



**LAPORAN FINAL PROJECT SISTEM EMBEDDED  
DEPARTEMEN TEKNIK ELEKTRO  
UNIVERSITAS INDONESIA**

**“GASGUARD” SMART GAS LEAKAGE DETECTION SYSTEM**

**KELOMPOK 24**

<b>ABEDNEGO ZEBUA</b>	<b>2306161883</b>
<b>ARSINTA KIRANA NISA</b>	<b>2306215980</b>
<b>CHRISTOVER ANGELO LASUT</b>	<b>2306220343</b>
<b>GEDE RAMA</b>	<b>2306161914</b>

## **PENDAHULUAN**

Segala puji dan syukur kami panjatkan kehadiran Allah SWT atas rahmat dan karunia-Nya sehingga kami dapat menyelesaikan proyek akhir mengenai sistem deteksi kebocoran gas karbon monoksida tepat pada waktunya.

Proyek ini kami buat sebagai bagian dari pemenuhan tugas akhir pada mata kuliah Sistem Embedded. Proyek ini merupakan usaha kami dalam menciptakan solusi keamanan yang efektif untuk mendeteksi gas berbahaya secara dini, yang sangat penting dalam menjaga keselamatan lingkungan dan kesehatan manusia. Dari proyek ini, kami menyadari betapa pentingnya sistem monitoring gas yang tidak hanya akurat namun juga responsif dan praktis untuk digunakan dalam berbagai kondisi.

Kami mengucapkan terima kasih kepada seluruh anggota Kelompok 24 yang telah bekerja sama dalam pembuatan program, perancangan alat, simulasi, dan juga penyusunan laporan sehingga proyek ini dapat selesai dengan hasil yang memuaskan. Terima kasih kepada asisten laboratorium yang telah membimbing dan memberikan arahan selama proyek ini dibuat. Kami menyadari bahwa proyek akhir kami masih jauh dari kata sempurna, oleh sebab itu kami sangat mengharapkan kritik dan saran yang membangun dari pembaca agar proyek ini dapat dikembangkan dan dapat memberi manfaat yang lebih besar di masa depan.

Depok, 18 Mei 2025

Kelompok 24

## DAFTAR ISI

<b>BAB I.....</b>	<b>3</b>
<b>PENDAHULUAN.....</b>	<b>3</b>
1.1 LATAR BELAKANG MASALAH.....	3
1.2 SOLUSI.....	3
1.3 KRITERIA STANDAR.....	4
1.4 PEMBAGIAN TUGAS.....	4
1.5 TIMELINE DAN MILESTONES.....	5
<b>BAB 2.....</b>	<b>5</b>
<b>IMPLEMENTASI.....</b>	<b>5</b>
2.1 DESAIN RANGKAIAN DAN SKEMATIK.....	5
2.2 PENGEMBANGAN SOFTWARE.....	7
2.3 INTEGRASI HARDWARE DAN SOFTWARE.....	8
<b>BAB 3.....</b>	<b>9</b>
<b>PENGUJIAN DAN EVALUASI.....</b>	<b>9</b>
3.1 PENGUJIAN.....	9
3.2 HASIL.....	11
3.3 EVALUASI.....	12
<b>BAB 4.....</b>	<b>13</b>
<b>KESIMPULAN.....</b>	<b>13</b>

# **BAB I**

## **PENDAHULUAN**

### **1.1 LATAR BELAKANG MASALAH**

Karbon monoksida (CO) merupakan gas beracun yang sangat berbahaya karena bersifat tidak berwarna, tidak berbau, dan tidak berasa, sehingga sulit dikenali oleh manusia tanpa bantuan alat. Gas ini dapat terbentuk dari proses pembakaran tidak sempurna pada bahan bakar seperti gas elpiji, kayu, batu bara, dan minyak tanah. Dalam ruangan tertutup atau dengan ventilasi buruk, akumulasi CO dapat terjadi dengan cepat dan tidak disadari, yang dapat menyebabkan gangguan kesehatan serius seperti pusing, mual, gangguan pernapasan, kehilangan kesadaran, bahkan kematian.

Tingginya risiko yang ditimbulkan oleh paparan gas karbon monoksida menjadikan deteksi dini kebocoran gas sebagai kebutuhan penting, khususnya pada rumah tangga, dapur industri, dan lingkungan tertutup lainnya. Sayangnya, masih banyak sistem deteksi yang tidak responsif atau mahal secara biaya. Oleh karena itu, diperlukan sebuah solusi yang cerdas, ekonomis, dan andal untuk mendeteksi kebocoran gas secara real-time dan memberikan peringatan secara visual maupun suara kepada pengguna sebelum kondisi menjadi kritis.

### **1.2 SOLUSI**

Sebagai respon terhadap permasalahan tersebut, proyek GasGuard – Smart Gas Leakage Detection System dirancang untuk menjadi solusi dalam mendeteksi kebocoran gas karbon monoksida secara otomatis. Sistem ini menggunakan sensor gas MQ-7 yang mampu membaca konsentrasi CO di lingkungan sekitar dan memberikan output analog yang kemudian diolah oleh mikrokontroler ATmega328P (pada papan Arduino Uno).

Seluruh sistem dikendalikan menggunakan bahasa pemrograman Assembly AVR, yang memberikan akses langsung ke perangkat keras dan memungkinkan optimasi performa, konsumsi daya, serta kecepatan respon sistem. GasGuard memberikan peringatan melalui LED berwarna—hijau untuk aman, kuning untuk peringatan, dan merah untuk bahaya—serta alarm buzzer jika level gas telah melebihi ambang batas. Sistem ini juga menampilkan nilai konsentrasi gas dalam bentuk angka digital melalui 7-segment display yang dikendalikan

oleh IC MAX7219. Dengan pendekatan ini, GasGuard mampu memberikan peringatan secara cepat dan jelas kepada pengguna, serta menjadi prototipe sistem deteksi gas yang ekonomis dan dapat diimplementasikan secara luas.

### 1.3 KRITERIA STANDAR

Proyek ini memiliki kriteria standar agar berfungsi optimal, responsif, dan andal sebagai berikut:

1. Sistem mampu mendeteksi kadar gas karbon monoksida (CO) dengan sensor MQ-7 dan ADC internal Atmega328P secara real time.
2. Level gas diklasifikasikan menjadi tiga kategori berdasarkan nilai ADC:
  - Aman ( $ADC < 100$ ) → LED hijau menyala, buzzer mati.
  - Peringatan ( $100 \leq ADC < 200$ ) → LED kuning menyala, buzzer mati.
  - Bahaya ( $ADC \geq 200$ ) → LED merah menyala, buzzer aktif sebagai alarm.
3. Level gas ditampilkan dalam format digital pada 7-segment display menggunakan driver MAX7219 dengan tampilan jelas dan stabil.
4. Sistem harus mampu beroperasi terus-menerus tanpa gangguan, dengan logika program utama diimplementasikan dalam bahasa Assembly AVR.
5. Tersedia tombol reset yang terhubung ke interrupt eksternal untuk menginisialisasi ulang sistem atau memicu mode pengujian.
6. Sistem mengirimkan status dan nilai ADC secara serial melalui UART untuk monitoring eksternal.

### 1.4 PEMBAGIAN TUGAS

Peran dan tugas dalam kelompok kami adalah sebagai berikut:

NAMA	TUGAS
Abednego Zebua	Membuat rangkaian awal simulasi proteus, membuat kode test masing-masing komponen dan serta kode utama, dan finishing serta debugging pada rangkaian fisik.

Arsinta Kirana Nisa	Mencari alat-alat yang diperlukan untuk membuat rangkaian, membantu pembuatan rangkaian, membuat laporan dan PPT.
Christover Angelo	Membantu membuat rangkaian, laporan, pembuatan PPT, serta testing rangkaian fisik.
Gede Rama	Membantu dalam pembuatan PPT, rangkaian dan testing rangkaian fisik.

### 1.5 TIMELINE DAN MILESTONES

Berikut adalah timeline dari pengerjaan proyek akhir kami:

1. 11 Mei 2025 - Pembuatan desain rangkaian dan pembuatan kode untuk menguji parts pada rangkaian.
2. 12 Mei 2025 - 17 Mei 2025 - Pengembangan dan debugging code agar berjalan sesuai dengan kriteria standar.
3. 17 Mei 2025 - Pembuatan rangkaian fisik.
4. 18 Mei 2025 - Pengujian final product code dan rangkaian untuk memastikan produk bekerja sesuai kriteria.

## **BAB 2**

### **IMPLEMENTASI**

#### **2.1 DESAIN RANGKAIAN DAN SKEMATIK**

Hardware GasGuard dirancang menggunakan satu buah Arduino Uno (mikrokontroler ATmega328P) yang mengontrol seluruh sistem deteksi gas karbon monoksida secara terpadu. Sistem ini terdiri dari beberapa komponen utama yang saling terintegrasi untuk memastikan performa optimal dan responsif.

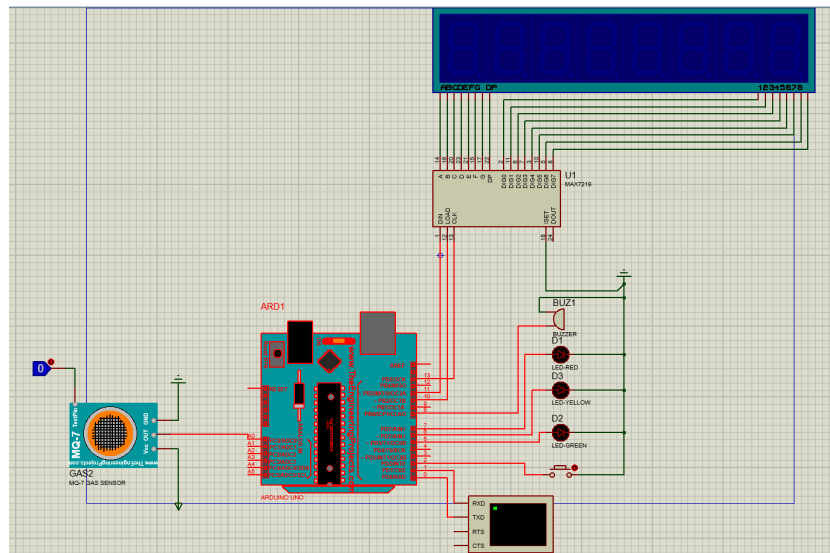
Sensor MQ-7 terhubung pada pin analog ADC0 (PC0/A0) Arduino Uno untuk mendeteksi konsentrasi gas CO di lingkungan sekitar melalui output analognya. Sinyal dari sensor ini dibaca oleh ADC internal mikrokontroler untuk diproses lebih lanjut.

Untuk menampilkan level gas secara digital, sistem menggunakan 4-digit 7-segment display yang dikendalikan oleh IC MAX7219 melalui protokol SPI. Pin SPI pada Arduino yang digunakan adalah MOSI (PB3/11), SCK (PB5/13), dan CS (PB2/10). MAX7219 mengelola tampilan angka dengan stabil dan jelas sehingga memudahkan pengguna membaca level gas secara realtime.

Tiga buah LED indikator disambungkan pada pin digital PD5 (LED Hijau), PD6 (LED Kuning), dan PD7 (LED Merah), masing-masing menunjukkan status level gas: aman, peringatan, dan bahaya. LED ini memberikan tanda visual langsung sesuai kondisi gas yang terdeteksi.

Sistem juga dilengkapi dengan buzzer aktif yang terhubung ke pin PB0 (Digital 8) untuk mengeluarkan suara alarm ketika kadar gas memasuki level bahaya. Alarm ini hanya aktif saat diperlukan agar menghemat daya dan menghindari gangguan.

Sebuah push button disambungkan pada pin PD2 (Digital 2) dan dikonfigurasi sebagai interrupt eksternal (INT0). Tombol ini berfungsi untuk mereset sistem atau mengaktifkan mode pengujian, memastikan kendali pengguna terhadap sistem.



Desain hardware ini memastikan GasGuard dapat bekerja secara efisien, memberikan output visual dan audio yang jelas, serta responsif terhadap perubahan kadar gas di lingkungan sekitar. Dengan integrasi mikrokontroler, sensor, tampilan digital, indikator LED, buzzer, dan tombol reset, sistem mampu memberikan perlindungan dini yang efektif dan mudah dioperasikan.

## 2.2 PENGEMBANGAN SOFTWARE

Software pada proyek ini dikembangkan sepenuhnya menggunakan bahasa Assembly AVR khusus untuk mikrokontroler ATmega328P pada Arduino Uno. Kode ini memiliki tugas untuk membaca data analog dari sensor MQ-7 melalui ADC internal, mengendalikan indikator LED, alarm buzzer, dan menampilkan nilai konsentrasi gas pada 7-Segment display yang dikendalikan oleh IC MAX7219.

Main program akan menginisialisasi seluruh periferal seperti GPIO, UART, ADC, SPI, dan interrupt eksternal untuk tombol reset. Pada main loop, mikrokontroler secara berkala memulai konversi ADC untuk membaca sensor MQ-7, kemudian nilai ADC tersebut dikonversi menjadi tampilan desimal dan ditampilkan pada 7-segment display.

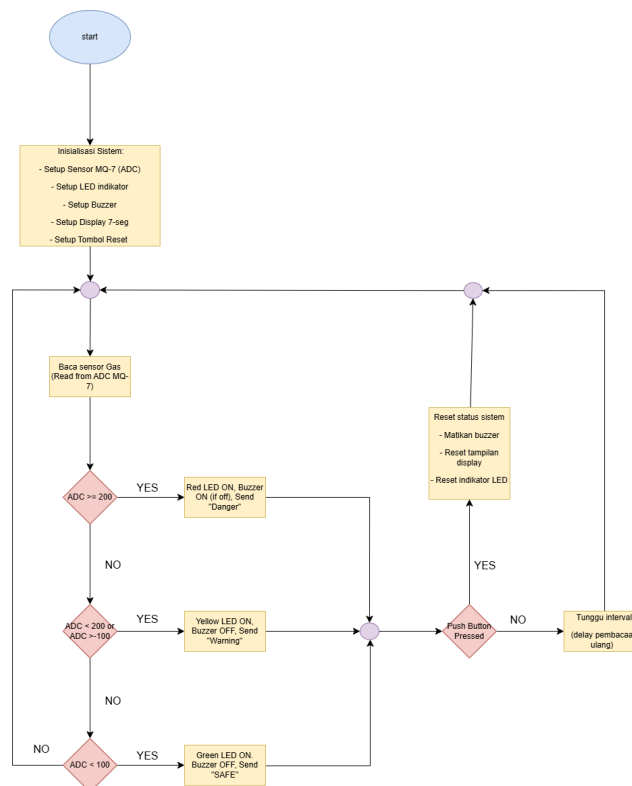
Lalu berdasarkan nilai ADC, program mengevaluasi level gas dan mengatur LED yang sesuai (hijau untuk aman, kuning untuk peringatan, merah untuk bahaya). Buzzer aktif hanya saat level bahaya dan dikontrol dengan flag latch agar tidak terus menyala tanpa henti. Status juga dikirim via UART sebagai monitoring serial eksternal.

Interrupt eksternal pada push button memungkinkan pengguna untuk mereset buzzer dan menampilkan pola tampilan test pada display. Semua fungsi pendukung seperti delay, SPI



write, dan UART transmission juga diimplementasikan secara manual dalam Assembly untuk mengoptimalkan performa dan penggunaan sumber daya.

Berikut adalah flowchart general dari cara kerja alat “GasGuard”.



Secara sederhana alur kerja alat ini dapat dijelaskan oleh langkah-langkah berikut:

1. Inisialisasi sistem dan perangkat keras: stack pointer, I/O port (LED, buzzer, tombol, SPI), UART, ADC, MAX7219 display, dan interrupt eksternal tombol reset.
2. Menjalankan startup sequence: buzzer test, LED blink sequence, dan menampilkan pola uji pada display 7-seg.
3. Memulai loop utama yang berjalan terus menerus.
4. Membaca data sensor gas MQ-7 melalui ADC internal mikrokontroler.

Menampilkan nilai pembacaan sensor pada 7-segment display menggunakan IC MAX7219.

5. Mengevaluasi level gas berdasarkan nilai ADC:
  - Jika nilai ADC kurang dari 100, LED hijau menyala dan buzzer mati (level aman).
  - Jika nilai ADC antara 100 hingga 199, LED kuning menyala dan buzzer mati (peringatan).
  - Jika nilai ADC 200 atau lebih, LED merah menyala dan buzzer aktif sebagai alarm (bahaya).

6. Mengirimkan status dan nilai ADC via UART untuk monitoring eksternal.
7. Memberikan delay sekitar 500 ms sebelum membaca sensor ulang.
8. Jika tombol reset ditekan (interrupt INT0), buzzer dimatikan, tampilan display direset ke pola default, dan pesan reset dikirim via UART.
9. Program kembali ke loop utama untuk membaca sensor dan memonitor level gas secara real-time.

## 2.3 INTEGRASI HARDWARE DAN SOFTWARE

Integrasi antara hardware dan software dimulai dengan perancangan skematik menggunakan Proteus untuk memastikan semua komponen terhubung dengan benar dan sesuai pin mikrokontroler. Hardware utama meliputi sensor MQ-7 pada pin ADC0, tiga LED indikator pada port D pin 5,6,7, buzzer pada port B pin 0, tombol reset pada port D pin 2, dan IC MAX7219 untuk mengendalikan 7-segment display melalui SPI (pin PB2, PB3, PB5).

Setelah desain skematik diverifikasi, kode Assembly yang telah dibuat diunggah ke mikrokontroler Arduino Uno untuk mengatur perilaku hardware secara real-time. Sistem diuji untuk memastikan pembacaan sensor MQ-7 dapat diubah menjadi output digital pada display dan indikator LED sesuai level gas. Fungsi buzzer dan tombol reset juga diuji agar bekerja sesuai logika yang diharapkan.

Pengujian integrasi ini memastikan bahwa software mampu mengendalikan hardware dengan baik, memberikan notifikasi visual dan audio yang jelas dan responsif terhadap perubahan kadar gas. Pendekatan pemrograman Assembly memungkinkan kontrol presisi dan efisiensi pemrosesan pada mikrokontroler yang digunakan.

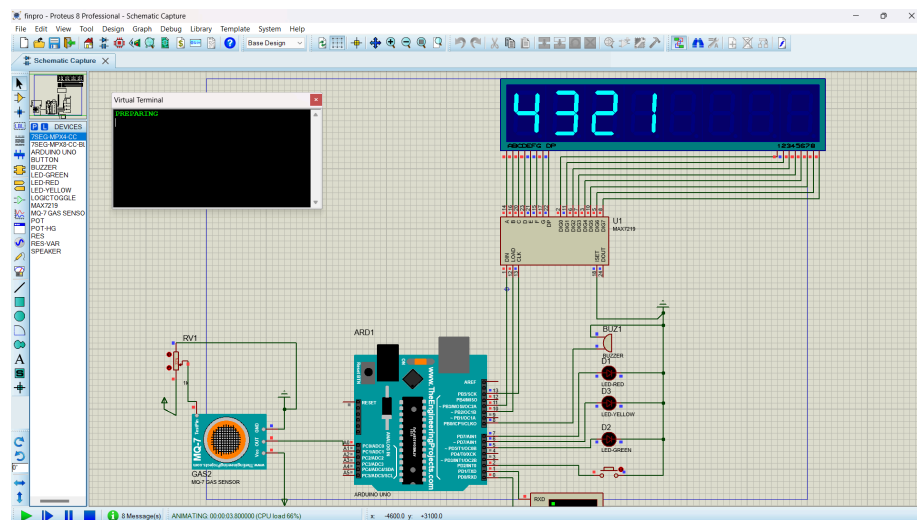
## BAB 3

### PENGUJIAN DAN EVALUASI

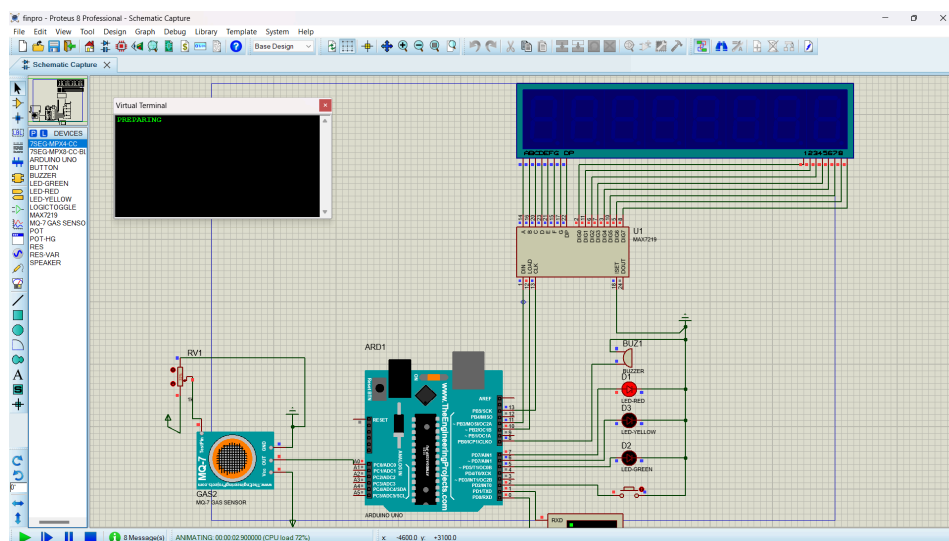
#### 3.1 PENGUJIAN

Sebelum dilakukan test pada rangkaian asli, kami menggunakan proteus untuk melakukan simulasi kecil untuk memastikan bahwa setiap komponen seperti LED, MAX7219, Buzzer, dan Sensor berjalan dengan baik. Pada proteus, karena tidak bisa memberikan input analog pada sensor MQ-7 maka akan dilakukan testing hingga tahap inisialisasi dapat berjalan dengan lancar.

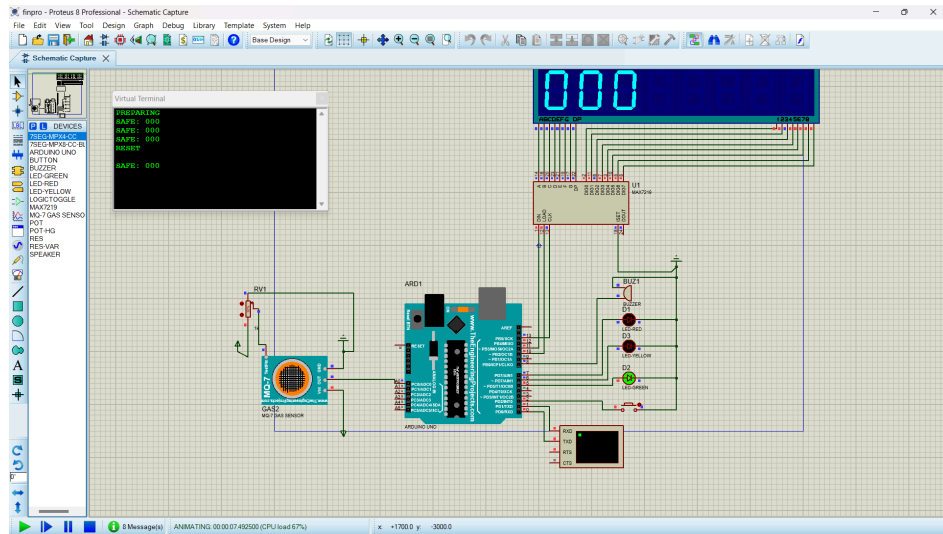
- Preparing GasGuard



- LED Prepare



- Reset Button

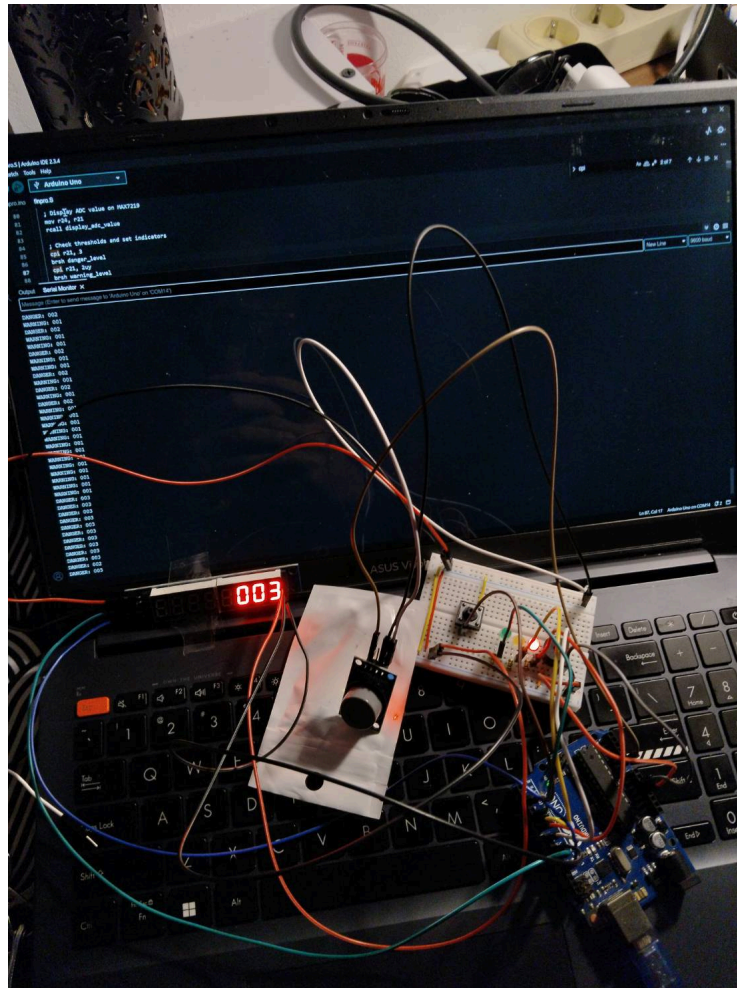


Pada tahap inisialisasi awal, sistem berhasil menampilkan pesan "PREPARING" melalui komunikasi UART sebagai tanda startup berhasil dijalankan. Selain itu, pengujian awal perangkat keras menunjukkan bahwa LED dan buzzer bekerja dengan baik, di mana LED menyala secara bergantian. Tampilan awal pada 7-segment juga berhasil menampilkan angka "1234", yang menandakan bahwa perangkat telah siap untuk digunakan.

### 3.2 HASIL

Pada tahap pembacaan sensor, sensor MQ-7 memberikan output analog yang kemudian dibaca oleh ADC internal dari mikrokontroler ATmega328P melalui pin PC0 (ADC0). Karena kondisi sensor belum sepenuhnya optimal dan respons terhadap gas karbon monoksida (CO) belum stabil, maka batas ambang (threshold) ADC disesuaikan agar sistem tetap dapat menunjukkan perubahan level gas secara responsif. Rentang nilai ADC yang digunakan adalah sebagai berikut (nilai untuk pengetesan, dapat disesuaikan melalui kode):

- $ADC \geq 3 \rightarrow$  Bahaya (LED merah menyala dan buzzer aktif)
- $ADC \geq 2 \rightarrow$  Waspada (LED kuning menyala)
- $ADC < 2 \rightarrow$  Aman (LED hijau menyala)



Untuk bagian tampilan dan indikator, sistem berhasil menampilkan nilai ADC secara real-time melalui 7-segment display menggunakan IC MAX7219. Indikator LED menyala sesuai dengan level bahaya yang telah ditentukan, dan sistem juga mengirimkan pesan status melalui komunikasi UART ke Serial Monitor Arduino IDE, seperti:

- SAFE: 001
- SAFE: 002
- WARNING: 002
- DANGER: 003

yang mencerminkan nilai ADC yang sedang terbaca oleh sensor.

Push button yang sebagai input reset eksternal terhubung pada pin PD2 juga berfungsi dengan baik. Ketika push button ditekan, buzzer akan langsung dimatikan dan tampilan pada 7-Segment akan reset ke “1234” lalu pesan RESET berhasil terkirim melalui UART sebagai feedback ke user.

Secara keseluruhan, sistem terbukti berjalan stabil selama pengujian. Program mampu melakukan loop pembacaan ADC secara terus-menerus tanpa mengalami crash. Komunikasi SPI dan pembacaan ADC berlangsung paralel tanpa konflik, dan delay antar pembacaan cukup untuk menampilkan perubahan nilai secara akurat tanpa kehilangan data atau gangguan performa.

### 3.3 EVALUASI

Secara keseluruhan, komponen telah berfungsi sesuai peran masing-masing. Sensor dapat memberikan output analog yang terbaca oleh mikrokontroler, LED dapat menyala sesuai level gas CO, dan tampilan pada 7-Segment berhasil menampilkan nilai ADC secara real-time. Sistem juga memberikan peringatan melalui kombinasi visual dan audio, sehingga user dapat mengenali kondisi gas di sekitar.

Namun, terdapat beberapa kendala yang ditemukan selama pengujian. Sensor MQ-7 membutuhkan waktu pemanasan dan kalibrasi yang cukup lama, serta memiliki respons yang kurang konsisten pada fase awal pengujian. Hal ini menyebabkan pembacaan ADC sulit dijadikan acuan akurat terhadap konsentrasi gas CO. Oleh karena itu, threshold ADC disesuaikan (diturunkan) agar fungsi sistem dapat tetap divisualisasikan dengan baik dalam skenario demo. Selain itu, sistem belum mampu menampilkan nilai konsentrasi gas dalam satuan ppm secara real-time karena keterbatasan kalibrasi sensor.

Sebagai solusi sementara, nilai ambang ADC diatur secara manual menjadi 3 (bahaya) dan 2 (waspada) untuk memastikan bahwa fungsi indikator dan buzzer dapat ditunjukkan dengan jelas. Tombol interrupt juga berhasil diuji untuk melakukan reset sistem dengan cepat, baik pada kondisi normal maupun saat buzzer sedang aktif.

Untuk pengembangan lebih lanjut, rekomendasi kami adalah menambahkan delay atau filtering terhadap ADC supaya noise pembacaan yang tiba-tiba akan berkurang. Selain itu, penggunaan sensor gas digital atau modul kalibrasi gas CO akan membantu menghasilkan data yang lebih akurat dan terukur dalam satuan ppm. Lalu, fitur logging data sensor bisa diintegrasikan ke komputer melalui UART sehingga user bisa memantau pembacaan sensor dari waktu ke waktu.

## **BAB 4**

### **KESIMPULAN**

Proyek akhir ini dibuat untuk membangun sebuah sistem yang dapat mendeteksi kebocoran gas karbon monoksida (CO) dengan akurat. Sistem ini dibuat dengan indikator visual audio dan audio berbasis mikrokontroler ATmega328P dan pemrograman bahasa Assembly AVR. Sistem ini mengintegrasikan sensor MQ-7 untuk membaca konsentrasi gas, indikator LED berwarna sebagai penanda tingkat bahaya, buzzer sebagai alarm, serta display 7-segment berbasis IC MAX7219 untuk menampilkan nilai pembacaan sensor secara real-time. Selain itu, tombol reset eksternal disediakan untuk mereset sistem dengan cepat dan memberikan umpan balik kepada pengguna melalui komunikasi UART.

Dari hasil uji yang dijalankan, sistem dapat berjalan stabil dan memberikan repons sesuai. inisialisasi sistem berjalan lancar yang ditandai dengan pesan “PREPARING” pada UART, uji coba buzzer, dan tampilan awal “1234” pada 7-Segment. Sistem juga mampu untuk membaca output analog dari sensor dan menyesuaikan tingkat bahaya berdasarkan threshold ADC, dan memberikan peringatan baik menggunakan LED dan display maupun pada serial monitor.

Namun, terdapat kendala pada aspek kalibrasi sensor. MQ-7 memerlukan waktu pemanasan dan memiliki respons awal yang kurang stabil, sehingga nilai ambang ADC perlu diturunkan untuk memungkinkan sistem mendeteksi perubahan dengan cepat. Karena keterbatasan ini pula, nilai konsentrasi gas belum dapat ditampilkan secara aktual dalam satuan ppm.

Secara keseluruhan, sistem telah memenuhi sebagian besar kriteria standar dan berhasil menunjukkan fungsi utama dengan baik. Keandalan visual dan audio cukup kuat untuk menandakan status lingkungan secara langsung. Untuk pengembangan selanjutnya, kami merekomendasikan penambahan filter ADC untuk mengurangi noise, penggunaan sensor gas yang sudah terkalibrasi, serta integrasi fitur logging data melalui UART agar pengguna dapat memantau histori pembacaan sensor secara lebih akurat dan berkelanjutan.

## REFERENSI

1. "Assembly via Arduino - Programming IO Ports," *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/10/assembly-via-arduino-programming-io.html>
2. "Assembly via Arduino - Programming ADC." *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/10/assembly-via-arduino-programming-adc.html>
3. "Assembly via Arduino - Programming Serial Port," *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/10/assembly-via-arduino-programming-serial.html> (accessed May 18, 2025).
4. "Assembly via Arduino - ADC Values on Serial Monitor," *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/10/assembly-via-arduino-adc-values-on.html> (accessed May 18, 2025).
5. "Assembly via Arduino - Programming SPI," *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/11/assembly-via-arduino-programming-spi.html> (accessed May 18, 2025).
6. "Assembly via Arduino - Programming MAX7219," *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/11/assembly-via-arduino-programming-max7219.html> (accessed May 18, 2025).
7. "Assembly via Arduino - Programming Timer 1." *Blogspot.com*, 2021.  
<https://akuzechie.blogspot.com/2021/11/assembly-via-arduino-programming-timer-1.html>