

LAPORAN UJIAN AKHIR SEMESTER

PENDETEKSI POLUSI UDARA BERBASIS IOT DENGAN ESP32

*Baihaqi Rachman, Salma Salsabila
Nasywa Anindya Q.E, Maradu Denny*

Fakultas Vokasi, Universitas Brawijaya

Email :

Abstrak

Polusi udara merupakan salah satu permasalahan lingkungan yang berdampak buruk terhadap kesehatan manusia dan makhluk hidup lainnya. Seiring dengan meningkatnya aktivitas industri dan kendaraan bermotor, kualitas udara semakin menurun dan sulit untuk dipantau secara manual. Oleh karena itu, dibutuhkan suatu sistem yang dapat memantau kualitas udara secara real-time dan terintegrasi dengan teknologi berbasis *Internet of Things (IoT)*. Pada proyek ini, dirancang dan disimulasikan sistem pendeteksi polusi udara menggunakan mikrokontroler *ESP32*, *sensor gas MQ2*, serta indikator LED (merah, kuning, hijau) dan LCD I2C sebagai penampil lokal. Platform *Wokwi* digunakan untuk menyimulasikan perangkat keras, sedangkan integrasi ke aplikasi *Blynk* memungkinkan pemantauan data kualitas udara dari jarak jauh melalui jaringan internet. Sistem ini diharapkan mampu memberikan informasi langsung mengenai tingkat polusi udara secara efisien dan akurat.

Kata kunci: *IoT, ESP32, MQ2, Wokwi, Blynk*

A. PENDAHULUAN

Perkembangan teknologi yang pesat dewasa ini telah memberikan dampak signifikan dalam berbagai aspek kehidupan manusia, termasuk dalam hal pemantauan lingkungan. Salah satu permasalahan lingkungan yang semakin meningkat dan mendapat perhatian luas adalah polusi udara. Polusi udara merupakan pencemaran udara oleh berbagai zat berbahaya yang berasal dari aktivitas manusia maupun alam, seperti asap kendaraan bermotor, emisi industri, kebakaran hutan, dan lain sebagainya. Dampak dari polusi udara tidak hanya dirasakan oleh lingkungan, tetapi juga berpengaruh secara langsung terhadap kesehatan manusia, seperti gangguan pernapasan, iritasi mata, bahkan penyakit kronis seperti asma dan kanker paru-paru. Dalam menghadapi masalah tersebut, dibutuhkan solusi yang mampu memberikan informasi kualitas udara secara cepat dan akurat agar masyarakat dapat segera mengambil tindakan preventif. Salah satu pendekatan yang dapat digunakan adalah pemanfaatan teknologi berbasis *Internet of Things (IoT)*. *IoT* memungkinkan berbagai perangkat elektronik saling terhubung dan berkomunikasi melalui jaringan internet, sehingga data dapat dikirim dan diakses dari mana saja secara real-time.

Melalui proyek ini, dirancang sistem pendeteksi polusi udara berbasis IoT menggunakan mikrokontroler ESP32 yang terhubung dengan sensor gas MQ2. Sensor ini mampu mendeteksi keberadaan gas berbahaya seperti asap, LPG, butana, metana, dan alkohol di udara. Data hasil deteksi ditampilkan secara lokal melalui LCD dan indikator LED berwarna (merah, kuning, hijau) untuk memberikan peringatan visual terhadap kondisi udara. Selain itu, sistem juga terhubung ke aplikasi Blynk melalui jaringan WiFi untuk memungkinkan pemantauan jarak jauh secara real-time. Dengan memanfaatkan simulator Wokwi, perancangan sistem ini dapat diuji secara virtual sebelum direalisasikan secara fisik, sehingga lebih efisien dalam tahap pengembangan.

B. LATAR BELAKANG

Kualitas udara yang buruk telah menjadi perhatian global karena dampaknya yang sangat merugikan bagi kesehatan masyarakat dan keberlangsungan lingkungan. Menurut data dari World Health Organization (WHO), jutaan kematian setiap tahunnya berkaitan langsung dengan paparan udara yang tercemar. Di negara berkembang, di mana tingkat industrialisasi dan urbanisasi meningkat pesat, masalah ini menjadi semakin krusial. Sayangnya, tidak semua wilayah memiliki sistem pemantauan udara yang memadai, baik karena keterbatasan anggaran maupun akses teknologi. Secara tradisional, sistem pemantauan udara menggunakan perangkat-perangkat berskala besar dan mahal yang hanya tersedia di lokasi tertentu, seperti stasiun meteorologi atau pusat riset lingkungan. Hal ini menyebabkan kurangnya ketersediaan data yang merata dan real-time, khususnya di daerah padat penduduk atau kawasan rawan polusi. Untuk itu, diperlukan sebuah sistem yang tidak hanya murah dan portabel, tetapi juga mudah digunakan serta mampu menyajikan data secara cepat.

IoT hadir sebagai solusi modern dalam pengembangan sistem pemantauan lingkungan. Teknologi ini memungkinkan perangkat seperti sensor dan mikrokontroler saling terhubung dan mengirimkan data ke cloud atau aplikasi melalui jaringan internet. Dalam proyek ini, mikrokontroler ESP32 dipilih karena memiliki kemampuan WiFi internal dan performa pemrosesan yang cukup tinggi, sehingga sangat cocok untuk sistem monitoring berbasis IoT. Sensor gas MQ2 digunakan untuk mendeteksi keberadaan gas berbahaya di udara, sedangkan indikator LED dan LCD digunakan sebagai sistem peringatan dan penampil lokal. Dengan bantuan platform simulasi Wokwi, seluruh rangkaian dapat diuji tanpa perlu perangkat keras fisik, memungkinkan pengembangan lebih cepat dan minim risiko. Setelah sistem tervalidasi, kode dan rancangan dapat dipindahkan ke lingkungan pengembangan seperti Visual Studio Code (VSCode) untuk pengelolaan proyek lebih lanjut. Data yang dikirim ke aplikasi Blynk membuat pengguna dapat memantau kualitas udara di sekitarnya kapan saja dan di mana saja, bahkan dari ponsel mereka.

C. TUJUAN PERANCANGAN

1. Merancang dan mensimulasikan sistem pendeteksi polusi udara berbasis ESP32 dan sensor MQ2.
2. Mengembangkan sistem yang mampu memberikan informasi secara lokal melalui LCD dan LED indikator.
3. Mengintegrasikan sistem dengan platform Blynk agar data kualitas udara dapat dipantau secara real-time dari jarak jauh.

D. METODOLOGI

ALAT & BAHAN

Alat dan bahan yang dibutuhkan untuk melakukan simulasi ini adalah:

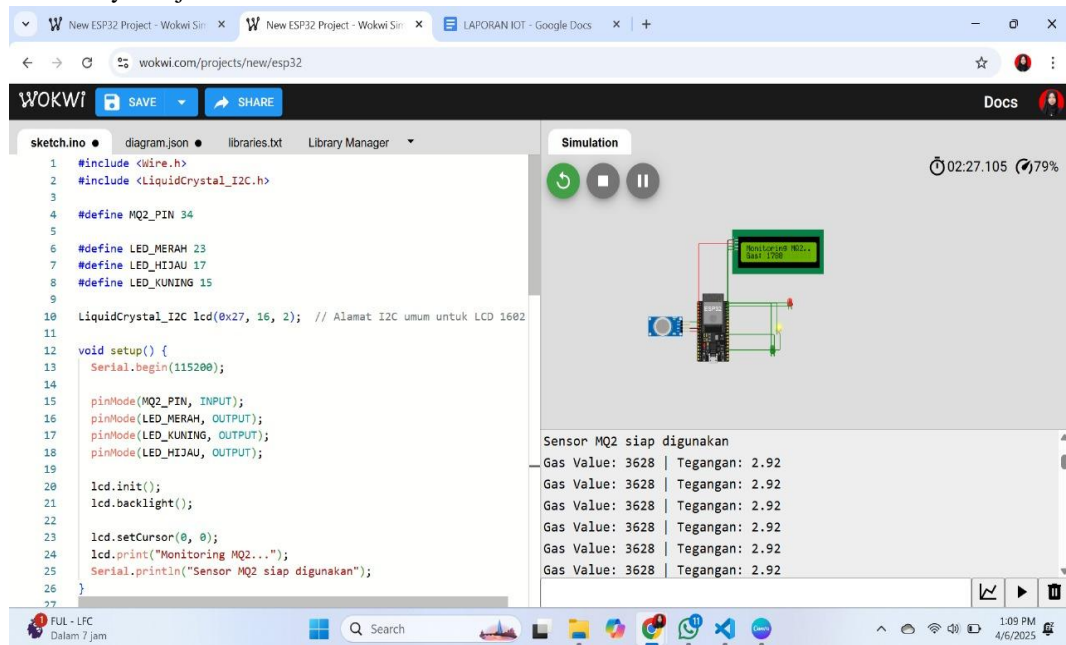
1. Mikrokontroler ESP32
2. LCD 12C
3. LED merah, kuning dan hijau
4. Sensor gas MQ2
5. Kabel jumper untuk mengkoneksikan

Sedangkan platform yang digunakan untuk merancang simulasi adalah wokwi website yang digunakan untuk mendesain arsitektur sensor, yang kemudian akan dipindahkan ke vscode untuk mempermudah dalam pengoperasiannya. Blynk sebagai konektor sensor dengan jaringan internet sehingga hasil dapat dipantau dimana saja secara real-time melalui blynk.

LANGKAH IMPLEMENTASI

1. Masuk ke wokwi web

Langkah pertama yang dilakukan adalah membuat rancangan di wokwi web serta melakukan pengujian apakah kodenya berjalan



Setelah desain sensor jadi serta kode dapat berjalan, salin kode pada wokwi web dan masukkan ke dalam vscode.

2. Membuat template di blynk

Langkah awal adalah login ke website blynk dan buat template baru dengan nama Sensor Udara, selanjutnya salin token,id template, serta nama template untuk nantinya ditempel pada bagian kode agar sensor terkoneksi dengan blynk.

Selanjutnya dalam set up datastream agar sesuai pin agar nantinya output sensor pada widget dapat terkoneksi.

The first screenshot shows the Blynk Console interface for a template named 'Sensor Udara'. The left sidebar contains navigation options: Dashboards, Developer Zone (selected), Devices, Automations, Users, Organizations, Locations, Demand Response, Fleet Management, and In-App Messaging. The main area displays the 'Home' section for the template, showing '1 Devices' (PRO) and a list of devices including 'Sensor Udara'. A 'What's next?' section lists four steps: 'Configure template', 'Set Up Datastreams', 'Set up the Web Dashboard', and 'Add first Device'. The 'Template settings' section shows 'ESP32, WIFI' and 'Firmware configuration' instructions. A code block displays the following code:

```
#define BLYNK_TEMPLATE_ID  
"TMPL6KPzhVNEF"  
#define BLYNK_TEMPLATE_NAME "Sensor  
Udara"
```

The second screenshot shows the 'Datastreams' section for the 'Sensor Udara' template. It features a search bar and a table with 3 datastreams. A 'New Datastream' button is visible in the top right corner.

	Id	Name	Pin	Color	Data Type	Units
1		Indikator gas	V4		Integer	ppm
2		Teks Label	V5		String	fal
3		Indikator LED	V6		Integer	fal

3. Menjalankan kode di vscode

- a. Buat projek platformIo baru di vscode
- b. Tambahkan file diagram dengan kode yang telah diuji coba pada website wokwi tadi
- c. Masukkan kode program kedalam src main.cpp
- d. Tambahkan juga library lcd 12c pada bagian lib_deps dalam file platformio.ini
- e. Tambahkan juga file wokwi.toml dengan isi sebagai berikut

```
[wokwi]
version = 1
firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'
elf = '.pio\build\esp32doit-devkit-v1\firmware.elf'
```

- f. Setelah itu build code dan pastikan tidak ada yang error
- g. Masuk ke dalam website blynk dan cek apakah pada bagian template Sensor Udara terdapat indikasin On, jika terdapat on maka kode dan blynk sudah dapat terhubung.
- h. Setelah itu atur widget pada website blynk sesuai dengan apa yang dibutuhkan.

E. HASIL DAN PEMBAHASAN

Hasil

1. Simulasi Wokwi Berjalan Sukses:

Sensor MQ2 berhasil membaca nilai gas dan menampilkannya di terminal serial serta LCD.

Nilai gas dan tegangan ditampilkan secara real-time.

LCD menampilkan pesan seperti:

```
makefile
CopyEdit
Monitoring MQ2...
Gas: 804
```

2. **Output Serial** menunjukkan pembacaan nilai gas dan tegangan:
Contoh: Gas Value: 3698 Tegangan: 2.92
3. **LED indikator** (dalam bentuk LED bar graph) menyala sesuai level gas:
Semakin tinggi nilai gas, maka led merah akan menyala.
4. **Integrasi dengan Blynk berhasil:**

Data sensor ditampilkan pada aplikasi Blynk dengan status kategori kualitas udara, seperti:

Aman (Hijau), **Waspada** (Kuning), **Bahaya!** (Merah).

Notifikasi dan indikator LED di Blynk sesuai dengan level gas dari sensor MQ2.

Pembahasan

1. **Sensor MQ2:**
Digunakan untuk mendeteksi keberadaan gas berbahaya di udara seperti LPG, asap, dan gas lain.
Menghasilkan nilai analog yang kemudian dikonversi menjadi tegangan oleh ESP32.
2. **ESP32:**
Bertindak sebagai pusat pemrosesan.
Membaca nilai analog dari sensor MQ2.
Menampilkan data ke **LCD 16x2**.
Mengirim data ke **Blynk Cloud** via koneksi Wi-Fi.
3. **LCD 16x2:**
Menampilkan status pemantauan dan nilai gas secara langsung di hardware.
4. **LED Bar / LED indikator:**
Digunakan untuk memberikan indikasi visual terhadap level gas yang terdeteksi.
Biasanya LED hijau (aman), kuning (waspada), dan merah (bahaya).
5. **Blynk App:**
Menyajikan antarmuka monitoring berbasis mobile.
Memungkinkan pengguna melihat status udara dari jarak jauh secara real-time.
Status seperti “Aman”, “Waspada”, dan “Bahaya!” muncul berdasarkan ambang batas nilai gas.

F. HASIL UJI COBA HARDWARE

ALAT DAN BAHAN

1. Breadboard
2. Kabel Jumper
3. Sensor MQ-2
4. Led Merah
5. LCD I2C 16x2
6. ESP 32
7. Kabel USB
8. Laptop
9. Korek

LANGKAH – LANGKAH PERAKITAN FISIK

1. Hubungkan Sensor MQ-2 ke ESP32

- Sensor MQ-2 memiliki 4 pin: VCC, GND, DO (Digital Out), AO (Analog Out). Dalam proyek ini, kita hanya menggunakan pin DO.
- Sambungkan VCC MQ-2 ke 3.3V ESP32
- Sambungkan GND MQ-2 ke GND ESP32
- Sambungkan DO MQ-2 ke GPIO 34 di ESP32
- Pin DO akan memberikan sinyal LOW ketika gas terdeteksi.

2. Hubungkan LED sebagai Indikator

- LED akan menyala ketika gas terdeteksi.
- Hubungkan kaki pendek (katoda) LED ke GND (melalui jalur GND di breadboard)

3. Hubungkan LCD 16x2 I2C ke ESP32

- LCD I2C hanya menggunakan 4 kabel: VCC, GND, SDA, dan SCL.
- Sambungkan VCC LCD ke 3.3V ESP32
- Sambungkan GND LCD ke GND ESP32
- Sambungkan SDA LCD ke GPIO 21 ESP32
- Sambungkan SCL LCD ke GPIO 22 ESP32

LANGKAH – LANGKAH DI ARDUINO IDE

1. Install Board ESP32

Jika belum:

Buka Arduino IDE > Preferences

Tambahkan ini ke Additional Board Manager URLs:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

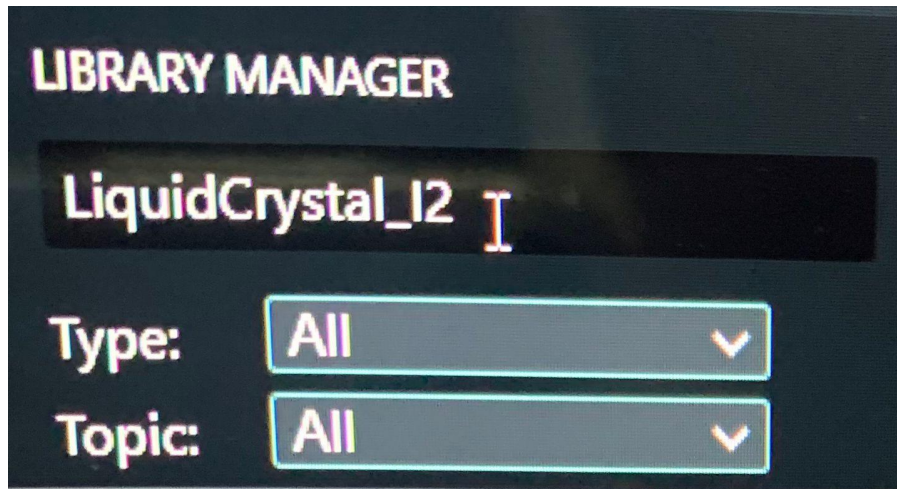
Lalu buka Tools > Board > Board Manager, cari dan install esp32.

2. Install Library

Buka Sketch > Include Library > Manage Libraries, lalu:

Cari dan install LiquidCrystal_I2C (by Frank de Brabander)

Cari dan install Wire



3. Source Code

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#define MQ2_DO_PIN 34    // Pin digital dari sensor MQ-2

#define LED_PIN 19       // Pin untuk LED indikator

#define LCD_ADDR 0x27    // Alamat I2C LCD (ubah jadi 0x3F kalau tidak tampil)

LiquidCrystal_I2C lcd(LCD_ADDR, 16, 2);

void setup() {

    Serial.begin(115200);

    pinMode(MQ2_DO_PIN, INPUT);

    pinMode(LED_PIN, OUTPUT); // Inisialisasi LED

    lcd.init();

    lcd.backlight();

    lcd.setCursor(0, 0);
```



```

lcd.print("Status MQ-2:");

}

void loop() {

    int gasStatus = digitalRead(MQ2_DO_PIN);

    lcd.setCursor(0, 1);

    if (gasStatus == LOW) {

        Serial.println("Gas terdeteksi!");

        lcd.print("Gas terdeteksi ");

        digitalWrite(LED_PIN, HIGH); // Nyalakan LED

    } else {

        Serial.println("Aman.");

        lcd.print("Aman          ");

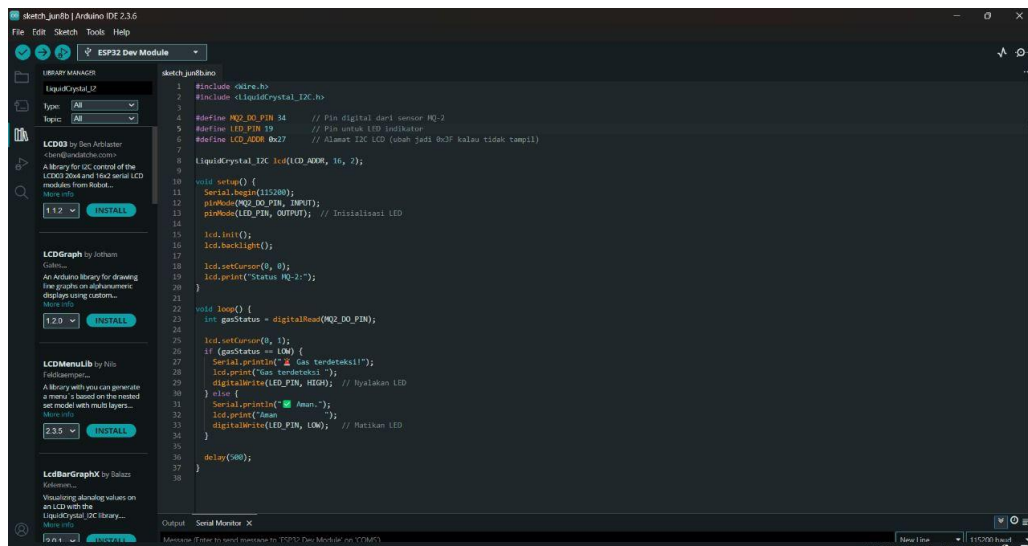
        digitalWrite(LED_PIN, LOW); // Matikan LED

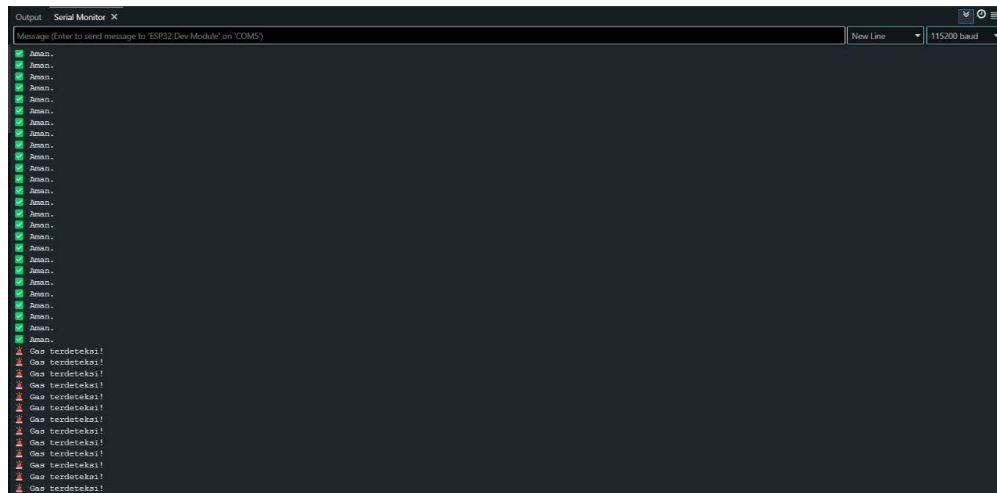
    }

    delay(500);

}

```

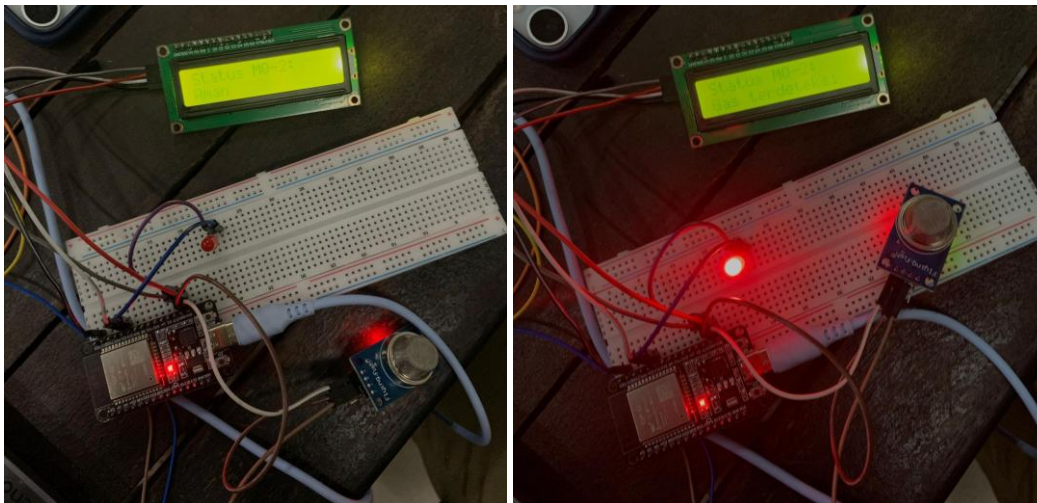




G. Cara Pengujian

1. Sambungkan ESP32 ke laptop dengan kabel USB
2. Upload sketch dari Arduino IDE
3. Buka Serial Monitor dengan baud rate 115200 untuk melihat nilai gas
4. Dekatkan korek api atau asap ke sensor MQ-2 (jangan dibakar langsung)
5. Perhatikan:
6. LCD menampilkan: Gas terdeteksi
7. LED menyala
8. Jika tidak ada gas: Aman, dan LED mati

Hasil Output



Kesimpulan

Simulasi ini menunjukkan bahwa sistem monitoring kualitas udara berbasis ESP32 dan sensor MQ2 dapat bekerja dengan baik di Wokwi Simulator, dan integrasi dengan Blynk berhasil untuk keperluan pemantauan jarak jauh. Sistem ini sangat berguna sebagai prototype untuk aplikasi smart home atau sistem deteksi gas berbahaya.

LAMPIRAN

