo*

*Fakultas Vokasi,Universitas Brawijaya*

*Email : alvarozeka7@student.ub.ac.id*

**Abstract** **:** Simulasi penggunaan ESP32 dengan sensor suhu dan kelembaban DHT22 merupakan salah satu cara untuk mempelajari dan mengimplementasikan Internet of Things (IoT) dalam memantau kondisi lingkungan. ESP32 adalah mikrokontroler yang dilengkapi dengan Wifi dan Bluetooth, sehingga cocok untuk aplikasi IoT. Modul DHT22 adalah sensor yang mampu mengukur suhu dan kelembaban dengan akurasi yang cukup tinggi. Dalam praktek ini, ESP32 akan membaca data dari sensor DHT22 dan menampilkannya melalui antarmuka serial atau platform IoT. Simulasi ini bertujuan untuk memahami cara kerja sensor DHT22, pemrograman ESP32, serta integrasi data ke platform IoT. Hasil simulasi menunjukkan bahwa ESP32 mampu membaca data dari DHT22 dengan baik dan mengirimkannya ke platform IoT secara real-time, sehingga dapat digunakan untuk memantau kondisi lingkungan secara efektif.

**Kata Kunci : Suhu,Esp32,DHT22**

**1. Introduction**

**1.1 Latar belakang**

Perkembangan teknologi Internet of Things (IoT) telah membawa perubahan signifikan dalam berbagai aspek kehidupan, terutama dalam bidang otomasi dan sistem kontrol. IoT memungkinkan perangkat elektronik untuk saling terhubung dan berkomunikasi melalui jaringan internet, sehingga dapat meningkatkan efisiensi dan efektivitas dalam berbagai aplikasi.

Sensor DHT22 adalah komponen yang sering digunakan untuk mengukur suhu dan kelembaban dengan akurasi yang tinggi. Sensor ini mampu mengukur suhu dalam rentang -40°C hingga 80°C dengan akurasi ±0.5°C, serta kelembaban dalam rentang 0% hingga 100% dengan akurasi ±2%. Kombinasi antara ESP32 dan DHT22 memungkinkan pembuatan sistem pemantauan lingkungan yang efektif dan real-time.

Praktek pembuatan simulasi ini bertujuan untuk memahami cara kerja sensor DHT22, pemrograman ESP32, serta integrasi data ke platform IoT seperti Blynk atau ThingSpeak. Dengan memahami konsep ini, diharapkan dapat dikembangkan sistem pemantauan lingkungan yang lebih kompleks dan aplikatif di masa depan. Simulasi ini juga menjadi langkah awal bagi para pengembang untuk mengeksplorasi lebih jauh potensi IoT dalam berbagai bidang.

**1.2 Tujuan eksperimen**

Eksperimen ini bertujuan untuk memahami konsep dasar mikrokontroler ESP32 serta mengimplementasikan sensor suhu dan kelembaban DHT22 dalam suatu sistem pemantauan. Peserta diharapkan dapat mempelajari prinsip kerja sensor DHT22 dalam mengukur suhu dan kelembaban serta memahami cara menghubungkannya dengan ESP32. Selain itu, eksperimen ini juga bertujuan untuk membuat simulasi rangkaian ESP32 dengan sensor DHT22 menggunakan perangkat lunak yang sesuai serta mengembangkan kode pemrograman untuk membaca dan menampilkan data dari sensor.

Melalui eksperimen ini, peserta akan menguji dan menganalisis data suhu serta kelembaban yang diperoleh dari sensor DHT22, kemudian mengevaluasi keakuratan data serta kemungkinan kesalahan dalam pembacaan sensor. Selain itu, peserta juga akan mempelajari cara menampilkan data hasil pembacaan sensor pada Serial Monitor atau layar tampilan seperti LCD/OLED. Lebih lanjut, eksperimen ini bertujuan untuk mengembangkan pemahaman tentang integrasi ESP32 dengan platform IoT untuk mengirim dan menyimpan data secara real-time.

**2. Methodology (Metodologi)**

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

Alat yang digunakan :

ESP32 - Sebagai mikrokontroler utama untuk mengontrol LED.

DHT22 - Sebagai sensor suhu dan kelembaban.

Kabel Jumper - Untuk menghubungkan komponen satu sama lain.

Breadboard - Sebagai media untuk merangkai komponen tanpa perlu menyolder.

Adaptor USB atau Kabel Micro-USB - Untuk memberikan daya ke ESP32.

Laptop - Untuk mensimulasikan Sistem

Software yang digunakan  
  
 Visual Studio Code (VSCode) – Alternatif IDE yang lebih fleksibel tnpa harus menunggu antrian di Website Wokwi

PlatformIO IDE (Extension di VSCode) – Sebagai lingkungan pengembangan yang mendukung ESP32 dengan fitur yang lebih canggih dibandingkan Arduino IDE.

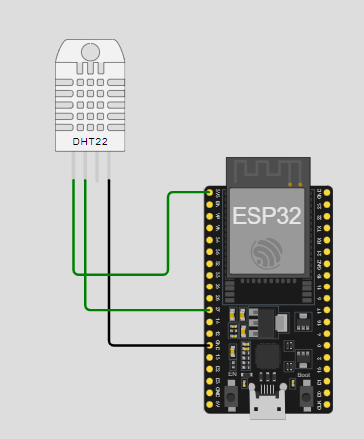
Wokwi Simulator Extension – Untuk melakukan simulasi ESP32 langsung di VSCode sebelum mengimplementasikannya ke perangkat fisik.

Wokwi – Simulator berbasis web untuk menguji rangkaian ESP32 secara virtual sebelum implementasi fisik.

**2.2 Implementation Steps (Langkah Implementasi)**

Pertama : Menyusun Diagram Sistem

  
Tambahkan Part seperti LED di project



Sambungkan bagian pertama ke DHT22 ke ESP.3V3.Bagian kedua DHT22 ke ESP.27 (nomor berapapun).Bagian ke empat DHT22 ke ESP.DND

Selanjutnya Tambahkan Code berikut

#include <Arduino.h>

#include "DHT.h"

#define DHTPIN 27

#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

void setup() {

**Serial**.begin(9600);

**Serial**.println(F("DHTxx test!"));

dht.begin();

}

void loop() {

delay(500);

float h = dht.readHumidity();

// Read temperature as Celsius (the default)

float t = dht.readTemperature();

// Read temperature as Fahrenheit (isFahrenheit = true)

float f = dht.readTemperature(true);

// Check if any reads failed and exit early (to try again).

if (isnan(h) || isnan(t) || isnan(f)) {

**Serial**.println(F("Failed to read from DHT sensor!"));

return;

}

// Compute heat index in Fahrenheit (the default)

float hif = dht.computeHeatIndex(f, h);

// Compute heat index in Celsius (isFahreheit = false)

float hic = dht.computeHeatIndex(t, h, false);

**Serial**.print(F("Humidity: "));

**Serial**.print(h);

**Serial**.print(F("% Temperature: "));

**Serial**.print(t);

**Serial**.print(F("°C "));

**Serial**.print(f);

**Serial**.print(F("°F Heat index: "));

**Serial**.print(hic);

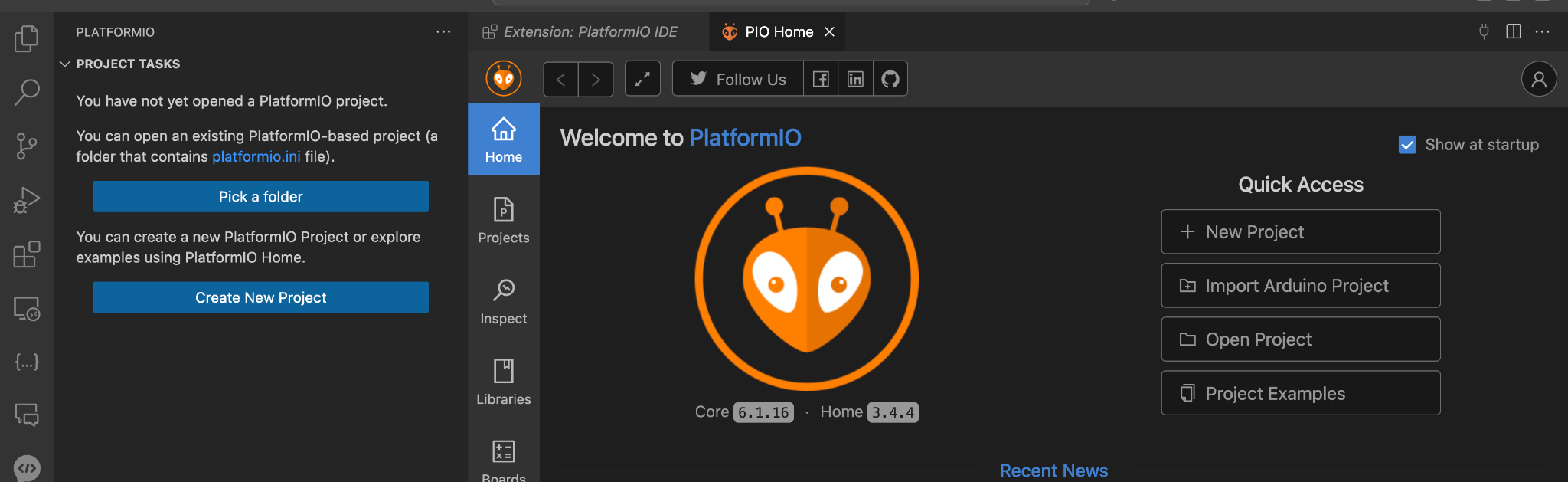
**Serial**.print(F("°C "));

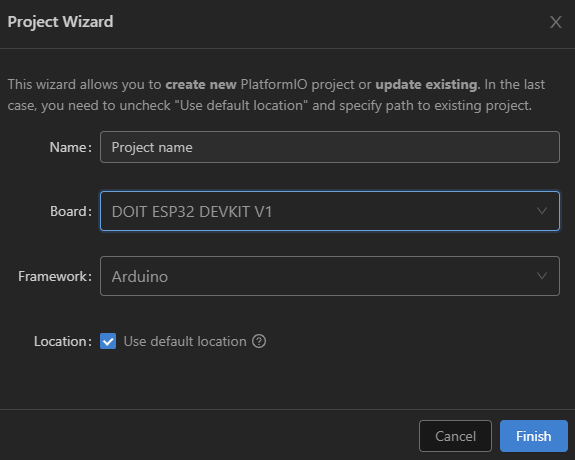
**Serial**.print(hif);

**Serial**.println(F("°F"));

}

Buat project baru di PlatformIO IDE

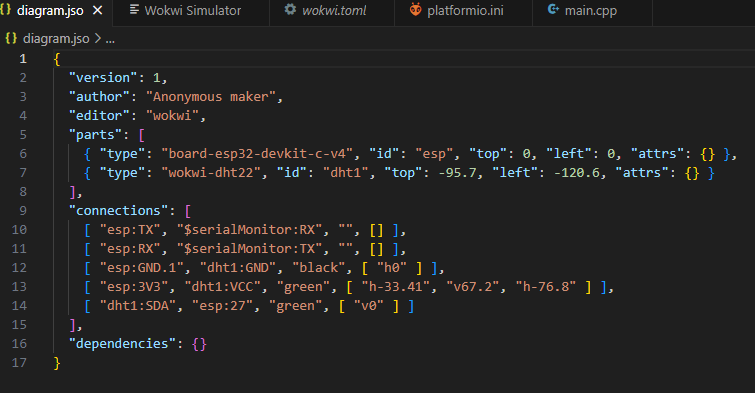




Pilih board DOIT ESP32 DEVKIT V! dan untuk nama bisa bebas.

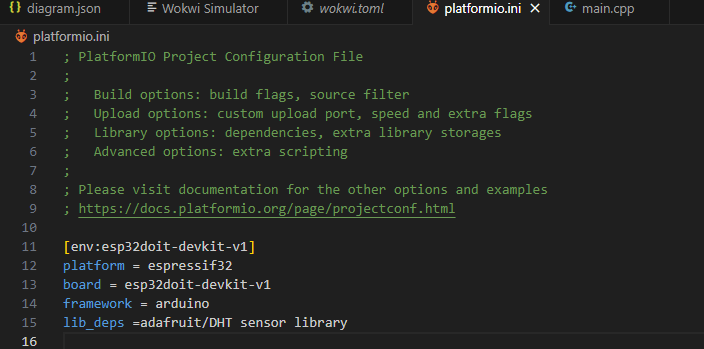
buat file diagram.json untuk menaruh diagram yang telah kita buat di website

Ubah dulu nama file diagram.json untuk menambahkan diagram menjadi diagram.jso



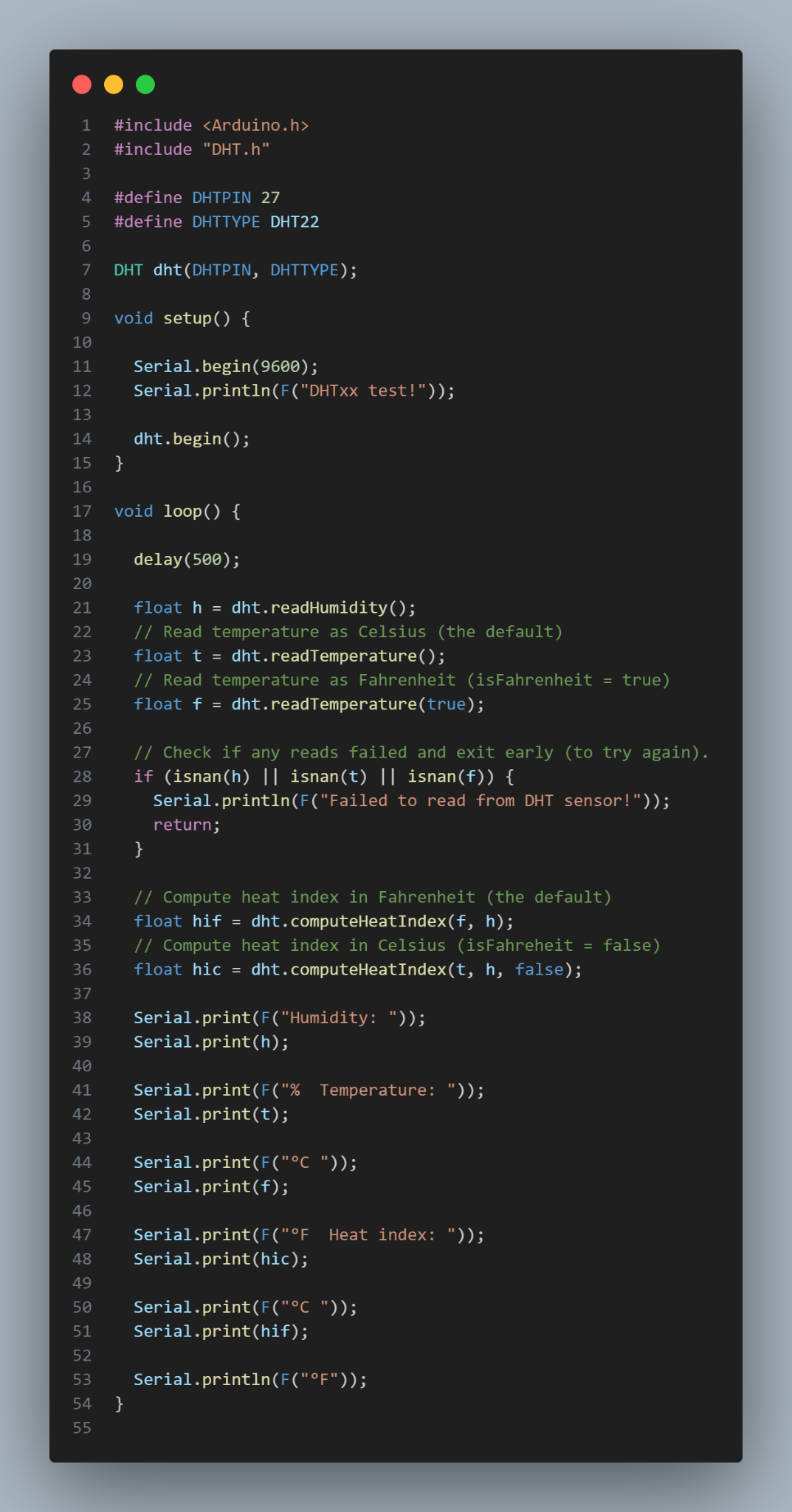
Paste diagram.json yang ada di wokwi web dan ubah nama file menjadi diagram.json lagi.

Edit FIle Platformio.ini menjadi

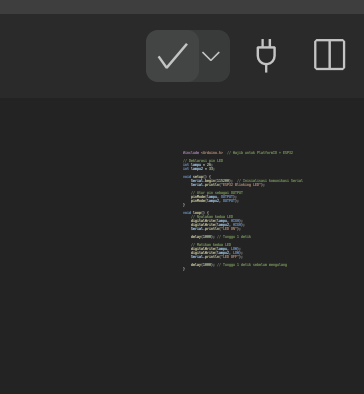


Kemudian pilih file src/main.cpp dan copy paste code yang telah kita buat di wokwi web.

Tambah bagian "#include <Arduino.h>" dan tambahkan (#include "DHT.h") diatas source code.



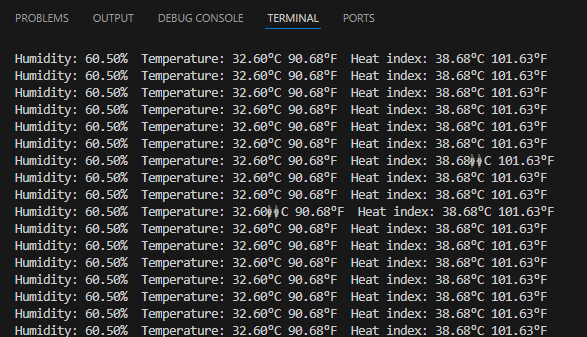
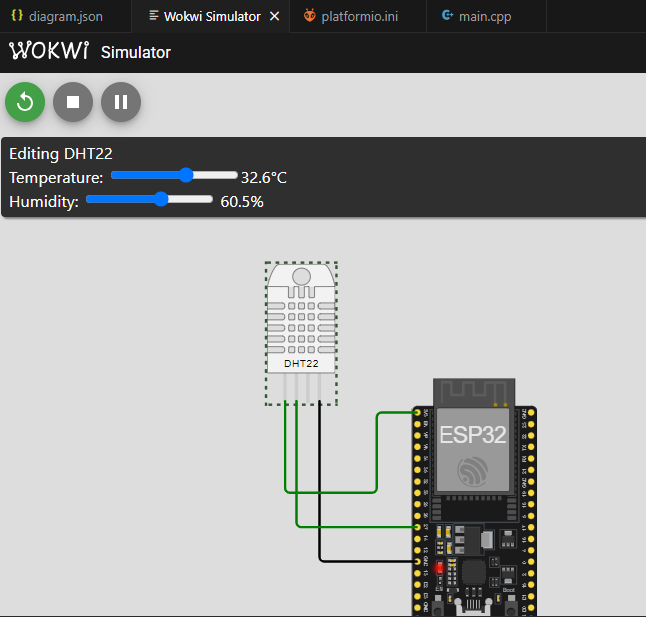
Setelah itu build code tadi dengan menekan tombol centang di kanan atas



Setiap kita mengubah code yang ada di main.cpp harus selalu di build

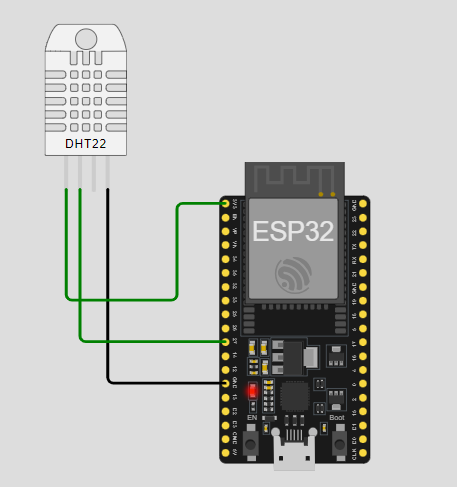
**3. Results and Discussion (Hasil dan Pembahasan)**

**3.1 Experimental Results (Hasil Eksperimen)**



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

1.Diagram



2.Kode Program

