

Pergerakan Jalan Stabil Robot *Hexapod* di Atas Medan yang Tidak Rata

Rudy¹, Lukas¹

ABSTRACT: *Over the years, natural disasters such as volcano eruption, landslide, as well as flood have occurred and caused victims. Immediate response is required to rescue them, however, detecting and rescuing victims accurately in natural disasters is not easy to be done. The dangerous area with a slippery and sloppy land, as well as the worry that there could be an aftershock disaster, makes it hard to be reached by the rescue team, to get into the area and detect victims. It takes so much time for them to detect and drive the ambulance to the desired location and bring the victims into the car. To solve those problems mentioned by using a robotic technology. The robotic technology can be used to rescue victims during natural disasters with design the prototype of a hexapod that implements inverse kinematics and gait algorithm. Hence, the robot can walk passing the uneven surface and keep being balanced. The hexapod robot is chosen because it has a good stability level when running. The hexapod robot is successfully tested in six different field conditions and is fully capable of maintaining stability in every test such as standing still on a moving surface, running on sloping surfaces, grass and sandy surfaces with a percent error of less than 10%.*

Keywords: *uneven surface, hexapod, stabile motion of robot*

ABSTRAK: Beberapa tahun terakhir, bencana alam seperti gunung meletus, tanah longsor, dan banjir sering terjadi dan menimbulkan korban jiwa. Upaya penyelamatan korban jiwa dalam bencana tersebut menjadi sulit dilakukan, karena daerah bencana sangat sulit dicapai oleh tim penyelamat. Permukaan tanah daerah bencana yang tidak rata dan potensi bencana susulan menjadi salah satu kendala dalam penyelamatan korban oleh tim penyelamat. Proses penyelamatan korban biasa dilakukan oleh tim penyelamat dengan membawa mobil ambulans ke lokasi bencana dan tim penyelamat menyusuri daerah bencana untuk menemukan korban. Masalah yang akan muncul adalah tim penyelamat saat mencari korban membahayakan diri saat berada di daerah bencana yang masih berpotensi mengalami bencana susulan, mobil ambulans untuk mengangkut korban tidak bisa menuju lokasi korban karena permukaan tanah daerah bencana yang tidak rata. bila korban berada jauh dari lokasi mobil ambulans, tim penyelamat akan membutuhkan waktu untuk membawa korban menuju mobil ambulans. Sehubungan dengan ini diusulkan suatu rancangan pemecahan masalah dengan menggunakan teknologi robot. Robot digunakan untuk melewati daerah yang memiliki permukaan yang tidak rata berupa sebuah prototype robot *hexapod* yang menerapkan inverse kinematic dan algoritma gait, sehingga robot dapat melewati daerah yang memiliki permukaan tidak rata serta menjaga kestabilan badan robot saat berjalan. Robot *hexapod* dipilih karena memiliki tingkat kestabilan yang baik saat berjalan. Robot hexapod berhasil diuji dalam enam kondisi bidang berbeda dan sepenuhnya mampu menjaga kestabilan di setiap pengujian seperti ketika berdiri diam di permukaan yang bergerak, berjalan di permukaan miring, permukaan rumput maupun berpasir dengan dengan persen kesalahan kurang dari 10%.

Kata kunci: permukaan tidak rata, hexapod, kestabilan badan robot

PENDAHULUAN

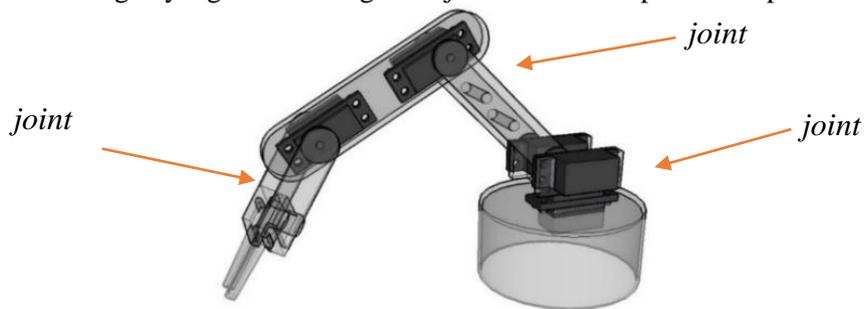
Beberapa tahun terakhir, bencana alam seperti gunung meletus, tanah longsor, dan banjir sering terjadi dan menimbulkan banyak korban jiwa. Upaya penyelamatan korban dalam bencana tersebut menjadi sulit dilakukan, karena daerah bencana sangat sulit dicapai oleh tim penyelamat. Permukaan tanah daerah bencana yang tidak rata dan potensi bencana susulan menjadi salah satu kendala dalam penyelamatan korban oleh tim penyelamat tersebut. Proses penyelamatan korban dilakukan oleh tim penyelamat dengan membawa mobil ambulans ke lokasi bencana. Lalu, Tim penyelamat menyusuri daerah bencana untuk menemukan korban. Masalah yang akan timbul adalah jaminan keselamatan dari tim saat berada di daerah bencana yang berpotensi mengalami bencana susulan, seperti kejadian pada 10 April 2017 yaitu tanah longsor di daerah Ponorogo, Satu mobil pengangkut anjing pelacak, satu mobil milik relawan, satu ekskavator, tiga alat penyemprot air, dan alat komunikasi tim SAR gabungan tertimbun material tanah longsor[10]. Mobil ambulans untuk mengangkut korban tidak bisa menuju lokasi karena permukaan tanah daerah bencana yang tidak rata (*uneven*), dan bila korban berada jauh dari lokasi mobil ambulans, maka tim penyelamat akan memakan waktu untuk membawa korban menuju mobil ambulans. Dalam Penelitian ini diusulkan suatu rancangan pemecahan masalah dengan menggunakan teknologi robot. Robot yang dirancang berupa robot berkaki enam (*hexapod*) yang mampu melewati daerah yang memiliki permukaan yang tidak rata serta menjaga kestabilan badan robot saat berjalan.

Istilah robot berasal dari bahasa Ceko Slowakia yaitu “*Robota*” yang berarti pekerja. Istilah ini dikemukakan oleh seorang pengarang sandiwara yang bernama Karel Čapek pada tahun 1920. Pengertian robot sangat beragam, sebagian orang membayangkan bahwa robot itu adalah sebuah mesin yang menyerupai manusia (*humanoid*) tetapi sesungguhnya robot tidak hanya berbentuk seperti manusia. Robot pada awalnya diciptakan untuk menggantikan kerja manusia untuk sesuatu yang berulang, membutuhkan ketepatan yang tinggi dan juga untuk menggantikan manusia bila harus berhubungan dengan daerah berbahaya[1]. Robot adalah sebuah alat mekanik yang dikendalikan oleh perangkat elektronik dan dapat bekerja dengan pengawasan manusia atau dengan menggunakan program yang telah didefinisikan terlebih dahulu (*kecerdasan buatan/artificial intelligent*)[4].

Salah satu jenis robot yang digunakan dalam dunia industri adalah robot tangan. Robot ini biasanya digunakan dalam industri otomotif untuk melakukan pekerjaan-pekerjaan seperti: mengecat, mengelas, mengebor, dan memasang baut pada tempat yang sulit dijangkau. Pada prinsipnya robot tangan ini berbentuk seperti tangan manusia. Robot ini memiliki persendian atau yang selanjutnya akan disebut dengan istilah *joint*. Jumlah *joint* pada robot ini disesuaikan dengan kebutuhan, tetapi pada umumnya adalah tiga *joint*. Robot yang memiliki satu *joint*

¹ Jurusan Teknik Elektro, Universitas Katolik Indonesia Atma Jaya

memiliki satu derajat kebebasan, robot yang memiliki dua joint memiliki dua derajat kebebasan dan begitu seterusnya. Contoh robot tangan yang memiliki tiga derajat kebebasan dapat dilihat pada Gambar 1



■ **Gambar 1** Robot tangan dengan tiga derajat kebebasan [3]

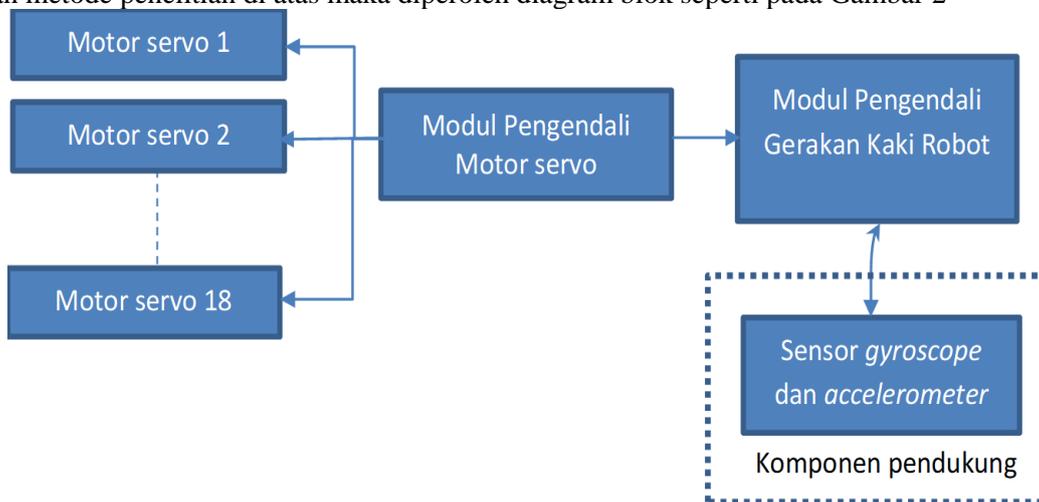
Robot tangan yang ada pada Gambar 1 dapat bergerak sesuai dengan titik koordinat yang ditentukan sebagai masukan (*input*) pada robot tersebut. Hal ini dapat terjadi karena robot tersebut menggunakan perhitungan *inverse kinematic* untuk menghitung besarnya sudut pada setiap *joint* yang ada, berdasarkan posisi badan robot dan koordinat yang menjadi masukan robot untuk pergerakan tangannya. Perhitungan *inverse kinematic* ini juga digunakan dalam pemrograman pergerakan robot berkaki enam (*hexapod*). Alasan peneliti menggunakan robot *hexapod*, karena robot secara statistik dapat lebih menjaga kestabilan dengan menggunakan 3 kaki atau lebih, maka robot *hexapod* mempunyai fleksibilitas dan keandalan yang tinggi. Jika ada kaki yang tidak berfungsi, maka ada kemungkinan robot tersebut masih bisa berjalan. Terlebih lagi, tidak semua kaki robot dibutuhkan untuk stabilitas, kaki lainnya dapat bergerak bebas untuk mencari tempat pijakan baru[2].

Tujuan penelitian

Penelitian ini bertujuan untuk merancang sebuah prototype robot hexapod dan menerapkan inverse kinematic dan algoritma gait, sehingga robot mampu melewati daerah yang memiliki permukaan tidak rata serta menjaga kestabilan badan robot saat berjalan.

Diagram blok

Berdasarkan metode penelitian di atas maka diperoleh diagram blok seperti pada Gambar 2



■ **Gambar 2** Diagram blok rancangan

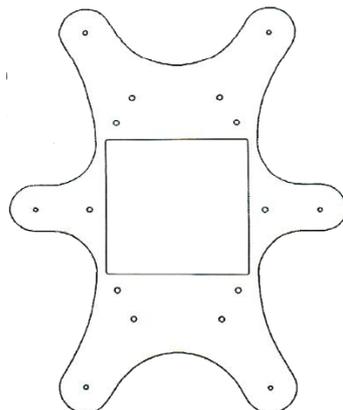
Realisasi sistem

Berdasarkan rancangan yang sudah dibuat maka dilakukan tahap realisasi, terdiri dari beberapa realisasi modul yaitu: konstruksi mekanik robot *hexapod*, modul pengendali motor servo, dan modul pengendali gerakan kaki robot sementara Modul sensor *gyroscope* dan *accelerometer* tidak dirancang karena sudah ada dalam bentuk produk jadi.

Konstruksi mekanik robot hexapod

Konstruksi mekanik dari robot *hexapod* ini terdiri dari beberapa bagian yaitu: *baseplate* badan, *coxa* atas, *coxa* bawah, *tibia* dan juga *femur*. Perancangan konstruksi robot ini menggunakan aplikasi *autodesk inventor* sebagai media untuk menggambar dan juga mensimulasikan gambar robot menjadi gambar tiga dimensi.

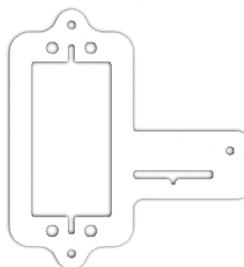
Konstruksi robot *hexapod* ini menggunakan bahan *acrylic* dengan ukuran ketebalan 5 mm untuk setiap bagian robot. Proses pemotongan *acrylic* menjadi bagian-bagian robot sesuai dengan gambar menggunakan proses *cutting laser*.



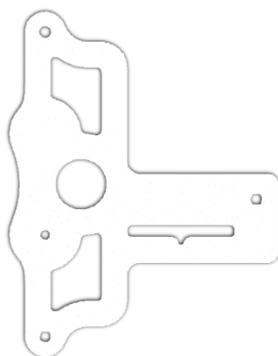
■ **Gambar 3** Bentuk *baseplate* badan skala 1: 2,3 cm



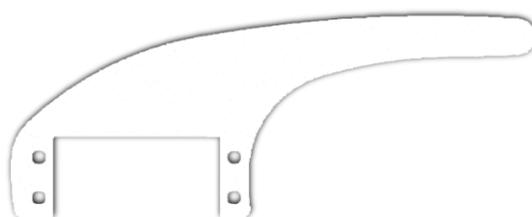
■ **Gambar 4** Bentuk komponen *femur* skala 1: 1 cm



■ **Gambar 5** Komponen *coxa* atas skala 1: 2 cm



■ **Gambar 6** Komponen *coxa* bawah skala 1: 2 cm



■ **Gambar 7** bentuk komponen *tibia* skala 1: 2 cm

Setiap Gambar yang dibuat menggunakan satuan milimeter. Gambar 3 sampai Gambar 7 merupakan gambaran bentuk komponen konstruksi tidak memiliki ukuran yang sama dengan robot yang dibuat.

Modul pengendali motor servo

Modul pengendali motor servo berfungsi untuk menghasilkan PWM yang akan digunakan untuk mengendalikan motor servo. Modul ini menerima masukan dari mikrokontroler atau komputer dengan

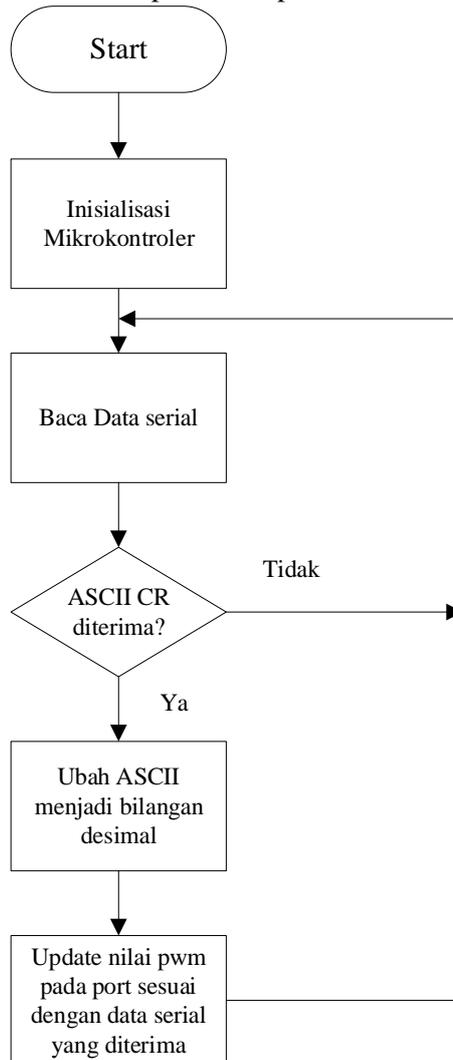
menggunakan komunikasi UART. Modul ini menggunakan komunikasi pada *baudrate* 38400. *Crystal* osilator yang digunakan pada mikrokontroler ATmega-168 adalah 14,75 MHz. Penggunaan *crystal* osilator dengan frekuensi tersebut bertujuan untuk mendapatkan *error-rate* sekecil mungkin pada transfer data menggunakan komunikasi UART. Untuk mendapatkan *baudrate* 38400, *register* UBRRn harus diisi dengan nilai tertentu. Nilai *register* UBRRn dapat dihitung dengan rumus:

$$UBRRn = \frac{f_{osc}}{16baudrate} - 1$$

Setelah dihitung dengan rumus di atas, maka nilai *register* UBRRn adalah 23,007161. *Register* UBRRn adalah *register* 16-bit sehingga nilai yang dapat ditulis di *register* tersebut adalah bilangan bulat. Maka nilai *register* tersebut adalah 23. Nilai *error-rate* pada konfigurasi di atas adalah:

$$errorrate = \frac{23.007161 - 23}{23} \times 100\% = 0.031\%$$

Diagram alir modul pengendali motor servo dapat dilihat pada Gambar 8.



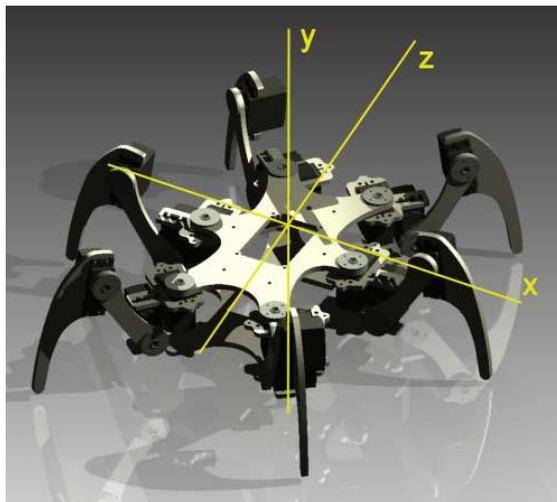
■ **Gambar 8** Diagram alir modul pengendali motor servo

Modul pengendali gerakan kaki robot

Modul ini berfungsi untuk menerima masukan dari sensor *gyroscope* dan *accelerometer* dan mengatur pergerakan kaki robot agar dapat bergerak secara *omnidirectional*. Dalam pembahasan realisasi modul ini akan dibagi menjadi bagian perangkat keras dan juga bagian perangkat lunak yang akan dibahas secara terpisah.

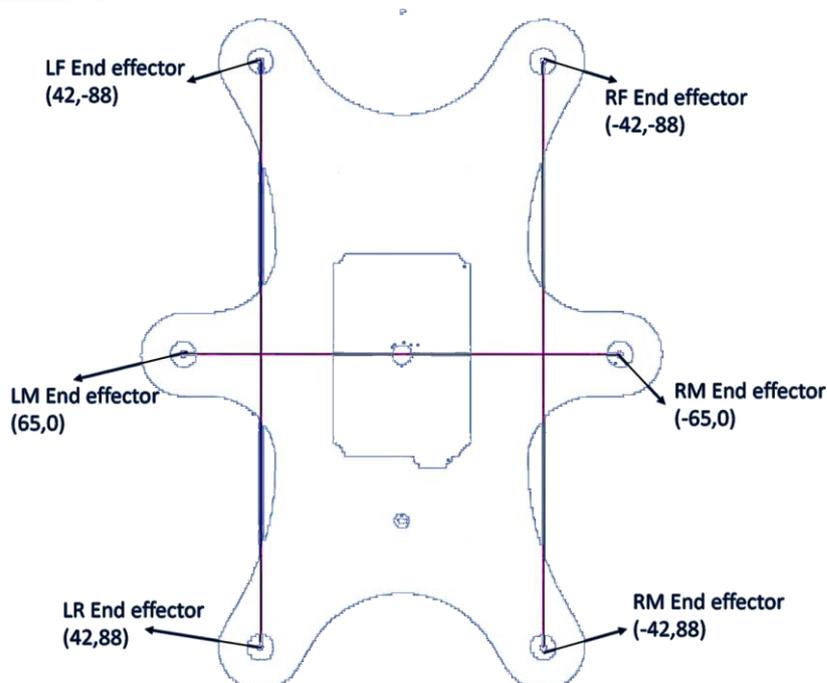
Realisasi program algoritma *gait*

Program algoritma *gait* bertujuan untuk menghasilkan nilai-nilai *end effector* pada setiap kaki sehingga robot dapat bergerak secara *omnidirectional*. Robot *hexapod* menempati ruang tiga dimensi yang memiliki sumbu-sumbu koordinat x, y, dan z. Arah dari sumbu-sumbu koordinat telah ditetapkan oleh peneliti dan dapat dilihat pada Gambar 9. Titik perpotongan ketiga sumbu koordinat terletak pada titik tengah badan robot.



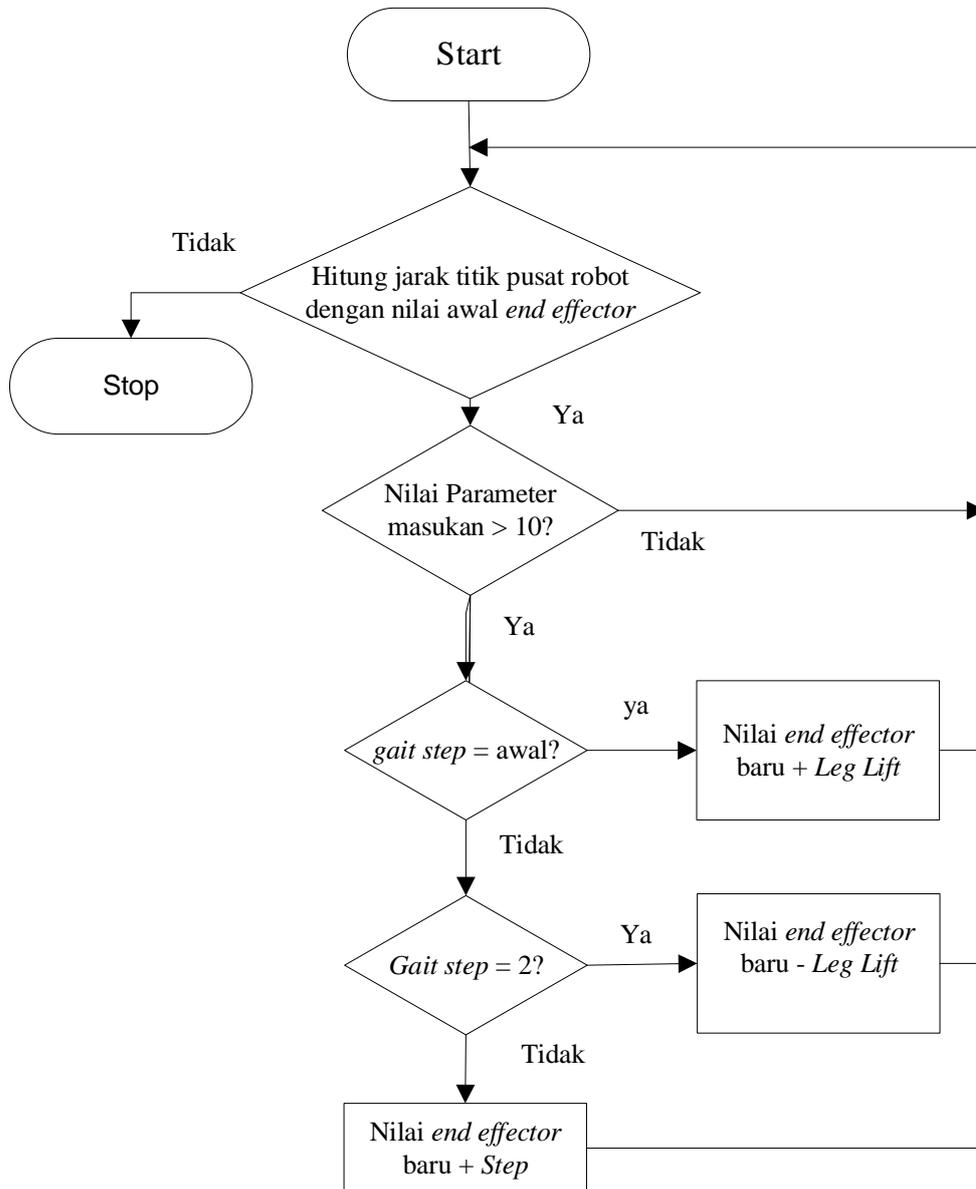
■ **Gambar 9** Sumbu koordinat pada robot[5]

Penamaan dan posisi setiap nilai awal *end effector* setiap kaki dapat dilihat pada Gambar 10. Skala dalam sumbu koordinat tersebut adalah satu milimeter. Artinya jika nilai koordinat pada sumbu z adalah 30 maka nilai sebenarnya adalah 30 milimeter.



■ **Gambar 10** Nilai awal *end effector*[5]

Program algoritma *gait* menghitung nilai-nilai *end effector* baru sesuai dengan pergerakan robot. Ada tiga parameter pergerakan robot yaitu pergerakan terhadap sumbu x, pergerakan terhadap sumbu z, dan juga pergerakan rotasi terhadap sumbu y. Parameter pergerakan terhadap sumbu x disebut *TravelLengthX*, parameter pergerakan terhadap sumbu z disebut *TravelLengthZ*, dan parameter pergerakan rotasi terhadap sumbu y disebut *TravelRotationY*. Jadi yang menjadi parameter masukan pada program algoritma *gait* adalah *TravelLengthX*, *TravelLengthZ*, dan *TravelRotationY*. Algoritma *gait* yang digunakan adalah algoritma *ripple gait 6 step*. Algoritma ini terdiri dari enam buah *step* yang masing-masing *step*-nya menghasilkan nilai *end effector* baru untuk setiap kaki robot.



■ Gambar 11 Diagram alir program algoritma gait

Pada *step* satu kaki *Right front leg* diangkat dengan ketinggian angkatan kaki yang telah ditetapkan pada parameter *LegLift* dan diposisikan pada koordinat x dan z yang sesuai dengan parameter masukan *TravelLengthX* dan *TravelLengthZ*. Pada *step* kedua kaki *Right front leg* diturunkan sampai pada koordinat sumbu y awal sebelum kaki diangkat. Nilai *TravelLengthX* dan *TravelLengthZ* dibagi empat yang kemudian akan digunakan untuk ditambahkan ke nilai *end effector* sebagai besarnya *step* untuk langkah kaki robot. Diagram alir program algoritma *gait* dapat dilihat pada Gambar 11.

Program algoritma *gait* direalisasikan dalam sebuah *procedure* program. Pemrograman menggunakan bahasa pemrograman C.

Realisasi program modul pengendali gerakan kaki

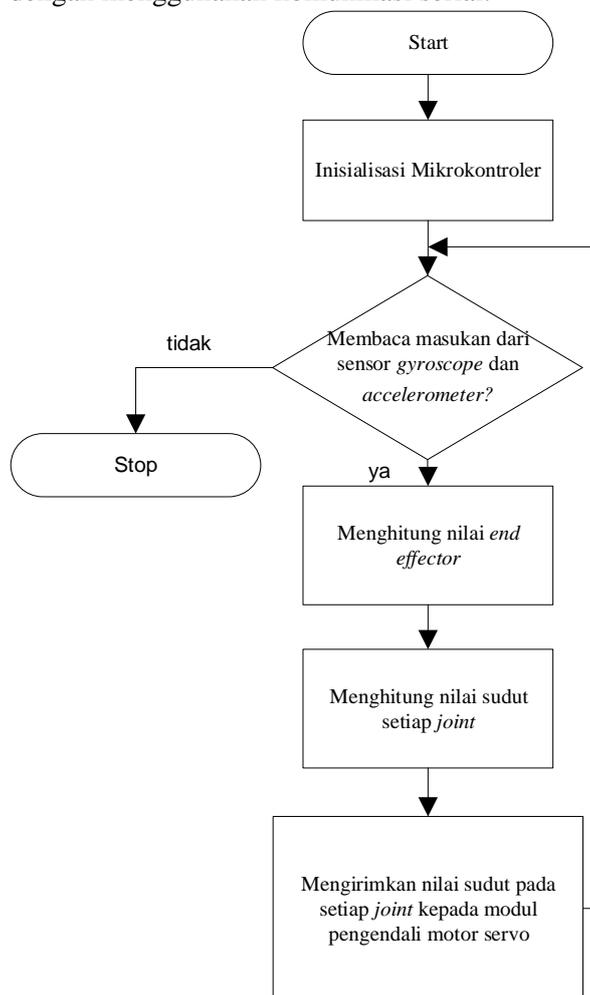
Realisasi program modul pengendali gerakan kaki robot adalah gabungan dari program algoritma *gait* dengan program *inverse kinematic*. Mikrokontroler membaca masukan dari sensor *gyroscope* dan *accelerometer* dengan menggunakan komunikasi serial. Data tersebut kemudian dijadikan sebagai nilai parameter *TravelLengthZ*, *TravelLengthX*, dan *TravelRotationY* pada program algoritma *gait* sehingga menghasilkan nilai *end effector* yang baru pada setiap kaki. Nilai *end effector* pada setiap kaki digunakan program *inverse kinematic* untuk menghitung nilai sudut setiap *joint* pada keenam kaki robot *hexapod*. Nilai sudut yang ada kemudian dikirimkan oleh mikrokontroler kepada modul pengendali motor servo agar modul pengendali motor servo dapat mengatur posisi motor servo sehingga sesuai dengan besar sudut yang ditetapkan melalui perhitungan *inverse kinematic*. Flowchart keseluruhan program dapat dilihat pada Gambar 12.

Nilai sudut pada setiap *joint* harus diubah dalam representasi lebar pulsa. Dari spesifikasi motor servo yang terdapat pada bagian 3.3.1 dapat diperoleh keterangan bahwa motor servo akan berada pada posisi tengah (posisi 90°) pada lebar pulsa 1500µs. Motor servo berada pada posisi 0° pada lebar pulsa 650 µs dan posisi 180°

pada lebar pulsa 2350 μ s. Dari data tersebut dapat dihitung resolusi pergerakan motor servo setiap perubahan lebar pulsa 1 μ s. Besarnya resolusi tersebut adalah :

$$\text{Resolusi Motor Servo} = \frac{180^\circ}{1700\mu\text{s}} = 0.1058823529411^\circ/\mu\text{s}$$

Setelah semua nilai sudut joint dirubah dalam nilai lebar pulsa maka nilai lebar pulsa tersebut dikirimkan kepada modul pengendali motor servo dengan menggunakan komunikasi serial.



■ **Gambar 12** Diagram alir modul pengendali gerakan kaki robot

Pengujian keseluruhan sistem

Setelah keseluruhan komponen diintegrasikan, perlu diadakan pengujian untuk mengetahui sistem berjalan sesuai dengan baik atau tidak. Pengujian dilakukan dengan menetapkan pusat x, y, z pada robot seperti pada Gambar 9.

Terlebih dahulu kita memberi nama pada sumbu X, Y, Z. Sumbu X sebagai *pitch*, sumbu Y sebagai *yaw*, dan sumbu Z sebagai *roll*. Berdasarkan hasil pembacaan sensor *gyroscope* ditetapkan pembacaan nilai sensor sebagai berikut:

- bila badan robot dalam keadaan menunduk ke depan, maka pembacaan nilai *pitch* akan bernilai negatif,
- bila badan robot dalam keadaan miring ke belakang, maka pembacaan nilai *pitch* akan bernilai positif,
- bila badan robot dalam keadaan miring ke kanan, maka pembacaan nilai *roll* akan bernilai negatif,
- bila badan robot dalam keadaan miring ke kiri, maka pembacaan nilai *roll* akan bernilai positif,

untuk pembacaan nilai *Yaw* tidak diperhatikan mengingat keseimbangan badan robot sudah bisa di hitung dan dianalisa menggunakan dua sumbu *pitch* dan *roll*.

Pengujian 1 keseimbangan badan robot berdiri pada permukaan bergerak

Pengujian pertama dilakukan untuk mengetahui apakah robot mampu menjaga keseimbangan badannya ketika permukaan dalam keadaan bergerak. Kondisi pengujian adalah sebagai berikut robot akan dinyalakan dan berdiri dalam keadaan diam, lalu permukaan robot dimiringkan dengan tingkat kemiringan yang berbeda-beda dan dilakukan pengambilan data sebanyak lima kali pengujian. Pengujian ini seperti terlihat pada Gambar 13



■ **Gambar 13** Keseimbangan badan robot ketika berdiri pada permukaan yang bergerak

Pengujian 2 keseimbangan badan robot berjalan di permukaan rumput

Pengujian kedua dilakukan untuk mengetahui apakah robot mampu menjaga keseimbangan badannya ketika berjalan di permukaan rumput. Kondisi pengujian adalah sebagai berikut robot akan dinyalakan dan berjalan di permukaan rumput dalam beberapa saat dan dilakukan pengambilan data sebanyak lima kali pengujian. Pengujian ini seperti terlihat pada Gambar 14



■ **Gambar 14** keseimbangan badan robot ketika berjalan di permukaan rumput

Pengujian 3 keseimbangan badan robot berjalan di permukaan miring 6,83°

Pengujian ke tiga dilakukan untuk mengetahui apakah robot mampu menjaga keseimbangan badannya ketika berjalan di permukaan miring 6,83°. Kondisi pengujian adalah sebagai berikut robot akan dinyalakan dan berjalan di permukaan miring dalam beberapa saat dan dilakukan pengambilan data sebanyak lima kali pengujian. Pengujian ini seperti terlihat pada Gambar 15



■ **Gambar 15** Keseimbangan badan robot ketika berjalan di permukaan miring

Pengujian 4 keseimbangan badan robot ketika berjalan di tanah berpasir

Pengujian ke empat dilakukan untuk mengetahui apakah robot mampu menjaga keseimbangan badannya ketika berjalan di tanah berpasir. Kondisi pengujian adalah sebagai berikut robot akan dinyalakan dan berjalan di tanah berpasir dalam beberapa saat dan dilakukan pengambilan data sebanyak lima kali pengujian. Pengujian ini seperti terlihat pada Gambar 16



■ **Gambar 16** Keseimbangan badan robot ketika berjalan di tanah atau tanah gembur

Pengujian 5 keseimbangan badan robot berjalan di permukaan miring 12,12°

Pengujian ke lima dilakukan untuk mengetahui apakah robot mampu menjaga keseimbangannya ketika berjalan di permukaan miring 12,12°. Kondisi pengujian adalah sebagai berikut robot akan dinyalakan dan berjalan di permukaan miring dalam beberapa saat dan dilakukan pengambilan data sebanyak lima kali pengujian. Pengujian ini seperti terlihat pada Gambar 17



■ **Gambar 17** Keseimbangan badan robot ketika berjalan di permukaan miring

Pengujian 6 keseimbangan badan robot berjalan di permukaan miring 10,67°

Pengujian ke enam dilakukan untuk mengetahui apakah robot mampu menjaga keseimbangannya ketika berjalan di permukaan miring 10,67°. Kondisi pengujian adalah sebagai berikut robot akan dinyalakan dan berjalan di permukaan miring dalam beberapa saat dan dilakukan pengambilan data sebanyak lima kali pengujian. Pengujian ini seperti terlihat pada Gambar 18



■ **Gambar 18** Keseimbangan badan robot ketika berjalan di permukaan miring

Seluruh hasil pengujian satu hingga enam dilakukan pencarian nilai rata-rata untuk mengetahui hasil keseluruhan pengujian untuk dibuat kesimpulan. Hasil rata-rata pengujian secara keseluruhan ditampilkan sebagai berikut seperti pada Tabel 1.

■ **Tabel 1** Hasil nilai rata-rata seluruh pengujian

Rata-rata	Sudut pengukuran				Persen Kesalahan			
	Pitch		roll		Pitch		roll	
	Max	Min	Max	Min	MAX	Σ	MAX	Σ
pengujian 1	10,71	-11,17	10,27	-12,16	422,62	2,20	468,52	1,86
pengujian 2	2,53	-2,83	3,12	-2,10	3,70	0,15	81,80	2,38
pengujian 3	12,18	-19,75	20,33	-9,41	796,15	2,01	133,81	2,00
pengujian 4	24,37	-13,98	21,77	-10,73	571,99	2,03	179,67	2,30
pengujian 5	10,31	-12,90	11,75	-13,58	641,09	0,87	330,42	4,24
pengujian 6	6,66	-7,79	5,54	-7,50	456,15	1,89	155,02	2,15
Rata-rata	11,13	-11,40	12,13	-9,25	481,95	1,53	224,87	2,49

Berdasarkan Tabel 1 dapat disimpulkan bahwa robot dapat menjaga keseimbangan badan pada setiap pengujian, dengan nilai rata-rata sudut pengukuran nilai *pitch* maksimum 11,13° dan nilai minimum -11,4° serta hasil sudut pengukuran nilai *roll* maksimum 12,13° dan nilai minimum -9,25°. Hasil persen kesalahan nilai *pitch* maksimum 481,95% dengan nilai rata-rata persen kesalahan 1,53%, serta hasil persen kesalahan nilai *roll* maksimum 224,87% dengan nilai rata-rata persen kesalahan 2,49%.

KESIMPULAN

Berdasarkan dari pengujian dan realisasi rancangan maka dapat disimpulkan sebagai berikut :

1. Perancangan yang dibuat telah berhasil dengan baik berjalan stabil pada permukaan yang tidak rata (*uneven*).
2. Kestabilan robot saat berjalan berada pada rentang nilai *pitch* maksimum 11,13° dan nilai minimum -11,4° serta rentang nilai *roll* maksimum 12,13° dan nilai minimum -9,25°, dari 5 kali pengujian untuk masing-masing kasus dan bidang pengujian.
3. Rancangan telah berhasil dengan baik dalam menjaga kestabilan badan saat berjalan dengan hasil nilai persen kesalahan nilai *pitch* 1,53% dan nilai *roll* 2,49%.

SARAN

Saran untuk pengembangan yang lebih lanjut yaitu :

1. Penambahan sensor GPS pada robot untuk mengetahui posisi keberadaan robot pada daerah bencana.
2. Penambahan secara mekanis sistem *shock breaker* untuk membantu pengurangan getaran pada saat berjalan.

DAFTAR PUSTAKA

- [1] Abhijit Mahapatra, Shibendu Shekhar Roy, Dilip Kumar Pratihar, "Modeling and Simulation of Wave Gait of a Hexapod Walking Robot: A CAD/CAE Approach", Departement of Mechanical Engineering, National Institute of Technology, Durgapur, India, 2012.
- [2] Dean J (1991) A model of leg coordination in the stick insect, *Carausius morosus*: II. Description of the kinematic model and simulation of normal step patterns. *Biological Cybernetics*, 64:403-411
- [3] History of Industrial Robots, IFR International Federation of Robotics, IFR Robotics, Frankfurt am Main, Germany, pp.10
- [4] Kadir Abdul, Buku Pintar Pemrograman Arduino, Yogyakarta, MediaKom, 2015
- [5] Phoenix, Mei 2017
<http://www.robotshop.com/forum/project-phoenix-a-hexapod-robot-t2743>
- [6] Pitowarno Endra, Robotika desain, control, dan kecerdasan buatan. Yogyakarta, ANDI, 2006
- [7] Syahrul, Pemrograman Mikrokontroler AVR Bahasa Assembly dan C. Bandung, Informatika, 2014
- [8] Suyadhi Taufiq, Build Your Own Line Follower Robot, Yogyakarta, ANDI, 2008
- [9] Thomas R. Kurfess, Robotics and Automation Handbook, New York, Taylor & Francis, 2005
- [10] Tim sar terkena bencana susulan, Mei 2017 <https://nasional.tempo.co/read/864608/syok-selepas-longsor-susulan-ponorogo-tim-menenangkan-diri-sehari>