

MOBILE MANIPULATOR ROBOT PADA KASUS

PEMINDAH BARANG

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh:

Ely Firman NPM: 1051806

Faradila Irianti NPM: 1051807

POLITEKNIK MANUFAKTUR NEGERI

BANGKA BELITUNG

TAHUN 2022

LEMBAR PENGESAHAN

**MOBILE MANIPULATOR ROBOT PADA KASUS
PEMINDAH BARANG**

Oleh:

Ely Firman/1051806

Faradila Irianti/1051807

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1

Pembimbing 2



Muhammad Iqbal Nugraha, M.Eng



Yudhi, M.T

Penguji 1

Penguji 2



Aan Febriansyah, M.T



Indra Dwisaputra, M.T

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa 1 : Ely Firman NIRM : 1051806

Nama Mahasiswa 2 : Faradila Irianti NIRM : 1051807

Dengan Judul : *Mobile Manipulator Robot* Pada Kasus Pemindah Barang

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 19 Januari 2022

Nama Mahasiswa

1. Ely Firman

2. Faradila Irianti

Tanda Tangan



ABSTRAK

Dunia industri sangat mengharapkan peningkatan kinerja dengan pemanfaatan dan kreasi dari teknologi terbaru. Robot lengan merupakan salah satu jenis robot yang sering digunakan untuk melakukan beragam tugas untuk menggantikan manusia di dunia industri. Dengan mengolaborasikan fungsi lengan dan roda, maka akan dihasilkan kemampuan untuk memindahkan barang yang lebih efektif dan efisien. Dengan sistem line tracing atau pendeteksi garis, mobile manipulator robot akan bergerak menelusuri garis yang telah ada untuk menuju ke tempat pengambilan benda, lalu lengan robot akan berfungsi untuk mengambil dan meletakkan kembali benda ke tempat tujuan. Hasil keluaran dari penelitian kali ini berupa mobile manipulator robot yang dapat menuju lokasi benda kerja berdasarkan lintasan garis berwarna hitam dengan latar putih. Bagian manipulator dapat melakukan satu kali siklus pengambilan benda kerja dan menempatkannya menuju satu tujuan yang telah ditetapkan sebelumnya dengan berat benda maksimum 200 gram.

Kata Kunci: *kinematika, mobile robot, manipulator robot*

ABSTRACT

The industrial world really expects an increase in performance by utilizing and creating the latest technology. Robot arm is one type of robot that is often used to perform various tasks to replace humans in the industrial world. By collaborating the function of the arm and wheel, it will produce the ability to move goods more effectively and efficiently. With a line tracing system or line detection, the mobile manipulator robot will move along the existing line to get to the place where the object is picked up, then the robot arm will function to pick up and put the object back to its destination. The output of this research is a mobile manipulator robot that can go to the location of the workpiece based on the trajectory of a black line on a white background. While the manipulator can perform one cycle of taking the workpiece and placing it towards a predetermined destination with maximum mass of the workpiece is 200 grams.

Keywords: kinematics, mobile robot, manipulator robot

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarakatuh

Segala puji bagi Allah SWT yang telah memberikan kami kemudahan sehingga kami bisa menyelesaikan Laporan Proyek Akhir yang berjudul “*Mobile Manipulator Robot Pada Kasus Pindah Barang*”. Shalawat serta salam juga tidak lupa kami curahkan untuk baginda Nabi Muhammad SAW yang akan memberikan syafaatnya di akhirat nanti.

Laporan Proyek Akhir ini dapat diselesaikan berkat adanya usaha dan kerja sama tim yang baik serta adanya bantuan dan dukungan dari berbagai pihak. Pada kesempatan ini penulis ingin mengucapkan terima kasih kepada pihak-pihak tersebut diantaranya:

1. Tuhan Yang Maha Esa atas segala rahmat, rezeki dan hidayah yang diberikan kepada penulis untuk menyelesaikan Proyek Akhir ini dengan baik dan tepat waktu.
2. Orang tua kami yang selalu memberikan dukungan dan doa yang sangat berharga.
3. Bapak I Made Andik Setiawan, M.Eng, Ph.D. selaku Direktur di Politeknik Manufaktur Negeri Bangka Belitung yang telah memberikan kemudahan bagi kami untuk menyelesaikan Proyek Akhir ini.
4. Bapak Muhammad Iqbal Nugraha, M.Eng selaku pembimbing 1 yang telah meluangkan waktu, tenaga dan pikiran untuk memberi pengarahan dalam penulisan Laporan Proyek Akhir.
5. Bapak Yudhi, M.T selaku pembimbing 2 yang telah memberikan saran dan solusi selama proses perencanaan dan pembuatan alat serta penyusunan laporan.
6. Seluruh staf pengajar di Politeknik Manufaktur Negeri Bangka Belitung.
7. Rekan- rekan mahasiswa Politeknik Manufaktur Negeri Bangka Belitung.
8. Pihak-pihak lain yang telah memberikan bantuan selama proses pembuatan Proyek Akhir yang tidak bisa disebutkan satu persatu.

Dalam pembuatan Laporan Proyek Akhir ini, penulis menyadari bahwa Laporan ini masih jauh dari kata sempurna, baik dari segi isi maupun penulisannya. Oleh karena itu penulis mengharapkan kritik dan saran yang bersifat membangun demi kebaikan penulis kedepannya.

Akhir kata Penulis ucapkan terima kasih, semoga laporan ini dapat berguna dalam menambah pengetahuan dan wawasan untuk penulis dan pembaca.



DAFTAR ISI

LEMBAR PENGESAHAN	ii
PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK	iv
ABSTRACT	v
Kata Pengantar	vi
Daftar Isi.....	viii
Daftar Tabel	xi
Daftar Gambar.....	xii
Daftar Lampiran	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	2
BAB II DASAR TEORI	3
2.1 <i>Mobile Manipulator Robot</i>	3
2.1.1 <i>Kinematics Mobile Manipulator</i>	3
2.1.2 <i>Kinematics Mobile</i>	5
BAB III METODE PELAKSANAAN	7
3.1 Studi Literatur	8
3.2 Perancangan Mekanik Robot	8
3.3 Pembuatan Mekanik Robot.....	9

3.4 Perancangan Elektrik Robot.....	10
3.5 Pembuatan Rangkaian Elektrik Robot	13
3.6 <i>Assembling</i> antara Mekanik dan Elektrik Robot.....	13
3.7 Pengujian Komponen Robot	13
3.8 Pembuatan Sistem Kendali Robot.....	14
3.9 Pengujian Sistem Kendali Keseluruhan	14
3.10 Analisis dan Pembuatan Laporan Akhir	15
BAB IV PEMBAHASAN.....	16
4.1 Deskripsi Alat	16
4.2 Perancangan dan Pembuatan Mekanik Robot.....	17
4.2.1 Pembuatan Robot <i>Mobile</i>	17
4.2.2 Pembuatan Robot Manipulator	19
4.3 Perancangan dan Pembuatan Elektrik Robot	20
4.4 Pemasangan Komponen Lainnya.....	22
4.4.1 Pemasangan Sensor Garis	22
4.4.2 Pemasangan LCD.....	24
4.4.3 Pemasangan Sensor Ultrasonik.....	24
4.5 Pengujian Komponen Robot	25
4.5.1 Blok Diagram.....	25
4.5.2 Pengujian Nilai PWM dan RPM.....	25
4.5.3 Pengujian Motor Servo	27
4.6 Pengujian Sistem Kendali	28
4.6.1 Pengujian Sistem Pengikut Garis (<i>Line Follower</i>).....	28
4.6.2 Pengujian Manipulator Lengan	30
BAB V KESIMPULAN DAN SARAN.....	35

5.1 Kesimpulan	35
5.2 Saran.....	35
Daftar Pustaka	37
Lampiran	39



DAFTAR TABEL

Tabel 4.1 Hasil Pengujian Motor Servo.....	27
Tabel 4.2 DH Parameter.....	30
Tabel 4.3 Hasil Forward Kinematik.....	32



DAFTAR GAMBAR

Gambar 2.1 <i>Mobile manipulator robot</i>	3
Gambar 3.1 <i>Flowchart</i> metode pelaksanaan	8
Gambar 3.2 Rancangan konstruksi <i>mobile manipulator robot</i> pada kasus pemindah barang	9
Gambar 3.3 Rancangan ukuran konstruksi <i>mobile manipulator robot</i> pada kasus pemindah barang	9
Gambar 3.4 Rancangan untuk rangkaian elektrikal	12
Gambar 3.5 Diagram alir sistem kerja <i>mobile manipulator robot</i> pada kasus pemindah barang	14
Gambar 4.1 Arena Kerja Robot	17
Gambar 4.2 Tampak samping mekanik <i>platform mobile robot</i>	18
Gambar 4.3 Tampak bawah mekanik <i>platform mobile robot</i>	18
Gambar 4.4 Tampak depan mekanik <i>platform mobile robot</i>	19
Gambar 4.5 Tampak belakang mekanik <i>platform mobile robot</i>	19
Gambar 4.6 Tampilan mekanik dari manipulator robot	20
Gambar 4.7 Perancangan <i>layout</i> sistem elektrikal pada papan PCB	21
Gambar 4.8 Tampak papan sirkuit setelah dicetak dipasangkan dengan beberapa komponen elektrik	21
Gambar 4.9 Rangkaian elektrikal secara keseluruhan	21
Gambar 4.10 Rangkaian skematik modul sensor garis	23
Gambar 4.11 Tata letak modul sensor garis	23
Gambar 4.12 Tampilan sensor garis pada <i>mobile robot</i>	23
Gambar 4.13 Pemasangan LCD	24
Gambar 4.14 Pemasangan sensor ultrasonik	24
Gambar 4.15 Blok diagram <i>mobile manipulator robot</i> pada kasus pemindah barang	25
Gambar 4.16 Grafik Pengujian PWM dan RPM	26

Gambar 4.17 Ilustrasi pendeteksian sensor terhadap garis	28
Gambar 4.18 Blok diagram kontrol sistem pengikut garis	29
Gambar 4.19 Sistem pendeteksian garis pada <i>line follower</i>	30
Gambar 4.20 Pemodelan Manipulator Robot.....	30



DAFTAR LAMPIRAN

Lampiran 1 Daftar Riwayat Hidup

Lampiran 2 Program Arduino Mega 2560

Lampiran 3 Pengujian *Duty Cycle* Pada Servo



BAB I

PENDAHULUAN

1.1 Latar Belakang

Penggunaan teknologi robotika menjadi hal yang sedang digalakkan pada dunia industri. Dengan menggunakan teknologi terbaru, dunia industri mengharapkan peningkatan nilai efisiensi dan efektifitas dalam pekerjaan. Sebelumnya Sudimanto dan Kevin[1] merancang sebuah robot pemindah barang dengan menggunakan konsep robot beroda dan menggunakan sebuah *forklift*, namun penelitian ini memiliki batasan dalam fleksibilitas pergerakan robot pada saat memindahkan barang. Sementara Angin dan kawan-kawan[2] melakukan perancangan robot lengan pemindah barang yang kemudian diketahui bahwa penelitian ini terbatas pada kondisi maksimum dari panjang lengan yang digunakan. Penggunaan *mobile manipulator robot* tak hanya menawarkan kemampuan memindahkan barang tapi juga peningkatan efektivitas kerja robot. *Mobile manipulator* itu sendiri merupakan suatu sistem robot manipulator yang terpasang pada *mobile robot*. Dibandingkan dengan menggunakan robot manipulator saja atau *mobile robot* saja, menggabungkan dua jenis robot ini akan menghasilkan *workspace* yang lebih besar. Selain itu, *mobile manipulator robot* memungkinkan untuk beradaptasi dengan perubahan lingkungan dan melakukan berbagai tugas. Oleh karena itu menurut Pegman[3], teknologi ini memiliki potensi besar dalam industri manufaktur.

Namun dengan menggabungkan dua jenis robot ini akan menimbulkan permasalahan baru. Pertama, penggabungan dua jenis robot ini akan menciptakan redundansi. Kedua, akan ada interaksi dinamis antara *platform mobile* dan manipulator. Ketiga, manipulator dan *platform mobile* memiliki *bandwidth* yang berbeda. *Platform mobile* biasanya memiliki respon dinamis yang lebih lambat daripada manipulator[4] sehingga diperlukan koordinasi gerakan yang tepat untuk melakukan sebuah tugas. Pada proyek akhir ini akan dilakukan pengendalian gerak

yang tepat untuk menjadikan robot *mobile* manipulator dapat mengerjakan tugas secara efektif dan efisien.

1.2 Rumusan Masalah

Adapun rumusan masalah yang akan dibahas pada proyek akhir ini adalah sebagai berikut:

1. Bagaimana sistem kinematika pada *mobile manipulator robot*.
2. Bagaimana agar robot dapat bergerak dan memindahkan barang dengan baik.

1.3 Batasan Masalah

Adapun batasan masalah dalam proyek akhir ini ialah sebagai berikut:

1. Robot bergerak berdasarkan lintasan yang berupa garis berwarna hitam dengan *background* putih.
2. Robot bekerja satu kali pengambilan benda dan menemukannya ke satu tempat tujuan.
3. Massa benda kerja maksimal tidak lebih dari 200 gram dengan dimensi maksimal 25 mm x 20 mm x 80 mm dan berbentuk balok.

1.4 Tujuan Penelitian

Adapun tujuan pembuatan proyek akhir ini adalah sebagai berikut:

1. Membuat *prototype* robot yang dapat memindahkan barang dari lokasi yang satu ke lokasi yang dituju sesuai dengan jalur yang telah ditentukan.
2. Merancang strategi kerja agar robot memenuhi tanggapan dan kinerja yang diinginkan.
3. Mempermudah pekerjaan dalam memindahkan barang di industri.
4. Sebagai sarana pembelajaran kendali robot untuk pengembangan berikutnya.

BAB II

DASAR TEORI

2.1 Mobile Manipulator Robot

Mobile manipulator robot merupakan penggabungan dari beberapa fungsi yang ada pada *mobile robot* dan fungsi yang ada pada robot *non mobile* atau manipulator[4]. Sehingga robot ini akan memiliki fungsi yang saling melengkapi, dimana robot dapat bergerak dari satu tempat ke tempat yang lain. Dengan demikian robot manipulator yang berfungsi untuk mengambil dan memindahkan barang akan dipasang diatas *mobile platform* yang dapat berpindah tempat sesuai dengan lintasan yang diberikan.



Gambar 2.1 *Mobile manipulator robot* [5]

2.1.1 Kinematics Mobile Manipulator

Kinematika di dunia robotika adalah suatu pernyataan yang berhubungan dengan deskripsi matematik geometri dari sebuah robot. Melalui persamaan kinematik ini dapat diketahui hubungan antara konsep geometri ruang sendi pada robot dengan konsep koordinat yang pada umumnya digunakan untuk menentukan posisi atau kedudukan dari sebuah objek. Model kinematik ini juga yang nantinya dapat menjadi dasar referensi masukkan yang akan digunakan untuk menggerakkan tiap-tiap aktuator agar dapat mencapai posisi yang dihendaki [6].

Pada penelitian sebelumnya, Syam dan Hair[7] menyelesaikan kerja sama antar dua buah *mobile manipulation robot* menggunakan metode kinematika gerak yang diselesaikan menggunakan persamaan kecepatan linear tiap rodanya. Sugar dan Kumar[8] mengembangkan *multiple mobile manipulator* berbasis kontrol kinematik. Mereka menganalisa berdasarkan pada tugas-tugas yang akan dilakukan oleh robot seperti menggenggam, manipulator dan mengangkat benda-benda besar yang fleksibel tanpa tujuan khusus. Melalui penelitiannya, kinematika dapat diselesaikan dengan menggunakan analisis lengan robot yang sesuai dan berhasil ditunjukkan untuk menjalankan kerja sama dua dan tiga *mobile manipulator robot*.

Perrier et al. [9] menerapkan *matriks homogenous* dan *dual quaternion* untuk menggambarkan masalah redundansi dalam kinematika. Mereka mempertimbangkan *global motion* dari *mobile manipulator* dari titik ke titik dan menghitung jalur yang memiliki kendala yang berbeda (*nonholonomic* dan *holonomic*). Karya-karya mereka berhasil digunakan dan diterapkan untuk menyelesaikan redundansi pada kinematika khususnya pada masalah *joint limitation*, kecepatan dan keterbatasan *steering* radius. Sementara Bayle dkk[10] memfokuskan pada manipulabilitas dalam analisis kinematika. Mereka menunjukkan bagaimana manipulabilitas dapat diperluas untuk mewakili metode operasional dalam konfigurasi sistem. Penelitian ini dapat digunakan untuk mengkonfigurasi ulang posisi lengan robot yang terpasang di bagian atas *platform* untuk memaksimalkan operasi *workspace* robot.

Randis[11] melakukan penelitian untuk mengetahui kinematika pada *excavator backhoe* yang merupakan salah satu jenis *mobile manipulator robot*. Adapun metode yang digunakan adalah metode analisa *Denevit-Hartenberg* (DH) pada kinematika dan *Jacobian Matriks* untuk persamaan gerak yang efisien. Representasi *Denevit-Hartenberg* prinsip dasarnya adalah melakukan transformasi koordinat antar dua *link* yang berdekatan. Representasi ini menyatakan hubungan antara dua *link* yang berhubungan pada DH parameter digunakan metode matriks matriks *homogeneous transformation* berordo 4×4 [12]. Melalui matriks ini

diketahui sistem koordinat dari sebuah link dengan link lain yang terhubung pada pangkalnya [13].

Tabel 2.1 Batasan Parameter DH Table

a_i	= Jarak dari	Z_i	Ke	Z_{i+1}	Di ukur sepanjang	X_i
α_i	= Sudut antara	Z_i	Ke	Z_{i+1}	Di ukur sekitar	X_i
d_i	= Jarak dari	X_{i-1}	Ke	X_i	Di ukur sepanjang	Z_i
θ_i	= Sudut antara	X_{i-1}	Dan	X_i	Di ukur sekitar	Z_i

Sementara itu, persamaan umum dari DH Parameter adalah [13]

$${}^{i-1}T_1 = \begin{pmatrix} \cos \theta_i & -\sin \theta_i & 0 & \alpha_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & \sin \alpha_{i-1} d_i \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & \cos \alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.1)$$

Berdasarkan Table DH Parameter didapatkan transformasi matriks dalam persamaan 2.2[14]. r merujuk pada hasil rotasi sementara P_x , P_y dan P_z merujuk pada posisi akhir dari *end-effector* robot.

$${}^0T_3 = {}^0T_1 = {}^1T_2 = {}^2T_3 = {}^3T_4 = \begin{pmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{32} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.2)$$

2.1.2 Kinematics Mobile

Mobile robot adalah jenis robot yang memiliki ciri khas dengan aktuator berupa roda ataupun sesuatu yang dapat berpindah tempat. Syam dan Hair [7] melakukan penelitian kerjasama antar dua buah *mobile manipulator robot* dengan menggunakan kinematika *mobile* menghasilkan nilai error yang kecil diantara kedua robotnya membuat *nonholonomic* yang bergerak bergantung pada jalur yang diraihnya. Sementara Julyus dan kawan-kawan [15] menggunakan robot beroda dengan metode *differential drive*, metode ini merupakan salah satu teknik pergerakan robot yang memanfaatkan perbedaan kecepatan dua roda dengan

penambahan satu buah roda *freewheeling* sebagai penyeimbang pada bagian depan. Kedua penelitian sebelumnya menggunakan pendekatan kinematik yang secara matematis didapatkan persamaan kecepatan linier (V_t) dan kecepatan angular (ω_t) dari *mobile robot* ini sebagai persamaan 2.3.

$$\begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} \quad (2.3)$$

V_R menunjukkan kecepatan terhadap roda kanan sementara V_L menunjukkan kecepatan roda kiri, dan L menunjukkan jarak antara kedua roda utama. Persamaan 2.3 menunjukkan keterkaitan antara kecepatan angular robot dengan kecepatan linier yang kemudian dapat di rumuskan menuju penggunaan kinematika *mobile robot*. Sehingga dapat diperoleh persamaan 2.4 yang menunjukkan kinematika balik dari *mobile robot* ini.

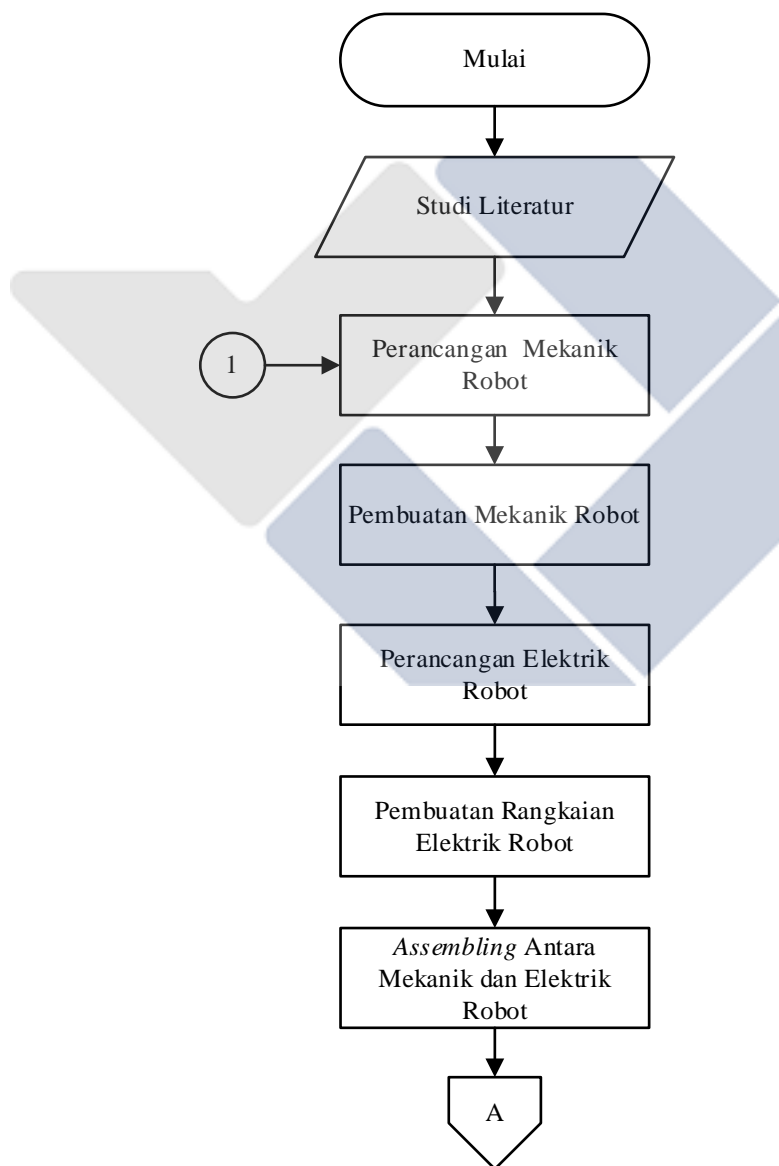
$$\begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} = \begin{bmatrix} 1 & \frac{L}{2} \\ 1 & -\frac{L}{2} \end{bmatrix} \begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix} \quad (2.4)$$

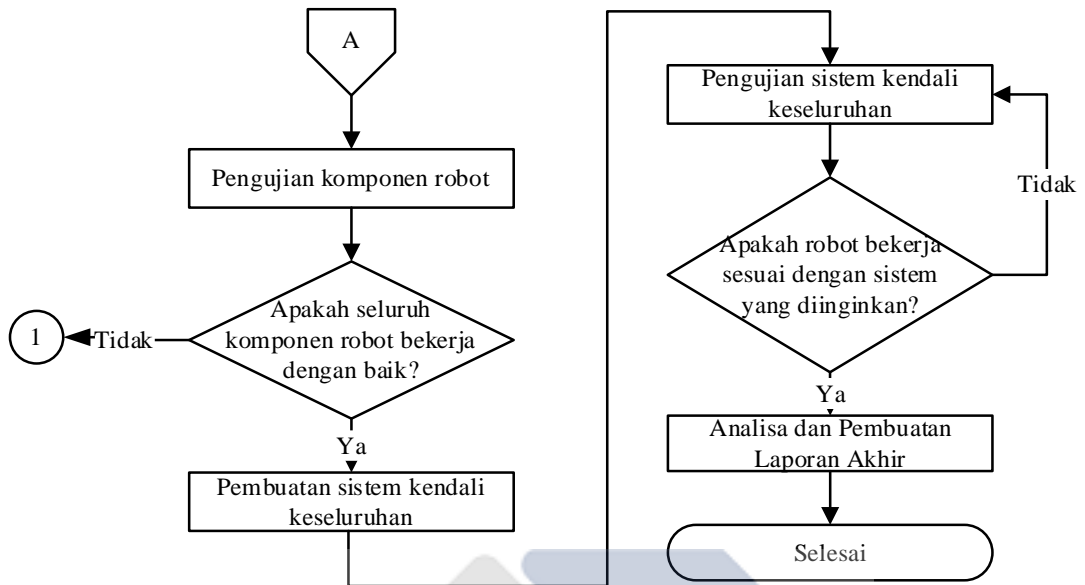
Saat kecepatan linier dan kecepatan angular diketahui, maka setiap sumbu kartesian dapat diketahui kecepatannya dengan cara memproyeksikan vektor kecepatan kepada sumbu kartesian tersebut [15].

BAB III

METODE PELAKSANAAN

Adapun metode pelaksanaan ini meliputi sebagai berikut: pemahaman literatur, perakitan *prototype* robot, memodelkan algoritma kendali dan sistem kontrol, melakukan simulasi, menganalisa dan memberi kesimpulan.





Gambar 3.1 *Flowchart* metode pelaksanaan

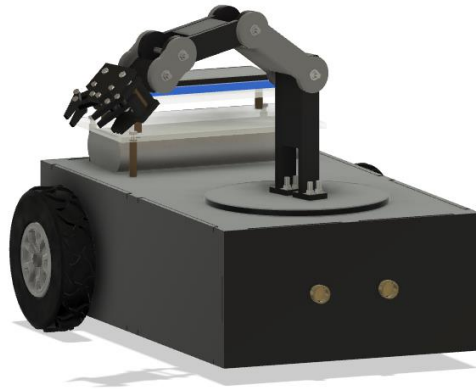
Penjelasan lebih lanjut dari *flowchart* di atas dapat dilihat pada langkah-langkah berikut ini:

3.1 Studi Literatur

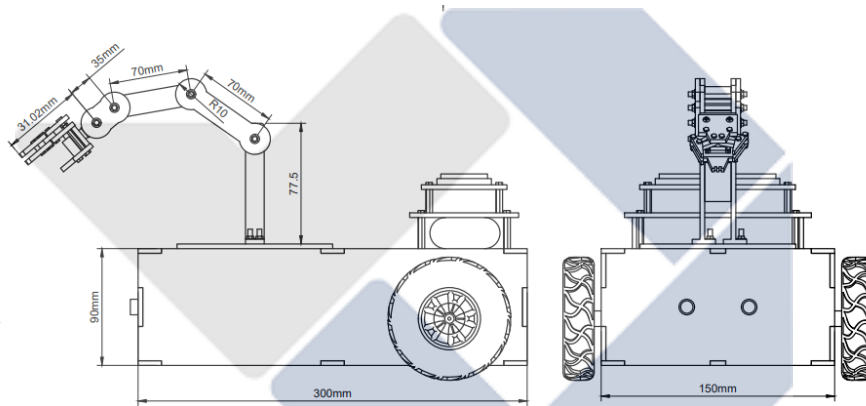
Studi literatur berisi tentang kajian dari beberapa acuan yang diperoleh baik berupa karya ilmiah, jurnal, buku, maupun bersumber dari internet yang berkaitan dengan robot ini sebagai penunjang untuk mempermudah dalam proses pengerjaan proyek akhir ini.

3.2 Perancangan Mekanik Robot

Perancangan mekanik atau konstruksi dari *Mobile Manipulator Robot* ini dibuat dengan *software Autodesk Fusion 360*. Rancangan konstruksi pada proyek akhir ini dalam bentuk *prototype* yang terdiri dari robot *mobile* dan robot manipulator. Rancangan mekanik dapat dilihat pada gambar 3.2 sementara untuk perancangan gambar beserta ukuran terdapat pada gambar 3.3 di bawah ini.



Gambar 3.2 Rancangan konstruksi *mobile manipulator robot* pada kasus pemindah barang



Gambar 3.3 Rancangan ukuran konstruksi *mobile manipulator robot* pada kasus pemindah barang

3.3 Pembuatan Mekanik Robot

Pembuatan kerangka mekanik dari robot *mobile* dan robot manipulator di pilih menggunakan bahan akrilik. Robot *mobile* merupakan *base* dari robot manipulator dengan panjang 30 cm, lebar 15 cm dan tinggi 9 cm. Di dalam robot *mobile* tersebut tempat untuk meletakkan komponen - komponen lain seperti Sensor Ultrasonik, Baterai, Motor DC, Arduino, dan komponen lainnya.

3.4 Perancangan Elektrik Robot

Perancangan elektrik robot dilakukan dengan menentukan komponen-komponen elektrik yang akan digunakan. Dalam proyek akhir ini terdapat dua buah jenis robot yang akan digabungkan menjadi satu. Dalam robot manipulator dengan 5 buah derajat kebebasan yang digunakan, dibutuhkan komponen yang dapat menggerakkan masing-masing sendi yang ada. Untuk itu digunakan 6 buah motor servo yang akan dipasang pada tiap-tiap sendi. Pada bagian *waist*, *shoulder*, *elbow* dan *pitch* digunakan motor servo jenis MG-996R, penggunaan motor servo jenis ini menyesuaikan dengan posisi dan kondisi pergerakan yang akan dilakukan oleh tiap-tiap sendi robot. Untuk bagian *waist* dan *shoulder* dibutuhkan motor servo dengan torsi yang cukup untuk menompang lengan robot dan beban yang ada di posisi berikutnya, sementara bagian *elbow* dan *pitch* juga membutuhkan torsi yang cukup untuk bergerak serta memiliki jangkauan pergerakan sebesar 180° . Pada bagian roll digunakan motor servo MG-90S yang memiliki pergerakan putaran hingga 360° , dan pada bagian *gripper* digunakan servo motor jenis SG-90S yang memiliki berat yang lebih ringan dibandingkan servo lainnya karena mengingat posisi peletakan motor servo ini dan juga tenaga yang digunakan hanya untuk mencengkam benda kerja sehingga torsi yang digunakan juga relatif lebih kecil dari motor servo lainnya. Untuk mengendalikan motor servo secara paralel digunakan servo *driver* jenis PCA 9685 [16].

Sementara itu untuk fungsi dari *mobile robot* yang akan bekerja dengan sistem pengikut garis hitam diatas garis putih, dibutuhkan komponen utama yaitu komponen yang dapat berfungsi untuk mengidentifikasi dan membedakan antara garis putih dan hitam, untuk itu diperlukan serangkaian komponen yang kemudian disebut sebagai sensor garis. Sensor garis yang lazim digunakan pada robot pengikut garis atau *line follower* adalah sensor seperti *optical LED* dan *photodiode*, pada proyek akhir ini digunakan modul sensor garis yang berisi 14 pasang sensor *optical LED* dan *photodiode* dan sebuah *multiplexer 16-channel* yang berfungsi untuk menghemat pin yang digunakan pada mikrokontroler. Hasil dari pembacaan garis ini akan diproses oleh mikrokontroler, adapun mikrokontroler yang digunakan pada proyek akhir ini adalah Arduino dengan jenis Mega 2560.

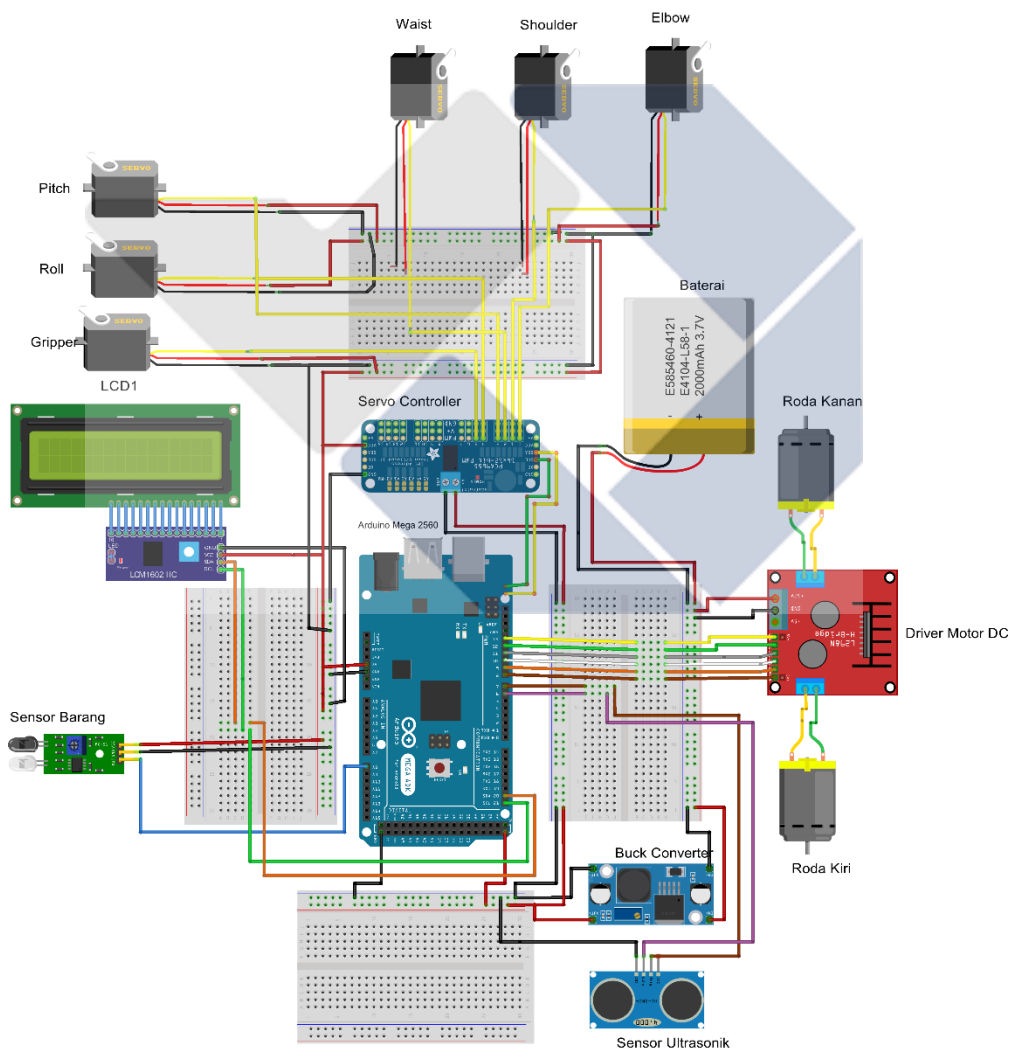
Pemilihan mikrokontroller jenis ini dimaksudkan untuk menyeimbangkan banyaknya masukan dan keluaran yang akan dilakukan oleh mikrokontroller untuk menjalankan keseluruhan sistem kendali pada robot. Mikrokontroller akan memberikan perintah untuk menggerakkan robot mobil melalui sebuah driver yang mengendalikan pergerakan dari roda. Jenis driver yang digunakan untuk menggerakkan roda adalah modul driver L298N, modul jenis ini memiliki dua buah keluaran yang membuat driver ini sesuai untuk digunakan pada proyek akhir ini. Driver motor ini akan mengontrol pergerakan dari dua buah motor DC yang menggerakkan roda kiri dan kanan dari robot mobil. Adapun jenis motor DC yang digunakan pada proyek akhir ini adalah jenis Motor DC PG-36, motor DC jenis ini digunakan karena memiliki maksimal torsi efektif sebesar ± 7.6 kgfcm yang sangat dibutuhkan agar robot mobil dapat bergerak dengan efektif walaupun memiliki beban manipulator yang cukup berat di atasnya.

Secara keseluruhan sistem *mobile manipulator robot* pada proyek akhir kali ini akan berhenti bergerak saat mendeteksi halangan (*obstacle*) ataupun meja kerja (*station*). Merujuk pada penelitian yang dilakukan oleh Putra dan Ridwan [17], penggunaan sensor ultrasonik jenis HC-04 yang berfungsi sebagai pengirim, penerima, dan pengontrol gelombang ultrasonik, dapat dimanfaatkan untuk mengukur jarak benda dari 2 cm - 4 m dengan akurasi 3 mm. Maka dari itu alat ini digunakan sebagai sensor jarak untuk memberikan informasi saat terdeteksi benda dengan jarak 5 cm maka robot akan berhenti. Selain itu, robot juga akan mencekam benda saat diujung lengan atau *end-effector* terdeteksi benda kerja, sehingga dibutuhkan sebuah sensor *Infra-red* yang berfungsi mendeteksi benda kerja.

Rangkaian elektrikal ini menggunakan baterai Li-Po 11.1 Volt sebagai sumber energi listrik. Sementara itu masukkan pada Arduino membutuhkan tegangan sebesar 12 Volt, dan masukkan pada *servo controller* membutuhkan tegangan maka dari itu digunakan dua buah *buck-boost converter* yang akan menurunkan tegangan ke 5 volt dan menaikkan tegangan ke 12 volt. Penggunaan *buck-boost converter* ini juga ditujukan sebagai pengaman dan penstabil tegangan agar saat tegangan yang masuk tidak stabil (dapat berlebih atau kekurangan) maka

komponen yang dipasang setelah *buck-boost converter* tidak rusak akibat *short circuit*. Sebagai *interface* dari informasi yang akan menampilkan perintah digunakan LCD dengan ukuran 16x2.

Setelah semua komponen yang dibutuhkan diketahui maka akan dilakukan perancangan tata letak rangkaian menggunakan *software Eagle* yang kemudian akan dicetak pada papan sirkuit ditahap berikutnya. Gambar 3.4 menunjukkan hasil perancangan dari sistem elektrikal yang kemudian akan dicetak dan dilakukan pemasangan komponen dan *wiring*.



Gambar 3.4 Rancangan untuk rangkaian elektrikal

3.5 Pembuatan Rangkaian Elektrik Robot

Perakitan atau pembuatan rangkaian elektrik robot dilakukan dengan melakukan *wiring* dan pemasangan komponen pada *Project Circuit Board* (PCB) yang telah dirancang pada tahapan sebelumnya. Rancangan *layout* elektrik yang di buat pada *software Eagle* akan dicetak pada *transfer paper*. Lalu untuk mencetak *layout* pada permukaan tembaga PCB, transfer paper dengan sisi *glossy* diposisikan menghadap langsung ke permukaan PCB lalu digunakan teknik pencetakan manual dengan menggunakan setrika. Setelah seluruh tinta menempel pada permukaan PCB, dilakukan proses pelarutan tembaga (*etching*) yang tidak diinginkan menggunakan larutan kimia besi klorida (FeCl_3). Setelah tembaga yang tidak diinginkan selesai dilarutkan, papan sirkuit dibersihkan dari larutan kimia dan siap untuk digunakan pada tahapan berikutnya. PCB yang telah bersih, akan dilakukan pengeboran lubang komponen yang akan dipasang. Setelah lubang di bor, PCB harus kembali dibersihkan dari sisa-sisa debu sehabis pengeboran, hal ini harus diperhatikan dengan seksama agar saat melakukan *wiring* ataupun penyolderan tidak menyebabkan korsleting pada papan. Proses berikutnya yaitu pemasangan komponen dan penyolderan pada papan sirkuit, proses ini dilakukan dengan perlahan agar didapatkan hasil yang maksimal. Setelah komponen terpasang, dilakukan pengujian *wiring*, hal ini dilakukan sebagai upaya pencegahan terjadinya *short circuit* pada papan sirkuit.

3.6 Assembling antara Mekanik dan Elektrik Robot

Assembling sistem mekanik dan elektrik robot dilakukan dengan menggabungkan antara robot *mobile* dan robot manipulator menjadi satu lalu diikuti pemasangan sistem elektrik yang di telah desain pada bagian robot *mobile*.

3.7 Pengujian Komponen Robot

Pengujian komponen yang dilakukan untuk mengetahui komponen-komponen yang akan digunakan dapat bekerja sesuai dengan fungsinya atau tidak.

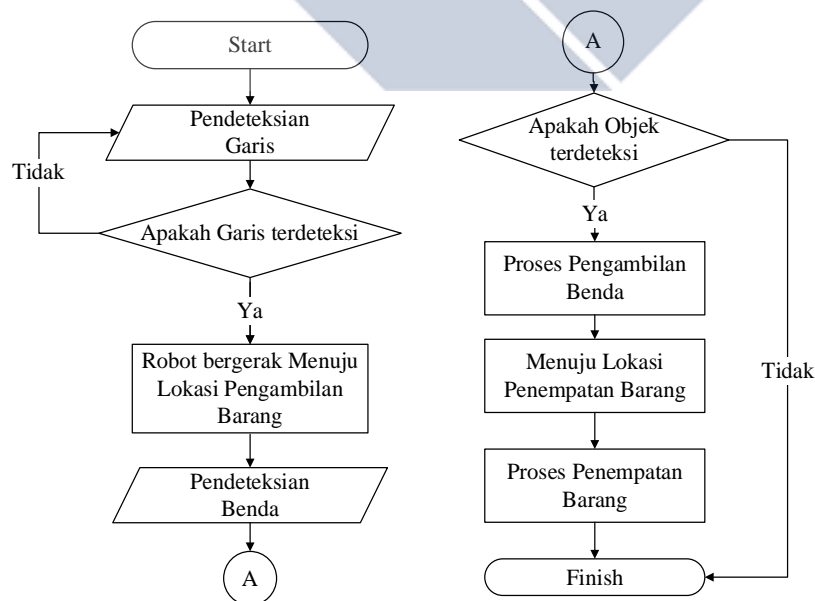
Pengujian ini dilakukan untuk memaksimalkan fungsi kerja dari masing-masing komponen. Adapun uji coba komponen-komponen elektrik yang akan dilakukan seperti uji coba Arduino Mega 2560, uji coba Sensor, uji coba Motor DC, dan uji coba Motor Servo.

3.8 Pembuatan Sistem Kendali Robot

Pembuatan sistem kendali ini dilakukan dengan membuat program melalui *software* Arduino IDE untuk menjalankan keseluruhan robot sesuai dengan instruksi yang dirancang sebelumnya. Pada sistem kendali bagian lengan, akan dibuat sistem untuk menentukan titik koordinat sumbu X, Y dan Z dengan memasukkan besaran sudut pada masing - masing joint. Sementara pada bagian *mobile* robot dibuat program untuk berjalan mengikuti garis.

3.9 Pengujian Sistem Kendali Keseluruhan

Pengujian ini ditujukan untuk menguji keberhasilan dari keseluruhan sistem *mobile manipulator robot* saat dijalankan. Diagram alir pada Gambar 3.5 menunjukkan alur kerja sistem yang diinginkan.



Gambar 3.5 Diagram alir sistem kerja *mobile manipulator robot* pada kasus pemindah barang

3.10 Analisis dan Pembuatan Laporan Akhir

Pada tahapan ini dilakukan analisis data pada hasil uji coba yang telah dilakukan yang bertujuan untuk mengetahui kekurangan terhadap alat yang telah dibuat baik dari segi mekanik, rangkaian elektrik serta keseluruhan yang telah dibuat. Kemudian dituliskan dalam laporan akhir dengan tujuan merangkum keseluruhan yang berhubungan dengan proyek akhir sekaligus memberikan informasi yang didapat pada proyek akhir yang telah dibuat.



BAB IV

PEMBAHASAN

Pada bab pembahasan ini memaparkan proses dan hasil pengerjaan proyek akhir berdasarkan metode yang telah diuraikan pada bab sebelumnya.

4.1 Deskripsi Alat

Mobile manipulator robot ini berjalan mengikuti garis hitam di atas latar putih. Robot akan berhenti ketika sudah berada di *station* atau tempat pengambilan benda. Saat berada di *station*, bagian manipulator robot akan mengambil benda. Benda akan tetap dipegang oleh manipulator dan dibawa menuju ke *station* atau tempat peletakan benda. Adapun kemampuan robot dalam membawa benda kerja diketahui bahwa benda berbentuk balok dengan dimensi 20 mm x 25 mm x 80 mm merupakan benda kerja yang paling maksimal dapat dibawa oleh robot. Hal ini menyesuaikan dengan kemampuan torsi motor yang digunakan untuk mencapit benda yaitu sebesar 2.5 kg-cm dan torsi dari servo lainnya yang akan menumpu keseluruhan berat beban lengan. Dimensi benda kerja disesuaikan dengan ukuran lebar *end-effector* yakni sebesar 4cm, disesuaikan dengan tinggi sensor barang yaitu 4,5 cm, dan dengan memperhatikan banyaknya permukaan benda kerja yang tercengkam oleh *gripper*. Semakin banyak permukaan benda kerja yang berkontak langsung oleh *gripper* maka semakin baik kemampuan *gripper* untuk mempertahankan posisi dan keadaan benda kerja.

Arena kerja yang akan dilalui oleh robot adalah arena kerja dengan garis berwarna hitam dan latar putih, garis hitam yang dibuat merepresentasikan jalur yang digunakan di industri. Adapun lebar garis yang digunakan sebesar $\frac{3}{4}$ inch yang disesuaikan dengan lebar dari sensor garis yang digunakan yaitu 2 cm. Selain itu arena yang digunakan sengaja dibuat memiliki dua buah tikungan, hal ini dibuat dengan tujuan menguji konfigurasi pergerakan robot dapatkah bergerak sesuai dengan algoritma yang diberikan saat menemui belokkan ataupun bentuk garis yang baru. Diawal arena dibentuk garis hitam memanjang dan lebih tebal dari garis

lainnya, bentuk garis seperti ini dibuat sebagai awalan *start* dan sebagai tempat sensor garis untuk mengidentifikasi warna hitam dan warna putih (kalibrasi) di tahap awal. Sementara garis hitam memanjang pada *station* tempat pengambilan benda digunakan sebagai indikator bahwa robot harus berhenti saat semua sensor mendeteksi garis hitam dan memulai proses pengambilan benda. Berikutnya gambar 4.1 menunjukkan ilustrasi arena kerja yang dilalui oleh robot.



Gambar 4.1 Arena Kerja Robot

4.2 Perancangan dan Pembuatan Mekanik Robot

Pada perencanaan pembuatan mekanik robot dibagi menjadi tiga tahapan. Tahapan pertama adalah perencanaan *base* robot yaitu robot *mobile*. Tahapan yang kedua merupakan perencanaan robot manipulator atau robot lengan. Tahapan yang ketiga adalah pemasangan komponen dan alat lainnya seperti komponen kontrol elektrik, baterai, sensor, indikator (LCD) dan komponen/aksesoris yang dianggap penting lainnya. Berikutnya akan dibahas tahapan yang dilakukan dalam proses pembuatan mekanik *mobile manipulator robot* pada kasus pemindahan barang:

4.2.1 Pembuatan Robot *Mobile*

Berdasarkan judul yang diajukan pada proyek akhir ini, adapun yang menjadi *platform* atau base dalam mobilitas robot adalah jenis robot *mobile*. Pembuatan *base* robot menggunakan perhitungan dan pertimbangan agar robot dapat melaksanakan instruksi sesuai dengan yang diinginkan. Adapun beberapa pertimbangan yang harus diperhatikan adalah beban robot secara keseluruhan, kemampuan robot untuk dapat bergerak bebas, hingga kecepatan pergerakan robot

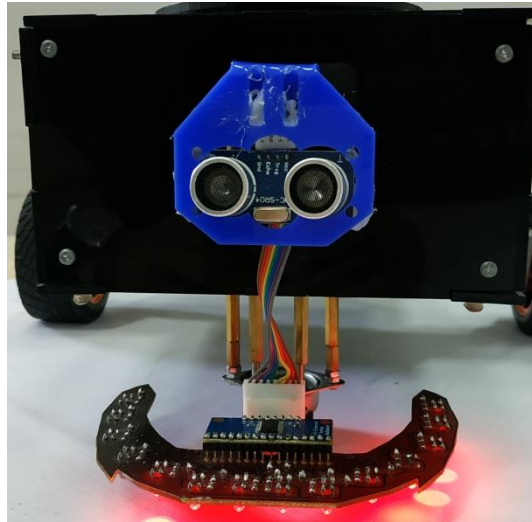
nantinya. Hal lainnya yang perlu diperhatikan seperti pemilihan roda, yang dalam hal ini dimulai dari jenis bahan dan ukuran roda, serta kemampuan torsi motor dc untuk menggerakkan *platform mobile*. Pada bagian *mobile robot* ini juga dirancang untuk meletakkan komponen-komponen seperti Sensor Garis, Motor DC, rangkaian elektrik dan lainnya. Gambar 4.2 menunjukkan bentuk mekanik dari *mobile robot* dari tampak samping. Sementara Gambar 4.3 menunjukkan tampak bawah *mobile robot* yang terdapat sensor garis dan roda *freewheeling*. Gambar 4.4 menampilkan *mobile robot* dari tampak depan. Dan Gambar 4.5 menampilkan tampak belakang dari *platform mobile* yang terdapat saklar untuk mengaktifkan dan mematikan sistem robot.



Gambar 4.2 Tampak samping mekanik *platform mobile robot*



Gambar 4.3 Tampak bawah mekanik *platform mobile robot*



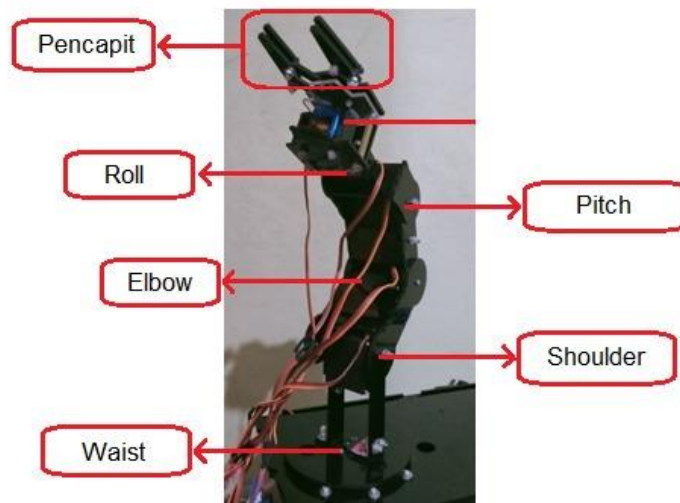
Gambar 4.4 Tampak depan mekanik *platform mobile robot*



Gambar 4.5 Tampak belakang mekanik *platform mobile robot*

4.2.2 Pembuatan Robot Manipulator

Pembuatan manipulator atau robot lengan ini harus dilakukan dengan teliti karena akan mempengaruhi keakuratan pergerakan dari robot. Robot lengan juga harus dapat menopang beberapa motor servo yang digunakan. Pada robot manipulator ini terdapat enam buah motor servo yaitu tiga buah servo *joint* yaitu *waist*, *shoulder*, *elbow* dan tiga buah servo pada *end effector* yaitu *pitch*, *roll*, dan *gripper*. Penggunaan motor servo disesuaikan dengan fungsi yang akan dijalankan oleh masing-masing servo, maka dari itu digunakan tiga buah jenis servo yang berbeda yaitu jenis MG996R pada *waist*, *shoulder*, *elbow* dan *pitch*, jenis MG90S pada *roll* dan jenis SG90 pada *gripper*. Gambar 4.6 menunjukkan tampilan mekanik dari manipulator robot.

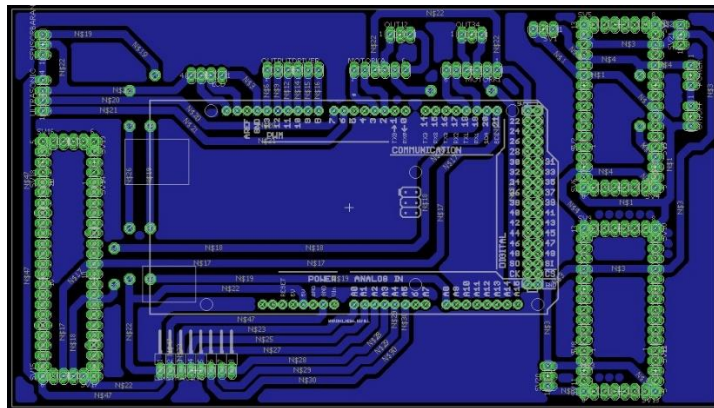


Gambar 4.6 Tampilan mekanik dari manipulator robot

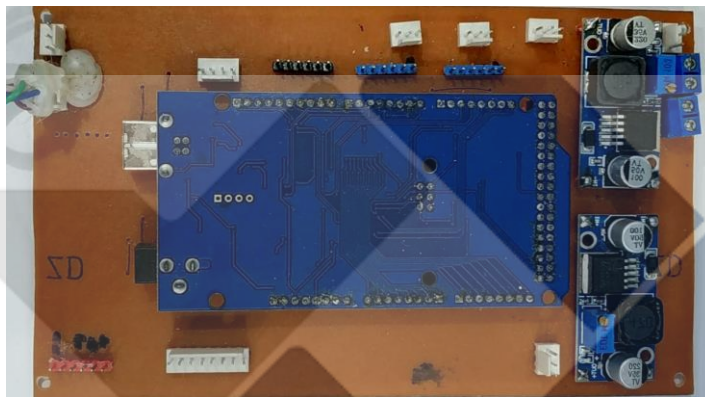
4.3 Perancangan dan Pembuatan Elektrik Robot

Sistem elektrikal pada robot dirancang menggunakan *software Eagle*, yang kemudian akan diimplementasikan pada PCB untuk dipasang pada *mobile robot*. Komponen yang disusun pada PCB antara lain *Arduino Mega 2560*, *Servo Controller PCA9685* dan *buck converter*. Rancangan *layout* elektrikal robot dapat dilihat pada gambar 4.7.

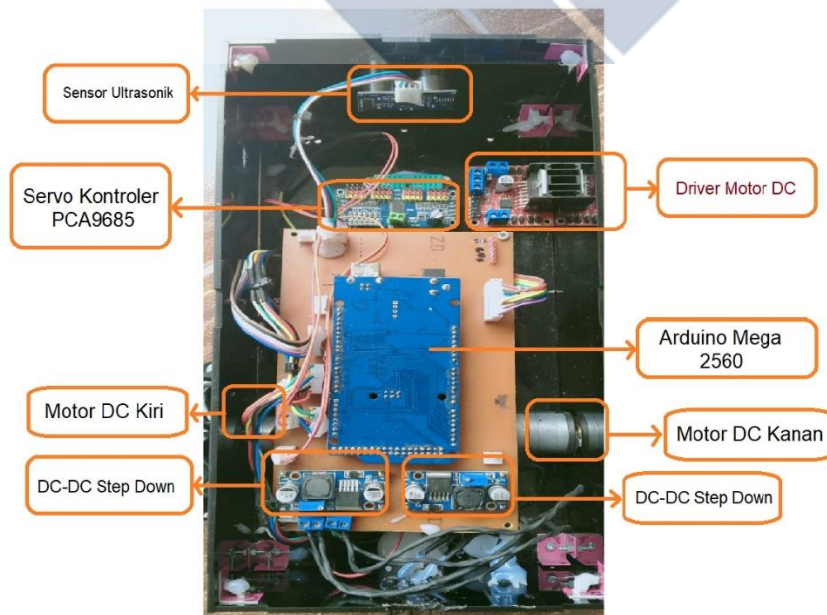
Setelah PCB siap digunakan, maka dilanjutkan dengan perakitan antar komponen hingga menjadi satu sistem elektrikal yang siap untuk digunakan. Perakitan tiap-tiap komponen ini dilakukan dengan perlahan dan memperhatikan rancangan sistemnya. Gambar 4.8 memperlihatkan tampak dari papan sirkuit setelah dirakit dengan beberapa komponen. Disetiap komponen yang dipasang perlu dilakukan pemeriksaan kembali antar koneksi komponennya, hal ini akan memicu terjadinya hubungan singkat antar komponen atau *short circuit*. Akibat dari *short circuit* ini akan berisiko merusak komponen, sehingga komponen tidak berfungsi dengan maksimal, bahkan hingga tidak dapat digunakan lagi. Perakitan elektrikal robot secara keseluruhan dapat dilihat pada gambar 4.9.



Gambar 4.7 Perancangan *layout* sistem elektrikal pada papan PCB



Gambar 4.8 Tampak papan sirkuit setelah dicetak dipasangkan dengan beberapa komponen elektrik



Gambar 4.9 Rangkaian elektrikal secara keseluruhan

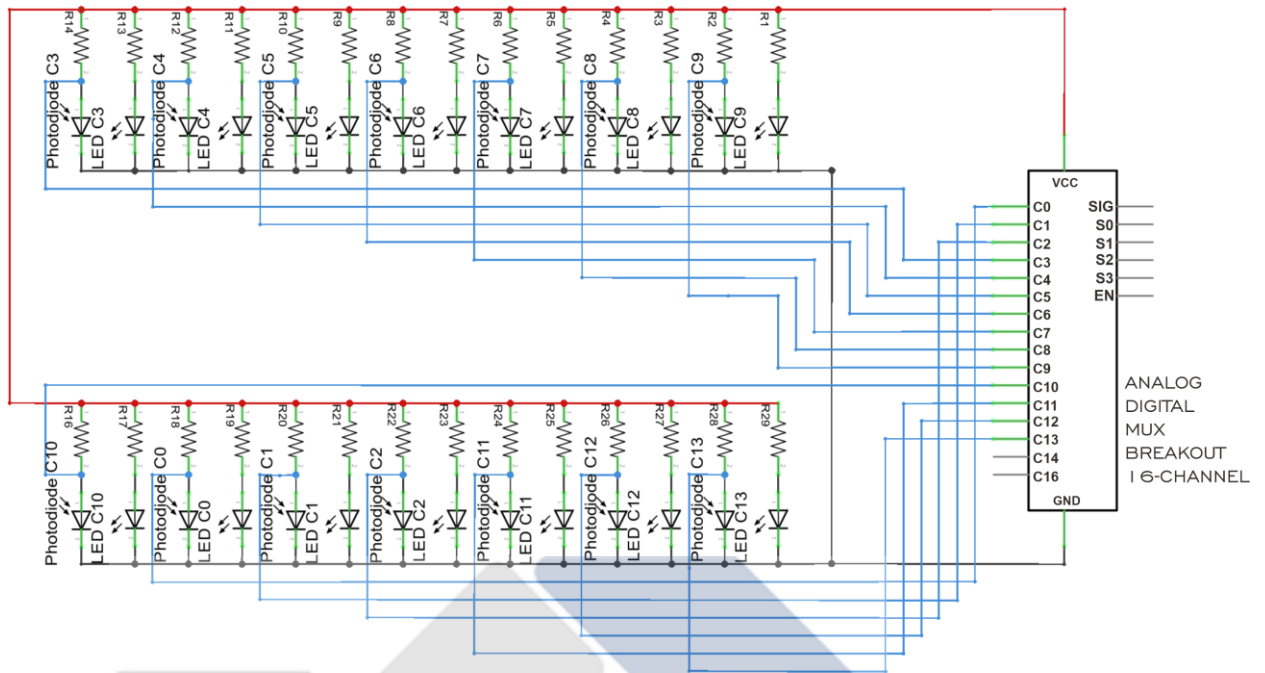
4.4 Pemasangan Komponen Lainnya

4.4.1 Pemasangan Sensor Garis

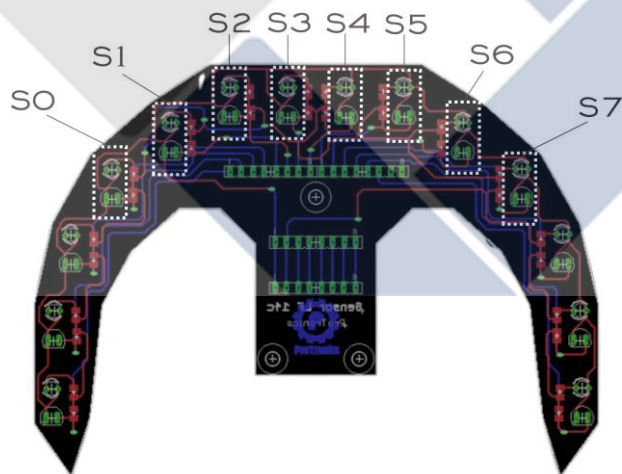
Sensor untuk pendeteksian garis diletakkan di bagian bawah robot. Jarak antara garis dan sensor perlu diperhatikan agar mendapatkan hasil yang optimal dari pembacaan sensor. Jarak optimal antara garis dan sensor minimal 0.5 cm dan maksimal sejauh 1 cm. Ketika sensor diletakkan lebih dari 1 cm hasil pembacaan sensor menunjukkan nilai yang tidak akurat. Nilai sensor yang terbaca akan banyak dipengaruhi oleh cahaya lingkungan sekitar, hal ini dapat menyebabkan pergerakan robot tidak terkendali. Sementara saat sensor diletakkan terlalu dekat dengan garis, nilai yang dihasilkan sulit dibedakan antara garis hitam dan latar putih yang ada. Selain itu penempatan sensor garis yang terlalu dekat akan berisiko untuk bersentuhan langsung dengan lintasan sehingga dapat menyebabkan kerusakan pada sensor.

Adapun sensor garis yang digunakan berupa sebuah modul sensor yang berisi 14 pasang sensor *optical* LED dan *photodiode*, dan sebuah multiplexer *16-channel*. Gambar 4.10 menunjukkan rangkaian skematik dari modul sensor garis yang digunakan, melalui rangkaian skematik ini dapat dilihat bahwa masing-masing *channel* C0-C13 pada multiplexer tersambung dengan sebuah sensor *photodiode* yang berfungsi untuk menangkap banyaknya cahaya yang dipantulkan oleh media pemantul (dalam hal ini garis hitam atau latar putih) yang terkena pancaran cahaya dari *optical* LED. Dari rangkaian skematik ini pula kemudian dibuat sebuah modul dengan tata letak yang baik pada sebuah papan sirkuit seperti yang ditunjukkan pada gambar 4.11.

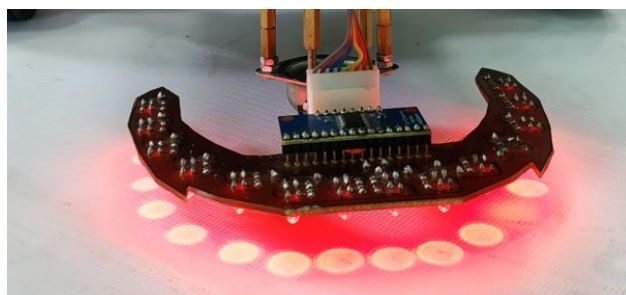
Namun pada proyek akhir kali ini, sensor yang digunakan hanya sebanyak 8 pasang sensor *optical* LED dan *photodiode* yang diinisialisasikan dengan kode huruf S0-S7 seperti yang tertera pada gambar 4.11. Sensor S0-S7 akan menggunakan hasil pembacaan dari *photodiode* hasil keluaran dari channel C3 sampai C10 pada multiplexer. Penggunaan 8 buah sensor ini ditujukan agar robot dapat mendeteksi garis dengan bentuk lintasan yang lebih luas dan bervariasi. Sehingga pergerakan robot ketika mendeteksi garis menjadi lebih halus. Gambar 4.12 menunjukkan saat modul sensor garis aktif dan siap untuk digunakan.



Gambar 4.10 Rangkaian skematik modul sensor garis



Gambar 4.11 Tata letak modul sensor garis



Gambar 4.12 Tampilan sensor garis pada *mobile robot*

4.4.2 Pemasangan LCD

Pemasangan LCD diletakkan di bagian atas robot. LCD dipasang dengan tujuan sebagai *interface* dari informasi yang akan disampaikan, seperti informasi perintah saat melakukan kalibrasi data sensor garis. LCD yang digunakan merupakan LCD dengan ukuran 16x4.



Gambar 4.13 Pemasangan LCD

4.4.3 Pemasangan Sensor Ultrasonik

Sensor ultrasonik dipasang pada bagian depan robot yang tujuannya menyesuaikan dengan fungsi dari sensor ini. Sensor ultrasonik digunakan untuk mendeteksi *obstacle* atau *station* pada jarak tertentu agar robot tidak menabrak *obstacle* maupun *station* tersebut.

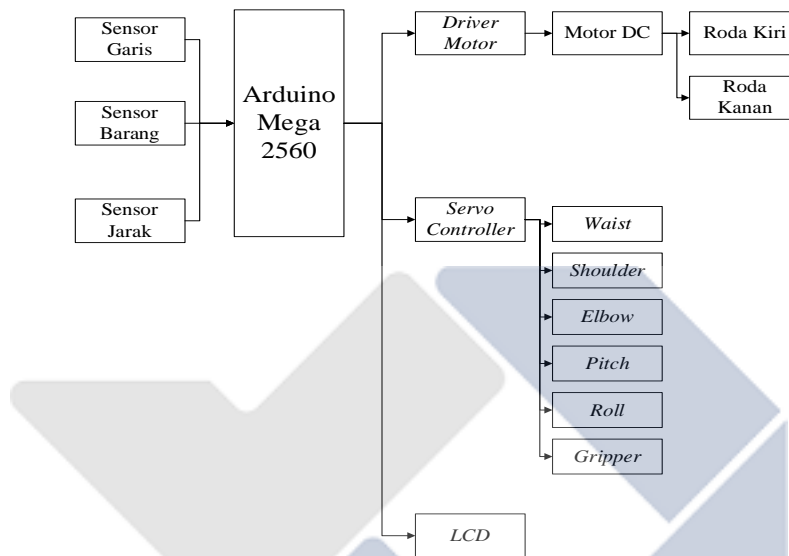


Gambar 4.14 Pemasangan sensor ultrasonik

4.5 Pengujian Komponen Robot

4.5.1 Blok Diagram

Untuk menggambarkan sistem kerja dari proyek akhir ini dibuat blok diagram sistem kerja dari *mobile manipulator robot* pada kasus pemindah barang dan ditunjukkan pada gambar 4.15 berikut:



Gambar 4.15 Blok diagram *mobile manipulator robot* pada kasus pemindah barang

4.5.2 Pengujian Nilai PWM dan RPM

Pengujian nilai PWM dilakukan untuk mengetahui karakteristik dari RPM masing-masing motor DC pada *mobile robot*. Melalui pengujian ini didapatkan nilai minimum dan maksimum PWM yang dapat digunakan dalam pengaturan kecepatan roda *mobile robot*. Adapun nilai PWM dan RPM di tampilkan dalam grafik pada gambar 4.16. Grafik pada gambar 4.16 ini merupakan hasil pengujian pergerakan motor DC saat kondisi tanpa beban. Pengukuran RPM pada motor DC diukur menggunakan alat ukur *tacho meter*.



Gambar 4.16 Grafik Pengujian PWM dan RPM

Berdasarkan grafik pada gambar 4.1 di atas, diketahui bahwa motor DC mulai bergerak saat diberikan PWM sebesar 15 dan RPM yang terukur terus meningkat seiring peningkatan PWM yang diberikan. Sehingga dapat disimpulkan bahwa PWM 15 merupakan RPM minimum dari motor DC, sedangkan pada PWM maksimum yaitu sebesar 255 pulsa terukur 625 RPM. Nilai yang RPM yang terukur antara motor kanan dan motor kiri dapat dikatakan linear, walaupun terdapat fluktuasi pada saat PWM 180-210. Pada PWM 180 sampai dengan PWM 210 terjadi ketidakstabilan antara motor DC kanan dan motor DC kiri, hal itu kemungkinan terjadi karena faktor kesalahan pada saat mengukur komponen, dapat juga dipengaruhi oleh alat ukur yang kurang akurat. Daya dari baterai yang digunakan juga mempengaruhi kecepatan motor DC, sehingga nilai RPM yang terukur dapat menjadi menurun. Namun, saat diketahui terdapat fluktuasi pada PWM tertentu, dilakukan pengujian berulang kali agar data yang ditunjukkan merupakan nilai yang tepat, tapi nilai yang terukur tetap menunjukkan hasil yang sama seperti grafik yang tertara pada gambar 4.16 sehingga dapat disimpulkan bahwa motor kanan memiliki karakteristik seperti grafik tersebut.

Data ini kemudian akan digunakan untuk pemberian sinyal PWM pada pengaturan kecepatan di *mobile robot*. Nilai PWM yang mengalami fluktuasi akan lebih dihindari pemakaiannya saat menginginkan gerakan lurus, dikarenakan dapat menyebabkan ketidak seimbangan pada pergerakan robot. Saat nilai yang RPM

yang digunakan antara motor kanan dan kiri berbeda, hal ini akan membuat robot melakukan konfigurasi gerakan berbelok ke satu arah yang RPM-nya lebih tinggi.

4.5.3 Pengujian Motor Servo

Pengujian motor servo dilakukan untuk mengetahui jangkauan yang dimiliki oleh masing-masing servo motor yang dimiliki oleh manipulator lengan. Tabel 4.1 menunjukkan sudut yang dapat diraih oleh tiap motor servo beserta nilai *duty cycle* yang akan menjadi rujukan pulsa masukkan pada pergerakan manipulator robot.

Tabel 4.1 Hasil Pengujian Motor Servo

Posisi Servo	<i>Duty Cycle</i> Minimum (%)	<i>Duty Cycle</i> Maksimum (%)	Jangkauan Sudut (derajat)
Waist	2,4	12,2	0-180
Shoulder	2,4	11,0	0-180
Elbow	2,3	11,4	0-180
Pitch	2,9	11,8	0-180
Roll	5,5	14,6	0-360
Gripper	2,5	11,8	0-180

Berdasarkan hasil pengujian pada tabel 4.1 diketahui nilai-nilai minimum dan maksimum *duty cycle* dari setiap servo berbeda. Dengan jangkauan yang sama misalnya pada *waist* dan *shoulder* yaitu 0° - 180° lebar pulsa aktif yang digunakan ternyata berbeda. Hal inilah yang mendasarkan perlu diadakannya pengujian servo yang digunakan, dengan tujuan mengetahui setiap kondisi yang dimiliki oleh masing-masing servo. Modulasi lebar pulsa atau PWM dapat diperoleh dengan bantuan sebuah gelombang kotak atau siklus kerja (*duty cycle*) ini. Pengaturan lebar pulsa inilah yang kemudian akan digunakan sebagai kendali pergerakan lengan robot. Dengan mengetahui lebar pulsa minimum dan maksimum yang dibutuhkan untuk mengaktifkan servo, maka akan lebih mudah untuk menentukan lebar pulsa

membuat servo bergerak ke arah yang diinginkan. Adapun *duty cycle* yang terdapat pada tabel 4.1 dihitung menggunakan rumus:

$$Duty\ Cycle_{minimum} = \frac{PWM\ Minimum}{Pulsa\ Maksimal} \times 100\% \quad (4.1)$$

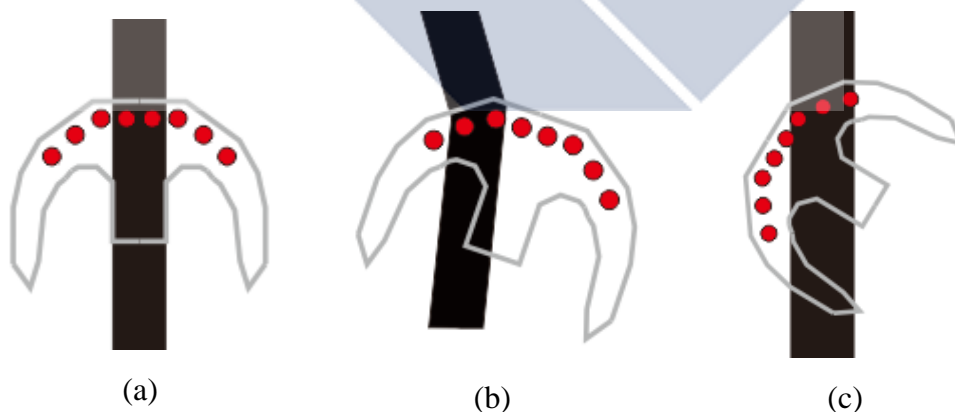
$$Duty\ Cycle_{maksimal} = \frac{PWM\ Maksimal}{Pulsa\ Maksimal} \times 100\% \quad (4.2)$$

Tiap-tiap perubahan sudut yang akan dilakukan oleh servo akan berpengaruh pada jumlah pulsa yang akan diberikan saat menentukan posisi akhir dari manipulator robot.

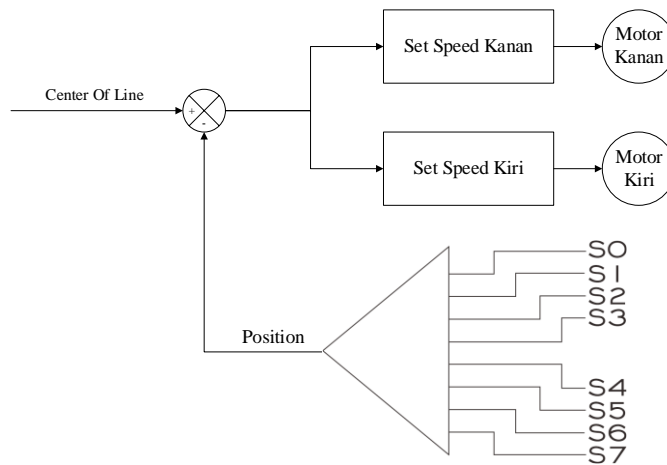
4.6 Pengujian Sistem Kendali

4.6.1 Pengujian Sistem Pengikut Garis (*Line Follower*)

Sistem pengikut garis diawali dengan kalibrasi data analog yang didapatkan dari pembacaan garis hitam dan latar putih. Ketika robot diaktifkan, LCD akan menampilkan perintah untuk melakukan kalibrasi. Kemudian data analog yang di dapatkan akan dirubah menjadi data *biner* dengan tujuan kemudahan saat melakukan logika pemrograman.



Gambar 4.17 Ilustrasi pendeteksian sensor terhadap garis (a) untuk gerakan lurus (b) arah pergerakan ke kiri (c) arah pergerakan ke kanan



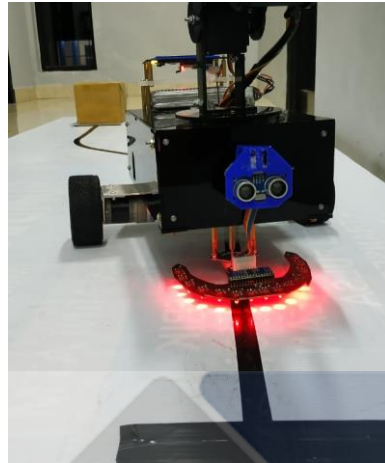
Gambar 4.18 Blok diagram kontrol sistem pengikut garis

Pergerakan *mobile robot* akan berdasarkan pada pembacaan sensor garis seperti yang diilustrasikan oleh gambar 4.17 yang kemudian dibuat menjadi algoritma pemrograman seperti yang tertera pada gambar 4.18. Arah pergerakan yang terjadi akan bergantung kepada hasil pembacaan sensor terhadap garis hitam pada arena. Kode sensor yang terdeteksi dimulai dari S0 dari sebelah kiri menuju S7 yang berada paling kanan seperti yang dijelaskan pada gambar 4.11 pada poin sebelumnya. Adapun untuk macam-macam pergerakan yang terjadi pada sistem pengikut garis ini tertera pada tabel 4.2. Pada tabel 4.2 ini dijelaskan kondisi-kondisi yang mungkin terjadi pada pergerakan robot dan kecepatan PWM yang digunakan untuk menggerakkan robot. Untuk menentukan kecepatan PWM yang digunakan metode *trial and error* agar mendapatkan kecepatan yang sesuai dan efisien untuk setiap pergerakan yang terjadi.

Kelincahan pergerakan *mobile robot* diatur melalui perbedaan kecepatan di setiap logika pembacaan sensor garis. Selain itu, ukuran roda yang digunakan juga akan mempengaruhi gerakan dari *mobile robot* ini, semakin besar roda yang digunakan maka pergeseran gerakan semakin besar. Sementara saat menggunakan roda yang lebih kecil, gerakan yang didapatkan oleh *mobile robot* akan lebih halus dibandingkan dengan roda yang besar.

Pembacaan sensor sangat dipengaruhi oleh hasil kalibrasi garis hitam dan latar putih yang dilakukan di awal. Ketika kalibrasi yang dilakukan tidak tepat,

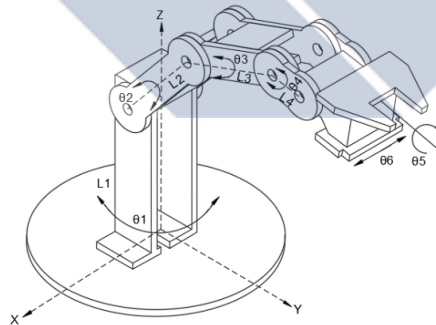
maka sensor pembacaan sensor akan kurang akurat dan menyebabkan pergerakan robot tidak terkendali. Gambar 4.19 menunjukkan gambaran saat pengujian sistem pendeteksian garis atau *line follower*.



Gambar 4.19 Sistem pendeteksian garis pada *line follower*

4.6.2 Pengujian Manipulator Lengan

Untuk mendapatkan posisi dari *end effector* digunakan metode analisa *Denevit-Hartenberg*. Permodelan robot lengan diilustrasikan sebagai gambar 4.20.



Gambar 4.20 Pemodelan Manipulator Robot

Berdasarkan gambar permodelan robot lengan dapat diambil parameter seperti pada tabel 4.2

Tabel 4.2 DH Parameter

i	α_i	a_i	D_i	Θ_i
1	90	0	L1	Θ_1
2	0	L2	0	Θ_2
3	0	L3	0	Θ_3

Setelah mendapatkan ke-empat parameter, dapat dirumuskan menjadi persamaan matriks (T) sebagai berikut:

$${}^0_1T = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

$${}^1_2T = \begin{bmatrix} C_2 & -S_2 & 0 & L_2 \cdot C_2 \\ S_2 & C_2 & 0 & L_2 \cdot S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

$${}^2_3T = \begin{bmatrix} C_3 & -S_3 & 0 & L_3 \cdot C_3 \\ S_3 & C_3 & 0 & L_3 \cdot S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.5)$$

$${}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \quad (4.6)$$

$${}^0_3T = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} C_2 & -S_2 & 0 & L_2 \cdot C_2 \\ S_2 & C_2 & 0 & L_2 \cdot S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} C_3 & -S_3 & 0 & L_3 \cdot C_3 \\ S_3 & C_3 & 0 & L_3 \cdot S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.7)$$

$${}^0_3T = \begin{bmatrix} C_1 \cdot C_{23} & -C_1 \cdot S_{23} & S_1 & C_1(L_2 \cdot C_2 + L_3 \cdot C_{23}) \\ S_1 \cdot C_{23} & -S_1 \cdot S_{23} & -C_1 & S_1(L_2 \cdot C_2 + L_3 \cdot C_{23}) \\ S_{23} & C_{23} & 0 & L_1 + L_2 \cdot S_2 + L_3 \cdot S_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.8)$$

$$\begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} C_1(L_2 \cdot C_2 + L_3 \cdot C_{23}) \\ S_1(L_2 \cdot C_2 + L_3 \cdot C_{23}) \\ L_1 + L_2 \cdot S_2 + L_3 \cdot S_{23} \end{pmatrix} \quad (4.9)$$

Adapun penjelasan variabel yang digunakan sebagai berikut:

- L_1 = Panjang lengan 1
- L_2 = Panjang lengan 2
- L_3 = Panjang lengan 3
- S_x = Sin θ_x ; dimana $x=1,2$, atau 3
- C_x = Cos θ_x ; dimana $x=1,2$, atau 3

Melalui persamaan (4.9) posisi *end-effector* dari manipulator lengan dapat diketahui. Untuk menguji kemampuan manipulator lengan dilakukan pengujian seperti yang dijelaskan pada tabel 4.3.

Tabel 4.3 Hasil Forward Kinematik

Percobaan Ke	Sudut <i>Joint</i> (derajat)			Posisi Robot (mm)						Error			Rata-rata Error	
				Perhitungan			Pengukuran							
	θ_1	θ_2	θ_3	P_x	P_y	P_z	P_x	P_y	P_z	P_x	P_y	P_z		Posisi
1	90	180	90	0	-70	7	8	-75	2	8	5	5	11	8
2	90	180	90	0	-70	7	5	-72	2	5	2	5	7	
3	90	180	90	0	-70	7	5	-74	4	5	4	3	7	
4	90	90	45	0	-49	196	7	-53	189	7	4	7	11	10
5	90	90	45	0	-49	196	5	-50	190	5	1	6	8	
6	90	90	45	0	-49	196	7	-55	190	7	6	6	11	
7	0	45	45	49	0	196	47	4	198	2	4	2	5	7
8	0	45	45	49	0	196	48	8	200	1	8	4	9	
9	0	45	45	49	0	196	51	6	197	2	6	1	6	
10	120	45	45	-25	43	196	-32	46	198	7	3	2	8	9
11	120	45	45	-25	43	196	-32	48	198	7	5	2	9	
12	120	45	45	-25	43	196	-34	46	196	9	3	0	10	
13	0	0	90	70	0	147	76	8	140	6	8	7	12	11
14	0	0	90	70	0	147	75	8	142	5	8	5	11	
15	0	0	90	70	0	147	75	6	142	5	6	5	9	
16	25	0	90	63	30	147	56	21	139	7	9	8	14	13
17	25	0	90	63	30	147	60	22	136	3	8	11	14	
18	25	0	90	63	30	147	61	22	137	2	8	10	13	
19	130	90	135	32	-38	98	35	-42	96	3	4	2	5	6
20	130	90	135	32	-38	98	36	-42	95	4	4	3	6	
21	130	90	135	32	-38	98	36	-44	96	4	6	2	8	
22	130	45	135	13	-16	126	18	-15	121	5	1	5	7	6
23	130	45	135	13	-16	126	16	-14	124	3	2	2	4	
24	130	45	135	13	-16	126	17	-14	122	4	2	4	6	
25	130	135	135	32	-38	56	34	-41	59	2	3	3	5	6
26	130	135	135	32	-38	56	34	-42	60	2	4	4	6	
27	130	135	135	32	-38	56	36	-44	58	4	6	2	8	
Rata-rata error (mm)										5	5	4	8	

Pengujian pergerakan manipulator robot yang ditunjukkan oleh tabel 4.3 menunjukkan hasil yang sedikit berbeda. Di setiap sumbu kartesian, terdapat selisih posisi akhir dari gerakan yang dihasilkan oleh robot lengan. Adapun perbedaan nilai yang terukur dan nilai posisi yang didapatkan dari hasil persamaan kinematik dapat terjadi akibat proses pengukuran yang dilakukan secara manual. Perbedaan hasil yang terjadi juga dipengaruhi oleh *backlash* pada servo. Selain itu kondisi atau performa dari motor servo yang digunakan akan mempengaruhi hasil pengukuran dari posisi *end-effector* robot.

Setiap komponen memiliki spesifikasi masa penggunaan efektif atau maksimal pada kondisi dengan atau tanpa beban, hal ini dapat mempengaruhi hasil posisi lengan robot. Sebab saat kondisi motor servo sudah tidak optimal, maka pergerakan yang dihasilkan oleh robot juga akan menurun keakuratannya. Ada beragam hal yang memicu menurunnya performa dari servo motor yang digunakan, diantaranya adalah kondisi pembawaan beban yang bertumpu pada beberapa servo saja. Pada saat melakukan percobaan atau saat melakukan pengujian, robot tentu akan berada pada posisi yang beragam, tidak jarang posisi-posisi yang ditentukan menyebabkan hanya servo tertentu yang bekerja terlalu keras, sehingga memicu penurunan kualitas kerja dari servo tersebut. Pada proyek akhir ini kondisi ini sangatlah sering terjadi pada servo motor dengan tipe SG90S yang digunakan pada *gripper*. Selama proyek akhir ini dilakukan, penggantian servo jenis ini terjadi hingga tiga kali, hal ini disebabkan oleh tingginya mobilitas penggunaan *gripper* saat melakukan percobaan. Dilansir dari laman diskusi [18] berdasarkan pengalaman yang dimiliki oleh beberapa orang bahwa benar adanya servo dengan gear berbahan plastik memiliki umur kurang lebih 100 jam pemakaian, sehingga penggunaan servo jenis ini harusnya disesuaikan dengan pemakaian yang akan dilakukan.

Selain itu, perubahan sudut yang terkadang terlalu signifikan juga menguji kualitas dari setiap servo, perubahan secara mendadak ini sering kali menyebabkan gear yang ada pada servo bekerja secara berlebihan atau terpaksa, sehingga timbullah kondisi *backlash* atau bahkan berdampak pada rusaknya gear yang ada.

Untuk mendapatkan hasil yang maksimal disetiap performa servo, maka digunakan tipe pemrograman perulangan, yang dimana maksud dari metode ini adalah menghindari pergerakan yang terlalu signifikan dan menyebabkan masalah diatas, sehingga pergerakan yang terjadi adalah gerakan sedikit demi sedikit menuju posisi akhir yang telah ditetapkan.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan data dari hasil pengujian yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

1. Berdasarkan pengujian servo dan pengukuran *duty cycle* diketahui bahwa dengan jangkauan yang sama dan tipe servo yang sama, *duty cycle* yang dimiliki dapat saja berbeda. Dengan demikian, konversi jumlah lebar pulsa PWM yang dibutuhkan untuk menggerakkan tiap-tiap motor servo pada robot lengan ke posisi yang sama juga berbeda.
2. Akurasi pergerakan *end-effector* dapat ditinjau dari perbandingan pengujian yang dilakukan dan nilai pengukuran yang didapatkan dari persamaan kinematika maju, hasil rata-rata nilai error menunjukkan perbedaan 5 mm pada sumbu X, 5 mm pada sumbu Y, dan 4 mm pada sumbu Z. Jika ditinjau dari tiap gerakan yang dilakukan, rata-rata perbedaan antara perhitungan dengan pengukuran adalah sejauh 8 mm.
3. Pergerakan robot secara keseluruhan masih menunjukkan gerakan osilasi yang cukup besar. Dimana pergerakan ini didapatkan dari hasil pembacaan sensor garis yang kemudian mengatur kecepatan gerak dari *mobile robot* sesuai dengan logika yang telah diberikan.

5.2 Saran

Dari hasil proyek akhir yang telah dilaksanakan, masih terdapat beberapa kekurangan, untuk memaksimalkan fungsi robot dibutuhkan beberapa pengembangan seperti penambahan tombol dan penggunaan kontrol PID. Dengan menambahkan tombol, robot dapat memiliki beragam mode penggunaan, seperti mode manual untuk memasukkan titik koordinat terakhir, atau menggunakan mode

otomatis saat menentukan titik koordinat. Selain itu, penambahan kontroller PID akan membuat pergerakan *line follower* semakin stabil. Kontroller PID juga dapat diterapkan pada robot manipulator, sehingga pergerakan lengan menjadi lebih stabil dan akurat.



DAFTAR PUSTAKA

- [1] S. Sudimanto and K. Kevin, "Perancangan Robot Pemindah Barang Berbasis Line Follower," *TESLA J. Tek. Elektro*, vol. 22, no. 1, p. 1, 2020, doi: 10.24912/tesla.v22i1.7807.
- [2] D. P. Angin *et al.*, "Perancangan Robot Lengan Pemindah Barang Berdasarkan Jarak," *Semin. Nas. Inov. dan Ilmu Komput.*, no. April, pp. 84–87, 2018.
- [3] G. Pegman, "The Strategic Research Agenda for Robotics in Europe," 2010.
- [4] Y. Yamamoto, "Coordinated Control of A Mobile Manipulator," *Robotica*, vol. 16, no. March, pp. 607–613, 1994, [Online]. Available: http://repository.upenn.edu/cis_reports/240/.
- [5] I. Farkhatdinov and J.-H. Ryu, *Switching of Control Signals in Teleoperation Systems: Formalization and Application*. 2008.
- [6] R. Supriyanto, Hustinawati, R. Nugraini, A. Bima Kurniawan, Y. Permadi, and A. Sa'ad, *ROBOTIKA*, vol. 1. 2010.
- [7] R. Syam and J. Hair, "Desain Kerjasama Mobile Manipulator Robot," *J. Otomasi Kontrol dan Instrumentasi*, vol. 8, no. 2, p. 125, 2016, doi: 10.5614/joki.2016.8.2.1.
- [8] T. G. Sugar and V. Kumar, "Control of cooperating mobile manipulators," *IEEE Trans. Robot. Autom.*, vol. 18, no. 1, pp. 94–103, 2002, doi: 10.1109/70.988979.
- [9] C. Perrier, P. Dauchez, and F. Pierrot, *A global approach for motion generation of non-holonomic mobile manipulators*, vol. 4. 1998.
- [10] B. Bayle, J. Y. Fourquet, and M. Renaud, "Manipulability analysis for mobile manipulators," *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2, pp.

1251–1256, 2001, doi: 10.1109/robot.2001.932782.

- [11] Randis, Suharlana, and R. Jayanti, “ANALISIS KINEMATIKA MOBILE MANIPULATOR PADA EXCAVATOR BACKHOE,” *DINAMIKA*, vol. 7, no. 1, pp. 25–32, 2015.
- [12] J. J. Craig, *Book Review: Introduction to Robotics: Mechanics and Control*, vol. 41, no. 4. 2006.
- [13] E. Pitowarno, “Robotika: Desain, Kontrol, dan Kecerdasan Buatan,” *Yogyakarta Andi Offset*, 2006.
- [14] J. Rosen, “Models of Robot Manipulation - EE 543,” 2006.
- [15] M. Julyus, C. Manullang, and M. K. Dewi, “Kendali Robot Beroda Otonom dengan Invers Kinematics Autonomous Wheeled Robot Control with Inverse Kinematics,” vol. 7, pp. 62–73, 2020.
- [16] B. Earl, “Adafruit PCA9685 16- Channel Servo Driver,” 2012. <https://learn.adafruit.com/16-channel-pwm-servo-driver/using-the-adafruit-library> (accessed Nov. 20, 2021).
- [17] P. S. Frima Yudha and R. A. Sani, “Implementasi Sensor Ultrasonik Hc-Sr04 Sebagai Sensor Parkir Mobil Berbasis Arduino,” *EINSTEIN e-JOURNAL*, vol. 5, no. 3, 2019, doi: 10.24114/einstein.v5i3.12002.
- [18] Tompuma, “How to make servo lives longer?,” *Arduino*, 2020. <https://forum.arduino.cc/t/how-to-make-servo-lives-longer/680853> (accessed Jan. 27, 2022).



LAMPIRAN



LAMPIRAN 1
DAFTAR RIWAYAT
HIDUP

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Ely Firman
Tempat & Tanggal Lahir : Tiang tara, 20 April 2000
Alamat Rumah : Jl. Raya Pangkal Pinang Mentok
Km 47 Desa Tiang Tara
No Hp : 082181451595
Email : elyfirmanatonid25@gmail.com
Jenis Kelamin : Laki-laki
Agama : Islam



2. Riwayat Pendidikan

SD Negeri 06 Bakam (2006 - 2012)
SMP Negeri 3 Bakam (2012 – 2015)
SMK Negeri 1 Bakam (2015 – 2018)

3. Pendidikan Non Formal

-

Sungailiat, 26 Januari 2022

Ely Firman

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Faradila Irianti
Tempat & Tanggal Lahir : Sungailiat, 3 Desember 2000
Alamat Rumah : Jl. Durian No.36 Komp. RSS
No Hp : 082184187480
Email : faradilair@gmail.com
Jenis Kelamin : Perempuan
Agama : Islam



2. Riwayat Pendidikan

SD Negeri 10 Sungailiat (2006 - 2012)
SMP Negeri 2 Sungailiat (2012 – 2015)
SMA Negeri 1 Sungailiat (2015 – 2018)


3. Pendidikan Non Formal

-

Sungailiat, 26 Januari 2022



Faradila Irianti



LAMPIRAN 2
PROGRAM ARDUINO
MEGA 2560

```

#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <Ultrasonic.h>
#include <Adafruit_PWMServoDriver.h>

Ultrasonic ultrasonic1(7, 6);
LiquidCrystal_I2C lcd(0x27, 16, 2);

Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver();
uint8_t servonum0 = 15; /*servo 0 paling bawah untuk kanan kiri
(waist)*/
uint8_t servonum1 = 14; /*sevo 1 sendi pertama untuk naik turun
(shoulder)*/
uint8_t servonum2 = 13; /*sevo 2 sendi kedua untuk naik turun
(elbow)*/
uint8_t servonum3 = 12; /*sevo 3 sendi ketiga untuk naik turun
(pitch)*/
uint8_t servonum4 = 11; /*sevo 4 sendi rotate kiri kanan pada
capit (roll)*/
uint8_t servonum5 = 10; /*sevo 5 capit buka tutup (gripper)*/
int sensor_pin = A8;
int cek;

const unsigned int en=A5;
const unsigned int s0=A2;
const unsigned int s1=A1;
const unsigned int s2=A0;
const unsigned int s3=A3;
const unsigned int sig=A4;

const int in2= 12;
const int in1 = 11;
const int in4 = 10;
const int in3 = 9;
const int ENA=13;
const int ENB=8;
uint8_t DataSensor;

int Kec_Min = 40;
int Kec_Max = 100;
int kiri=0, kanan=0;
double NilaiPosisi,outPID;
int x,y,r,s sensor;
int data_sensor[8];
unsigned int white[8], black[8], thres[8], Sensor[8];
void setup()
{
  pinMode(en, OUTPUT);
  pinMode(s0, OUTPUT);
  pinMode(s1, OUTPUT);
  pinMode(s2, OUTPUT);
  pinMode(s3, OUTPUT);
  Serial.begin(9600);
  lcd.init();
  lcd.init();
  lcd.backlight();

```

```

    pwm.begin();
    pwm.setOscillatorFrequency(27000000);
    pwm.setPWMFreq(50);
    pinMode(sensor_pin, INPUT);

    standby1();
    kalibrasi();
    threshold();
    tampil_thres();

}

void loop()
{

standby1();
baca_();
    while(sensor > 10)
    {
        y=1;
        baca_();
        lineFollow();
    }

stopped(0,0);
scan_sensor();

    while(DataSensor == 0b11111111)
    {
        ambil();
        y=0;
        r=1;
        scan_sensor();
    }

    while(cek ==1)
    {
        belokkiri(95,95);
        r=0;
        s=1;
        scan_sensor();
    }

    while(s==1 && DataSensor == 0b11111111)
    {
        s=0;
        taro();
    }

}

void selektor(){
    digitalWrite(en, LOW);
    digitalWrite(s0, HIGH);
    digitalWrite(s1, HIGH);
    digitalWrite(s2, LOW);
    digitalWrite(s3, LOW);
}

```

```

data_sensor[0]=analogRead(sig);
digitalWrite(s0, LOW);
digitalWrite(s1, LOW);
digitalWrite(s2, HIGH);
digitalWrite(s3, LOW);
data_sensor[1]=analogRead(sig);
digitalWrite(s0, HIGH);
digitalWrite(s1, LOW);
digitalWrite(s2, HIGH);
digitalWrite(s3, LOW);
data_sensor[2]=analogRead(sig);
digitalWrite(s0, LOW);
digitalWrite(s1, HIGH);
digitalWrite(s2, HIGH);
digitalWrite(s3, LOW);
data_sensor[3]=analogRead(sig);
digitalWrite(s0, HIGH);
digitalWrite(s1, HIGH);
digitalWrite(s2, HIGH);
digitalWrite(s3, LOW);
data_sensor[4]=analogRead(sig);
digitalWrite(s0, LOW);
digitalWrite(s1, LOW);
digitalWrite(s2, LOW);
digitalWrite(s3, HIGH);
data_sensor[5]=analogRead(sig);
digitalWrite(s0, HIGH);
digitalWrite(s1, LOW);
digitalWrite(s2, LOW);
digitalWrite(s3, HIGH);
data_sensor[6]=analogRead(sig);
digitalWrite(s0, LOW);
digitalWrite(s1, HIGH);
digitalWrite(s2, LOW);
digitalWrite(s3, HIGH);
data_sensor[7]=analogRead(sig);
}

void kalibrasi()
{
  lcd.setCursor(0,0);
  lcd.print("Ambil data putih");
  Serial.println(" Ambil data putih ");
  delay(5000);
  //ambil data dari sensor , simpan ke variabel putih
  selektor();
  for (int i=0; i<8; i++)
  {
    white[i]=data_sensor[i];

    Serial.print(white[i]);
    Serial.print(", ");
  }

  lcd.setCursor(0,1);
  lcd.print("Selesai");
}

```

```

    Serial.println(" Ambil data putih selesai ");
    // delay(1000);
    // lcd.clear();
    delay(2000);

    lcd.clear();
    delay(1500);
    lcd.setCursor(0,0);
    lcd.print("Ambil data hitam");
    Serial.println(" Ambil data hitam");
    delay(5000);
    //ambil data dari sensor , simpan ke variabel hitam
    selektor();
    for (int i=0; i<8; i++)
    {
        black[i]=data_sensor[i];

        Serial.print(black[i]);
        Serial.print(", ");
    }

    lcd.setCursor(0,1);
    lcd.print("Selesai");
    Serial.println(" Ambil data hitam selesai ");
    delay(1000);

    lcd.clear();
    lcd.setCursor(2,0);
    lcd.print("Start Engine");
}

void threshold()
{
    Serial.println("Nilai Threshold");
    for(int i=0; i<8; i++)
    {
        thres[i] = (black[i]+white[i])/2;
    }
}

void tampil_thres()
{
    for (int i=0; i<8; i++)
    {
        Serial.print(thres[i]);
        Serial.print(", ");
    }
}

void standby1(){
    //Robot standby
    pwm.setPWM(servonum0, 0, 400);
    delay(1000);
    pwm.setPWM(servonum1, 0, 140);
    delay(1000);
}

```

```

pwm.setPWM(servonum2, 0, 150);
delay(1000);
pwm.setPWM(servonum3, 0, 420);
delay(1000);
pwm.setPWM(servonum4, 0, 480);
delay(1000);
pwm.setPWM(servonum5, 0, 220);
delay(1000);
}

void ambil(){
  for (uint16_t pulselen3 = 420; pulselen3 > 300; pulselen3--){
    {
      pwm.setPWM(servonum3, 0, pulselen3);
      delay (40);
    }
  }

  for (uint16_t pulselen4 = 480; pulselen4 > 100; pulselen4--){
    {
      pwm.setPWM(servonum4, 0, pulselen4);
      delay (40);
    }
  }

  for (uint16_t pulselen1 = 140; pulselen1 < 300; pulselen1++){
    {
      pwm.setPWM(servonum1, 0, pulselen1);
      delay (40);
    }
  }

  for (uint16_t pulselen2 = 150; pulselen2 > 120; pulselen2--){
    {
      pwm.setPWM(servonum2, 0, pulselen2);
      delay (40);
    }
  }

  //-----Servo 3 menuju objek
  for (uint16_t pulselen3 = 300; pulselen3 > 180; pulselen3--){
    {
      pwm.setPWM(servonum3, 0, pulselen3);
      delay (40);
    }
  }
  //-----Servo 2 dorong
  for (uint16_t pulselen1 = 300; pulselen1 < 350; pulselen1++){
    {
      pwm.setPWM(servonum1, 0, pulselen1);
      delay (40);
    }
  }
  //-----Servo 5 capit
  barang();

  for (uint16_t pulselen1 = 350; pulselen1 > 280; pulselen1--){
    {
      pwm.setPWM(servonum1, 0, pulselen1);
      delay (40);
    }
  }
}

```

```

    }
}

void taro(){
    for (uint16_t pulselen3 = 250; pulselen3 > 300; pulselen3--)
        {
            pwm.setPWM(servonum3, 0, pulselen3);
            delay (40);
        }
    for (uint16_t pulselen1 = 230; pulselen1 > 200; pulselen1-
-)
        {
            pwm.setPWM(servonum1, 0, pulselen1);
            delay (40);
        }
    for (uint16_t pulselen2 = 150; pulselen2 > 100;
pulselen2--)
        {
            pwm.setPWM(servonum2, 0, pulselen2);
            delay (40);
        }
    //-----Servo 3 menuju objek
    for (uint16_t pulselen3 = 250; pulselen3 > 180; pulselen3--
)
        {
            pwm.setPWM(servonum3, 0, pulselen3);
            delay (40);
        }
    for (uint16_t pulselen1 = 200; pulselen1 < 300; pulselen1++)
        {
            pwm.setPWM(servonum1, 0, pulselen1);
            delay (40);
        }
    for (uint16_t pulselen2 = 100; pulselen2 < 150; pulselen2++)
        {
            pwm.setPWM(servonum2, 0, pulselen2);
            delay (40);
        }
    for (uint16_t pulselen3 = 180; pulselen3 > 120; pulselen3--
)
        {
            pwm.setPWM(servonum3, 0, pulselen3);
            delay (40);
        }
    for (uint16_t pulselen1 = 300; pulselen1 < 350; pulselen1++)
        {
            pwm.setPWM(servonum1, 0, pulselen1);
            delay (40);
        }
    for (uint16_t pulselen3 = 120; pulselen3 < 180; pulselen3++)
        {
            pwm.setPWM(servonum3, 0, pulselen3);
            delay (40);
        }
    for (uint16_t pulselen5 = 550; pulselen5 > 220; pulselen5--
)
        {

```

```

        pwm.setPWM(servonum5, 0, pulselen5);
        delay(25);
    }
}

void barang()
{
    cek = digitalRead(sensor_pin);

    if (cek==0)
    {
        pwm.setPWM(servonum5, 0, 500);
    }
    if (cek==1)
    {
        pwm.setPWM(servonum5, 0, 100);
    }
}

void maju(uint8_t _inputPwm1, uint8_t _inputPwm2)
{
    digitalWrite(in1, LOW);    digitalWrite(in3, HIGH);
    digitalWrite(in2, HIGH);  digitalWrite(in4, LOW);
    analogWrite(ENA, _inputPwm2); analogWrite(ENB, _inputPwm1);
}

void belokkiri(uint8_t _inputPwm1, uint8_t _inputPwm2)
{
    digitalWrite(in1, HIGH);   digitalWrite(in3, HIGH);
    digitalWrite(in2, LOW);    digitalWrite(in4, LOW);
    analogWrite(ENA, _inputPwm2); analogWrite(ENB, _inputPwm1);
}

void stopped(uint8_t _inputPwm1, uint8_t _inputPwm2)
{
    digitalWrite(in1, LOW);    digitalWrite(in3, LOW);
    digitalWrite(in2, LOW);    digitalWrite(in4, LOW);
    analogWrite(ENA, 0); analogWrite(ENB, 0);
}

void baca_ ()
{
    Serial.print("Sensor 01: ");
    sensor = ultrasonic1.read();
    Serial.print(sensor); // Prints the distance on the default unit
    (centimeters)
    Serial.println("cm");
    //    delayMicroseconds(2);
}

void scan_sensor()
{
    selektor();
    Serial.println((String)data_sensor[0] + '\t' + data_sensor[1] +
'\t' + data_sensor[2] + '\t' + data_sensor[3]
+ '\t' + data_sensor[4] + '\t' + data_sensor[5]+
'\t' + data_sensor[6]+ '\t' + data_sensor[7]);
}

```



```

for(int i=0; i<8; i++)
{
    if(data_sensor[i] > thres[i])
    {
        Sensor[i]=1;
    }
    else
    {
        Sensor[i]=0;
    }
}
Serial.println("Nilai Sensor = ");
Serial.println((String)Sensor[0] + '\t' + Sensor[1] + '\t' +
Sensor[2] + '\t' + Sensor[3]
+ '\t' + Sensor[4] + '\t' + Sensor[5] + '\t' +
Sensor[6] + '\t' + Sensor[7]);
DataSensor =
Sensor[0]*128+Sensor[1]*64+Sensor[2]*32+Sensor[3]*16+Sensor[4]*8+S
ensor[5]*4+Sensor[6]*2+Sensor[7]*1;
Serial.println("Nilai Binner = ");
Serial.println(DataSensor,BIN);
}

void posisi()
{
    scan_sensor();
    switch(DataSensor)
    {
        case 0b00000001: maju(180,165); x=0; break;
        case 0b00000011: maju(180,165); x=0; break;
        case 0b00000111: maju(180,165); x=0; break;
        case 0b00000010: maju(180,170); x=0; break;
        case 0b00000110: maju(180,170); x=0; break;
        case 0b00001110: maju(180,170); x=0; break;
        case 0b00000100: maju(180,175); x=0; break;
        case 0b00001100: maju(180,175); x=0; break;
        case 0b00011100: maju(180,175); x=0; break;
        case 0b00001000: maju(175,175); x=0; break;
        case 0b00011000: maju(170,170); break;
        case 0b00111000: maju(175,180); x=1; break; //kiri
        case 0b00010000: maju(175,180); x=1; break;
        case 0b00110000: maju(175,180); x=1; break;
        case 0b01110000: maju(170,180); x=1; break;
        case 0b00100000: maju(170,180); x=1; break;
        case 0b01100000: maju(170,180); x=1; break;
        case 0b01000000: maju(165,180); x=1; break;
        case 0b11000000: maju(165,180); x=1; break;
        case 0b10000000: maju(165,180); x=1; break;
        case 0b00000000:
            if(x==0) {belokkanan(180,180);break;}
            else {belokkiri(180,180);break;}
    }
    Serial.println("Nilai Posisi = ");
    Serial.println(x);
}

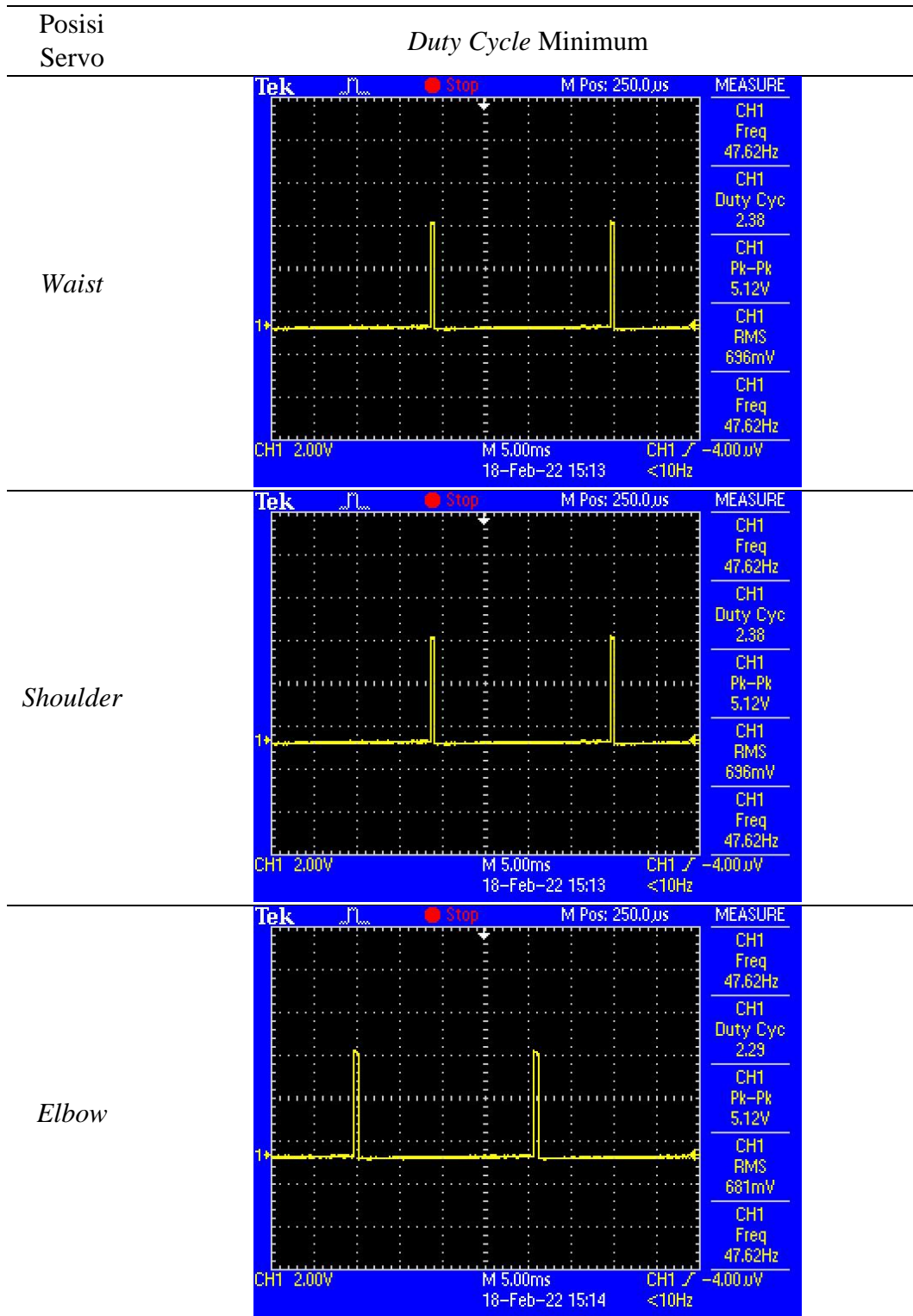
```

LAMPIRAN 3

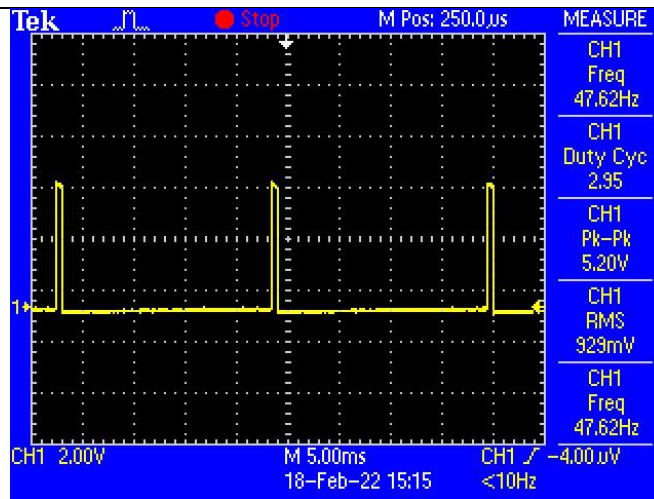
Pengujian *Duty Cycle* Pada Servo

PENGUJIAN DUTY CYCLE PADA SERVO

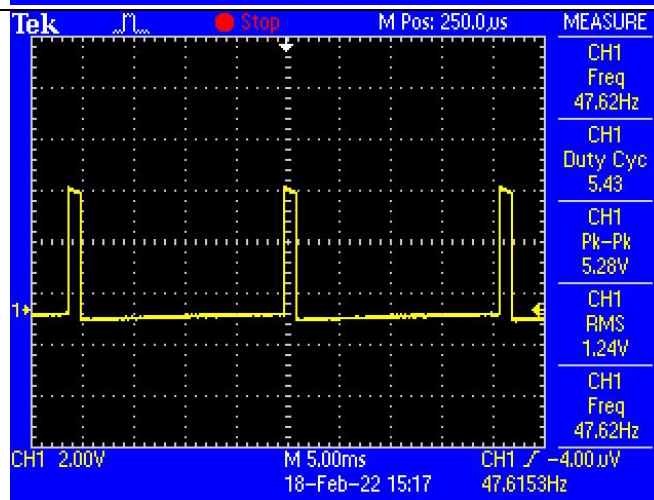
1. Duty Cycle Minimum



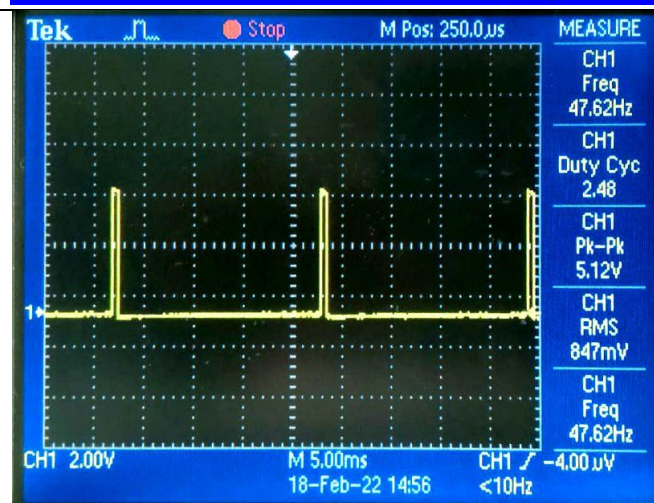
Pitch



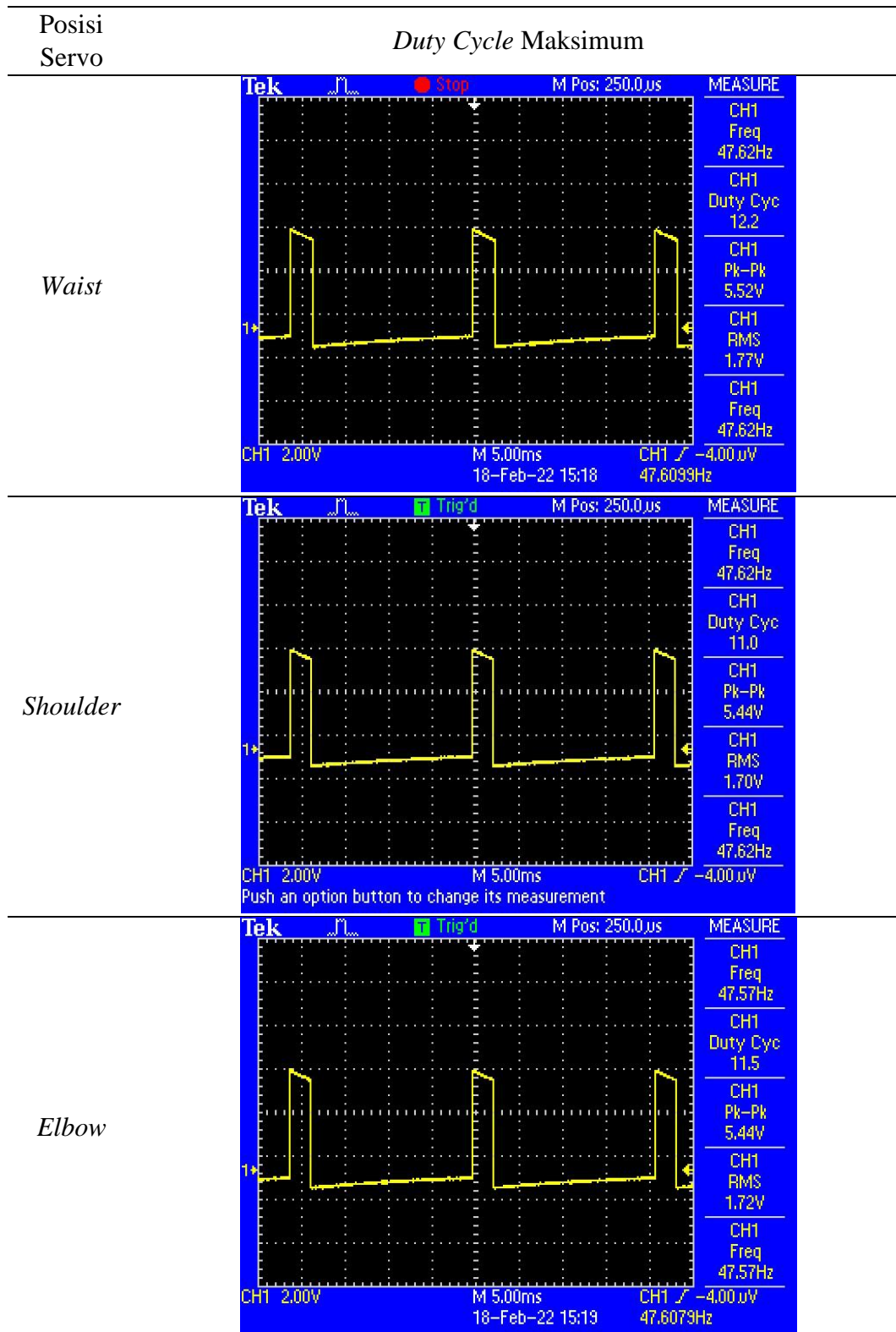
Roll



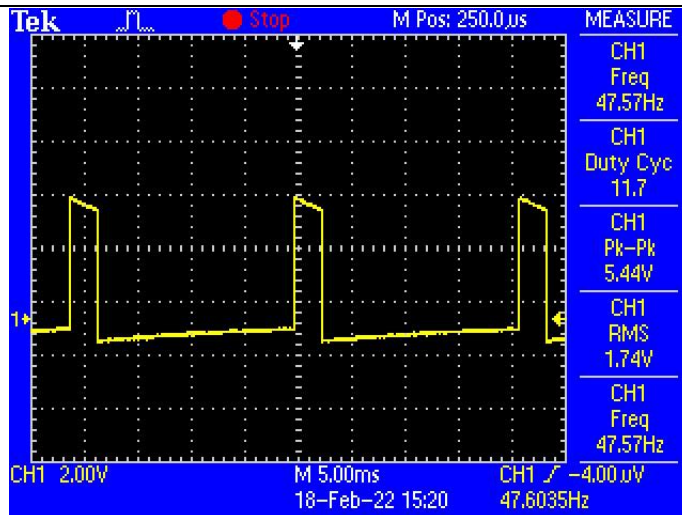
Gripper



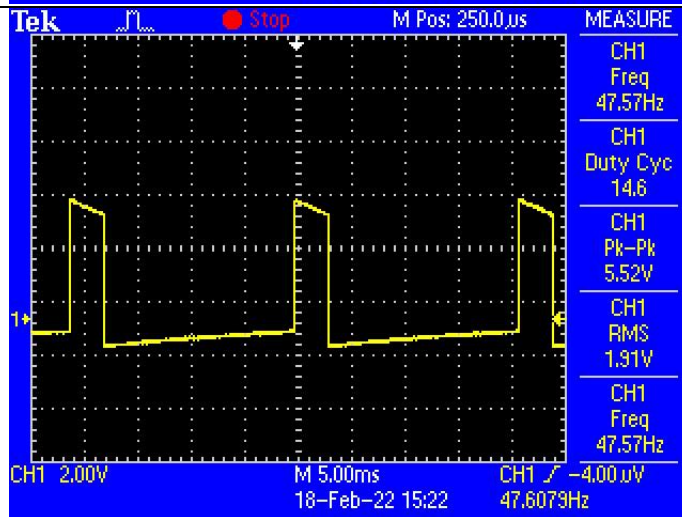
2. Duty Cycle Maksimum



Pitch



Roll



Gripper

