

BAB II

LANDASAN TEORI

2.1 Tinjauan Penelitian Terdahulu

Studi literatur yang penulis lakukan yaitu mempelajari beberapa penelitian yang ada hubungannya dengan skripsi ini sebagai rujukan dan perbandingan pada metode yang digunakan serta hasil yang didapatkan pada penelitian ini.

Andi Chairunnas (2017) mengenai penerapan algoritma *tripod gait* pada hexapod menggunakan arduino mega128. Pada penelitian ini membahas bagaimana membangun sistem kendali pada robot *hexapod* dengan menerapkan pola langkah *tripod gait* pada robot *hexapod* sehingga akurasi pergerakan yang diterapkan pada robot *hexapod* akan menghasilkan pola gerakan yang maksimal

Wahyudi dkk (2017) mengenai Implementasi Sistem Kendali PD (*Proportional Derivative*) Pada Navigasi *Wall Follower* Robot Berkaki Enam (*Hexapod*). Pada penelitian ini menggunakan metode *wall follower* akan diimplementasikan menggunakan sistem kendali PD. Sistem kendali ini terdiri dari kontroler *proporsional* dan *derivatif*.

Nurul dkk (2021) mengenai sistem navigasi robot berkaki menggunakan sensor lidar dengan metode PID. Pada penelitian ini menggunakan sistem kontrol PID karena efisien untuk meluruskan pergerakan robot saat menyusuri dinding area perlombaan.

Dari penelitian sebelumnya terdapat masih banyak kekurangan terutama pada sistem kendali robot dalam menelusuri dinding dan ruangan sehingga timbul

keinginan untuk merancang sebuah robot berkaki enam (*hexapod*) dengan kemampuan *wall following* berbasis kendali PID dalam pola langkah *tripod gait* dan *wave gait*, sebuah sistem kendali yang diharapkan mampu memberikan kestabilan pada pergerakan robot sehingga robot mampu bernavigasi dengan lebih baik dan berjalan mengikuti dinding pada arena KRSRI.

Tabel 2.1 Penelitian Terdahulu

No	Penelitian	Hasil Review	Sistem Kontrol	Navigasi	Pola Langkah	
					<i>Tripod</i>	<i>Wave</i>
1	(Andi Chairunnas, 2017)	Menerapkan pola langkah <i>tripod gait</i> pada robot <i>hexapod</i> sehingga akurasi pergerakan yang diterapkan akan menghasilkan pola gerakan yang maksimal	X	X	√	X
2	(Wahyudi, 2017)	Menggunakan metode <i>wall follower</i> akan diimplementasikan menggunakan sistem kontrol PD.	PD	<i>Wall Following</i>	X	X
3	(Tumbel, Poekoel and Kambey, 2018)	Menerapkan navigasi <i>Wall Following</i> untuk memadamkan api	X	<i>Wall Following</i>	X	X

Tabel 2.1 (Lanjutan)

No	Penelitian	Hasil Review	Sistem Kontrol	Navigasi	Pola Langkah	
					<i>Tripod</i>	<i>Wave</i>
4	(Dzulfiqar and Widodo, 2019)	Menerapkan sistem kontrol PID untuk pelacakan titik api	PID	X	X	X
5	(Nurul, Winarno, 2021)	Menggunakan sistem kontrol PID	PID	X	X	X
6	Penelitian yang diusulkan	Menerapkan sistem kontrol PID <i>wall following</i> menggunakan <i>tripod gait</i> dan <i>wave gait</i>	PID	<i>Wall Following</i>	√	√

2.2 Pengertian Robot

Robot adalah sistem yang menggantikan pekerjaan manusia dan mempunyai kemampuan fungsional manusia. Definisi robot menurut Institut Robot Amerika (1979). Robot merupakan manipulator yang dapat di program ulang.

2.3 Pengertian Robot *Hexapod*

Robot hexapod adalah jenis robot yang memiliki enam kaki, mirip dengan serangga atau hewan lain yang memiliki enam kaki. Kata "hexapod" berasal dari bahasa Yunani, di mana "hexa" berarti enam, dan "pod" berarti kaki. Robot ini dirancang dengan enam kaki untuk memberikan stabilitas, kemampuan bergerak di berbagai jenis medan, dan kemampuan navigasi yang lebih baik daripada robot dengan jumlah kaki yang lebih sedikit.



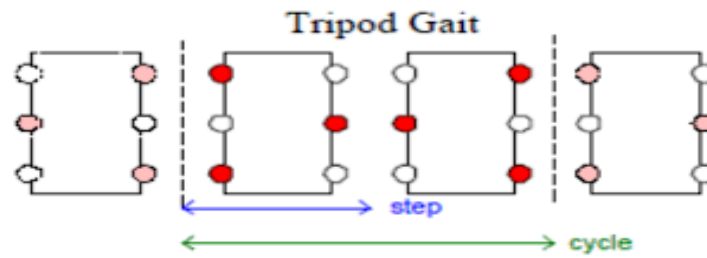
Gambar 2.1 Robot *Hexapod*

2.4 *Gait*

Gait pada robot *hexapod* adalah pola gerakan atau langkah yang digunakan oleh robot untuk berjalan atau bergerak. Ini mencakup urutan gerakan kaki dan keseimbangan tubuh yang digunakan oleh robot untuk berpindah dari satu titik ke titik lainnya. *Gait* pada robot *hexapod* sangat penting karena memengaruhi kinerja dan kemampuan robot untuk bergerak dengan stabil, efisien, dan sesuai dengan tujuan yang diinginkan.

2.5 *Tripod Gait*

Pola *gait tripod* ini adalah salah satu dari beberapa pola pergerakan yang digunakan oleh robot *hexapod* untuk mencapai stabilitas dan keseimbangan saat bergerak. Pada saat tertentu, tiga kaki yang bersentuhan dengan permukaan memberikan dukungan dan daya dorong, sementara tiga kaki yang lain diangkat dan bersiap untuk melangkah ke posisi berikutnya

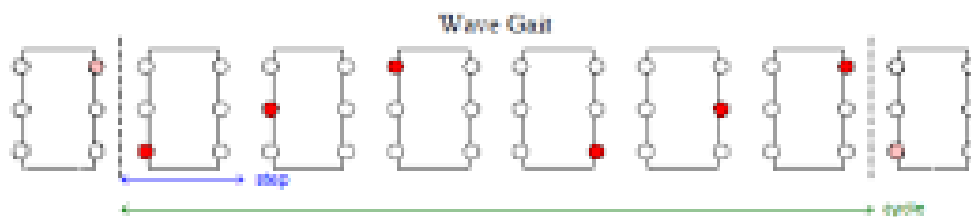


Gambar 2.2 Pola Tripod Gait

Gerakan *tripod* di atas memperlihatkan 3 kaki yang diangkat dan melangkah ke depan sedangkan kaki lainnya tetap di labirin. Gerakan ini terbukti paling cepat, karena mampu mengangkat 3 kaki sekaligus dalam satu fase gerakan.

2.6 Wave Gait

Wave gait pada robot hexapod adalah salah satu pola pergerakan kaki yang digunakan oleh robot hexapod dengan enam kaki untuk berjalan atau bergerak. Dalam pola gait wave, gerakan kaki dimulai dari satu ujung tubuh robot dan bergerak ke sisi lain secara berurutan, mirip dengan efek gelombang yang berjalan melintasi permukaan air.

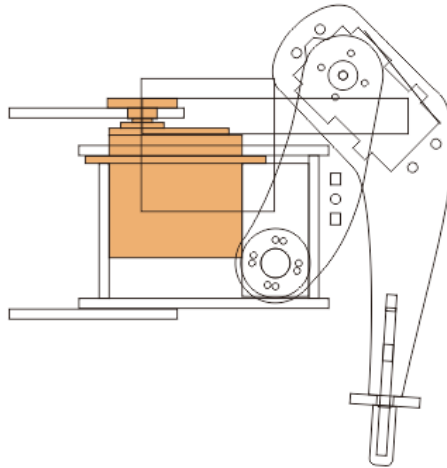


Gambar 2.3 Pola Wave Gait

Pergerakan gait yang terlihat pada gambar di atas menunjukkan bahwa kaki-kaki yang berwarna merah bergerak maju sementara kaki-kaki yang lainnya tetap bersentuhan dengan permukaan tanah. Gerakan kaki ini dilakukan secara

simultan dan mengharuskan kaki-kaki yang tidak bergerak untuk menjaga stabilitas posisi robot.

2.7 Gait Robot *Hexapod*



Gambar 2.4 Desain Struktur Kaki

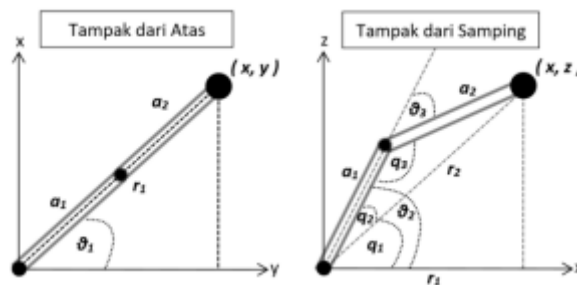
Komponen-komponen ini sebagai bagian dari struktur dan sistem pergerakan robot. Berikut adalah penjelasan umum tentang masing-masing komponen:

Coxa adalah bagian pangkal kaki robot hexapod yang menghubungkan kaki ke tubuh robot. Biasanya, *coxa* adalah sambungan rotasi pertama yang memungkinkan kaki untuk bergerak dalam berbagai arah dan sudut terhadap badan robot. Pada banyak rancangan hexapod, *coxa* sering berputar di sekitar sumbu vertikal, memungkinkan gerakan horizontal.

Femur adalah bagian tengah dari kaki dan merupakan sambungan rotasi kedua pada robot *hexapod*. *Femur* menghubungkan *coxa* dengan *tibia*. Gerakan rotasi pada sambungan femur memungkinkan kaki untuk naik dan turun.

Tibia adalah bagian ujung kaki yang terhubung dengan *femur*. Sambungan rotasi pada *tibia* memungkinkan gerakan kaki ke depan dan ke belakang. Pada beberapa desain hexapod, *tibia* juga dapat memiliki sambungan rotasi tambahan yang memungkinkan gerakan lateral.

Ketiga komponen ini bekerja bersama-sama untuk menghasilkan gerakan yang kompleks dan koordinatif pada robot *hexapod*. Dengan mengontrol sudut dan urutan gerakan pada setiap sambungan, robot *hexapod* dapat berjalan, berlari, atau melakukan gerakan lainnya dengan keseimbangan yang baik.



Gambar 2.5 Simbol Kaki

$$\vartheta_1 = \tan^{-1} \left(\frac{x}{y} \right) \times 57.29$$

$$r_1 = \sqrt{x^2 + y^2}$$

$$r_2 = \sqrt{r_1^2 + z^2}$$

$$\beta = \cos^{-1} \left(\frac{r_1}{r_2} \right)$$

$$q_1 = \cos^{-1} \frac{a_1^2 + r_2^2 - a_2^2}{2 \times a_1 \times r_2}$$

$$\vartheta_2 = -(q_1 + \beta) \times 57.29$$

$$q_2 = \cos^{-1} \frac{a_1^2 + a_2^2 - r_2^2}{2 \times a_1 \times r_2}$$

$$\vartheta_3 = 180 - (q_2 \times 57.29)$$

Keterangan: a_1 dan a_2 adalah konstan yang mewakili panjang femur dan tibia. Dalam persamaan ini, theta ke- n (θ_n) dihitung sebagai sudut untuk servo ke- n . Masukan yang digunakan dalam persamaan ini adalah posisi end efektor, yang dinyatakan dalam bentuk koordinat pada sumbu x , y , dan z .

2.8 Algoritma *Wall Following*

Algoritma dapat didefinisikan sebagai suatu rangkaian logika, metode, dan langkah-langkah sistematis yang digunakan untuk memecahkan masalah tertentu. Algoritma juga dapat dianggap sebagai urutan langkah-langkah yang tersusun secara logis dan terstruktur. Salah satu metode navigasi yang umumnya digunakan untuk menjelajahi struktur dinding disebut "*wall following*." Metode ini umumnya digunakan oleh robot yang memiliki kemampuan untuk mengikuti dinding atau labirin guna menyelesaikan tugas-tugas tertentu. Secara dasar, tujuan algoritma ini adalah menjaga jarak robot terhadap dinding pada tingkat yang diinginkan sambil terus bergerak maju.

Cara kerja algoritma *wall following* adalah dengan menjaga jarak antara robot dan dinding agar tetap konstan. Apabila terjadi perubahan dalam jarak ini, robot akan menggerakkan dirinya sendiri untuk menyesuaikan jarak tersebut. Jika jarak antara robot dan dinding terlalu besar, robot akan mendekat, sedangkan jika terlalu kecil atau terlalu dekat dengan dinding, robot akan menjauh. Proses ini akan terus diulang secara berulang-ulang selama robot bergerak mengikuti dinding.

2.9 Sistem Kendali PID

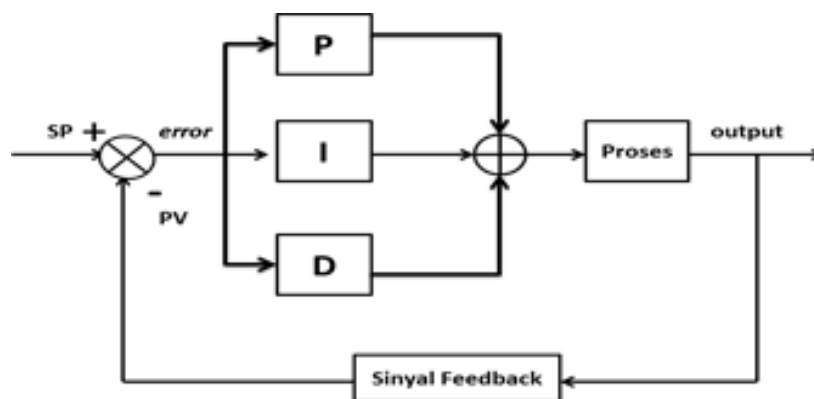
Dalam sistem kontrol, terdapat berbagai tindakan pengendalian yang dikenal, seperti kontrol proporsional, integral, dan derivatif. Setiap tindakan pengendalian ini memiliki keunggulannya sendiri. Misalnya, kontrol *proporsional* memiliki kelebihan dalam mempercepat waktu naik, kontrol *integral* berguna untuk mengurangi kesalahan, dan kontrol *derivatif* dapat membantu mengurangi kesalahan atau meredam osilasi berlebih. Oleh karena itu, untuk mencapai hasil keluaran yang cepat dan akurat, kita dapat menggabungkan ketiga tindakan pengendalian ini menjadi tindakan pengendalian PID.

Parameter pengendali *Proporsional Integral Derivatif* (PID) selalu bergantung pada karakteristik sistem yang diatur. Dalam hal ini, apapun kompleksitas plant tersebut, perilaku plant harus dipahami terlebih dahulu sebelum mencari parameter PID yang sesuai.

2.10 Tuning kendali PID Pada Robot *Wall Following*

Penyetelan kontrol PID pada robot pengikut dinding (*wall follower*) bertujuan untuk menentukan parameter aksi kontrol Proporsional, Integral, dan Derivatif. Proses ini dapat dilakukan melalui metode uji coba dan perbaikan. Kelebihan dari metode ini adalah tidak memerlukan identifikasi plant, pembuatan model matematis plant, atau penentuan parameter plant melalui pendekatan grafis atau analitis. Pendekatan ini cukup dengan memberikan konstanta PID pada rumus PID secara berulang hingga mencapai hasil yang diinginkan, dengan mempertimbangkan karakteristik kontrol PID yang sesuai.

Penggunaan kendali PID mengimplikasikan pengolahan sinyal kesalahan atau error. Nilai error ini diolah dengan rumus PID untuk menghasilkan sinyal kendali atau sinyal kontrol yang akan diteruskan ke aktuator. Berikut diagram blok umpan balik loop tertutup dalam perancangan kendali PID pada robot wall follower seperti di bawah ini.



Gambar 2.6 Diagram PID

Sumber : (Pradana, 2012)

Dari diagram blok yang tertera di atas, dapat dijelaskan sebagai berikut:

1. *SP*, yang bisa diartikan secara sederhana sebagai nilai referensi atau nilai yang diinginkan.
2. *PV*, yang merujuk pada nilai saat ini, yakni nilai yang diukur oleh sensor pada saat yang bersamaan atau sinyal umpan balik dari sensor.
3. *Error*, yaitu perbedaan atau selisih antara nilai yang diukur oleh sensor (*PV*) dan nilai referensi (*SP*), yang juga dapat disebut sebagai deviasi atau ketidaksesuaian antara variabel yang terukur dan nilai yang diinginkan.

2.11 Konstanta PID

Tuning konstanta PID melibatkan proses eksperimen berdasarkan metode *trial and error* untuk mencapai nilai *proporsional*, *derivatif*, dan *integral* yang sesuai dalam rumus PID, sehingga sistem dapat berfungsi dengan stabil. Cara untuk menyetel PID pada robot *wall follower* dapat dijelaskan sebagai berikut:

1. Pertama adalah menggunakan kontrol *proporsional* dengan mengabaikan nilai *integral* dan *derivatif*. *Integral* dan *derivatif* diberi nilai nol.
2. Selanjutnya, kita meningkatkan konstanta *proporsional* secara bertahap hingga mencapai keadaan yang stabil, meskipun robot masih mengalami osilasi.
3. Untuk mengatasi osilasi tersebut, kita menambahkan konstanta *derivatif* dengan membagi dua nilai *proporsional*. Kita mengamati bagaimana sistem robot merespons dan mencapai stabilitas yang lebih baik.
4. Ketika sistem robot telah mencapai stabilitas, kita dapat mempertimbangkan untuk menambahkan kontrol *integral* sebagai pilihan. Namun, kita harus berhati-hati dalam menentukan nilai *integral*, karena nilai yang salah dapat membuat sistem robot tidak stabil.
5. Nilai *set point* kecepatan dan batas atas/bawah memberikan panduan tentang kecepatan yang diinginkan oleh robot.
6. Selain itu, nilai waktu *sampling* (waktu pencuplikan) juga memengaruhi perhitungan PID, terutama saat menggunakan kontrol *integral* dan *derivatif*.

Tuning PID melalui metode *trial and error* adalah cara yang umum digunakan untuk mendapatkan pengaturan yang optimal untuk robot *wall follower*.

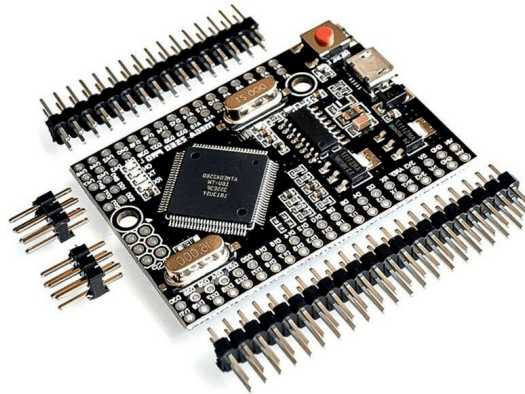
2.12 Invers Kinematik

Definisi invers kinematika yang dijelaskan oleh Buss (2009) dalam "*Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse, and Damped Least Squares Method*" adalah untuk mengatur pergerakan pada sebuah struktur multibody yang kaku, biasanya menggunakan teknik *invers* kinematika (IK). Dalam konteks IK, kita mengasumsikan adanya titik spesifik yang disebut sebagai "*end-effector*" yang berada pada salah satu bagian dari struktur tersebut dan ditugaskan untuk mencapai suatu "posisi target". Untuk memecahkan masalah IK ini, perlu mencari pengaturan untuk sendi-sendi (*joint*) sehingga konfigurasi keseluruhan dari *multibody* dapat mengatur posisi *end-effector* sesuai dengan target yang diinginkan. *invers* kinematika mengambil pendekatan sebaliknya, yaitu mencari parameter-parameter tersebut agar dapat mencapai konfigurasi yang diinginkan.

Dengan demikian, dapat disimpulkan bahwa *invers* kinematika digunakan untuk mengendalikan gerakan robot sehingga posisi *end-effector* sesuai dengan target yang diinginkan. Ini adalah pendekatan yang sangat berguna terutama dalam kasus robot *hexapod* yang memiliki banyak bagian tubuh yang perlu dikontrol secara presisi.

2.13 Arduino Mega 2560

Arduino adalah platform pengembangan perangkat keras (*hardware*) *open-source* yang dirancang untuk memudahkan pembuatan dan pengembangan berbagai jenis proyek elektronika. Platform ini mencakup mikrokontroler, lingkungan pengembangan perangkat lunak (*software*),



Gambar 2.7 Arduino Mega 2560

Arduino Mega 2560 adalah salah satu jenis papan pengembangan mikrokontroler yang dikembangkan oleh Arduino. Papan ini merupakan versi yang lebih canggih dan memiliki lebih banyak sumber daya dibandingkan dengan model Arduino yang lebih sederhana seperti Arduino Uno. Arduino Mega 2560 memiliki sejumlah fitur dan spesifikasi yang membuatnya cocok untuk proyek-proyek yang memerlukan lebih banyak pin input/output (I/O), kapasitas memori yang lebih besar, dan lebih banyak sumber daya komputasi.

2.14 Motor Servo

Motor servo adalah motor yang mampu bekerja dua arah yaitu searah jarum jam (*clockwise*) dan berlawanan jarum jam (*counterclockwise*) dimana arah dan sudut pergerakan rotornya dapat dikendalikan hanya dengan memberikan pengaturan *duty cycle* sinyal PWM (*Pulse Width Modulation*) pada bagian pin kontrolnya.

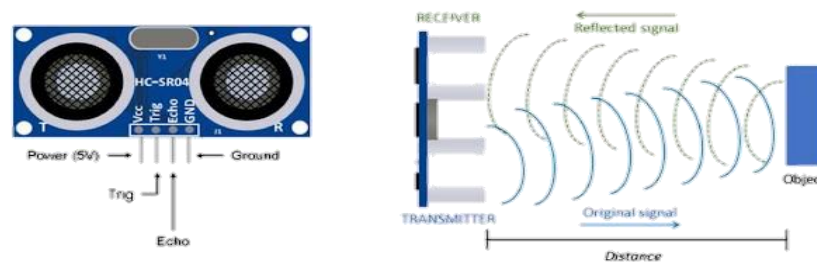


Gambar 2.8 Servo MG995

Motor servo biasanya terbagi menjadi dua jenis, yaitu servo *continuous* dan servo standar. Servo *continuous* memiliki kemampuan untuk berputar hingga 360 derajat penuh, sedangkan servo standar hanya memiliki rentang putaran sebesar 180 derajat. Dalam menghadapi perubahan yang cepat, seperti dalam hal perubahan posisi, kecepatan, dan percepatan, motor servo harus memiliki kinerja yang sangat baik. Kinerja rotasi motor servo tidak ditentukan oleh seberapa tinggi tegangan suplai yang diberikan ke motor, melainkan oleh besarnya sinyal input (PWM) yang diterimanya.

2.15 Sensor Ultrasonik HC-SR04

Sensor ultrasonik HC-SR04 adalah perangkat elektronik yang digunakan untuk mengukur jarak antara sensor dan objek dengan menggunakan gelombang ultrasonik. Sensor ini sering digunakan di berbagai aplikasi, termasuk robotika.



Gambar 2.9 Sensor Ultrasonik HC-SR04

HC-SR04 terdiri dari dua komponen inti, yaitu pemancar ultrasonik (*ultrasonic transmitter*) dan penerima ultrasonik (*ultrasonic receiver*). Peran utama dari pemancar ultrasonik adalah untuk menghasilkan gelombang ultrasonik dengan frekuensi 40 KHz, sementara penerima ultrasonik bertugas untuk menangkap pantulan gelombang ultrasonik yang memantul dari suatu objek. Dengan mengukur waktu yang diperlukan untuk gelombang ultrasonik melakukan perjalanan dari pemancar ke objek dan kembali ke penerima, sensor dapat menghitung jarak antara sensor dan objek yang berada di depannya.

2.16 Step Up dan Step Down

Step up adalah proses meningkatkan tegangan listrik dan mengaturnya sesuai dengan kebutuhan perangkat elektronik. Transformator dengan jumlah lilitan sekunder yang lebih banyak daripada lilitan primer digunakan untuk melakukan peningkatan tegangan. Transformator ini digunakan dalam pembangkit listrik untuk mengubah tegangan yang dihasilkan oleh generator menjadi tegangan yang lebih tinggi, yang kemudian digunakan untuk transmisi listrik jarak jauh



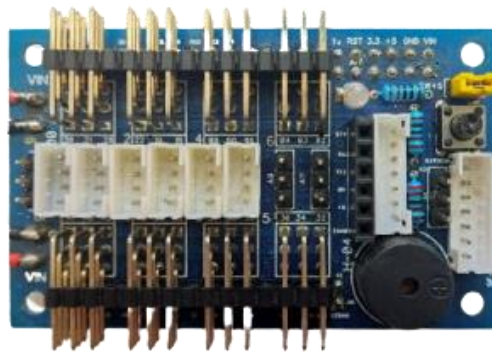
Gambar 2.10 Step Up dan Step Down

Step down adalah suatu jenis transformator yang mengurangi tegangan keluaran. Transformator step-down memiliki jumlah lilitan pada sekunder yang

lebih sedikit dibandingkan dengan jumlah lilitan pada primer, sehingga berperan sebagai pengurang tegangan. Jenis transformator ini mudah ditemukan di berbagai tempat, terutama dalam adaptor AC-DC. Tujuan dari transformator step-down adalah mengubah tegangan tinggi dengan arus rendah menjadi tegangan rendah dengan arus tinggi. Peran utama dari transformator step-down adalah mengurangi tegangan listrik dan menyesuaikannya kebutuhan perangkat elektronik.

2.17 Papan Arduino Mega Shield

Arduino Mega adalah salah satu varian dari papan mikrokontroler Arduino yang memiliki lebih banyak pin input/output dan sumber daya dari papan Arduino standar. Arduino Mega Shield adalah tambahan periferil atau ekstensi yang dapat dipasang di atas papan Arduino Mega untuk memperluas fungsionalitasnya.



Gambar 2.11 Arduino Mega Shield

2.18 Battery

Baterai merupakan perangkat yang mengandung sel-sel listrik yang mampu menyimpan energi dan mengubahnya menjadi tenaga. Baterai menghasilkan listrik melalui proses kimia yang dapat dibalikkan. Baterai, atau disebut juga akumulator, merupakan suatu jenis sel listrik di mana terjadi proses elektrokimia yang dapat diubah-ubah.



Gambar 2.12 Battery LIPO

Baterai jenis ini sering menjadi pilihan utama dalam bidang robotika karena memiliki berbagai keunggulan, seperti bobotnya yang ringan, harganya yang terjangkau, kapasitas penyimpanan energi yang cukup besar, dan hampir tidak mengandung bahan berbahaya. Meskipun begitu, baterai ini memiliki risiko meledak, terutama jika terpapar api. Oleh karena itu, sangat penting untuk tidak membuang atau membakar baterai Li-Ion yang rusak, melainkan mengambil tindakan yang tepat untuk mengelola baterai tersebut dengan aman.