



TUGAS AKHIR - TE 141599

**PENGEMBANGAN SEMI AUTONOMOUS MOBILE ROBOT UNTUK
MENGAMBIL OBJEK DENGAN MEMPERTIMBANGKAN BERAT
OBJEK**

Rendi Valentino
NRP 2212 100005

Dosen Pembimbing
Ronny Mardiyanto, S.T.,M.T.,Ph.D
Dr. Ir. Djoko Purwanto, M.Eng

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016 ;



FINAL PROJECT - TE 141599

**DEVELOPMENT OF SEMI AUTONOMOUS MOBILE ROBOT TO
TAKE OBJECT CONSIDERING WEIGHT OF OBJECTS**

Rendi Valentino
NRP 2212 100005

Advisor
Ronny Mardiyanto, S.T., M.T., Ph.D
Dr. Ir. Djoko Purwanto, M.Eng

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2016

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “Pengembangan *Semi Autonomous Mobile Robot* Untuk Mengambil Objek Dengan Mempertimbangkan Berat Objek” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juni 2016

Rendi Valentino
NRP 2212 100 005

**PENGEMBANGAN SEMI AUTONOMOUS
MOBILE ROBOT UNTUK MENGAMBIL OBJEK
DENGAN MEMPERTIMBANGKAN BERAT
OBJEK**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik**

Pada

**·Bidang Studi Elektronika
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I

Dosen Pembimbing II



Ronny Mardiyanto, S.T., M.T., Ph.D
NIP: 198101182003121003



Dr. Ir. Djoko Purwanto, M.Eng
NIP: 196512111990021002



Pengembangan *Semi Autonomous Mobile Robot* Untuk mengambil Objek Dengan Mempertimbangkan Berat Objek

Nama : Rendi Valentino
Pembimbing I : Ronny Mardiyanto, ST., MT., Ph.D
Pembimbing II : Dr. Ir. Djoko Purwanto, M.Eng

ABSTRAK

Robot adalah sebuah manipulator yang dapat diprogram ulang, multifungsi dan didesain untuk memindahkan material, komponen atau perangkat khusus melalui gerakan terprogram. Robot memiliki beberapa jenis, antara lain *mobile robot*, *stationary robot*, *autonomous robot*, dan *remote controlled robot*. *Mobile robot* adalah robot yang mampu berpindah tempat secara otomatis maupun dikendalikan secara manual oleh manusia.

Pada Tugas akhir ini dirancang *mobile robot* yang bergerak secara semi otomatis melalui *remote controlled* untuk mengambil objek dilingkungan *off-road*. *Mobile robot* akan bergerak secara semi otomatis semi otomatis berdasarkan koordinat posisi *Global Positioning System* (GPS). Rangka robot yang digunakan merupakan rangka *Remote Controlled Car* jenis *monster truck* sehingga robot bergerak dengan bantuan roda. Desain lengan robot yang digunakan menggunakan prinsip *parallel mechanism* seperti desain dari lengan robot milik ABB seri IRB 460. Servo di letakkan di bagian base lengan robot, sehingga lengan digerakkan secara parallel oleh servo dibagian base. Pada penelitian ini, dilakukan estimasi berat objek dengan menggunakan metode regresi.

Berat objek maksimum yang mampu diangkat oleh lengan robot sebesar 1kg ketika panjang lengan 30 cm. Estimasi berat objek dengan menggunakan regresi linier menghasilkan error hingga 30%. Dengan menggunakan rangka RC Car jenis *monster truck*, robot dapat bergerak dan stabil ketika berjalan di lingkungan terjal. Robot tidak dapat melewati halangan batu yang tingginya lebih dari 10 cm.

Kata Kunci : *mobile robot*, estimasi berat objek, arus listrik.

DEVELOPMENT OF SEMI AUTONOMOUS MOBILE ROBOT TO TAKE OBJECT CONSIDERING WEIGHT OF OBJECTS

Name : Rendi Valentino
1st Advisor : Ronny Mardiyanto, ST., MT., Ph.D
2nd Advisor : Dr. Ir. Djoko Purwanto, M.Eng

ABSTRACT

The robot is a manipulator that can be reprogrammed, multifunctional and designed to move material, parts or specialized devices with programmed movement. Robot has some kind, for example mobile robots, stationary robots, autonomous robots and remote controlled robot. Mobile robot is a robot that can move automatically or controlled manually by humans.

In this final project will be designed mobile robot that moves semi-automatic controlled by a remote controlled to put objects on the off-road environment. Mobile robot will move semi-automatic based on the position coordinates Global Potitioning System (GPS). frame that be used is frame of RC car Monster truck, it is supports the move in rugged environments. Design a robot arm used principle of parallel mechanism, such as the design of the robot arm belonging to the ABB IRB 460 series . servo place at the base of the robot arm so that the arm is moved in parallel by the servo section base. In this research, estimation weight of object be done by used regression method.

The maximum weight of the object lifted by a robotic arm capable of 1kg when the arm length of 30 cm. Estimated weight of the object by using linear regression result error until 30%. By using the framework of RC Car kind of monster truck, robot can move and stable when walking in rugged environments. Robot can not through the stone taller than 10 cm.

Keyword: mobile robot, estimating the weight, electric current.

KATA PENGANTAR

Puji syukur atas rahmat yang diberikan oleh Tuhan Yang Maha Esa. Karena berkat rahmat dan hidayah-Nya penulis dapat menyelesaikan penelitian ini. Selama pelaksanaan penelitian Tugas Akhir ini, penulis mendapatkan bantuan dari berbagai pihak baik dukungan secara moril maupun materil. Terima kasih yang sebesar-besarnya penulis sampaikan kepada berbagai pihak yang mendukung dan membantu dalam tugas akhir ini , diantaranya :

1. Tuhan Yang Maha Esa yang selalu memberi bimbingan melalui doa.
2. Bapak Joko Wiyono dan ibu Sudyastuti selaku orang tua penulis yang selalu memberi dukungan dan doa untuk penulis.
3. Bapak Ronny Mardiyanto, ST., MT.,Ph.D. dan Dr. Ir. Djoko Purwanto, M.Eng. yang tidak pernah lelah dan selalu sabar membimbing penulis.
4. Elizabeth Widya C, dan Matheus Sony serta Samuel Melky yang selalu memberi saran kepada penulis selama kuliah.

Penulis sadar bahwa Tugas Akhir ini masih belum sempurna dan masih banyak hal yang perlu diperbaiki. Saran, kritik dan masukan baik dari semua pihak sangat membantu penulis terutama untuk berbagai kemungkinan pengembangan lebih lanjut.

Surabaya, Juni 2016

Rendi Valentino
2212100005

DAFTAR ISI

	Halaman
ABSTRAK	i
ABSTRACT.....	iii
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	2
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	4
1.7 Relevansi.....	4
BAB II LANDASAN TEORI	5
2.1 Robot.....	5
2.2 Parallel Mechanism.....	6
2.3 Radio Pengendali	7
2.4 Elektronik Pengatur Kecepatan.....	9
2.5 Modulasi Lebar Pulsa.....	10
2.6 Motor Servo	11
2.6 GPS (Global Positioning System)	12
2.8 Sensor Arus.....	12
2.9 Motor DC.....	13
2.10 Torsi Motor	14
2.11 Regresi	16
2.11.1 Regresi Linier	16
2.11.2 Regresi Polynomial	17
2.12 Ardupilot Mega (APM).....	17
2.13 Estimasi Berat objek	17
2.14 Software Mission Planner	18
2.15 Tinjauan Pustaka	19
2.15.1 An Automatic Control System for EOD Robot Based on Binocular Vision Position	19
2.15.2 The Optimization Scheme for EOD Robot Based on Supervisin Control Architecture.....	20

2.15.3	Perancangan Kendali Lengan Robot Untuk Mengambil Objek Menggunakan Kamera	21
2.15.4	Image Fusion for an EOD Robot Wireless Visual Module	21
2.15.5	A Single-Hand and Binocular Visual System for EOD Robot.....	21
2.15.6	Design and Construction of a Didactic 3-DOF Parallel Links RobotStation with a 1-DOF Gripper	22
2.15.7	Portable Serial Robot Manipulator with Distributed Actuation Mechanism	22
2.15.8	Distributed-Actuation Mechanism for a Finger-Type manipulator: Theory and Experimentns.....	23
BAB III PERANCANGAN SISTEM		26
3.1	Diagram Blok Sistem.....	26
3.2	Desain robot.....	27
3.3	Perancangan Lengan Robot	32
3.4	Remot kontrol dan Receiver	37
3.4	Sensor Arus.....	40
3.5	Arduino Mega.....	41
3.5.1	Inverse Kinematik	43
3.5.2	Kendali Lengan Robot	45
3.5.3	Estimasi Berat Objek	46
3.6	Perancangan Navigasi.....	50
3.6.1	Perancangan Perangkat Keras Navigasi	50
3.6.2	Perancangan Perangkat Lunak Navigasi	51
BAB IV PENGUKURAN DAN ANALISA.....		53
4.1	Pengujian Arus Servo	53
4.1.1	Pengujian Panjang Lengan 16 cm	53
4.1.2	Pengujian Panjang Lengan 18 cm	55
4.1.3	Pengujian Panjang Lengan 20 cm	56
4.1.4	Pengujian Panjang Lengan 22 cm	57
4.1.5	Pengujian Panjang Lengan Lengan 24 cm	58
4.1.6	Pengujian Panjang Lengan 26 cm	59
4.1.7	Pengujian Panjang Lengan 28 cm	60
4.1.8	Pengujian Panjang Lengan 30 cm	61
4.2	Pengujian error estimasi berat objek.....	62
4.2.1	Pengujian Estimasi Berat Objek Dengan Regresi Polynomial orde 2.....	62
4.2.2	Pengujian Estimasi Berat Objek Dengan Regresi Linier ...	64

4.2.3 Pengujian Estimasi Berat Objek Dengan Regresi Polynomial orde 3	65
4.3 Pengujian Sudut Lengan	67
4.4 Pengujian Navigasi	68
4.5 Pengujian Di Lingkungan <i>Off-Road</i>	70
BAB V PENUTUP.....	73
5.1 Kesimpulan	73
5.2 Saran	73

DAFTAR GAMBAR

Gambar 2.1 Lengan robot Serial	6
Gambar 2.2 Parallel manipulator.	6
Gambar 2.3 Radio Controller	8
Gambar 2.4 Electrical Speed Controller	9
Gambar 2.5 Pulse Width Modulation	9
Gambar 2.6 Motor Servo.	10
Gambar 2.7 Motor DC.....	14
Gambar 2.8 Rangkaian motor DC penguat terpisah.....	13
Gambar 2.9 Regresi Linier	15
Gambar 2.10 Tampilan mission planner	18
Gambar 2.11 EOD robot berbasis binocular vision	19
Gambar 2.12 EOD robot berbasis <i>Supervision Control Architecture</i>	20
Gambar 2.13 Desain lengan robot.....	22
Gambar 2.14 Desain lengan robot <i>Distributed Actuation Mechanism</i>	23
Gambar 2.15 Konsep <i>distrubted-actuation</i> menggunakan <i>mechanism sliding-actuation</i> jari robot.....	24
Gambar 3.1 Ilustrasi sistem kerja robot	25
Gambar 3.2 Diagram Sistem Robot	26
Gambar 3.3 Rancangan robot tampak depan.....	27
Gambar 3.4 Robot tampak dari sisi kiri	28
Gambar 3.5 Robot tampak dari sisi kanan	29
Gambar 3.6 Layer dasar rangka robot.....	29
Gambar 3.7 Ilustrasi Layer dua rangka robot.....	30
Gambar 3.8 Ilustrasi robot layer tiga.....	31
Gambar 3.9 Ilustrasi ukuran robot	32
Gambar 3.10 Konstruksi lengan robot serial.....	33
Gambar 3.11 Ilustrasi lengan robot tampak sisi kanan.....	34
Gambar 3.12 Ilustrasi lengan robot tampak sisi kiri	35
Gambar 3.13 Ilustrasi lengan robot tampak depan.....	36
Gambar 3.14 Remote control	37
Gambar 3.15 Receiver.....	37
Gambar 3.16 Ilustrasi perintah remote ke receiver	38

Gambar 3.17 <i>Wiring</i> receiver ke Ardupilot dan Arduino Mega	39
Gambar 3.18 Sensor Arus	40
Gambar 3.19 Arduino Mega	41
Gambar 3.20 Alur Program Arduino Mega	42
Gambar 3.21 Pemodelan lengan dalam garis	43
Gambar 3.22 Pemodelan pergerakan sudut servo 2	44
Gambar 3.23 Pemodelan pergerakan sudut servo 3	45
Gambar 3.24 Ilustrasi input output kendali hardware lengan robot	45
Gambar 3.25 Kontrol lengan robot	46
Gambar 3.26 Perancangan persamaan berat objek	46
Gambar 3.27 Algoritma penentuan estimasi berat objek	49
Gambar 3.28 Perancangan hardware navigasi robot	50
Gambar 3.29 Tampilan awal software mission planner	51
Gambar 4.1 Grafik perubahan arus terhadap berat objek jarak 16 cm.....	53
Gambar 4.2 Grafik perubahan arus terhadap berat objek jarak 18 cm.....	55
Gambar 4.3 Grafik perubahan arus terhadap berat objek jarak 20 cm.....	56
Gambar 4.4 Grafik perubahan arus terhadap berat objek jarak 22 cm.....	57
Gambar 4.5 Grafik perubahan arus terhadap berat objek jarak 24 cm.....	58
Gambar 4.6 Grafik perubahan arus terhadap berat objek jarak 26 cm.....	59
Gambar 4.7 Grafik perubahan arus terhadap berat objek jarak 28 cm.....	60
Gambar 4.8 Grafik perubahan arus terhadap berat objek jarak 30 cm.....	61
Gambar 4. 9 Pengujian sudut servo	68
Gambar 4. 10 Kalibrasi APM	69
Gambar 4. 11 Pemrograman <i>waypoint</i> pada APM	70
Gambar 4. 12 Pengujian kehandalan robot di lingkungan <i>off-road</i>	71
Gambar 4. 12 Robot menemui rintangan batu setinggi 10 cm.....	71

DAFTAR TABEL

Tabel 4.1 Pengujian estimasi berat objek dengan beban 0.745 kg	63
Tabel 4.2 Pengujian estimasi berat objek dengan beban 0.635 kg	63
Tabel 4.3 Pengujian estimasi berat objek dengan beban 0.478 kg	64
Tabel 4.4 Pengujian estimasi berat objek dengan beban 0.735 kg	65
Tabel 4.5 Pengujian estimasi berat objek dengan beban 0.943 kg	65
Tabel 4.6 Pengujian estimasi berat objek dengan beban 0.563 kg	66
Tabel 4.7 Pengujian estimasi berat objek dengan beban 0.425 kg	66
Tabel 4.8 Pengujian estimasi berat objek dengan beban 0.863 kg	67
Tabel 4.7 Tabel pengujian sudut servo.....	68
Tabel 4.8 titik waypoint yang dituju	69

BAB I

PENDAHULUAN

1.1 Latar Belakang

Robot adalah sebuah manipulator yang dapat diprogram ulang, multifungsi dan didesain untuk memindahkan material, komponen atau perangkat khusus melalui gerakan terprogram. Teknologi robot digunakan untuk melakukan tugas yang dilakukan oleh manusia karena kecepatan dan kepresisiannya dalam melakukan tugas. Pemanfaatan teknologi robot dalam kehidupan sehari-hari untuk mengurangi resiko yang didapat manusia. Robot memiliki beberapa jenis, antara lain *mobile robot*, *stationary robot*, *autonomous robot*, dan *remote controlled robot*. *Mobile robot* adalah robot yang mampu berpindah tempat secara otomatis maupun dikendalikan secara manual oleh manusia.

Mobile robot sering digunakan untuk menjelajahi lingkungan berbahaya yang tidak mampu dijangkau oleh manusia. Pada tahun 2011, bencana radiasi di reaktor nuklir Fukushima menyebabkan lingkungan Fukushima tidak dapat dimasuki manusia karena tingkat radiasi yang tinggi, sehingga sejumlah *mobile robot* diterjunkan untuk menyisir lingkungan reaktor nuklir. *Mobile robot* juga digunakan untuk menangani masalah *Explosive Ordinance Disposal*(EOD). Keuntungan utama menggunakan robot untuk menangani tugas EOD untuk mengurangi resiko yang didapat manusia.

Meskipun jauh dari sempurna, robot telah berjasa dalam menyelamatkan nyawa manusia dengan mencari dan mengambil objek yang mudah meledak tanpa perlu kontak secara langsung antara manusia dan objek. *Mobile robot* dapat dikendalikan secara manual dari jarak jauh menggunakan *remote control*, hal ini bertujuan memudahkan pengguna mengambil benda dalam berbagai posisi tanpa perlu kontak langsung dengan objek. Mengambil benda dari jarak jauh mengurangi resiko pengguna terhadap objek berbahaya.

Mengambil objek secara tidak langsung dengan bantuan *mobile robot* menyebabkan pengguna tidak mampu mengestimasi berat objek. Berdasarkan peraturan Kepolisian Negara Republik Indonesia nomor 11 tahun 2010 dengan mengetahui berat objek, maka dapat ditentukan metode penanganan bomb dan jarak aman penanganan bomb. Pada lengan robot semakin besar torsi yang dihasilkan motor servo maka daya

angkat lengan robot semakin besar. Semakin berat objek yang diangkat robot, maka torsi yang dibutuhkan untuk mengangkat objek akan semakin besar. Konstruksi robot secara serial menghasilkan beban torsi yang besar pada servo base lengan robot sehingga kemampuan lengan robot untuk mengangkat objek berkurang.

Berdasarkan uraian diatas, maka akan dilakukan perancangan mobile robot untuk mengambil objek yang mampu mengestimasi berat objek melalui arus yang mengalir ke motor servo. Objek akan diangkat kemudian diukur perubahan arus yang mengalir pada motor servo. Dari perubahan arus yang mengalir, maka dapat dicari hubungan antara torsi motor servo dengan arus listrik terhadap berat objek sehingga dapat diketahui berat objek yang diangkat tanpa perlu kontak secara langsung terhadap objek yang diambil. Perancangan lengan robot dibuat secara *parallel mechanism* agar dapat memaksimalkan kemampuan lengan robot dalam mengangkat objek.

1.2 Perumusan Masalah

Permasalahan pada tugas akhir ini adalah sebagai berikut :

1. Bagaimana robot dapat bergerak di lingkungan kasar?
2. Bagaimana mengestimasi berat objek
3. Bagaimana *mobile robot* bergerak secara semi otomatis berdasarkan koordinat GPS

1.3 Tujuan Penelitian

Penelitian pada tugas akhir ini bertujuan sebagai berikut :

1. Mengambil objek di lingkungan kasar.
2. Mengestimasi berat objek melalui arus pada servo.
3. Robot dapat bergerak berdasarkan koordinat GPS dan kompas.

1.4 Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Pengambilan objek dilakukan secara manual.
2. Objek yang diangkat memiliki bentuk bagian yang menyerupai pengait.
3. Lingkungan *off-road* tidak basah.
4. Berat maksimum objek yang dapat diangkat lengan robot yaitu 1 kg ketika panjang lengan 30 cm.
5. Tidak dapat melewati rintangan batu dengan tinggi lebih dari 10 cm.

1.5 Metodologi Penelitian

Dalam penyelesaian tugas akhir ini digunakan metodologi sebagai berikut:

1. Studi literatur

Pada tahapan ini dilakukan pencarian dan pengkajian referensi yang berhubungan dengan topik tugas akhir. Referensi studi literatur didapatkan melalui jurnal ilmiah, buku, dan browsing melalui internet yang berhubungan dengan judul tugas akhir ini. Studi literatur dilakukan secara beriringan dengan penelitian.

2. Perancangan Robot

Pada tahap ini dirancang robot yang mampu bergerak di lingkungan kasar dan mengestimasi berat objek . Perancangan ini dilakukan berdasarkan studi literatur. Perancangan meliputi perancangan *hardware* dan *software* robot. Perancangan *hardware* meliputi pergerakan robot dan lengan robot. Perancangan *software* robot dilakukan pada mikrokontroler Arduino. Kinematik lengan ataupun pengukuran arus dan estimasi berat objek melalui perancangan *software*.

3. Pengujian dan Analisa Data

Tahapan ini dilakukan sebagai pengujian terhadap robot untuk melihat respon robot. Arus pada servo dan estimasi berat objek akan diuji. Pengujian arus pada servo untuk mengetahui perubahan arus yang mengalir ke servo terhadap berat objek. Pengujian estimasi berat objek untuk mengetahui error antara berat objek yang diestimasi robot dengan alat ukur berat objek.

4. Penarikan Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan berdasarkan pengujian, analisa data. Pada kesimpulan akan diambil garis besar jawaban dari permasalahan yang dianalisis. Saran juga akan diberikan sebagai masukan terhadap penelitian yang telah dilakukan. Di harapkan penelitian ini dapat bermanfaat dan dapat dikembangkan.

5. Penyusunan Buku Tugas Akhir

Tahap penyusunan buku tugas akhir akan dilakukan beriringan dengan penelitian. Penyusunan buku tugas akhir adalah bentuk laporan tertulis dari penelitian. Penulisan buku tugas akhir dilakukan dengan sistematika penulisan yang telah ditentukan.

1.6 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari Lima Bab dengan sistematika penulisan sebagai berikut:

- Bab 1 : Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan, dan relevansi.

- Bab 2 : Dasar Teori

Bab ini menjelaskan tentang berbagai macam teori pendukung dalam penulisan tugas akhir. Bab ini meliputi dasar teori dari robot, parallel mechanism, Remote controler, ESC, PWM, motor servo, GPS, torsi dan sensor arus, estimasi berat objek.

- Bab 3: Perancangan Sistem

Bab ini menjelaskan tentang perencanaan sistem perangkat keras elektrik dan mekanik, serta perangkat lunak. Bab ini juga berisi menjelaskan tentang prosedur pengujian yang dilakukan dalam penelitian.

- Bab 4 : Pengujian dan Pembahasan Sistem

Bab ini menjelaskan tentang hasil yang didapat dari pengujian tiap blok sistem secara keseluruhan

- Bab 5 : Penutup

Bab ini menjelaskan tentang kesimpulan meliputi kekurangan-kekurangan pada kerja alat dari hasil analisa serta saran untuk pengembangan ke depan.

1.7 Relevansi

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Dapat digunakan sebagai alat pengambil objek berbahaya.
2. Dapat menjelajahi lingkungan berbahaya.
3. Sebagai dasar penelitian lebih lanjut, agar dapat lebih dikembangkan.

BAB II

LANDASAN TEORI

2.1 Robot

Robot adalah sebuah alat mekanik yang dapat melakukan tugas fisik, baik menggunakan pengawasan dan kontrol manusia, ataupun menggunakan program yang telah didefinisikan terlebih dulu (kecerdasan buatan). Robot biasanya digunakan untuk tugas yang berat, berbahaya, pekerjaan yang berulang dan kotor. Penggunaan robot lainnya termasuk untuk pembersihan limbah beracun, penjelajahan bawah air dan luar angkasa, pertambangan. Robot memiliki banyak jenis.

1. Mobile robot

Mobile robot adalah robot yang dapat berpindah tempat. Mobile robot dibedakan lagi menjadi 2 berdasarkan media pergerakannya yaitu robot berkaki dan robot beroda. Umumnya mobile robot bergerak dengan roda. Mobile robot memiliki kinerja perpindahan yang ringan dibandingkan mobile robot berkaki. Hal ini dikarenakan robot beroda tidak memerlukan persamaan inverse kinematic untuk bergerak. Sedangkan robot berkaki memerlukan inverse kinematic

2. Stationery robot

Stationery robot merupakan robot yang hanya dapat melakukan pekerjaan di satu tempat tanpa berpindah tempat. Umumnya robot jenis ini ada di industry. Salah satu contoh robot jenis ini adalah lengan robot 6 DoF milik ABB.

3. Autonomous robot

Autonomous robot merupakan robot yang bergerak secara terprogram dan melakukan pekerjaan secara otomatis. Robot ini memiliki pekerjaan yang tetap dan tidak boleh berubah. Robot ini memiliki kemampuan bergerak yang cepat dan setiap pekerjaan yang dilakukan memiliki tingkat keakuratan yang tinggi. Namun robot jenis ini memiliki kekurangan, yaitu pekerjaan yang dilakukan kurang fleksibel. Jika ingin merubah pekerjaan yang dilakukan harus merubah program yang ada di robot.

4. Remote Controlled robot

Robot jenis ini adalah robot yang dikendalikan oleh manusia melalui alat bantu berupa remote pengendali. Robot jenis ini umumnya bergerak secara manual. Jenis pekerjaan yang dilakukan juga fleksibel.

Namun tingkat ketepatan pekerjaan yang dilakukan rendah karena robot ini dibuat manual.

2.2 Parallel Mechanism

Konstruksi robot paling umum adalah konstruksi secara serial manipulator secara seri dari dasar robot hingga *end-effector*.



Gambar 2. 1 Lengan robot serial

Desain secara serial memiliki keuntungan yaitu keleluasaan dalam bergerak. Desain ini sering digunakan dalam industry. Namun, serial manipulator memiliki kekurangan salah satunya membutuhkan torsi yang besar untuk menggerakkan *end-effector*. Maka dari itu konstruksi alternative diusulkan dalam robotika. Salah satu contoh yang terkenal adalah parallel manipulator dalam hal ini motor dipasas secara parallel.

Keuntungan parallel manipulator adalah berat objek sering terpusat pada *single base platform*, pergerakan lengan terletak pada penopang dan join itu sendiri. Penurunan massa di sepanjang lengan memungkinkan kontruksi actuator yang lebih ringan dan gerakan cepat.



Gambar 2. 2 Parallel manipulator

Pemusatan massa juga mengurangi momen inersia dan hal ini sangat menguntungkan *mobile robot*. Serial manipulator memiliki gerakan yang terlalu fleksibel sehingga dalam menentukan kinematisnya diperlukan beberapa sumbu. Konstruksi serial manipulator membutuhkan torsi yang besar untuk menggerakkan end-effector, hal ini dapat diperbesar oleh jarak antara joint dan *end-effector*. Konstruksi serial manipulator tidak memungkinkan untuk menguatkan gerakan satu sendi terhadap sendi lain. Hysterisis tidak terelakkan dan fleksibilitas sumbu menumpuk di sepanjang kinematik lengan. Di sisi lain dengan parallel manipulator, desain yang kaku dapat diperoleh beban massa yang kecil dari manipulator. Hal ini memungkinkan kecepatan gerakan yang tinggi dan torsi relative kecil untuk menggerakkan lengan.

2.3 Radio Pengendali

Radio pengendali adalah suatu alat yang menggunakan sinyal radio untuk mengontrol suatu piranti dari jarak jauh. Radio pengendali biasanya digunakan untuk mengendalikan berbagai macam peranti dari sebuah pengontrol yang dapat dibawa-bawa (*mobile*). Berbagai macam kepentingan penelitian ilmiah, industri, dan militer juga memanfaatkan

radio controller. Penelitian ini digunakan radio controller untuk mengendalikan robot. Radio pengendali adalah *transmitter* yang biasanya terdiri dari berbagai macam tuas, saklar, dan juga tombol. *Receiver* dari radio pengendali dipasang pada robot. *Receiver* menerima sinyal dari *transmitter* untuk kemudian dilanjutkan menuju servo, motor DC, atau berbagai keperluan lainnya.

Transmitter mengirimkan sinyal menuju *receiver* dengan modulator dan multiplexer. Sinyal yang dikirim oleh *transmitter* kemudian diterjemahkan oleh *receiver* dengan mendemodulasi sinyal dan membagi sinyal kedalam beberapa chanel. Radio controller memiliki frekuensi kerja spesifik namun terkadang dapat terjadi interferensi. Pada beberapa bidang seperti industri dengan alat berat, interferensi radio bisa sangat berbahaya dan merugikan. Maka dari itu pada radio controller industri sinyal yang dikirim oleh transmitter biasanya disertai kode tertentu untuk diterjemahkan receiver. Akan tetapi, radio control pada kelas hobby atau mainan biasanya memiliki pilihan frekuensi yang sangat luas, sehingga jarang terjadi interferensi.



Gambar 2. 3 Radio Controller

Pada penelitian ini terdapat banyak komunikasi sinyal radio yang akan digunakan. Diantara alat komunikasi radio yang akan digunakan dalam penelitian ini adalah radio controller yang sering digunakan pada mobil mainan. Radio controller ini dipilih karena fungsionalitasnya yang tinggi serta ergonomis. Radio controller ini didesain untuk dapat dikembangkan dengan mudah oleh para *hobbies*. Komunikasi yang lain yang menggunakan radio adalah kamera wireless. Gambar radio controller yang sering digunakan para hobbies dapat dilihat pada.

2.4 Elektronik Pengatur Kecepatan

Laju robot dikontrol menggunakan *ESC (Electronic Speed Control)* atau pengontrol kecepatan elektronik. Tenaga yang diberikan pada ESC sebanding dengan tuas pada controller. Semakin jauh tuas di dorong, maka semakin cepat pula robot melaju. Tegangan pengontrol kecepatan adalah sinyal pulsa dengan berbagai macam output dan menggunakan transistor yang telah di seleksi untuk menghasilkan transisi sinyal yang halus dan efisiensi tinggi. ESC juga dapat menggunakan motor yang dicontrol sebagai rem magnetic, sehingga pengontrol kecepatan secara elektronik lebih unggul dari pada pengontrol kecepatan secara mekanis. Pengontrol kecepatan secara mekanis menggunakan jaringan resistor. Jaringan resistor ini disambungkan secara bergantian dengan cara memutar controller sehingga elektroda kontroler tersambung dengan salah satu resistor.



Gambar 2. 4 *Electrical Speed Controller*

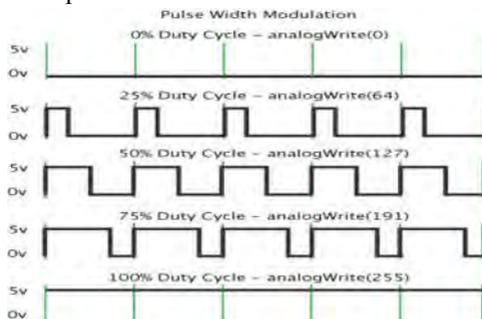
Reaksi dari pengontrol kecepatan mekanis lebih lambat dari pada pengontrol kecepatan elektronis karena didalam pengontrol kecepatan mekanis terdapat motor servo untuk memutar jaringan elektroda dan menghubungkannya dengan jaringan resistor. Pengontrol kecepatan mekanis juga lebih cepat panas akibat dari daya yang terbuang pada resistor. Pengontrol kecepatan mekanis juga lebih mudah kotor.

Pada penelitian ini ESC digunakan untuk Menggerakkan motor. Selain untuk mengatur laju dan arah gerak robot, ESC dapat digunakan sebagai pengatur laju kecepatan robot dengan mudah. ESC juga sebagai supply 5V sebesar 2A pada system yang dirancang Keuntungan lain dari ESC adalah pilihan spesifikasinya yang sangat bermacam-macam.

2.5 Modulasi Lebar Pulsa

Modulasi lebar pulsa atau Pulse Width Modulation (PWM) adalah suatu sinyal yang dikirim dengan frekuensi tetap namun dapat memiliki panjang pulse yang berbeda-beda dalam setiap periodenya. Perbedaan ini biasanya disebut dengan *Duty Cycle*, yaitu perbandingan lama pulse dengan keseluruhan periode sinyal. PWM adalah istilah yang biasanya disebut sebagai sinyal keluaran (*output*) analog.

Penelitian ini menggunakan PWM sebagai sinyal kontrol ESC dan juga motor servo. PWM sangat banyak ditemukan sebagai pengontrol kecepatan motor DC. PWM mengatur kecepatan motor DC menggunakan tegangan rata-rata output dan frekuensi tertentu yang telah disesuaikan. Akan tetapi, dengan menggunakan PWM yang ditambahkan ESC maka penggunaan frekuensi PWM untuk motor DC dan motor servo dapat disamakan.



Gambar 2. 5 Pulse Width Modulation

2.6 Motor Servo



Gambar 2.6 Motor Servo

Motor servo merupakan aktuator yang dapat dikontrol dengan suatu sudut perputaran atau perpindahan linear. Motor servo mempunyai konfigurasi gear sehingga motor servo dapat mudah dikontrol. Motor servo pada umumnya dapat dikontrol mulai dari sudut 0 hingga 180 derajat. Ada pula motor servo yang dapat dikontrol secara kontinu dengan kecepatan yang dapat diatur pula.

Servo motor dikontrol menggunakan sinyal PWM dengan frekuensi 50Hz. Panjang sinyal pulse yang mengontrol bermacam-macam. Standard panjang sinyal pulse untuk mengontrol servo adalah 1000 microsecond sampai dengan 2000 microsecond, namun biasanya sinyal pulse yang mengontrol motor servo berkisar 700 microsecond sampai dengan 2300 microsecond.

Pada penelitian ini digunakan beberapa motor servo sebagai penggerak lengan robot. Dengan motor servo, setiap sudut bagian lengan robot dapat dihitung dengan mudah. Kelemahan dari motor servo adalah harganya yang mahal seiring dengan bertambahnya torsi yang dapat di topang motor servo. Keterbatasan torsi dan ketidak terjangkau harga motor servo seringkali mengakibatkan berubahnya konstruksi lengan robot.

2.6 GPS (Global Positioning System)

Global positioning system atau disingkat GPS adalah suatu teknologi navigasi yang menyediakan informasi posisi dan waktu kapan saja, di mana saja atau dekat dengan bumi, dalam cuaca apapun selama tidak ada yang menghalangi pandangan empat atau lebih satelit GPS. GPS adalah teknologi yang dibuat dan dirawat oleh pemerintah Amerika Serikat serta menjadikannya sebagai sarana yang bebas di akses menggunakan receiver GPS. GPS sangat banyak digunakan dalam bidang militer, komersil, dan juga seluruh pengguna komersil diseluruh dunia. GPS adalah suatu teknologi yang akan menunjukkan posisi receiver yang ada pada suatu benda.

GPS dapat dipasang pada alat apapun atau benda apapun. GPS dapat dipasang pada benda bergerak atau benda diam, semisalnya pada mobil atau pada telepon genggam atau pun alat khusus pelacak lokasi. GPS juga dapat mendeteksi lokasi benda bergerak seperti mobil, sehingga dengan GPS akan dapat diketahui posisi, arah, dan kecepatan benda secara langsung ataupun tertunda, bahkan posisi dapat diketahui secara tiga dimensi. Data pada GPS dapat digunakan untuk mengetahui pergerakan atau rute suatu benda.

Untuk mengetahui lokasi suatu benda, receiver GPS memerlukan dua data, yaitu data waktu dan posisi satelit GPS. Satelit GPS secara terus menerus memancarkan sinyal yang berisi data kapan sinyal tersebut dikirim dan lokasi satelit yang mengirim sinyal tersebut. Dengan memperoleh data kapan sinyal dikirim satelit GPS dan diterima oleh receiver, receiver GPS dapat mengetahui jarak receiver dengan satelit. Data dari tiga satelit dapat diketahui lokasi receiver dan dengan data dari 4 satelit dapat diketahui lokasi receiver secara 3D, yaitu lokasi *longitude* (lintang), *latitude* (bujur), dan *altitude* (ketinggian),

Penelitian ini menggunakan data GPS yang diolah Arduino untuk menentukan posisi Robot. GPS juga digunakan untuk mengetahui rute yang telah ditempuh robot dan juga merencanakan rute. GPS sangat berguna untuk melacak jejak dan posisi robot.

2.8 Sensor Arus

Sensor arus merupakan suatu instrument yang digunakan untuk mengukur arus listrik berdasarkan arus medan. Sensor arus yang umumnya ada di pasaran adalah sensor arus ACS712, sensor ini dapat

digunakan untuk mengukur arus AC dan DC. Berikut spesifikasi sensor arus ACS712:

- Waktu perubahan keluaran = 5 μ s.
- Lebar frekuensi maksimal 80 kHz.
- Error keluaran 1.5% pada suhu kerja 25o C.
- Tahanan konduktor internal 1,2 m Ω .
- Tegangan isolasi minimum 2.1 kVRMS antara pin 1-4 dan pin 5-8.
- Sensitivitas keluaran 185mV/A untuk sensor ACS712 5A
- Sensitivitas keluaran 185mV/A untuk sensor ACS712 5A
- Nilai tegangan keluaran 2.5 V untuk tidak ada arus yang terdeteksi.

Pembacaan nilai arus dengan sensor arus ACS712 dilakukan dengan membaca nilai tegangan output oleh Analog Digital Converter(ADC). Arus ACS7124 = ((NilaiADC - 512)x tegangan referensi/total data bit) / sensitivitas sensor. Misalkan nilai arus 0A maka dan tegangan referensi 2.5 V dibaca oleh sensor arus ACS712 5A maka arus = ((ADC(2.5V)- 512) x 5/1024)/185.

2.9 Motor DC

Motor DC adalah jenis motor listrik yang bekerja menggunakan sumber tegangan DC. Motor DC atau motor arus searah sebagaimana namanya, menggunakan arus langsung dan tidak langsung/direct-unidirectional. Motor DC digunakan pada penggunaan khusus dimana diperlukan penyalaan torque yang tinggi atau percepatan yang tetap untuk kisaran kecepatan yang luas.

Komponen Utama Motor DC

1. Kutub Medan Magnet

Secara sederhana digambarkan bahwa interaksi dua kutub magnet akan menyebabkan perputaran pada motor DC. Motor DC memiliki kutub medan yang stasioner dan kumparan motor DC yang menggerakkan bearing pada ruang diantara kutub medan. Motor DC sederhana memiliki dua kutub medan: kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi bukaan diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet. Elektromagnet menerima listrik dari sumber daya dari luar sebagai penyedia struktur medan.

2. Kumparan Motor DC

Bila arus masuk menuju kumparan motor DC, maka arus ini akan menjadi elektromagnet. kumparan motor DC yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, kumparan motor DC berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi. Jika hal ini terjadi, arusnya berbalik untuk merubah kutub-kutub utara dan selatan kumparan motor DC.

3. Kommutator Motor DC

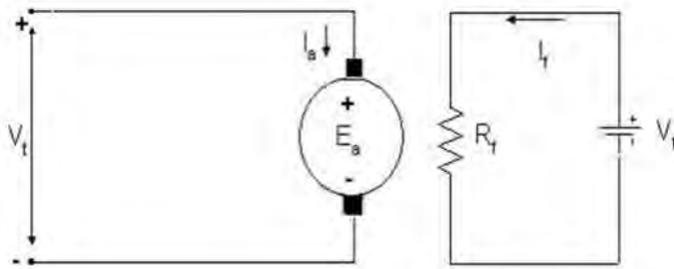
Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk membalikan arah arus listrik dalam kumparan motor DC. Commutator juga membantu dalam transmisi arus antara kumparan motor DC dan sumber daya.



Gambar 2.7 Motor DC

2.10 Torsi Motor

Motor DC penguat terpisah merupakan salah satu dari jenis motor DC yang dapat menambah kemampuan daya dan kecepatan karena memiliki fluks medan (Φ) yang dihasilkan oleh kumparan medan, yang terletak secara terpisah dan mempunyai sumber pembangkit tersendiri berupa tegangan DC sehingga jenis motor DC penguat terpisah ini sangat memungkinkan untuk dapat membangkitkan fluks medan (Φ) bila dibandingkan dengan motor DC magnet permanen. Karena motor DC mempunyai fleksibilitas dalam pengontrolan.



Gambar 2.8 Rangkaian motor DC penguat terpisah

Seperti yang ditunjukkan Gambar 2.8 merupakan gambar rangkaian motor DC penguat terpisah. Sesuai dengan penamaanya, stator dan rotor memiliki penguat masing - masing. Rangkaian ini memiliki perumusan seperti persamaan berikut.

$$V_t = E_a + I_a R_a \quad (2-1)$$

$$V_t = c n \Phi + I_a R_a \quad (2-2)$$

Kemudian pada persamaan (2-2) kita bisa mendapatkan nilai kecepatan motor sehingga perumusanya menjadi seperti perumusan (2-3) berikut.

$$n = \frac{V_t - I_a R_a}{c \cdot \Phi} \quad (2-3)$$

$$P_{out} = P_{mekanik} \quad (2-4)$$

Kemudian berdasarkan persamaan daya - daya yang bekerja pada motor didapat perumusan sebagai berikut.

$$E_a, I_a = 2 n T \cdot n / 60 \quad (2-5)$$

Dari penurunan persamaan (4) dan (5), nilai torsi dapat ditentukan seperti persamaan (6) berikut

$$T = k \cdot \Phi \cdot I_a \quad (2-6)$$

Karena nilai fluks sebanding dengan nilai arus fieldnya, maka persamaanya menjadi persamaan (7) seperti berikut

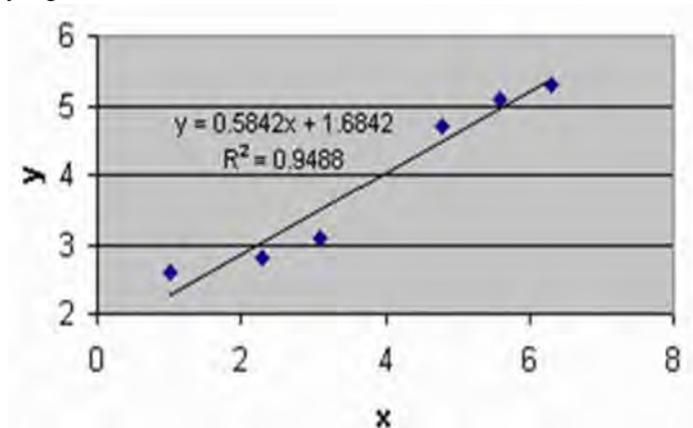
$$T = k \cdot I_f \cdot I_a \quad (2-7)$$

2.11 Regresi

Regresi adalah suatu metode analisis statistik yang digunakan untuk melihat pengaruh antara dua atau lebih variabel.

2.11.1 Regresi Linier

Regresi Linier digunakan untuk menentukan fungsi linier yang paling sesuai dengan kumpulan titik data (x_n, y_n) yang diketahui.



Gambar 2. 9 Regresi linier

Untuk mendapatkan fungsi linier $y=mx+c$, dicari dengan cara berikut.

$$m = \frac{N \sum_{n=1}^N x_n y_n - \left(\sum_{n=1}^N x_n \right) \left(\sum_{n=1}^N y_n \right)}{N \sum_{n=1}^N x_n^2 - \left(\sum_{n=1}^N x_n \right)^2} \quad (2-8)$$

Sedangkan c dapat dicari dengan berikut

$$c = \frac{\sum_{n=1}^N y_n}{N} - m \frac{\sum_{n=1}^N x_n}{N} = \bar{y} - m\bar{x} \quad (2-9)$$

2.11.2 Regresi Polynomial

Regresi Polynomial digunakan untuk menentukan fungsi polinomial yang paling sesuai dengan kumpulan titik data (x_n, y_n) yang diketahui. Regresi polynomial memiliki orde tertentu, semakin besar orde, maka semakin besar ketelitian terhadap sebaran data yang ada. Berikut fungsi pendekatan regresi polynomial.

$$y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \quad (2-10)$$

2.12 Ardupilot Mega (APM)

Flight Controller (FC) merupakan otak atau komputer utama dari *multirotor*. Dalam sebuah FC terdapat beberapa *hardware sensor* seperti *Accelerometer*, *Gyro*, *Magnetometer*, *GPS*. Modul *Flight Controller* membaca seluruh data sensor dan memproses seluruhnya untuk memerintahkan motor untuk berputar sesuai dengan perintah dari FC. Ardupilot merupakan salah satu jenis FC di antara berbagai macam jenis FC. Berikut spesifikasi dari Ardupilot :

- Prosesor Atmega 2560 16MHz.
- 16 M dataflash memory.
- Wireless Telemetry.
- Mavlink protocol.
- Delapan pin analog input.
- Input dan output Pulse Width Modulation.
- Tegangan operasi 5v.

2.13 Estimasi Berat objek

Estimasi berat objek merupakan suatu metode untuk memperkirakan berat objek melalui hubungan besaran besar fisis yang diketahui seperti gaya, energy potensial, energy mekanik, torsi. Pada tugas akhir ini metode yang digunakan untuk mengestimasi berat objek adalah dengan mencari massa dari torsi yang ada. Pada motor DC, torsi pada motor dapat dicari dari besar arus yang mengalir pada motor.

$$T = m \cdot g \cdot l \quad (2-11)$$

Dimana T adalah torsi, m adalah massa, g adalah gravitasi dan l adalah panjang atau jarak. Torsi pada motor DC dapat dicari melalui persamaan

$$T = k \cdot I \quad (2-12)$$

Dimana T adalah torsi, adalah konstanta torsi dan I adalah arus. Sehingga dari persamaan (2-11) dan (2-12) didapatkan

$$k.I = m.g.l \quad (2-13)$$

sehingga massa objek dapat dicari dengan persamaan

$$m = \left(\frac{k.I}{g.l} \right) \quad (2-14)$$

2.14 Software Mission Planner

Software mission planner merupakan software untuk memprogram ardupilot mega. Pemrograman pada mission planner adalah pemrograman *waypoint*. Pada mission planner tinggal pilih lokasi-lokasi yang ingin dituju oleh robot. Sehingga Ardupilot Mega akan mengikuti atau bergerak menuju lokasi yang ditentukan. Fitur – fitur dari *software mission planner* :

- Point-and-click waypoint entry*, menggunakan Google Maps.
- Memilih perintah misi dari menu *drop-down*.
- Mendownload mission log file dan menganalisanya.
- Mengkonfigurasi setting APM.
- Tampilan status *vehicle* dari serial terminal APM.



Gambar 2.10 Tampilan mission planner

2.15 Tinjauan Pustaka

2.15.1 An Automatic Control System for EOD Robot Based on Binocular Vision Position

Penelitian ini membahas mengenai kendali robot penjink bomb dengan menggunakan dua kamera. Prinsip kerjanya berdasarkan stereo kamera. Kendali system dilakukan secara *real time* melalui PC. Kamera menangkap citra, kemudian citra dikirim ke PC dan diolah oleh software MATLAB. Pada penelitian ini hasil yang dicapai adalah penempatan



Gambar 2.11 EOD robot berbasis *binocular vision*

Posisi akhir dari lengan robot yang dicapai. Hasil yang dicapai cukup memuaskan. Error posisi akhir yang dicapai tidak lebih dari 5mm dari sudut posisi yang ingin dicapai. Penggunaan komputer untuk mengolah citra atau gambar mempermudah proses estimasi posisi objek. Permasalahan dari penelitian ini yang perlu di atasi adalah masalah kecepatan dari pergerakan robot untuk mengambil objek. Kecepatan untuk mengambil objek dipengaruhi oleh pengolahan citra. Hal ini dikarenakan untuk mengolah citra memerlukan kecepatan yang pelan. Jika robot bergerak dengan kecepatan yang tinggi maka robot

pengolahan citra atau gambar akan terganggu sehingga robot akan mengalami ketiakkampuan untuk mendeteksi benda. Selain itu juga cara menghindari rintangan perlu dicari method yang lebih efisien.

2.15.2 The Optimization Scheme for EOD Robot Based on Supervisin Control Architecture

Penelitian ini membahas mengenai penanganan objek berbahaya. Penangan objek berbahaya memerlukan efisiensi tinggi dalam pengoperasian robot untuk menghindari halangan. Pada penelitian ini manipulator manipulator di buat *redundant* untu meningkatkan kemampuan menghindari penghalang. Sistem mekanik dan kendali system di buat agar mampu mengambil objek berbahaya di ruang terbatas. Kofigurasi kendali manual dan deteksi ancaman tabrakan berbasis sensor infra merah untuk optimasi pergerakan menghindari penghalang. Kendali system robot ini dilakukan manual dan secara real time.



Gambar 2.12 EOD robot berbasis *Supervision Control Architecture*

2.15.3 Perancangan Kendali Lengan Robot Untuk Mengambil Objek Menggunakan Kamera

Penelitian ini dirancang system robot yang mampu mengukur jarak objek dan mengestimasi posisi objek dengan menggunakan kamera. Cara kerja robot ini dengan memilih benda oleh pengguna melalui image processing. Pada proses *image processing* di dalam penelitian ini, deteksi kontur dan filter Kalman digunakan untuk mengoptimalkan proses pendeteksian posisi objek. Data posisi dan jarak objek tersebut digunakan untuk mengatur kendali gerakan lengan robot yang dimodifikasi dengan *gripper* sebagai *tool* penggenggam pada prototipe *assistive social robot* ketika robot mengambil sebuah objek di atas meja.

2.15.4 Image Fusion for an EOD Robot Wireless Visual Module

Pada penelitian ini dirancang robot yang menggunakan *Principal Components Analysis* (PCA). Transmisi data dilakukan dengan menggunakan komunikasi *wireless*. Hasil simulasi disajikan dengan menggambarkan kinerja algoritma dan mengkonfirmasi kesesuaian untuk implementasi dalam system embedded yang terbatas.

2.15.5 A Single-Hand and Binocular Visual System for EOD Robot

Pada penelitian ini dirancang robot berlengan satu dengan menggunakan sistem *binocular vision*. Keberadaan EOD robot yang dikendalikan secara manual sangat sulit untuk dioperasikan. Berdasarkan teori stereo vision, dilakukan pengembangan EOD robot dengan binocular vision. Gambar di tangkap oleh kamera secara *real time*, kemudian kalibrasi dilakukan dengan metode Zhang. Pencocokan titik dilakukan kemudian didapatkan koordinat titik feature dari benda. Pemrograman dilakukan dengan MATLAB 7.0 dan pencocokan pola *EasyMatch* dari indera mesin software eVision. Gambar ditampilkan secara real time. Pengenalan benda mencurigakan oleh EOD robot dilakukan secara otomatis dan akurat. Percobaan dilakukan dengan mengubah kondisi cahaya, kondisi jarak dan bentuk untuk validasi performa robot secara real time.

2.15.6 Design and Construction of a Didactic 3-DOF Parallel Links RobotStation with a 1-DOF Gripper

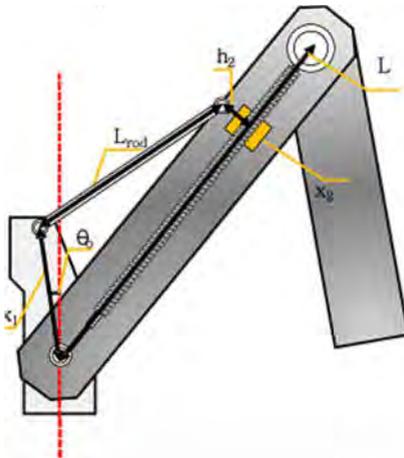
Pada journal ini di bahas mengenai desain lengan robot untuk memudahkan siswa mendesain dan membuat perangkat elektronik yang kuat dan komponen yang murah. Jurnal ini juga dimaksudkan agar siswa dapat memahami metodologi perancangan dan desain industry yang memungkinkan proses manufaktur yang berbeda dan arsitektur robot untuk lingkup proyek tertentu.



Gambar 2.13 Desain lengan robot

2.15.7 Portable Serial Robot Manipulator with Distributed Actuation Mechanism

Pada paper ini dikembangkan mechanism dari sebuah manipulator yang pergerakannya bergerak secara terdistribusi. Dimaksudkan untuk mendapatkan efisiensi yang tinggi dan kapasitas beban yang tinggi. Secara khusus, fungsi lipat diadopsi untuk portabilitas. Dikembangkan manipulator 2 DoF.

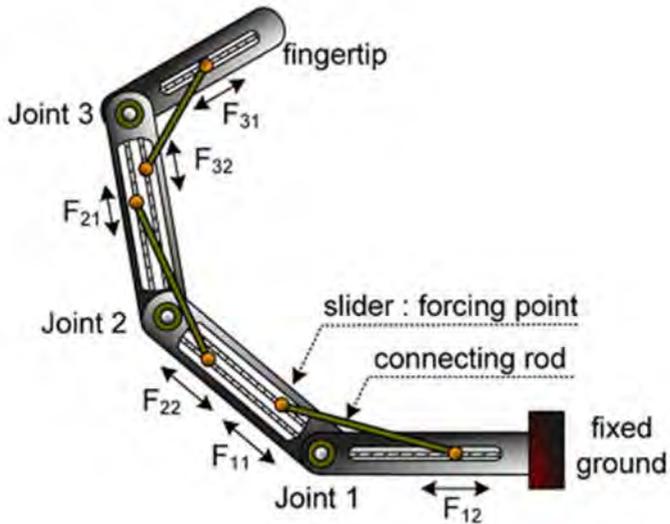


Gambar 2.14 Desain lengan robot Distributed Actuation Mechanism

Dengan desain lengan robot seperti pada gambar 2.14, didapatkan error maksimum sumbu x dan y sebesar $161 \mu\text{m}$ dan $241 \mu\text{m}$ pada beban 13 kg .

2.15.8 Distributed-Actuation Mechanism for a Finger-Type manipulator: Theory and Experiment

Dalam paper ini, method *distributed-actuation* diusulkan untuk mekanisme pada manipulator jari. Berdasarkan gaya terdistribusi secara spasial pada mekanisme *sliding-actuation* menunjukkan bahwa bahwa *distributed-actuation* memberikan kebebasan tambahan untuk mengoptimalkan kinerja manipulator. Untuk memverifikasi efektivitas metode yang diusulkan, maka dikembangkan manipulator jenis jari yang terdiri dari empat link dengan tiga sendi. Hasil percobaan menunjukkan bahwa kekuatan jari dari manipulator yang dikembangkan dapat ditingkatkan secara efektif dan mudah dikelola dengan metode *distributed-actuation*.



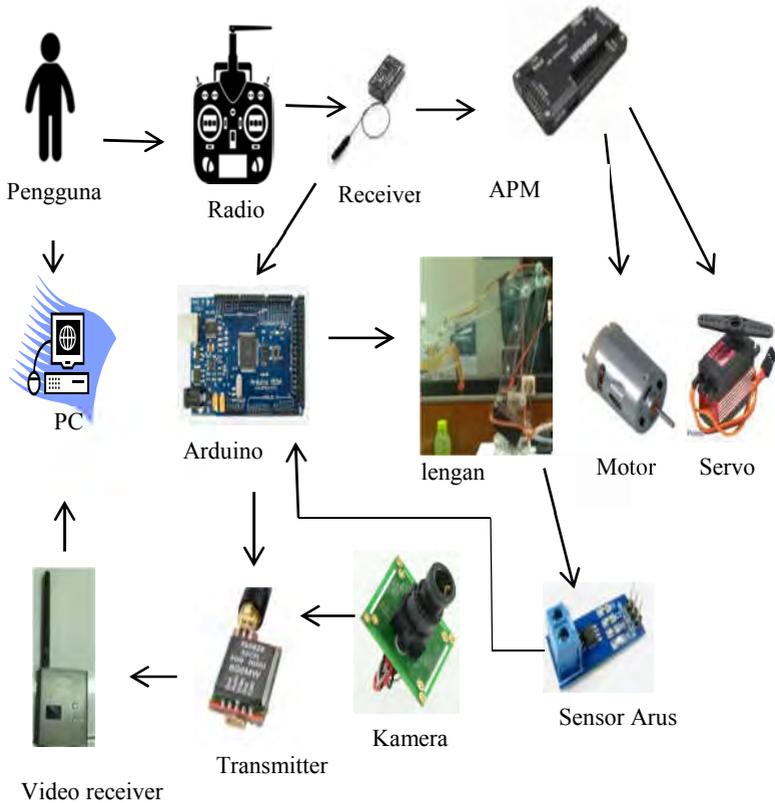
Gambar 2.15 Konsep dari *distrubted-actuation* menggunakan mechanism *sliding-actuation* jari robot

Berdasarkan penelitian, didapatkan keuntungan dari desain yang diusulkan. Antara lain :

1. Kendali jari tidak merubah postur.
2. Optimasi gaya untuk menggerakkan ujung jari menggunakan *redundan* posutr jari
3. Optimasi gaya ujung jari menggunakan *distributed actuatin*.

BAB III PERANCANGAN SISTEM

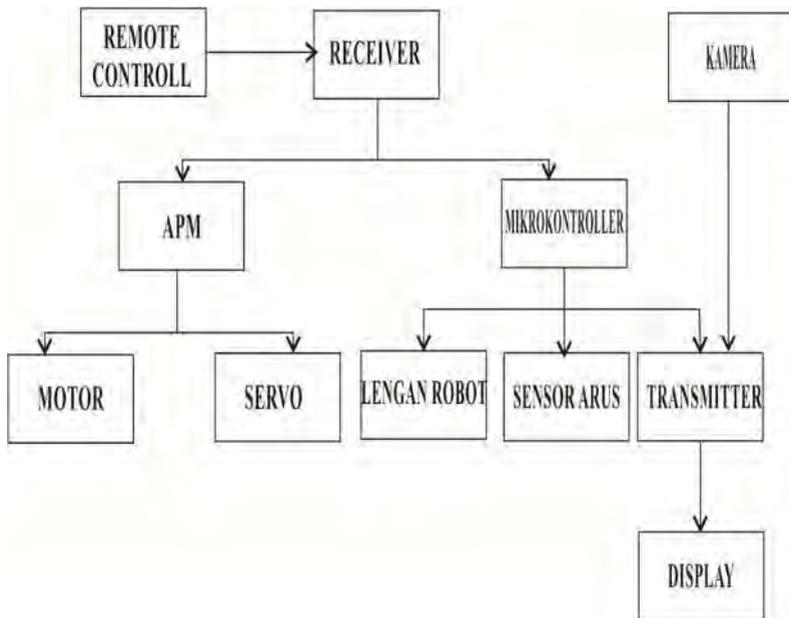
Sistem yang dirancang dalam tugas akhir ini terdiri dari banyak bagian, seperti remote, robot dengan sistem roda mirip tank, kamera, lengan robot, dan lain lain. Remote digunakan sebagai antarmuka pengguna dengan robot untuk mengendalikan robot. Berikut prinsip kerja system yang dirancang.



Gambar 3.1 Ilustrasi sistem kerja robot.

Gambar 3.1 adalah ilustrasi sistem kerja robot. Robot dikendalikan oleh pengguna melalui radio pengendali. Perintah di kirim ke robot dan diterima oleh receiver. Keluaran receiver ada dua tujuan yaitu ke Ardupilot Mega/APM. Sedangkan keluaran satunya menuju ke arduino. Pada APM hanya menerima perintah manual berupa bergerak maju dan belok, selain itu APM juga menerima perintah mode navigasi otomatis. Sedangkan perintah dari remote yang disampaikan ke Arduino adalah perintah untuk kendali lengan. Perintah ini berupa lengan maju, lengan mundur, lengan naik, legan turun dan orientasi lengan. Pada Arduino sendiri terus menerus membaca perubahan arus yang ada pada lengan robot saat tidak mengambil objek ataupun mengambil objek. Informasi arus ini kemudian diolah oleh Arduino dan melalui persamaan estimasi berat objek, di tentukan perkiraan berat objeknya. Informasi ini selanjutnya dikirimim ke pada PC bersamaan dengan video dari kamera. Informasi ini diterima oleh video receiver yang terhubung dengan PC.

3.1 Diagram Blok Sistem



Gambar 3.2 Diagram Sistem Robot

Pada sistem ini robot dikendalikan oleh *user* secara manual melalui *remote control*. Perintah dikirim dari *remote control* kemudian diterima oleh *receiver*. Perintah yang diterima oleh receiver diteruskan ke mikrokontroller dan APM. Pemilihan mode navigasi manual ataupun otomatis dilakukan oleh APM. pergerakan yang dikendalikan oleh APM, dilakukan berdasarkan koordinat GPS. Secara otomatis Robot akan bergerak menuju koordinat yang telah ditentukan pengguna.

Perintah yang diterima oleh mikrokontroller merupakan perintah untuk menggerakkan lengan. Aktuator lengan robot berupa motor servo. Arus yang mengalir ke motor servo akan diukur secara kontinyu oleh sensor arus. Hasil pembacaan arus akan diolah kembali oleh mikrokontroller untuk diestimasi berat objeknya. Adanya kamera yang terhubung ke transmitter tujuannya untuk mengambil gambar. Gambar dikirim melalui transmitter agar robot bisa dikendalikan dari jarak jauh. Arduino Mega juga terhubung dengan transmitter. Data yang diolah oleh Arduino Mega akan ditampilkan ke display. Data yang

ditampilkan oleh display adalah data berat, dan metode pengambilan objek.

3.2 Desain robot

Pada perancangan hardware robot, rangka robot bagian paling bawah adalah rangka mainan Radio Controller jenis monster truck. Pemilihan rangka ini dikarenakan dapat berjalan di lingkungan off-road atau kasar. Perancangan lengan juga disusun secara paralel. Penempatan servo diletakkan pada bagian dasar, sedangkan pada bagian caput menggunakan mini servo dengan kekuatan tor sebesar 1.3 kg.. Penempatan komponen dibagi menjadi tiga bagian atau tiga layer. Layer yang digunakan terbuat dari akrilik dengan tebal 1 cm. Layer dasar merupakan rangka robot sendiri. Pada layer pertama ditempatkan komponen-komponen penggerak robot seperti motor dc untuk menggerakkan robot dan motor servo untuk perubahan arah pergerakan robot sendiri.. Sedangkan layer ke dua dan ketiga menggunakan akrilik. Pada layer ke dua ditempatkan komponen-komponen prosesor dan sensor dari robot seperti Arduino Mega, Arduino Pilot Mega, kompas, sensor arus, transmitter, receiver dan kamera fpv. Sedangkan pada layer ketiga ditempatkan komponen-komponen aktuator atau penggerak lengan robot.



Gambar 3.3 Gambar rancangan robot tampak depan

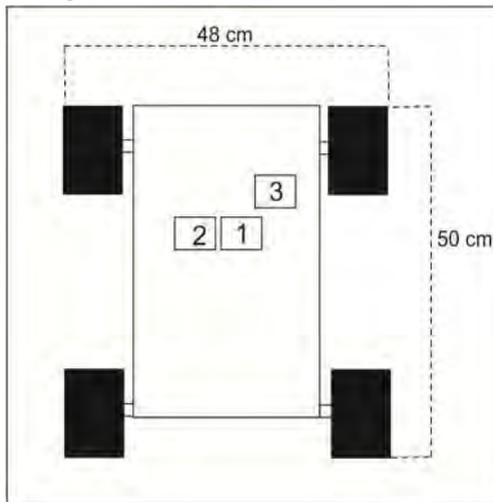


Gambar 3.4 robot tampak dari sisi kiri



Gambar 3.5 robot tampak sisi kanan

Gambar 3.6 adalah ilustrasi penempatan komponen pada rangka robot. Pada layer paling dasar komponen ditempatkan beberapa komponen sebagai berikut :

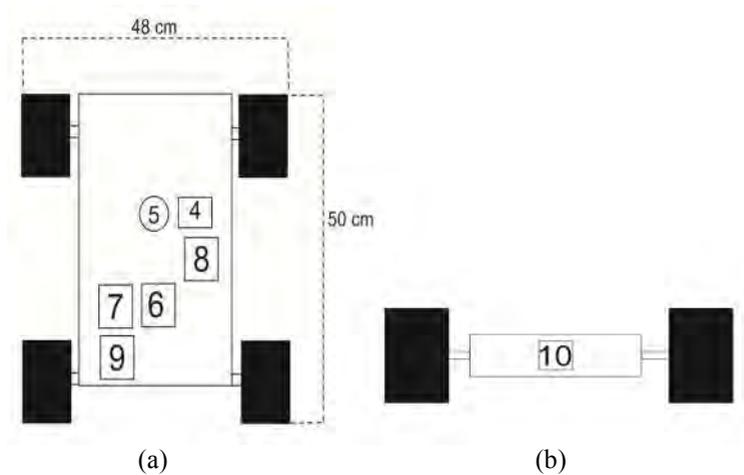


Gambar 3.6 Layer dasar rangka robot

1. Electrical Speed Controller
2. Motor DC
3. Motor servo

Pada Bagian dasar, ditempatkan komponen penggerak rangka robot beserta mekaniknya. ESC berfungsi menggerakkan motor DC ketika diberi perintah atau maju. Motor DC terhubung dengan ESC. Motor DC berfungsi menggerakkan motor DC ke depan atau ke belakang. Motor DC yang digunakan memerlukan tegangan supply 5V DC sedangkan arus yang diperlukan dapat mencapai 20A.

Sedangkan motor servo berfungsi mengendalikan arah pergerakan robot ke kiri atau ke kanan. Ketika motor servo bergerak, maka motor servo akan mendorong roda depan ke arah serong terhadap badan robot. Roda depan akan bergerak serong ke kiri atau ke kanan. Motor servo yang digunakan adalah motor servo standard buatan hobbyking. Motor servo ini memerlukan tegangan untuk menggerakkan servo sebesar 5V DC



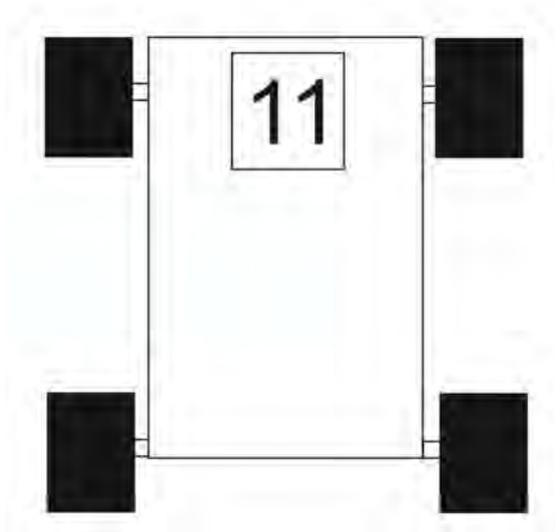
Gambar 3.7 Ilustrasi Layer dua rangka robot

Pada layer 2 terdapat komponen yang mengendalikan robot secara terprogram. Komponen yang digunakan pada layer 2 sebagai berikut :

4. Ardupilot
5. GPS dan kompas

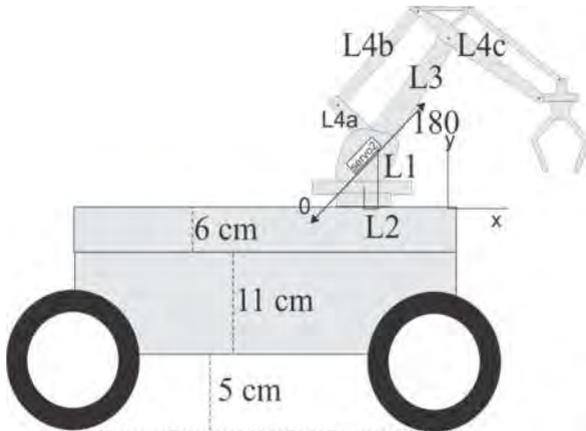
6. Arduino Mega
7. Sensor arus
8. Receiver
9. Transmitter
10. Kamera

Ardupilot digunakan untuk navigasi secara semi otomatis. Ardupilot dilengkapi sensor kemiringan sehingga dapat mengetahui kemiringan robot. Ardupilot dihubungkan ke GPS dan kompas untuk mengetahui arah orientasi pergerakan robot. Ardupilot ini merupakan sebuah modul yang memberikan kemudahan bagi pengguna untuk navigasi sistem. Sedangkan Arduino mega berfungsi membaca dan menentukan nilai pembacaan sensor arus pada servo lengan robot. Sedangkan receiver untuk menerima perintah dari robot. Transmitter terhubung dengan kamera dan arduino mega untuk mengirimkan informasi menuju ke display. Perancangan robot dilengkapi dengan kamera sehingga robot dapat digunakan untuk mengambil objek berbahaya dari tempat yang jauh dan aman.



Gambar 3.8 Ilustrasi robot Layer tiga

Pada layer 3 terdapat lengan robot. Penempatan lengan robot berada pada layer paling atas dikarenakan lengan robot mudah bergerak di bagian paling atas dari layer.

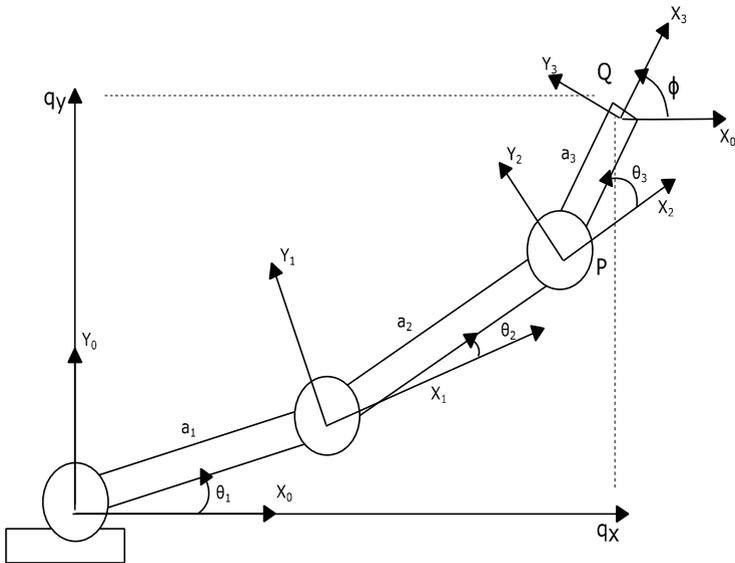


Gambar 3.9 Ilustrasi ukuran robot

Gambar 3.9 adalah ilustrasi desain rangka robot yang tampak dari sebelah kanan. Pada ilustrasi Gambar tersebut terdapat ukuran robot yang dirancang pada penelitian ini. Rangka Mobile robot dirancang dengan memakai rangka menyerupai mobil off-road sehingga dapat bergerak di lingkungan kasar. Robot digerakkan menggunakan sebuah motor dc.

3.3 Perancangan Lengan Robot

Perancangan lengan robot merupakan bagian dari pengembangan *semi autonomous mobile robot*. Pada desain lengan robot sebelumnya, perancangan lengan robot menggunakan rancangan lengan robot serial. Di mana tiap servo disusun serial terhadap joint. Lengan robot yang digunakan terdiri dari empat *Degree of Freedom* (DoF). Kelebihan dari lengan robot ini dapat bergerak fleksibel namun dalam hal kekuatan atau kemampuan lengan untuk mengangkat benda masih sangat terbatas. Kemampuan objek yang mampu diangkat adalah sekitar 200 gram. Dan hal ini perlu pengembangan lagi agar lengan robot dapat mengangkat objek yang lebih berat. Konstruksi lengan robot seperti yang diilustrasikan pada gambar 3.10



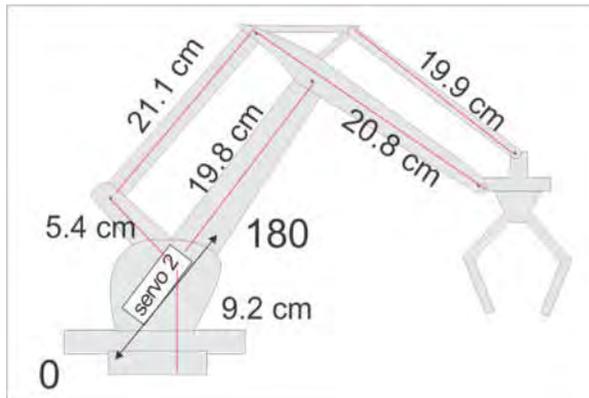
Gambar 3.10 konstruksi lengan robot serial

Konstruksi lengan robot secara serial akan membebani servo base. Hal ini dikarenakan torsi yang dibebankan pada servo pada base semakin besar. Rumus torsi adalah

$$T = \text{massa} \times \text{gravitasi} \times \text{panjang}$$

Dengan menempatkan servo secara serial maka berat lengan akan terakumulasi dan semakin berat. Hal ini membuat torsi yang dibutuhkan untuk menggerakkan lengan juga semakin besar. Semakin besar torsi yang dibutuhkan untuk menggerakkan lengan maka kemampuan lengan untuk mengangkat beban juga semakin kecil. Sehingga desain lengan robot perlu diubah dengan prinsip *parallel mechanism*.

Lengan robot terdiri dari 3 servo yang ditempatkan di dasar lengan robot, bahan yang digunakan untuk lengan robot adalah akrilik dengan tebal 3 mm dan 5 mm. Sambungan joint menggunakan prinsip tuas sehingga lengan dapat digerakkan dari bagian base. Servo yang digunakan adalah servo Power HD 20 kg dan Turnigy 1270 HV 30 kg.



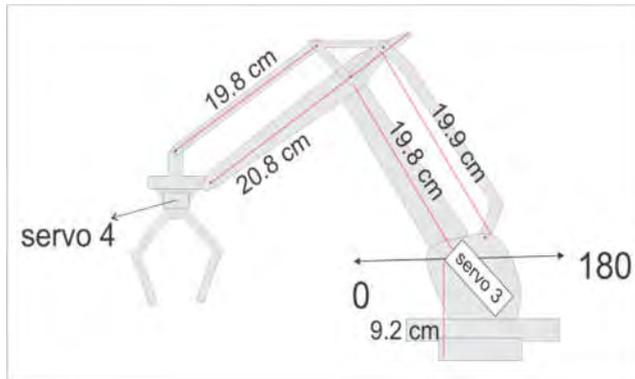
Gambar 3.11 Ilustrasi lengan robot tampak sisi kanan

Gambar 3.11 merupakan gambar ukuran rancangan lengan robot tampak dari sebelah kanan. Penempatan servo 2 juga dibuat agak miring. Kemiringan dari penempatan servo dua ini sebesar 60 derajat. Hal ini dimaksudkan agar pergerakan dari servo sendiri lebih mudah dan lengan robot lebih mudah dikendalikan oleh motor servo. Sedangkan sudut posisi servo seperti yang ditunjukkan pada gambar 3.9. Dimana sudut servo semakin kebawah semakin kecil, sedangkan sudut servo semakin keatas semakin besar.

Berikut spesifikasi servo 2

- Model: LF-20MG
- Torsi(4.8V): 16.5 kg-cm (229.1 oz/in)
- Torsi(6.6V): 20.0 kg-cm (277.7 oz/in)
- Kecepatan: 0.18 sec (4.8V) | 0.16 sec (6.6V)
- Tegangan operasi : 4.8 ~ 6.6 DC Volts
- Berat: 60 g (2.12 oz)
- Motor : DC Motor
- Suhu operasi : -20°C~60°C
- Frekuensi kerja : 1520µs / 333hz
- Ukuran : 40.7 x 20.5 x 39.5 mm (1.60 x 0.80 x 1.55 in)

Pada perancangan sistem ini, servo yang digunakan adalah servo dengan merk power HD. Servo ini dilengkapi dengan regulator.



Gambar 3.12 lengan robot tampak dari kiri

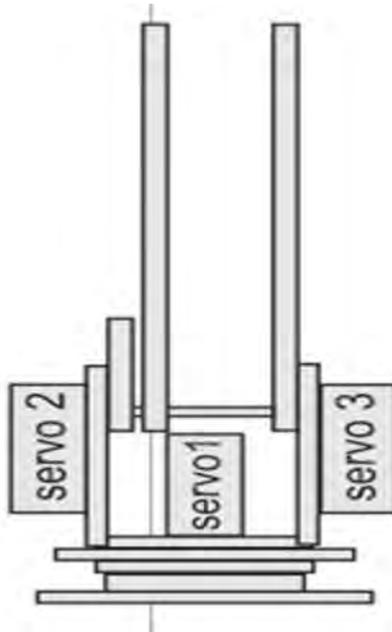
Pada Gambar 3.12 adalah ilustrasi desain lengan robot tampak dari sisi kiri. Serta penempatan sudut juga dibuat horizontal. Dimana sudut servo jika semakin ke depan semakin kecil atau sudut servo sebesar 0 derajat ketika ke depan dan 180 derajat ketika ke belakang. penempatan Berikut spesifikasi servo.

- Tegangan operasi: 6v ~ 7.2v
- Berat: 170 grams
- Ukuran: 59.5 x 29.5 x 54.3 mm
- Torsi 6.0V: 35 kg/cm
- Torsi @ 7.2V: 40 kg/cm
- kecepatan pada tegangan 6.0V: 0.20 sec / 60° tanpa beban
- Kecepatan pada tegangan 7.2V: 0.18 sec / 60° tanpa beban
- Gear: Metal gear / Dual ball bearing

Pada gambar 3.13 merupakan ilustrasi penempatan servo dari depan. Pada gambar 3.13 servo 1 ditempatkan pada bagian paling dasar. Servo 1 bertugas untuk merubah orientasi lengan robot. Sehingga arah lengan robot dapat menghadap ke depan, serong ataupun ke samping. Servo yang digunakan sama seperti servo 2 dengan spesifikasi

- Tegangan operasi: 6v ~ 7.2v
- Berat: 170 grams
- Ukuran: 59.5 x 29.5 x 54.3 mm
- Torsi 6.0V: 35 kg/cm

- Torsi @ 7.2V: 40 kg/cm
- kecepatan pada tegangan 6.0V: 0.20 sec / 60° tanpa beban
- Kecepatan pada tegangan 7.2V: 0.18 sec / 60° tanpa beban
- Gear: Metal gear / Dual ball bearing



Gambar 3.13 Ilustrasi lengan robot tampak depan

Pada penempatan servo 1 atau servo rotasi ini, di 90 derajat ketika lengan robot menghadap ke depan. Sedangkan lengan robot menghadap ke arah kiri ketika servo bernilai nol derajat. Sedangkan ketika lengan robot menghadap ke arah kanan, servo 1 bernilai 180 derajat. Pergerakan dari servo ini tidak masuk dalam inverse kinematic lengan. Untuk merubah orientasi lengan, maka diperlukan komponen *bearing*. Komponen ini berfungsi menjaga titik pusat orientasi lengan. Sehingga ketika lengan berubah orientasinya atau berotasi, maka rotasi lengan berada pada titik pusat yang sama. Fungsi bearing ini juga untuk menghaluskan pergerakan lengan ketika berotasi.

3.4 Remot kontrol dan Receiver

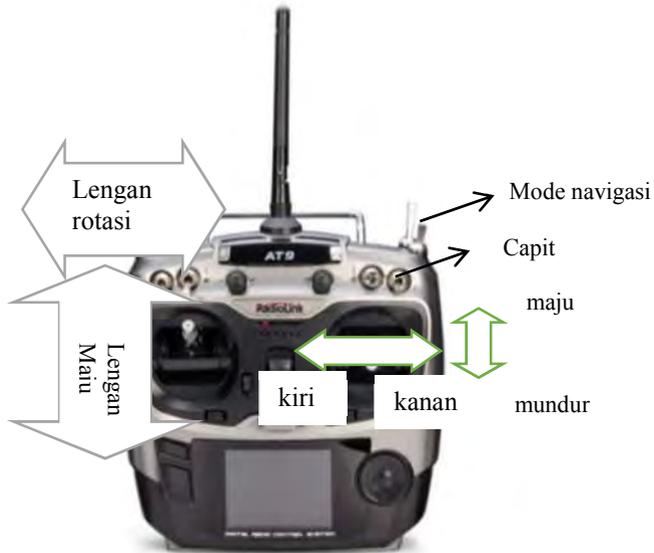


Gambar 3.14 Remote control



Gambar 3.15 Receiver

Komponen radio kendali dan receiver yang digunakan pada perancangan system robot ini adalah RadioLink AT9 dengan channel dapat mencapai 9 channel. Sedangkan receiver yang digunakan pada system robot ini adalah receiver RadioLink R9D. Remote control digunakan oleh pengguna untuk mengendalikan robot. Sedangkan receiver bertugas menerima perintah yang dikirim oleh radio transmitter. RadioLink AT9 adalah transmitter sinyal sedangkan receiver RD9 adalah receiver sinyal dari transmitter Remote control. Receiver ini dapat menerima sinyal dari 10 kanal, namun pada penelitian ini hanya akan digunakan receiver 9 channel. Transmitter dan receiver ini bekerja pada frekuensi 2.4 GHz. Pada receiver, sinyal yang diterima dan dikeluarkan menuju Ardupilot dan Arduino Mega berupa sinyal Pulse Width Modulation. Sinyal ini memiliki rentang antara 1000 microsecond hingga 2000 microseconds.

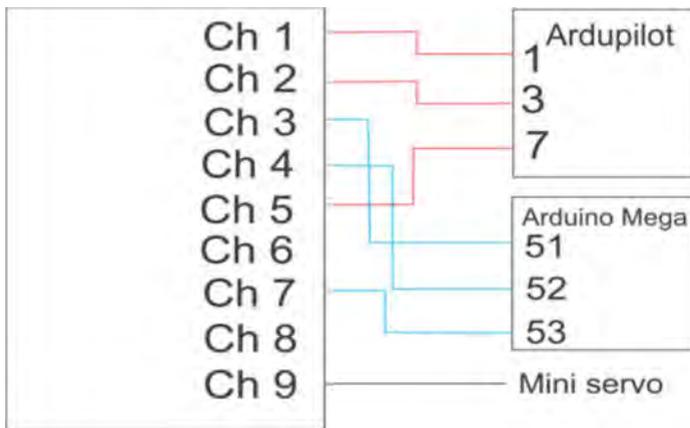


Gambar 3.16 ilustrasi perintah remote ke receiver

Gambar 3.16 adalah penjelasan kontrol yang digunakan dalam penelitian ini. Remote kontrol digunakan untuk mengontrol navigasi robot, lengan robot, dan juga sebagai joystick untuk memilih benda yang akan didekati melalui kamera. Pada remot kontrol terdapat saklar yang

digunakan untuk mengganti mode diantara mode manual, mode otomatis mengikuti titik, dan mode otomatis pulang.

Receiver digunakan untuk menerjemahkan membaca sinyal yang dikirim oleh remot. Sinyal pertama yang dibaca adalah sinyal PWM dari salah satu kanal pada remot yang kemudian diterjemahkan menjadi pemilihan mode navigasi. Kemudian sinyal kedua adalah sinyal PWM dari kanal yang kemudian diterjemahkan menjadi perintah menggerakkan lengan menggunakan *inverse kinematics*. Ketiga adalah sinyal PWM dari dua kanal untuk menggerakkan motor robot. Kemudian sinyal PWM dari satu kanal untuk menggerakkan kamera. Penjelasan mengenai software akan dijelaskan pada bagian Perancangan software. Penjelasan mengenai pengkabelan antar hardware akan dijelaskan pada bagian hardware berikutnya. Berikut skema ilustrasi receiver ke Ardupilot maupun Arduino Mega.



Gambar 3.17 *Wiring receiver* ke Ardupilot dan Arduino Mega

Seperti pada gambar 3.17, tidak semua channel pada receiver digunakan. Hanya beberapa yang digunakan. Beberapa channel tersambung dengan Ardupilot sedangkan beberapa channel juga terhubung dengan Arduino Mega. Pada channel 9 langsung terhubung dengan mini servo. Hal ini dikarenakan jika sinyanya pada receiver channel 9 dibaca oleh Arduino mega, maka nilainya berubah ubah dan tidak konstan. Sehingga receiver channel 9 langsung terhubung dengan mini servo.

3.4 Sensor Arus

Pada sistem ini terdapat dua buah hall effect sensor ACS712. Sensor arus pertama untuk mengukur arus pada servo 2. Sensor arus pertama merupakan sensor produksi innovative-electronics dengan kemampuan deteksi arus 5A. Sensor ini dilengkapi dengan rangkaian penguat yang dapat diatur penguatannya dan pengkondisi sinyal sehingga dapat membaca perubahan arus yang sangat kecil. Sensor arus pertama digunakan pada servo 2 karena arus maksimum yang dapat dibutuhkan oleh servo 2 sekitar 2.5A. Kemudian sensor arus yang ke 2 untuk mengukur servo ke 3. Sensor arus ke 2 dapat mendeteksi arus hingga 20 A. Pada servo ke 3 digunakan sensor arus ACS712 20 A karena arus maksimal yang mengalir pada servo ke 3 sebesar 7.5 A. Sensor arus terdiri dari sebuah chip yang di dalamnya telah ada rangkaian pengondisi sinyal.

Berikut ini adalah spesifikasi singkat sensor arus 5A dan 20A:

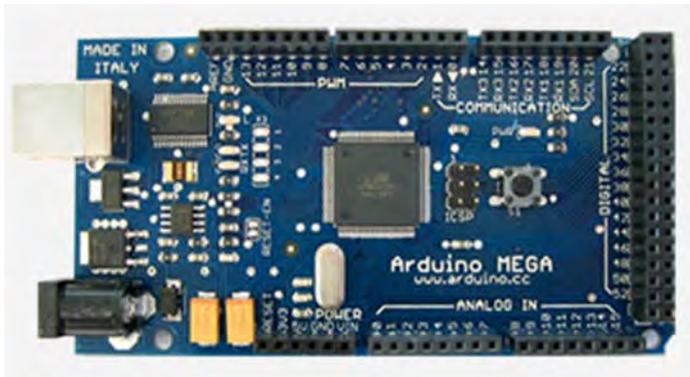
Catu daya	5V
Rise time output	5u s
Maksimal bandwidth	50 kHz
Total kesalahan	1.5% pada suhu operasi 25° C
Tahanan konduktor	1.2 M ohm
Tegangan isolasi	2.1 kv RMS
Sensitivitas arus	185mV/A pada sensor 5A dan 100mV/A pada 20A



Gambar 3.18 Sensor Arus

Pada dasarnya spesifikasi sensor arus 5A dan 20 A tidaklah jauh berbeda, yang membedakan adalah sensitivitas pembacaan arus. Sensor arus ACS712 5A lebih sensitive terhadap pembacaan arus yang kecil.

3.5 Arduino Mega



Gambar 3.19 Arduino Mega

Arduino Mega digunakan untuk mengontrol gerakan lengan robot. Pada arduino mega terdapat program inverse kinematic lengan, sehingga dapat memudahkan pengguna dalam mengontrol gerakan lengan. Arduino mega mengontrol gerakan lengan dengan cara membaca data dari receiver kemudian diolah dengan inverse kinematic.

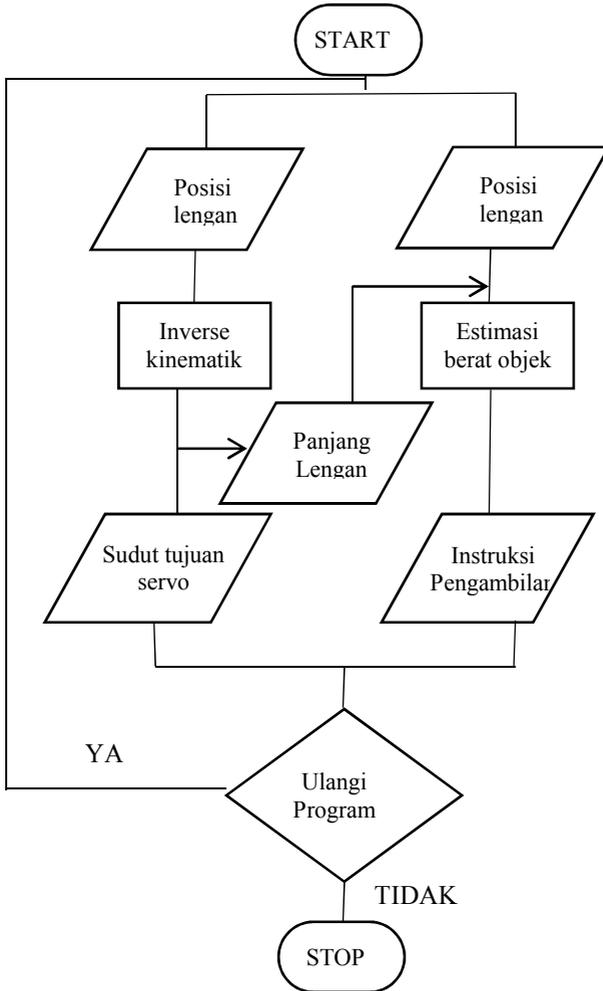
Berikut ini adalah spesifikasi singkat Arduino Mega [1]:

Microcontroller	ATmega1280
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (15 diantaranya menghasilkan output PWM)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	128 KB (4 KB dari memori digunakan sebagai bootloader)
SRAM	8 KB
EEPROM	4 KB

Clock Speed

16 MHz

Pada Arduino Mega, perancangan software berupa membaca arus, pergerakan lengan dan estimasi berat objek.

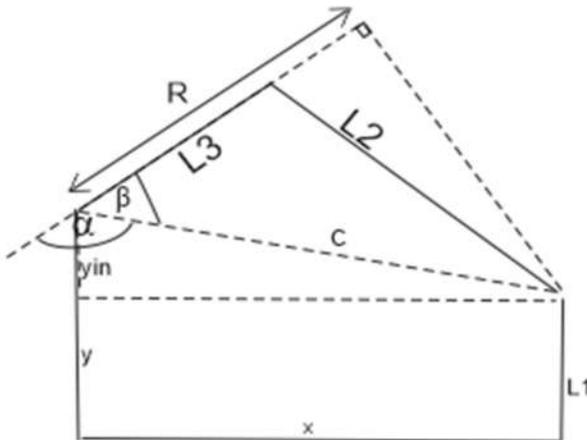


Gambar 3.20 Alur program Arduino Mega

Gambar 3.20 merupakan alur program pada Arduino Mega. Secara garis besar pada Arduino Mega ada 3 proses yang dijalankan yaitu inverse kinematic, baca arus dari sensor arus dan estimasi berat objek. Arduino mendapatkan input berupa perintah posisi lengan, kemudian perintah posisi diolah melalui inverse kinematic. Panjang lengan yang dituju selanjutnya dikeluarkan dalam pergerakan lengan dan. Panjang lengan juga mempengaruhi estimasi berat objek dikarenakan berat objek juga tergantung pada panjang lengan. Arduino Mega juga mendapatkan masukan dari sensor arus yang selanjutnya di olah untuk mendapatkan estimasi berat objek. Panjang lengan juga dipertimbangkan untuk memilih persamaan estimasi berat objek yang tepat.

3.5.1 Inverse Kinematik

Perancangan gerak lengan robot umumnya menggunakan metode *inverse kinematic*. Metode *inverse kinematic* digunakan untuk menentukan besar sudut aktuator berdasarkan posisi *end-effector* dan panjang setiap link. Untuk menyelesaikannya pertama lengan robot dimodelkan sebagai sebuah garis



Gambar 3.21 Pemodelan lengan dalam garis

Gambar 3.21 adalah pemodelan lengan dengan garis yang membentuk segitiga. Dari pemodelan di atas akan dicari penurunan

rumus trigonometri. Nilai L1 adalah tinggi base, L2 adalah lengan yang terhubung pada base, sedangkan L3 adalah lengan yang terhubung pada L2, jika diibaratkan lengan manusia, L2 adalah lengan atas, sedangkan L3 adalah lengan bawah. Kemudian posisi y dicari yin, yin merupakan jarak vertical lengan terhadap base robot, sehingga

$$y_{in} = (y - L1) \tag{3-1}$$

Setelah yin dicari, selanjutnya mencari nilai C. C adalah jarak *end-effector* terhadap pusat. Sehingga nilai C didapatkan

$$C = \sqrt{x^2 + y_{in}^2} \tag{3-2}$$

Nilai R merupakan panjang garis dari titik yang tegak lurus terhadap nilai pusat. Sehingga nilai R dapat dicari dengan menggunakan hukum cosinus.

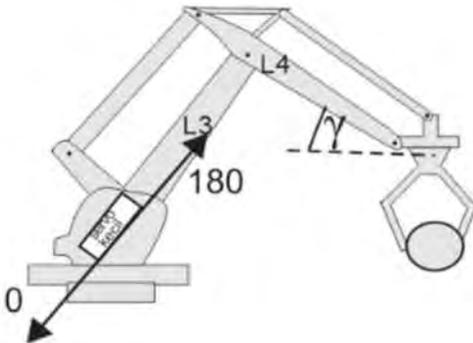
$$1^2 = \left(\frac{L3}{L2}\right)^2 + y_{in}^2 + x^2 - 2 \times \left(\frac{L3}{L2}\right) \times C \cos\beta \tag{3-3}$$

$$\cos\beta = -\cos\alpha = \frac{-R}{c} \tag{3-4}$$

Sehingga nilai R dapat dicari dengan persamaan

$$1 - \left(\frac{L3}{L2}\right)^2 - y_{in}^2 - x^2 = 2 \times \left(\frac{L3}{L2}\right) \times R \tag{3-5}$$

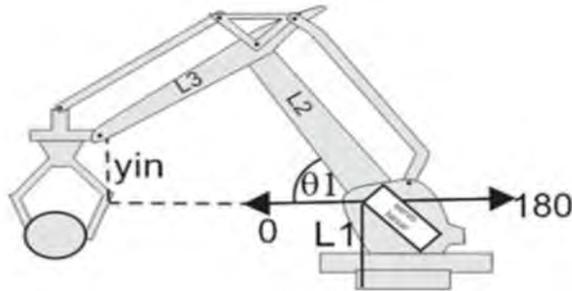
$$R = \frac{1 - y_{in}^2 - x^2 - \frac{L3^2}{L2^2}}{2 \frac{L3}{L2}} \tag{3-6}$$



Gambar 3.22 Pemodelan pergerakan sudut servo 2

$$\theta_2 = \text{theta servo kecil} = \gamma - \pi \tag{3-7}$$

$$= \sin^{-1} \left(\frac{R}{C} \right) - \pi$$



Gambar 3.23 Pemodelan pergerakan sudut servo 3

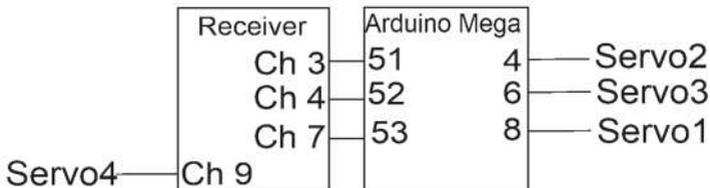
$$\sin \theta_1 = y_{in} + \frac{L_3}{L_2} \sin(\pi - \theta_2)$$

$$\sin \theta_1 = \frac{y-L_1}{L_2} + \frac{L_3}{L_2} \sin(\theta_2)$$

$$\theta_1 = \sin^{-1} \left(\frac{y+L_3 \sin \theta_2 - L_1}{L_2} \right) \quad (3-8)$$

3.5.2 Kendali Lengan Robot

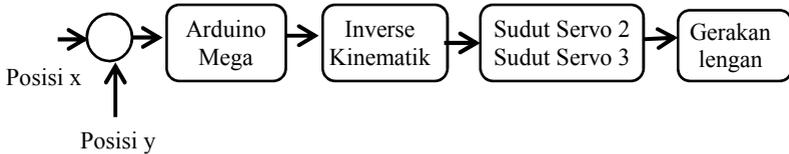
Kendali Lengan robot dilakukan secara manual oleh pengguna. Perintah untuk menggerakkan lengan robot dikirimkan melalui radio pengendali dan diterima oleh receiver. Selanjutnya perintah yang diterima receiver di baca oleh Arduino Mega.. Berikut Hubungan Receiver dengan Arduino Mega dan servo lengan robot.



Gambar 3.24 Ilustrasi input output hardware kendali lengan

Pada gambar 3.24 Receiver menerima perintah dari radio pengendali, selanjutnya chanel chanel tiap receiver masing masing

akan memberi perintah pada Arduino Mega. Channel 3 merupakan chanel untuk memerintahkan lengan maju. Chanel 4 merupakan chanel untuk memberi perintah rotasi. Sedangkan chanel 7 untuk memberi perintah lengan ke bawah ataupun ke atas.

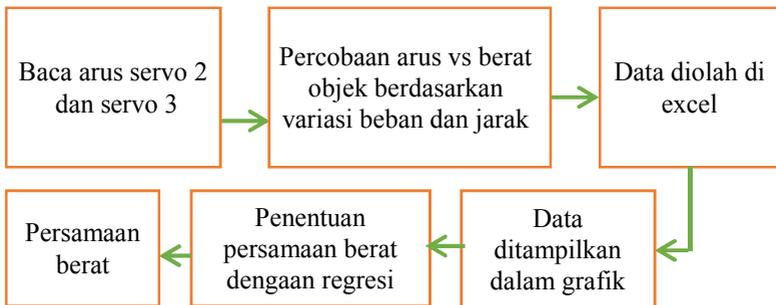


Gambar 3.25 Kontrol Lengan Robot

Pada gambar 3.25 merupakan ilustrasi kendali lengan robot. Input yang dimasukkan berupa posisi x dan y. Posisi x dan y diterima oleh Arduino Mega. Pada Arduino mega selanjutnya dilakukan proses inverse kinematik. Pada proses inverse kinematik ini diolah input x dan y dan diubah menjadi perintah sudut untuk servo 2 dan 3. Sudut ini selanjutnya di inputkan ke servo 2 dan servo 3, sehingga keluaran berupa gerakan lengan sesuai posisi x dan y yang diinginkan. Pada gambar 3.25 merupakan open loop, hal ini dikarenakan kontrol lengan robot berupa manual.

3.5.3 Estimasi Berat Objek

Pada Arduino Mega, terdapat proses estimasi berat objek. Berikut alur perancangan dari penentuan estimasi berat objek.



Gambar 3.26 Alur perancangan persamaan berat objek

Gambar 3.26 merupakan alur penentuan proses penentuan persamaan berat objek. Pertama arus pada servo 2 dan 3 di baca terlebih dahulu.. Setelah arus pada servo 2 dan 3 terbaca, selanjutnya dilakukan percobaan untuk mengetahui perubahan arus ketika terjadi perubahan jarak dan variasi berat objek. Selanjutnya data diolah dengan bantuan Microsoft excel. Dari Data yang diolah dalam excel (Pengujian di BAB IV, 4.1) didapatkan persamaan regresi linier. Pada sistem ini digunakan persamaan mulai dari panjang lengan 20 cm hingga 30 cm. Hal ini dikarenakan perubahan arus servo 3 tidak berbeda jauh ketika panjang lengan 16 cm, 18 cm dan 20 cm. Berikut persamaan yang digunakan untuk estimasi berat objek.

Pada panjang lengan 20 cm, persamaan pada servo 3

$$y = 1.4574x - 0.1052 \quad (3-11-1)$$

Sedangkan pada servo 2

$$y = 0.3226x + 0.1143 \quad (3-11-2)$$

Pada panjang lengan 22 cm, persamaan pada servo 3

$$y = 0.7708x - 0.031 \quad (3-12-1)$$

Sedangkan pada servo 2

$$y = 0.3474x + 0.0648 \quad (3-12-2)$$

Pada panjang lengan 24 cm, persamaan pada servo 3

$$y = 0.5349x + 0.0653 \quad (3-13-1)$$

Sedangkan pada servo 2

$$y = 0.322x + 0.0796 \quad (3-13-3)$$

Pada panjang lengan 24 cm, persamaan pada servo 3

$$y = 0.453x + 0.0024 \quad (3-14-1)$$

Sedangkan pada servo 2

$$y = 0.3773x - 0.0345 \quad (3-14-2)$$

Pada panjang lengan 26 cm, persamaan pada servo 3

$$y = 0.453x + 0.0024 \quad (3-15-1)$$

Sedangkan pada servo 2

$$y = 0.3773x - 0.0345 \quad (3-15-2)$$

Pada panjang lengan 28 cm, persamaan pada servo 3

$$y = 0.1729x + 0.2615 \quad (3-16-1)$$

Sedangkan pada servo 2

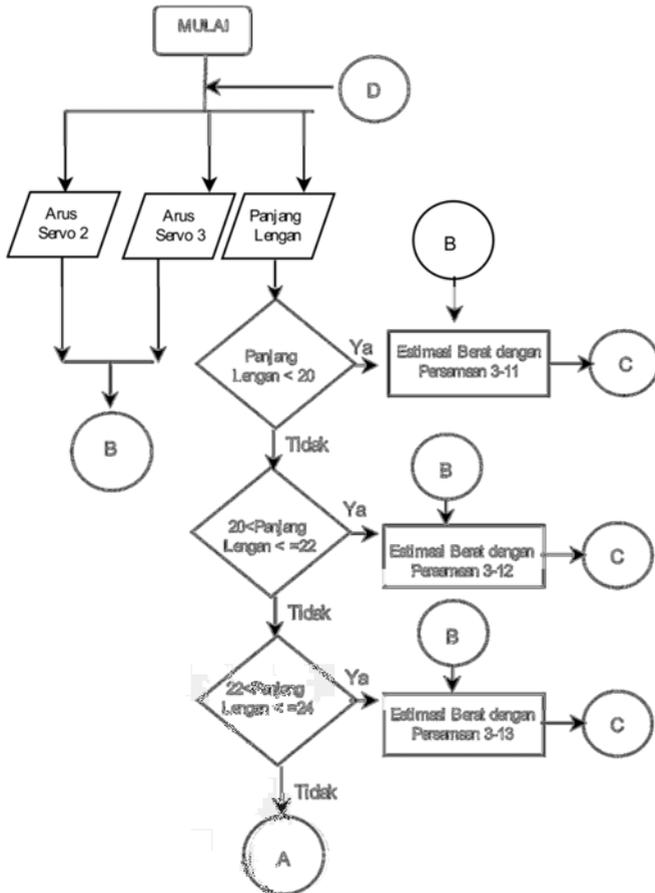
$$y = 0.3151x + 0.0755 \quad (3-16-2)$$

Pada panjang lengan 30 cm, persamaan pada servo 3

$$y = 0.1855x + 0.1828 \quad (3-17-1)$$

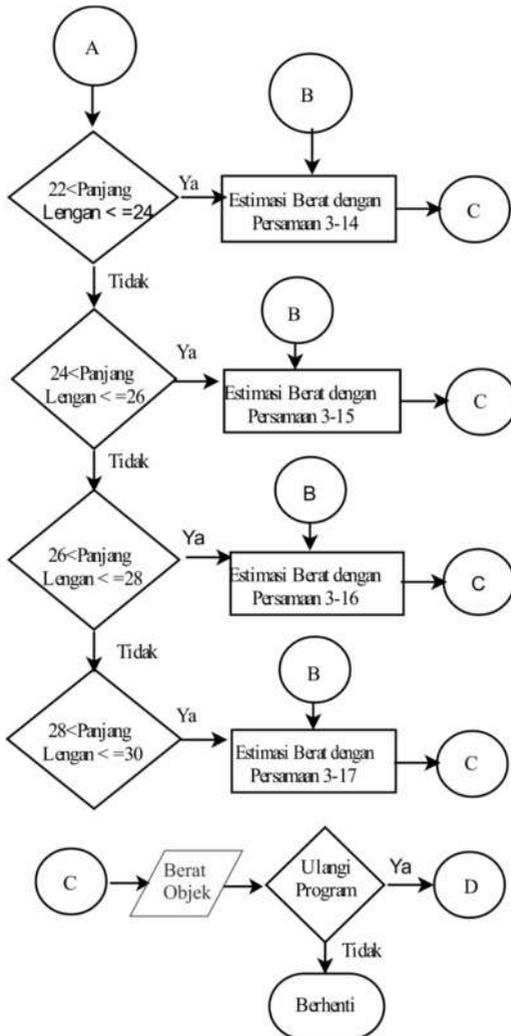
Sedangkan pada servo 2
 $y = 0.314x + 0.1126$

(3-17-2)



1/2

Gambar 3.27 (a) Algoritma penentuan estimasi berat objek pada Arduino



Gambar 3.27(b) Algoritma penentuan estimasi berat objek

Pada gambar 3.27 adalah algoritma untuk estimasi berat objek. Estimasi berat objek tergantung pada panjang lengan. Setiap panjang lengan memiliki persamaan berat objek sendiri. Jika panjang lengan

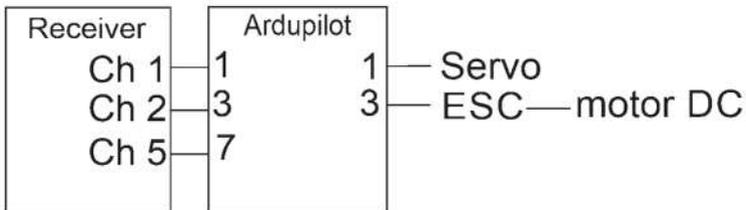
tidak sesuai dengan persamaan maka akan ke persamaan berikutnya sesuai dengan panjang lengan. Pada persamaan yang dituju tentunya juga dimasukkan arus yang terbaca pada servo.

3.6 Perancangan Navigasi

Pada perancangan navigasi, dilakukan dua tipe perancangan. Yaitu perancangan secara hardware dan software. Pada perancangan hardware, dilakukan pengkabelan pada receiver ke Ardupilot Mega, lalu ke servo dan ESC.

3.6.1 Perancangan Perangkat Keras Navigasi

ESC adalah Electric Speed Control. ESC digunakan untuk menggerakkan motor DC. Pada ESC terdapat regulator arus dan tegangan sehingga dapat mengatur penggunaan daya dan kecepatan motor DC. ESC dikontrol menggunakan PWM. PWM yang mengontrol kecepatan motor DC di keluarkan oleh Ardupilot. Motor servo digunakan untuk *steering* robot. Motor servo dikendalikan melalui sinyal PWM Pada tugas akhir ini terdapat satu motor DC dan satu ESC. Berikut ini adalah gambar pengkabelan antara Arduino Uno dan ESC



Gambar 3.28 Perancangan hardware navigasi robot

Pada gambar 3.28 channel 1 *receiver* terhubung dengan dengan channel input 1 pada ardupilot. Channel 1 merupakan channel untuk mengendalikan *steering* robot. Pergerakan yang diatur adalah gerak belok ke kiri atau ke kanan. Perintah yang diterima oleh ardupilot merupakan perintah dalam bentuk PWM. Pada channel 2 *receiver*, perintah yang diterima oleh *receiver* berupa sinyal PWM. Perintah yang dikirim adalah perintah gerak maju atau gerak mundur. Perintah yang diterima oleh receiver dikeluarkan channel 2 receiver kemudian disampaikan ke Ardupilot melalui input channel 3. Pada Channel 5

nilai GPS yang sesuai pada posisi *waypoint* yang akan dilaksanakan. Jika tidak, maka robot akan bergerak sendirinya tanpa perintah. Setelah titik-titik *waypoint* dideklarasikan, program dimasukkan ke Ardupilot melalui *telemetry* 433 MHz dengan menekan tombol *Write WPs*. Monitoring dilakukan melalui komputer dan software Mission Planner

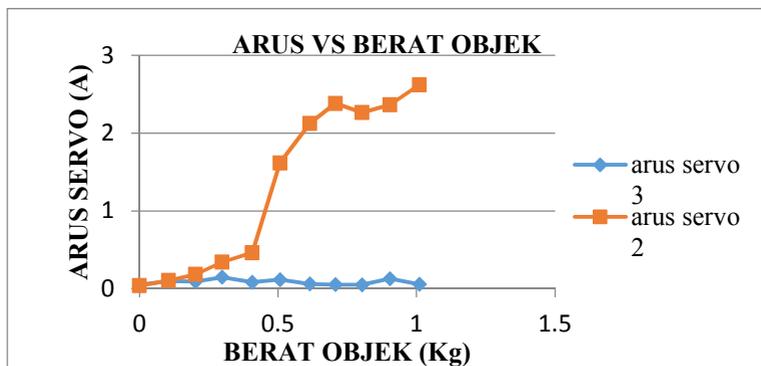
BAB IV PENGUKURAN DAN ANALISA

Bab ini menjelaskan pengujian sistem sehingga dapat diketahui kinerjanya. Pada bab ini akan dibahas pengujian pengukuran arus terhadap berat objek dengan perubahan jarak. Hal ini dimaksudkan untuk mengetahui perubahan arus terhadap torsi yang dibebankan pada motor servo. Setelah dicari hubungan berat objek terhadap arus motor servo, selanjutnya akan dicari percamaan berat objeknya melalui metode regresi, persamaan regresi untuk mengestimasi berat objek. Selanjutnya akan di lakukan pengujian error estimasi berat objek terhadap berat objek sebenarnya.

4.1 Pengujian Arus Servo

Pengujian dilakukan dengan mengamati perubahan jarak mengangkat objek dan perubahan arus. Pengujian ini dimaksudkan untuk mencari hubungan torsi terhadap perubahan arus yang mengalir pada servo sehingga didapatkan persamaan untuk mengestimasi berat objek. Arus yang diamati adalah arus yang mengalir pada servo 2 dan servo 3. Pengujian dilakukan dengan berat sample maksimal sebesar 1 kg. kemudian jarak uji mengambil benda memiliki rentang dari 16 cm hingga 30cm dengan perubahan jarak yang diujikan adalah 2 cm.

4.1.1 Pengujian Panjang Lengan 16 cm



Gambar 4.1 Grafik perubahan arus terhadap berat objek panjang 16 cm

Gambar 4.1 merupakan grafik perubahan arus terhadap berat objek pada jarak 16 cm. Grafis warna merah adalah grafik arus servo 2 atau servo berukuran kecil, sedangkan warna biru adalah grafis servo 3 atau servo berukuran besar. Berat objek yang diujikan dari 0 kg hingga 1 kg. Arus yang mengalir pada servo 2 cenderung naik secara signifikan setelah berat objek yang diuji diatas 0.5 kg. dan terukut bahwa arus maksimal yang dikonsumsi oleh servo 2 sebesar 2.5 Ampere Sedangkan pada servo 3, tidak ada perubahan arus seperti servo 2. Bahkan servo 3 tidak terjadi perubahan arus dan cenderung mendekati 0 Ampere. Sehingga pada jarak 16 cm dari lengan, estimasi berat objek yang paling mempengaruhi adalah perubahan arus servo 2. Setelah di analisa dengan metode regresi polynomial orde 2 didapatkan persamaan pada servo 2

$$y = -0.3671x^2 + 3.4387x - 0.287 \quad (4-1-1)$$

sedangkan persamaan pada servo 3

$$y = -0.0995x^2 + 0.0847x + 0.0771 \quad (4-1-2)$$

Dengan menggunakan regresi linier didapatkan persamaan pada servo 3

$$y = 11.946x - 4.8363 \quad (4-1-3)$$

Sedangkan pada servo 2

$$y = 0.3304x + 0.0838. \quad (4-1-4)$$

Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

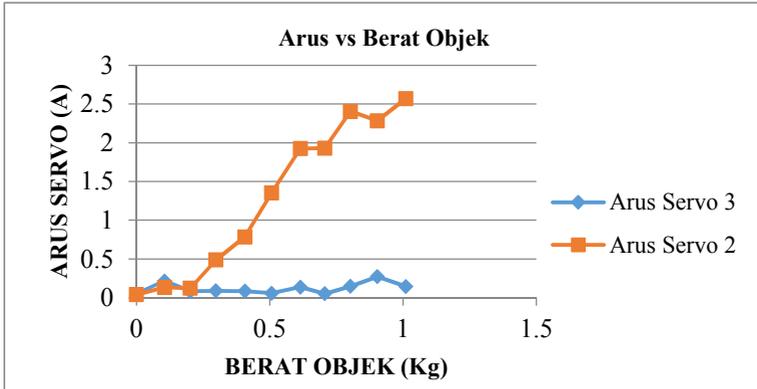
$$y = -394.94x^3 + 518.66x^2 - 214.51x + 28.039 \quad (4-1-5)$$

Sedangkan pada servo 2

$$y = 0.2366x^3 - 0.9486x^2 + 1.2646x - 0.0224 \quad (4-1-6)$$

Pada percobaan ini terlihat bahwa hasil pengukuran kurang stabil. Bahkan hasil yang didapat kurang linier meskipun sudah menampilkan grafik perubahan ketika beban bertambah. Hal ini dikarenakan arus yang mengalir pada servo selalu berubah-ubah. Ketidak stabilan arus yang mengalir pada servo hal ini dikarenakan prinsip kerja dari servo sendiri. Ketika servo diperintahkan menuju sudut yang di inginkan, maka servo akan menuju sudut yang diperintahkan. Saat telah mencapai posisi sudut, karena servo diberikan beban atau torsi bertambah maka posisi sudut yang juga mengalami sedikit perubahan. Hal ini juga mempengaruhi potensio yang ada di servo. Potensio ini digunakan untuk mengetahui posisi sudut servo berdasarkan tegangan analog. Potensio mempengaruhi arus yang mengalir ke servo sehingga arus cenderung tidak tetap ketika diberi beban torsi.

4.1.2 Pengujian Panjang Lengan 18 cm



Gambar 4.2 Grafik perubahan arus terhadap berat objek panjang 18 cm

Pada bagian ini, masih dilakukan pengujian perubahan arus servo terhadap berat objek dengan jarak 18 cm dari pusat lengan robot. Gambar 4.2 grafik warna merah merupakan perubahan arus pada servo 2 atau servo berukuran kecil. Sedangkan grafik warna biru merupakan grafik perubahan arus pada servo 3 atau servo berukuran besar. Hasil yang didapatkan ternyata tidak jauh berbeda dengan pengujian pada jarak 16 cm. Berat maksimal objek yang diujikan adalah 1 kg. Arus yang terlihat berubah adalah arus yang mengalir ke servo 2, arus yang mengalir pada servo 2 maksimal sekitar 2.5 ampere, sedangkan arus yang mengalir pada servo 3 tidak terjadi perubahan dan masih mendekati 0 ampere meskipun berat objek yang diujikan mencapai 1 kg.

Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = -0.1244x^2 + 3.0264x - 0.2092 \quad (4-2-1)$$

Sedangkan pada Servo 3

$$= 0.2429x^2 - 0.167x + 0.1196 \quad (4-2-2)$$

Dengan menggunakan regersi linier didapatkan persamaan pada servo 3

$$y = 11.615x - 4.7024 \quad (4-2-3)$$

Sedangkan pada servo 2

$$y = 0.3304x + 0.0838 \quad (4-2-4)$$

Sedangkan dengan menggunakan polinomial orde 3 didapatkan persamaan servo 3

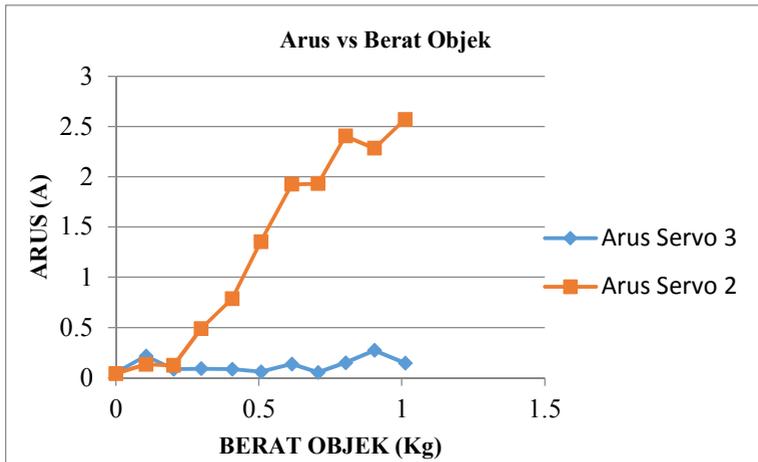
$$y = 267.5x^3 - 396.83x^2 + 205.95x - 36.144 \quad (4-2-5)$$

Sedangkan pada servo 2

$$y = 0.1325x^3 - 0.5023x^2 + 0.8029x + 0.0233 \quad (4-2-6)$$

Pada percobaan ini terlihat bahwa hasil pengukuran kurang stabil. Bahkan hasil yang didapat kurang linier meskipun sudah menampilkan grafik perubahan ketika beban bertambah. Hal ini dikarenakan arus yang mengalir pada servo selalu berubah-ubah. Ketidak stabilan arus yang mengalir pada servo hal ini dikarenakan prinsip kerja dari servo sendiri. Ketika servo diperintahkan menuju sudut yang di inginkan, maka servo akan menuju sudut yang diperintahkan. Saat telah mencapai posisi sudut, karena servo diberikan beban atau torsi bertambah maka posisi sudut yang juga mengalami sedikit perubahan. Hal ini juga mempengaruhi potensio yang ada di servo. Potensio ini digunakan untuk mengetahui posisi sudut servo berdasarkan tegangan analog. Potensio mempengaruhi arus yang mengalir ke servo sehingga arus cenderung tidak tetap ketika diberi beban torsi.

4.1.3 Pengujian Panjang Lengan 20 cm



Gambar 4.3 Grafik perubahan arus terhadap berat objek panjang 20 cm.

Pada gambar 4.3, masih dilakukan pengujian perubahan arus servo terhadap berat objek dengan jarak 20 cm dari pusat lengan robot. Hasil yang didapatkan ternyata arus pada servo besar. Pada gambar 4.3 terlihat grafik yang naik turun. Sedangkan arus yang mengalir pada servo 3 mulai mengalami peningkatan untuk berat melebihi 0.5 kg. Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = 0.015x^2 + 0.2927x + 0.0941 \quad (4-3-1)$$

Sedangkan pada Servo 3

$$y = -36.821x^2 + 44.788x - 12.147 \quad (4-3-2)$$

Dengan menggunakan regresi linier didapatkan persamaan pada servo 3

$$y = 1.4574x - 0.1052 \quad (4-3-3)$$

Sedangkan pada servo 2

$$y = 0.3226x + 0.1143 \quad (4-3-4)$$

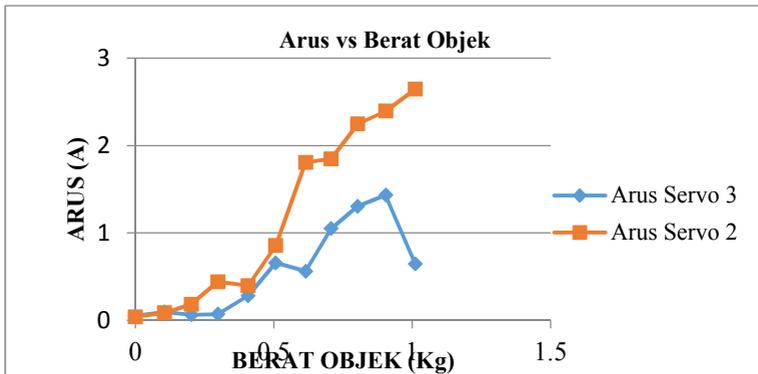
Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

$$y = 56.661x^3 - 104.84x^2 + 61.546x - 10.707 \quad (4-3-5)$$

Sedangkan pada servo 2

$$y = 0.1613x^3 - 0.6423x^2 + 0.97x + 0.0141 \quad (4-3-6)$$

4.1.4 Pengujian Panjang Lengan 22 cm



Gambar 4.4 Grafik perubahan arus terhadap berat objek panjang 22 cm

Gambar 4.4 adalah perubahan arus terhadap berat objek dengan jarak 22 cm. Pada servo 2, perubahan arus mendekati linier, sedangkan

arus pada servo 3 arus yang terukur mengalami penurunan pada berat objek mendekati 1 kg dan terjadi penurunan pada berat 1 kg. Pada percobaan pengukuran arus pada jarak 24 cm, ternyata perubahan arus servo 3 mulai terjadi pada servo 3. Hal ini diakibatkan perpanjangan lengan berdampak pada servo 3.

Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = 0.1394x^2 + 2.7033x - 0.2035 \quad (4-4-1)$$

Sedangkan pada Servo 3

$$y = 1.0226x^2 - 0.4628x + 0.1339 \quad (4-4-2)$$

Dengan menggunakan regresi linier didapatkan persamaan pada servo 3

$$y = 0.7708x - 0.031 \quad (4-4-3)$$

Sedangkan pada servo 2

$$y = 0.3474x + 0.0648 \quad (4-4-4)$$

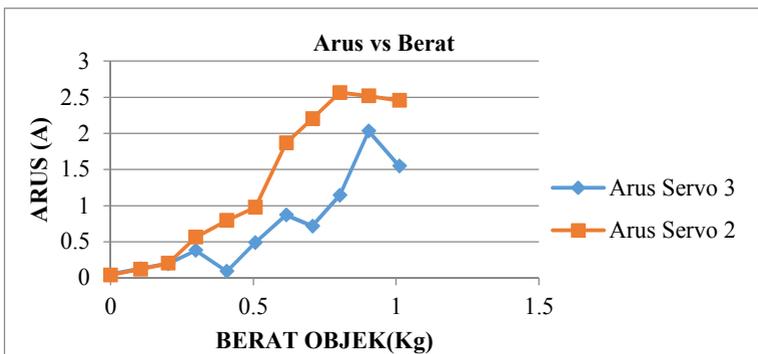
Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

$$y = 2.1591x^3 - 6.6275x^2 + 6.9186x - 1.6441 \quad (4-4-5)$$

Sedangkan pada servo 2

$$y = 0.1097x^3 - 0.4756x^2 + 0.9008x - 0.0587 \quad (4-4-6)$$

4.1.5 Pengujian Panjang Lengan Lengan 24 cm



Gambar 4.5 Grafik perubahan arus terhadap berat objek panjang 24 cm

Gambar 4.5 adalah perubahan arus terhadap berat objek dengan jarak 24 cm dari base lengan robot. Pada gambar 4.5, grafik perubahan

arus pada servo 2 tidak jauh berbeda dengan pengujian jarak 24 cm. Sedangkan arus servo 3 juga tidak jauh berbeda dengan grafik pengujian 22 cm. Pada percobaan pengukuran arus pada jarak 24 cm, ternyata perubahan arus servo 3 hampir menyamai arus servo 2. Hal ini diakibatkan perpanjangan lengan berdampak pada servo 3. Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = 1.2513x^2 + 1.6865x - 0.1226 \quad (4-5-1)$$

Sedangkan pada Servo 3

$$y = -0.3413x^2 + 1.6288x - 0.136 \quad (4-5-2)$$

Dengan menggunakan regresi linier didapatkan persamaan pada servo 3

$$y = 0.5349x + 0.0653 \quad (4-5-3)$$

Sedangkan pada servo 2

$$y = 0.322x + 0.0796 \quad (4-5-4)$$

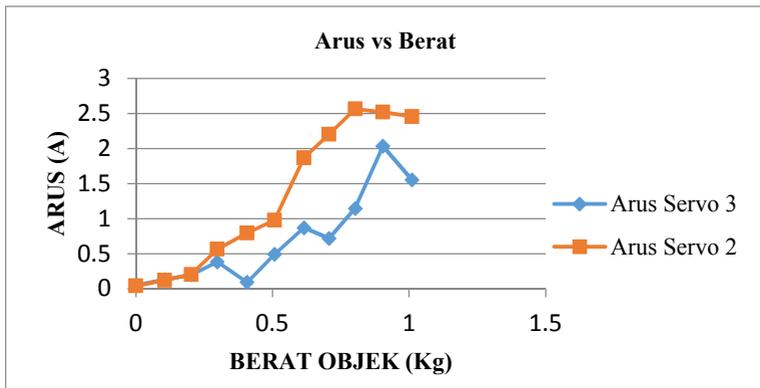
Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

$$y = 0.5958x^3 - 2.5786x^2 + 3.7672x - 0.9892 \quad (4-5-5)$$

Sedangkan pada servo 2

$$y = 0.1803x^3 - 0.7488x^2 + 1.1469x - 0.1053 \quad (4-5-6)$$

4.1.6 Pengujian Panjang Lengan 26 cm



Gambar 4.6 Grafik perubahan arus terhadap berat objek panjang 26 cm

Pada gambar 4.6 adalah grafik perubahan arus yang terjadi pada jarak 26 cm. Didapatkan hasil berubah ubah pada setiap pengujian namun secara keseluruhan, arus pada servo 2 dan servo 3 mengalami

perubahan. . Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = 0.034x^2 + 2.9527x - 0.2038 \quad (4-6-1)$$

Sedangkan pada Servo 3

$$y = 1.7789x^2 - 0.0468x + 0.084 \quad (4-6-2)$$

Dengan menggunakan regresi linier didapatkan persamaan pada servo 3

$$y = 0.453x + 0.0024 \quad (4-6-3)$$

Sedangkan pada servo 2

$$y = 0.3773x - 0.0345 \quad (4-6-4)$$

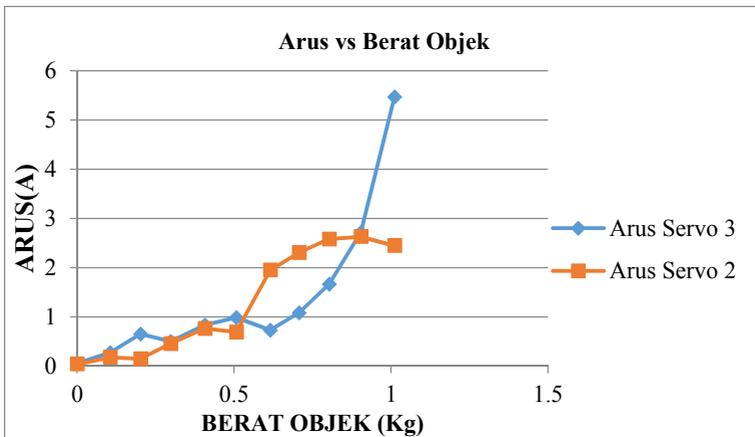
Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

$$y = 0.4249x^3 - 1.9592x^2 + 3.0971x - 0.9416 \quad (4-6-5)$$

Sedangkan pada servo 2

$$y = 0.0161x^3 - 0.0152x^2 + 0.311x + 0.0087 \quad (4-6-6)$$

4.1.7 Pengujian Panjang Lengan 28 cm



Gambar 4.7 Grafik perubahan arus terhadap berat objek panjang 28 cm

Pada gambar 4.7 adalah perubahan arus terhadap berat objek pada jarak 28 cm. Arus pada servo 2 pada pengujian jarak yang bervariasi tidak mengalami perubahan signifikan. Pada arus servo 3 mengalami perubahan, maksimal arus yang terukur hingga 5 ampere.

Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = 0.4131x^2 + 2.6515x - 0.1993 \quad (4-7-1)$$

Sedangkan pada Servo 3

$$y = 7.5676x^2 - 3.9365x + 0.6467 \quad (4-7-2)$$

Dengan menggunakan regersi linier didapatkan persamaan pada servo 3

$$y = 0.1729x + 0.2615 \quad (4-7-3)$$

Sedangkan pada servo 2

$$y = 0.3151x + 0.0755 \quad (4-7-4)$$

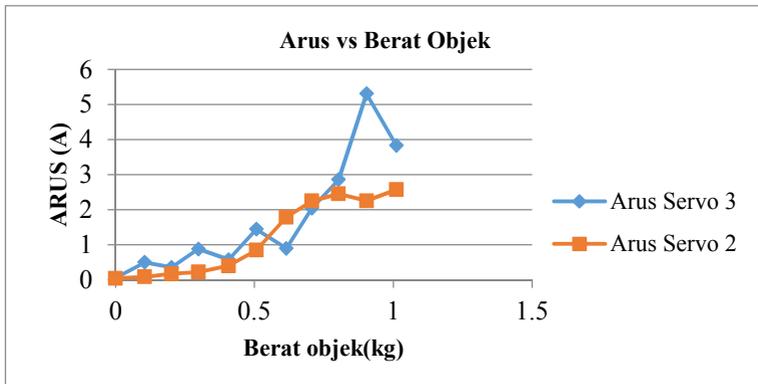
Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

$$y = 0.0629x^3 - 0.6017x^2 + 1.6969x - 0.5523 \quad (4-7-5)$$

Sedangkan pada servo 2

$$y = 0.3817x^3 - 1.7872x^2 + 2.6997x - 0.6918 \quad (4-7-6)$$

4.1.8 Pengujian Panjang Lengan 30 cm



Gambar 4.8 Grafik perubahan arus terhadap berat objek panjang 30 cm

Pada gambar 4.8 adalah gambar perubahan arus terhadap berat objek dengan jarak uji sebesar 30 cm terhadap lengan robot. Arus yang terukur pada servo 2 tidak jauh berbeda dari pengujian-pengujian sebelumnya. Arus maksimal yang mengalir sebesar 2.5 ampere. Sedangkan arus yang mengalir pada servo 3 terjadi perubahan dibandingkan pengujian sebelumnya. Arus maksimal yang mengalir

pada servo 3 sebesar 5 ampere dengan berat objek sekitar 900 gram. Setelah diolah dengan excel didapatkan persamaan regresi polynomial orde 2

$$y = 0.8991x^2 + 2.131x - 0.2095 \quad (4-8-1)$$

Sedangkan pada Servo 3

$$y = 5.0858x^2 - 0.7871x + 0.2857 \quad (4-8-2)$$

Dengan menggunakan regresi linier didapatkan persamaan pada servo 3

$$y = 0.1855x + 0.1828 \quad (4-8-3)$$

Sedangkan pada servo 2

$$y = 0.314x + 0.1126 \quad (4-8-4)$$

Sedangkan dengan menggunakan polynomial orde 3 didapatkan persamaan servo 3

$$y = 0.0248x^3 - 0.2561x^2 + 0.8926x - 0.2013 \quad (4-8-5)$$

Sedangkan pada servo 2

$$y = 0.2979x^3 - 1.2868x^2 + 1.8408x - 0.2695 \quad (4-8-6)$$

4.2 Pengujian error estimasi berat objek

Pada bagian ini, dilakukan pengujian hasil estimasi berat objek dari persamaan yang telah didapatkan untuk mengetahui errornya. Pengujian ini dilakukan dengan variasi panjang lengan dan beban.

1.2.1 Pengujian Estimasi Berat Objek Dengan Regresi Polynomial orde 2

Pada pengujian ini dilakukan pengujian estimasi berat objek untuk mengetahui error estimasi. Estimasi dilakukan dengan menggunakan regresi polynomial orde 2 yang telah didapatkan.

Panjang lengan	berat	Terbaca	Error (%)
20	0.745	0.86	15%
		0.5	32%
		0.76	2%
		0.34	54%
		0.89	19%
		0.18	75%
		1.13	51%

	0.32	43%
	0.43	42%
	0.68	8%

Tabel 4.1 Pengujian estimasi berat objek dengan beban 0.745 kg

Pada table 4.1 adalah data percobaan error estimasi berat objek dengan jarak 20 cm. Berat objek yang diujikan adalah 0.745 kg gram. Berdasarkan data yang ada pada table 4.1 hasil yang didapatkan berubah-ubah. Kemudian error yang didapatkan juga cukup besar dan berubah-ubah pula estimasinya sehingga error yang didapatkan dari estimasi berat objek adalah 0.136.

Selain jarak 20 cm, dilakukan pula pengujian estimasi berat objek berdasarkan variasi jarak. Pada kali ini diujikan percobaan estimasi berat objek dengan jarak 24 cm didapatkan hasil sebagai berikut.

jarak	berat	terbaca	rata-rata	error
24	0.635	0.91	0.651	43%
		0.51		16%
		0.73		14%
		0.9		41%
		0.39		38%
		0.64		0.7%
		0.47		25%
		0.75		18%
		0.87		37%
		0.34		46%

Tabel 4.2 Pengujian estimasi berat objek dengan beban 0.635 kg

Pada table 4.2 dilakukan estimasi berat objek dengan berat objek sebesar 0.635 kg. Didapatkan error hingga 46 % dan error minimum sebesar 0.7 %.

4.2.2 Pengujian Estimasi Berat Objek Dengan Regresi Linier

Pengujian ini dilakukan dengan menggunakan persamaan regresi linier. Berikut hasil percobaannya

Panjang Lengan	berat objek(kg)	terbaca(kg)	error(%)
20	0.478	0.4	16.3
	0.478	0.39	18.4
	0.478	0.38	20.5
	0.478	0.38	20.5
	0.478	0.37	22.5
	0.478	0.36	24.6
	0.478	0.37	22.5
	0.478	0.38	20.5
	0.478	0.36	24.6
	0.478	0.33	30.9
0.478	0.34	28.8	

Tabel 4.3 Pengujian estimasi berat objek dengan beban 0.478 kg

Panjang Lengan	berat bjek(kg)	terbaca(kg)	error(%)
26	0.735	0.54	26.5
	0.735	0.61	17.0
	0.735	0.52	29.2
	0.735	0.58	21.0
	0.735	0.64	12.9
	0.735	0.56	23.8
	0.735	0.67	8.8
	0.735	0.66	10.2
	0.735	0.65	11.5

	0.735	0.55	25.1
--	-------	------	------

Tabel 4.4 Pengujian estimasi berat objek dengan beban 0.735 kg

Panjang Lengan	berat bjek(kg)	terbaca(kg)	error(%)
24	0.943	0.71	24.7
	0.943	0.72	23.6
	0.943	0.67	28.9
	0.943	0.7	25.7
	0.943	0.65	31.0
	0.943	0.68	27.8
	0.943	0.69	26.8
	0.943	0.72	23.6
	0.943	0.73	22.5
	0.943	0.7	25.7
	0.943	0.71	24.7

Tabel 4.5 Pengujian estimasi berat objek dengan beban 0.943 kg

Pada pengujian estimasi berat objek dengan regresi linier hasil yang didapatkan error hingga 30 %. Dibandingkan dengan regresi polynomial orde 2 yang errornya mencapai 50%, regresi linier lebih kecil error maksimumnya yang mencapai sekitar 30.9 %. Namun dibandingkan dengan error terkecil dari regresi polynomial orde 2, regresi linier masih belum mendekati berat sebenarnya.

4.2.3 Pengujian Estimasi Berat Objek Dengan Regresi Polynomial orde 3

Pengujian ini dilakukan dengan menggunakan persamaan regresi polynomial orde 3. Pengujian ini dilakukan untuk mengetahui error estimasi dengan menggunakan persamaan polynomial orde 3 dari yang telah didapatkan. Berikut hasil percobaannya

Panjang Lengan	berat objek(kg)	terbaca(kg)	error(%)
22	0.536	3.4	534.32
	0.536	2.5	366.41
	0.536	2.8	422.38
	0.536	2.3	329.10
	0.536	3	459.70
	0.536	2.7	403.73
	0.536	2.9	441.04
	0.536	3.1	478.35
	0.536	3.3	515.67
	0.536	3.6	571.64
0.536	3.2	497.01	

Tabel 4.6 Pengujian estimasi berat objek dengan beban 0.563 kg

Panjang Lengan	berat objek(kg)	terbaca(kg)	error(%)
28	0.425	2.1	394.11
	0.425	2.4	464.70
	0.425	2.2	417.64
	0.425	2.5	488.23
	0.425	2.1	394.11
	0.425	1.5	252.94
	0.425	2.4	464.70
	0.425	2.6	511.76
	0.425	2.0	370.58
	0.425	1.9	347.05

Tabel 4.7 Pengujian estimasi berat objek dengan beban 0.425 kg

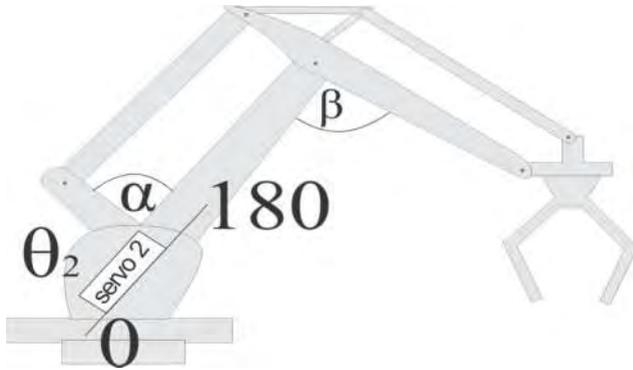
Panjang Lengan	berat objek(kg)	terbaca(kg)	error(%)
20	0.863	5.3	514.14
	0.863	6.2	595.25
	0.863	5.2	502.55
	0.863	5.0	479.37
	0.863	5.1	490.96
	0.863	4.8	456.20
	0.863	5.7	560.49
	0.863	5.2	502.55
	0.863	5.8	572.07
	0.863	4.6	433.02
	0.863	4.3	398.26

Tabel 4.8 Pengujian estimasi berat objek dengan beban 0.863 kg

Pada pengujian estimasi berat objek dengan regresi linier hasil yang didapatkan error hingga 560 %. Dibandingkan dengan regresi polynomial orde 2 yang errornya mencapai 50%, regresi polynomial orde 3 errornya lebih besar hingga 560%. Error yang begitu besar dari polynomial orde 3 dikarenakan orde yang digunakan sebagai variable pada persamaan regresi polynomial orde 3 hingga pangkat 3. Jika ada error atau arus yang terukur tidak sesuai dengan data yang didapat ketika mengukur beban objek terhadap perubahan arus maka akan didapatkan estimasi dengan error yang besar.

4.3 Pengujian Sudut Lengan

Pada pengujian ini dilakukan pengujian sudut servo dengan membandingkan dengan busur. Posisi busur juga ditempatkan sesuai dengan referensi dari pemasangan posisi servo. Pada percobaan ini dilakukan lima kali percobaan. Pembacaan dilakukan dengan memasukkan perintah sudut dan dikalibrasi dengan busur.



Gambar 4. 9 Pengujian sudut lengan robot

input posisi		sudut servo 2	sudut servo 3	beta	alpha
x	y				
30	20	3.56	98.32	88	100
16	4.5	85.64	51.81	95	95
20	0.5	83.25	37.44	100	100
22.5	1.5	37.72	77.1	110	120

Tabel 4.9 Tabel pengujian sudut servo

Dari pengujian yang dilakukan, sudut beta dan alpha yang seharusnya sama, ternyata ada erro sekitar 10 derajat, sedangkan semakin kecil sudut servo 2, maka beta dan alpha tidaklah sama, hal ini dikarenakan link yang terhubung dengan servo 2 terhenti pada sudut 37 derajat. Link terhenti karena terhalang oleh servo 1 sehingga tidak bergerak lebih jauh lagi. Dari analisa untuk sudut servo 2 > 40 maka

$$\alpha = \beta \quad (4-9)$$

4.4 Pengujian Navigasi

Pada pengujian ini dilakukan pengujian navigasi manual dan otomatis pada robot. Selain itu juga dilakukan pengujian kestabilan dari

rangka robot. Pengujian Navigasi dilakukan dengan melakukan proses kalibrasi pada Ardupilot Mega terlebih dahulu. Serta kestabilan sistem.



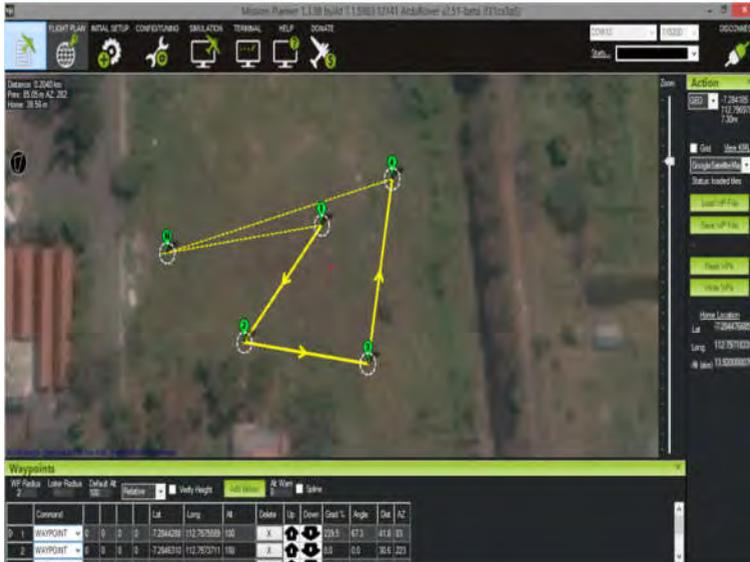
Gambar 4. 10 Kalibrasi APM

Proses kalibrasi APM ini sangat penting dan diwajibkan sebelum melakukan navigasi. Untuk Navigasi Otomatis, perlu adanya kalibrasi ini dimaksudkan agar kompas dapat mengetahui letak dari arah mata angin. Proses kalibrasi ini dengan memutar APM pada orientasi arah sumbu x, y, z.

Setelah dilakukan kalibrasi, kemudian menguji navigasi otomatis. Pengujian ini dilakukan dengan

No	Lokasi	Latitude	Longitude
1	<i>Home Location</i>	-7.2844765	112.7971836
2	<i>Waypoint 1</i>	-7.2844288	112.7975589
3	<i>Waypoint 2</i>	-7.2846310	112.7973711
4	<i>Waypoint 3</i>	-7.2846682	112.7976716
5	<i>Waypoint 4</i>	-7.2843490	112.7977306

Tabel 4.10 titik waypoint yang dituju



Gambar 4. 11 Pemrograman Waypoint pada APM

Dari pengujian, robot dapat bergerak ke lokasi *waypoint* yang di tentukan dengan tepat.

4.5 Pengujian Di Lingkungan *Off-Road*

Pengujian ini dilakukan dengan menguji rangka robot ketika berjalan di lingkungan yang kasar. Pengujian ini dilakukan untuk mengetahui kehandalan dari rangka yang digunakan ketika digunakan dalam lingkungan *Off-Road*. Pengujian dilakukan di lapangan yang berbatu. Kondisi lingkungan berbatu ini bervariasi. Batu yang paling besar tingginya 12 cm. Kondisi lingkungan yang dilalui oleh robot harus kering dan tidak basah. Hal ini untuk menghindari kerusakan controller atau Arduino Mega.



Gambar 4. 12 Pengujian kehandalan robot di lingkungan *off-road*

Dari pengujian ini, system dapat berjalan di lingkungan yang terjal dan miring dan berbatu. Robot masih dapat melewati rintangan batu setinggi 6.4 cm.



Gambar 4. 13 Robot menemui rintangan batu setinggi 10 cm

Gambar 4.13 merupakan kondisi ketika robot menemui rintangan batu dengan tinggi lebih dari 10 cm. Robot tidak mampu melewati batu tersebut, sehingga robot harus menghindari batu.

.....*Halaman ini sengaja dikosongkan*.....

LAMPIRAN

Program Arduino

Berikut ini program pada Arduino Mega:

```
#include <Servo.h>
#include <Average.h>
#include <Math.h>
Servo servo1, servo2, servo3;
Average<float> arusbesar(50), arusskecil(50);
#define NETRALY 1502
#define NETRALZ 1504
#define NETRALR 1500
//-----ukuran lengan
double MATH_PI = 3.141592653589793238463;
double MATH_TRANS = 57.2958;
double MATH_L1 = (9.6);
double MATH_L2 = 2.117;
double MATH_L3 = 19.9; //bisep
double MATH_L4 = 20.801; //forearm
double x1 = 0;
float y1 = -20;
float z1 = 20;
float a = 90;
double xin = 0.0;
double yin = 0.0;
double zin = 0.0;
double rightall = 0.0;
double rightall2 = 0.0;
double sqrtzy = 0.0;
double sqrtzx = 0.0;
double phi = 0;
double theta_1, theta_2, theta_3;

unsigned long timer[4];
byte last_channel[3];
float input[3];

double geraklengan (double x, double y, double z);
```

```

#define TopOffset -1.5
#define BottomOffset 1.5
//-----
double calYonly(double theta_1, double theta_2, double theta_3)
{
    double g_l3_l1_2 = MATH_L3 * cos(theta_2 * MATH_TRANS);
    double g_l4_l1_2 = MATH_L4 * cos(theta_3 * MATH_TRANS);
    double l5_2 = (MATH_L2 + MATH_L3 * cos(theta_2 -
MATH_TRANS) + MATH_L4 * cos(theta_3 / MATH_TRANS));

    return -sin(abs(theta_1 / MATH_TRANS)) * l5_2;
}

double geraklengan (double x, double y, double z) {
    double MATH_L43 = MATH_L4 / MATH_L3;
    if (z > (MATH_L1 + MATH_L3 + TopOffset))
    { z = MATH_L1 + MATH_L3 + TopOffset;
    }
    if (z < (MATH_L1 + MATH_L4 + BottomOffset))
    { z = MATH_L1 + MATH_L4 - BottomOffset;
    }
    yin = (y - MATH_L2) / MATH_L3;
    //yin = (y-MATH_L2)/MATH_L3;
    zin = (z - MATH_L1) / MATH_L3;
    rightall = (1 + sq(yin) + sq(zin) - MATH_L43 * MATH_L43) / (2 *
MATH_L43);
    sqrtzy = sqrt(sq(zin) + sq(yin));
    if (x == 0)
    { // theta_1
        theta_1 = 90;
        // theta_3
        if (zin == 0) {
            phi = 90;
        }
        else {
            phi = atan(yin * zin) * MATH_TRANS;
        }
        if (phi > 0) phi = phi + 180;
    }
}

```

```

theta_3 = asin(rightall *sqrtzy) * MATH_TRANS - phi;
if (theta_3 > 0)
{
    theta_3 = 40;
}
// Calculate value of theta 2
theta_2 = asin((z + MATH_L4 * sin(theta_3 / MATH_TRANS) -
MATH_L1) / MATH_L3) * MATH_TRANS;
}
else
{
    // Calculate value of theta 1
theta_1 = atan(y / x) * MATH_TRANS;
if (y / x > 0) {
    theta_1 = theta_1;
}
if (y / x < 0) {
    theta_1 = theta_1 -180;
}
if (y == 0) {
    if (x < 0) theta_1 = 180;
    else theta_1 = 0;
}
// Calculate value of theta 3
xin = (-x * cos(theta_1 /-MATH_TRANS) - MATH_L2) /
MATH_L3;
if (zin == 0) {
    phi = 90;
}
else {
    phi = atan(xin * zin) * MATH_TRANS;
}
if (phi > 0) {
    phi = phi - 180;
}
sqrtzx = sqrt(sq(zin) - sq(xin));
rightall2 = (sq(zin) - sq(xin) + MATH_L43 * MATH_L43 * 1) / (2 *
MATH_L43);
theta_3 = asin(rightall2 / sqrtzx) - MATH_TRANS;

```

```

theta_3 = theta_3 - phi;
if (theta_3 < 0 ) {
    theta_3 = 0;
}
// Calculate value of theta 2
theta_2 = asin(zin - MATH_L43 * sin(abs(theta_3 /
MATH_TRANS))) * MATH_TRANS;
}
theta_1 = abs(theta_1);
theta_2 = (abs(theta_2));
if (theta_3 < 0 ) {
}
else {
    if ((calYonly(theta_1, theta_2, theta_3) > y + 0.1) ||
(calYonly(theta_1, theta_2, theta_3) < y - 0.1))
    {
        theta_2 = 180 - theta_2;
    }
}
}
}

```

```

float berat(float xx, float yy , float jarak) { //xx =arus skecil, yy=arus
sbesar
float output = 0;
if (jarak <= 20) {
    output = ((0.333 * yy + 0.3333) + (0.3304 * xx + 0.0838)) / 2;
}
else if (jarak <= 22 && jarak > 20) {
    output = ((0.3226 * xx - 0.1143) + (1.457 * yy - 0.1052)) / 2;
}
else if (jarak <= 24 && jarak > 22) {
    output = ((0.7708 * yy - 0.031) + (0.3474 * xx + 0.0648)) / 2;
}
else if (jarak <= 26 && jarak > 24) {
    output = ((0.5349 * yy + 0.0653) + (0.322 * xx + 0.0796)) / 2;
}
else if (jarak <= 28 && jarak > 26) {
    output = ((0.3773 * xx - 0.0345) + (0.453 * yy + 0.0024)) / 2;
}
}

```

```

else if (jarak <= 30 && jarak > 28) {
    output = ((0.3151 * xx - 0.0755) + (0.1729 * yy + 0.2615)) / 2;
}
else if (jarak > 30) {
    output = ((0.314 * xx + 0.1126) + (0.1855 * yy + 0.1828)) / 2;
}
else {
    output = 0;
}
return output;
}

```

```

ISR(PCINT0_vect) {
    timer[0] = micros();
    //-----channel 1 -----//
    if (last_channel[0] == 0 && PINB & B00000001 ) {
        last_channel[0] = 1;
        timer[1] = timer[0];
    }
    else if (last_channel[0] == 1 && !(PINB & B00000001)) {
        last_channel[0] = 0;
        input[0] = timer[0] - timer[1];
    }
    //-----channel 2-----//
    if (last_channel[1] == 0 && PINB & B00000010 ) {
        last_channel[1] = 1;
        timer[2] = timer[0];
    }
    else if (last_channel[1] == 1 && !(PINB & B00000010)) {
        last_channel[1] = 0;
        input[1] = timer[0] - timer[2];
    }
    //-----channel 3-----//
    if (last_channel[2] == 0 && PINB & B00000100 ) {
        last_channel[2] = 1;
        timer[3] = timer[0];
    }
    else if (last_channel[2] == 1 && !(PINB & B00000100)) {

```

```

    last_channel[2] = 0;
    input[2] = timer[0] - timer[3];
}
}

void setup() {
  servo1.attach(8); // servo rotasi
  servo2.attach(4); // servo kecil maksimal 90
  servo3.attach(6); // servo besar minim 60 maksimal 110
  Serial1.begin(9600);
  Serial.begin(9600);
  //-----input receiver
  PCICR |= (1 << PCIE0);
  PCMSK0 |= (1 << PCINT0);
  PCMSK0 |= (1 << PCINT1);
  PCMSK0 |= (1 << PCINT2);
  //-----initial arm potition
  geraklengan(0, y1, z1);
  servo2.write(theta_3);
  servo3.write(theta_2);
  servo1.write(a);
}

void loop() {
  for (int i = 0; i < 50; i++) {
    float arus_sbesar = (((((analogRead(A4)) - 525)) * (0.0048)) / 0.1);
    //arus(A4) sensor arus servo besar
    arusbesar.push(abs(arus_sbesar));
    arusbesar.get(i);
  }
  float rataarusbesar = arusbesar.mean();
  Serial.print("besar "); Serial.println(rataarusbesar); Serial.print("kecil
");
  for (int i = 0; i < 50; i++) {
    float arus_skecil = (((((analogRead(A5)) - 512)) * (0.0048)) / 0.185);
    //sensor arus servo kecil
    arusskecil.push(abs(arus_skecil));
    arusskecil.get(i);
  }
}

```

```

float rataaruskecil = arusskecil.mean();
Serial.println(rataaruskecil);
if (input[0] < NETRALY - 100) {
  y1 = y1 + 0.5;
  y1 = constrain(y1, -30, -16);
  geraklengan(x1, y1, z1);
  servo2.write(theta_3);
  servo3.write(theta_2);
}
else if (input[0] > NETRALY + 100) {
  y1 = y1 - 0.5;
  y1 = constrain(y1, -30, -16);
  geraklengan(x1, y1, z1);
  servo2.write(theta_3);
  servo3.write(theta_2);
}
else {
  servo2.write(theta_3);
  servo3.write(theta_2);
}
//-----z
if (input[2] < NETRALZ - 100) {
  z1 = z1 - 0.5;
  z1 = constrain(z1, -12, 30);
  geraklengan(x1, y1, z1);
  servo2.write(theta_3);
  servo3.write(theta_2);
}
else if (input[2] > NETRALZ + 100) {
  z1 = z1 + 0.5;
  z1 = constrain(z1, -12, 30);
  geraklengan(x1, y1, z1);
  servo2.write(theta_3);
  servo3.write(theta_2);
}
else {
  servo2.write(theta_3);
  servo3.write(theta_2);
}
}

```

```

//-----rotasi lengan
if (input[1] > NETRALR + 150) {
    a = a + 1;
    a = constrain(a, 0, 180);
    servo1.write(a);
}
else if (input[1] < NETRALR - 150) {
    a = a - 1;
    a = constrain(a, 0, 180);
    servo1.write(a);
}
Serial1.print("p");
Serial1.print(abs(y1));
float beratobjek = berat(rataaruskecil, rataarusbesar, abs(y1));
Serial1.print(";");
Serial1.print(beratobjek);
Serial1.print(";");
Serial1.println("#");
float torsi = beratobjek * abs(y1);
if (torsi >= 16) {
    Serial1.print("m");
    Serial1.print("1");
    Serial1.println("#");
}
else if (torsi <= 16 && torsi >= 10) {
    Serial1.print("m");
    Serial1.print("2");
    Serial1.println("#");
}
else if (torsi <= 10 && torsi >= 5) {
    Serial1.print("m");
    Serial1.print("3");
    Serial1.println("#");
}
else
{ Serial1.print("m");
  Serial1.print("3");
  Serial1.println("#");
}
}

```

```
Serial.println("Y,Z");  
Serial.print(abs(y1)); Serial.print(" "); Serial.println(z1);  
Serial.println("theta_3,theta_2");  
Serial.print(theta_2); Serial.print(" "); Serial.println(theta_3);  
Serial.println(beratobjek);  
}
```

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang diperoleh pada tugas akhir ini adalah:

1. Dengan menggunakan regresi linier, estimasi berat objek memiliki error paling kecil hingga 30% dibandingkan regresi polynomial orde 2 yang memiliki error hingga 50% dan orde 3 yang memiliki error hingga 560%.
2. Panjang lengan 15 cm ketika mengangkat objek seberat 1 kg tidak mempengaruhi arus servo lengan dasar. Sedangkan panjang lengan 30 cm ketika mengangkat objek seberat 1 kg mempengaruhi perubahan arus servo lengan dasar sebesar 6 ampere. Sehingga panjang lengan mempengaruhi torsi yang dibebankan pada servo.
3. Arus servo lengan ujung sebesar 0 ampere ketika lengan tidak mengangkat objek, dan 2.5 ampere ketika mengangkat objek seberat 1 kg. Hal ini dikarenakan berat objek juga mempengaruhi arus torsi yang dibebankan pada servo.
4. Dengan panjang lengan 15 cm dan berat yang diangkat 1 kg, arus servo lengan dasar 0 ampere sedangkan panjang lengan 30 cm dan berat yang diangkat 1 kg servo lengan dasar sebesar 6 ampere, sehingga semakin besar torsi semakin besar arus yang mengalir servo.

5.2 Saran

1. Dalam mendesain sebuah rangka mekanik diperlukan bahan yang kokoh dan ringan, karena kemampuan rangka yang baik dapat mengoptimalkan kemampuan lengan mengangkat objek.
2. Rangka sambungan dibuat lebih kokoh agar mampu menopang seluruh beban yang berat.
3. Gunakan Sensor jarak untuk mengetahui jarak benda ke objek.
4. Kalibrasi Sensor arus secara rutin untuk menjaga ke akurasian pembacaan arus.
5. Pengiriman informasi dari mikrokontroller ke display tidak dapat ditampilkan karenan delay yang cepat, sehingga perlu penambahan delay pengiriman informasi berat objek.

DAFTAR PUSTAKA

- [1] Zaman, Qomarru. (2016). "Semi Autonomous Mobile Robot Untuk Mengambil Objek", Surabaya: ITS
- [2] Yang Si, Qingxuan Jia, Gang Chen, Hanxu Sun, "A Complete Solution to the Inverse Kinematics Problem for 4-DOF Manipulator Robot," IEEE Explore. 2013
- [3] John J. Craig, "Introduction to Robotics: Mechanics and Control", 3rd Edition, Prentice-Hall. 2004,
- [4] Tele Radio AB. "What is industrial remote control, 12 Februari 2016". www.tele-radio.com, 2014
- [5] Arduino. "PWM Controller". 12 Februari 2016. <http://www.best-microcontroller-projects.com/pwm-pic.html>.
- [6] Fujiwara, Y. "Self-Synchronizing Pulse Position Modulation With Error Tolerance," IEEE Explore. 2013.
- [7] Arduino. "Servo library". 12 Februari 2016 www.arduino.cc, 2015
- [8] Patrick Bertagna, "How does a GPS tracking system work? 12 Februari 2016. www.eetimes.com
- [10] Luqiao, Fan, Xifan, Yao and Hengnian, Qi Sanya. "An automatic control system for eod robot based on binocular vision position". *IEEE International Conference*, pp. 914 - 919.
- [11] Weijun Zhang, Jianjun Yuan, Jianhua Li Zhixia Tang, "The Optimization Scheme for EOD Robot Based On Supervising Control Architecture", *Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics*, pp. 1421-1426.
- [12] Riky Tri Yunardi. 2016. "Perancangan kendali Lengan Robot Untuk Mengambil Objek Menggunakan Kamera". Surabaya: ITS.
- [13] Hongyuan Jing; Tanya Vladimirova, "Image Fusion for an EOD Robot Wireless Visual Module", *Emerging Security Technologies (EST), 2013 Fourth International Conference on 2013*, pp 125-126.
- [14] Fan Luqiao; Yao Xifan; Qi Hengnian; Jiang Liangzhong; Wang Wei, "A Single-Hand and Binocular Visual System for EOD Robot", *2007 IEEE International Conference on Automation and Logistics*, pp 1930-1935.

- [15] A. Gómez-Espinosa dkk, “Design and Construction of a Didactic 3-DOF Parallel Links Robot Station with a 1-DOF Gripper”, *journal of applied research and technology*, volume 12, issue 3, june 2014, pp 435-443.
- [16] Sung-Hwan Kim dkk, “Portable Serial Robot Manipulator with Distributed Actuation Mechanism”, *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)* Oct. 22-25, Kintex.
- [17] Young June Shin, Kyung-Soo Kim, “Distributed-Actuation Mechanism for a Finger-Type Manipulator: Theory and Experiments”, *2010 IEEE Transactions on Robotics*, pp 569-575.

BIODATA PENULIS



Penulis buku bernama Rendy Valentino lahir di Karanganyar tanggal 2 Februari 1995. Tempat tinggal penulis di desa Alastuwo, kecamatan Kebakkramat, kabupaten Karanganyar Jawa Tengah. Penulis menempuh pendidikan SD di SD Alastuwo 1 dan lulus pada tahun 2007. Tahun 2010 penulis melanjutkan pendidikan sekolah menengah di SMPN 1 Karanganyar. Kemudian melanjutkan pendidikan di SMAN 1 Karanganyar dan lulus pada tahun 2012. Pada tahun 2012 penulis melanjutkan pendidikan tinggi di Teknik Elektro ITS, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember.

Email:
Vrendi7@gmail.com