

LAPORAN PRAKTIKUM ESP32 Sensor Suhu & Kelembaban INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

Anisa Amalia

Fakultas Vokasi, Universitas Brawijaya

Email: anisaamalia0412@gmail.com

Praktikum ini bertujuan untuk mengimplementasikan ESP32 dalam membaca data suhu dan kelembaban menggunakan sensor DHT22 serta menampilkannya secara real-time. Eksperimen dilakukan dengan menghubungkan sensor DHT22 ke ESP32, kemudian memprogram mikrokontroler menggunakan Arduino IDE untuk membaca dan menampilkan data pada Serial Monitor. Pengujian dilakukan dalam beberapa kondisi lingkungan untuk mengamati keakuratan sensor dalam mendeteksi perubahan suhu dan kelembaban. Hasil yang diperoleh menunjukkan bahwa sensor DHT22 dapat memberikan data dengan tingkat akurasi yang cukup baik, meskipun terdapat sedikit keterlambatan dalam pembacaan data. Kesimpulannya, ESP32 dikombinasikan dengan sensor DHT22 dapat digunakan sebagai solusi monitoring suhu dan kelembaban yang efisien serta berpotensi untuk dikembangkan dalam berbagai aplikasi berbasis IoT.

1. Introduction (Pendahuluan)

1.1 Latar belakang

Perkembangan teknologi di bidang Internet of Things (IoT) semakin pesat dan banyak diterapkan dalam berbagai aspek kehidupan, termasuk dalam sistem pemantauan lingkungan. Salah satu parameter penting dalam pemantauan lingkungan adalah suhu dan kelembaban udara, yang dapat memengaruhi berbagai sektor seperti pertanian, kesehatan, dan industri. Untuk mendapatkan data suhu dan kelembaban secara real-time dengan tingkat akurasi yang baik, diperlukan perangkat yang mampu membaca, mengolah, dan mengirimkan data secara efisien.

ESP32 merupakan salah satu mikrokontroler yang banyak digunakan dalam proyek IoT karena memiliki konektivitas Wi-Fi dan Bluetooth, serta kemampuan pemrosesan yang lebih tinggi dibandingkan mikrokontroler lainnya. Sementara itu, sensor DHT22 menjadi pilihan yang tepat untuk mengukur suhu dan kelembaban karena memiliki tingkat akurasi yang lebih baik dibandingkan dengan sensor generasi sebelumnya. Kombinasi ESP32 dan sensor DHT22 memungkinkan pembuatan sistem monitoring suhu dan kelembaban yang dapat digunakan dalam berbagai aplikasi, seperti pemantauan kondisi ruangan, pertanian pintar, hingga sistem otomatisasi rumah.

Melalui praktikum ini, dilakukan implementasi ESP32 dan sensor DHT22 untuk membaca suhu dan kelembaban serta menampilkan data yang diperoleh. Dengan memahami cara kerja perangkat keras dan pemrograman ESP32, diharapkan mahasiswa dapat mengembangkan sistem pemantauan yang lebih kompleks dan terintegrasi dengan teknologi IoT di masa depan.

1.2 Tujuan eksperimen

Praktikum ini bertujuan untuk memahami dan mengimplementasikan penggunaan mikrokontroler ESP32 dalam membaca serta menampilkan data suhu dan kelembaban menggunakan sensor DHT22. Adapun tujuan spesifik dari eksperimen ini adalah sebagai berikut:

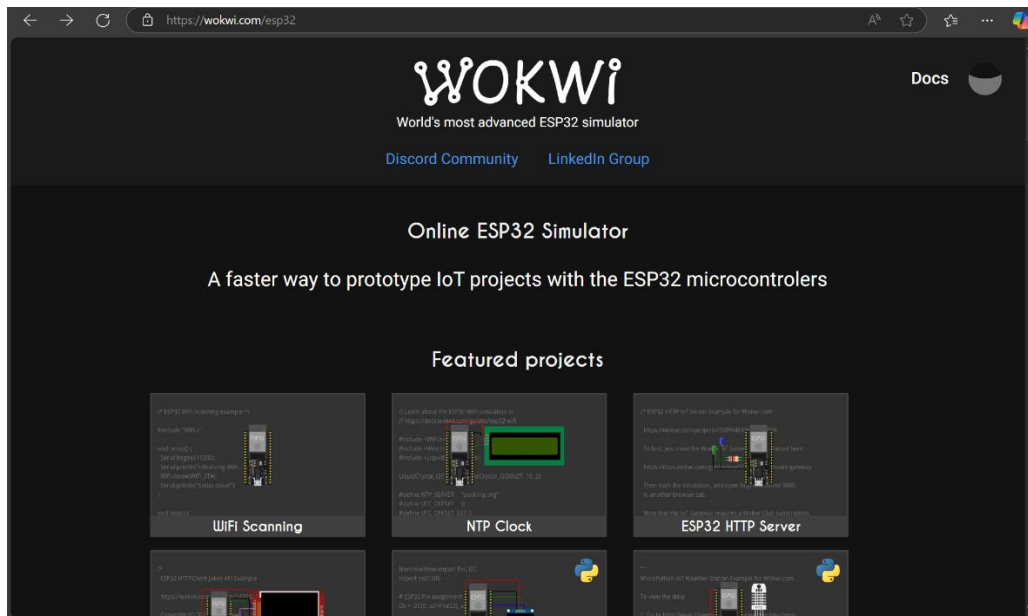
1. Mempelajari cara kerja sensor DHT22 dalam mengukur suhu dan kelembaban udara.
2. Menghubungkan dan mengonfigurasi sensor DHT22 dengan ESP32 menggunakan Arduino IDE.

3. Memprogram ESP32 untuk membaca data suhu dan kelembaban serta menampilkannya secara real-time melalui Serial Monitor.
4. Menganalisis akurasi dan responsivitas sensor DHT22 dalam berbagai kondisi lingkungan.
5. Mengevaluasi potensi penggunaan ESP32 dan sensor DHT22 dalam sistem pemantauan berbasis IoT.

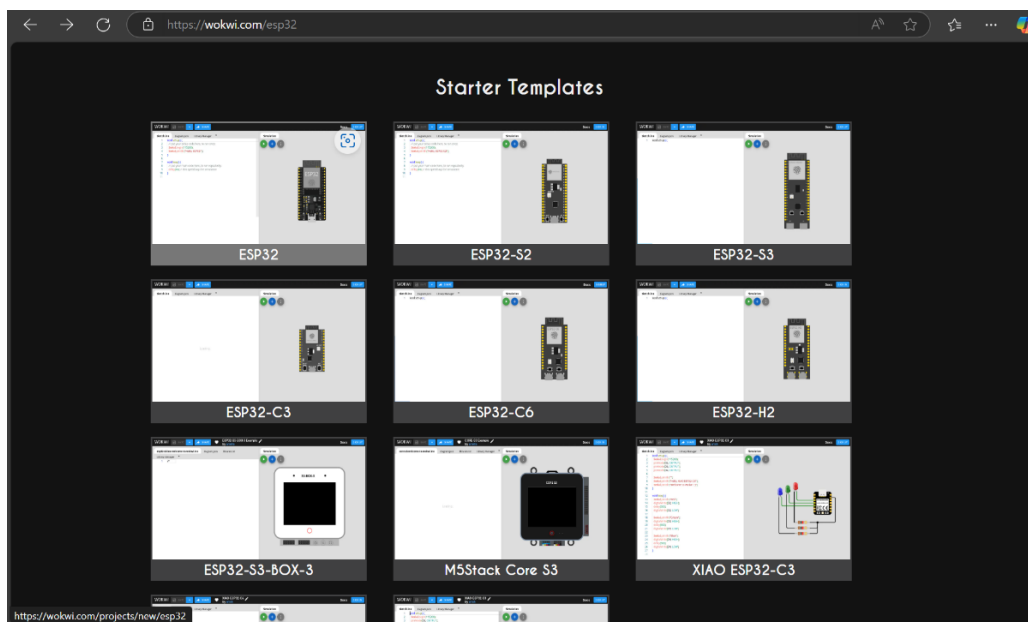
2. Methodology (Metodologi)

2.1 Tools & Materials (Alat dan Bahan)

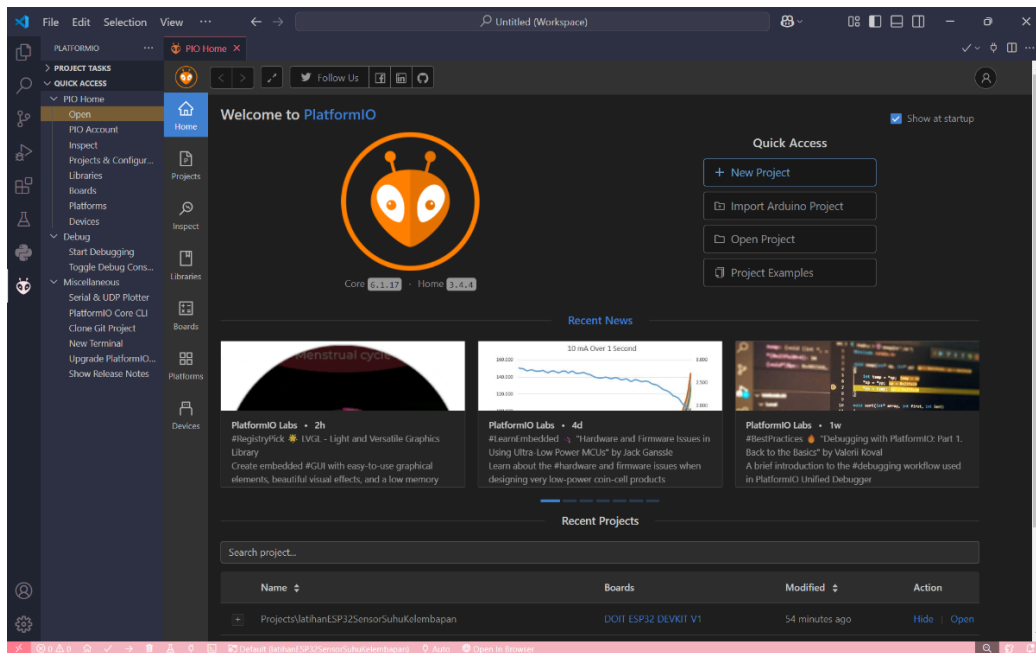
1. Platform Wokwi



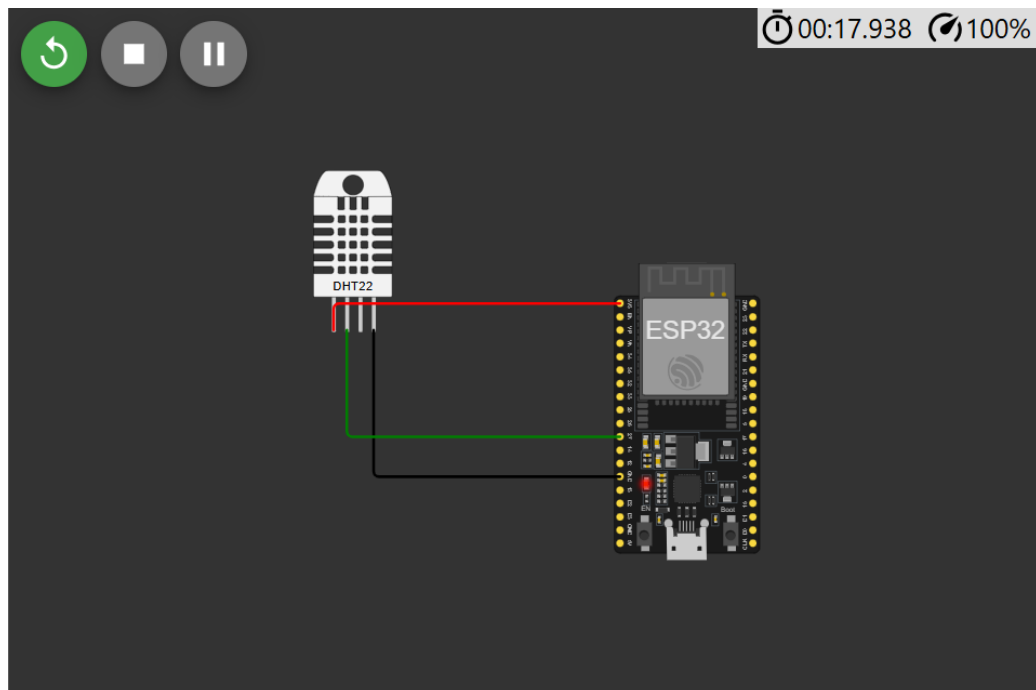
2. Template ESP32



3. Platform.IO

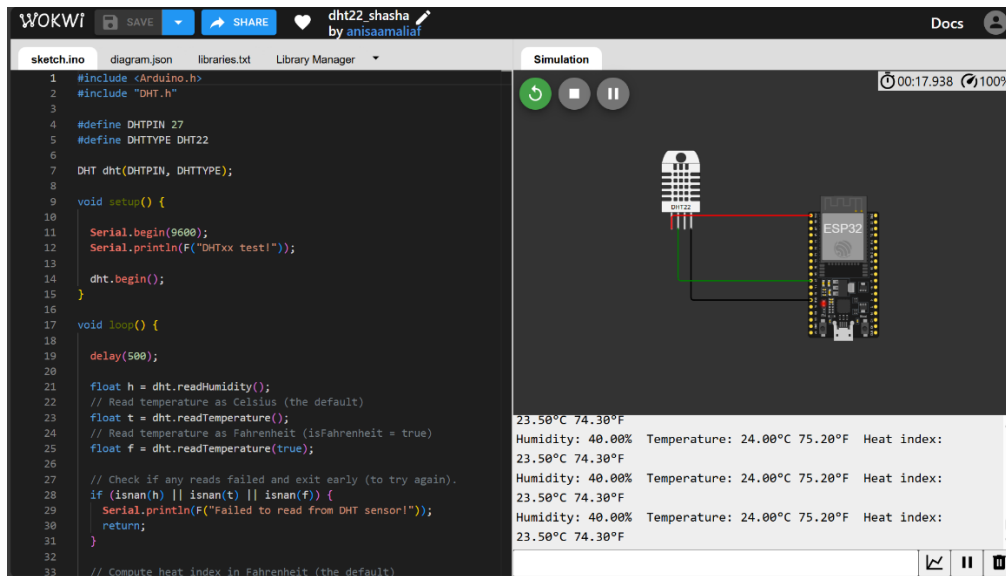


4.ESP32 & DHT22



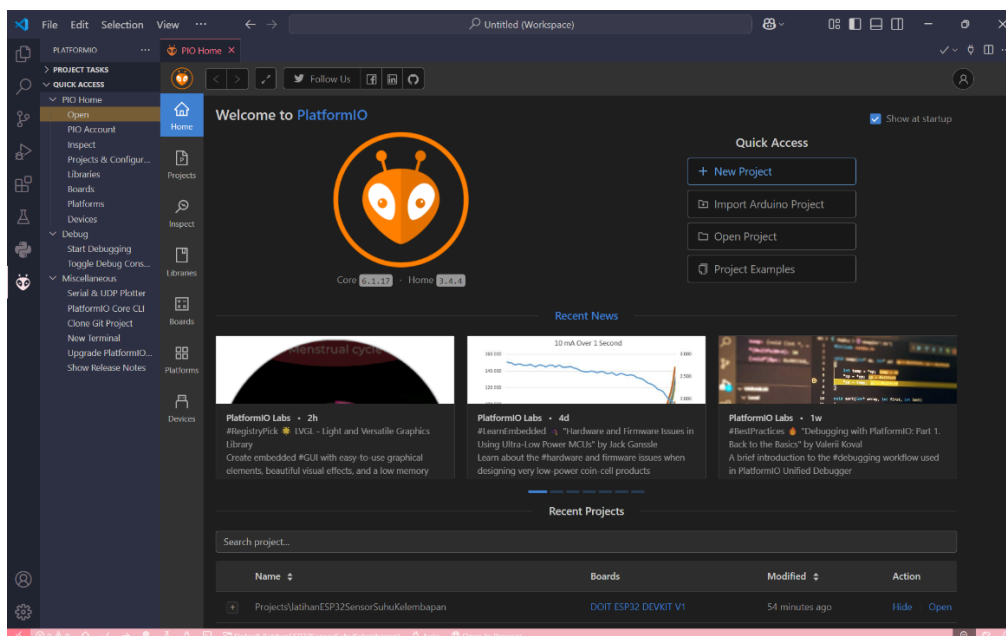
2.2 Implementation Steps (Langkah Implementasi)

1. Buka web wokwi.com dan lakukan langkah-langkah ini:



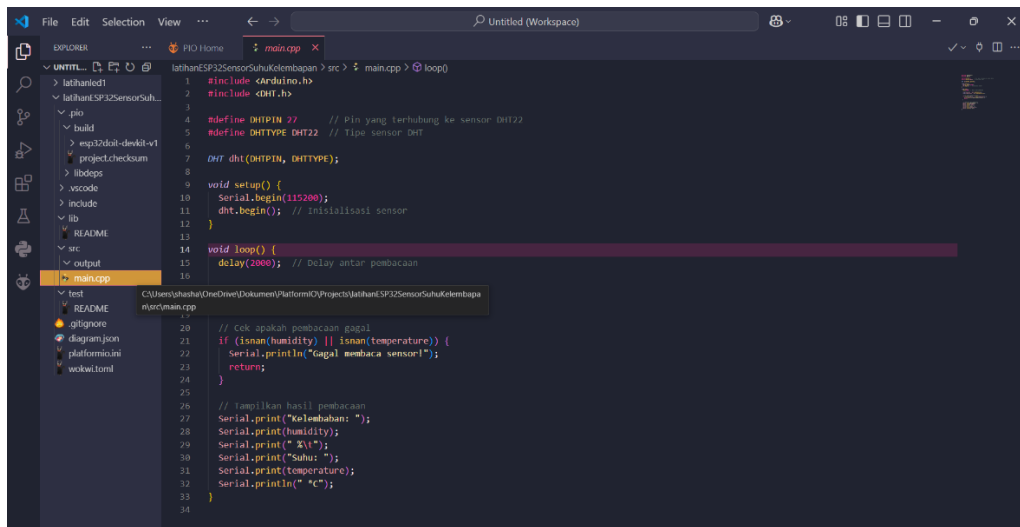
1. Pilih ESP32
2. Tambahkan new part dan pilih DHT22
3. Sambungkan kabelnya sesuai warna ke titik masing-masing, seperti di modul
4. Tulis kode yang sudah diberikan di modul ke sketch.ino
5. Kemudian start simulation

2. Buat project baru pada platform io:



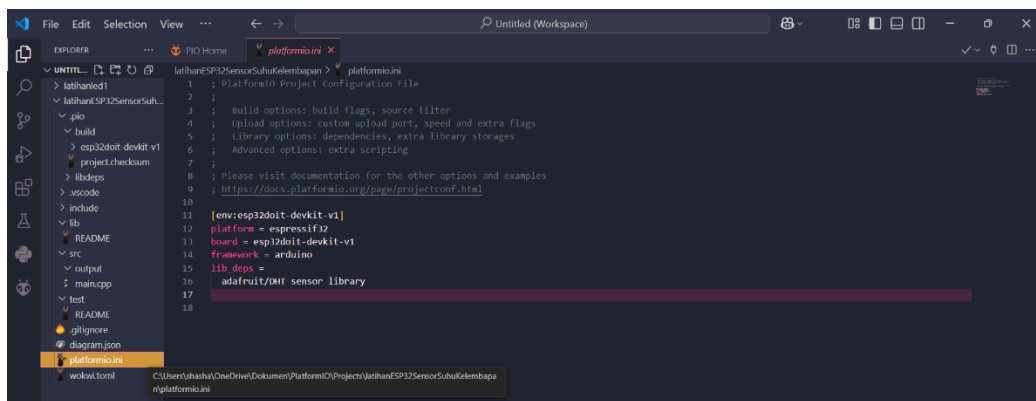
1. Buat new project dan tulis name serta board

3 Tulis koding C++ untuk project ini pada file **src/main.cpp**:



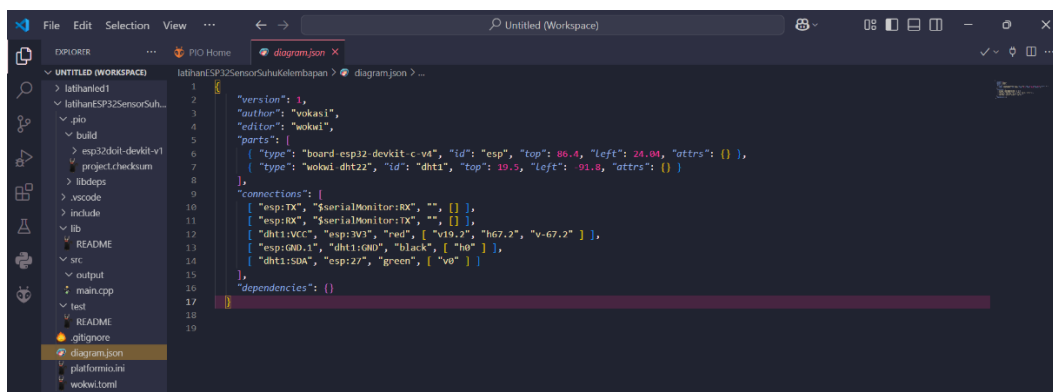
```
1 #include <Arduino.h>
2 #include <DHT.h>
3
4 #define DHTPIN 27 // Pin yang terhubung ke sensor DHT22
5 #define DHTTYPE DHT22 // Tipe sensor DHT
6
7 DHT dht(DHTPIN, DHTTYPE);
8
9 void setup() {
10   Serial.begin(115200);
11   dht.begin(); // Inisialisasi sensor
12 }
13
14 void loop() {
15   delay(2000); // Delay antar pembacaan
16
17   // Cek apakah pembacaan gagal
18   if (!isnan(humidity) || !isnan(temperature)) {
19     Serial.println("Gagal membaca sensor!");
20     return;
21   }
22
23   // Tampilkan hasil pembacaan
24   Serial.print("Kelembaban: ");
25   Serial.print(humidity);
26   Serial.print(" %\n");
27   Serial.print("Suhu: ");
28   Serial.print(temperature);
29   Serial.println(" °C");
30 }
```

4. Edit file **platformio.ini** menjadi seperti ini:



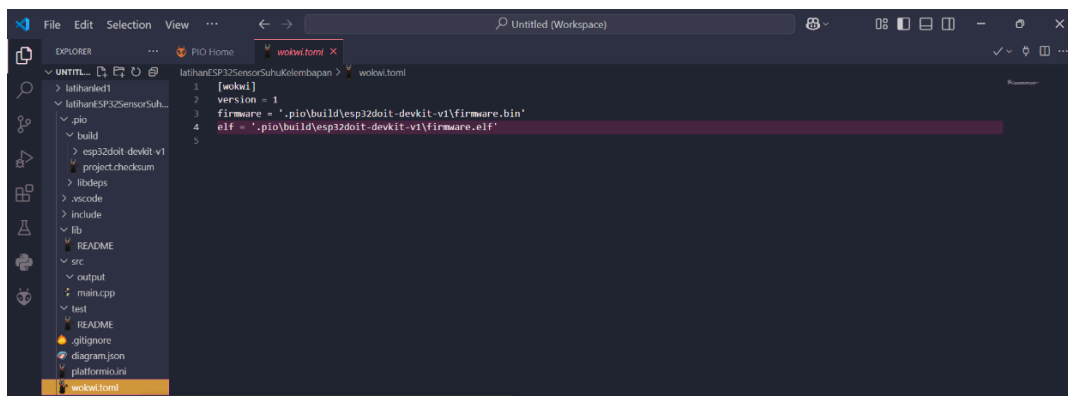
```
1 ; PlatformIO Project Configuration File
2 ;
3 ; Build options: build flags, source filter
4 ; Upload options: custom upload port, speed and extra flags
5 ; Library options: dependencies, extra library storages
6 ; Advanced options: extra scripting
7 ;
8 ; Please visit documentation for the other options and examples
9 ; https://docs.platformio.org/page/projectconf.html
10
11 [env:esp32dev]
12 platform = espressif32
13 board = esp32dev
14 framework = arduino
15 lib_deps =
16   adafuit/DHT sensor library
17
```

5. Buat file baru **diagram.json** , dan copy paste dari diagram.json pada platform online wokwi.com:

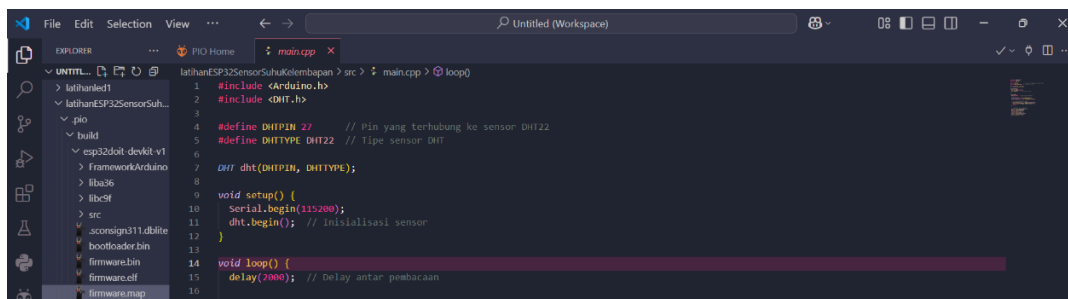


```
1 {
2   "version": 1,
3   "author": "wokwi",
4   "editor": "wokwi",
5   "parts": [
6     { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 86.4, "left": 24.04, "attrs": {} },
7     { "type": "wokwi-dht22", "id": "dht1", "top": 10.5, "left": 91.8, "attrs": {} }
8   ],
9   "connections": [
10    [ "esp:TX", "sserialMonitor:RX", "", [] ],
11    [ "esp:RX", "sserialMonitor:TX", "", [] ],
12    [ "dht1:VCC", "esp:3V3", "red", [ "v19.2", "h67.2", "v-67.2" ] ],
13    [ "esp:GND.1", "dht1:GND", "black", [ "h0" ] ],
14    [ "dht1:SDA", "esp:27", "green", [ "v0" ] ]
15  ],
16   "dependencies": {}
17 }
```

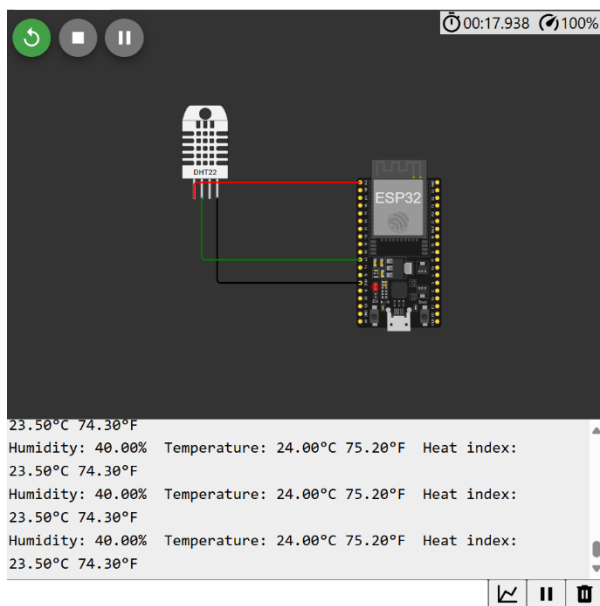
6. Buat file baru **wokwi.toml**, dan isikan file tersebut dengan koding sebagai berikut:



7. Langkah berikutnya lakukan **compile** pada file **main.cpp**, anda akan mendapatkan 2 file baru yaitu **firmware.bin** dan **firmware.elf**:



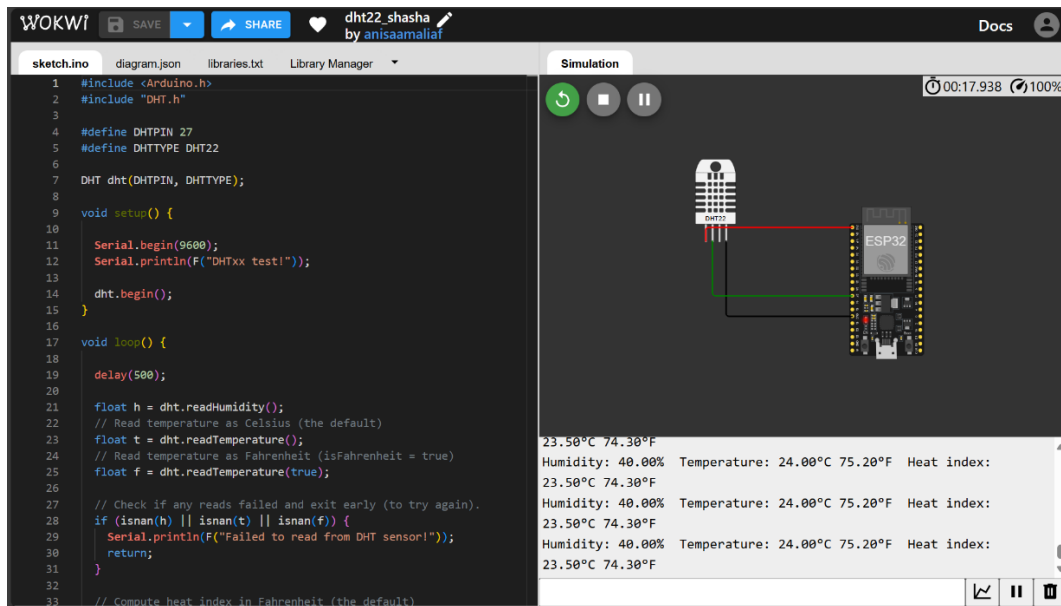
8. Jalankan simulator:



3. Results and Discussion (Hasil dan Pembahasan)

3.1 Experimental Results (Hasil Eksperimen)

1. Wokwi



2. Platform.IO

