

PRAKTIKUM MENAMPILKAN SUHU, KELEMBAPAN, DAN INTENSITAS CAHAYA DALAM MATA KULIAH INTERNET OF THINGS (IoT)

M. Alfian Veda Kumara

Fakultas Vokasi, Universitas Brawijaya

alvianvedkum@student.ub.ac.id

ABSTRAK

Perkembangan Internet of Things (IoT) memungkinkan pengumpulan dan pemantauan data lingkungan secara real-time. Praktikum ini bertujuan untuk merancang dan mengimplementasikan sistem pemantauan suhu, kelembaban, dan intensitas cahaya menggunakan sensor berbasis IoT. Sensor yang digunakan meliputi DHT22 untuk suhu dan kelembaban, serta sensor LDR untuk intensitas cahaya, yang terhubung dengan mikrokontroler berbasis ESP32. Data yang diperoleh dikirimkan ke platform IoT untuk visualisasi dan analisis. Hasil praktikum menunjukkan bahwa sistem dapat berfungsi dengan baik dalam menangkap dan menampilkan data lingkungan secara real-time. Implementasi ini dapat digunakan untuk berbagai aplikasi, seperti sistem pemantauan rumah pintar dan pertanian berbasis IoT.

Kata Kunci: Internet of Things (IoT), suhu, kelembaban, intensitas cahaya, sensor, pemantauan real-time.

ABSTRACT

The development of the Internet of Things (IoT) enables real-time collection and monitoring of environmental data. This practicum aims to design and implement a monitoring system for temperature, humidity, and light intensity using IoT-based sensors. The sensors used include the DHT22 for temperature and humidity and the LDR sensor for light intensity, which are connected to an ESP32-based microcontroller. The collected data is transmitted to an IoT platform for visualization and analysis. The practicum results show that the system functions effectively in capturing and displaying environmental data in real-time. This implementation can be applied to various applications, such as smart home monitoring systems and IoT-based agriculture.

Keywords: Internet of Things (IoT), temperature, humidity, light intensity, sensor, real-time monitoring.

Pendahuluan

Internet of Things (IoT) telah menjadi teknologi yang berkembang pesat dan memiliki berbagai aplikasi dalam kehidupan sehari-hari, terutama dalam bidang pemantauan lingkungan. IoT memungkinkan perangkat elektronik untuk saling terhubung dan berkomunikasi melalui jaringan internet, sehingga data yang dikumpulkan dapat diakses secara real-time. Salah satu penerapan IoT yang penting adalah dalam pemantauan suhu, kelembaban, dan intensitas cahaya, yang banyak digunakan dalam berbagai bidang seperti rumah pintar (*smart home*), pertanian cerdas (*smart farming*), industri, hingga sistem manajemen gedung.

Dalam dunia industri, pemantauan suhu dan kelembapan sangat penting untuk menjaga kondisi optimal dalam berbagai proses produksi, terutama di sektor farmasi, makanan, dan elektronik. Sementara itu, dalam sektor pertanian, pengukuran suhu dan kelembapan udara dapat membantu petani dalam mengoptimalkan kondisi pertumbuhan tanaman serta efisiensi penggunaan air. Selain itu, intensitas cahaya juga merupakan parameter penting yang mempengaruhi produktivitas tanaman dalam sistem pertanian hidroponik maupun konvensional. Oleh karena itu, pemantauan kondisi lingkungan secara *real-time* dengan teknologi IoT dapat meningkatkan efisiensi dan efektivitas dalam berbagai sektor.

Dalam praktikum ini, dilakukan implementasi sistem pemantauan berbasis IoT yang berfungsi untuk mengukur dan menampilkan data suhu, kelembapan, serta intensitas cahaya secara real-time. Sistem ini menggunakan sensor DHT22 untuk mengukur suhu dan kelembapan, serta sensor LDR (Light Dependent Resistor) untuk mengukur intensitas cahaya. Data yang diperoleh dari sensor kemudian dikirimkan ke mikrokontroler berbasis ESP32, yang selanjutnya akan meneruskan data tersebut ke platform IoT untuk divisualisasikan. Dengan sistem ini, pengguna dapat memantau kondisi lingkungan secara langsung melalui perangkat yang terhubung dengan internet, seperti komputer atau smartphone.

Tujuan Praktikum

Praktikum ini memiliki beberapa tujuan utama, di antaranya:

1. Menggunakan sensor DHT22 dan LDR untuk mengukur suhu, kelembapan, dan intensitas cahaya.
2. Menggunakan mikrokontroler ESP32 untuk membaca dan mengirim data sensor ke platform IoT.
3. Menampilkan data sensor secara real-time melalui internet.
4. Menganalisis keakuratan data sensor dan faktor yang memengaruhinya.

Melalui praktikum ini, mahasiswa diharapkan tidak hanya memahami teori, tetapi juga memiliki pengalaman langsung dalam mengimplementasikan sistem otomasi berbasis IoT. Hasil dari praktikum ini dapat menjadi dasar untuk pengembangan proyek-proyek IoT yang lebih kompleks di masa mendatang.

Metodologi

Praktikum ini bertujuan untuk merancang dan menguji sistem pemantauan suhu, kelembapan, dan intensitas cahaya berbasis Internet of Things (IoT) menggunakan alat simulator Wokwi dan editor kode Visual Studio Code (VS Code). Pendekatan yang digunakan mencakup dua aspek utama, yaitu perancangan rangkaian elektronik menggunakan Wokwi dan pengembangan kode pemrograman menggunakan VS Code. Dengan kedua tahapan ini, sistem dapat disimulasikan dan diuji tanpa memerlukan perangkat keras fisik, yang memberikan keuntungan dari segi efisiensi waktu dan sumber daya.

Tahap pertama dalam metodologi ini adalah perancangan sistem menggunakan Wokwi Simulator. Wokwi adalah platform berbasis web yang memungkinkan pengguna untuk merancang dan menguji rangkaian elektronik secara virtual. Dalam simulasi ini, komponen utama yang digunakan adalah ESP32 sebagai mikrokontroler untuk mengendalikan sensor DHT22, sensor LDR, dan LCD 16x2. Sensor DHT22 digunakan untuk mengukur suhu dan kelembapan, sementara sensor LDR digunakan untuk mengukur intensitas cahaya. Semua komponen ini disusun dalam breadboard virtual di Wokwi. Koneksi antar komponen dilakukan menggunakan kabel jumper virtual, di mana pin-pin pada ESP32 dihubungkan ke sensor DHT22, LDR, dan LCD

sesuai dengan desain rangkaian yang diinginkan. Proses perancangan dilakukan dengan seksama untuk memastikan rangkaian bekerja sesuai dengan tujuan praktikum.

Setelah rangkaian selesai dirancang di simulator Wokwi, langkah berikutnya adalah penulisan kode menggunakan Visual Studio Code. VS Code adalah editor yang umum digunakan untuk menulis program berbasis C++ atau platform lain yang kompatibel dengan ESP32, seperti Arduino IDE. Program ini ditulis dengan tujuan mengontrol sensor DHT22 dan LDR, serta menampilkan hasil pengukuran suhu, kelembapan, dan intensitas cahaya pada LCD 16x2. Kode ini mencakup fungsi untuk membaca data dari sensor dan menampilkan hasilnya secara real-time pada LCD menggunakan pustaka LiquidCrystal.h. Selain itu, program juga menyertakan fungsi untuk mengatur koneksi Wi-Fi pada ESP32 agar dapat mengirimkan data ke platform IoT seperti ThingSpeak atau Blynk, jika diperlukan.

Setelah kode selesai ditulis, tahap selanjutnya adalah pengujian menggunakan Wokwi Simulator. Salah satu keuntungan menggunakan Wokwi adalah kemampuannya untuk mengunggah dan menjalankan kode langsung di dalam simulator tanpa memerlukan perangkat keras fisik. Dengan menjalankan program di Wokwi, pengguna dapat memverifikasi apakah kode yang ditulis berfungsi seperti yang diharapkan. Hasil yang diinginkan adalah tampilan suhu, kelembapan, dan intensitas cahaya yang terupdate secara real-time di layar LCD. Jika ditemukan kesalahan dalam simulasi, kode dapat diperbaiki langsung di VS Code dan diuji kembali dalam Wokwi. Proses ini memungkinkan perbaikan dan penyesuaian dilakukan dengan cepat tanpa harus beralih ke perangkat keras fisik.

Simulator Wokwi memberikan simulasi yang sangat realistis dari rangkaian elektronik dan program, memungkinkan identifikasi kesalahan pada desain atau kode sebelum melanjutkan ke implementasi fisik. Proses ini sangat penting dalam merancang sistem IoT, karena menghemat waktu dan mengurangi kemungkinan kesalahan yang dapat terjadi pada tahap implementasi fisik. Dengan simulasi ini, pengguna dapat mengeksplorasi berbagai konfigurasi sensor dan pengaturan LCD, seperti pembaruan data secara berkala atau pengaturan tampilan pada layar LCD. Hasil yang berhasil pada tahap simulasi memberikan keyakinan bahwa sistem berfungsi dengan baik dan siap diimplementasikan.

Secara keseluruhan, metode yang diterapkan dalam praktikum ini menggunakan Wokwi Simulator dan Visual Studio Code memungkinkan pengembangan sistem IoT secara efisien dan terstruktur. Proses perancangan dan pengujian di dalam simulator memberikan kesempatan untuk memodifikasi dan mengoptimalkan desain tanpa harus bergantung pada perangkat keras fisik, yang sangat berguna pada tahap pengembangan awal. Setelah rangkaian dan kode diuji dengan sukses dalam simulasi, implementasi fisik dapat dilakukan dengan lebih percaya diri. Meskipun dalam praktikum ini fokus utama adalah pada perancangan dan pengujian menggunakan Wokwi dan VS Code, metode ini memberikan wawasan yang lebih dalam tentang alur kerja sistem IoT, mulai dari perancangan rangkaian, pengkodean, hingga pengujian perangkat. Dengan demikian, pengguna dapat lebih memahami aplikasi IoT dan siap untuk mengimplementasikannya dalam proyek nyata.

Hasil dan Pembahasan

Pada pengujian yang dilakukan menggunakan Wokwi Simulator, sistem pemantauan suhu, kelembapan, dan intensitas cahaya berbasis ESP32 berhasil berfungsi dengan baik sesuai dengan tujuan yang telah ditetapkan. Data yang ditampilkan di LCD 16x2 sesuai dengan pengukuran yang dilakukan oleh sensor DHT22 untuk suhu dan kelembapan, serta sensor LDR untuk intensitas cahaya. LCD berhasil menampilkan nilai suhu, kelembapan,

dan intensitas cahaya secara *real-time* sesuai dengan yang diharapkan. Data tersebut diperbarui secara berkala, yang menandakan bahwa sistem dapat menangkap dan menampilkan data lingkungan dengan akurat.

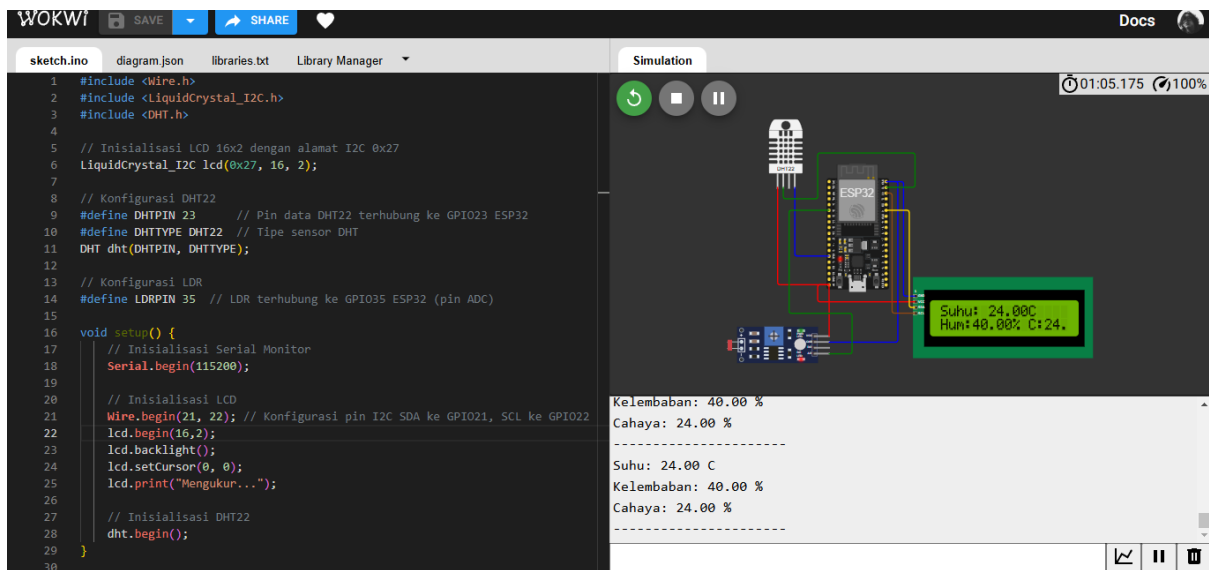
Hasil ini sesuai dengan desain awal yang telah direncanakan, di mana LCD 16x2 berhasil menampilkan nilai suhu, kelembapan, dan intensitas cahaya dengan format yang jelas dan mudah dipahami. Penggunaan Wokwi Simulator memungkinkan untuk memverifikasi fungsi sistem tanpa memerlukan perangkat keras fisik, sehingga dapat menghemat waktu dan mengurangi risiko kesalahan dalam implementasi. Selama simulasi, tidak ditemukan masalah signifikan yang mengganggu jalannya program, dan tampilan di LCD berfungsi sesuai dengan yang diinginkan, yang menandakan bahwa desain rangkaian dan kode sudah diterapkan dengan benar.

Namun, pada pengujian awal ditemukan adanya sedikit perbedaan antara nilai pengukuran yang dihasilkan oleh sensor DHT22 dengan nilai suhu dan kelembapan yang diharapkan. Setelah dilakukan pengecekan dan tinjauan terhadap kode program, ditemukan bahwa kode untuk membaca nilai sensor DHT22 belum sepenuhnya optimal. Penyesuaian dilakukan pada pembacaan data dan penundaan (*delay*) antar pembacaan sensor untuk memastikan hasil yang lebih stabil. Setelah modifikasi, pengukuran suhu dan kelembapan sudah lebih akurat dan sesuai dengan nilai yang diharapkan, membuktikan pentingnya penyesuaian parameter dalam kode untuk menghasilkan data yang lebih tepat.

Pentingnya penggunaan simulator seperti Wokwi sangat terasa dalam tahap ini, karena memungkinkan untuk menguji dan memverifikasi sistem secara menyeluruh sebelum implementasi pada perangkat keras fisik. Simulator ini mempermudah analisis kesalahan atau masalah yang muncul serta memungkinkan pengujian dengan cepat tanpa risiko merusak komponen fisik. Penggunaan Wokwi dalam proses ini sangat efisien, karena memberikan gambaran yang jelas tentang bagaimana sistem akan berfungsi dalam kondisi yang nyata, serta memberi kesempatan untuk memperbaiki kesalahan sebelum implementasi fisik.

Selama pengujian, dilakukan juga beberapa percakapan mengenai potensi pengembangan sistem lebih lanjut. Sebagai contoh, sistem dapat dimodifikasi untuk menghubungkan data sensor dengan platform IoT seperti ThingSpeak atau Blynk, sehingga memungkinkan pemantauan jarak jauh. Sistem juga dapat dikembangkan untuk berfungsi dengan sensor tambahan yang mengatur status tertentu, misalnya mengaktifkan perangkat lain berdasarkan kondisi suhu atau cahaya. Dengan tambahan-tambahan ini, sistem pemantauan ini dapat diaplikasikan pada berbagai sektor, seperti rumah pintar atau pertanian berbasis IoT, yang membutuhkan pemantauan lingkungan secara *real-time*.

Secara keseluruhan, praktikum ini berhasil menunjukkan bahwa penggunaan ESP32 sebagai mikrokontroler utama, sensor DHT22 untuk suhu dan kelembapan, serta sensor LDR untuk intensitas cahaya, dapat menghasilkan sistem yang berfungsi dengan baik. Simulasi yang dilakukan di Wokwi Simulator membuktikan bahwa sistem berjalan sesuai dengan desain yang direncanakan, dan hasil simulasi memberikan gambaran yang jelas tentang integrasi komponen dalam sistem IoT. Melalui pengujian dan simulasi ini, kami dapat memastikan bahwa sistem yang dikembangkan memiliki potensi untuk dikembangkan lebih lanjut dan diaplikasikan pada berbagai solusi teknologi pintar.



Lampiran

Kode Program

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>

// Inisialisasi LCD 16x2 dengan alamat I2C 0x27
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Konfigurasi DHT22
#define DHTPIN 23      // Pin data DHT22 terhubung ke GPIO23 ESP32
#define DHTTYPE DHT22 // Tipe sensor DHT
DHT dht(DHTPIN, DHTTYPE);

// Konfigurasi LDR
#define LDRPIN 35      // LDR terhubung ke GPIO35 ESP32 (pin ADC)

void setup() {
    // Inisialisasi Serial Monitor
    Serial.begin(115200);

    // Inisialisasi LCD
    Wire.begin(21, 22); // Konfigurasi pin I2C SDA ke GPIO21, SCL ke GPIO22
    lcd.begin(16, 2);
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("Mengukur...");

    // Inisialisasi DHT22
    dht.begin();
}

```

```

}

void loop() {
    // Membaca suhu dan kelembaban dari DHT22
    float suhu = dht.readTemperature();
    float kelembaban = dht.readHumidity();

    // Membaca intensitas cahaya dari LDR
    int nilaiLDR = analogRead(LDRPIN);

    // Konversi nilai ADC ke persentase (0 - 100%)
    float intensitasCahaya = map(nilaiLDR, 0, 4095, 0, 100);
    intensitasCahaya = constrain(intensitasCahaya, 0, 100);

    // Menampilkan data ke Serial Monitor
    Serial.print("Suhu: "); Serial.print(suhu); Serial.println(" C");
    Serial.print("Kelembaban: "); Serial.print(kelembaban); Serial.println("
%");
    Serial.print("Cahaya: "); Serial.print(intensitasCahaya); Serial.println("
%");
    Serial.println("-----");

    // Menampilkan data ke LCD
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Suhu: "); lcd.print(suhu); lcd.print("C");
    lcd.setCursor(0, 1);
    lcd.print("Hum:"); lcd.print(kelembaban); lcd.print("% C:");
    lcd.print(intensitasCahaya);

    // Delay sebelum pembacaan berikutnya
    delay(2000);
}

```