**Monitoring Suhu dan Kelembaban Menggunakan ESP32, Sensor DHT22, dan LCD 16x2**

*Ibnu Jaisyurrahman Faiz*

*Program Studi Teknologi Informasi, Fakultas Vokasi, Universitas Brawijaya*

*Email: (*[*ibnufaiz72ub@student.ub.ac.id*](mailto:ibnufaiz72ub@student.ub.ac.id)*)*

**Abstrak:** Pada proyek ini, dirancang sebuah sistem pemantauan suhu dan kelembaban menggunakan ESP32 sebagai mikrokontroler utama, sensor DHT22 sebagai alat pengukur suhu dan kelembaban, serta LCD 16x2 sebagai media tampilan hasil pengukuran. Sistem ini bekerja dengan membaca data dari sensor DHT22 secara periodik dan menampilkannya dalam bentuk angka pada LCD. Implementasi dilakukan dengan menggunakan bahasa pemrograman C++ dalam lingkungan Arduino IDE. Hasil pengujian menunjukkan bahwa sistem dapat menampilkan data suhu dalam satuan Celsius dan kelembaban dalam bentuk persentase secara real-time. Dengan adanya sistem ini, diharapkan pengguna dapat dengan mudah memantau kondisi lingkungan sekitar secara efisien dan praktis.

Kata Kunci : *ESP32, DHT22, LCD 16x2, Pemantauan Suhu, Kelembaban, IoT, Arduino IDE*

**Abstract:** In this project, a temperature and humidity monitoring system was designed using the ESP32 as the main microcontroller, the DHT22 sensor for measuring temperature and humidity, and a 16x2 LCD as the display medium. The system operates by periodically reading data from the DHT22 sensor and displaying the values on the LCD screen. The implementation was carried out using the C++ programming language within the Arduino IDE environment. Testing results indicate that the system successfully displays temperature in Celsius and humidity in percentage format in real-time. This system is expected to help users easily monitor environmental conditions efficiently and practically.

Key Word: *ESP32, DHT22, 16x2 LCD, Temperature Monitoring, Humidity, IoT, Arduino IDE*

1. **Pendahuluan**
2. **Latar belakang**

Pemantauan suhu dan kelembaban merupakan aspek penting dalam berbagai bidang, seperti pertanian, kesehatan, industri, dan sistem kontrol lingkungan. Suhu dan kelembaban yang tidak stabil dapat berdampak negatif pada kondisi lingkungan, kualitas produk, serta kenyamanan manusia. Oleh karena itu, diperlukan sistem pemantauan yang dapat memberikan data secara real-time dan akurat.

Dalam perkembangan teknologi saat ini, mikrokontroler seperti ESP32 telah banyak digunakan dalam sistem pemantauan berbasis Internet of Things (IoT) karena memiliki konektivitas Wi-Fi dan kemampuan pemrosesan yang baik. Sensor DHT22 digunakan karena mampu mengukur suhu dan kelembaban dengan akurasi tinggi. Selain itu, LCD 16x2 digunakan sebagai media tampilan agar data dapat langsung terlihat oleh pengguna tanpa perlu perangkat tambahan.

Dengan mengembangkan sistem pemantauan suhu dan kelembaban menggunakan ESP32, sensor DHT22, dan LCD 16x2, pengguna dapat dengan mudah mengetahui kondisi lingkungan secara langsung. Eksperimen ini bertujuan untuk membangun dan menguji sistem yang dapat menampilkan informasi suhu dan kelembaban secara real-time dengan respon yang cepat dan akurat.

1. **Tujuan eksperimen**

* Merancang dan mengimplementasikan sistem pemantauan suhu dan kelembaban menggunakan ESP32, DHT22, dan LCD 16x2.
* Mengukur dan menampilkan data suhu serta kelembaban secara real-time pada LCD.
* Menguji keakuratan sensor DHT22 dalam membaca suhu dan kelembaban lingkungan.
* Memahami prinsip kerja sensor DHT22 dan integrasinya dengan ESP32.
* Mengevaluasi efektivitas sistem dalam menampilkan data dengan cepat dan stabil.

1. **Metodologi**
2. **Alat dan Bahan**

Pada eksperimen ini, simulasi dilakukan menggunakan Wokwi dan implementasi lebih lanjut dilakukan di Visual Studio Code (VS Code). Berikut adalah daftar alat dan bahan yang digunakan:

* Wokwi Simulator – Untuk simulasi rangkaian secara virtual sebelum implementasi fisik.
* Visual Studio Code (VS Code) – Untuk menulis dan mengunggah kode ke ESP32.
* Arduino IDE / PlatformIO – Plugin tambahan untuk pengembangan kode pada VS Code.
* ESP32 – Mikrokontroler utama yang mengontrol sistem.
* Sensor DHT22 – Sensor untuk mengukur suhu dan kelembaban.
* LCD 16x2 – Tampilan untuk menampilkan hasil pengukuran.

1. **Langkah Implementasi**

Langkah implementasi untuk membuat Monitoring Suhu dan Kelembaban Menggunakan ESP32, Sensor DHT22, dan LCD 16x2 sebagai berikut:

* Buka Wokwi di https://wokwi.com.
* Buat proyek baru dengan memilih ESP32 sebagai mikrokontroler.
* Tambahkan komponen: ESP32, Sensor DHT22 dan LCD 16x2
* Hubungkan kabel sesuai dengan rangkaian berikut:

DHT22

VCC → 3.3V ESP32

GND → GND ESP32

Data → GPIO 21 ESP32

* LCD 16x2

RS → GPIO 19

E → GPIO 23

D4 → GPIO 18

D5 → GPIO 17

D6 → GPIO 16

D7 → GPIO 15

VSS dan RW → GND

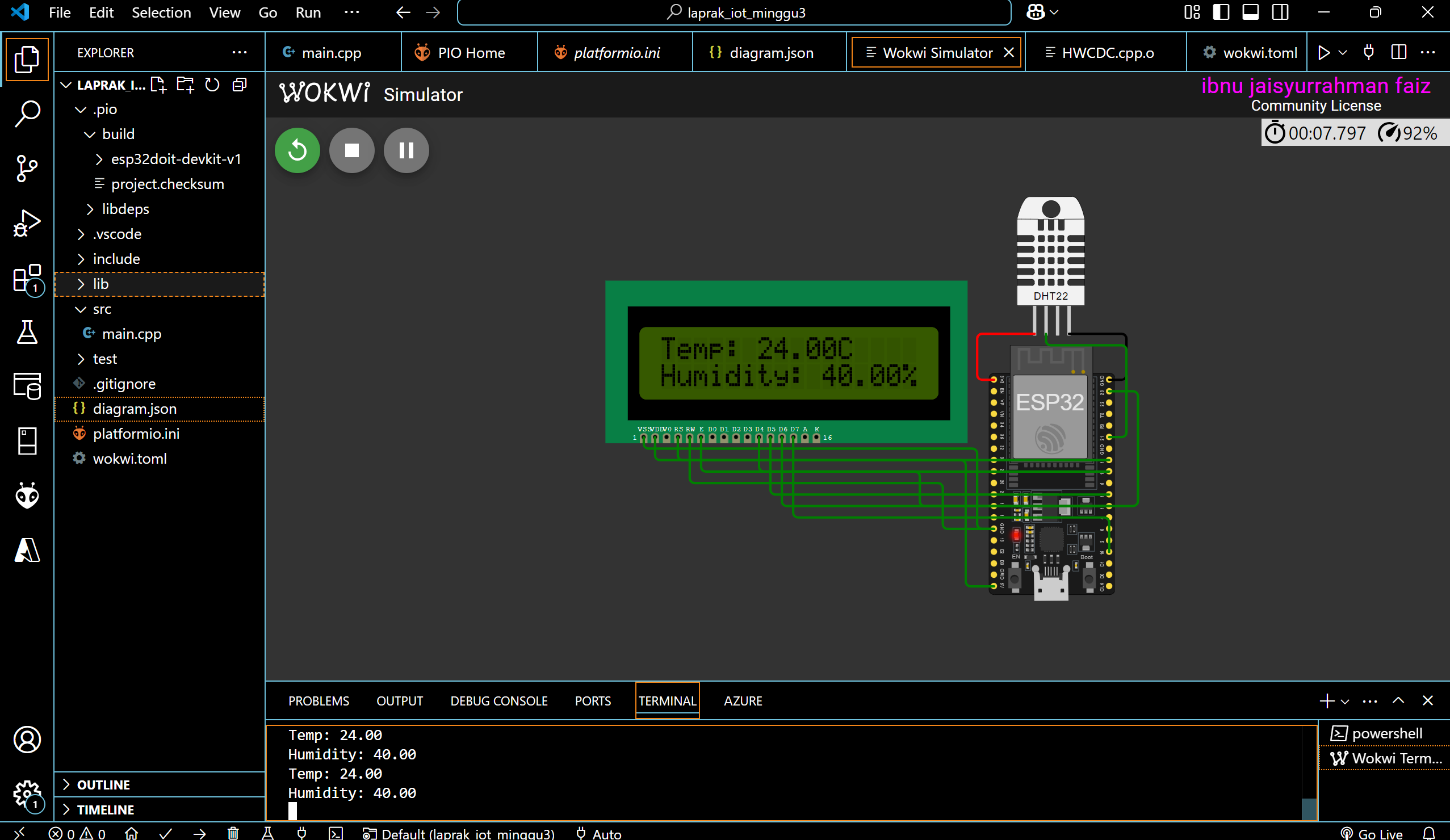
VDD dan A → 5V

K → GND

* Masukkan kode program ke editor Wokwi, lalu jalankan simulasi.
* #include <DHT.h>
* #include <LiquidCrystal.h>
* // Konfigurasi LCD (RS, E, D4, D5, D6, D7)
* LiquidCrystal lcd(19, 23, 18, 17, 16, 15);
* // Konfigurasi DHT22
* #define DHTPIN 21
* #define DHTTYPE DHT22
* #define TIMEDHT 1000
* float humidity, celsius;
* uint32\_t timerDHT = TIMEDHT;
* DHT dht(DHTPIN, DHTTYPE);
* void getTemperature() {
* if ((millis() - timerDHT) > TIMEDHT) {
* timerDHT = millis();
* humidity = dht.readHumidity();
* celsius = dht.readTemperature();
* if (isnan(humidity) || isnan(celsius)) {
* **Serial**.println("Failed to read from DHT sensor!");
* return;
* }
* // Menampilkan data di Serial Monitor
* **Serial**.print("Temp: ");
* **Serial**.println(celsius);
* **Serial**.print("Humidity: ");
* **Serial**.println(humidity);
* // Menampilkan data di LCD
* lcd.clear();
* lcd.setCursor(0, 0);
* lcd.print("Temp: ");
* lcd.print(celsius);
* lcd.print("C");
* lcd.setCursor(0, 1);
* lcd.print("Humidity: ");
* lcd.print(humidity);
* lcd.print("%");
* }
* }
* void setup() {
* **Serial**.begin(115200);
* **Serial**.println("Hello, ESP32!");
* // Inisialisasi sensor DHT
* dht.begin();
* // Inisialisasi LCD
* lcd.begin(16, 2);
* lcd.print("Starting...");
* }
* void loop() {
* delay(500);
* getTemperature();
* }

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

Setelah kode diunggah ke ESP32 menggunakan VS Code (PlatformIO/Arduino IDE), ESP32 mampu mengolah dan mengirim data ke LCD serta Serial Monitor tanpa kendala.



1. **Lampiran**

* Kode program main.cpp
* #include <DHT.h>
* #include <LiquidCrystal.h>
* // Konfigurasi LCD (RS, E, D4, D5, D6, D7)
* LiquidCrystal lcd(19, 23, 18, 17, 16, 15);
* // Konfigurasi DHT22
* #define DHTPIN 21
* #define DHTTYPE DHT22
* #define TIMEDHT 1000
* float humidity, celsius;
* uint32\_t timerDHT = TIMEDHT;
* DHT dht(DHTPIN, DHTTYPE);
* void getTemperature() {
* if ((millis() - timerDHT) > TIMEDHT) {
* timerDHT = millis();
* humidity = dht.readHumidity();
* celsius = dht.readTemperature();
* if (isnan(humidity) || isnan(celsius)) {
* Serial.println("Failed to read from DHT sensor!");
* return;
* }
* // Menampilkan data di Serial Monitor
* Serial.print("Temp: ");
* Serial.println(celsius);
* Serial.print("Humidity: ");
* Serial.println(humidity);
* // Menampilkan data di LCD
* lcd.clear();
* lcd.setCursor(0, 0);
* lcd.print("Temp: ");
* lcd.print(celsius);
* lcd.print("C");
* lcd.setCursor(0, 1);
* lcd.print("Humidity: ");
* lcd.print(humidity);
* lcd.print("%");
* }
* }
* void setup() {
* Serial.begin(115200);
* Serial.println("Hello, ESP32!");
* // Inisialisasi sensor DHT
* dht.begin();
* // Inisialisasi LCD
* lcd.begin(16, 2);
* lcd.print("Starting...");
* lcd.print("Hello :)");
* }
* void loop() {
* delay(500);
* getTemperature();
* }
* Kode program diagram.json
* {
* "version": 1,
* "author": "Anonymous maker",
* "editor": "wokwi",
* "parts": [
* { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {} },
* { "type": "wokwi-dht22", "id": "dht1", "top": -124.5, "left": 23.4, "attrs": {} },
* { "type": "wokwi-lcd1602", "id": "lcd1", "top": -54.17, "left": -320, "attrs": {} }
* ],
* "connections": [
* [ "esp:TX", "$serialMonitor:RX", "", [] ],
* [ "esp:RX", "$serialMonitor:TX", "", [] ],
* [ "dht1:GND", "esp:GND.2", "black", [ "v0", "h48", "v38.4" ] ],
* [ "dht1:VCC", "esp:3V3", "red", [ "v0", "h-48", "v38.4" ] ],
* [ "dht1:SDA", "esp:21", "green", [ "v9.6", "h67.3", "v76.8" ] ],
* [ "lcd1:VSS", "esp:GND.1", "green", [ "v9.6", "h278.4", "v67.2" ] ],
* [ "lcd1:VDD", "esp:5V", "green", [ "v19.2", "h259.3", "v96" ] ],
* [ "lcd1:RW", "esp:GND.1", "green", [ "v38.4", "h211.1", "v28.8" ] ],
* [ "lcd1:RS", "esp:19", "green", [ "v0" ] ],
* [ "lcd1:D4", "esp:18", "green", [ "v28.8" ] ],
* [ "lcd1:D5", "esp:17", "green", [ "v0" ] ],
* [ "lcd1:D6", "esp:16", "green", [ "v0" ] ],
* [ "esp:15", "lcd1:D7", "green", [ "v-28.8", "h-263.96" ] ],
* [ "lcd1:E", "esp:23", "green", [ "v28.8", "h182.4", "v28.8", "h182.4", "v-19.2" ] ]
* ],
* "dependencies": {}
* }