



**CYBER-PHYSICAL SYSTEM FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA**

TOXIC GASSES DETECTOR

GROUP 4

MIRANTI ANGGUNSARI	2106731472
ALIYAH RIZKY AL-AFIFAH POLANDA	2206024682
KEVIN NAUFAL ARYANTO	2206062850
KANIA AIDILLA FIRKA	2206062983

PREFACE

Puji syukur kami panjatkan kehadiran Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan proyek akhir untuk Praktikum Sistem Siber Fisik ini dengan baik dan tepat waktu. Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan praktikum yang telah kami jalani selama satu semester.

Proyek akhir ini berfokus pada implementasi dan pengembangan sistem siber fisik yang merupakan integrasi erat antara komputasi, jaringan, dan proses fisik. Melalui praktikum ini, kami memperoleh kesempatan untuk menerapkan teori-teori yang telah dipelajari di kelas ke dalam praktik nyata, mengasah kemampuan analitis, serta meningkatkan keterampilan teknis dalam bidang ini.

Kami menyadari bahwa dalam penyusunan laporan ini, banyak pihak yang telah memberikan bantuan dan dukungan, baik secara langsung maupun tidak langsung. Oleh karena itu, kami ingin menyampaikan terima kasih yang sebesar-besarnya kepada dosen pengampu mata kuliah Sistem Siber Fisik, asisten laboratorium, dan teman-teman satu tim atas kerja sama, komitmen, dan kontribusinya dalam menyelesaikan proyek ini.

Kami menyadari bahwa laporan ini masih jauh dari sempurna dan masih banyak kekurangan yang perlu diperbaiki. Oleh karena itu, kritik dan saran yang membangun sangat kami harapkan demi perbaikan dan peningkatan kualitas di masa mendatang.

Akhir kata, kami berharap laporan ini dapat bermanfaat bagi pembaca dan memberikan kontribusi positif bagi perkembangan ilmu pengetahuan di bidang sistem siber fisik. Semoga Tuhan Yang Maha Esa senantiasa memberikan rahmat dan hidayah-Nya kepada kita semua.

Depok, May 27, 2024

Group 4

TABLE OF CONTENTS

CHAPTER 1.....	4
INTRODUCTION.....	4
1.1 PROBLEM STATEMENT.....	4
1.3 ACCEPTANCE CRITERIA.....	5
1.4 ROLES AND RESPONSIBILITIES.....	6
1.5 TIMELINE AND MILESTONES.....	6
CHAPTER 2.....	8
IMPLEMENTATION.....	8
2.1 HARDWARE DESIGN AND SCHEMATICS.....	8
2.2 SOFTWARE DEVELOPMENT.....	9
2.3 HARDWARE AND SOFTWARE INTEGRATION.....	14
CHAPTER 3.....	17
TESTING AND EVALUATION.....	17
3.1 TESTING.....	17
3.2 RESULT.....	17
3.3 EVALUATION.....	18
CHAPTER 4.....	19
CONCLUSION.....	19

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

Dalam era modern saat ini, keberadaan gas berbahaya dalam ruangan dapat menjadi ancaman serius bagi kesehatan manusia dan keamanan lingkungan. Gas-gas seperti LPG, karbon monoksida (CO), metana (CH₄), propana, dan gas berbahaya lainnya sering kali tidak terdeteksi oleh indera manusia sehingga menimbulkan risiko tersembunyi. Oleh karena itu, diperlukan sebuah sistem yang dapat secara real-time mendeteksi keberadaan gas berbahaya dan mengambil tindakan pencegahan yang diperlukan untuk memastikan keselamatan.

Proyek ini bertujuan untuk mengembangkan sebuah sistem deteksi konsentrasi gas berbahaya dalam ruangan menggunakan sensor gas MQ. Sistem ini akan memantau konsentrasi gas secara kontinu dan jika konsentrasi gas melebihi ambang batas yang telah ditetapkan, indikator seperti buzzer akan menyala sebagai peringatan. Selain itu, untuk menampilkan tingkat konsentrasi gas yang terdeteksi, sistem akan menggunakan modul display MAX7219 yang terhubung melalui protokol komunikasi Serial Peripheral Interface (SPI). Sebagai langkah mitigasi tambahan, ketika terdeteksi konsentrasi gas berbahaya yang tinggi, sistem ini juga akan mengaktifkan exhaust fan untuk membuang gas berbahaya dari ruangan, sehingga membantu menjaga kualitas udara dan mengurangi potensi bahaya.

Dengan implementasi sistem ini, diharapkan dapat tercipta lingkungan yang lebih aman dan terhindar dari risiko paparan gas berbahaya yang tidak terdeteksi. Laporan akhir praktikum ini akan membahas secara rinci desain sistem, implementasi perangkat keras dan perangkat lunak, serta evaluasi kinerja sistem dalam kondisi nyata.

1.2 PROPOSED SOLUTION

Dalam proyek ini, kami bertujuan untuk mendeteksi konsentrasi gas berbahaya dalam ruangan menggunakan sensor gas MQ. Sistem ini dirancang untuk memberikan peringatan visual dan auditori serta mengevakuasi gas berbahaya yang terdeteksi. Berikut adalah rincian solusi yang ditujukan:

- MQ-2 Gas Sensor: Sensor ini akan mendeteksi konsentrasi berbagai jenis gas berbahaya di dalam ruangan, seperti karbon monoksida (CO), metana (CH₄), dan gas berbahaya lainnya.
- Mikrokontroler: Mikrokontroler akan digunakan untuk membaca data dari sensor gas dan mengendalikan komponen lain dalam sistem.
- Buzzer: Komponen ini akan digunakan sebagai indikator peringatan. Buzzer akan menyala jika konsentrasi gas berbahaya melebihi batas yang ditetapkan.
- MAX7219: Modul ini akan digunakan untuk menampilkan status udara berdasarkan kandungan gas secara real-time menggunakan protokol SPI.
- Exhaust Fan: Kipas ini akan diaktifkan untuk membuang gas berbahaya dari dalam ruangan jika deteksi menunjukkan konsentrasi yang melebihi ambang batas.

Proses Kerja Sistem:

- Deteksi Gas: Sensor MQ-2 akan mendeteksi konsentrasi gas berbahaya di udara dan mengirimkan data analog ke mikrokontroler.
- Pengolahan Data: Mikrokontroler akan mengolah data dari sensor dan membandingkannya dengan batas konsentrasi yang telah ditetapkan.
- Indikator Peringatan: Jika konsentrasi gas melebihi batas, mikrokontroler akan mengaktifkan buzzer sebagai indikator peringatan.
- Tampilan Pembacaan Sensor: Status udara akan ditampilkan pada modul MAX7219 menggunakan protokol SPI, sehingga memungkinkan pemantauan real-time. Apabila tidak terdapat gas berbahaya yang terdeteksi, maka MAX7219 akan menampilkan “SAFE” dan apabila terdeteksi adanya gas berbahaya, maka MAX7219 akan menampilkan “DANGER”.
- Evakuasi Gas: Jika terdapat gas berbahaya yang terdeteksi, mikrokontroler akan mengaktifkan exhaust fan untuk mengeluarkan gas berbahaya dari ruangan, sehingga mengurangi risiko bahaya.

1.3 ACCEPTANCE CRITERIA

Toxic Gases Detector dapat dikatakan berhasil berjalan apabila memenuhi syarat-syarat berikut:

1. Sensor gas MQ-2 dapat mendeteksi adanya kandungan gas berbahaya, seperti LPG, alkohol, asap, propana, hidrogen, metana, maupun karbon monoksida pada lingkungan dan status kualitas udara akan ditampilkan oleh Arduino pada MAX7219.
2. Data yang dikirim dari Arduino dapat ditampilkan pada Modul 7 Segment MAX7219.
3. Buzzer dapat menyala apabila terdapat kandungan gas berbahaya yang terdeteksi oleh sensor gas MQ-2.
4. Kipas dapat berfungsi apabila terdapat kandungan gas berbahaya yang terdeteksi oleh sensor gas MQ-2.

1.4 ROLES AND RESPONSIBILITIES

The roles and responsibilities assigned to the group members are as follows:

Roles	Responsibilities	Person
MQ-2 Programmer	Membuat kode untuk mendeteksi gas berbahaya dengan sensor gas MQ-2	Miranti Anggunsari
MAX7219 Programmer	Membuat kode untuk menerima data hasil pembacaan sensor dan menampilkannya di Modul 7 Segment MAX7219	Aliyah Rizky Al-Afifah Polanda
Proteus Designer	Membuat desain rangkaian di Proteus	Kevin Naufal Aryanto
Menyusun Laporan	Membuat flowchart serta mengedit laporan	Kania Aidilla Firka

Table 1. Roles and Responsibilities

1.5 TIMELINE AND MILESTONES

TASK		May																														
		8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31							
PLANNING																																
1.1	Brainstorming Idea																															
1.2	Konsultasi																															
1.3	Menentukan Komponen																															
ANALYSIS																																
2.1	Pembagian Tugas																															
2.2	Scheduling																															
DESIGN																																
3.1	Mendesain Rangkaian Proteus																															
IMPLEMENTATION																																
4.1	Menulis Kode Assembly																															
4.2	Membuat Rangkaian Asli																															
4.3	Menyusun Laporan																															
TESTING																																
5.1	Testing																															
5.2	Presentasi Proyek																															

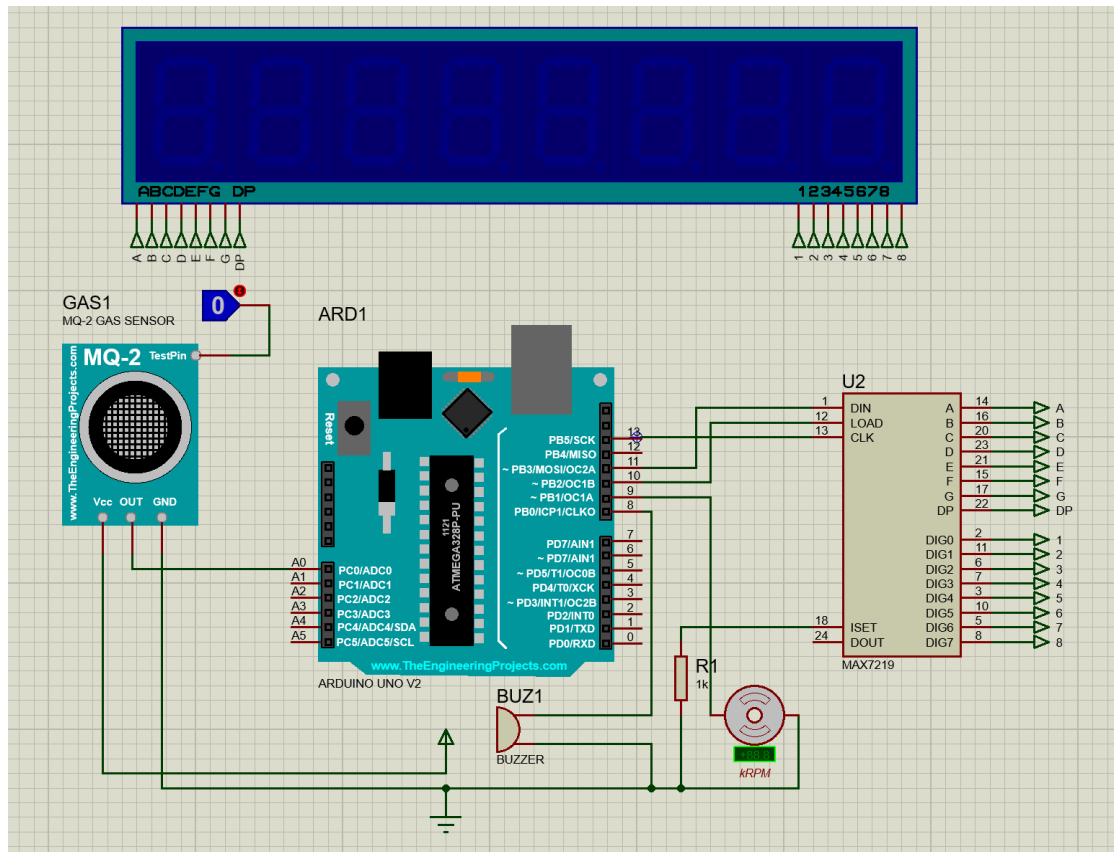
Table 2. Team Schedule

- Brainstorming: Ide-ide dikumpulkan dan didiskusikan oleh tim. Tahap ini adalah saat untuk mengeksplorasi berbagai solusi potensial dan mendefinisikan tujuan proyek secara keseluruhan.
- Konsultasi: Tim berkonsultasi dengan asisten laboratorium untuk mendapatkan masukan, klarifikasi, dan bimbingan mengenai proyek yang sedang dikerjakan. Tahap ini bertujuan untuk memastikan bahwa konsep yang dikembangkan sesuai dengan standar dan kebutuhan yang ada.
- Menentukan Komponen: Semua komponen yang diperlukan untuk proyek dipilih dan dipesan. Tahap ini mencakup identifikasi spesifikasi teknis dan kompatibilitas komponen yang akan digunakan.
- Membuat Rangkaian Proteus: Desain rangkaian dibuat dan disimulasikan menggunakan Proteus. Tahap ini membantu dalam memvisualisasikan dan memvalidasi desain sebelum pembangunan fisik dimulai.
- Menulis code: Tahapan dimana pengembangan kode dilakukan, yang mencakup penulisan, pengujian, dan debugging program. Kode ini akan mengatur fungsi-fungsi spesifik dari proyek yang dirancang.
- Mengimplementasikan rangkaian: Rangkaian yang telah dirancang dan disimulasikan di Proteus disusun secara fisik.
- Menyusun laporan: Tim menyusun dokumentasi lengkap tentang proyek, termasuk hasil, analisis, dan kesimpulan. Laporan ini berfungsi sebagai catatan resmi dan panduan untuk evaluasi lebih lanjut.
- Testing: Sistem diuji untuk memastikan semua komponen berfungsi sesuai dengan yang diharapkan.

CHAPTER 2

IMPLEMENTATION

2.1 HARDWARE DESIGN AND SCHEMATICS



Pic 1. Proteus Design

a. Mikrokontroler

- Menggunakan Arduino UNO sebagai pusat kontrol.
- Pin digital akan digunakan untuk komunikasi dengan sensor gas, buzzer, display MAX7219, dan kontrol exhaust fan.

b. Sensor Gas MQ2

- Sensor gas MQ2 dihubungkan ke salah satu pin analog pada Arduino untuk membaca nilai konsentrasi gas.
- Koneksi: VCC (5V), GND, A0 ke pin analog Arduino.

c. Buzzer

- Buzzer dihubungkan ke salah satu pin digital pada Arduino.

- Koneksi: Buzzer positif (+) ke pin digital Arduino (PB0), katoda (-) ke GND.

d. Display MAX7219

- Display dihubungkan melalui protokol SPI.
- Koneksi: VCC (5V), GND, DIN (data input) ke pin digital Arduino (MOSI), LOAD ke pin digital Arduino (SS), ISET ke GND melalui resistor, dan CLK (clock) ke pin digital Arduino (SCK).

e. Exhaust Fan

- Exhaust fan dihubungkan ke pin digital pada Arduino melalui relay atau transistor untuk kontrol on/off.
- Koneksi: VCC fan ke sumber daya eksternal, GND fan ke GND, pin kontrol ke pin digital Arduino melalui relay/transistor.

2.2 SOFTWARE DEVELOPMENT

Source Code:

```

;-----;
; Toxic Gases Detector using MQ-2 Sensor ;
;-----;

#define __SFR_OFFSET 0x00
#include "avr/io.h"
;-----

.equ  SCK, 5
.equ  MOSI, 3
.equ  SS, 2

;=====

.global main
;=====

main:

    CBI DDRC, 0          ; Set PC0 as input

    RCALL SPI_MAX7219_init

;-----

agn:

    IN    R21, PINC      ; Read the input from port C

    MOV   R20, R21

```

```

CPI    R20, 0x01          ; Compare the value of PC0 with 1
BREQ   MAX7219_disp_text ; If PC0 is 1, branch to MAX7219_disp_text
RJMP   MAX7219_disp_text2; If PC0 is 0, jump to MAX7219_disp_text2

RJMP   agn                ; Repeat the process
;=====
SPI_MAX7219_init:
LDI    R17, (1<<MOSI)|(1<<SCK)|(1<<SS)
OUT    DDRE, R17          ; Set MOSI, SCK, SS as output

LDI    R17, (1<<SPE)|(1<<MSTR)|(1<<SPR0)
OUT    SPCR, R17          ; Enable SPI as master, fscck=fosc/16

LDI    R17, 0x0A          ; Set segment intensity (0 to 15)
LDI    R18, 8             ; Intensity level = 8
RCALL  send_bytes        ; Send command & data to MAX7219

LDI    R17, 0x0B          ; Set scan limit command
LDI    R18, 0x06          ; 8 digits connected to MAX7219
RCALL  send_bytes        ; Send command & data to MAX7219

LDI    R17, 0x0C          ; Set turn ON/OFF command
LDI    R18, 0x01          ; Turn ON MAX7219
RCALL  send_bytes        ; Send command & data to MAX7219

RET

;=====
MAX7219_disp_text:
RCALL  active

LDI    R17, 0x02
LDI    R18, 0b00111101    ; Data = d
RCALL  send_bytes        ; Send command & data to MAX7219

LDI    R17, 0x03
LDI    R18, 0b01111101    ; Data = a
RCALL  send_bytes        ; Send command & data to MAX7219

LDI    R17, 0x04
LDI    R18, 0b01110110    ; Data = n
RCALL  send_bytes        ; Send command & data to MAX7219

```

```

    LDI    R17, 0x05

    LDI    R18, 0b01111011    ; Data = g
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x06

    LDI    R18, 0b01101111    ; Data = e
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x07

    LDI    R18, 0b01000110    ; Data = r
    RCALL  send_bytes          ; Send command & data to MAX7219

    RCALL  delay_2s

    RCALL  agn
;-----
MAX7219_disp_text2:
    RCALL  inactive

    LDI    R17, 0x02

    LDI    R18, 0b00000001    ; Data = -
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x03

    LDI    R18, 0b01011011    ; Data = S
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x04

    LDI    R18, 0b01110111    ; Data = A
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x05

    LDI    R18, 0b01000111    ; Data = F
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x06

    LDI    R18, 0b01001111    ; Data = E
    RCALL  send_bytes          ; Send command & data to MAX7219

    LDI    R17, 0x07

    LDI    R18, 0b00000001    ; Data = -

```

```

        RCALL send_bytes          ; Send command & data to MAX7219

        RCALL delay_2s

        RCALL agn

;-----

send_bytes:

        CBI    PORTB, SS          ; Enable slave device MAX7219

        OUT    SPDR, R17          ; Transmit command

112:

        IN     R19, SPSR

        SBRS   R19, SPIF          ; Wait for byte transmission to complete

        RJMP   112

        OUT    SPDR, R18          ; Transmit data

113:

        IN     R19, SPSR

        SBRS   R19, SPIF          ; Wait for byte transmission to complete

        RJMP   113

        SBI    PORTB, SS          ; Disable slave device MAX7219

        RET

;=====

active:

        CBI    PORTB, SS          ; Enable slave device

        SBI    DDRB, 1            ; Set PB1 as output for fan

        SBI    DDRB, 0            ; Set PB0 as output for buzzer

        SBI    PORTB, 1          ; Turn on fan

        SBI    PORTB, 0          ; Turn on buzzer

        SBI    PORTB, SS          ; Disable slave device

        RET

;-----

inactive:

        CBI    PORTB, SS          ; Enable slave device

        CBI    DDRB, 1            ; Turn off fan

        CBI    DDRB, 0            ; Turn off buzzer

```

```

        SBI    PORTB, SS        ; Disable slave device

        RET

;=====
delay_2s:                ; Delay 2s

        LDI    R21, 200
16:      LDI    R22, 255
17:      LDI    R23, 164
18:      DEC    R23

        BRNE   18

        DEC    R22

        BRNE   17

        DEC    R21

        BRNE   16

        RET

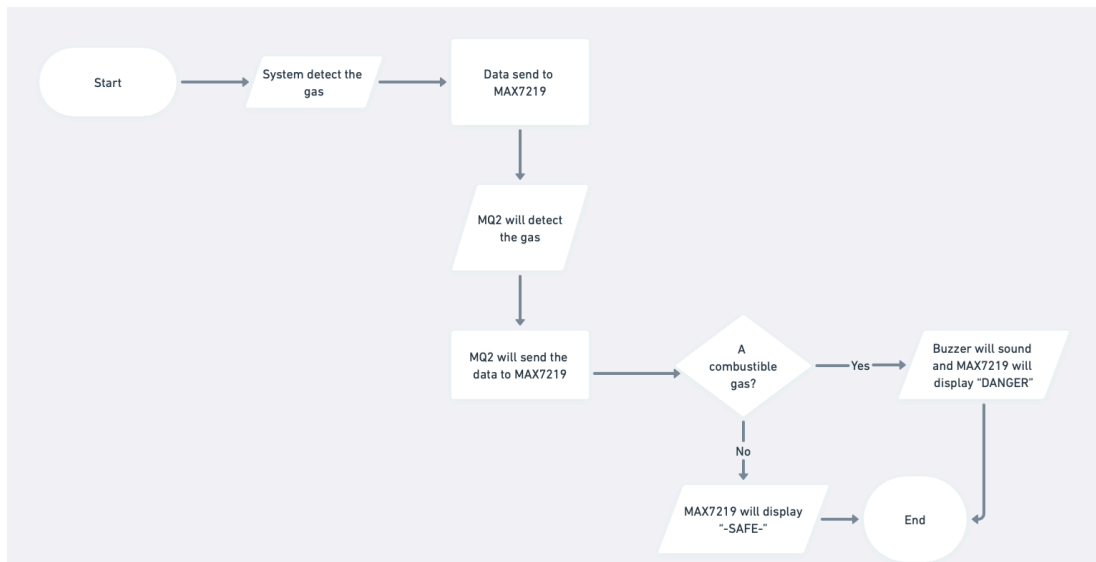
```

Penjelasan setiap subrutin:

- a. **SPI_MAX7219_init**. Subrutin ini digunakan untuk inisialisasi master dalam protokol SPI dan mengatur konfigurasi untuk MAX7219. MAX7219 akan mengatur 7-segment untuk memiliki intensitas level 8 dan akan menggunakan semua bit pada 7-segment (8 bit).
- b. **agn**. Subrutin ini digunakan untuk memeriksa input dari sensor MQ2 secara terus-menerus. Input dari sensor akan dibandingkan dengan nilai 1, jika input bernilai sama, maka program akan branch ke subrutin MAX7219_disp_text, yang berarti sensor mendeteksi gas berbahaya. Sedangkan jika berbeda, maka program akan branch ke MAX7219_disp_text2, yang berarti sensor tidak mendeteksi gas berbahaya.
- c. **MAX7219_disp_text**. Subrutin ini digunakan untuk mengirimkan teks yang akan ditampilkan ke 7-segment. Register R17 menyimpan nilai untuk bit yang akan dikonfigurasi dan R18 digunakan untuk menyimpan data yang akan ditampilkan oleh bit tersebut. Teks yang akan ditampilkan adalah “danger”, menandakan bahwa sensor mendeteksi gas berbahaya. Juga akan dilakukan pemanggilan subrutin active untuk menghidupkan buzzer dan exhaust fan.
- d. **MAX8219_disp_text2**. Fungsi subrutin ini sama seperti MAX7219_disp_text. Namun, teks yang akan ditampilkan adalah “-SAFE-” , menandakan bahwa sensor tidak mendeteksi gas berbahaya. Juga akan dilakukan pemanggilan subrutin inactive untuk mematikan buzzer dan exhaust fan.

- e. **send_bytes**. Subrutin ini akan dipanggil untuk setiap komunikasi antara master dan slave, terutama dalam mengirimkan data untuk ditampilkan ke 7-segment. Perangkat slave akan diaktifkan sebelum transmisi berlangsung dan perangkat akan dinonaktifkan jika transmisi telah selesai.
- f. **active**. Subrutin ini digunakan untuk menghidupkan buzzer dan exhaust fan menggunakan protokol SPI.
- g. **inactive**. Subrutin ini digunakan untuk mematikan buzzer dan exhaust fan menggunakan protokol SPI.
- h. **delay_2s**. Subrutin digunakan untuk memberikan delay 2 detik pada setiap pembacaan input dari sensor.

Flowchart:



Pic 2. Flowchart

2.3 HARDWARE AND SOFTWARE INTEGRATION

Berdasarkan rangkaian yang telah dibuat secara fisik maupun yang telah dibuat pada software Proteus, dapat dilihat bahwa sensor MQ-2 disambungkan dengan Arduino Uno melalui pin analog dengan menghubungkan pin A0 pada sensor dengan pin PC0 pada Arduino Uno, dan juga pin VCC dihubungkan ke VCC breadboard dan pin GND dihubungkan ke GROUND breadboard. Selain itu, terdapat juga modul MAX7219 dihubungkan dengan Arduino Uno. Dimana pin DIN dihubungkan dengan pin PB3, pin LOAD dihubungkan dengan pin PB2, pin CLK dihubungkan dengan pin PB5, serta pin ISET

disambungkan dengan ground. Kami juga menggunakan buzzer yang dihubungkan dengan pin PB0 sebagai output, serta exhaust fan yang terhubung ke pin PB1 sebagai output.

Untuk pengekseskuan kode Assembly yang telah dibuat, kami menggunakan software Arduino IDE dengan mengupload kode ke Arduino Uno yang telah disambungkan ke komputer. Uploading data dapat dilakukan menggunakan Arduino IDE karena pada software ini, tersedia library yang diperlukan oleh Arduino untuk membaca dan memproses kode secara real-time.

Kami mengimplementasikan modul-modul yang telah dipelajari selama mengikuti praktikum Sistem Siber Fisik pada Semester Genap 2023/2024. Berikut ini penjelasan terperinci mengenai setiap implementasi modul yang kami gunakan pada proyek kami:

Implementasi I/O

Pada mikrokontroler yang digunakan dalam proyek ini, terdapat tiga port digital yang dapat diakses untuk input/output, yaitu Port B, C, dan D. Setiap port memiliki 8 bit yang mewakili 8 pin digital yang dapat dikendalikan atau dibaca. Port B memiliki pin 8 hingga 15, Port C memiliki pin 0 hingga 7, dan Port D memiliki pin 0 hingga 7.

Port B digunakan untuk mengendalikan perangkat elektronik seperti buzzer dan fan. Pin 0 dan 1 dari Port B digunakan untuk mengontrol buzzer dan fan secara langsung. Port tersebut diatur dengan menggunakan instruksi SBI dan CBI untuk menetapkan pin sebagai output atau input. Port C digunakan sebagai input dengan mengatur PC0 sebagai input untuk membaca sensor. Pengaturan mode input dilakukan dengan instruksi CBI pada register DDRC. Port B juga digunakan untuk komunikasi SPI dengan perangkat MAX7219. Pin 2, 3, dan 5 dari Port B digunakan untuk SPI (Serial Peripheral Interface) untuk berkomunikasi dengan perangkat tersebut. Instruksi OUT digunakan untuk mengatur register DDRB untuk menetapkan pin-pin tersebut sebagai output.

Aritmatika

Pada proyek ini, kami mengimplementasikan beberapa instruksi aritmatika di dalam kode untuk dapat menjalankan rangkaian. Kami menggunakan modul aritmatika ini untuk melakukan perbandingan antara nilai yang dibaca oleh sensor MQ-2 dengan nilai HIGH atau 1 sebagai indikator bahwa adanya gas berbahaya yang terdeteksi oleh sensor dengan menggunakan instruksi CPI dan BREQ, serta dalam kode delay. Untuk memberikan delay

pada program, digunakan instruksi DEC atau decrement untuk mengurangi nilai pada register dan terus melakukan pengulangan hingga nilai pada register mencapai 0.

Communication

Pada proyek ini, kami menggunakan protokol komunikasi SPI untuk menghubungkan sensor MQ-2 dengan Arduino, dan hasilnya akan ditampilkan pada MAX7219. Dengan menggunakan SPI, data dapat dikirim dan diterima dengan cepat antara mikrokontroler dan perangkat keras eksternal, memungkinkan kontrol dan akses data yang efisien. MAX7219 akan menampilkan teks "Safe" yang menandakan tidak ada gas, dan "Danger" yang menunjukkan adanya gas yang terdeteksi oleh sensor MQ-2.

Sensor

Untuk implementasi modul ini, digunakan sensor gas MQ-2. Sensor ini digunakan untuk mendeteksi keberadaan gas berbahaya dalam ruangan, seperti LPG, alkohol, propana, hidrogen, metana, dan karbon monoksida. Sensor terhubung dengan Arduino dan menghasilkan output berupa 1 (jika mendeteksi gas berbahaya) dan 0 (jika tidak mendeteksi gas berbahaya).

CHAPTER 3

TESTING AND EVALUATION

3.1 TESTING

Setelah fase desain dan implementasi telah berhasil dilakukan, developing proyek dilanjutkan dengan fase testing. Pada fase ini, rangkaian dan kode yang telah dibuat akan diuji dalam kondisi normal dan kondisi dimana terdapat gas berbahaya pada udara untuk menguji apakah sensor gas MQ-2, buzzer, exhaust fan, serta modul MAX7219 dapat bekerja dengan baik.

Proyek dikatakan berhasil ketika diberikan kondisi dimana terdapat gas berbahaya di sekitar sensor gas MQ-2 dan buzzer dapat menyala sebagai indikator, serta MAX7219 menampilkan kata “Danger”, dan exhaust fan aktif untuk meminimalisir gas berbahaya yang ada di udara.

3.2 RESULT

Berdasarkan percobaan yang telah dilakukan, diperoleh hasil sebagai berikut. Melalui percobaan dengan kondisi udara yang aman untuk mengetahui apakah sensor MQ-2 telah berhasil mengambil data dari udara sekitar dan mengirimkannya ke Arduino, maka dapat dilihat pada hasil berikut.

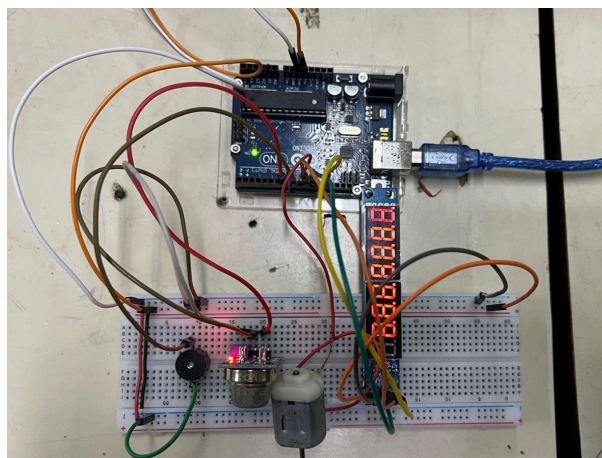


Fig 2. Hasil rangkaian pada keadaan udara normal

Kemudian, akan dilakukan percobaan kembali dengan kondisi dimana terdapat gas berbahaya di udara untuk mengetahui apakah sensor MQ-2 dapat mendeteksi gas dan mengirimkannya data ke MAX7219 dan apakah buzzer dan exhaust fan akan menyala pada kondisi tersebut.

3.3 EVALUATION

Proyek Toxic Gases Detector bertujuan untuk mendeteksi keberadaan gas berbahaya dalam suatu ruangan secara efektif. Sistem ini menggunakan sensor MQ-2 yang membaca data mengenai keberadaan gas, kemudian mengirimkannya ke Arduino untuk diproses. Ketika gas berbahaya terdeteksi, sistem akan mengaktifkan beberapa respon: buzzer akan berbunyi sebagai alarm, tampilan 7 segment akan menampilkan teks “danger” sebagai peringatan visual, dan motor DC akan mengaktifkan kipas untuk membantu mengurangi konsentrasi gas. Dengan demikian, sistem ini memungkinkan pengguna untuk mengetahui keberadaan gas berbahaya lebih awal, sehingga risiko yang ditimbulkan dapat diminimalisir atau dihindari.

CHAPTER 4

CONCLUSION

Toxic Gases Detector merupakan sistem yang dirancang sebagai pendeteksi konsentrasi gas berbahaya dalam ruangan. Di dalam rangkaian ini terdapat sensor gas MQ-2 yang berfungsi penting untuk menjadi pendeteksi kandungan gas berbahaya, seperti LPG, alkohol, asap, propana, hidrogen, metana, maupun karbon monoksida di udara. Sensor ini mengambil data secara langsung dari udara dan mengirimkan data pembacaan tersebut ke Arduino Uno. Berdasarkan data yang telah diterima ini, Arduino akan melakukan analisis data, dimana apabila hasil yang didapat menunjukkan bahwa terdapat gas berbahaya yang terdeteksi, maka buzzer akan menyala sebagai indikator dan exhaust fan akan aktif untuk meminimalisir konsentrasi gas berbahaya di udara sekitar.

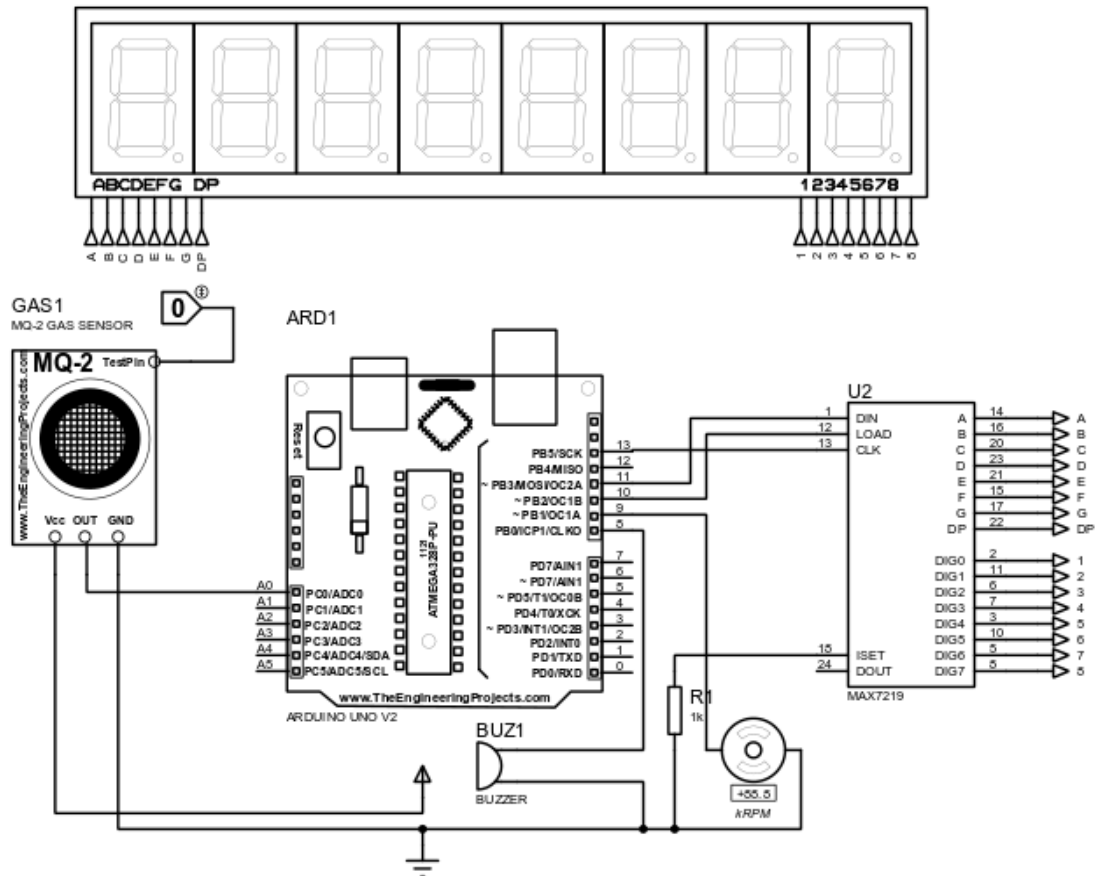
Proyek ini mengimplementasikan beberapa modul yang telah dipelajari selama praktikum Sistem Siber Fisik, antara lain yaitu modul 2, 3, 4, 5, 6, 8, dan 9. Dengan ini kami harapkan proyek Toxic Gases Detector dapat membantu masyarakat dalam tindak pencegahan terjadinya keracunan gas, maupun bencana kebakaran. Kami harap proyek kami dapat dipergunakan dengan baik, kemudian juga kami berharap bahwa Toxic Gases Detector dapat dikembangkan menjadi rangkaian yang lebih baik lagi.

REFERENCES

- [1] Digital Laboratory UI. (2024). “Modul 2 - Modul 9 Praktikum Sistem Siber Fisik.” EMAS2. [Online]. Available: <https://emas2.ui.ac.id/course/view.php?id=64823>. [Accessed: May. 9, 2024].
- [2] Y. D. Chandra, Kosdiana, M. Riastuti. (Jul. 30, 2021). “Gas Leak Detector Prototype Using MQ2 Sensor and Arduino Uno-Based Air Suction Tool.” ICONEST. [Online]. Available: <http://prosiding.unirow.ac.id/index.php/iconest/article/view/1099/614>. [Accessed: May. 9, 2024].
- [3] Robokart. (Apr. 12, 2019). “How To Use Gas Sensor MQ-2 With Arduino.” Youtube. [Online]. Available: https://youtu.be/g-NvPPEj3oQ?si=A_bviXNu_yFtqujc. [Accessed: May. 11, 2024].
- [4] Atmel. (n.d.). “ATmega328P.” Microchip Technology. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf. [Accessed: May. 16, 2024].
- [5] Hanwei Electronics. (n.d.). “Technical Data MQ-2 Gas Sensor.” Mouser Electronics. [Online]. Available: <https://www.mouser.com/datasheet/2/321/605-00008-MQ-2-Datasheet-370464.pdf>. [Accessed: May. 16, 2024].
- [6] Mr. Sottong. “Arduino MAX7219 7-Segment Display Tutorial,” instructables.com. [Online]. Available: <https://www.instructables.com/MAX7219-7-Segment-Using-Arduino/>.

APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation

