

RANCANG BANGUN KENDALI ROBOT HEXAPOD MENGGUNAKAN SMARTPHONE

Syahban Rangkuti ¹

Program Studi Teknik Elektro Universitas Faletehan Bandung

Email: syahban3477@gmail.com

ABSTRACTS: *Hexapod robot technology has developed rapidly, starting from robots that can work independently (autonomously) to robots that are controlled via smartphones. In this research, we will design a 6-legged robot (hexapod) that can move stably and follow the commands given. The ESP32 DevKit microcontroller module is used as the center of the hexapod robot control system because it has many digital inputs and outputs, has I2C serial communication, and also has wireless communication. One method used to obtain stable robot movement is to use an inertial measurement unit (IMU) sensor, which is capable of producing angular movement data in a 3-dimensional plane on the pitch (ϕ), roll (θ) and yaw (ψ) axes. The shape of the hexapod robot body is made so that it can be easily integrated with the robot legs that will be used. To drive the robot's legs, a servo motor is used for each joint. Each robot leg consists of 3 joints, so the 6-legged robot drive system requires 18 units of servo motors. Inverse kinematics with a PID control system can control robot movement by applying a tripod gait step pattern. The smartphone is used to control the direction of movement of the hexapod robot in forward, backward, left turn, right turn, and stop directions using a Bluetooth or WiFi connection. Based on the test results, the hexapod robot can be appropriately controlled according to the commands given via smartphone. The test results of 5 meters in a straight direction for forward and backward movement produced a robot speed of around 20cm/S with a deviation error ranging between 7% - 9% from the target point.*

Keyword: *Robot Hexapod; IMU; ESP32; Smartphone.*

ABSTRAK: Teknologi robot *hexapod* telah berkembang pesat, mulai dari robot yang dapat bekerja secara mandiri (*autonomous*) maupun robot yang dikendalikan melalui *smartphone*. pada penelitian ini akan merancang bangun robot berkaki 6 (*hexapod*) yang mampu bergerak dengan stabil dan sesuai dengan perintah yang diberikan. Modul mikrokontroler ESP32 DevKit digunakan sebagai pusat sistem kendali robot *hexapod* karena memiliki banyak input dan output digital serta memiliki komunikasi serial *I2C* dan juga memiliki komunikasi *wireless*. Salah satu cara yang digunakan untuk mendapatkan gerakan robot stabil dapat menggunakan sensor *inertial measurement unit (IMU)* yang mampu menghasilkan data pergerakan sudut pada bidang 3 dimensi pada sumbu *pitch* (ϕ), *roll* (θ), dan *yaw* (ψ). Bentuk badan robot *hexapod* dibuat agar dapat dengan mudah diintegrasikan dengan kaki robot yang akan digunakan. Sebagai penggerak kaki robot digunakan motor *servo* untuk setiap sendinya, setiap kaki robot terdiri dari 3 sendi, sehingga untuk sistem penggerak robot berkaki 6 membutuhkan 18 unit motor *servo*. *Invers* kinematik dengan sistem kendali *PID* dapat digunakan untuk mengatur gerak robot dengan menerapkan pola langkah *tripod gait*. *Smartphone* digunakan mengatur arah gerakan robot *hexapod* pada arah maju, mundur, belok kiri, belok kanan, dan berhenti dengan menggunakan koneksi *bluetooth* atau *wifi*. Berdasarkan hasil pengujian robot *hexapod* dapat dikendalikan dengan baik sesuai dengan perintah yang diberikan melalui *smartphone*. Hasil pengujian sejauh 5 meter pada arah lurus untuk gerakan maju dan mundur menghasilkan kecepatan robot sekitar 20cm/S dengan *error* simpangan berkisar antara 7% - 9% dari titik yang dituju.

Kata Kunci: *Robot Hexapod; IMU; ESP32; Smartphone.*

PENDAHULUAN

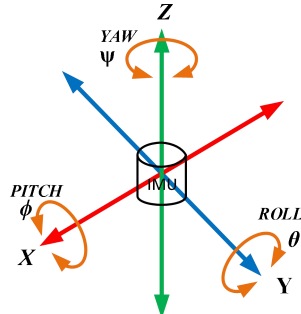
Robot secara umum dikelompokkan menjadi 2 kelompok yaitu robot yang dapat berpindah tempat dikenal dengan istilah *mobile robot* dan robot yang tak dapat berpindah tempat atau dikenal dengan istilah *manipulator robot*. Perpindahan robot disesuaikan dengan media tempat Bergeraknya seperti robot daratan, robot air dan robot terbang. Perpindahan robot disesuaikan dengan media tempat Bergeraknya seperti robot daratan, robot air dan robot terbang. Untuk robot daratan dibagi menjadi 2 tipe yaitu robot beroda dan robot berkaki. Salah satu jenis robot berkaki yang banyak dikembangkan adalah robot berkaki 6 atau dikenal dengan istilah robot *hexapod*.

Robot *hexapod* harus mampu berjalan atau bergerak dengan baik sehingga robot dapat bergerak sesuai dengan arah pergerakan yang diharapkan. Secara umum langkah robot *hexapod* dibagi menjadi 3 pola langkah yaitu *metachronal*, *ripple*, dan *tripod*. Pola langkah *metachronal* adalah menggerakkan kaki robot secara bergantian satu persatu, sedangkan pola langkah *ripple* adalah menggerakkan 2 kaki robot secara bersamaan dan dilanjutkan dengan 2 kaki yang lainnya secara bergantian, untuk pola langkah *tripod* yaitu menggerakkan 3 kaki robot secara bersamaan dan dilanjutkan dengan 3 kaki yang lainnya secara bergantian [1]. Secara umum robot *hexapod* yang menggunakan pola langkah *tripod* akan bergerak atau berpindah lebih cepat dibandingkan dengan pola langkah *metachronal* ataupun *ripple*.

Untuk menjaga kestabilan robot *hexapod* saat berjalan dapat menggunakan perangkat sensor *Inertial Measurement Unit (IMU)* [2]. Sensor *IMU* dapat memperkirakan posisi relatif dan akselerasi dari gerakan

¹ Prodi Teknik Elektro, Universitas Faletehan.

robot *hexapod*. Sensor *IMU 6 DOF* mampu melakukan pengukuran *gyroscope* dan *accelerometer*. Data dari *gyroscope* dapat digunakan untuk mempertahankan sudut motor, kecepatan putaran sudut *gyroscope* dinyatakan dalam satuan derajat perdetik ($^{\circ}/S$). Data dari *accelerometer* digunakan untuk menghasilkan putaran *servo* untuk memperoleh posisi yang diinginkan. Gerakan yang dihasilkan oleh putaran *servo* yang diintegrasikan dengan perangkat mekanik pada bentuk tertentu akan menghasilkan sejumlah gerakan yang dikenal dengan istilah *degree of freedom* [3]. Pada bidang 3 dimensi arah pergerakan sudut dibagi menjadi 3 bagian yaitu *pitch* (ϕ), *roll* (θ), dan *yaw* (ψ) dengan akselerasi gerakan mengacu pada sumbu x, y dan z.



■ Gambar 1. Arah gerakan sensor *IMU 6 DOF*

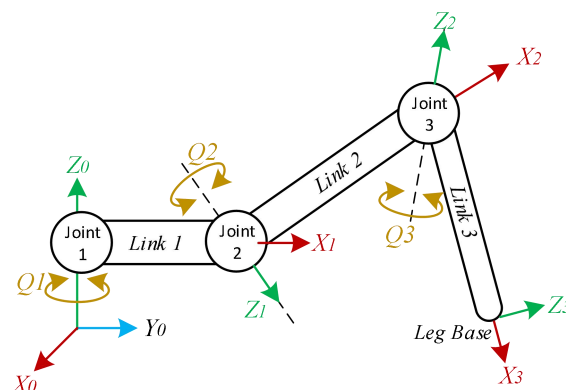
Motor *servo* akan menghasilkan gerakan berputar berdasarkan sinyal digital yang diberikan padanya. Instruksi berupa sinyal digital akan menggerakkan output *servo* pada sudut tertentu dengan kecepatan putar tertentu. Output putaran *servo* akan menghasilkan gerakan mekanik yang akan menggerakkan kaki robot *hexapod* sesuai dengan yang diinginkan. Untuk mengatur kecepatan putaran dan sudut motor *servo* maka digunakan sistem kendali *PID* [4]. Sensor *IMU* akan digunakan sebagai data input atau umpan balik (*feedback*) untuk menjaga kestabilan robot *hexapod*.

Penggabungan antara sensor *IMU* dan motor *servo* dengan konstruksi mekanik tertentu dapat membentuk suatu robot, misalnya robot *hexapod*. Agar robot *hexapod* dapat bergerak sesuai dengan yang diinginkan maka dapat dikendalikan dengan menggunakan *smartphone*, sehingga untuk pengendaliannya dapat dilakukan dengan menggunakan teknologi *wireless* seperti *bluetooth* [5] atau *wifi* [6].

METODOLOGI PENELITIAN

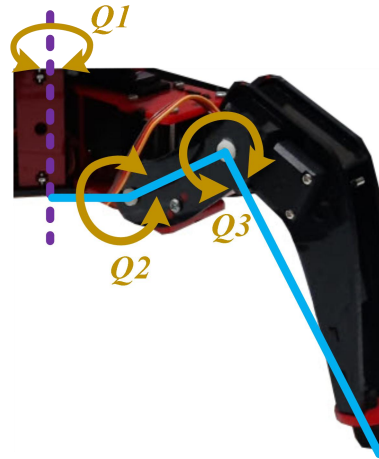
Robot yang dirancang pada penelitian ini adalah robot berkaki 6, sesuai dengan namanya berarti robot *hexapod* mempunyai 6 kaki untuk pergerakannya. Perangkat robot *hexapod* terdiri dari perangkat yang digerakkan yaitu berupa mekanik robot dan perangkat elektronik sebagai sistem kendali gerakan robot [7]. Perangkat mekanik dapat berupa bagian dari tempat menyimpan perangkat elektronik dan dapat juga merupakan objek yang digerakkan secara elektronik agar diperoleh gerakan robot seperti yang diinginkan.

Untuk mengatur gerakan robot *hexapod* dapat digunakan analisa *invers kinematik* [8]. *Invers kinematik* digunakan untuk menganalisa gerakan kinematik dari robot untuk mendapatkan besar sudut dari setiap *joint* jika mempunyai data koordinat posisi (x, y, z) [9]. Penentuan besar setiap sudut *joint* dilakukan oleh *servo*, setiap *link* terhubung pada *servo*. Ujung dari *link* terakhir dikenal dengan istilah *leg base* yang akan menentukan posisi robot *hexapod* [10].



■ Gambar 2. Konsep dasar gerakan sudut pada kaki robot

Titik sendi atau *Joint* merupakan output putaran dari motor *servo*, karena pada robot *hexapod* menggunakan 6 kaki maka pada setiap kaki pasti mempunyai *Joint 1*. *Joint 1* pada robot *hexapod* dikenal juga dengan istilah *Coxa*. Motor *servo* untuk *joint* pada *coxa* ini dipasangkan pada badan (*body*) robot. *Joint 1* akan menggerakkan *Link 1* sesuai dengan putaran motor *servo* yang berfungsi sebagai *Joint 1*. Besar sudut yang dihasilkan oleh *Joint 1* ($Q1$) sesuai dengan putaran output yang dihasilkan oleh *servo* yang berfungsi sebagai *Joint 1*. Ketika *Joint 1* bergerak maka secara otomatis akan membawa seluruh objek yang ada dibelakangnya, dimulai dari *Link 1* sampai *Leg Base*.

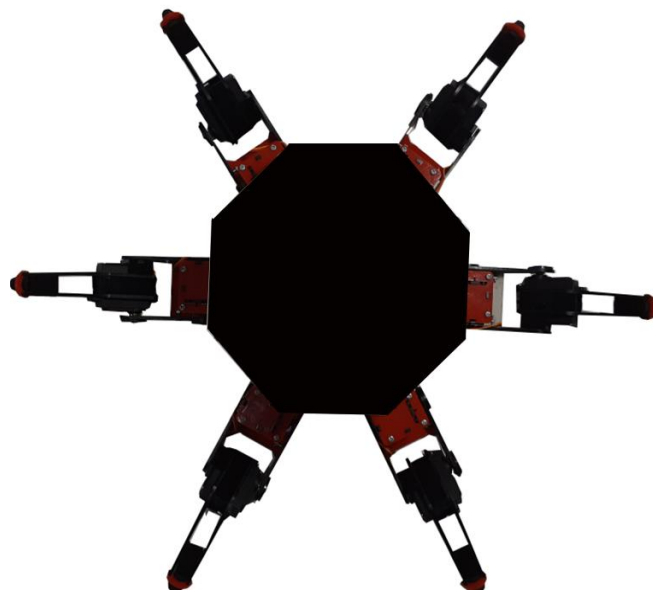


■ **Gambar 3.** Gerakan sudut pada kaki robot

Perubahan sudut pada *Link 1* akan membawa *Joint 2* sehingga dapat mengatur pergerakan untuk *Link 2*. Arah pergerakan dari *Link 2* bergantung dengan besar sudut $Q2$ yang dihasilkan pada output *servo* dari *Joint 2*, begitu juga dengan arah pergerakan dari *Link 3* bergantung dengan besar sudut *Joint 3* ($Q3$). Prinsip kerja untuk setiap kaki pada robot *hexapod* adalah sama.

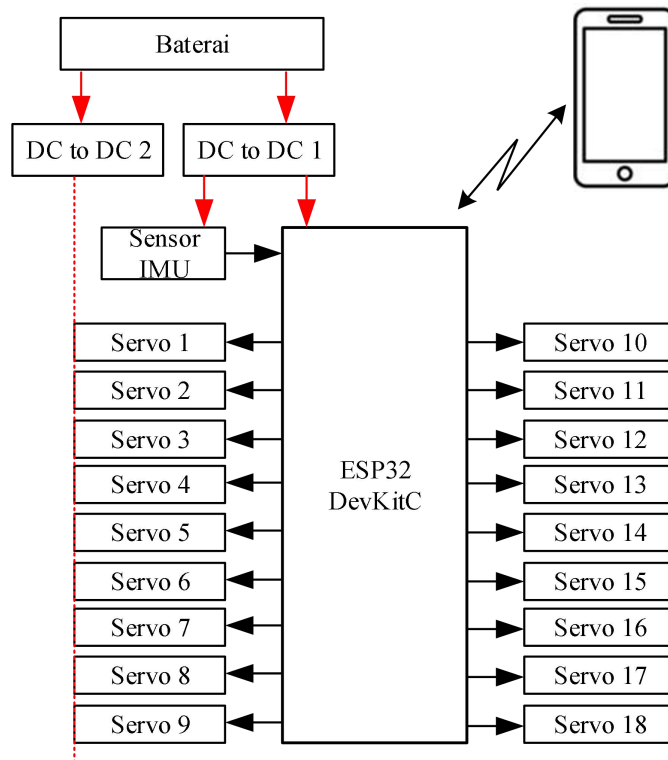
Perancangan Perangkat Keras

Konstruksi robot yang dibangun dapat beragam tergantung dengan keinginan dan juga pertimbangan untuk pengembangan dari fungsi robot *hexapod* selanjutnya. Gambar di bawah memperlihatkan desain mekanik dari robot *hexapod* yang akan dibangun. Jarak untuk kaki robot diatur sedemikian rupa sehingga jaraknya simetris antara satu dengan yang lainnya.



■ **Gambar 4.** Gerakan sudut pada kaki robot

Berdasarkan dari jumlah kaki dan juga jumlah sendi pada setiap kaki robot ditambah dengan sensor *IMU* dan sumber listrik untuk penggerak robot maka dapat dibuat diagram blok untuk rangkaian elektronik untuk robot *hexapod*. Diagram blok perangkat elektronik yang digunakan dapat dilihat pada gambar diagram blok perangkat keras elektronik robot *hexapod*.



■ **Gambar 5.** Diagram blok elektronik robot *hexapod*

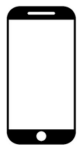
Pada robot *hexapod* terdapat komponen elektromekanik yang berfungsi sebagai penggerak atau aktuator yaitu 18 unit motor *servo*. Jenis motor servo yang digunakan harus yang sejenis sehingga pengaturannya lebih mudah dilakukan. Motor *servo* yang digunakan adalah motor *servo* yang membutuhkan sumber listrik searah (*direct current*). Besar sudut putaran output yang dihasilkan oleh *servo* bervariasi dari 90° , 180° , dan 360° , pada perancangan ini menggunakan motor *servo* dengan sudut putaran maksimum sebesar 180° .

Untuk perangkat sistem kendali kendalanya menggunakan modul ESP32 DevKitC V4. Modul ESP32 ini memiliki banyak pin input output digital, komunikasi serial *I2C*, dan juga menyediakan komunikasi *wireless* berupa *bluetooth* dan *wifi* sehingga dapat berkomunikasi secara langsung pada *smartphone*. Data dari sensor *IMU* merupakan data input yang akan dibaca oleh ESP32 DevKitC melalui komunikasi serial *Inter Integrated Circuit (I2C)*. Output digital dari ESP32 berguna untuk memberikan sinyal digital pada seluruh motor *servo* yang akan digunakan sehingga output motor *servo* dapat berputar sesuai dengan besar sudut yang diberikan.

Sumber listrik pada modul elektronik seperti sensor *IMU* dan ESP32 menggunakan sumber listrik searah yang terpisah dengan komponen elektromekanik seperti motor *servo*, pemisahan sumber listrik searah ini dilakukan untuk menghindari interferensi yang mungkin terjadi diakibatkan oleh induksi dari motor *servo*. Untuk memberikan sumber energi secara terpisah maka dilakukan pembagian distribusi sumber listrik arus searah dari baterai. Pemisahan sumber listrik arus searah dari baterai ini dilakukan untuk mengantisipasi gangguan dari motor *servo* pada sinyal kendali dari modul ESP32 maupun dari modul sensor. Modul *DC to DC Converter 1* digunakan untuk menjaga kestabilan sumber listrik arus searah dari baterai yang akan didistribusikan pada modul sensor *inertial measurement unit (IMU)* dan modul ESP32 DevKitC V.4. Modul *DC to DC Converter 2* berguna untuk mendistribusikan sumber listrik arus searah pada 18 unit motor *servo*.

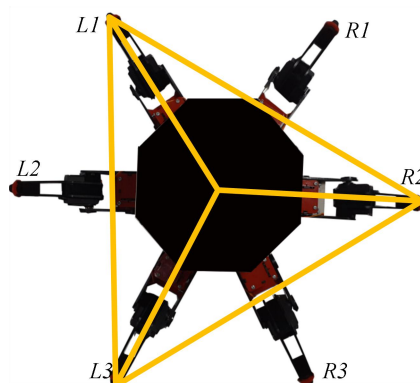
Berdasarkan diagram blok elektronik untuk robot *hexapod* maka dibutuhkan beberapa modul elektronik. Kebutuhan modul elektronik untuk robot *hexapod* dapat dilihat pada tabel di bawah ini.

■ Tabel 1. Daftar modul komponen elektronika yang dibutuhkan

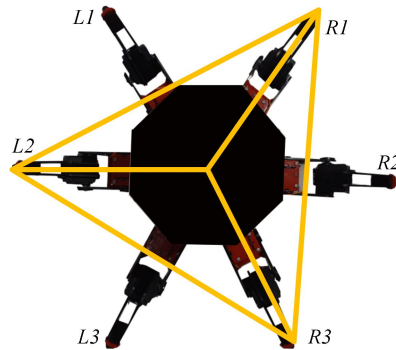
No.	Nama Komponen	Jumlah	Keterangan
1	Baterai <i>Lithium Polymer</i> 	1	Baterai <i>lithium polymer</i> berfungsi sebagai sumber listrik arus searah untuk seluruh modul elektronik yang digunakan. Tegangan yang dihasilkan baterai berkisar 7-12VDC.
2	DC to DC Converter 	2	Berfungsi untuk membatasi sumber tegangan yang akan digunakan oleh modul elektronik. Modul <i>DC to DC Converter</i> dapat menjaga kestabilan tegangan saat sumber listrik dari baterai berubah.
3	ESP32 DevKitC V.4 	1	Modul ESP32 berfungsi sebagai modul sistem kendali utama yang akan mengatur membaca input dari sensor serta dapat mengatur sistem elektronik dan komunikasi <i>wireless</i> dengan <i>smartphone</i> .
4	Motor <i>Servo</i> 	18	Motor <i>servo</i> berguna sebagai aktuator yang menggerakkan robot <i>hexapod</i> sesuai dengan perintah yang diterima dari perangkat sistem kendali. Motor <i>servo</i> yang digunakan mampu bergerak dengan sudut putaran sebesar 180°, dan untuk menggerakkannya dibutuhkan sumber listrik sekitar 5VDC.
5	Sensor IMU 	1	Sensor <i>inertia measurement unit (IMU)</i> 6 DOF, dimana 3 DOF untuk <i>gyroscope</i> dan 3 DOF untuk <i>Accelerometer</i> . Untuk komunikasi dengan mikrokontroler dapat menggunakan komunikasi serial <i>I2C</i> .
6	<i>Smartphone</i> 	1	<i>Smartphone</i> berguna untuk mengatur arah gerakan atau <i>remote control</i> gerakan dari robot <i>hexapod</i> . <i>Smartphone</i> yang digunakan harus mampu berkomunikasi secara <i>wireless</i> baik dengan menggunakan <i>bluetooth</i> ataupun <i>wifi</i> .

Robot *hexapod* terdiri dari 6 kaki yang dapat dikendalikan dengan menggunakan motor *servo*, jumlah motor *servo* yang digunakan bergantung dengan jumlah sendi yang diinginkan pada setiap kaki robot. Pada umumnya digunakan 3 motor *servo* pada setiap kaki robot, sehingga dibutuhkan 18 motor *servo* untuk seluruh komponen penggerakannya. Pada penelitian ini akan digunakan 3 motor *servo* untuk setiap kaki robot, dan jenis motor *servo* yang digunakan dapat menghasilkan sudut maksimal sebesar 180°.

Untuk mengatur gerakan robot *hexapod* dapat digunakan pola langkah *tripod gait*, dengan menggunakan pola *tripod gait* maka robot dapat berjalan dengan kombinasi gerakan awal kaki L1, R2, dan L3 diangkat secara bersamaan dengan sudut dan kecepatan tertentu. Gambar di bawah menunjukkan kaki robot yang digunakan saat langkah awal.

■ Gambar 6. Pola gerakan langkah awal robot *hexapod*

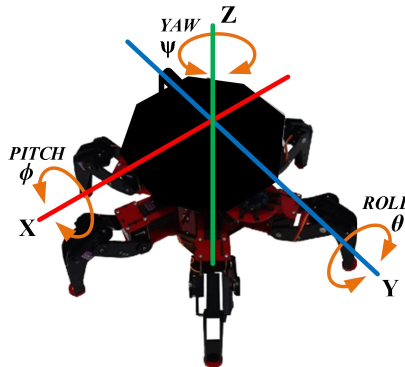
Setelah posisi awal pergerakan dilakukan dan telah mencapai pada posisi yang diinginkan maka selanjutnya mengangkat kaki R1, L2 dan R3. Pola pergerakan ini dilakukan terus menerus untuk menggerakkan robot. Pola langkah kedua dapat dilihat pada gambar di bawah ini.



■ **Gambar 7.** Pola gerakan langkah kedua robot *hexapod*

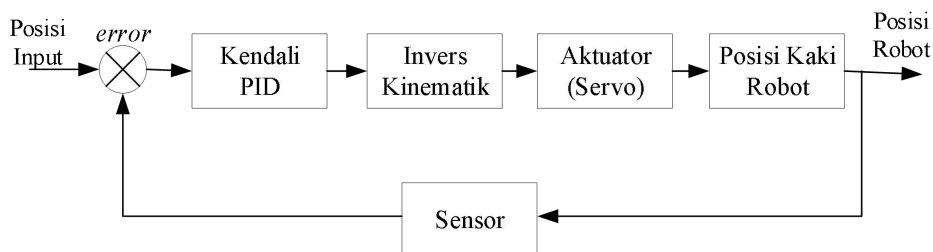
Pola langkah ini dapat diterapkan untuk gerakan pada arah maju, mundur, belok kiri, dan belok kanan. Hal yang membedakan gerakannya adalah dengan cara mengatur sudut dari motor *servo* pada setiap sendi dari kaki robot. Gerakan untuk maju dan mundur berlawanan arah, begitu juga untuk gerakan belok kiri dan kanan berlawanan arah.

Robot *hexapod* harus mampu bergerak dengan stabil, untuk menjaga kestabilan robot *hexapod* maka dilengkapi dengan sensor *inertial measurement unit (IMU)* yang memiliki fitur 6 *degree of freedom (6 DOF)*. Pada sensor *IMU* 6 derajat kebebasan dibagi menjadi 2 bagian yaitu *accelerometer 3 DOF*, *gyroscope 3 DOF*. Seluruh data sensor yang dihasilkan akan dibaca oleh perangkat sistem kendali untuk menjaga keseimbangan robot saat berjalan.



■ **Gambar 8.** Penerapan sensor IMU pada robot

Perancangan metoda gerak dilakukan agar gerak robot dapat bergerak berbagai arah. Metoda yang digunakan untuk pergerakan kaki robot adalah metoda *invers kinematik*, dengan metoda ini akan diperoleh posisi robot yang diinginkan sesuai dengan pergerakan sudut dari motor *servo*. Untuk mendukung metode *invers kinematik* maka digunakan sistem kendali *Proportional Integral Derivative (PID)*.



■ **Gambar 9.** Diagram blok sistem kendali robot *hexapod*

Perancangan *PID* dikembangkan berdasarkan persamaan dasar kendali *PID* dimana posisi input yang diberikan terhadap posisi output dengan menggunakan sensor sebagai umpan balik untuk memperbaiki *error* yang terjadi. Secara matematis kendali *PID* di deskripsikan dengan persamaan (1).

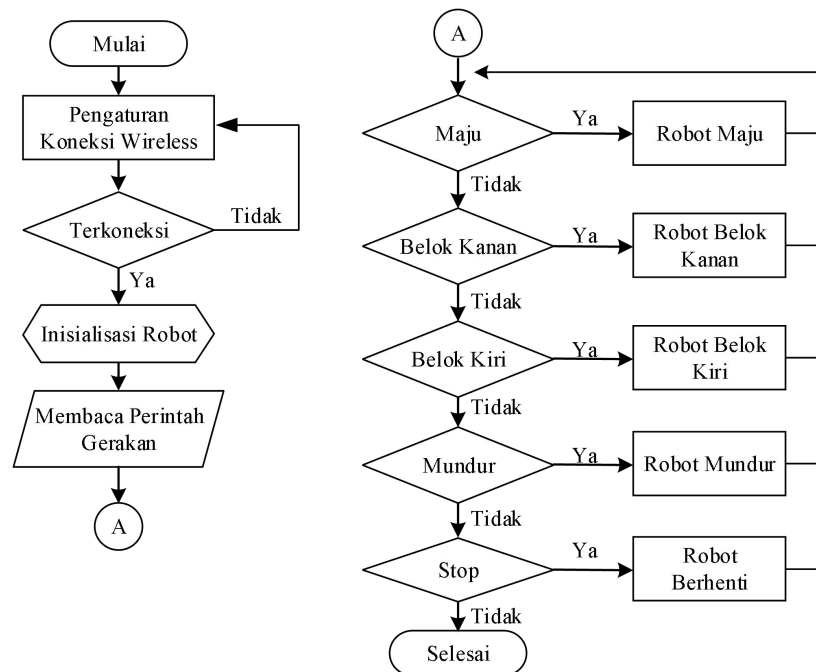
$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \dots\dots\dots (1)$$

Posisi input didapatkan dari data yang diberikan melalui *smartphone*, perintah gerakan untuk robot kemudian diterima oleh modul ESP32, setelah data diterima maka modul ESP32 akan mengolahnya dengan menerapkan konsep kendali *PID* dan kemudian diolah menggunakan mode *invers* kinematik, kemudian memberikan instruksi motor *servo* mana saja yang akan bergerak berdasarkan perintah yang didapatkan. Sensor *IMU* akan menjadi umpan balik (*feedback*) untuk menjaga kestabilan robot saat berjalan. Nilai dari sensor akan dijadikan nilai masukan dan kemudian dibandingkan dengan nilai *setpoint* yang telah ditentukan sehingga dapat diolah lebih lanjut oleh kendali *PID* untuk menentukan nilai sudut *servo* pada model *invers* kinematik.

PID sebagai sistem kendali gerakan robot dapat digunakan untuk memberikan masukan nilai *yaw* sehingga robot dapat berjalan lurus. Nilai *pitch* dan *roll* dapat digunakan untuk menyeimbangkan posisi robot sesuai dengan nilai *setpoint* yang diberikan. Nilai *yaw*, *pitch*, dan *roll* dapat diambil dari sensor *IMU*.

Perancangan Perangkat Lunak

Untuk perangkat lunak ada dua jenis program yang harus dibuat, yaitu program untuk ditanamkan pada modul ESP32 dan program aplikasi yang akan dijalankan melalui *smartphone*. Langkah awal untuk perancangan perangkat lunak adalah membuat *flowchart* untuk pergerakan robot yang akan ditanamkan pada modul ESP32. Beberapa hal yang perlu diperhatikan adalah konsep sistem *remote control* yang berasal dari *smartphone*. Blok diagram untuk *flowchart* program dapat dilihat pada gambar berikut.



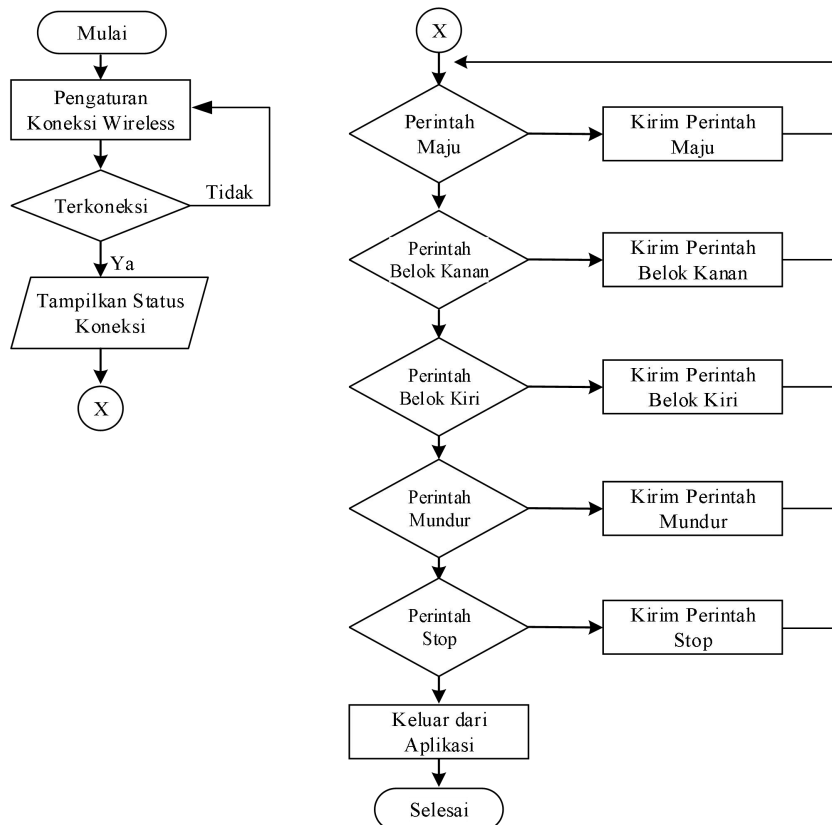
■ **Gambar 10.** *Flowchart* program untuk sistem kendali robot *hexapod* pada modul ESP32

Hal pertama yang harus dilakukan adalah membuat koneksi *wireless* antara *smartphone* dan robot *hexapod* yang akan dikendalikan gerakannya. Jenis koneksi yang dapat dilakukan adalah koneksi melalui *bluetooth* atau *wifi*, hanya salah satu jenis koneksi saja yang dapat dilakukan, tidak dapat dilakukan keduanya secara bersamaan. Setelah sistem komunikasi *wireless* selesai dilakukan maka tahap selanjutnya adalah inisialisasi posisi awal robot. Robot diatur sedemikian rupa sehingga didapat posisi awal yang ideal. Proses inisialisasi ini bergantung dengan ukuran dimensi robot dan panjang kaki robot. Posisi awal robot sebaiknya berada diantara posisi tertinggi dan posisi terendah badan robot. Setelah dilakukan inisialisasi posisi awal robot maka selanjutnya membaca intruksi atau perintah maju, mundur, belok kiri, belok kanan, dan stop yang berasal dari *smartphone*. Saat menerima perintah maju maka robot harus bergerak maju dengan pola langkah *tripod gait*, begitu juga dengan perintah yang lainnya, robot akan melakukan pergerakan sesuai dengan perintah yang diterima.

Setelah selesai membuat program yang ditanamkan pada modul ESP32 maka langkah selanjutnya adalah membuat aplikasi *remote control* robot yang akan dijalankan melalui *smartphone*. Pada *smartphone* hal pertama kali yang dilakukan adalah melakukan koneksi *wireless* antara

smartphone dan robot *hexapod*. Ada 2 jenis komunikasi yang dapat dilakukan yaitu menggunakan *bluetooth* atau menggunakan *wifi*, hanya salah satu jenis komunikasi saja yang dapat dipilih pada satu koneksi, tidak boleh melakukan 2 koneksi secara bersamaan.

Setelah koneksi *wireless* dipilih, maka akan ditampilkan status koneksinya apakah menggunakan *bluetooth* atau menggunakan *wifi*. Status terkoneksi dan tidak terkoneksi juga akan ditampilkan pada aplikasi *remote control* robot *hexapod*. Bila di klik tombol gerakan robot *hexapod* yang terdapat pada aplikasi *remote control* robot *hexapod* maka secara langsung akan memberikan perintah pada robot untuk melakukan gerakan sesuai dengan perintah yang diberikan.



■ **Gambar 11.** Flowchart program untuk sistem kendali pada *smartphone*

Perintah untuk pergerakan robot ada 5 yaitu perintah maju, belok kiri, belok kanan, mundur, dan berhenti. Tampilan dari aplikasi sistem kendali pergerakan robot *hexapod* dapat dilihat pada gambar berikut.



■ **Gambar 12.** Tampilan aplikasi kendali gerakan robot pada *smartphone*

Pada tampilan aplikasi robot *hexapod* terdapat tombol *radio button* untuk memilih jenis koneksi yang akan dilakukan antara *smartphone* dan robot *hexapod*, setelah terpilih jenis koneksi yang akan dilakukan maka dapat menekan tombol koneksi sehingga komunikasi dapat dilakukan. Sebagai indikator dari status koneksi maka ditampilkan informasi jenis koneksi dan status koneksi apakah terhubung atau tidak. Tombol terakhir yang disediakan pada aplikasi *smartphone* adalah tombol keluar, ketika tombol keluar dilakukan maka akan secara otomatis menutup aplikasi tersebut dan sekaligus memutuskan koneksi wireless yang telah dilakukan antara *smartphone* dan robot *hexapod*.

HASIL DAN PEMBAHASAN

Ujicoba robot *hexapod* dilakukan pada bidang datar dengan ketinggian tertentu untuk mengetahui kestabilan dan gerakan robot. Pada saat robot pertama kali dijalankan maka robot harus bergerak pada titik referensi. Tinggi referensi inilah yang menjadi patokan gerakan robot *hexapod*. Ujicoba robot *hexapod* dilakukan dengan beberapa tahapan diantara adalah menentukan sudut dari output motor *servo*, gerakan mekanik dari kaki robot, dan pembacaan sensor *IMU*. Pada kondisi awal robot *hexapod* bergerak maka robot harus berdiri pada ketinggian tertentu dan menjadi tinggi referensi, selanjutnya digunakan sensor *IMU* untuk membuat badan robot sejajar pada bidang permukaan. Posisi sensor *IMU* dapat digunakan sebagai referensi untuk membuat badan robot menjadi rata.

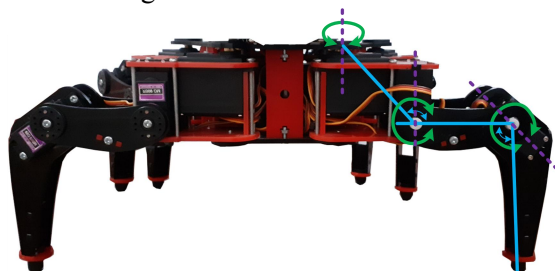
Titik referensi pada robot ditentukan berdasarkan nilai letak badan bagian bawah robot. Setelah ditentukan titik referensi maka pada robot *hexapod* harus dibatasi jarak terendah dan tertinggi badan robot terhadap bidang permukaan yang menjadi pijakan robot. Nilai terendah dari bagian bawah robot terhadap bidang permukaan adalah 1cm, dan nilai tertingginya adalah 10 cm. Nilai hasil pengujian sudut sensor *IMU* pada robot *hexapod* pada jarak terendah dan titik referensi serta jarak terjauh dapat dilihat pada tabel di bawah. Sudut maksimum pada setiap sudut adalah 90° .

■ **Tabel 2.** Hasil pengujian sensor *IMU*

Sudut	Jarak Terendah	Error pada Jarak Terendah (%)	Jarak Referensi	Error pada Jarak Referensi (%)	Jarak Terjauh	Error pada Jarak terjauh (%)
<i>Pitch</i> (ϕ)	0,50	0,56	0,61	0,68	0,70	0,78
<i>Roll</i> (θ)	0,81	0,90	0,90	1,00	1,00	1,11
<i>Yaw</i> (ψ)	0,61	0,68	0,71	0,79	0,80	0,89

Berdasarkan hasil pengujian sudut pada *pitch*, *roll*, dan *yaw* pada robot *hexapod* diketahui bahwa semakin tinggi jarak badan robot *hexapod* dari permukaan maka semakin besar *error* yang dihasilkan. *Error* ini terjadi karena beberapa hal, diantaranya adalah penempatan sensor *IMU* tidak rata dan konstruksi mekanik robot tidak terlalu presisi, tetapi dengan nilai *error* yang kecil bila dibandingkan sudut gerakan robot. Nilai *error* rata-rata pada *Pitch* (ϕ) adalah 0,67%, nilai *error* rata-rata pada *Roll* (θ) adalah 1,01%, dan nilai *error* rata-rata pada *Yaw* (ψ) adalah 0,79%.

Pada kondisi awal robot *hexapod* bergerak maka robot harus berdiri pada ketinggian tertentu berdasarkan jarak referensi yang telah ditentukan. Nilai ketinggian dari bagian bawah badan robot akan menentukan sudut motor *servo* yang dipasang pada setiap kaki robot. Penentuan sudut motor *servo* pada titik referensi, jarak terendah dan jarak tertinggi dari badan robot bagian bawah harus dilakukan karena nilai sudut ini yang menjadi batasan untuk pergerakan sudut robot. Motor *servo* yang digunakan mampu menghasilkan sudut maksimal sebesar 180° , tetapi sudut 0° motor *servo* berada ditengah, sehingga rentang maksimum gerakan sudut motor *servo* dari -90° sampai 90° .



■ **Gambar 13.** Gerakan sudut motor *servo* pada kaki robot

Robot *hexapod* memiliki 6 kaki dan setiap kaki memiliki 3 motor *servo* sehingga total motor *servo* yang digunakan berjumlah 18 motor *servo*. Tabel di bawah merupakan nilai sudut motor *servo*

pada saat badan bagian bawah robot berada pada jarak terjauh atau tertinggi. Jarak tertinggi dari badan bawah robot *hexapod* terhadap permukaan dirancang dengan jarak 10 cm. Pada jarak tertinggi robot *hexapod*, setiap kaki robot telah ditentukan besar sudutnya, besar sudut yang diinginkan pada *servo* motor 1 (*joint 1*) adalah 90° , besar sudut *servo* motor 2 (*joint 2*) adalah 45° , dan besar sudut *servo* motor 3 (*joint 3*) adalah 65° . Tabel pengujian jarak terjauh badan robot dapat dilihat pada tabel di bawah.

■ **Tabel 3.** Hasil pengujian motor *servo* pada jarak tertinggi

Kaki Robot	<i>Servo/Joint</i>	Sudut <i>Servo</i>	<i>Link</i>	Sudut <i>Link</i>	Error Sudut <i>Servo</i> (%)
Kaki 1 (L1)	<i>Servo 1</i>	89	1	90	1,11
	<i>Servo 2</i>	-45	2	-45	0,00
	<i>Servo 3</i>	-63	3	21	3,08
Kaki 2 (L2)	<i>Servo 1</i>	90	1	88	0,00
	<i>Servo 2</i>	-44	2	-44	2,22
	<i>Servo 3</i>	-66	3	23	1,54
Kaki 3 (L3)	<i>Servo 1</i>	90	1	89	0,00
	<i>Servo 2</i>	-44	2	-45	2,22
	<i>Servo 2</i>	-64	3	20	1,54
Kaki 4 (R1)	<i>Servo 1</i>	89	1	90	1,11
	<i>Servo 2</i>	-44	2	-44	2,22
	<i>Servo 3</i>	-66	3	21	1,54
Kaki 5 (R2)	<i>Servo 1</i>	89	1	90	1,11
	<i>Servo 2</i>	-45	2	-46	0,00
	<i>Servo 3</i>	-64	3	20	1,54
Kaki 6 (R3)	<i>Servo 1</i>	89	1	89	1,11
	<i>Servo 2</i>	-45	2	-45	0,11
	<i>Servo 3</i>	-64	3	20	1,54

Hasil pengujian sudut pada setiap motor *servo* di titik referensi dapat diketahui bahwa sudut yang dihasilkan terdapat beberapa perbedaan pada hampir setiap sudut *joint* dan *link* pada kaki robot. Nilai *error* untuk sudut maksimum pada motor *servo* atau *joint* dari kaki robot adalah 3,08%, tetapi nilai perbedaannya masih dalam tahap toleransi. Perbedaan sudut robot ini juga sangat berpengaruh pada kontruksi robot dan penempatan atau posisi dari motor *servo*.

Posisi pada titik terendah perlu dilakukan pengujian juga, jarak terendah badan bagian bawah robot *hexapod* adalah 1 cm. Pada jarak referensi badan robot robot *hexapod*, setiap kaki robot telah ditentukan besar sudutnya, besar sudut yang diinginkan pada *servo* motor 1 (*joint 1*) adalah 90° , besar sudut *servo* motor 2 (*joint 2*) adalah -65° , dan besar sudut *servo* motor 3 (*joint 3*) adalah -20° . Hasil pengujian untuk titik terendah dapat dilihat pada tabel di bawah.

■ **Tabel 4.** Hasil pengujian *servo* pada jarak terendah

Kaki Robot	<i>Servo/ Joint</i>	Sudut <i>Servo</i>	<i>Link</i>	Sudut <i>Link</i>	Error Sudut <i>Servo</i> (%)
Kaki 1 (L1)	<i>Servo 1</i>	88	1	88	2,22
	<i>Servo 2</i>	-64	2	-64	1,54
	<i>Servo 3</i>	-20	3	25	0,00
Kaki 2 (L2)	<i>Servo 1</i>	89	1	89	1,11
	<i>Servo 2</i>	-65	2	-65	0,00
	<i>Servo 3</i>	-21	3	25	5%
Kaki 3 (L3)	<i>Servo 1</i>	89	1	89	1,11
	<i>Servo 2</i>	-64	2	-64	1,54
	<i>Servo 3</i>	-21	3	24	5,00
Kaki 4 (R1)	<i>Servo 1</i>	90	1	89	1,11
	<i>Servo 2</i>	-65	2	-65	0,00
	<i>Servo 3</i>	-21	3	24	5,00
Kaki 5 (R2)	<i>Servo 1</i>	89	1	90	1,11
	<i>Servo 2</i>	-65	2	-64	0,00
	<i>Servo 3</i>	-20	3	25	0,00
Kaki 6	<i>Servo 1</i>	89	1	90	1,11

Kaki Robot	Servo/ Joint	Sudut Servo	Link	Sudut Link	Error Sudut Servo (%)
(R3)	Servo 2	-65	2	-65	0,00
	Servo 3	-20	3	25	0,00

Dari hasil pengujian sudut pada setiap motor *servo* pada titik terendah dapat diketahui bahwa sudut *joint* dan sudut *link* yang dihasilkan terdapat sedikit perbedaan pada hampir setiap *joint* dan *link* dari kaki robot. Nilai *error* ini sangat bervariasi tergantung dengan konstruksi atau penempatan motor *servo*. Nilai *error* maksimum yang diperoleh berdasarkan hasil pengujian adalah sekitar 5%.

Pengujian *PID* dilakukan untuk mendapatkan nilai yang terbaik untuk sumbu *yaw*, *pitch* dan *roll*. Formula untuk *tuning* empiris dari *PID* yang digunakan adalah Ziegler-Nichols. Pengujian sistem kendali yang dilakukan hanya 3 saja yaitu kendali P, PI, dan PID. Hasil pengujian *PID* dengan *Kp* memiliki nilai $k_1=0,6$, $k_2=1,1$, $k_3=2,1$, dan pada *Ki* diberi nilai $k_1=0,5$, $k_2=1$, $k_3=1,2$, dan pada *Kd* diberi nilai $K_1=0,25$, dan $k_2=0,45$, dan $k_3=5$, dapat dilihat pada tabel berikut.

■ **Tabel 5.** Hasil posisi *Yaw*, *Pitch* dan *Roll* dengan kendali *PID*

No	Kendali	<i>Yaw</i>				<i>Pitch</i>				<i>Roll</i>			
		<i>Rise Time</i> (s)	<i>Settling Time</i> (s)	<i>Over shoot</i> (°)	<i>Steady State Error</i> (°)	<i>Rise Time</i> (s)	<i>Settling Time</i> (s)	<i>Over shoot</i> (°)	<i>Steady State Error</i> (°)	<i>Rise Time</i> (s)	<i>Settling Time</i> (s)	<i>Over shoot</i> (°)	<i>Steady State Error</i> (°)
1	P	0,45	4,65	6,10	2,50	-	-	-	-	-	-	-	-
2	PI	0,50	5,75	18,5	0,50	4,50	4,60	0,25	0,25	1,90	1,90	0,20	0,15
3	PID	0,35	4,50	9,5	0,45	1,50	1,25	0	0	1,50	1,50	0	0

Berdasarkan hasil pengujian sistem kendali dapat dilihat pada sumbu *yaw*, *pitch*, dan *roll*, bahwa sistem kendali *PID* lebih stabil dan respon lebih cepat untuk mengatur kestabilan robot *hexapod* dibandingkan dengan hanya pengaturan P atau PI saja. Saat hanya menggunakan kendali *Proporsional* (P) saja, pada sumbu *Pitch* dan *Roll* tidak mencapai nilai *setpoint*. Dengan menggunakan kendali *PID* kecepatan respon akan bertambah cepat dan kestabilan sistem secara keseluruhan menjadi lebih baik karena kondisi terakhir lebih mendekati nilai yang diinginkan. Sebagai tanda robot *hexapod* bergerak stabil, pada saat berjalan atau bergerak, posisi badan robot seimbang dan tidak condong pada satu arah tertentu saja.

Pengujian kecepatan robot terhadap perintah yang diberikan maka dilakukan pengujian gerakan robot arah maju dan mundur dengan mengacu pada garis lurus, dengan menguji maju mundur robot *hexapod* pada garis lurus maka dapat diketahui kecepatan maksimum robot saat berjalan dan dapat seberapa besar *error* yang terjadi pada robot yang telah dibuat. Pola langkah yang digunakan pada robot *hexapod* yang diuji adalah *Tripod Gait*. Pengujian gerak maju atau gerak lurus robot dilakukan pada lintasan garis lurus sebagai acuan sepanjang 5 meter. Tabel di bawah memperlihatkan hasil pengujian pergerakan dan kecepatan robot.

■ **Tabel 6.** Hasil pengujian gerakan dan kecepatan robot *hexapod*

No.	Jenis Pengujian	Hasil Pengujian
A Arah Maju (Jarak Tempuh 5 Meter)		
1	Kepatan robot arah maju	24,90 detik
2	Kecenderungan pergerakan	Arah kanan
3	Pergeseran gerakan terhadap titik tujuan	35 cm
4	<i>Error</i> arah pergerakan	7 %
B Arah Mundur (Jarak Tempuh 5 Meter)		
1	Kepatan robot arah mundur	25,10 detik
2	Kecenderungan pergerakan	Arah kiri
3	Pergeseran gerakan terhadap titik tujuan	45 cm
4	<i>Error</i> arah pergerakan	9 %
C Belok Kanan		
1	Kecepatan putaran robot saat belok kanan (searah jarum jam) dengan sudut putar 90°.	4,10 detik
D Belok Kiri		
1	Kecepatan putaran robot saat belok kiri	4,20 detik

No.	Jenis Pengujian	Hasil Pengujian
A	Arah Maju (Jarak Tempuh 5 Meter) (berwalanan arah jarum jam) dengan sudut putar 90°.	

Pengujian terakhir yang dilakukan adalah pengujian koneksi *wireless* dan pengujian pergerakan dari robot sesuai dengan perintah yang diberikan. Untuk model pengujian dilakukan dalam bentuk apakah robot dapat berkomunikasi dan berinteraksi dengan *smartphone*. Tabel di bawah memperlihatkan hasil pengujian dari pergerakan robot berdasarkan perintah yang diberikan melalui *smartphone*.

■ Tabel 7. Hasil pengujian seluruh sistem

No.	Jenis Pengujian	Hasil Pengujian
1	Pengujian koneksi <i>wifi</i>	Berhasil
2	Pengujian status koneksi <i>wifi</i>	Berhasil
3	Jarak Pengujian Koneksi <i>Wifi</i> 20 meter	Berhasil
4	Pengujian koneksi <i>bluetooth</i>	Berhasil
5	Pengujian status koneksi Bluetooth	Berhasil
6	Jarak Pengujian koneksi bluetooth 19 meter	Berhasil
7	Jarak pengujian koneksi bluetooth 20 meter	Tidak Berhasil
8	Pengujian status koneksi <i>bluetooth</i>	Berhasil
9	Pengujian perintah Maju	Berhasil
10	Pengujian perintah Mundur	Berhasil
11	Pengujian perintah Belok Kiri	Berhasil
12	Pengujian perintah Belok Kanan	Berhasil
13	Pengujian perintah Stop	Berhasil
14	Pengujian tombol keluar dari aplikasi	Berhasil

Pada tabel tambah bahwa pengujian keseluruhan sistem berhasil dilakukan dengan jarak maksimum pengujian antara *smartphone* dengan robot hexapod sekitar 20 meter. Pengujian keseluruhan tombol pada *smartphone* untuk mengendalikan arah gerakan robot berhasil dilakukan.

KESIMPULAN

Dari hasil penelitian robot *hexapod* yang dikendalikan melalui *smartphone*, dapat disimpulkan beberapa hal, antara lain:

1. Konstruksi perangkat elektromekanik pada robot *hexapod* harus didesain agar robot dapat bergerak dan melangkah dengan baik. Konstruksi mekanik robot akan berpengaruh pada gerakan robot *hexapod*.
2. Robot *hexapod* dibuat dengan menggunakan 18 motor *servo* sebagai penggerakannya, setiap kaki robot dilengkapi dengan 3 motor *servo* sebagai sumber penggerakannya sehingga menghasilkan 3 derajat kebebasan pada setiap kaki robot.
3. Untuk menambah kestabilan badan robot sehingga robot dapat bergerak seimbang maka pada robot *hexapod* dilengkapi dengan sensor *inertial measurement unit*. Dengan menggunakan sensor *inertial measurement unit* maka robot dapat bergerak dengan lebih cepat karena keseimbangannya terjaga dengan baik.
4. Untuk menggerakkan robot *servo* maka diterapkan sistem kendali *Proportional Integral Derivative (PID)* sehingga robot *hexapod* dapat bergerak stabil dengan menggunakan pola langkah *tripod gait*. Kestabilan robot *hexapod* berpengaruh pada kecepatan gerakan robot saat berjalan. Kecepatan maksimum pada saat robot berjalan maju adalah 24,90 detik dengan *error* simpangan sebesar 7% atau 35 cm terhadap titik tujuan sejauh 5 meter.
5. Robot *hexapod* dapat dikendalikan melalui *smartphone* dengan menggunakan komunikasi *wireless* seperti *bluetooth* ataupun *wifi*, tetapi jaraknya masih terbatas sesuai dengan jangkauannya dari *bluetooth* ataupun *wifi*. Jarak maksimum untuk *remote control* sekitar 20 meter, diluar jarak tersebut robot sudah tidak dapat dikendalikan lagi karena sudah diluar jangkauan.
6. Perintah Maju, Mundur, Kiri, Kanan, dan Stop cukup diberikan 1 kali saja. Bila telah diberi perintah 1 kali maka robot akan terus melaksanakan perintah tersebut sampai ada perintah yang lainnya. Misalnya diberi perintah maju 1 kali melalui *smartphone* maka robot *hexapod* akan terus maju sampai ada perintah berhenti atau perintah yang lainnya. Robot dapat berjalan dengan maju dan mundur dengan kecepatan rata sekitar 5 meter / 25 detik atau sekitar 20 cm/detik.

7. Sebagai tambahan untuk pengembangan robot ini agar robot dapat bergerak dengan baik dan pada saat berjalan dan tidak menabrak benda atau objek yang menghalanginya maka dapat digunakan sensor ultrasonik atau dapat juga dilengkapi dengan kamera *wireless*. Nilai pembacaan jarak dari sensor ultrasonik dapat ditampilkan atau hasil dari kamera *wireless* dapat ditampilkan pada layar *smartphone*.

DAFTAR PUSTAKA

- [1] A. A. Syakur, R. Maulana, dan E. Setiawan, "Perancangan dan Implementasi sistem Pola Berjalan pada Robot Hexapod Menggunakan Inverse Kinematic," Jurnal Pengembangan Teknologi Informasi dan Komputer, vol.4, no.6, pp.1875-1881, Juni 2020.
- [2] A. I. A. Raham, R. Maulana, dan H. Fitriyah, "Kinematika Robot dengan Sistem Kendali Proporsional Intergral dan Derivative menggunakan Sensor IMU pada Robot Hexapod," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol.6, no.7, pp.3316-3323, Juli 2022.
- [3] M. Asrofi, S. Sumardi, dan B. Setiyono, "Stabilisasi Robot Berkaki 6 (Hexapod) pada Bidang Miring Menggunakan 9 DOF IMU Berbasis Invers Kinematik," Jurnal Transient, vol.4, no.1, pp.97-105, Maret 2015.
- [4] A. Najmurokhman, Kusnandar, I. Irfansyah, dan A. Dalami, "Rancang Bangun Auto Balancing Robot Menggunakan Metode Kendali PID," Jurnal Telka, vol.5, no.1, pp 15-23, Mei 2019.
- [5] Rendyansyah, H. Ubaya, A. Faydinar, dan Y. Ramdhan, "Kendali Gerak Robot Hexapod Menggunakan Remote Control Berbasis Android," Jurnal J-Innovation, vol.7, no.1, pp.7-14, Juni 2018.
- [6] G. Firasanto, "Pengendalian Robot Hexapod Berbasis Internet of Things (IoT)," Journal of Electrical Power, Instrumentation and Control, Vol.4., No.1, pp.1-9, Juni 2021.
- [7] M. F. Abdi, dan Fitriyadi, "Model Kendali Berbasis Perilaku pada Robot Berkaki Hexapod 3 DOF," Jurnal Progresif, col.12, no.2, pp.1387-1524, Agustus 2016.
- [8] J. Andika, dan K. S. Salamah, "Analisis Kinematika pada Robot Hexapod," Jurnal Teknologi Elektro, Vol.9, No.2, Mei 2018.
- [9] S. Muslimin, K. Salahuddin, dan E. Prihatini, "Implementasi Inverse Kinematics Terhadap Pola Gerak Hexapod Robot 2 DOF," Jurnal Dielektrika, Vol.4, No.2, pp.142-146, Agustus 2017.
- [10] A. Wajiansyah, Supriadi, A. F.O. Gaffar, dan A. B. W. Putra, "Modeling of 2-DOF Hexapod Leg Using Analytical Method," Journal of Robotics and Control, Vol.2. No.5, pp.435-440, September 2021.