

## **BAB 4**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Rancang Bangun Robot Tempat Tidur Pasien**

Proses perancangan Robot Tempat Tidur Pasien terdiri dari dua tahap yang sangat krusial, yaitu perakitan robot dan pemasangan rangkaian pin pada komponen robot. Dalam tahap perakitan, setiap komponen harus dipasang dengan cermat dan teliti untuk memastikan kekokohan dan kinerja robot yang optimal. Sementara itu, pemasangan rangkaian pin melibatkan penghubungan yang sangat akurat antara berbagai komponen elektronik, dengan tujuan memastikan bahwa robot memiliki konektivitas yang stabil dan fungsional yang dapat diandalkan dalam menjalankan tugasnya. Dengan demikian, keseluruhan proses perancangan robot ini mengharuskan tingkat presisi yang tinggi dalam setiap langkahnya.

##### **4.1.1 Perakitan Robot**

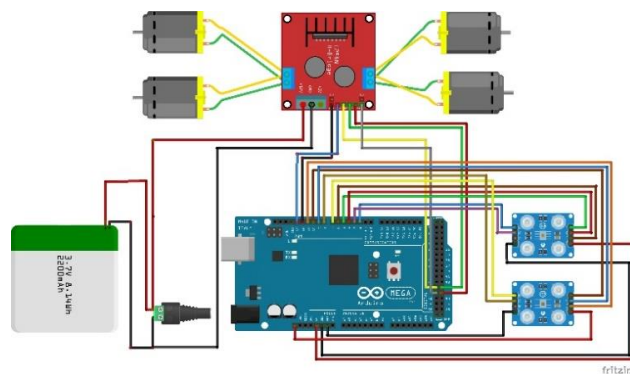
Tahapan perakitan robot dimulai setelah berhasil membuat model tiga dimensi menggunakan aplikasi pemodelan. Kemudian, proses pencetakan Chassis robot dan rangka kasurnya dilakukan melalui penggunaan 3D Printer. Setelah itu, Arduino Mega dan baterai ditempatkan dengan cermat pada Rangka Utama Chassis, yang juga berfungsi sebagai tempat pemasangan Sensor warna dan roda-roda yang telah terpasang di Rangka Penyambung Roda dan Motor DC. Setelah semua komponen dipasang dengan teliti pada Chassis Robot, langkah selanjutnya adalah memasang rangka kasur di atasnya, sehingga menyempurnakan proses perakitan robot secara menyeluruh. Selain itu, sebelum menggabungkan rangka kasur, perlu dilakukan pengujian komponen secara menyeluruh untuk memastikan bahwa semua sistem berfungsi dengan baik.



Gambar 4. 1 Robot tempat tidur pasien

#### 4.1.2 Konfigurasi Rangkaian PIN Pada Robot Kasur Pasien

Arduino Mega berfungsi sebagai pengendali utama dengan menghubungkan seluruh komponen melalui kabel-kabelnya setelah semua komponen pada robot terpasang. Komponen-komponen tersebut terhubung ke PIN yang telah disediakan oleh Arduino Mega. Diantaranya, terdapat 2 Sensor Warna dan Motor Driver yang dikoneksikan dengan Motor DC. Sementara untuk sumber daya, Baterai dihubungkan melalui *Power Jack*. Semua komponen yang dihubungkan ke Arduino Mega dicatat ke PIN mana disambungkan, dikarenakan akan diperlukan saat melakukan pemrograman untuk menjalankan program robot.



Gambar 4. 2 Konfigurasi rangkaian PIN pada robot kasur pasien

Konfigurasi Rangkaian PIN Pada Sensor Warna Kiri disambungkan ke Arduino Mega sebagai Berikut:

Tabel 4. 1 Konfigurasi rangkaian PIN pada sensor warna kiri

<b>No.</b>	<b>Sensor Warna Kiri</b>	<b>Ardino Mega</b>
1.	S0PIN	PIN 2
2.	S1PIN	PIN 3
3.	S2PIN	PIN 4
4.	S3PIN	PIN 5
5.	OUTPIN	PIN 6
6.	VCC	5V
7.	GND	GND

Konfigurasi Rangkaian PIN Pada Sensor Warna kanan disambungkan ke Arduino Mega sebagai Berikut:

Tabel 4. 2 Konfigurasi rangkaian PIN pada sensor warna kanan

<b>No.</b>	<b>Sensor Warna Kiri</b>	<b>Ardino Mega</b>
1.	S0PIN	PIN 7
2.	S1PIN	PIN 8
3.	S2PIN	PIN 9
4.	S3PIN	PIN 10
5.	OUTPIN	PIN 11
6.	VCC	5V
7.	GND	GND

Konfigurasi Rangkaian PIN Pada *Motor Driver* disambungkan ke Arduino Mega sebagai Berikut:

Tabel 4. 3 Konfigurasi rangkaian PIN pada *Motor Driver*

<b>No.</b>	<b>Sensor Warna Kiri</b>	<b>Ardino Mega</b>
1.	ENAPIN	PIN 12
2.	IN1PIN	PIN 13
3.	IN2PIN	PIN 44
4.	IN3PIN	PIN 45
5.	IN4PIN	PIN 47
6.	ENBPIN	PIN 46

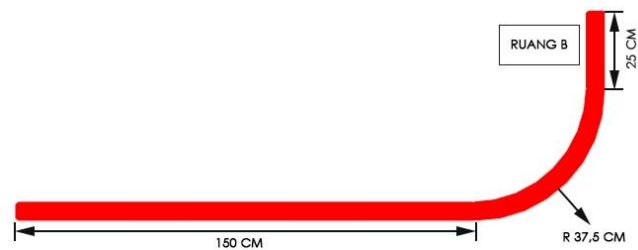
## 4.2 Pengujian Robot Tempat Tidur Pasien

Pengujian dilakukan pada empat garis berwarna yaitu merah, kuning, hijau, dan hitam secara terpisah. Pada garis merah, Robot Tempat Tidur Pasien akan bergerak lebih cepat daripada garis kuning, kemudian diikuti garis hijau dan hitam yang akan bergerak dengan kecepatan yang sama. Perbedaan kecepatan ini diatur melalui *Coding Program*.

### 4.2.1 Analisis Pada Lintasan Garis Merah

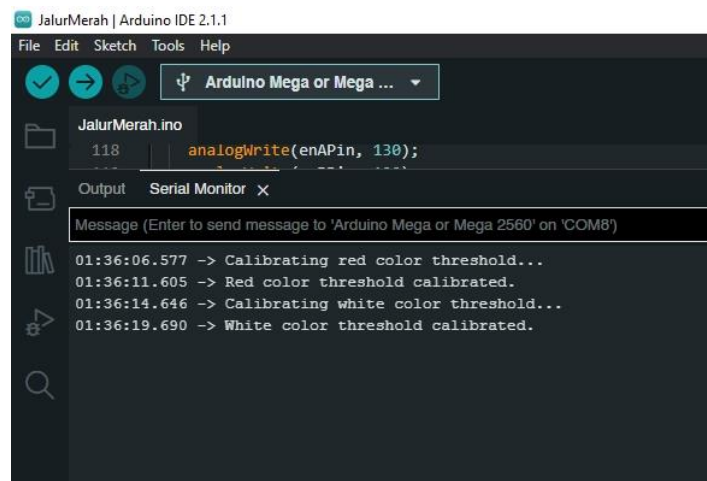
Robot Tempat Tidur Pasien akan mengoperasikan dirinya di atas jalur berwarna merah yang memiliki panjang total sekitar 233,93 cm. Panjang jalur ini telah dihitung berdasarkan karakteristik khusus yang mencakup tiga segmen utama. Pertama-tama, terdapat segmen lurus sepanjang 150 cm yang menantang robot untuk menjaga stabilitas dan akurasi pergerakan selama perjalanan yang cukup panjang ini. Kemudian, jalur tersebut memasukkan elemen berbelok ke kiri membentuk seperempat lingkaran dengan jari-jari sekitar 37,5 cm, yang menguji kemampuan robot dalam mengikuti kurva dengan presisi guna mempertahankan jalur yang telah ditentukan sebelumnya. Akhirnya, jalur ini mengakhiri perjalanannya dengan segmen lurus terakhir sepanjang 25 cm, yang menjadi ujian bagi kemampuan robot untuk menghentikan pergerakannya secara akurat pada posisi yang telah ditentukan. Setiap segmen dalam jalur ini mempresentasikan tantangan tersendiri yang harus diatasi oleh robot selama perjalanan.





Gambar 4. 3 Lintasan garis merah

Sebelum Robot Tempat Tidur Pasien beroperasi di lintasan garis merah, kedua Sensor warna akan mengalami proses mengatur warna pada lintasan garis merah dan latar belakang yang berwarna putih. Hal ini bertujuan agar robot dapat membedakan dengan baik antara garis merah dan latar belakang putih.



Gambar 4. 4 Proses mengatur warna lintasan garis merah

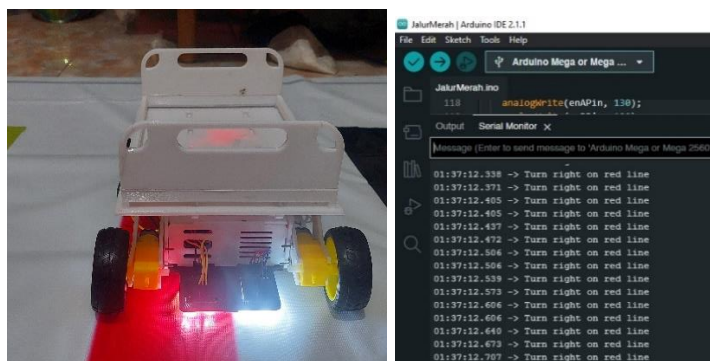
Setelah berhasil proses mengatur warna garis merah berhasil, Robot Tempat Tidur Pasien akan melanjutkan dengan mengatur warna latar belakang berwarna putih. Langkah ini dilakukan dengan menempatkan kedua sensor pada latar belakang putih setelah muncul kalimat "*Calibrating White Color Threshold*". Proses mengatur warna latar belakang putih dianggap berhasil ketika kalimat "*White Color Threshold Calibrated*" muncul. Setelah proses

mengatur warna berhasil, Robot Tempat Tidur Pasien dapat Melaju di Lintasan Garis Merah.

Untuk memastikan Robot Tempat Tidur Pasien dapat berjalan secara tepat dan akurat pada Lintasan Garis Merah, diterapkan konfigurasi khusus berdasarkan pembacaan kedua sensor warna. Ketika kedua sensor warna melintasi garis merah secara bersamaan, robot akan bergerak maju dengan jalur lurus. Namun, jika sensor warna kiri melintasi latar belakang putih dan sensor warna kanan melintasi garis merah, maka Robot Tempat Tidur Pasien akan berbelok ke kanan. Sebaliknya, jika sensor warna kiri melintasi garis merah dan sensor warna kanan melintasi latar belakang putih, maka Robot Tempat Tidur Pasien akan berbelok ke kiri.



Gambar 4. 5 Konfigurasi bergerak maju di garis merah



Gambar 4. 6 Konfigurasi berbelok ke kanan di garis merah



Gambar 4. 7 Konfigurasi berbelok ke kiri di garis merah

Pengujian Robot Tempat Tidur Pasien dilakukan sebanyak 10 kali untuk mengetahui kemampuan robot dalam beroperasi dengan aman dan efisien. Pengujian ini dilakukan dengan menggunakan jalur berwarna merah dengan panjang total sekitar 233,93 cm. Dengan melakukan pengujian ini, dapat dipastikan bahwa robot Tempat Tidur Pasien dapat beroperasi dengan aman dan efisien, sehingga dapat memberikan perawatan yang tepat dan cepat kepada pasien. Kecepatan robot dapat diperoleh dengan menggunakan Persamaan berikut:

$$v = \frac{s}{t} \quad (4. 1)$$

Dimana :

v : Kecepatan Robot Tempat Tidur Pasien (cm/detik)

s : Jarak titik awal lintasan ke titik akhir lintasan (cm)

t : Waktu tempot Robot Tempat Tidur Pasien (detik)

Hasil pengujian kecepatan Robot Tempat Tidur Pasien dapat di hitung menggunakan persamaan di atas adalah :

Tabel 4. 4 Pengujian pada lintasan garis merah

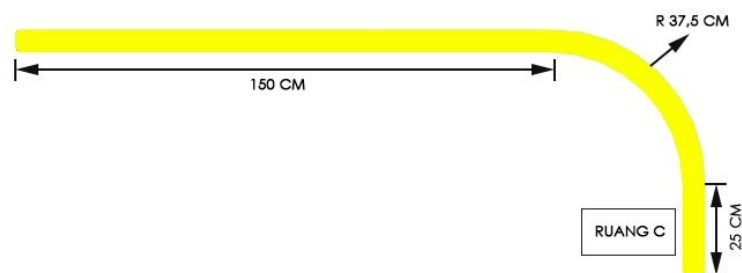
No	Jarak (cm)	Waktu (Detik)	Kecepatan (cm/detik)
1.	233,93	8,25	28,25
2.	233,93	8,43	27,74
3.	233,93	9,15	25,56
4.	233,93	8,48	27,58
5.	233,93	8,53	27,42
6.	233,93	9,04	25,87
7.	233,93	8,56	27,32
8.	233,93	8,33	28,08
9.	233,93	8,58	27,26
10.	233,93	8,23	28,34

Robot melakukan pengujian sebanyak 10 kali pada Lintasan Garis Merah dengan jarak 233,93 cm. Robot Tempat Tidur Pasien mampu menempuh perjalanan dengan waktu tempuh rata-rata sebesar 8,65 detik. Robot menempuh dengan Kecepatan rata-rata sekitar 27,45 cm/detik.

#### 4.2.2 Analisis Pada Lintasan Garis Kuning

Robot Tempat Tidur Pasien akan beroperasi pada jalur berwarna kuning dengan panjang total sekitar 233,93 cm. Panjang jalur ini diperoleh berdasarkan karakteristik jalur yang terdiri dari segmen lurus sepanjang 150 cm, diikuti dengan belokan ke kanan membentuk seperempat lingkaran dengan jari-jari sekitar 37,5 cm, dan diakhiri dengan segmen lurus sepanjang 25 cm. Setiap segmen jalur ini memiliki peran dan tantangan tersendiri dalam perjalanan robot. Segmen lurus pertama menuntut robot untuk menjaga kestabilan dan akurasi. Selanjutnya, belokan ke kanan mengharuskan robot untuk mengikuti kurva yang ditentukan. Pada akhir, segmen lurus terakhir memberikan tantangan bagi robot untuk menghentikan pergerakannya dengan cepat dan aman. Segmen ini memiliki panjang sekitar 25 cm. Pada segmen ini, robot harus mampu berhenti dengan kecepatan yang cukup rendah agar tidak membahayakan pasien.. Dengan adanya tantangan-tantangan ini, robot Tempat Tidur Pasien dapat diuji kemampuannya untuk beroperasi dengan aman dan efisien





Gambar 4. 8 Lintasan garis kuning

Sebelum Robot Tempat Tidur Pasien beroperasi di lintasan garis kuning, kedua Sensor warna akan mengalami proses mengatur warna pada lintasan garis kuning dan latar belakang yang berwarna putih. Hal ini bertujuan agar robot dapat membedakan dengan baik antara garis kuning dan latar belakang putih.

```

JalurKuning | Arduino IDE 2.1.1
File Edit Sketch Tools Help
Arduino Mega or Mega ...
JalurKuning.ino
248 Serial.println( "Calibrating yellow color threshold..."
249 Serial.println( "Yellow color threshold calibrated."
Output Serial Monitor X
Message (Enter to send message to 'Arduino Mega or Mega 2560' on 'COM8')
01:30:59.145 -> Calibrating yellow color threshold...
01:31:04.186 -> yellow color threshold calibrated.
01:31:07.192 -> Calibrating white color threshold...
01:31:12.239 -> White color threshold calibrated.

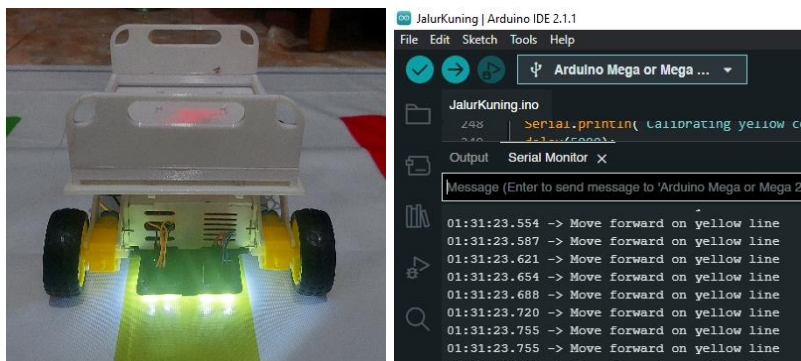
```

Gambar 4. 9 Proses mengatur warna lintasan garis kuning

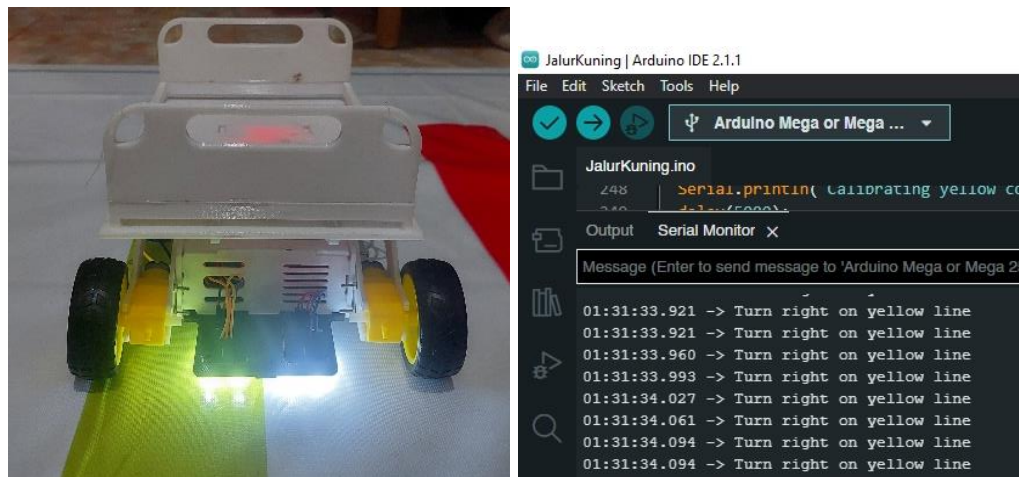
Proses mengatur warna dimulai dengan menempatkan kedua sensor pada garis kuning setelah muncul kalimat "*Calibrating Yellow Color Threshold*" pada serial monitor Arduino IDE. Mengatur warna garis merah dianggap berhasil ketika kalimat "*Yellow Color Threshold Calibrated*" muncul.

Setelah berhasil proses mengatur warna garis kuning berhasil, Robot Tempat Tidur Pasien akan melanjutkan dengan mengatur warna latar belakang berwarna putih. Langkah ini dilakukan dengan menempatkan kedua sensor pada latar belakang putih setelah muncul kalimat "*Calibrating White Color Threshold*". Proses mengatur warna latar belakang putih dianggap berhasil ketika kalimat "*White Color Threshold Calibrated*" muncul. Setelah proses mengatur warna berhasil, Robot Tempat Tidur Pasien dapat Melaju di Lintasan Garis Kuning.

Untuk memastikan Robot Tempat Tidur Pasien dapat berjalan secara tepat dan akurat pada Lintasan Garis Kuning, diterapkan konfigurasi khusus berdasarkan pembacaan kedua sensor warna. Ketika kedua sensor warna melintasi garis kuning secara bersamaan, robot akan bergerak maju dengan jalur lurus. Namun, jika sensor warna kiri melintasi latar belakang putih dan sensor warna kanan melintasi garis kuning, maka Robot Tempat Tidur Pasien akan berbelok ke kanan. Sebaliknya, jika sensor warna kiri melintasi garis kuning dan sensor warna kanan melintasi latar belakang putih, maka Robot Tempat Tidur Pasien akan berbelok ke kiri.



Gambar 4. 10 Konfigurasi bergerak maju di garis kuning



Gambar 4. 11 Konfigurasi berbelok ke kanan di garis kuning



Gambar 4. 12 Konfigurasi berbelok ke kanan di garis kuning

Pengujian Robot Tempat Tidur Pasien dilakukan sebanyak 10 kali untuk mengetahui kemampuan robot dalam beroperasi dengan aman dan efisien. Pengujian ini dilakukan dengan menggunakan jalur berwarna kuning dengan panjang total sekitar 233,93 cm. Dengan melakukan pengujian ini, dapat dipastikan bahwa robot Tempat Tidur Pasien dapat beroperasi dengan aman dan efisien, sehingga dapat memberikan perawatan yang tepat dan cepat kepada pasien. Kecepatan Robot Tempat Tidur Pasien dapat diperoleh dengan menggunakan Rumus Kecepatan pada Persaman 4.1.

Hasil pengujian kecepatan Robot Tempat Tidur Pasien dapat di hitung menggunakan persamaan di atas adalah :

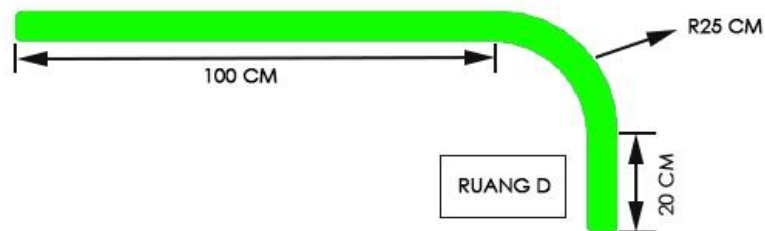
Tabel 4. 5 Pengujian pada lintasan garis kuning

No	Jarak (cm)	Waktu (Detik)	Kecepatan (cm/detik)
1.	233,93	10,11	23,13
2.	233,93	10,32	22,66
3.	233,93	10,15	23,04
4.	233,93	11,05	21,17
5.	233,93	10,41	22,47
6.	233,93	11,08	21,11
7.	233,93	10,18	22,97
8.	233,93	11,14	20,99
9.	233,93	10,25	22,82
10.	233,93	10,47	22,34

Robot melakukan pengujian sebanyak 10 kali pada Lintasan Garis Merah dengan jarak 233,93 cm. Robot Tempat Tidur Pasien mampu menempuh perjalanan dengan waktu tempuh rata-rata sebesar 10,50 detik. Robot menempuh dengan Kecepatan rata-rata sekitar 22,71 cm/detik.

#### 4.2.3 Analisis Pada Lintasan Garis Hijau

Robot Tempat Tidur Pasien akan beroperasi di lintasan garis hijau yang memiliki panjang 159,27 cm. Panjang garis ini diperoleh berdasarkan karakteristik garis yang terdiri dari garis lurus sepanjang 100 cm, kemudian akan berbelok ke kanan pada seperempat lingkaran dengan jari-jari 25 cm, dan garis lurus sepanjang terakhir sepanjang 20 cm. Setiap segmen jalur ini memiliki peran dan tantangan tersendiri dalam perjalanan robot. Setiap segmen jalur ini memiliki peran dan tantangan tersendiri dalam perjalanan robot. Segmen lurus pertama menuntut robot untuk menjaga kestabilan dan akurasi. Selanjutnya, belokan ke kanan mengharuskan robot untuk mengikuti kurva yang ditentukan. Pada akhir, segmen lurus terakhir menguji kemampuan robot dalam menghentikan pergerakannya.



Gambar 4. 13 Lintasan garis hijau

Sebelum Robot Tempat Tidur Pasien beroperasi di lintasan garis hijau, kedua Sensor warna akan mengalami proses mengatur warna pada lintasan garis hijau dan latar belakang yang berwarna putih. Hal ini bertujuan agar robot dapat membedakan dengan baik antara garis hijau dan latar belakang putih.

 A screenshot of the Arduino IDE interface. The Serial Monitor window is open, displaying the following messages:
 

```

01:24:28.367 -> Calibrating green color threshold...
01:24:33.409 -> Green color threshold calibrated.
01:24:36.438 -> Calibrating white color threshold...
01:24:41.460 -> White color threshold calibrated.
  
```

Gambar 4. 14 Proses mengatur warna lintasan garis hijau

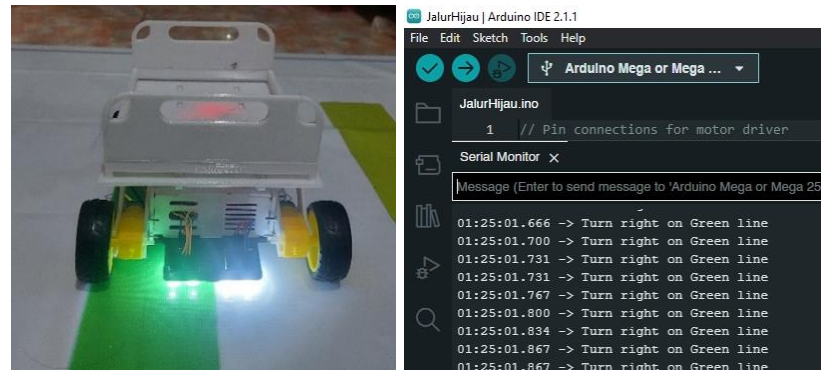
Proses mengatur warna dimulai dengan menempatkan kedua sensor pada garis hijau setelah muncul kalimat "*Calibrating Green Color Threshold*" pada serial monitor Arduino IDE. Mengatur warna garis merah dianggap berhasil ketika kalimat "*Green Color Threshold Calibrated*" muncul.

Setelah berhasil proses mengatur warna garis hijau berhasil, Robot Tempat Tidur Pasien akan melanjutkan dengan mengatur warna latar belakang berwarna putih. Langkah ini dilakukan dengan menempatkan kedua sensor pada latar belakang putih setelah muncul kalimat "*Calibrating White Color Threshold*". Proses mengatur warna latar belakang putih dianggap berhasil ketika kalimat "*White Color Threshold Calibrated*" muncul. Setelah proses pengklibrasian berhasil, Robot Tempat Tidur Pasien dapat Melaju di Lintasan Garis Hijau.

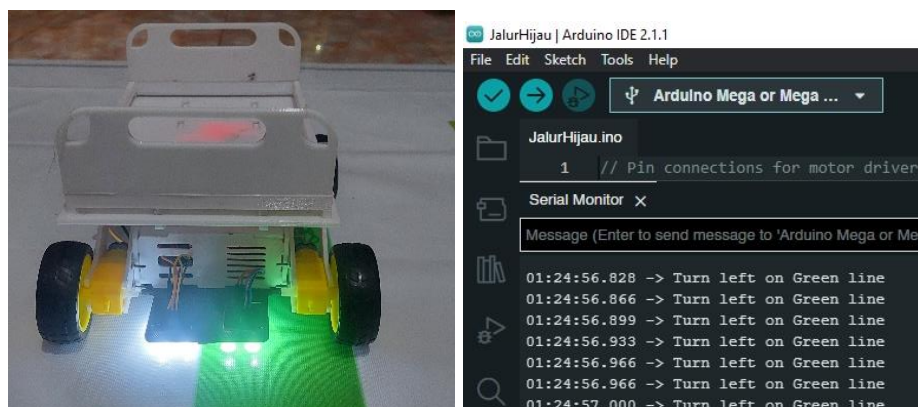
Untuk memastikan Robot Tempat Tidur Pasien dapat berjalan secara tepat dan akurat pada Lintasan Garis Hijau, diterapkan konfigurasi khusus berdasarkan pembacaan kedua sensor warna. Ketika kedua sensor warna melintasi garis hijau secara bersamaan, robot akan bergerak maju dengan jalur lurus. Namun, jika sensor warna kiri melintasi latar belakang putih dan sensor warna kanan melintasi garis hijau, maka Robot Tempat Tidur Pasien akan berbelok ke kanan. Sebaliknya, jika sensor warna kiri melintasi garis hijau dan sensor warna kanan melintasi latar belakang putih, maka Robot Tempat Tidur Pasien akan berbelok ke kiri.



Gambar 4. 15 Konfigurasi bergerak maju di garis hijau



Gambar 4. 16 Konfigurasi berbelok ke kanan di garis hijau



Gambar 4. 17 Konfigurasi berbelok ke kiri di garis hijau

Pengujian Robot Tempat Tidur Pasien dilakukan sebanyak 10 kali untuk mengetahui kemampuan robot dalam beroperasi dengan aman dan efisien. Pengujian ini dilakukan dengan menggunakan jalur berwarna hijau dengan panjang total sekitar 159,27 cm. Dengan melakukan pengujian ini, dapat dipastikan bahwa robot Tempat Tidur Pasien dapat beroperasi dengan aman dan efisien, sehingga dapat memberikan perawatan yang tepat dan cepat kepada pasien. Kecepatan Robot Tempat Tidur Pasien dapat diperoleh dengan menggunakan Rumus Kecepatan pada Persaman 4.1.

Hasil pengujian kecepatan Robot Tempat Tidur Pasien dapat di hitung menggunakan persamaan di atas adalah :

Tabel 4. 6 Pengujian pada lintasan garis hijau

No	Jarak (cm)	Waktu (Detik)	Kecepatan (cm/detik)
1.	159,27	9,01	17,67
2.	159,27	9,56	16,66
3.	159,27	9,75	16,33
4.	159,27	9,11	17,48
5.	159,27	9,04	17,61
6.	159,27	10,12	15,73
7.	159,27	9,27	17,18
8.	159,27	9,41	16,92
9.	159,27	9,67	16,47
10.	159,27	9,81	16,23

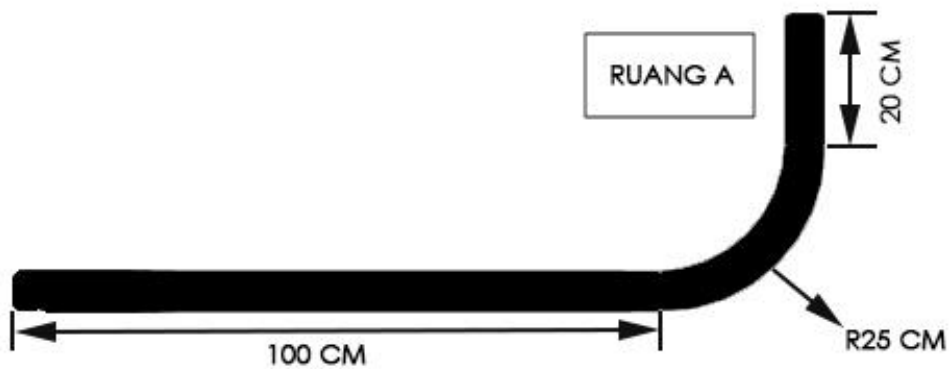
Robot Tempat Tidur Pasien telah menjalani sebanyak 10 kali pengujian di Lintasan Garis Merah yang memiliki panjang sekitar 159,27 cm. Hasil dari pengujian ini menunjukkan bahwa robot mampu menyelesaikan perjalanan dengan waktu tempuh rata-rata sekitar 9,57 detik. Selama perjalanan tersebut, robot mencapai kecepatan rata-rata sekitar 16,61 cm/detik.

#### 4.2.4 Analisis Pada Lintasan Garis Hitam

Robot Tempat Tidur Pasien akan beroperasi di lintasan garis hitam yang memiliki panjang 159,27 cm. Panjang garis ini diperoleh berdasarkan karakteristik garis yang terdiri dari garis lurus sepanjang 100 cm, kemudian akan berbelok ke kiri pada seperempat lingkaran dengan jari-jari 25 cm, dan garis lurus sepanjang terakhir sepanjang 20 cm.







Gambar 4. 18 Lintasan garis hitam

Sebelum Robot Tempat Tidur Pasien beroperasi di lintasan garis hitam, kedua Sensor warna akan mengalami proses mengatur warna pada lintasan garis hitam dan latar belakang yang berwarna putih. Hal ini bertujuan agar robot dapat membedakan dengan baik antara garis hitam dan latar belakang putih.

```

JalurHitam | Arduino IDE 2.1.1
File Edit Sketch Tools Help
Arduino Mega or Mega ...
JalurHitam.ino
35 int whiteR_right, whiteG_right, whiteB_right;
Output Serial Monitor x
Message (Enter to send message to 'Arduino Mega or Mega 2560' on 'COM8')
01:40:09.272 -> Calibrating black color threshold...
01:40:14.297 -> Black color threshold calibrated.
01:40:17.329 -> Calibrating white color threshold...
01:40:22.389 -> White color threshold calibrated.

```

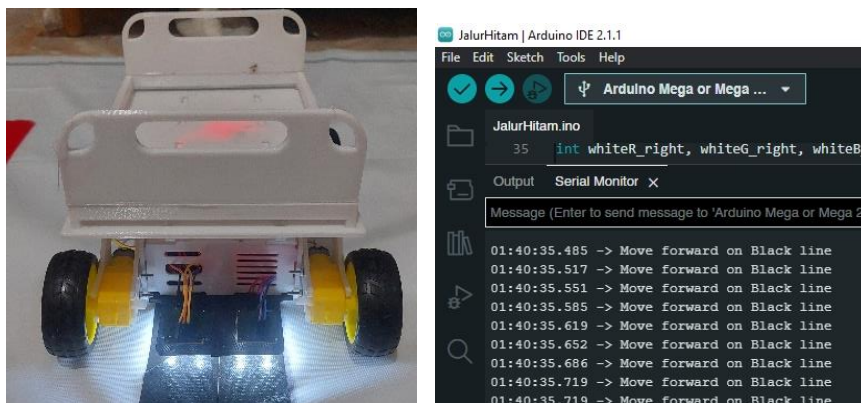
Gambar 4. 19 Proses mengatur warna lintasan garis hitam

Proses mengatur warna dimulai dengan menempatkan kedua sensor pada garis hitam setelah muncul kalimat "*Calibrating Black Color Threshold*" pada

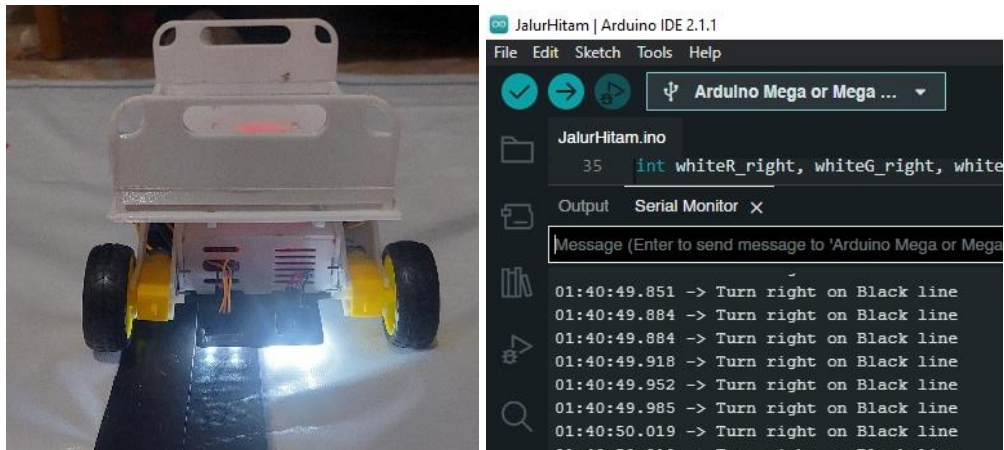
serial monitor Arduino IDE. Mengatur warna garis merah dianggap berhasil ketika kalimat "*Black Color Threshold Calibrated*" muncul.

Setelah berhasil proses mengatur warna garis hitam berhasil, Robot Tempat Tidur Pasien akan melanjutkan dengan mengatur warna latar belakang berwarna putih. Langkah ini dilakukan dengan menempatkan kedua sensor pada latar belakang putih setelah muncul kalimat "*Calibrating White Color Threshold*". Proses mengatur warna latar belakang putih dianggap berhasil ketika kalimat "*White Color Threshold Calibrated*" muncul. Setelah proses pengklibrasian berhasil, Robot Tempat Tidur Pasien dapat Melaju di Lintasan Garis Hitam.

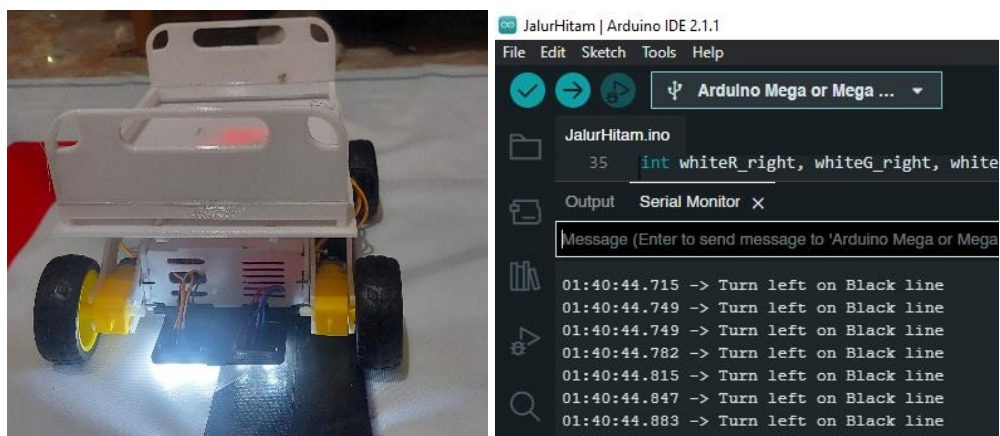
Untuk memastikan Robot Tempat Tidur Pasien dapat berjalan secara tepat dan akurat pada Lintasan Garis Hitam, diterapkan konfigurasi khusus berdasarkan pembacaan kedua sensor warna. Ketika kedua sensor warna melintasi garis hitam secara bersamaan, robot akan bergerak maju dengan jalur lurus. Namun, jika sensor warna kiri melintasi latar belakang putih dan sensor warna kanan melintasi garis hitam, maka Robot Tempat Tidur Pasien akan berbelok ke kanan. Sebaliknya, jika sensor warna kiri melintasi garis hitam dan sensor warna kanan melintasi latar belakang putih, maka Robot Tempat Tidur Pasien akan berbelok ke kiri.



Gambar 4. 20 Konfigurasi bergerak maju di garis hitam



Gambar 4. 21 Konfigurasi berbelok ke kanan di garis hitam



Gambar 4. 22 Konfigurasi berbelok ke kiri di garis hitam

Pengujian Robot Tempat Tidur Pasien dilakukan sebanyak 10 kali untuk mengetahui kemampuan robot dalam beroperasi dengan aman dan efisien. Pengujian ini dilakukan dengan menggunakan jalur berwarna hitam dengan panjang total sekitar 159,27 cm. Dengan melakukan pengujian ini, dapat dipastikan bahwa robot Tempat Tidur Pasien dapat beroperasi dengan aman dan efisien, sehingga dapat memberikan perawatan yang tepat dan cepat kepada pasien. Kecepatan Robot Tempat Tidur Pasien dapat diperoleh dengan menggunakan Rumus Kecepatan pada Persaman 4.1.

Hasil pengujian kecepatan Robot Tempat Tidur Pasien dapat di hitung menggunakan persamaan di atas adalah :

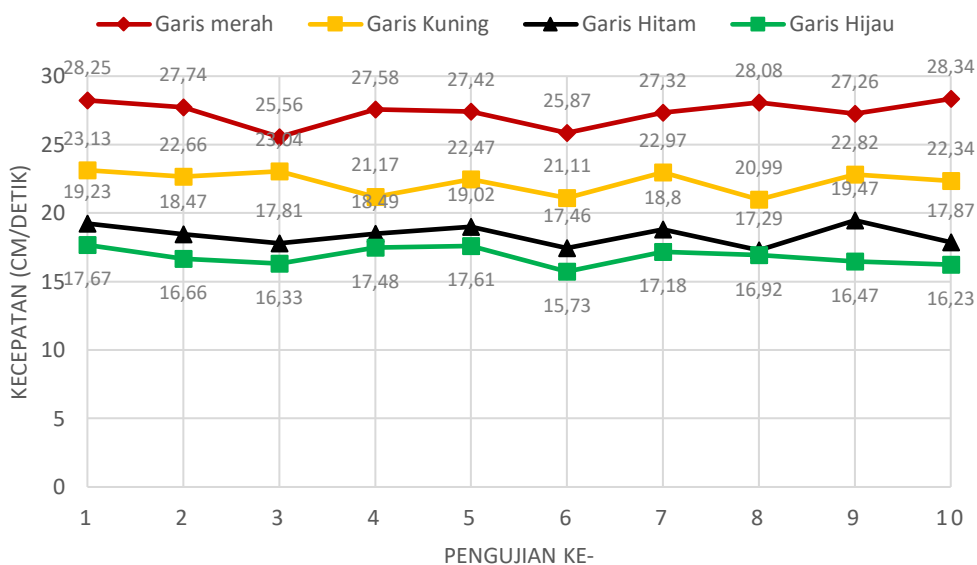
Tabel 4. 7 Pengujian pada lintasan garis hitam

No	Jarak (cm)	Waktu (Detik)	Kecepatan (cm/detik)
1.	159,27	8,28	19,23
2.	159,27	8,62	18,47
3.	159,27	8,94	17,81
4.	159,27	8,61	18,49
5.	159,27	8,37	19,02
6.	159,27	9,12	17,46
7.	159,27	8,47	18,80
8.	159,27	9,21	17,29
9.	159,27	9,18	19,47
10.	159,27	8,91	17,87

Robot melakukan pengujian sebanyak 10 kali pada Lintasan Garis Merah dengan jarak 159,27 cm. Robot Tempat Tidur Pasien mampu menempuh perjalanan dengan waktu tempuh rata-rata sebesar 8,77 detik. Robot menempuh dengan Kecepatan rata-rata sekitar 18,59 cm/detik.

### 4.3 Analisis Kecepatan Robot Tempat Tidur Pasien Berdasarkan Penggunaan Triase

Kecepatan Robot Tempat Tidur Pasien memiliki peran yang sangat penting dalam penanganan pasien dalam situasi darurat. Oleh karena itu, kecepatan robot akan disesuaikan berdasarkan warna lintasan yang ditempuhnya. Di setiap warna lintasan yang berbeda, Robot Tempat Tidur Pasien akan diatur untuk bergerak dengan kecepatan yang sesuai dengan warna garis yang sedang dilaluinya. Hal ini bertujuan untuk memastikan bahwa pergerakan robot dapat berjalan dengan optimal dan tepat sesuai dengan standar Triase yang digunakan. Dengan menyesuaikan kecepatan robot berdasarkan warna lintasan, maka robot dapat mengantarkan pasien ke tempat perawatan yang tepat dengan cepat dan aman.

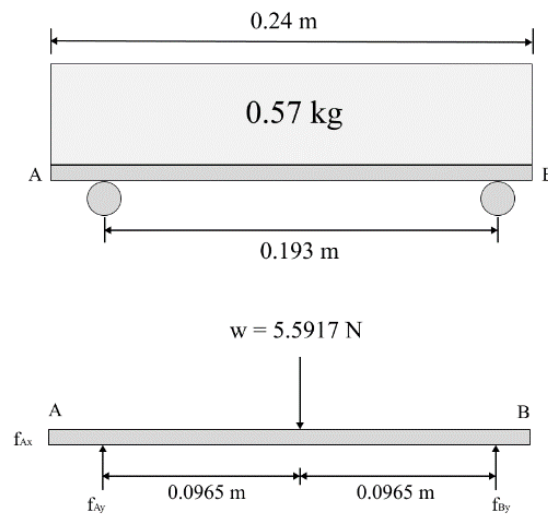


Gambar 4. 23 Grafik kecepatan pada setiap warna garis

Berdasarkan analisis pada Gambar 4.23, lintasan garis berwarna merah menunjukkan kecepatan paling cepat di antara lintasan garis lainnya. Hal ini disebabkan oleh prioritas tertinggi dalam sistem triase, di mana pasien dengan kondisi darurat yang memerlukan penanganan segera ditandai dengan warna merah. Lintasan garis berwarna kuning memiliki kecepatan di bawah lintasan berwarna merah karena pasien pada kategori ini masih dapat menunggu perawatan sementara waktu dan tidak membutuhkan penanganan mendesak seperti pasien dengan warna merah. Lintasan garis berwarna hitam menunjukkan kecepatan yang lebih lambat dari garis kuning karena pasien dalam kategori ini mengalami cedera yang sangat parah. Pasien dengan warna hitam dapat dikategorikan pasien yang kemungkinan untuk hidup sangat kecil, luka sangat parah. Sementara itu, lintasan garis berwarna hijau menunjukkan kecepatan yang paling lambat. Pasien dalam kategori ini tidak memiliki prioritas untuk menggunakan Tempat Tidur Pasien karena kondisi mereka tidak memerlukan penanganan mendesak. Penerapan sistem kecepatan robot berdasarkan warna lintasan ini dapat membantu meningkatkan keselamatan pasien dalam situasi darurat.

#### 4.4 Analisis Kesetimbangan Robot Tempat Tidur Pasien

Analisis Kesetimbangan Robot Tempat Tidur Pasien melibatkan perhitungan dalam distribusi massa, titik pusat gravitasi, dan interaksi antara komponen mekanik. Kesetimbangan termasuk dalam ilmu statika yaitu ilmu yang mempelajari benda dalam keadaan diam atau statis. Tahap awal dalam menganalisis kesetimbangan robot dengan menggambarkan diagram benda bebas robot.



Gambar 4. 24 Diagram benda bebas tempat tidur pasien

Setelah Diagram Benda Bebas Telah dibuat, barulah dapat di analisis kesetimbangan Robot Tempat Tidur Pasien. Robot Tempat Tidur Pasien dapat dikatakan setimbang jika semua resultan gaya yang bekerja pada robot yaitu sama dengan nol, dan jumlah dari semua momen gaya pada robot juga sama dengan nol. Jadi perhitungan dalam analisis kesetimbangan Robot Tempat Tidur Pasien sebagai berikut.

Untuk mencari nilai  $w$

$$w = m \times g \quad (4.2)$$

$$= 0.57 \times 9.81$$

$$= 5.5917 \text{ N}$$

Selanjutnya analisis gaya yang bekerja pada robot

$$\Sigma f_y = 0$$

$$f_{Ay} + f_{By} - w = 0$$

$$f_{Ay} + f_{By} - 5.5917 \text{ N} = 0 \quad (1)$$

$$\Sigma M_A = 0$$

$$- 5.5917 \text{ N} \times 0.0965 \text{ m} + f_{By} \times 0.193 \text{ m} = 0$$

$$- 0.53959905 + f_{By} \times 0.193 \text{ m} = 0$$

$$f_{By} = 2.79585 \text{ N} \quad (2)$$

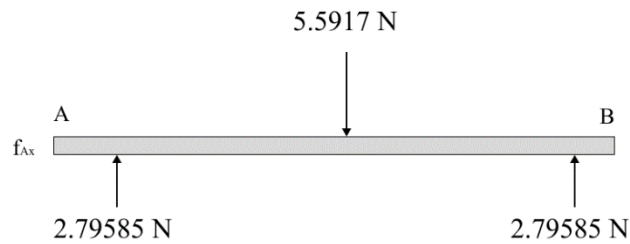
Substitusi Persamaan (1) ke Persamaan (2)

$$f_{Ay} + f_{By} - 5.5917 \text{ N} = 0$$

$$f_{Ay} + 2.79585 - 5.5917 \text{ N} = 0$$

$$f_{Ay} = 2.79585 \text{ N}$$

Gaya yang bekerja pada kedua tumpuan roda memiliki besaran yang sama, yaitu sekitar 2.79585 N, dengan arah yang berlawanan, yaitu  $f_{Ay}$  dan  $f_{By}$ . Pada kedua tumpuan tersebut, terdapat gaya-gaya yang saling berimbang, yang artinya gaya-gaya ini menjaga keseimbangan sistem. Jumlah gaya yang bekerja pada kedua tumpuan tersebut adalah 5.5917 N, yang terbagi secara simetris antara  $f_{Ay}$  dan  $f_{By}$ , sehingga sistem tetap dalam keadaan stabil.



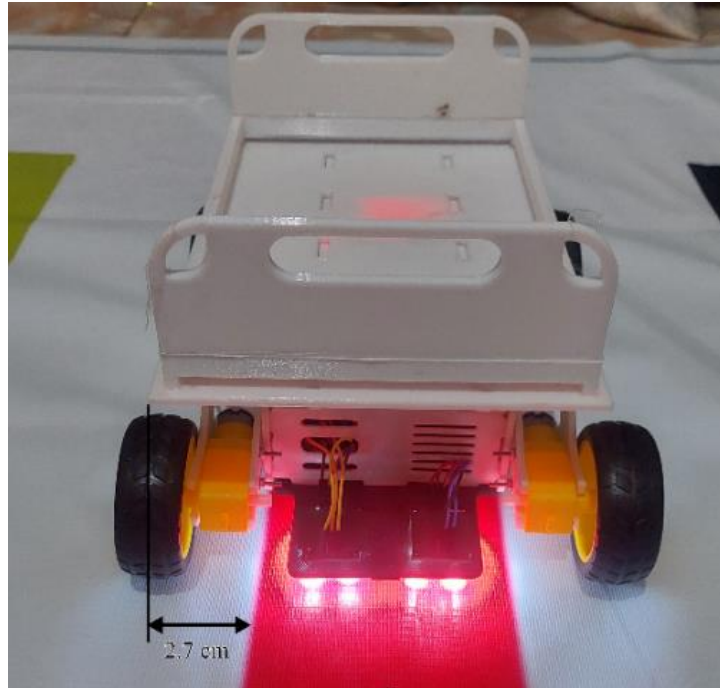
Gambar 4. 25 Analisis kesetimbangan robot tempat tidur pasien

#### 4.5 Analisis *Error* pada Robot Tempat Tidur Pasien

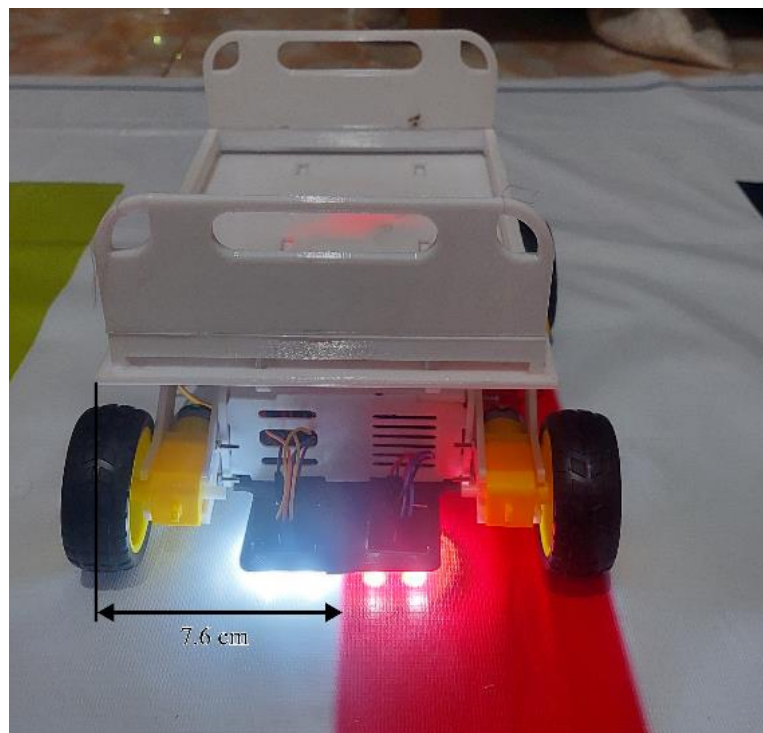
Dalam proses perjalanan melintasi jalur lintasan, Robot Tempat Tidur Pasien dapat mengalami *error* atau kegagalan yang mengakibatkan keluar dari lintasan yang ditentukan. *Error* atau kegagalan ini terjadi ketika robot keluar dari jalur yang seharusnya diikuti. *Error* dapat terdeteksi jika sensor membaca latar belakang lintasan yang berwarna putih. Analisis terhadap *error* melibatkan evaluasi terhadap jarak antara robot dan lintasan saat terjadi kesalahan, serta jumlah kesalahan yang terjadi saat robot berusaha melintasi lintasan. Dengan memahami seberapa jauh robot keluar dari lintasan dan seberapa sering kesalahan terjadi, dapat membantu dalam mengidentifikasi masalah potensial dalam perangkat keras atau perangkat lunak robot.

Dalam Gambar 4. 26 dan Gambar 4. 27 Robot Tempat Tidur Pasien berada tepat dalam lintasan dengan jarak sekitar 2,7 cm dari sisi robot ke garis lintasan. Posisi ini dianggap sebagai posisi ideal karena robot berada di tengah-tengah lintasan dengan baik. Namun, jika jarak antara sisi robot dan garis lintasan mencapai 7,6 cm, maka robot akan mengalami kesalahan atau *error*. Untuk mengatasi kesalahan ini, robot akan mengambil tindakan dengan berbelok, sehingga robot dapat kembali berada di tengah-tengah lintasan yang diinginkan. Dengan demikian, robot akan memperbaiki posisinya dan menjaga kinerjanya sesuai dengan jalur yang ditentukan.

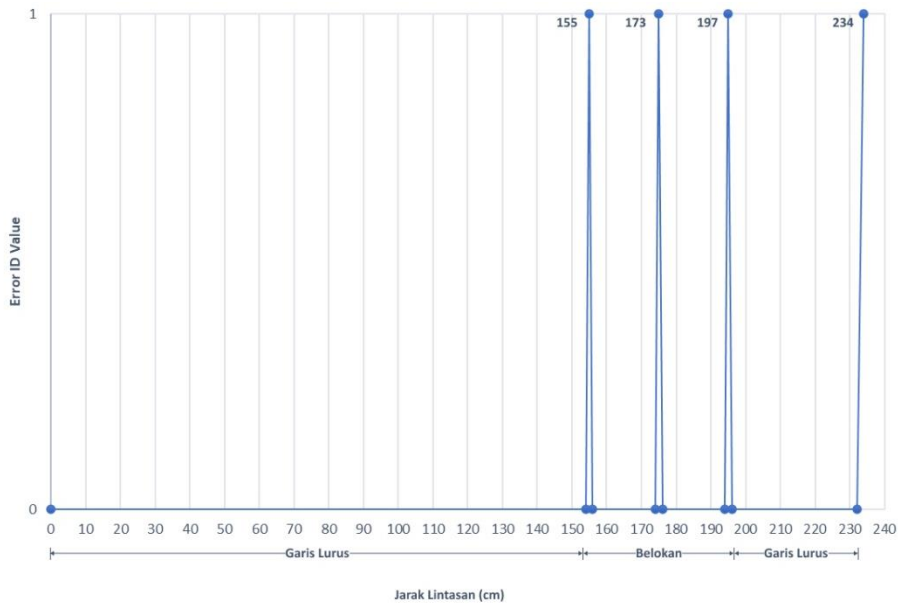




Gambar 4. 26 Jarak samping robot ke garis saat robot berada dalam lintasan



Gambar 4. 27 Jarak samping robot ke garis saat robot mengalami *error*

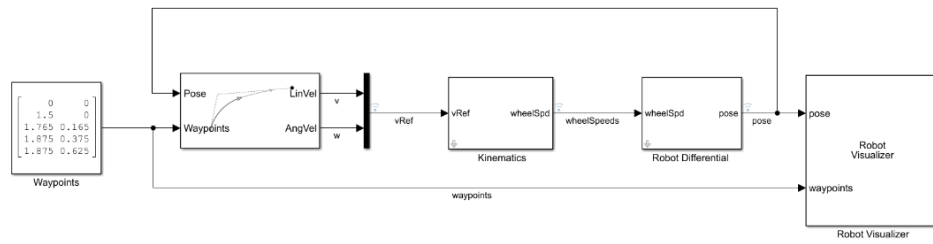


Gambar 4. 28 Nilai error robot pada saat melintasi garis lintasan

Dalam perjalanan melintasi garis lintasan, robot mengalami empat kali error atau kegagalan. Pada grafik di atas, dapat terlihat bahwa robot mengalami error pada jarak lintasan sekitar 155 cm, 173 cm, dan 197 cm. Namun, error pada jarak lintasan sekitar 234 cm menunjukkan bahwa robot telah berhenti karena mencapai ujung garis lintasan. Ini menggambarkan bahwa robot memiliki respons adaptif dalam menghadapi berbagai kendala di sepanjang jalurnya, dengan kemampuan untuk mengidentifikasi dan merespons kesalahan dalam pergerakannya.

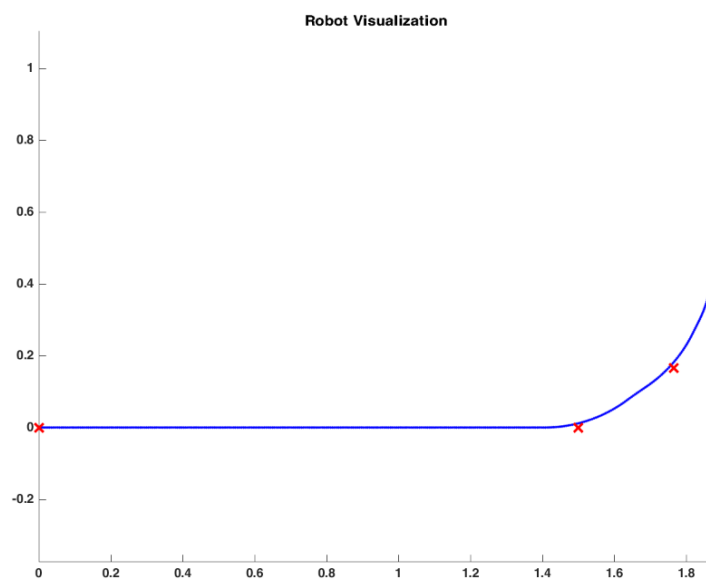
#### 4.6 Simulasi Robot Tempat Tidur Pasien Menggunakan *Simulink* Matlab

Dalam model Simulink komponen-komponen yang telah dirancang akan diwakili oleh blok-blok yang sesuai, dengan aliran data dan sinyal yang menggambarkan pergerakan robot dan respons terhadap lintasan. Hasil jarak tempuh yang dihasilkan akan ditampilkan sebagai output dari algoritma Line Follower.



Gambar 4. 29 Diagram blok simulink robot tempat tidur pasien

Gambar di atas adalah hasil dari simulasi keseluruhan pada mobil robot. Simulasi ini melibatkan kinematik mobil robot untuk mensimulasikan gerakan dan posisi robot saat bergerak. simulasi secara menggambarkan bagaimana mobil robot berinteraksi dengan bagaimana perubahannya dalam posisi terjadi seiring waktu. Hasil dari simulasi ini diwujudkan dalam bentuk grafik plot yang memvisualisasikan posisi aktual mobil robot sesuai dengan desain yang diterapkan pada pengambilan data atau nilai. Visualisasi grafik ini menggambarkan pergerakan robot dengan jelas, mengacu pada posisi yang sesuai dengan perencanaan yang telah diatur sebelumnya. Gambar hasil simulasi ini dapat dilihat pada gambar berikut



Gambar 4. 30 Hasil simulasi menggunakan simulink