

Mechatronics System Design Journal.

A Technician's Journal is a short and concise summary of the team's journey from the initial task analysis through the final design solution.

The documentation should include enough detail for another person to look at your notebook and be able to build your system, or to at least follow the steps your team took to get to your final design solution.

[LINE FOLLOWER AND OBSTACLE AVOIDING]

Team Information

Abstract

Robot line tracer merupakan jenis robot yang dirancang khusus untuk mengikuti garis yang telah ditentukan sebelumnya. Robot yang dilengkapi dengan sensor yang mampu mendeteksi garis, seperti sensor cahaya atau sensor infra-merah, yang digunakan untuk mengontrol gerakan roda agar tetap berada di atas garis tersebut. Melalui pemahaman dan implementasi robot line tracer, para pembuat dapat memperoleh wawasan mendalam tentang penggunaan sensor dalam mengendalikan gerakan robot dan memecahkan masalah navigasi sederhana, seperti mengikuti jalur yang telah ditentukan sebelumnya.

Ramadhan Tegar Imansyah

[ramadhantegar.i@gmail.com]

Muhammad Irfan Habib

[muh.irfanh473@gmail.com]



Table of Contents

1 Introduction and Initial Analysis	3
1.1 Project Context	3
1.2 Initial Thought Process.....	3
2 Requirement Analysis and Specification.....	4
2.1 User Requirements	4
2.2 System Requirements	4
2.3 Tools and Technologies.....	4
2.4 Target specification.....	6
3 Conceptual Design	6
3.1 System Architecture.....	6
3.2 Interface Design	7
3.3 Control Algorithm Design.....	10
4 Detailed Design and Development	11
4.1 Component Design.....	11
4.2 Coding and Implementation	13
4.3 Integration	17
4.4 Unique Features.....	18
5 Testing, Evaluation, and Optimization	18
5.1 Testing Strategy	18
5.2 Performance Evaluation.....	18
5.3 Optimization	18
6 Collaboration and Project Management.....	19
6.1 Teamwork Dynamics	19
6.2 Project Management	19
7 Conclusion and Reflection.....	19
7.1 Project Summary.....	19
7.2 Future Work	19
7.3 Personal and Group Reflections.....	20
8 Appendices.....	21
8.1 Bill of Materials	21
8.2 Electrical Wiring and System Layout.....	22



8.3 Code Repository	24
8.4 Additional Documentation.....	24
9 References	24



1 Introduction and Initial Analysis

1.1 Project Context

Sistem mekatronika telah menjadi bagian dari berbagai aplikasi otomasi dan kendali dalam berbagai industri. Dalam konteks pengembangan robot, peningkatan kompleksitas dan fungsi dari sistem mekatronika mengharuskan integrasi antarmuka pengguna yang intuitif dan efektif. Salah satu tugas penting dalam pengembangan robot adalah mengatur dan memantau pergerakan dengan akurasi tinggi, seperti pada robot line follower dan obstacle avoiding.

Robot line follower bertujuan untuk mengikuti lintasan atau jalur tertentu dengan memanfaatkan sensor dan kendali yang terintegrasi. Sementara itu, robot obstacle avoiding dirancang untuk menghindari rintangan dan mengubah jalur secara otomatis untuk mencegah tabrakan.

Pengembangan tampilan GUI (Graphical User Interface) memberikan kemampuan untuk memvisualisasikan dan mengontrol robot dengan cara yang lebih intuitif dan efisien bagi pengguna akhir. Dengan GUI yang tepat, pengguna dapat memonitor status robot, mengubah parameter operasi, dan memantau hasil sensor dengan mudah.

Dalam proyek bertujuan untuk mengintegrasikan sistem mekatronika dengan antarmuka pengguna berbasis GUI untuk meningkatkan fungsionalitas, keterbacaan, dan kontrol dari robot line follower dan obstacle avoiding. Dengan demikian, proyek ini diharapkan dapat membantu memajukan penggunaan sistem mekatronika dalam berbagai aplikasi praktis. dengan menggunakan sistem yang berbasis GUI ini, pengguna tidak perlu lagi merubah hardware saat melakukan pengembangan, pengguna hanya perlu melakukan upgrade pada software GUI.

1.2 Initial Thought Process

Pada proses awal kami melakukan pencarian data terkait fungsi kerja tiap komponen pada Multi-Line follower dengan melakukan pengecekan komponen apa saja yang ada pada robot tersebut, kemudian melakukan peninjauan terkait dengan skematik robot yang bertujuan untuk mengetahui pengkabelan sistem robot. Membuat dan melakukan trial dan error untuk program kerja robot sesuai dengan fitur yang diinginkan. Merancang tampilan GUI sesuai dengan fitur yang diinginkan dimana bertujuan untuk indikator tampilan.

- Robot dilengkapi dengan sensor ultrasonic sebagai pendeteksi jarak antara robot dan lingkungan sekitar robot sehingga dapat menghindari obstacle
- Robot juga dilengkapi dengan sensor IMU untuk navigasi pergerakan dari robot
- Pada GUI ditampilkan berbagai fitur seperti connect, monitoring, drive, dan kalibrasi



2 Requirement Analysis and Specification

2.1 User Requirements

Graphical User Interface (GUI) atau Antarmuka Pengguna Grafis merujuk pada suatu bentuk antarmuka pengguna yang memungkinkan interaksi antara pengguna dan perangkat lunak atau sistem menggunakan elemen-elemen grafis seperti ikon, tombol, jendela, dan menu. GUI dirancang untuk membuat pengguna dapat berinteraksi dengan perangkat lunak atau sistem dengan lebih mudah dan intuitif, tanpa perlu mengingat perintah-perintah teks atau sintaks khusus. GUI menggunakan metode interaksi pada peranti elektronik secara grafis (bukan perintah teks) antara pengguna dan komputer.

Sistem kontrol merujuk pada proses yang digunakan untuk mengelola, mengatur, dan mengawasi berbagai jenis aktivitas dalam berbagai konteks. Sistem kontrol multi line follower merujuk pada teknologi yang digunakan dalam robotika untuk mengontrol gerakan sebuah robot agar dapat mengikuti atau mengikuti lintasan multi-garis yang telah ditentukan sebelumnya. Sistem ini umumnya menggunakan sensor-sensor yang terpasang pada robot untuk mendeteksi lintasan atau jalur yang harus diikuti. Sensor-sensor ini dapat berupa sensor inframerah, sensor ultrasonik, sensor garis, atau sensor lainnya yang dapat mendeteksi perubahan warna atau pola tertentu di lintasan.

Sistem kontrol multi line follower dengan GUI (Graphical User Interface) dapat memberikan antarmuka yang lebih intuitif dan mudah digunakan untuk mengontrol robot yang mengikuti lintasan multi-garis. GUI memungkinkan untuk berinteraksi dengan robot, memberikan perintah, dan memantau pergerakan robot secara visual melalui antarmuka grafis.

2.2 System Requirements

Dalam pembuatan sistem multi line tracer dengan GUI melibatkan berbagai komponen yaitu :

1. Sensor garis dapat mendeteksi garis yang terdapat pada lantai. Data tersebut akan tersimpan pada array data garis.
2. Sensor ultrasonik dapat mendeteksi obstacle dan mengetahui jarak yang terdapat antara robot dengan lingkungan sekitar. Data sensor ultrasonik akan tersimpan pada array data ultrasonic.
3. Robot dapat bergerak sesuai dengan garis yang terdapat pada lantai. Pergerakan robot akan berhenti jika terdapat obstacle di depan robot.
4. Pada GUI dapat digunakan sebagai indicator penampil sehingga dapat berinteraksi dengan robot untuk menampilkan dan memonitor state robot sesuai dengan perintah yang terdeteksi.
5. Motor digunakan sebagai penggerak roda pada line tracer berdasarkan informasi atau data yang diinformasikan oleh mikrokontroler.



6. Mikrokontroler digunakan sebagai pengontrol sistem pada multi line tracer, mengambil data dari sensor garis, mengontrol motor dan memproses informasi.

2.3 Tools and Technologies

Software yang digunakan pada project Pemrograman Sistem Mekatronik yaitu *Visual Studio Code*(Platform IO)/Arduino IDE dan pyqt.

Adapun komponen yang digunakan pada Multi-Line Follower yaitu :

Tabel 1. Daftar komponen

No	Komponen
1	Arduino Nano
2	Sensor Ultrasonik HC-SR04
3	Battery
4	Driver Motor TB6612FNG
5	Buck converter
6	ESP 01
7	Line sensor
8	Motor DC
9	MPU 6050

Berikut adalah library yang digunakan pada Multi-Line Follower dengan GUI yaitu :

Tabel 2. Daftar library

No	Tool Atau Library	Keterangan
1	#include <Wire.h>	Digunakan pada Arduino untuk komunikasi antar perangkat menggunakan I2C seperti mengirim dan menerima data
2	#include <NewPing.h>	Untuk memasukkan pengukuran jarak dari sensor ultrasonik
3	#include <LiquidCrystal_I2C.h>	Untuk menampilkan data pada layer LCD



2.4 Target specification

Tabel 3. Target specification

Feature	Description	Measurement Metric	Target Value
Tampilan GUI	Robot dapat dimonitoring dan di kontrol melalui GUI.	Mengukur seberapa akurat antara tampilan dari GUI dan robot.	Tampilan GUI sesuai dengan robot
Obstacle avoidance	Robot dapat berhenti atau berubah arah jika ada rintangan atau hambatan	Mengukur seberapa akurat sensor ultrasonik dalam mengukur jarak antara robot dan objek di sekitarnya. Memastikan bahwa robot dapat merespons dengan benar terhadap perubahan kondisi lintasan atau keberadaan rintangan.	Robot berjalan maju kemudian berhenti atau berubah arah jika ada objek yang menghalangi.
Line sensor	Robot dapat mengikuti sesuai arah line yang telah dibuat	Mengukur seberapa akurat sensor dalam mendeteksi line sehingga robot dapat terus mengikuti line tanpa keluar dari line yang telah dibuat.	Robot berjalan maju sesuai jalur yang telah dibuat

3 Conceptual Design

3.1 System Architecture

Berikut adalah penjelasan komponen-komponen utama dalam ilustrasi level tinggi ini:

- **Graphical User Interface (GUI)**

Pada bagian ini, merupakan bagian yang menghubungkan antara pengguna dan sistem, dimana GUI dikembangkan dengan desain dari QT Designer dan memungkinkan pengguna untuk memberikan input dan menerima data dari sistem. Dalam projek ini, GUI didesain untuk dapat memantau nilai ketiga sensor ultrasonik, data sensor garis, dan data dari IMU, serta dapat memberikan input nilai PID pada sistem dan input jarak stop dari sensor ultrasonik.

- **Logika Kontrol**

Logika kontrol dari sistem ini adalah robot bisa diimplementasikan sekitar 30% untuk sistem deteksi atau menghindari dari halangan yang ada disekitarnya.

- **Komunikasi antar komponen**

Komunikasi antara ESP 01 dan GUI dapat dilakukan melalui protokol komunikasi yang sesuai dengan kebutuhan aplikasi, seperti HTTP/HTTPS, MQTT, WebSocket, RESTful API, dan Serial Communication

- **Hardware**

Pada bagian ini, perangkat keras yang digunakan yaitu arduino nano, ESP 01, Sensor ultrasonik HC-SR04, line sensor, IMU, serta motor dan driver.



- koneksi







Koneksi jaringan yang digunakan adalah menggunakan koneksi wireless berupa wifi antara GUI dan ESP 01

3.2 Interface Design

Tabel 4. Daftar tampilan GUI

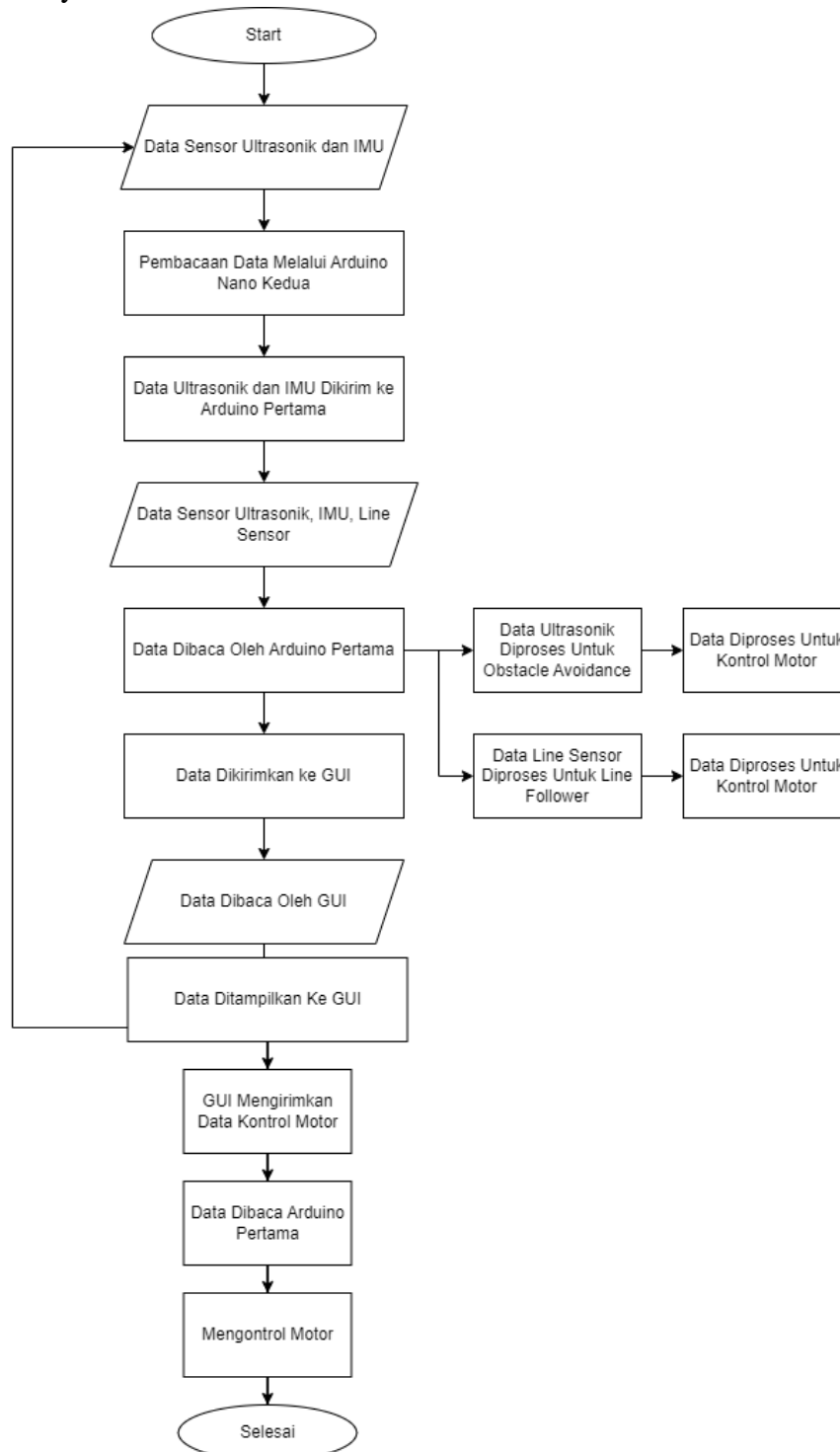
No	Tampilan GUI	Keterangan
1		Halaman pertama yang akan tertampil adalah menu untuk login menuju halaman utama dari GUI. Dimana pada halaman ini user diperintahkan untuk memasukkan username dan password untuk mengakses halaman selanjutnya (NB: username: user password: 123)
2		Halaman berikut yaitu adalah halaman loading menuju halaman utama
3		Halaman berikut adalah halaman utama Dimana tersedia beberapa menu utama dimana didalamnya terdapat connect page, monitoring page, drive page, mode page, dan calibrate page. Selain itu juga terdapat notification page, user page, rating page, about us page, dan helpdesk page.
4		Halaman berikut adalah halaman calibrate page Dimana pada menu ini robot melakukan kalibrasi dengan berputar selama beberapa detik.

5		<p>Halaman berikutnya adalah menu connect Dimana ini adalah menu untuk menghubungkan antara robot dan GUI</p>
6		<p>Halaman berikut adalah halaman yang berisi menu monitoring Dimana didalam menu tersebut terdapat fitur untuk monitoring line sensor, ultrasonic sensor, dan imu sensor.</p>
7		<p>Halaman berikut adalah menu drive Dimana nantinya menu ini berfungsi untuk mengendalikan robot sesuai keinginan user.</p>
8		<p>Halaman berikut merupakan tampilan dari menu mode Dimana ada 2 pilihan yaitu line tracer dan obstacle avoidance.</p>

		
9		Halaman berikut adalah tampilan dari profile page Dimana halaman ini berisi profil dari user dan beberapa informasi tentang user.
10		Halaman berikut adalah tampilan untuk rating page Dimana halaman ini berfungsi sebagai penilaian user terhadap GUI yang dibuat baik dari tampilan atau fiturnya.
11		Halaman berikut adalah halaman notification page Dimana halaman ini berfungsi untuk menampilkan notifikasi untuk user.
12		Berikut adalah halaman about us page Dimana halaman berisi informasi tentang GUI.
13		Berikut adalah halaman helpdesk page Dimana halaman ini berisi bantuan untuk user jika mengalami kendala pada saat menjalankan GUI.

3.3 Control Algorithm Design

Alur pemrosesan data dilakukan melalui komunikasi antara GUI dan Hardware. Dalam hal ini pada GUI dapat memberikan nilai input dari PID line sensor dan juga input penentuan jarak untuk stop. Selain itu pada hardware akan mengirim data dimana data tersebut dapat dipantau melalui GUI yaitu data nilai sensor ultrasonic, nilai sensor garis, dan juga nilai dari IMU berupa pitch, roll, dan yaw.




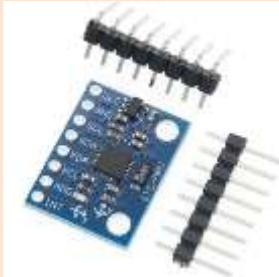




4 Detailed Design and Development

4.1 Component Design

Tabel 5. Desain Komponen

components	Module	Fungsional	Components picture
Arduino Nano	Sensor Infrared (IR) Array Motor dan roda Motor driver Baterai atau sumber daya eksternal	Membaca data dari sensor infrared untuk memberikan informasi posisi robot terhadap garis. Mengontrol motor menggunakan motor driver. Mengimplementasikan algoritma sederhana untuk mengontrol pergerakan robot berdasarkan pembacaan sensor. Menerapkan loop kontrol untuk mengoreksi pergerakan robot.	
Sensor Ultrasonik HC-SR04	Transmitter (pengirim). Receiver (penerima). Trig (trigger) pin. Echo pin. VCC dan GND.	Mengukur jarak dan objek dengan menggunakan gelombang ultrasonik. Mengukur jarak tanpa harus bersentuhan dengan objek yang hendak diukur. Mendeteksi objek atau dinding di sekitar robot, menghindari tabrakan, dan menjaga jarak dengan objek tertentu.	
Driver TB6612FNG	Kanal motor Input tegangan Pin kontrol Proteksi termal Mode sleep Indikator LED	Memungkinkan pengendalian perputaran arah motor DC secara independen Mengendalikan motor stepper dua fase Perlindungan termal yang rusak akibat suhu tinggi Mengkonfigurasi pin sesuai kebutuhan project	
MPU 6050	Akselerometer Gyroscope DMP (Digital Motor Processor) Tegangan operasional Komunikasi I2C (Inter-Integrated Circuit) Pin VCC dan GND Pin ADO (Alamat I2C pilihan)	Modul sensor inertial yang menyatukan accelerometer dan gyroscope dalam satu chip. Dengan demikian, MPU6050 dapat memberikan informasi tentang percepatan dan kecepatan sudut suatu objek dalam tiga dimensi (sumbu X, Y, dan Z).	

Buck Converter XL-4005	Step-Down (Buck) Converter Step-Up (Boost) Converter Step-Up/Step-Down (Buck-Boost) Converter Inverter Converter Adjustable Converter Isolated DC to DC Converter	Mengubah tegangan listrik dari satu tingkat ke tingkat yang berbeda.	
ESP 01	Mikrokontroler Antarmuka WiFi RAM (Random Access Memory) Flash Memory GPIO (General Purpose Input/Output) UART (Universal Asynchronous Receiver-Transmitter)	Modul WiFi berbasis ESP8266 yang populer digunakan untuk proyek IoT (Internet of Things). ESP-01 dapat digunakan untuk terhubung ke jaringan WiFi, memungkinkan perangkat untuk berkomunikasi secara nirkabel dengan server atau perangkat lain.	
Motor DC	N20 DC Motor bertegangan operasional umumnya berkisar antara 3V hingga 12V.	Perangkat elektronik yang dirancang untuk mengendalikan motor DC (Direct Current). Motor DC adalah jenis motor listrik yang beroperasi dengan arus searah. Modul motor DC menyediakan berbagai fitur dan kontrol untuk mengatur daya dan arah putaran motor.	
Line Sensor	Array sensor Inframerah atau fotodioda Emisor cahaya inframerah Tuning empfindlichkeit (sensitivitas penyesuaian) Digital atau analog output	Mendeteksi jalur atau garis Mendeteksi tingkat perbedaan warna guna menentukan garis Memberikan kontrol yang lebih presisi Sistem dapat mengukur kecepatan dan perubahan posisi robot selama bergerak.	
Battery	Mencakup fitur-fitur seperti perlindungan baterai (overcharge, over-discharge, over-current), koneksi untuk	Menyimpan tegangan listrik yang digunakan untuk menggerakkan perangkat elektronik. Melindungi termal dari kerusakan	



	pengisian, dan mungkin indikator tingkat daya.	Mencegah polusi lingkungan karena kandungan kimia yang ada di dalamnya
--	--	--



4.2 Coding and Implementation

Berikut ini adalah tautan program GUI dan Line Tracer:

https://github.com/tegarRTI24/LineTracerKelompok1MKB_45_53

Penjelasan program line tracer :

1. Program line tracer dikerjakan pada dua arduino yang mana arduino pertama untuk program pembacaan line sensor dan pemutaran motor pada robot. Dimulai dari melakukan definisi variabel

```
int AIN1 = 4; //kanan
int AIN2 = 2;
int BIN1 = 7; //kiri
int BIN2 = 0;
int STBY = 9;
int PWMA = 3; //kanan
int PWMB = 5; //kiri
float skiri = 163;
float skanan = 158.5;
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);

int s11 = A0;
int s12 = A1;
int s13 = A2;
int s14 = A3;
int s15 = A6;

int baca1,baca2,baca3,baca4,baca5;
```

2. Melakukan pendefinisian setup program dengan mendefinisikan setiap pin yang dipakai pada arduino pertama

```
void setup() {
  Serial.begin(9600);
  //kirim serial.begin(9600);
  pinMode(s11, INPUT);
  pinMode(s12, INPUT);
  pinMode(s13, INPUT);
  pinMode(s14, INPUT);
  pinMode(s15, INPUT);
  pinMode(AIN1, OUTPUT);
  pinMode(AIN2, OUTPUT);
  pinMode(BIN1, OUTPUT);
  pinMode(BIN2, OUTPUT);
  pinMode(STBY, OUTPUT);
  pinMode(PWMA, OUTPUT);
  pinMode(PWMB, OUTPUT);
  lcd.init();
  lcd.backlight();
}
```

3. Melakukan pembacaan data dari arduino kedua untuk pembacaan sensor dengan memprogramnya di void loop, serta mengirim data ke PyQt melalui serial print



```
void loop() {
  if (Serial.available() > 0) {
    String data = Serial.readStringUntil('\n');
    int index1 = data.indexOf(',');
    int index2 = data.indexOf(',', index1 + 1);
    int index3 = data.indexOf(',', index2 + 1);
    int index4 = data.indexOf(',', index3 + 1);
    int index5 = data.indexOf(',', index4 + 1);
    int index6 = data.indexOf(',', index5 + 1);
    int distance1 = data.substring(0, index1).toInt();
    int distance2 = data.substring(index1 + 1, index2).toInt();
    int distance3 = data.substring(index2 + 1, index3).toInt();
    float distance4 = data.substring(index3 + 1, index4).toInt();
    float distance5 = data.substring(index4 + 1, index5).toInt();
    float distance6 = data.substring(index5 + 1, index6).toInt();
    Serial.print(distance1);
    Serial.print(",");
    Serial.print(distance2);
    Serial.print(",");
    Serial.print(distance3);
    Serial.print(",");

    // Check the conditions and control the motor
    if (distance1 >= 20) {
      maju();
    } else if (distance2 <= 5 && distance1 <= 5) {
      kanan();
    } else if (distance3 <= 5 && distance1 <= 5) {
      kiri();
    } else if (distance1 <= 5) {
      stop();
    }
    bacaline();
    Serial.print(baca1);
    Serial.print(",");
    Serial.print(baca2);
    Serial.print(",");
    Serial.print(baca3);
    Serial.print(",");
    Serial.print(baca4);
    Serial.print(",");
    Serial.print(baca5);
    Serial.print(",");
    Serial.print(distance4);
    Serial.print(",");
    Serial.print(distance5);
    Serial.print(",");
    Serial.println(distance6);
    lcd.setCursor(0,0);
    lcd.print(baca1);
    lcd.print(" ");
    lcd.print(baca2);
    lcd.print(" ");
    lcd.print(baca3);
    lcd.print(" ");
    lcd.setCursor(0,1);
    lcd.print(baca4);
    lcd.print(" ");
    lcd.print(baca5);
    lcd.print(" ");
  }
}
```

4. Lalu pada arduino dua akan dilakukan pembacaan sensor oleh arduino dua dan akan dilakukan proses pengiriman data sebagai paramater kontrol motor di arduino pertama dengan program sebagai berikut



```

void loop() {
  if (Serial.available() > 0) {
    String data = Serial.readStringUntil('\n');
    int index1 = data.indexOf(',');
    int index2 = data.indexOf(',', index1 + 1);
    int index3 = data.indexOf(',', index2 + 1);
    int index4 = data.indexOf(',', index3 + 1);
    int index5 = data.indexOf(',', index4 + 1);
    int index6 = data.indexOf(',', index5 + 1);
    int distance1 = data.substring(0, index1).toInt();
    int distance2 = data.substring(index1 + 1, index2).toInt();
    int distance3 = data.substring(index2 + 1, index3).toInt();
    float distance4 = data.substring(index3 + 1, index4).toInt();
    float distance5 = data.substring(index4 + 1, index5).toInt();
    float distance6 = data.substring(index5 + 1, index6).toInt();
    Serial.print(distance1);
    Serial.print(",");
    Serial.print(distance2);
    Serial.print(",");
    Serial.print(distance3);
    Serial.print(",");

    // Check the conditions and control the motor
    if (distance1 >= 20) {
      maju();
    } else if (distance2 <= 5 && distance1 <= 5) {
      kanan();
    } else if (distance3 <= 5 && distance1 <= 5) {
      kiri();
    } else if (distance1 <= 5) {
      stop();
    }
    bacaline();
    Serial.print(baca1);
    Serial.print(",");
    Serial.print(baca2);
    Serial.print(",");
    Serial.print(baca3);
    Serial.print(",");
    Serial.print(baca4);
    Serial.print(",");
    Serial.print(baca5);
    Serial.print(",");
    Serial.print(distance4);
    Serial.print(",");
    Serial.print(distance5);
    Serial.print(",");
    Serial.println(distance6);
    lcd.setCursor(0,0);
    lcd.print(baca1);
    lcd.print(" ");
    lcd.print(baca2);
    lcd.print(" ");
    lcd.print(baca3);
    lcd.print(" ");
    lcd.setCursor(0,1);
    lcd.print(baca4);
    lcd.print(" ");
    lcd.print(baca5);
    lcd.print(" ");
  }
}

```

5. Semua data yang dihasilkan oleh program arduino dua akan diteruskan dan dibaca oleh arduino pertama. Kemudian dari arduino pertama data akan ditransmisikan ke PyQt sebagai aplikasi GUI melalui serial dengan program sebagai berikut



```
def update_labels(self):
    # Read a line from the serial port
    data = self.serial.readline().decode('utf-8').strip()

    distances = data.split(',')
    if len(distances) == 11:

        # Update the labels with the received distances
        self.cmultra1.setText(distances[0])
        self.cmultra_2.setText(distances[1])
        self.cmultra_3.setText(distances[2])
        self.rollku.setText(distances[8])
        self.pitchku.setText(distances[9])
        self.yawku.setText(distances[10])
        di1=int(distances[3])
        di2=int(distances[4])
        di3=int(distances[5])
        di4=int(distances[6])
        di5=int(distances[7])
        if di1 <= 750:
            icon=QIcon(":/image/Image/Group_32-removebg-preview.png")
            self.linesen1.setIcon(icon)
        if di1 >= 750:
            icon1=QIcon(":/image/Image/Group_31-removebg-preview.png")
            self.linesen1.setIcon(icon1)
        if di2 <= 750:
            icon2=QIcon(":/image/Image/Group 26.png")
            self.linesen2.setIcon(icon2)
        if di2 >= 750:
            icon3=QIcon(":/image/Image/Group 21.png")
            self.linesen2.setIcon(icon3)
        if di3 <= 750:
            icon4=QIcon(":/image/Image/Group 28.png")
            self.linesen3.setIcon(icon4)
        if di3 >= 750:
            icon5=QIcon(":/image/Image/Group 22.png")
            self.linesen3.setIcon(icon5)
        if di4 <= 750:
            icon6=QIcon(":/image/Image/Group 29.png")
            self.linesen4.setIcon(icon6)
        if di4 >= 750:
            icon7=QIcon(":/image/Image/Group 23.png")
            self.linesen4.setIcon(icon7)
        if di5 <= 750:
            icon8=QIcon(":/image/Image/Group_30-removebg-preview.png")
            self.linesen5.setIcon(icon8)
        if di5 >= 750:
            icon9=QIcon(":/image/Image/Group_33-removebg-preview.png")
            self.linesen5.setIcon(icon9)
```

6. PyQT GUI akan mengirimkan perintah juga pada arduino untuk menggerakkan motor melalui program sebagai berikut



```
def sendmaju(self):
    self.serial.write(str.encode("F"))
    print('maju')
def sendmundur(self):
    self.serial.write(str.encode("B"))
    print('mundur')
def sendkanan(self):
    self.serial.write(str.encode("K"))
    print('kanan')
def sendkiri(self):
    self.serial.write(str.encode("J"))
    print('kiri')
def obstacleku(self):
    self.serial.write(str.encode("O"))
    print('obstacle')
def linefolllku(self):
    self.serial.write(str.encode("L"))
    print('line')
def calibrateku(self):
    self.serial.write(str.encode("C"))
    print('calibrate')
def stopku(self):
    print('stop')
    self.serial.write(str.encode("S"))
```

7. Semua data yang terkoneksi akan dilakukan pemrosesan data untuk menggerakkan motor berdasarkan perintah yang diinginkan sebagai contoh untuk melakukan obstacle avoidance berdasarkan pembacaan data dari sensor di arduino kedua dan akan diteruskan kepada arduino pertama untuk menggerakkan motor sesuai jarak yang dibaca oleh arduino kedua

4.3 Integration

Integrasi dari GUI dan sistem kontrol pada multi line tracer merupakan aspek penting dalam mengoperasikan dan mengontrol robot tersebut dengan cara yang lebih intuitif dan efisien. Integrasi ini memungkinkan GUI untuk menyajikan informasi dari sensor secara visual dan memberikan pemahaman yang lebih baik terhadap kondisi sistem. Hal tersebut bertujuan untuk memudahkan penggunaan, pemantauan, dan pengaturan robot secara efisien. Adapun integrasi GUI dan sistem Kontrol yang terdapat pada multi line tracer ini ialah :

1. Apabila sensor ultrasonik mendeteksi ada hambatan didepan maka data yang dibaca oleh sensor ultrasonik akan dikirimkan ke GUI untuk menampilkan ke tampilan GUI, sehingga kita dapat mengetahui bahwa sensor ultrasonik mendeteksi sesuatu, dimana hal tersebut dapat mempengaruhi kerja motor agar bisa berhenti..
2. Apabila line sensor mendeteksi garis hitam maka data dari pembacaan dari line sensor akan dikirimkan pada GUI untuk menampilkan ke tampilan GUI. Dengan itu kita bisa mengetahui LED pada line sensor mana yang mendeteksi garis hitam sehingga motor bisa berjalan, sedangkan jika LED pada line sensor mendeteksi tidak adanya garis hitam maka data dari pembacaan dari line sensor yang dikirimkan pada GUI dan yang terlihat pada tampilan GUI bahwa tidak mendeteksi sehingga menyebabkan pengurangan kecepatan pada salah satu motor atau membuat keadaan motor menjadi berhenti bila line sensor tidak mendeteksi adanya garis hitam.



4.4 Unique Features

Pengimplementasian IoT pada line tracer yang memperluas kemampuan robot, seperti dapat terkoneksi secara jarak jauh saat terhubung ke internet, memungkinkan pengumpulan data secara terus menerus, memudahkan interaksi dengan operator atau pengguna salah satunya dengan menerapkan GUI. Penggunaan GUI pada multi line tracer dapat dioptimalkan dengan berbagai fitur yang memungkinkan seperti mengawasi, dan memahami cara kerja robot. Untuk fitur yang digunakan oleh multi line tracer ini yaitu sensor ultrasonik sebagai pendeteksi jarak dan objek yang terdapat disekitar robot dan sensor line sebagai pengidentifikasi garis pada jalur yang dilalui robot. Pengaplikasian algoritma kontrol PID dapat membantu robot dalam menyesuaikan kecepatan dan arah motor berdasarkan umpan balik sensor. Kemampuan line tracer dalam mendeteksi navigasi otonom juga akan mempermudah sensor dalam mendeteksi rintangan secara real-time.

5 Testing, Evaluation, and Optimization

5.1 Testing Strategy

Metode yang digunakan pada pengujian ini yaitu metode eksperimen dimana dilakukan trial and error pada program yang diujikan pada robot uji coba.

5.2 Performance Evaluation

Untuk cara kerja robot yang telah di program memiliki beberapa aspek kekurangan sehingga tidak mencapai target dan belum berfungsi dengan maksimal, hal tersebut bisa disebabkan oleh sistem yang belum bekerja sesuai fungsinya ataupun karena kondisi komponen yang tidak optimal. Pada sensor ultrasonik yang kadang tidak akurat dalam membaca objek yang ada disekitarnya, sensor line yang tidak bekerja mengikuti garis yang ditentukan, komunikasi antar mikro yang masih terdapat bug pada saat melakukan komunikasi serial. Hal tersebut menyebabkan robot tidak dapat mengetahui objek yang ada disekitarnya serta adanya beberapa data yang terlewatkan dan dapat mempersulit proses monitoring terhadap robot.

5.3 Optimization

Optimasi yang dilakukan untuk meningkatkan performa sistem berupa melakukan uji coba terus menerus sehingga jika adanya kendala dapat melakukan *Trial and error* lagi. Dilakukan juga pergantian komponen dengan robot lain yang serupa dimana bertujuan agar mengurangi kesalahan akibat kerusakan komponen. Optimasi GUI pada robot line tracer dengan berbagai jenis sensor bertujuan untuk meningkatkan pengalaman pengguna, mempermudah pemantauan, dan memungkinkan pengaturan yang efisien.



6 Collaboration and Project Management

6.1 Teamwork Dynamics

Tabel 6. Pembagian Kerja

No	Nama	Peran
1	Muhammad Irfan Habib	Membuat design GUI dan laporan
2	Ramadhan Tegar imansyah	Membuat kontrol robot dan GUI

6.2 Project Management

Tabel 7. Proyek Manajemen

No	Minggu	Pencapaian
1	Minggu 1	Menentukan target dari proyek
2	Minggu 2	Komunikasi antar mikrokontroller
3	Minggu 3	Dapat mengakses sensor ultrasonik
4	Minggu 4	Membuat tampilan dari GUI
5	Minggu 5	Mengakses sensor garis
6	Minggu 6	Membuat tampilan GUI
7	Minggu 7	<i>Trial and error</i>
8	Minggu 8	Penyusunan laporan

7 Conclusion and Reflection

7.1 Project Summary

Pencapaian yang didapatkan dalam mengerjakan proyek ini ialah menciptakan algoritma untuk mengidentifikasi dan mengikuti garis secara akurat, dapat mengimplementasikan program kedalam robot line tracer, dan mampu mengatasi rintangan yang ada pada lintasan garis.

Pembelajaran dalam proyek ini mengenai cara kerja dan fungsi tiap fitur pada GUI serta bisa mengakses sesuai dengan apa yang diinginkan melalui pengekstrakan GUI ke program dengan bahasa python, kemudian mencari cara mengakses tiap fitur sehingga bisa sesuai dengan yang diinginkan. Untuk robot line tracaer sendiri kita mempelajari bagaimana prinsip kerja setiap komponen yang ada pada line tracaer tersebut, kemudian bagaimana cara mengkoneksikan tiap komponen agar robot tersebut bisa berjalan sesuai dengan ketentuan. Hal tersulit ialah ketika memprogram robot, dimana sulit untuk mengintegrasikan tiap komponen agar bisa terkoneksi satu sama lain.



Hasil dari pembelajaran ini yaitu bisa mengetahui apa itu GUI dan fungsinya serta apa itu robot line tracer, sensor yang terpasang memiliki fungsi apa saja, dan cara memprogram robot tersebut sehingga robot mampu mengikuti jalur secara konsisten dan presisi, memiliki respons yang baik terhadap lintasan dan sensor jarak dan kemampuan untuk mengatasi rintangan dalam lintasan.

7.2 Future Work

Untuk peningkatan lebih lanjut robot multi line tracer ini dapat dikembangkan dengan peningkatan fitur yang ada pada GUI seperti, pengontrolan robot dengan menggunakan GUI, adanya pemantauan lintasan, pengaturan parameter, pemantauan sensor, dan adanya fitur-fitur tambahan lainnya. Kemudian GUI bisa menyimpan data dari hasil pembacaan sensor yang ada pada robot sehingga pemantauan robot dapat lebih optimal. Untuk peningkatan lebih lanjut dari robot ialah ditambahkan fitur kendali manual yang bisa dikontrol menggunakan joystick agar pengguna bisa mendapatkan 2 sensasi yang berbeda dalam mengontrol robot tersebut. Untuk peningkatan penelitian dan aplikasi robot secara lebih jauh, yang menyebabkan robot ini bisa di aplikasikan pada industri ataupun rumah sakit dimana ini akan membantu manusia dalam memudahkan pekerjaan sehari sehari.

7.3 Personal and Group Reflections

Berdasarkan pengerjaan multi line tracer dengan GUI (Graphical User Interface) merupakan suatu pengerjaan yang melibatkan pengembangan dan pemrograman robot atau perangkat yang mampu mengikuti atau melacak beberapa lintasan atau garis. Dalam proses pengerjaan ini memberikan pengalaman, tantangan dan pengetahuan yang didapat selama melakukan pengerjaan multi line tracer dengan GUI (Graphical User Interface) yaitu :

- Pengalaman

1. Dapat mengetahui dan menerapkan konsep pemrograman robotika serta sensor-sensor yang digunakan untuk melacak garis-garis
2. Meningkatkan pemahaman tentang bagaimana memecahkan masalah dan mengoptimalkan kinerja sistem yang dibuat.
3. Mengetahui kompleksitas pengembangan sistem navigasi robotika yang mampu mengikuti lintasan yang mana tidak hanya membutuhkan keterampilan teknis yang kuat, tetapi juga kreativitas dalam menemukan solusi untuk mengatasi berbagai kendala yang mungkin muncul. Serta kreativitas mengenai design dan layout yang sesuai dengan tampilan GUI

- Tantangan

Adanya pengerjaan multi line tracer ini memiliki tantangan yang cukup berat yaitu

1. Pembuatan program untuk kerja robot yang harus menyesuaikan dengan tujuan yang diinginkan,
2. Adanya trial dan error pada program robot yang memakan waktu cukup banyak dalam mengintegrasikan tiap sensornya
3. Perlunya pengidentifikasian tiap fitur dalam membuat GUI dan melakukan pengkoneksian dengan robot sehingga bisa menjadi satu kesatuan









- Pengetahuan
 1. Mendapatkan pengalaman yang lebih mendalam mengenai pengerjaan multi line tracer dengan GUI (Graphical User Interface)
 2. Dapat mengetahui
 3. Dapat memahami dan menerapkan cara untuk melakukan koneksi dan komunikasi antara multi line tracer dengan GUI.
 4. Melakukan Trial dan error terutama dalam mempelajari dan membuat GUI pada multi line tracer.

8 Appendices

8.1 Bill of Materials

Robot Line Tracer ini merupakan fasilitas yang telah disediakan dari Lab Vehicle sebagai modul pembelajaran untuk mahasiswa sehingga kelompok kami tidak mengeluarkan biaya.

Tabel 8. Daftar Biaya

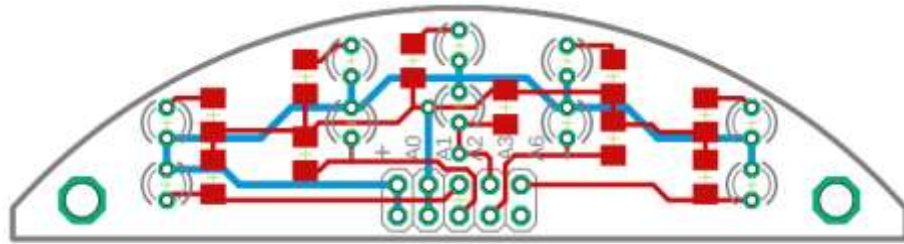
No	Komponen	Cost	No	Komponen	Cost
1	 Arduino Nano	Rp. 40.000	6	 Line sensor	Rp. 50.000
2	 Sensor Ultrasonik HC-SR04	Rp. 12.000	7	 Dip Switch	Rp. 2000
3	 Baterai Lippo 7,4V	Rp.150.000	8	 ESP 01	Rp. 15.000

- Schematics Main Board

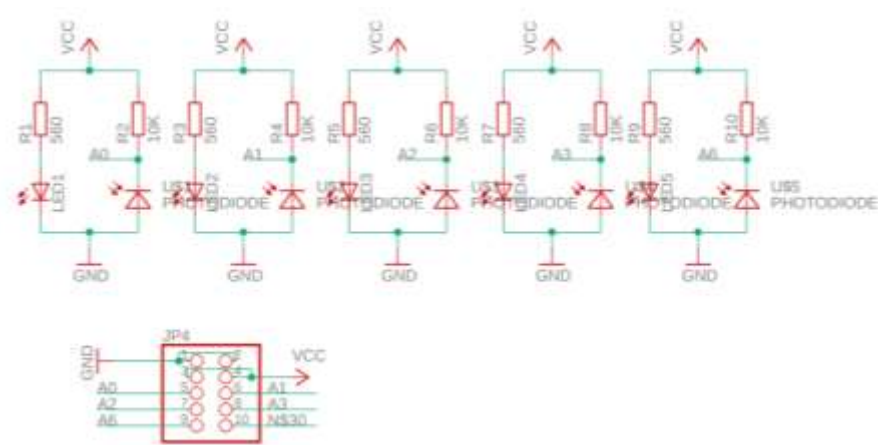




- Schematics Line Sensor



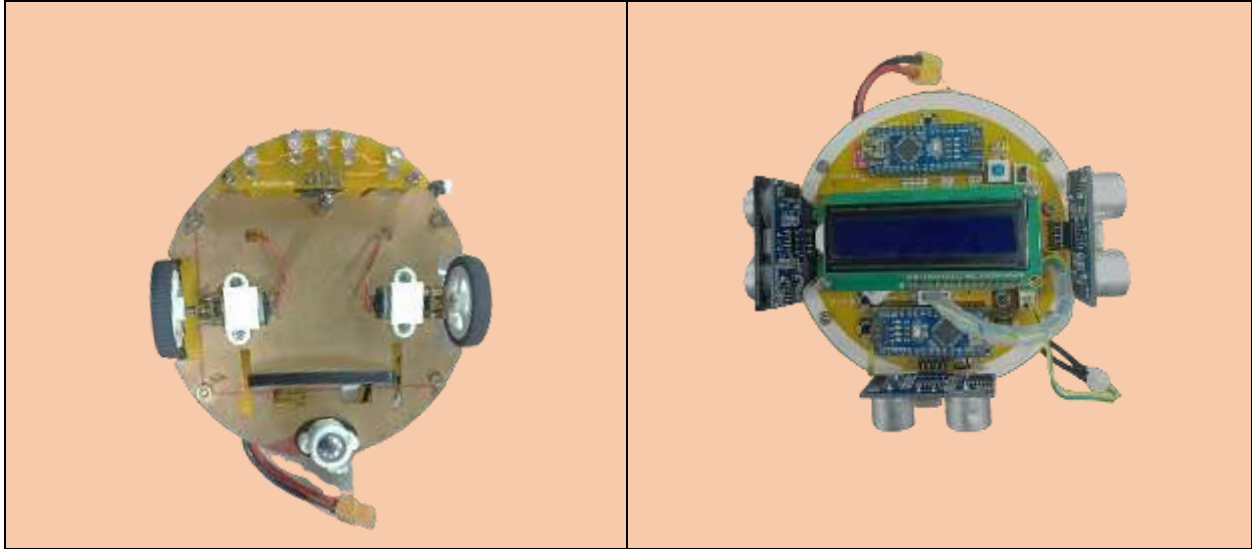
- Wiring Line Sensor



8.3 Code Repository

https://github.com/tegarRTI24/LineTracerKelompok1MKB_45_53

8.4 Additional Documentation



9 References

- Fahmi, D. 2016. RANCANG BANGUN ROBOT LINE FOLLOWER BERBASIS CAHAYA TAMPAK, 13-20.
- Jansen, B. 1998. THE GRAPHICAL USER INTERFACE. SIGCHI Bulletin, 30(2) : 2-10
- Floch, K, dkk. 2023. Motion Tracker Beta: A GUI based open-source motion tracking application. 5(2) : 1-10.