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

**Fakultas Vokasi, Universitas Brawijaya**

**PRAKTIK SIMULASI SENSOR SUHU KELEMBABAN**

*Naila Sahda Aulya*

*Fakultas Vokasi, Universitas Brawijaya*

*Email: naila.sahda18@gmail.com*

**ABSTRAK**

Praktik ini bertujuan untuk merancang dan mengimplementasikan sensor suhu dan kelembaban menggunakan ESP32 dalam lingkungan simulasi Wokwi pada Visual Studio Code (VS Code). Sensor yang digunakan adalah DHT22, yang dikendalikan melalui komunikasi digital dengan ESP32. Program ditulis dalam bahasa Arduino (C++) dengan pustaka DHT untuk membaca data suhu dan kelembaban, serta menampilkan hasilnya di Serial Monitor menggunakan Serial.println(). Hasil simulasi menunjukkan bahwa ESP32 dapat membaca dan menampilkan data suhu serta kelembaban secara real-time sesuai dengan kondisi yang diberikan dalam simulasi. Implementasi ini dapat dikembangkan lebih lanjut untuk berbagai aplikasi seperti sistem pemantauan lingkungan, rumah pintar, atau pertanian berbasis IoT.

**Kata kunci:** ESP32, Wokwi, sensor suhu, sensor kelembaban, VS Code, IoT.

**1. Introduction (Pendahuluan)**

**1.1 Latar belakang praktikum IoT yang dilakukan**

Perkembangan teknologi Internet of Things (IoT) semakin memudahkan pemantauan kondisi lingkungan secara real-time, salah satunya melalui penggunaan sensor suhu dan kelembaban. Sensor ini banyak diterapkan dalam berbagai bidang, seperti sistem pemantauan cuaca, pertanian cerdas, serta kontrol suhu dan kelembaban dalam ruangan. Dengan adanya simulasi berbasis perangkat lunak, seperti Wokwi pada Visual Studio Code (VS Code), proses pembelajaran dan pengujian sistem berbasis ESP32 dapat dilakukan tanpa memerlukan perangkat keras secara langsung. Oleh karena itu, eksperimen ini dilakukan untuk memahami cara kerja sensor suhu dan kelembaban menggunakan ESP32 serta menguji kemampuannya dalam membaca dan menampilkan data lingkungan melalui simulasi.

**1.2 Tujuan eksperimen**

Eksperimen ini bertujuan untuk:

* Mempelajari penggunaan sensor suhu dan kelembaban DHT22 dengan ESP32 dalam lingkungan simulasi Wokwi.
* Mengimplementasikan pemrograman sensor suhu dan kelembaban menggunakan bahasa Arduino (C++) di VS Code.
* Menampilkan data suhu dan kelembaban secara real-time melalui Serial Monitor.
* Menguji efektivitas simulasi dalam memahami prinsip kerja sensor tanpa menggunakan perangkat fisik.
* Mengembangkan dasar untuk implementasi lebih lanjut dalam sistem pemantauan berbasis IoT. 2. Methodology (Metodologi)

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

Dalam eksperimen ini, alat dan bahan yang digunakan meliputi:

Tools (Peralatan):

1. Laptop / Komputer digunakan untuk mengetik, mengedit, dan menjalankan program simulasi.
2. Vs Code merupakan code editor yang digunakan untuk mengetik dan mengembangkan program berbasis Arduino (C++).
3. ESP32 DevKit V4 – Mikrokontroler yang digunakan untuk mengendalikan LED sebagai simulasi lampu lalu lintas.
4. Wokwi Simulator – Platform simulasi berbasis web yang digunakan untuk menguji program ESP32 tanpa memerlukan perangkat keras fisik.
5. Arduino IDE – Software yang digunakan untuk menulis, mengunggah, dan men-debug kode pada ESP32.
6. DHT (Digital Humidity and Temperature) adalah sensor yang digunakan untuk mengukur suhu dan kelembaban udara secara digital.

Materials (Bahan):

* ESP32 sebagai mikrokontroler utama
* Sensor DHT22 untuk membaca suhu dan kelembaban
* Koneksi kabel virtual dalam Wokwi

Semua komponen di atas dapat digantikan dengan simulasi virtual menggunakan Wokwi, sehingga eksperimen dapat dilakukan tanpa perangkat keras fisik.

**2.2 Implementation Steps (Langkah Implementasi)**

Adapun Langkah-langkah implementasi dalam pembuatan sensor suhu kelembaban pada ESP32 menggunakan simulator wokwi :

**1. Persiapan Perangkat Keras dan Perangkat Lunak**

Sebelum memulai implementasi, langkah pertama adalah menyiapkan alat dan bahan:

* **Mikrokontroler**: ESP32
* **Sensor**: DHT22
* **Kabel jumper** untuk koneksi antar komponen
* **Software**:
  + Visual Studio Code (VS Code)
  + PlatformIO (extension pada VS Code)
  + Wokwi Simulator (untuk simulasi tanpa perangkat fisik)

**2. Instalasi dan Konfigurasi Lingkungan Pengembangan**

* Buka **VS Code** dan instal **PlatformIO** sebagai ekstensi.
* Buat proyek baru dan pilih **ESP32** sebagai board.
* Tambahkan pustaka **DHT sensor library** dengan perintah berikut di **platformio.ini :**

lib\_deps = adafruit/DHT sensor library

**3. Penulisan Program untuk Membaca Sensor DHT**

Buat file main.cpp dalam folder src/ dan tuliskan kode berikut:

#include <Arduino.h>

#include <DHT.h>

#define DHTPIN 27      // Pin yang terhubung ke sensor DHT22

#define DHTTYPE DHT22  // Tipe sensor DHT

DHT dht(DHTPIN, DHTTYPE);

void setup() {

  Serial.begin(115200);

  dht.begin();  // Inisialisasi sensor

}

void loop() {

  delay(2000);  // Delay antar pembacaan

  float humidity = dht.readHumidity();

  float temperature = dht.readTemperature();

  // Cek apakah pembacaan gagal

  if (isnan(humidity) || isnan(temperature)) {

    Serial.println("Gagal membaca sensor!");

    return;

  }

  // Tampilkan hasil pembacaan

  Serial.print("Kelembaban: ");

  Serial.print(humidity);

  Serial.print(" %\t");

  Serial.print("Suhu: ");

  Serial.print(temperature);

  Serial.println(" \*C");

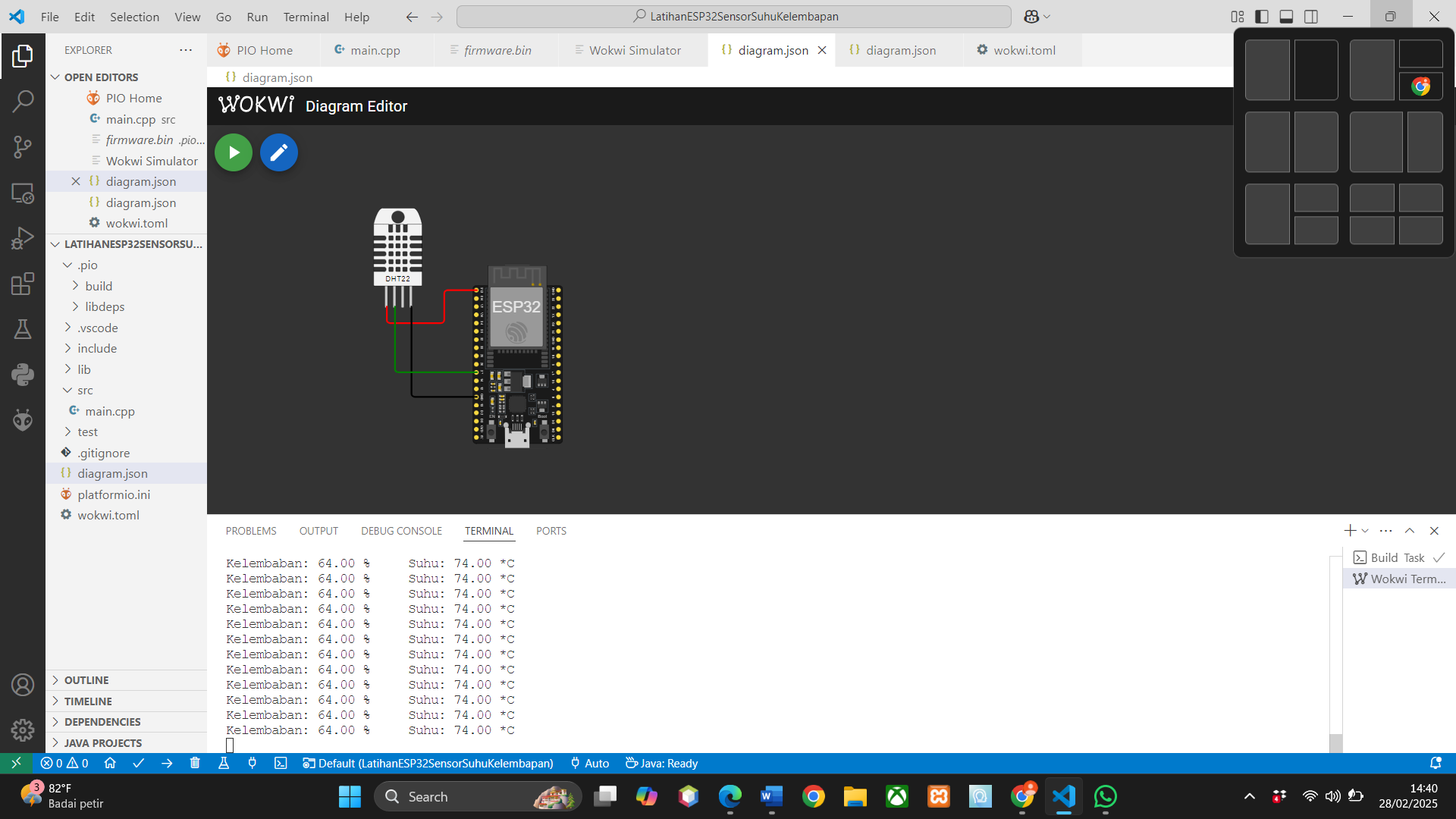
}

**4. Simulasi di Wokwi**

* Tambahkan ESP32 dan DHT22 di Wokwi Simulator
* Hubungkan VCC ke 3.3V, GND ke GND, dan DATA ke GPIO27
* Jalankan simulasi dan lihat output di Serial Monitor

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

**3.1 Experimental Results (Hasil Eksperimen)**



Gambar tersebut menunjukkan simulasi sensor **DHT22** yang digunakan untuk membaca suhu dan kelembaban menggunakan **ESP32** di **Wokwi** dalam **VS Code**. Sensor ini terhubung ke ESP32 dengan tiga kabel: **VCC (merah) untuk daya, GND (hitam) sebagai ground, dan DATA (hijau) ke GPIO27** untuk mengirim data.

Kode program membaca suhu dan kelembaban setiap **2 detik sekali** dan menampilkannya di **Serial Monitor** dengan format seperti:

Kelembaban: 64.0 % Suhu: 74.0 \*C

Nilai ini bersifat tetap karena simulasi, tetapi jika dijalankan dengan perangkat fisik, nilai akan berubah sesuai kondisi lingkungan. Simulasi ini menunjukkan bahwa kode telah berjalan dengan baik dan dapat dikembangkan lebih lanjut untuk proyek IoT seperti pemantauan cuaca atau sistem otomatisasi rumah.

**4. Appendix (Lampiran)**

