**LAPORAN PRAKTIKUM MEMBUAT LIGHT PUSH BUTTON ESP32 MENGGUNAKAN WOKWI**



**Dosen Pengampu :**

Ir. Subairi, ST., MT., IPM

Oleh

Annahl Narendra Wibowo

233140707111053

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS VOKASI**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2025**

**Abstrak**

Internet of Things (IoT) telah berkembang pesat dalam berbagai bidang, termasuk pemantauan lingkungan secara real-time. Pada praktikum ini, dilakukan implementasi sensor suhu dan kelembapan DHT22 serta sensor intensitas cahaya LDR menggunakan mikrokontroler ESP32 yang ditampilkan pada layar OLED. Proyek ini bertujuan untuk membaca, mengolah, dan menampilkan data dari sensor yang dikoneksikan ke ESP32 melalui antarmuka komunikasi digital dan analog. Data yang diperoleh ditampilkan pada serial monitor serta OLED sebagai indikator kondisi lingkungan. Hasil eksperimen menunjukkan bahwa sistem dapat bekerja dengan baik dalam membaca dan menampilkan data dari sensor secara real-time.

*Keywords— IoT, ESP32, DHT22, LDR, OLED, Pemantauan Lingkungan*

1. **Introduction (Pendahuluan)**

Perkembangan teknologi IoT memungkinkan pemantauan lingkungan secara efisien dan real-time. Penggunaan sensor untuk mengukur parameter lingkungan seperti suhu, kelembapan, dan intensitas cahaya sangat penting dalam berbagai aplikasi, seperti sistem pertanian pintar, pengendalian suhu ruangan, dan pemantauan kualitas udara. Dengan kemampuannya yang fleksibel, ESP32 menjadi pilihan utama dalam proyek berbasis IoT karena mendukung komunikasi nirkabel dan memiliki daya pemrosesan yang cukup tinggi.

Dalam praktikum ini, dilakukan integrasi antara ESP32 dengan sensor DHT22 untuk mengukur suhu dan kelembapan, serta sensor LDR untuk mendeteksi intensitas cahaya. Data hasil pengukuran akan ditampilkan pada layar OLED serta serial monitor untuk dianalisis lebih lanjut.

* 1. **Latar Belakang**

Dalam dunia elektronika, tombol dan LED sering digunakan untuk membuat sistem kendali sederhana. Tombol berfungsi sebagai input yang memberikan perintah kepada mikrokontroler, sedangkan LED bertindak sebagai indikator yang menunjukkan hasil dari perintah tersebut. Penggunaan mikrokontroler seperti ESP32 memungkinkan pengendalian LED dengan berbagai pola nyala berdasarkan input tombol.

ESP32 dipilih karena memiliki banyak pin GPIO yang dapat digunakan untuk menghubungkan tombol dan LED secara fleksibel. Dengan memahami konsep ini, kita dapat mengembangkan sistem yang lebih kompleks, seperti sistem kontrol berbasis sensor atau perangkat IoT (Internet of Things). Oleh karena itu, eksperimen ini bertujuan untuk memahami cara kerja tombol dan LED serta bagaimana mikrokontroler dapat digunakan untuk mengontrol pola kedipan LED berdasarkan input tombol.

* 1. **Tujuan Eksperimen**

Tujuan dari eksperimen ini adalah:

* Mengimplementasikan ESP32 sebagai pengendali utama dalam sistem pemantauan lingkungan.
* Membaca dan menampilkan data suhu, kelembapan, dan intensitas cahaya secara real-time.
* Menggunakan OLED sebagai media tampilan visual dari data sensor.
* Menguji efektivitas sistem dalam membaca dan mengolah data sensor.

1. **Methodology (Metodologi)**
   1. **Perancangan Rangkaian:**
   * ESP32 DevKit V1
   * Sensor DHT22 (Suhu dan Kelembapan)
   * Sensor LDR (Light Dependent Resistor)
   * Layar OLED 0.96" I2C
   * Kabel jumper dan breadboard
   * Software Arduino IDE dan Wokwi Simulator

**Diagram Rangkaian**

Rangkaian sistem terdiri dari:

* **DHT22**: Terhubung ke ESP32 menggunakan komunikasi digital (GPIO19)
* **LDR**: Terhubung ke ESP32 melalui pin analog (GPIO32)
* **OLED**: Terhubung menggunakan komunikasi I2C (SCL ke GPIO22, SDA ke GPIO21)
* **Catu daya**: Semua komponen menggunakan tegangan 3.3V dari ESP32

1. **Results and Discussion (Hasil dan pembahasan)**
   1. **Experimental Results**

Hasil project sensor yang dibuat menunjukkan hasil yang didapatkan sebagai berikut :

|  |  |
| --- | --- |
| **Hasil** | **Keterangan** |
| Sensor menampilkan suhu, kelembapan, dan intensitas cahaya | Berhasil |

**Dari Mana Data Ini Berasal?**

**Suhu & Kelembapan**

* Data ini berasal dari sensor **DHT22** yang terhubung ke **GPIO19**.
* Kode program menggunakan **DHT.h** untuk membaca nilai suhu dan kelembapan.

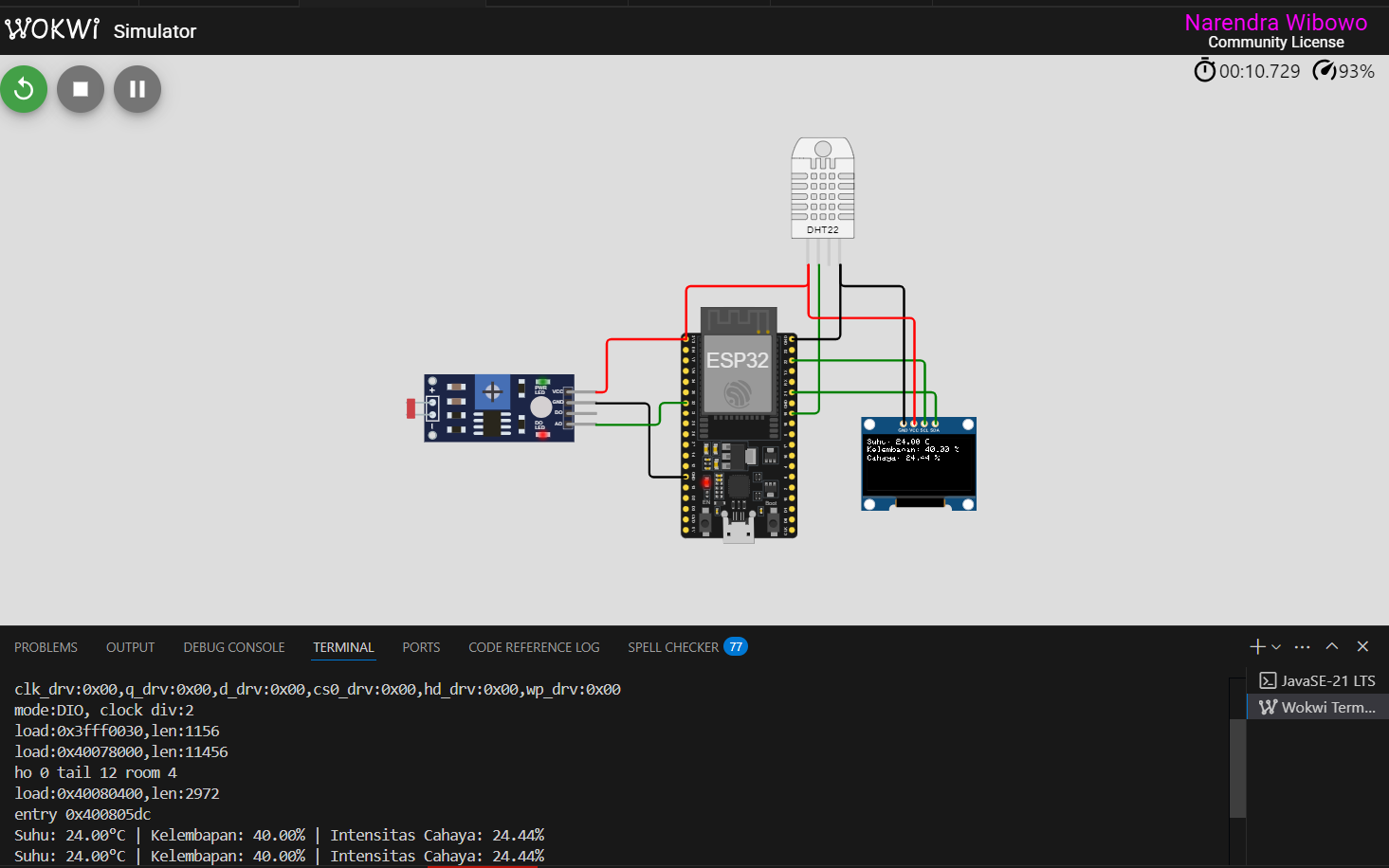
**Intensitas Cahaya**

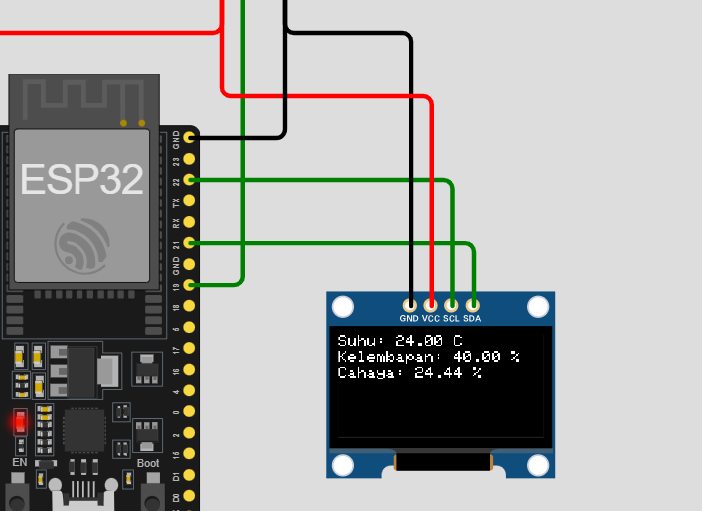
* Data ini berasal dari sensor **LDR (Light Dependent Resistor)** yang terhubung ke **GPIO32**.

**Apakah Hasil Ini Valid?**

* Simulator **Wokwi** tidak membaca sensor fisik, tetapi **mensimulasikan nilai berdasarkan kode**.
* Nilai bisa berbeda dari sensor asli, tetapi **konsep dan cara kerja program tetap benar**.

dokumentasi eksperimen meliputi screenshoot simulasi ESP32 di Visual Studio Code:





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

kode program yang digunakan untuk praktik ini saya menggunakan C++ dengan metode if else dan switch:

1. #include <Arduino.h>

2. #include <Wire.h>

3. #include <Adafruit\_GFX.h>

4. #include <Adafruit\_SSD1306.h>

5. #include <DHT.h>

6.

7. // Definisi OLED

8. #define SCREEN\_WIDTH 128

9. #define SCREEN\_HEIGHT 64

10. #define OLED\_RESET    -1

11. #define SCREEN\_ADDRESS 0x3C

12.

13. Adafruit\_SSD1306 display(SCREEN\_WIDTH, SCREEN\_HEIGHT, &Wire, OLED\_RESET);

14.

15. // Definisi DHT22

16. #define DHTPIN 19

17. #define DHTTYPE DHT22

18. DHT dht(DHTPIN, DHTTYPE);

19.

20. // Definisi LDR

21. #define LDRPIN 32

22.

23. void setup() {

24.     Serial.begin(115200);

25.

26.     // Inisialisasi OLED

27.     if (!display.begin(SSD1306\_SWITCHCAPVCC, SCREEN\_ADDRESS)) {

28.         Serial.println("Gagal menginisialisasi OLED!");

29.         while (true);

30.     }

31.     display.clearDisplay();

32.

33.     // Inisialisasi DHT22

34.     dht.begin();

35. }

36.

37. void loop() {

38.     // Baca sensor DHT22

39.     float suhu = dht.readTemperature();

40.     float kelembapan = dht.readHumidity();

41.

42.     // Baca nilai LDR

43.     int ldrValue = analogRead(LDRPIN);

44.     float intensitas = (ldrValue / 4095.0) \* 100; // Konversi ke persen (untuk ESP32)

45.

46.     // Cek apakah pembacaan DHT22 berhasil

47.     if (isnan(suhu) || isnan(kelembapan)) {

48.         Serial.println("Gagal membaca dari DHT22!");

49.         return;

50.     }

51.

52.     // Tampilkan di Serial Monitor

53.     Serial.print("Suhu: ");

54.     Serial.print(suhu);

55.     Serial.print("°C | Kelembapan: ");

56.     Serial.print(kelembapan);

57.     Serial.print("% | Intensitas Cahaya: ");

58.     Serial.print(intensitas);

59.     Serial.println("%");

60.

61.     // Tampilkan di OLED

62.     display.clearDisplay();

63.     display.setTextSize(1);

64.     display.setTextColor(WHITE);

65.

66.     display.setCursor(0, 0);

67.     display.print("Suhu: ");

68.     display.print(suhu);

69.     display.println(" C");

70.

71.     display.setCursor(0, 10);

72.     display.print("Kelembapan: ");

73.     display.print(kelembapan);

74.     display.println(" %");

75.

76.     display.setCursor(0, 20);

77.     display.print("Cahaya: ");

78.     display.print(intensitas);

79.     display.println(" %");

80.

81.     display.display();

82.

83.     delay(2000); // Tunggu 2 detik sebelum update berikutnya

84. }

85.

86.