**Praktik Pembuatan Rangkaian Pengukuran Suhu, Kelembapan, Dan Intensitas Cahaya**

*Khayru Rifaa Permana*

*Fakultas Vokasi , Universitas Brawijaya*

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

**Abstrak :**

Praktikum ini bertujuan untuk mengembangkan sistem pemantauan lingkungan yang menampilkan data suhu, kelembapan, dan intensitas cahaya pada layar LCD 16x2. Pengukuran suhu dan kelembapan dilakukan dengan memanfaatkan sensor DHT22, sedangkan tingkat intensitas cahaya dideteksi menggunakan sensor LDR. Seluruh proses ini diimplementasikan dengan bantuan mikrokontroler ESP32, yang berfungsi sebagai pusat kendali untuk membaca data dari sensor, mengolah informasi, serta menampilkannya secara real-time pada layar LCD. Praktikum ini dirancang untuk memberikan pemahaman lebih lanjut mengenai integrasi sensor dengan mikrokontroler serta penerapannya dalam sistem monitoring berbasis IoT.

**Kata Kunci :** *Arduino, ESP32, Wokwi, Internet Of Things.*

**Abstract:**

This practicum aims to develop an environmental monitoring system that displays temperature, humidity, and light intensity data on a 16x2 LCD screen. Temperature and humidity measurements are carried out using the DHT22 sensor, while the light intensity level is detected using the LDR sensor. This entire process is implemented with the help of the ESP32 microcontroller, which functions as a control center to read data from sensors, process information, and display it in real-time on the LCD screen. This practicum is designed to provide further understanding of the integration of sensors with microcontrollers and their application in IoT-based monitoring systems.

**Key Word :** *Arduino, ESP32, Wokwi, Internet Of Things.*

**PENDAHULUAN**

Dalam era modern saat ini, pemantauan kondisi lingkungan menjadi aspek yang sangat penting dalam berbagai bidang, seperti pertanian, industri, dan rumah pintar. Informasi mengenai suhu, kelembapan, serta intensitas cahaya sangat dibutuhkan untuk menjaga stabilitas lingkungan, meningkatkan efisiensi energi, dan memastikan kenyamanan serta keamanan dalam berbagai aplikasi.

Salah satu metode yang umum digunakan untuk memperoleh data lingkungan adalah dengan memanfaatkan sensor elektronik yang terintegrasi dengan mikrokontroler. Sensor DHT22 merupakan salah satu sensor yang mampu mengukur suhu dan kelembapan dengan tingkat akurasi yang baik, sementara sensor LDR (Light Dependent Resistor) digunakan untuk mendeteksi intensitas cahaya berdasarkan perubahan resistansinya terhadap pencahayaan sekitar.

Mikrokontroler ESP32 dipilih dalam implementasi sistem ini karena memiliki kemampuan pemrosesan yang cepat, konektivitas nirkabel, serta konsumsi daya yang relatif rendah. Dengan menggunakan ESP32, data dari sensor dapat diproses dan ditampilkan secara langsung pada layar LCD 16x2, memungkinkan pengguna untuk melakukan monitoring kondisi lingkungan secara real-time.

Melalui praktikum ini, diharapkan mahasiswa dapat memahami prinsip kerja sensor DHT22 dan LDR, serta mampu mengintegrasikan berbagai komponen elektronik dengan mikrokontroler untuk membangun sistem pemantauan lingkungan yang efektif. Selain itu, praktikum ini juga memberikan wawasan mengenai pemrograman mikrokontroler dan antarmuka tampilan data menggunakan LCD.

**METODOLOGI**

1. **Alat dan Bahan**

Mikrokontroler ESP32, Sensor LDR, LCD 16x2 dengan modul I2C, Resistor 10KΩ (untuk LDR), Kabel jumper, Breadboard, software Visual Studio Code, PlatformIo, dan platform Wokwi .

1. **Langkah Perancangan**
2. **Pilih ESP32** sebagai mikrokontroler pada situs [Wokwi](https://wokwi.com/).
3. **Tambahkan komponen pendukung** s Sensor LDR, LCD 16x2 dengan modul I2C, Resistor 10KΩ untuk menyusun rangkaian yang lengkap. Pastikan seluruh komponen tersambung dengan benar ke ESP32.
4. **Unduh dan pasang PlatformIO IDE serta Wokwi Simulator** melalui ekstensi pada Visual Studio Code. Selanjutnya, buat proyek baru dengan memilih **Board DOIT ESP32 DEVKIT V1** dan **Framework Arduino** sebagai basis pengembangannya.
5. **Kompilasi proyek** yang telah dibuat dengan menekan tombol **Compile** di bagian kiri bawah aplikasi untuk memastikan tidak ada kesalahan dalam kode sebelum menjalankan simulasi.
6. Buat file wokwi.toml dan isi file dengan code dibawah.

[wokwi]

version = 1

firmware = 'path-to-your-firmware.hex'

elf = 'path-to-your-firmware.elf'

1. Copy path firmware.bin dan firmware.elf lalu paste pada file wokwi.toml.

[wokwi]

version = 1

firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'

elf = '.pio\build\esp32doit-devkit-v1\firmware.elf'

1. Buat file diagram pada project dan copy script diagram.json di website.

{

  "version": 1,

  "author": "Khayru Permana",

  "editor": "wokwi",

  "parts": [

    {

      "type": "board-esp32-devkit-c-v4",

      "id": "esp",

      "top": -30.46,

      "left": -149.9,

      "rotate": 90,

      "attrs": {}

    },

    { "type": "wokwi-dht22", "id": "dht1", "top": -105.3, "left": 148.2, "attrs": {} },

    {

      "type": "wokwi-photoresistor-sensor",

      "id": "ldr1",

      "top": -92.8,

      "left": -124,

      "attrs": {}

    },

    {

      "type": "wokwi-lcd1602",

      "id": "lcd1",

      "top": 169.6,

      "left": 44,

      "attrs": { "pins": "i2c" }

    }

  ],

  "connections": [

    [ "esp:TX", "$serialMonitor:RX", "", [] ],

    [ "esp:RX", "$serialMonitor:TX", "", [] ],

    [ "dht1:VCC", "esp:3V3", "red", [ "v19.2", "h-182.4" ] ],

    [ "dht1:GND", "esp:GND.2", "black", [ "v0" ] ],

    [ "dht1:DATA", "esp:19", "green", [ "v0" ] ],

    [ "ldr1:VCC", "esp:3V3", "red", [ "h19.2", "v67.2", "h-86.4" ] ],

    [ "ldr1:GND", "esp:GND.2", "black", [ "h28.8", "v191.6" ] ],

    [ "ldr1:AO", "esp:34", "red", [ "h9.6", "v47.3", "h-115.2" ] ],

    [ "dht1:SDA", "esp:19", "green", [ "v124.8", "h-259.1" ] ],

    [ "lcd1:VCC", "esp:3V3", "red", [ "h-9.6", "v0.1", "h-19.2", "v-182.55" ] ],

    [ "lcd1:GND", "esp:GND.2", "black", [ "h-67.2", "v-76.8" ] ],

    [ "lcd1:SCL", "esp:22", "limegreen", [ "h-76.8", "v0.3", "h-9.6" ] ],

    [ "lcd1:SDA", "esp:21", "green", [ "h-38.4", "v0.2", "h-76.8" ] ]

  ],

  "dependencies": {}

}

1. **Ubah nama file diagram** menjadi **diagram.json** agar gambar rangkaian dapat ditampilkan dengan benar.
2. Buka file main.cpp di dalam folder src untuk menulis kode yang mengontrol sistem lampu lalu lintas. Atur agar lampu merah menyala selama 30 detik, lampu kuning selama 5detik, dan lampu hijau selama 20 detik. Pastikan program berjalan sesuai dengan konfigurasi pada file diagram.json untuk mensimulasikan rangkaian secara optimal.

#include <Arduino.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

#include <DHT.h>

#define DHTPIN 19

#define DHTTYPE DHT22

#define LDRPIN 34

#define TIMEDHT 1000

float humidity, celsius, lightIntensity;

uint32\_t timerDHT = TIMEDHT;

DHT dht(DHTPIN, DHTTYPE);

LiquidCrystal\_I2C lcd(0x27, 16, 2); // Alamat I2C bisa berbeda (0x3F atau 0x27)

int displayState = 0; // Untuk menyimpan state tampilan LCD

// Fungsi untuk membaca suhu dan kelembapan

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;

    }

  }

}

// Fungsi untuk membaca intensitas cahaya

void getLightIntensity() {

  int ldrValue = analogRead(LDRPIN);

  lightIntensity = map(ldrValue, 0, 4095, 0, 100); // Konversi ke persen

}

void setup() {

  Serial.begin(115200);

  Serial.println("ESP32 dengan DHT22 dan LDR");

  dht.begin();

  lcd.init();

  lcd.backlight();

  lcd.setCursor(0, 0);

  lcd.print("Initializing...");

  delay(2000);

}

void loop() {

  getTemperature();

  getLightIntensity();

  // Menampilkan data ke Serial Monitor

  Serial.print("Temp: ");

  Serial.print(celsius);

  Serial.print(" C, Humidity: ");

  Serial.print(humidity);

  Serial.print(" %, Light: ");

  Serial.print(lightIntensity);

  Serial.println(" %");

  // Menampilkan data secara bergantian di LCD

  lcd.clear();

  if (displayState == 0) {

    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("%");

  } else if (displayState == 1) {

    lcd.setCursor(0, 0);

    lcd.print("Light: ");

    lcd.print(lightIntensity);

    lcd.print("%");

  }

  displayState = (displayState + 1) % 3; // Looping tampilan

  delay(2000); // Delay sebelum berganti tampilan

}

**HASIL DAN PEMBAHASAN**

Sistem pemantauan suhu, kelembapan, dan intensitas cahaya berhasil diimplementasikan menggunakan mikrokontroler ESP32, sensor DHT22, dan sensor LDR, dengan output yang ditampilkan pada layar LCD 16x2 secara real-time. Sensor DHT22 mampu mendeteksi suhu dan kelembapan dengan tingkat akurasi yang cukup baik, meskipun terdapat sedikit keterlambatan dalam pembacaan data akibat karakteristik sensor yang memerlukan waktu stabilisasi. Sementara itu, sensor LDR merespons perubahan intensitas cahaya dengan baik, di mana nilai resistansi berubah sesuai dengan tingkat pencahayaan yang diterima. ESP32 menunjukkan performa optimal dalam membaca data dari sensor dan memperbarui tampilan LCD, dengan keunggulan dalam konsumsi daya yang relatif rendah serta kemampuannya dalam menangani beberapa proses secara bersamaan. Namun, terdapat beberapa kendala yang dihadapi dalam implementasi, seperti flickering pada tampilan LCD saat data diperbarui dan keterlambatan dalam pembacaan sensor DHT22. Untuk mengatasi kendala tersebut, diperlukan optimasi kode program agar pembacaan sensor lebih efisien, serta penggunaan metode filtering atau averaging untuk meningkatkan stabilitas hasil pengukuran. Secara keseluruhan, sistem ini berhasil memenuhi tujuan praktikum dalam menampilkan data suhu, kelembapan, dan intensitas cahaya secara real-time serta memberikan wawasan mengenai integrasi sensor dengan mikrokontroler ESP32.Berikut adalah dokumentasi hasil pengujian praktikum :



