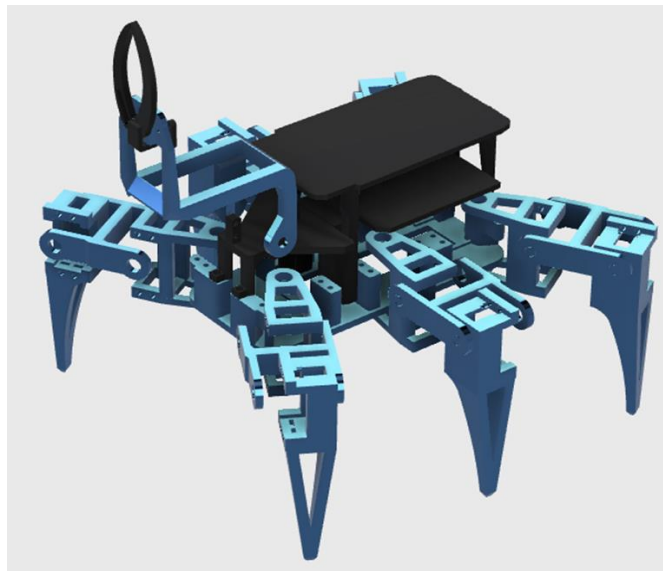


BAB III

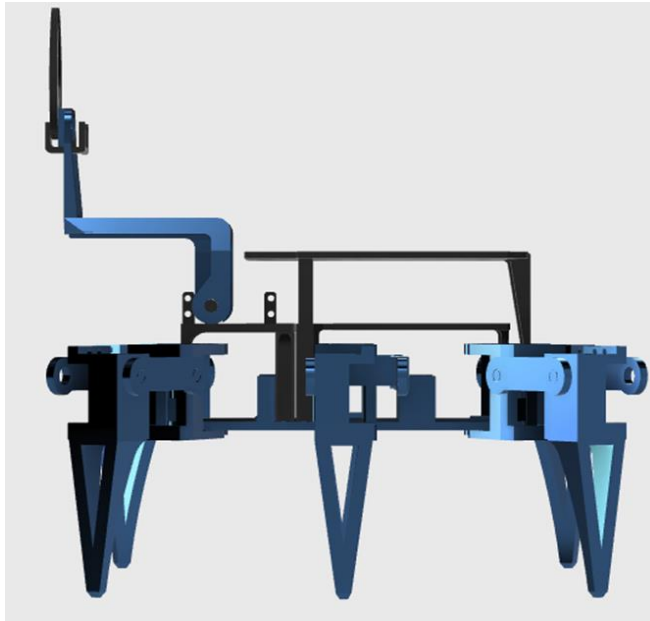
Perancangan dan Implementasi

3.1 Desain Robot

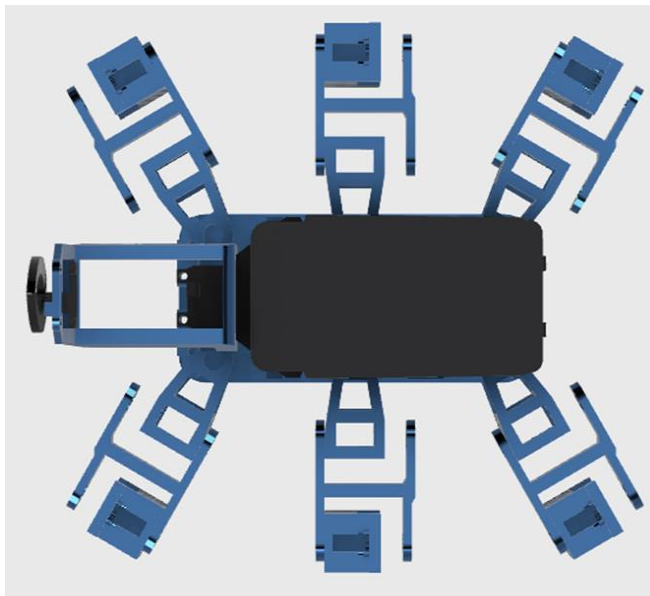
Desain robot Tempest diadaptasi dari laba-laba, namun hanya memiliki 6 kaki (heksapoda) yang didesain menggunakan aplikasi Fusion 360. Robot dibagi menjadi tiga bagian besar yaitu kaki, badan, dan lengan pencapit. Robot didesain berbentuk 6 kaki agar dapat menjaga keseimbangan saat melakukan gerakan, setiap kaki memiliki 3 *Degree of Freedom* (DOF) dan lengan pencapit yang memiliki 2 DOF. Robot memiliki dimensi 35 x 31.1 x 36 cm dalam kondisi berdiri seperti gambar berikut.



Gambar 3.1 Posisi Berdiri Robot Tempest



Gambar 3.2 Tampak Samping Posisi Berdiri Robot Tempest



Gambar 3.3 Tampak Atas Posisi Berdiri Robot Tempest



Gambar 3.4 Tampak Depan Posisi Berdiri Robot Tempest

Robot memiliki berat sebesar 3 kg bersama seluruh komponen yang digunakan dan robot seluruh bagian robot dicetak melalui 3D *print* yang menggunakan filamen berbahan PLA+.

3.2 Komponen Robot

3.2.1 Baterai Lithium Polymer 3S 5000 mAh 11.1 Volt



Gambar 3.5 Baterai Robot Tempest

Baterai yang digunakan robot memiliki kapasitas 5000 mAh dengan tegangan sebesar 11.1 V. Baterai berfungsi untuk menyuplai energi listrik ke seluruh komponen robot yang membutuhkannya, seperti mikrokontroler dan servo *driver*. Berat baterai adalah sekitar 418 gram.

3.2.2 Mikrokontroler Raspberry Pi 4B

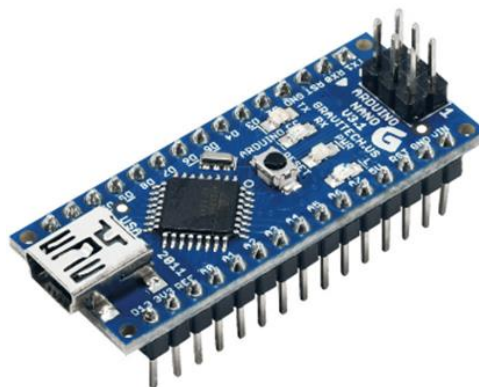


Gambar 3.6 Raspberry Pi 4B

Sumber: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>

Salah satu mikrokontroler yang digunakan oleh robot adalah Raspberry Pi 4B. Mikrokontroler ini berfungsi untuk menerima dan mengolah data dari berbagai macam sensor dan kamera yang digunakan oleh robot. Mikrokontroler ini memiliki berat sekitar 76 gram bersama *casing*. Raspberry juga tersambung dengan mikrokontroler lainnya, yaitu Arduino Nano dengan komunikasi serial.

3.2.3 Mikrokontroler Arduino Nano

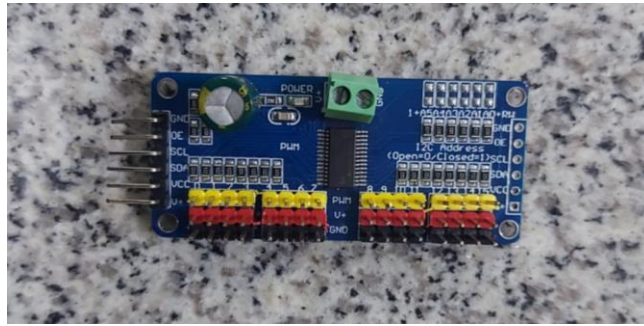


Gambar 3.7 Arduino Nano

Sumber: <https://docs.arduino.cc/hardware/nano/>

Mikrokontroler yang digunakan selanjutnya merupakan Arduino Nano, jumlah yang digunakan adalah 2 buah. Mikrokontroler ini digunakan untuk mengontrol servo *driver* dan menerima perintah dari Raspberry Pi 4B melalui komunikasi serial. Berat 2 buah mikrokontroler ini adalah 14 gram.

3.2.4 Servo Driver PCA9685



Gambar 3.8 Servo Driver PCA9685

Servo driver yang digunakan adalah PCA9685, jumlah yang digunakan adalah 2 buah. PCA9685 hanya dapat mengontrol 16 servo, sedangkan jumlah servo yang digunakan pada robot ada 20, sehingga diperlukan 2 buah servo driver. Servo driver ini mendapatkan derajat gerak servo dari Arduino Nano. Berat dari kedua servo driver ini mencapai 11 gram.

3.2.5 Servo MG996R



Gambar 3.9 Servo MG996R

Sumber: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1131873/ETC2/MG996R.html>

Salah satu servo yang digunakan adalah MG996R. Servo ini memiliki torsi sebesar 9.4 kgf.cm dan beroperasi pada tegangan 4.8 V. Servo yang digunakan mencapai 13 yang diletakkan pada kaki dan lengan robot. Berat total dari servo ini mencapai 715 gram.

3.2.6 Servo TD8120MG 20kg



Gambar 3.10 Servo TD8120MG

Sumber: https://cnc1.lv/PDF%20FILES/TD-8120MG_Digital_Servo.pdf

Servo yang digunakan selanjutnya adalah TD8120MG. Servo ini memiliki torsi sebesar 20.56 kg.cm dan beroperasi pada tegangan 4.8 V. Servo yang digunakan mencapai 6 yang diletakkan pada kaki robot. Berat total dari servo ini mencapai 360 gram.

3.2.7 Servo MG90S

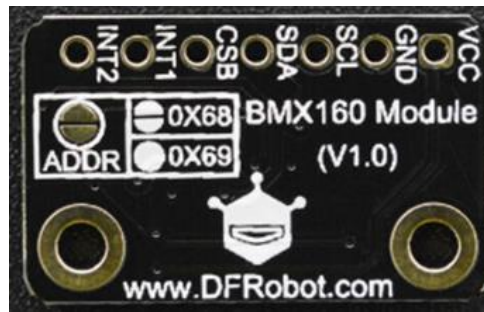


Gambar 3.11 Servo MG90S

Sumber: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1132104/ETC2/MG90S.html>

Servo yang digunakan selanjutnya adalah MG90S. Servo ini memiliki torsi sebesar 1.8 kg.cm dan beroperasi pada tegangan 4.8 V. Jumlah servo yang digunakan adalah 1, hanya pada capit robot. Berat dari servo ini mencapai 13.4 gram.

3.2.8 DFRobot BMX160 9 Axis Accelerometer Gyroscope



Gambar 3.12 DFRobot BMX160 9 Axis Accelerometer Gyroscope

Sumber: https://wiki.dfrobot.com/BMX160_9_Axis_Sensor_Module_SKU_SEN0373

Sensor *gyroscope* BMX160 ini digunakan sebagai *feedback* sistem agar robot dapat bergerak dengan lebih terarah. Sensor ini mendeteksi sudut putar *pitch*, *roll*, dan *yaw*. Sensor ini terhubung dengan Raspberry Pi, sehingga *feedback* dikirim langsung ke Raspberry Pi.

3.2.9 Webcam Logitech C505



Gambar 3.13 Webcam Logitech C505

Sumber: <https://www.logitech.com/assets/65955/9/c505-hd-webcam.ENG.pdf>

Webcam digunakan untuk keperluan *image processing* agar dapat membedakan korban asli dari *dummy*.. Berat webcam mencapai 124 gram.

3.2.10 Sensor *Time of Flight* (TOF) VL53L0X V2

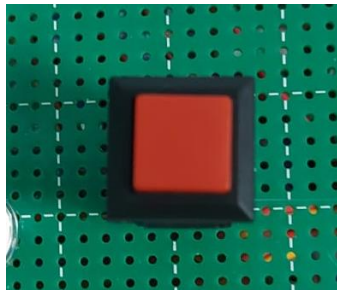


Gambar 3.14. Sensor TOF VL53L0X V2

Sumber: <https://esphome.io/components/sensor/vl53l0x.html>

Sensor ini digunakan untuk mendeteksi jarak antara robot dengan tembok atau rintangan lainnya pada *track*. Sensor yang digunakan berjumlah 3, yang ditempel pada samping kiri, samping kanan, dan bagian depan robot.

3.2.11 *Push Button*



Gambar 3.15 *Push Button*

Push button pada robot digunakan sebagai *trigger* agar robot mulai berjalan. Peraturan KRSRI 2024 menyatakan bahwa robot harus memiliki *push button* sebagai *trigger* untuk *start*.

3.2.12 LED



Gambar 3.16 LED

LED pada robot digunakan hanya sebagai penanda atau indikator bahwa robot sudah siap berjalan. LED terhubung dengan *push button*, sehingga kondisi lampu pada LED tergantung dengan *push button*.

3.2.13 Step-Down Buck Converter 20 A

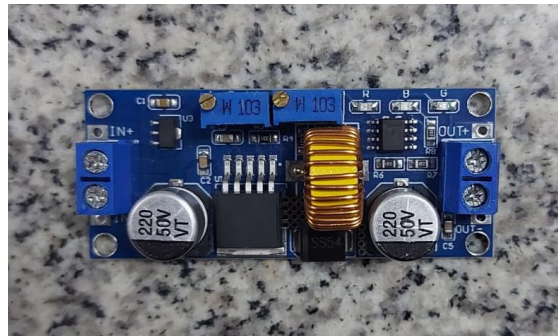


Gambar 3.17 Step-Down Buck Converter 20 A

Sumber: https://rajguruelectronics.com/Product/21609/300W%2020A%20DC-DC%20Buck%20Converter_datasheet.pdf

Sesuai dengan namanya, komponen ini digunakan untuk menurunkan tegangan yang dikeluarkan pada baterai. Dibutuhkan penyesuaian tegangan pada komponen PCA9685, karena PCA9685 beroperasi pada tegangan 5 V sedangkan baterai mengeluarkan 11 V. *Converter* yang digunakan 20 A karena PCA9685 terhubung dengan servo yang memiliki arus yang tergantung dengan bebannya. Arus semakin besar ketika beban yang dialami servo semakin besar juga.

3.2.14 Step-Down Buck Converter 5 A

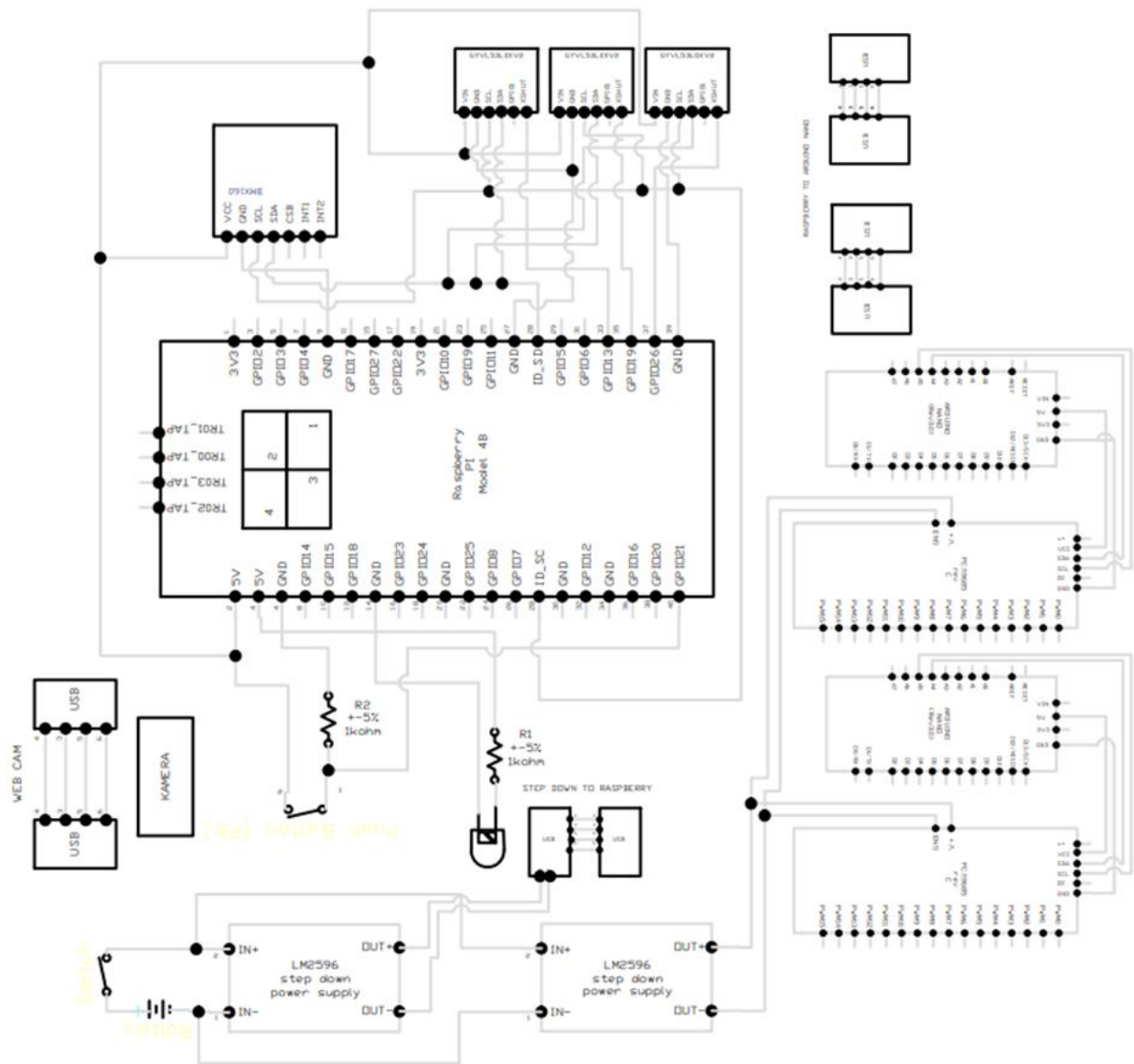


Gambar 3.18 Step-Down Buck Converter 5 A

Selain Buck Converter 20 A, digunakan juga Buck Converter 5 A untuk menurunkan tegangan dari baterai ke Raspberry Pi 4B. Raspberry Pi 4B beroperasi pada 5 V/3 A.

3.3 Wiring Diagram

Wiring diagram pada robot Tempest dapat diamati melalui gambar berikut.



Gambar 3.19 Wiring Diagram Robot Tempest

Baterai dihubungkan pada kedua Buck Converter, yang masing-masing terhubung dengan Raspberry Pi 4B dan PCA9685. Raspberry Pi 4B terhubung dengan sensor-sensor seperti, BMX160 dan TOF dengan komunikasi I2C. Webcam dan Arduino Nano akan terhubung dengan Raspberry Pi4B dengan USB Port.

3.4 Pengujian Servo dan Servo Driver

Servo yang digunakan memiliki tiga jenis, yaitu MG996R, TD8120, dan MG90s dan servo *driver* yang digunakan adalah PCA9685. Pengujian servo *driver* menggunakan aplikasi Arduino IDE dan mikrokontroler Arduino Nano. Pengujian servo *driver* berhasil jika setiap servo yang dihubungkan oleh *driver* dapat berputar 180° atau sesuai dengan derajat yang telah diatur pada Arduino IDE. Pengujian yang dilakukan pada setiap servo adalah derajat putar, tegangan kerja, arus kerja, dan arus *stall*. Berikut merupakan hasil pengujian servo.

Tabel 3.1 Pengujian Servo dan Servo Driver

Type	MG996R	TD8120	MG90s
Derajat Putar	180°	180°	180°
Tegangan	5 V	5 V	5 V
Arus Kerja	0.2 A	0.16 A	0.25 A
Arus <i>Stall</i>	2.2 A	2 A	0.78 A

```
PCA_SUDUT.ino
8   Serial.begin(9600);
9   board1.begin();
10  board1.setPWMFreq(60);
11
12  }
13
14
15  void loop()
16  {
17    board1.setPWM(0, 0, angleToPulse(180));
18    delay(1000);
19    board1.setPWM(0, 0, angleToPulse(0));
20    delay(1000);
21    board1.setPWM(0, 0, angleToPulse(180)); //a
22    // board1.setPWM(1, 0, angleToPulse(70)); //f
23    // board1.setPWM(2, 0, angleToPulse(30)); //b
24    // board1.setPWM(3, 0, angleToPulse(32)); //e
25    // board1.setPWM(4, 0, angleToPulse(65)); //c
26    // board1.setPWM(5, 0, angleToPulse(65)); //d
27    // board1.setPWM(6, 0, angleToPulse(0));
28    // board1.setPWM(7, 0, angleToPulse(180));
29    // board1.setPWM(8, 0, angleToPulse(180));
30    // board1.setPWM(9, 0, angleToPulse(0));
31    // board1.setPWM(10, 0, angleToPulse(0));
32    // board1.setPWM(11, 0, angleToPulse(180));
33  }
```

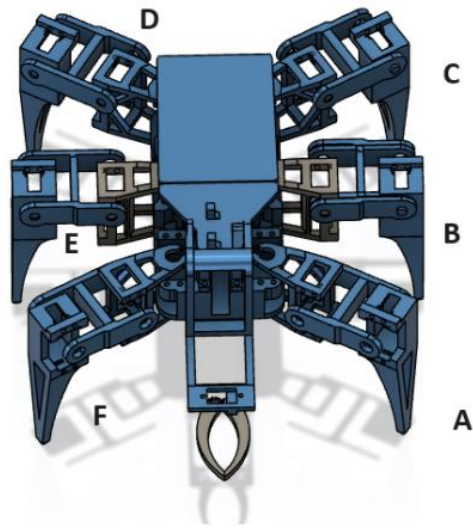
Gambar 3.20 Algoritma Pengujian Servo

Data di atas merupakan rata-rata dari seluruh jenis servo yang diuji menggunakan servo *driver* PCA9685 dan Arduino Nano sebagai mikrokontroler. Derajat putar merupakan sudut putar maksimum yang dapat dilakukan oleh servo, yang berarti seluruh servo yang digunakan dapat berputar sebesar 180° . Seluruh servo bekerja pada tegangan 5 V dan memiliki arus kerja di kisaran 0.2 A. Spesifikasi tegangan dan arus berfungsi untuk mengetahui *rating* servo untuk bekerja pada kondisi yang optimal, selain itu untuk menghindari adanya kerusakan servo karena kelebihan tegangan. Arus *stall* atau arus maksimum yang akan digunakan oleh servo ketika tidak bergerak, namun beban yang dialami oleh servo dalam keadaan maksimum. Data arus *stall* digunakan untuk mengetahui saat servo mengalami *stall* dalam proses pengujian *gait* saat masih menggunakan *power supply*, jika *power supply* menarik arus sebesar arus *stall* maka *gait* robot harus diperbaiki.

Servo MG996R diletakkan di antara badan dengan *coxa*, antara *tibia* dengan *femur*, dan lengan robot. Servo TD8120 diletakkan di antara *coxa* dengan *femur*, sedangkan servo MG90s diletakkan pada capit robot. Jumlah total servo yang digunakan ada 20 servo, yang terdiri dari 13 servo MG996R, 6 servo TD8120, dan 1 servo MG90s.

3.5 Pengujian Posisi Kaki Berdiri Robot

Stance robot merupakan hal yang harus ditentukan sebelum menjalankan robot berkaki. Proses pengujian menggunakan alat yang sama dengan pengujian servo, yaitu servo *driver* dan Arduino Nano. Hal pertama yang harus dilakukan sebelum menentukan derajat *stance* robot merupakan *zeroing* servo. *Zeroing* berarti menempatkan setiap servo pada posisi netral, sehingga proses kalibrasi servo menjadi lebih mudah dan tepat. Proses *zeroing* juga akan meningkatkan akurasi perputaran servo pada setiap kaki. *Zeroing* dimulai dengan membuat seluruh servo berada pada posisi 0° atau 180° tergantung dengan peletakan servo. Servo dipasang pada tempat yang sudah didesain dan diberikan baut dengan kencang. Penempatan servo pada tubuh robot tidak memungkinkan untuk setiap servo untuk dapat berputar 180° , oleh karena itu diperlukan jangkauan derajat putar yang dapat dilakukan setiap servo. Jangkauan awal servo merupakan posisi paling atas atau depan sebelum servo mengalami *stall*, sedangkan jangkauan akhir merupakan posisi paling bawah atau atas sebelum servo mengalami *stall*. Berikut merupakan tabel hasil pengujian, sehingga jangkauan derajat putar servo dapat diketahui.



Gambar 3.21 Bagian Kaki-Kaki Robot

Tabel 3.2 Hasil Derajat Putar Setiap Servo

Bagian	No. Servo	Derajat Putar Servo (°)
A	Servo 11 (<i>Coxa</i> / θ_{Ca})	0 – 130
	Servo 1 (<i>Femur</i> / θ_{Fa})	180 – 30
	Servo 17 (<i>Tibia</i> / θ_{Ta})	0 – 180
B	Servo 12 (<i>Coxa</i> / θ_{Cb})	0 – 70
	Servo 2 (<i>Femur</i> / θ_{Fb})	0 – 160
	Servo 18 (<i>Tibia</i> / θ_{Tb})	180 – 0
C	Servo 13 (<i>Coxa</i> / θ_{Cc})	0 – 130
	Servo 3 (<i>Femur</i> / θ_{Fc})	0 – 160
	Servo 19 (<i>Tibia</i> / θ_{Tc})	180 – 0
D	Servo 14 (<i>Coxa</i> / θ_{Cd})	0 – 180
	Servo 4 (<i>Femur</i> / θ_{Fd})	180 – 30
	Servo 20 (<i>Tibia</i> / θ_{Td})	0 – 180
E	Servo 15 (<i>Coxa</i> / θ_{Ce})	0 – 70
	Servo 5 (<i>Femur</i> / θ_{Fe})	180 – 20
	Servo 21 (<i>Tibia</i> / θ_{Te})	0 – 180
F	Servo 16 (<i>Coxa</i> / θ_{Cf})	0 – 130
	Servo 6 (<i>Femur</i> / θ_{Ff})	0 – 160
	Servo 22 (<i>Tibia</i> / θ_{Tf})	180 – 0
Lengan Pencapit	Servo 23	0 – 120
Pencapit	Servo 24	0 – 180

Derajat *stance* robot dapat dilihat melalui tabel berikut.

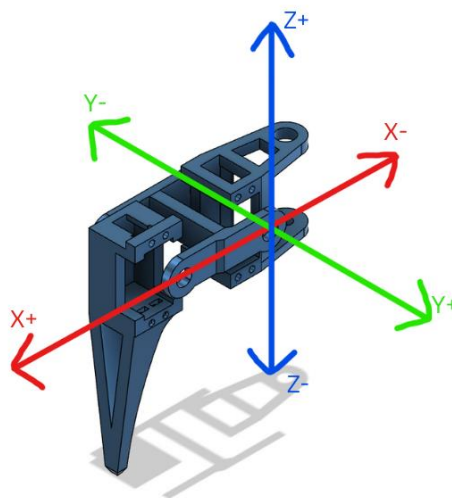
Tabel 3.3 Hasil Derajat *Stance* Setiap Servo

Bagian	No. Servo	Derajat <i>Stance</i> Servo (°)
A	Servo 11 (<i>Coxa</i> / θ_{Ca})	65
	Servo 1 (<i>Femur</i> / θ_{Fa})	90
	Servo 17 (<i>Tibia</i> / θ_{Ta})	140
B	Servo 12 (<i>Coxa</i> / θ_{Cb})	30
	Servo 2 (<i>Femur</i> / θ_{Fb})	80
	Servo 18 (<i>Tibia</i> / θ_{Tb})	5
C	Servo 13 (<i>Coxa</i> / θ_{Cc})	85
	Servo 3 (<i>Femur</i> / θ_{Fc})	90
	Servo 19 (<i>Tibia</i> / θ_{Tc})	15
D	Servo 14 (<i>Coxa</i> / θ_{Cd})	55
	Servo 4 (<i>Femur</i> / θ_{Fd})	105
	Servo 20 (<i>Tibia</i> / θ_{Td})	180
E	Servo 15 (<i>Coxa</i> / θ_{Ce})	40
	Servo 5 (<i>Femur</i> / θ_{Fe})	110
	Servo 21 (<i>Tibia</i> / θ_{Te})	180
F	Servo 16 (<i>Coxa</i> / θ_{Cf})	80
	Servo 6 (<i>Femur</i> / θ_{Ff})	90
	Servo 22 (<i>Tibia</i> / θ_{Tf})	20
Lengan Pencapit	Servo 23	90
Pencapit	Servo 24	5

Derajat-derajat tersebut merupakan posisi robot ketika robot dapat berdiri dengan stabil atau optimal.

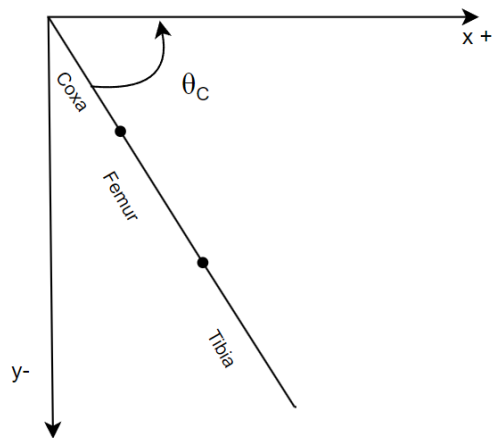
3.6 Normalisasi 0 derajat Kaki Robot

Normalisasi 0 derajat dapat dijelaskan melalui Gambar 3.22 hingga Gambar 3.28.

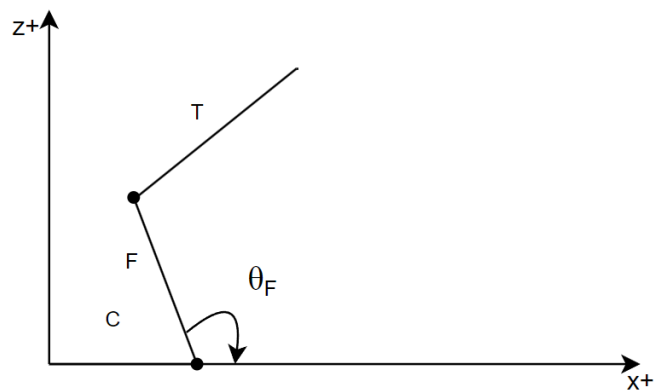


Gambar 3.22 Bidang Koordinat Kaki Kanan Robot (Tampak Depan)

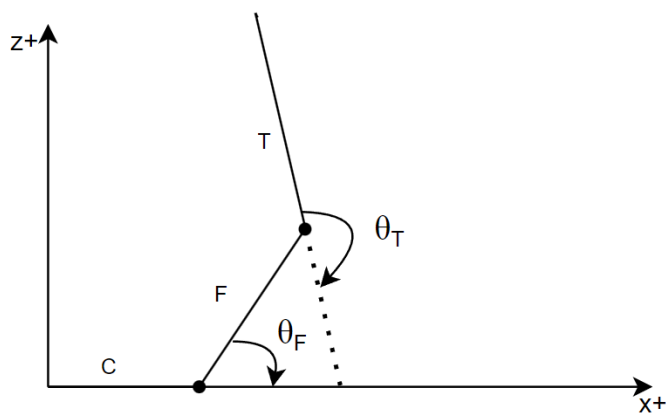
Bidang koordinat digunakan pada kaki kiri dan kaki kanan robot, sehingga terdapat perbedaan pada sumbu X dan Y untuk kaki kiri dan kanan.



Gambar 3.23 Normalisasi 0 derajat *Coxa* Bagian Kanan (Tampak Atas)

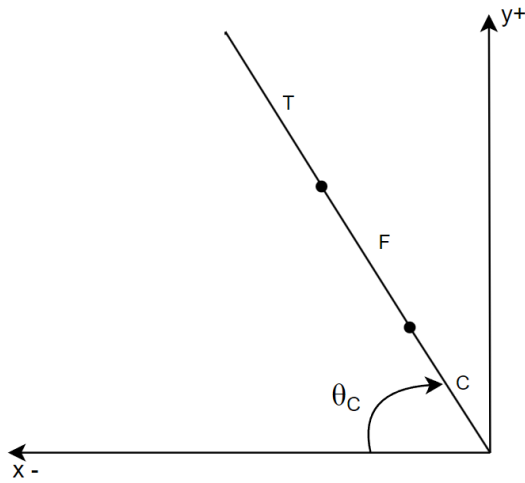


Gambar 3.24 Normalisasi 0 derajat *Femur* Bagian Kanan (Tampak Belakang)

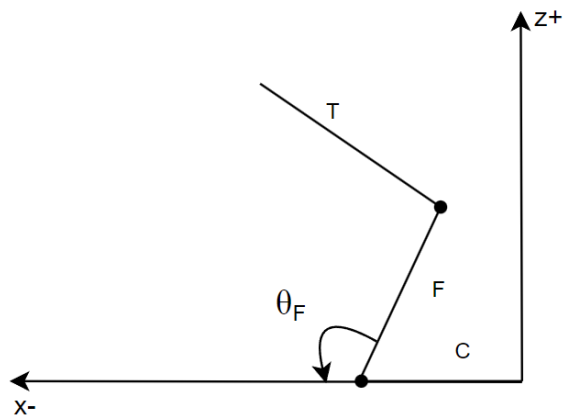


Gambar 3.25 Normalisasi 0 derajat *Tibia* Bagian Kanan (Tampak Belakang)

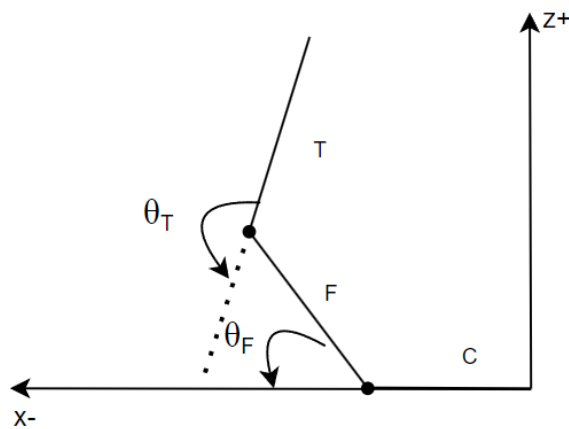
Posisi 0 derajat pada kaki kanan robot telah diilustrasikan pada Gambar 3.23 hingga Gambar 3.25 di atas. Derajat putar kaki kanan (D, E, dan F) untuk masing-masing bagian (*Coxa* θ_C , *Femur* θ_F , dan *Tibia* θ_T) dapat dilihat pada Tabel 3.2.



Gambar 3.26 Normalisasi 0 derajat *Coxa* Bagian Kiri (Tampak Atas)



Gambar 3.27 Normalisasi 0 derajat *Femur* Bagian Kiri (Tampak Belakang)

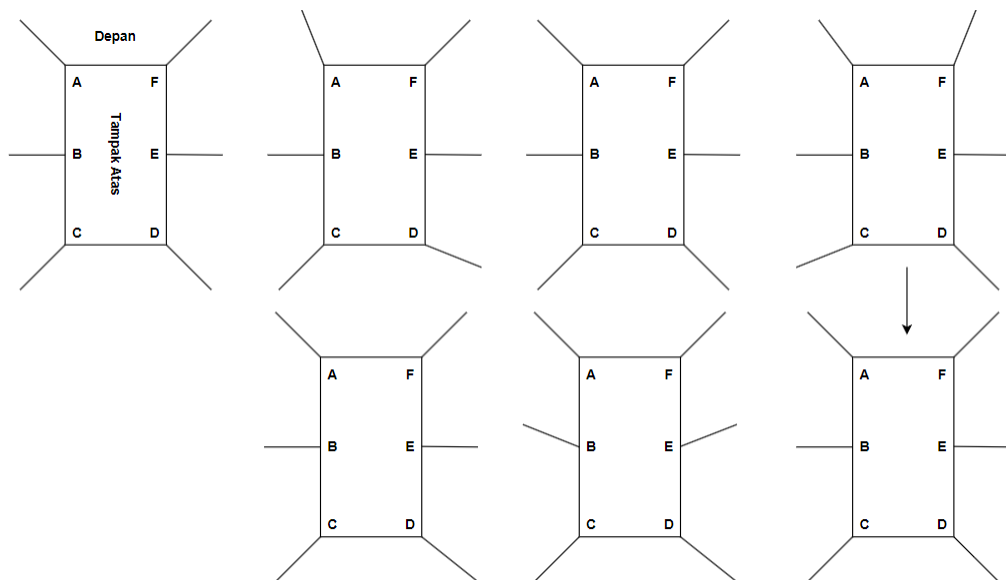


Gambar 3.28 Normalisasi 0 derajat *Tibia* Bagian Kiri (Tampak Belakang)

Posisi 0 derajat pada kaki kanan robot telah diilustrasikan pada Gambar 3.26 hingga Gambar 3.28 di atas. Derajat putar kaki kiri (A, B, dan C) untuk masing-masing bagian (*Coxa* θ_C , *Femur* θ_F , dan *Tibia* θ_T) dapat dilihat pada Tabel 3.2.

3.7 Gait Robot Heksapoda

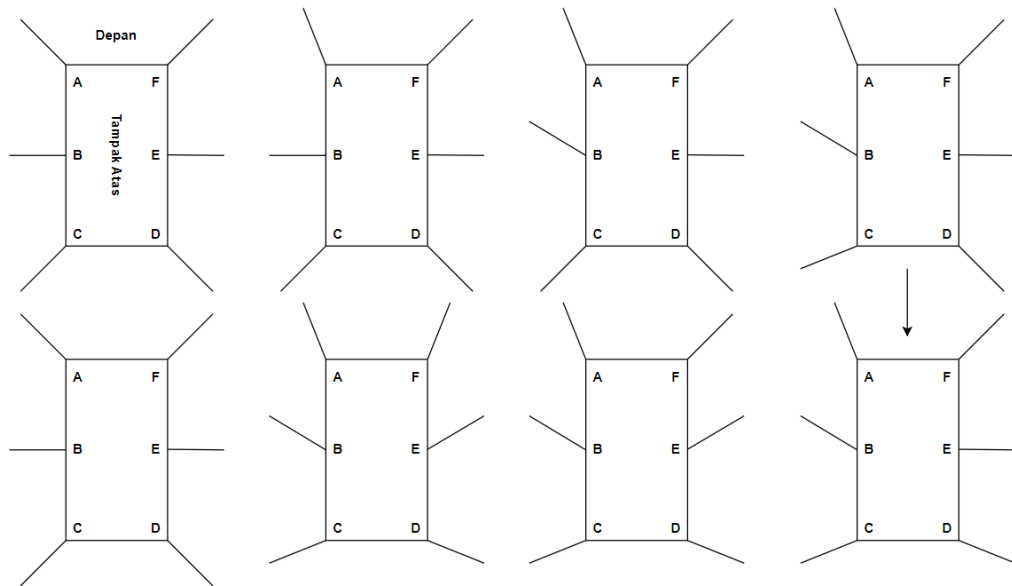
Terdapat beberapa *gait* yang dapat digunakan pada robot heksapoda. Pertama, ada *quadruped gait* yang berarti selalu ada 4 kaki yang menopang robot sementara dua kaki lainnya diayun ke depan. *Gait* dapat diamati melalui ilustrasi gambar di bawah.



Gambar 3.29 Ilustrasi *Quadruped Gait* Robot Heksapoda

Gait ini memberikan stabilitas yang sangat baik karena setiap saat ada 4 kaki yang menopang bobot robot, namun kecepatan robot dikorbankan untuk mendapatkan stabilitas tersebut. Pergerakan robot akan menjadi sangat lambat karena hanya ada 2 kaki yang memberikan efek maju pada robot pada saat yang bersamaan, sehingga *gait* ini tidak dipilih.

Gait selanjutnya adalah *gait wave* yang berarti pergerakan kaki robot akan bergantian secara urut seperti ombak. Hanya 1 kaki yang melakukan pergerakan maju pada setiap tahap, 5 kaki lainnya akan menopang bobot robot. *Gait* dapat diamati melalui ilustrasi gambar di bawah ini.

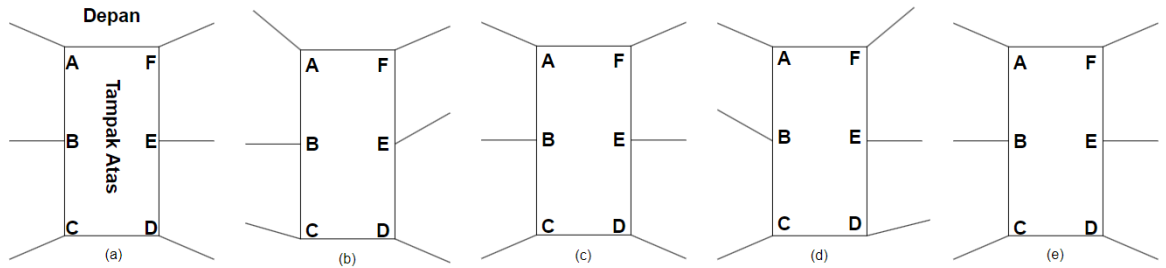


Gambar 3.30 Ilustrasi Wave Gait Robot Heksapoda

Gait ini memiliki stabilitas yang lebih baik dari *quadruped gait* namun sama seperti *quadruped gait*, stabilitas akan mengorbankan kecepatan gerak robot. Kecepatan gerak robot akan sangat lambat, lebih lambat dari *quadruped gait* karena kaki bergerak secara satu per satu, sehingga *gait* ini tidak dipilih.

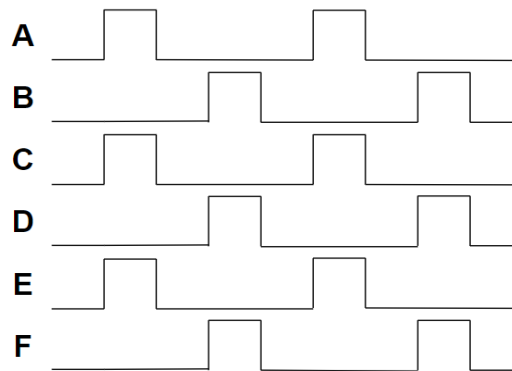
Gait selanjutnya merupakan *crawl gait* yang berarti semua kaki akan menopang robot ketika melakukan perjalanan. *Gait* ini memiliki pola gerakan yang serupa dengan *wave gait*, hanya ada satu kaki yang melakukan pergerakan maju pada setiap tahap, sedangkan kaki yang lain diam menopang robot. *Gait* ini akan menyeret kaki, sehingga tidak ada kaki yang perlu diangkat. Kondisi *track* KRSRI 2024 yang memiliki banyak rintangan seperti kelereng, batu hias, dan rantai pecah membuat *gait* ini tidak optimal. Kaki yang terus diseret akan meningkatkan resiko kerusakan karena harus mendorong rintangan-rintangan tersebut, sehingga *gait* tidak dipilih

Gait selanjutnya merupakan *gait* yang paling umum digunakan dalam robot heksapoda karena memberikan stabilitas yang baik pada saat berjalan di permukaan yang rata maupun yang kasar atau banyak rintangan, seperti dengan kondisi *track* robot KRSRI yang memiliki banyak rintangan berupa rantai pecah, rantai dengan batu hias, rantai dengan kelereng, dan rantai dengan gabungan batu hias dan kelereng. *Gait* juga tidak mengorbankan kecepatan robot seperti *gait-gait* sebelumnya. Berikut merupakan ilustrasi dari *gait* tripod yang digunakan.



Gambar 3.31 Ilustrasi *Tripod Gait* Maju Robot

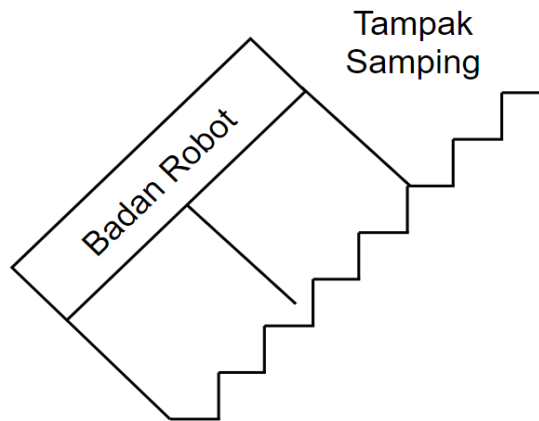
Gait ini memiliki stabilitas dan kecepatan yang sesuai dengan *track* KRSRI 2024, sehingga *tripod gait* menjadi *gait* pilihan pada robot Tempest. Penjelasan lengkap mengenai *gait* dapat diamati melalui gambar di bawah ini.



Gambar 3.32 *Timing Diagram Tripod Gait*

Posisi robot dimulai pada posisi *stance* yaitu keadaan *low* pada *timing* diagram. Tahap pertama, kaki A bersama dengan kaki C dan E maju ke depan, keadaan *high* pada *timing* diagram dan kaki lainnya masih berada di posisi *stance*, menyentuh tanah. Kaki A, C, dan E kembali ke posisi *stance* awal, sehingga keadaan *timing* diagram menjadi *low* kembali. Posisi kaki B, D, dan F akan mengangkat dari tanah walaupun belum bergerak maju ketika posisi kaki A, C, dan E kembali ke posisi *stance*, sehingga keadaan kaki A, C, dan E masih *low* pada *timing* diagram. Posisi kaki setelah itu adalah kaki B, D, dan F maju dan posisi *timing* diagram dalam keadaan *high* dan kaki lainnya masih berada di posisi *stance*, menyentuh tanah. Kaki B, D, dan F setelah itu kembali ke tanah dan keadaan dalam *timing* diagram menjadi *low*, setelah itu siklus berulang kembali. *Gait* untuk belok ke kanan dan kiri memiliki pola yang sama ketika jalan maju, perbedaan hanya pada derajat pada servo.

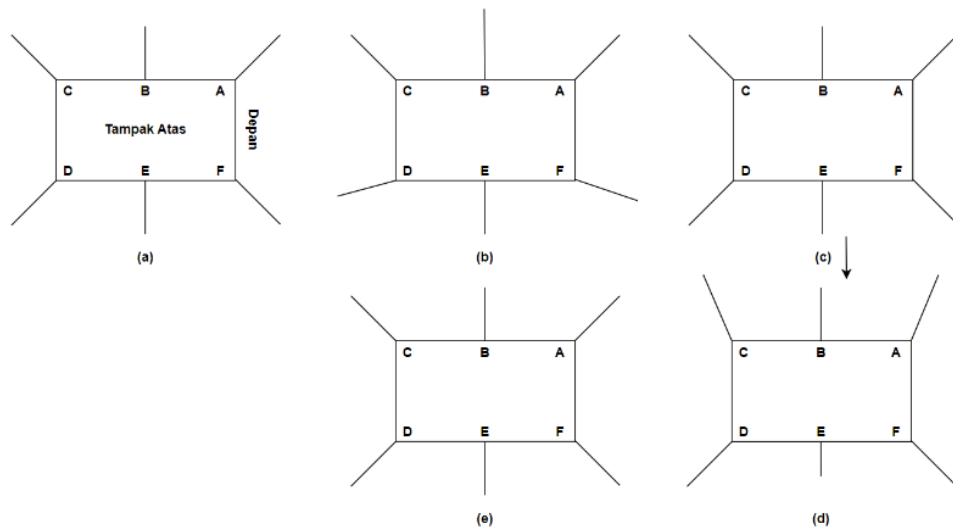
Pada *track* KRSRI 2024, terdapat tangga yang harus dinaiki robot. *Tripod gait* dapat diterapkan untuk menaiki tangga, namun harus ada modifikasi pada pola gerakan robot. Ilustrasi kaki-kaki robot pada saat menaiki tangga dapat diamati melalui gambar di bawah.



Gambar 3.33 Ilustrasi *Gait* Robot Saat Naik Tangga

Kaki tengah robot tidak menyentuh lantai, sehingga robot tidak dapat menaiki anak tangga dengan baik dan posisi robot tidak berubah. Pola *gait* robot tidak memerlukan modifikasi, hanya perubahan derajat pada servo yang menghubungkan femur dan coxa pada kaki tengah robot. Servo yang diubah adalah servo 2 dan servo 5, derajatnya diubah agar menjadi lebih turun dengan tujuan kaki tengah robot dapat menyentuh lantai agar dapat membantu pergerakan saat naik tangga.

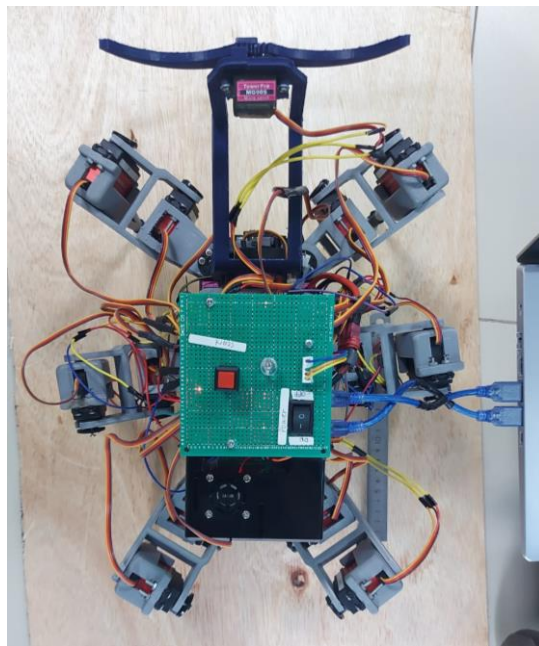
Terdapat pula *gait* lain yang dapat diimplementasikan selain *tripod gait*, yaitu *quadruped gait* dan *tripod gait (strafe)*. *Quadruped gait* dapat diimplementasikan karena memberikan kestabilan yang lebih baik dibandingkan dengan *tripod gait*. *Tripod gait (strafe)* juga dapat diimplementasikan, berikut merupakan ilustrasi pola pergerakan saat *strafing*.



Gambar 3.34 Ilustrasi Tripod Gait (Strafe) Maju Robot

3.8 Pengujian Gait Robot

Pengujian masing-masing *gait* robot dilakukan seperti gambar-gambar di bawah ini.



Gambar 3.35 Pengujian Gait Robot

Pengujian setiap *gait* robot dilakukan dengan menghubungkan mikroprosesor Arduino Nano dengan laptop. Arduino Nano digunakan untuk menggerakkan setiap servo pada robot, oleh karena itu tidak perlu menggunakan Raspberry Pi dalam pengujian ini.



(a)



(b)

Gambar 3.36 Pengukuran Jarak Maju *Gait*
 (a) Posisi Awal Robot (b) Posisi Akhir Robot

Kecepatan setiap *gait* diukur dengan jarak ditempuh robot yang dibagi dengan waktu robot untuk mencapai jarak tersebut. Sebagai contoh, pada Gambar 3.36 (a) menunjukkan posisi awal robot dan pada Gambar 3.36 (b) menunjukkan posisi akhir robot dengan menggunakan *quadruped gait*. Waktu yang dibutuhkan untuk robot mencapai posisi akhir adalah 3.38 s, oleh karena itu kecepatan yang dimiliki robot dengan menggunakan *quadruped gait* adalah 0.77 s. Berikut merupakan tabel kecepatan untuk setiap *gait* yang diuji.

Tabel 3.34 Hasil Derajat *Stance* Setiap Servo

Type <i>gait</i>	Kecepatan (cm/s)
<i>Quadruped</i>	0.77
<i>Wave</i>	0.47
<i>Crawl</i>	0.37
<i>Tripod</i>	2.25