

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Pada tinjauan pustaka metode pengumpulan data dilakukan dengan mencari jurnal dan literatur yang berkaitan dengan tugas akhir kemudian mempelajarinya. Berikut merupakan jurnal dari penelitian sebelumnya.

1. Penelitian tentang kinematika robot telah dilakukan oleh Julpri Andika Ketty Siti Salamah dalam sebuah jurnal berjudul Analisis Kinematik Pada Robot Hexapod tang bertujuan untuk menganalisis kinematik 3 derajat kebebasan pada kaki dari robot berkaki enam menggunakan parameter Denavit Hartenberg dari penelitian tersebut didapat kesimpulan bahwa Parameter Denavit Hartenberg dapat memecahkan dan menghasilkan persamaan kinematik maju dan balik pada lengan manipulator robot dengan 3 derajat kebebasan. Persamaan ini kemudian dapat diterapkan pada setiap kaki robot berkaki, khususnya pada robot berkaki 6 dengan 3 derajat kebebasan[3].
2. Penelitian tentang kontrol keseimbangan telah dilakukan oleh Nasution menggunakan robot EILERO versi empat kaki (quadruped). Pada penelitiannya Robot quadruped memiliki kelincahan yang lebih baik daripada robot hexapod. Namun kurang baik dalam menghadapi kemiringan lantai jika dibandingkan dengan robot hexapod[4].
3. Penelitian yang dilakukan oleh De Silva dan Sitte yaitu tentang merancang simulasi dinamis dari robot hexapod 18 derajat kebebasan dengan tujuan mengembangkan algoritma kontrol untuk berjalan mulus, efisien dan kuat di medan yang tidak beraturan. Mereka menggunakan sensor gaya sebagai tambahan dari sensor sudut sambungan konvensional sebagai proprioceptor. Gaya reaksi di kaki robot memberikan informasi yang diperlukan tentang interaksi robot dengan medan[5].

Berdasarkan data dari hasil penelitian tersebut dapat dipelajari dan diterapkan robot hexapod penyelamat. Berikut pada Tabel 2.1 merupakan tabel kesimpulan dari data hasil penelitian jurnal tersebut.

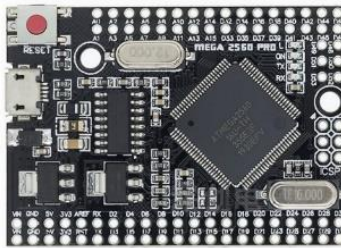
Tabel 2. 1 Kesimpula Jurnal yang digunakan

Nama Penulis	Pembahasan	kesimpulan
Julpri Andika dan Ketty Siti Salamah	Analisis kinematik pada robot hexapod yang bertujuan untuk menganalisis kinematik 3 derajat kebebasan pada kaki dari robot berkaki enam menggunakan parameter denavit hartenberg	Parameter Denavit Hartenberg dapat memecahkan dan menghasilkan persamaan kinematik maju dan balik pada lengan manipulator robot dengan 3 derajat kebebasan. Persamaan ini kemudian dapat diterapkan pada setiap kaki robot berkaki, khususnya pada robot berkaki 6 dengan 3 derajat kebebasan
Nasution	Kontrol keseimbangan robot hexapod EILERO menggunakan fuzzy logic	Robot quadruped memiliki kelincahan yang lebih baik daripada robot hexapod. Namun kurang baik dalam menghadapi kemiringan lantai jika dibandingkan dengan robot hexapod
De Silva dan Sitte	Merancang simulasi dinamis dari robot hexapod 18 derajat kebebasan dengan tujuan mengembangkan algoritma kontrol untuk berjalan mulus, efisien dan kuat di medan yang tidak beraturan.	Penggunaan sesnsor gaya pada kaki robot dapat memberikan informasi tentang interaksi robot dengan medan sehingga membuat robot berjalan mulus ketika di medan yang tidak beraturan

2.2 Dasar Teori

2.2.1 Arduino Mega Pro Mini

Arduino Mega 2560 Pro Mini merupakan board mikrokontroler yang kompatibel dengan Arduino Mega 2560. Board ini memiliki ukuran yang lebih kecil jika dibandingkan dengan board Arduino Mega 2560 yang memiliki ukuran yang besar. Pada robot ini arduino mega pro mini digunakan sebagai kontroler utama robot. Berikut pada Gambar 2.1 merupakan bentuk arduino mega promini dan juga spesifikasinya terdapat pada Tabel 2.2[6].



Gambar 2. 1 Arduino Mega Pro Mini

Tabel 2. 2 Spesifikasi Arduino

Komponen	Spesifikasi
<i>Microcontroller</i>	Atmega2560
USB-TTL	CH340
<i>Clock Frequency</i>	16MHz
Digital I/O	34
Analog I/O	16
<i>Memory Size</i>	256kb

2.2.2 Motor Servo MG996R

Motor Servo merupakan motor listrik dengan menggunakan sistem *closed loop*. Pada robot *hexapod* motor servo digunakan sebagai aktuator atau penggerak. Motor servo digunakan sebagai penggerak kaki agar robot dapat bergerak. Berikut pada Gambar 2.2 merupakan bentuk servo MG99R dan juga spesifikasinya pada Tabel 2.3[7].



Gambar 2. 2 Motor Servo MG996R

Tabel 2. 3 Spesifikasi Servo MG996R

Komponen	Spesifikasi
Servo	MG996R
<i>Stall torque</i>	9.6kg(4.8V), 11kg(6V)
<i>Operating speed</i>	0.17 s/60° (4.8 V), 0.14 s/60° (6 V)
<i>Operating voltage</i>	4.8 V a 7.2 V
<i>Running Current</i>	500mA – 900mA(6V)

2.2.3 Sensor MPU6050

Sensor MPU6050 digunakan untuk mendeteksi sudut kemiringan tertentu dengan terintegrasi pada sensor *gyroscope*[8]. Sensor ini digunakan untuk membaca kemiringan bodi robot dan juga sebagai umpan balik untuk stabilisasi badan robot. Sensor ini dipilih karena memiliki pembacaan interupsi yang cukup cepat. Berikut pada Gambar 2.3 merupakan bentuk sensor MPU6050 dan juga spesifikasinya pada Tabel 2.4[8].



Gambar 2. 3 Sensor MPU6050

Tabel 2. 4 Spesifikasi MPU6050

Komponen	Spesifikasi
Sensor	MPU6050
Catu Daya	3-5V
Komunikasi	I2C (SDA,SCL)
ADC	16-bit

2.2.4 Pixy Cam

Pixy adalah sensor penglihatan cepat untuk robotika DIY dan aplikasi serupa[9]. Pixy cam memiliki kemampuan untuk pendeteksian warna secara realtime berbasis camera. Komunikasi yang digunakan pixy cam adalah komunikasi (Serial Peripheral Interface) SPI dan dapat terhubung dengan arduino. Berikut pada Gambar 2.4 merupakan bentuk dari PIXY Cam[9].



Gambar 2. 4 PIXY Cam

2.2.5 Push Button

Push button switch (saklar tombol tekan) adalah perangkat / saklar sederhana yang berfungsi untuk menghubungkan atau memutuskan aliran arus listrik dengan sistem kerja tekan *unlock* (tidak mengunci). Pada robot ini push button digunakan sebagai tombol untuk menjalankan program robot. Berikut pada Gambar 2.5 merupakan bentuk dari *push button*[10].



Gambar 2. 5 Push Button

2.2.6 Stepdown 10A

Stepdown adalah mengubah tegangan tinggi dengan arus rendah menjadi tegangan rendah dengan arus tinggi. Fungsi utama *Stepdown* adalah menurunkan tegangan listrik dan menyesuaikannya dengan kebutuhan elektronika. Berikut pada Gambar 2.6 merupakan bentuk dari *Stepdown 10A*[11].



Gambar 2. 6 Stepdown 10A

2.2.7 *Limit Switch*

Limit switch adalah perangkat elektro-mekanis yang digunakan untuk mengirim sinyal listrik berdasarkan interaksi fisik. *Limit switch* pada robot *hexapod* ini digunakan pada ujung kaki robot dan berfungsi sebagai penghenti lintasan trayektori agar badan robot tetap stabil ketika salah satu kaki robot menginjak benda. Berikut pada Gambar 2.7 merupakan bentuk dari *limit switch*.



Gambar 2. 7 Limit Switch

2.3 *Robot Hexapod*

Robot *hexapod* adalah jenis robot yang dirancang khusus. Karakteristik utama dari robot *hexapod* ini adalah menggunakan enam kaki yang memberikan kestabilan dan mobilitas yang tinggi di berbagai medan sulit. Penggunaan enam kaki pada *hexapod* memberikan keunggulan dalam menavigasi melalui berbagai jenis permukaan. Selain itu, robot *hexapod* penyelamat dilengkapi dengan sensor-sensor canggih, MPU6050 untuk sebagai umpan balik untuk menstabilkan badan robot dan kamera untuk membantu dalam deteksi dan pemetaan kondisi di lingkungan sekitarnya. Robot *hexapod* juga memiliki lengan robot yang dirancang untuk melakukan tugas-tugas tertentu, seperti mengangkat korban dan memindahkan korban.

2.4 Lengan Robot

Lengan robot merupakan sebuah alat mekanik yang dapat melakukan tugas fisik menggunakan program yang telah didefinisikan terlebih dahulu untuk melakukan proses manipulasi dengan menggunakan lengan mekanis dengan tingkatan kebebasan dalam bergerak yang dapat di sesuaikan dengan kebutuhan. Ada dua jenis kinematik yang ada pada robot lengan, yaitu *forward kinematics* dan *inverse kinematics*.

Forward kinematics adalah analisis kinematik untuk mendapatkan koordinat posisi (x, y, z) jika diketahui sudut dari tiap sendi sedangkan *Inverse kinematics* adalah analisis kinematik untuk mendapatkan besar sudut dari masing-masing sendi jika diketahui koordinat posisi (x, y, z)[12].

2.5 DOF pada Lengan Robot

DOF (*Degrees of Freedom*) pada robot lengan merujuk pada jumlah derajat kebebasan atau gerakan yang dimiliki oleh lengan robot. Secara sederhana, DOF menggambarkan sejauh mana robot lengan dapat bergerak atau mengubah posisi dan orientasinya di ruang tiga dimensi. Sebuah DOF mewakili satu sumbu atau parameter gerakan yang dapat dikendalikan. Jumlah DOF yang lebih besar memberikan robot kemampuan untuk mencapai posisi dan orientasi yang lebih banyak dalam ruang tiga dimensi, meningkatkan fleksibilitas dan kemampuan untuk menangani berbagai tugas[13].

2.6 Metode Pergerakan kaki

Terdapat beberapa metode pergerakan kaki robot hexapod, di antaranya adalah *ripple gait*, *wave gait*, dan *tripod gait*.

1. *Ripple Gait*

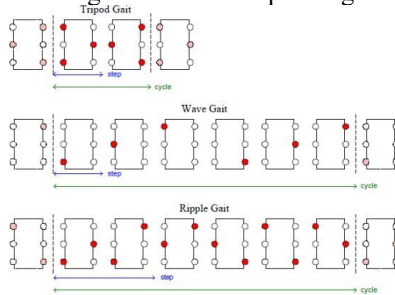
Pada metode ini robot berjalan dengan menggerakkan 2 buah kaki dalam satu waktu. Pada metode ini kaki-kaki yang bergerak berada pada sisi yang berbeda. Penggunaan metode ini membuat gerakan lebih stabil dari metode *tripod gait* dan lebih cepat dari metode *wave gait*[14].

2. *Wave Gait*

Pada metode ini robot menggerakkan kaki satu demi satu dalam menghasilkan gerakan. Dengan menggunakan metode ini, pada suatu waktu hanya ada satu kaki yang terangkat sehingga dengan menggunakan metode ini robot menjadi lebih stabil ketika bergerak namun lambat[14].

3. *Tripod Gait*

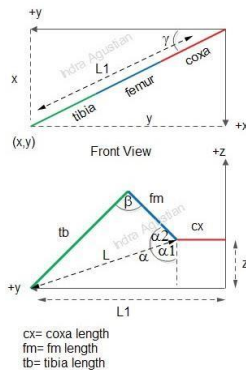
Pada metode ini robot berjalan dengan mengangkat 3 buah kaki pada satu waktu dan meninggalkan 3 kaki lainnya sebagai pijakan. Dengan menggunakan metode ini 1 siklus dapat diselesaikan dalam 2 langkah sehingga memungkinkan robot dapat bergerak cepat[14].



Gambar 2. 8 Ilustrasi Pola Gait

2.7 *Invers Kinematic*

Pada umumnya *inverse kinematic* digunakan untuk menggerakkan sebuah robot *manipulator* berdasarkan posisi titik koordinat yang dikehendaki. Untuk dapat menggerakkan kaki robot kita harus mendefinisikan posisi titik koordinat dalam *Cartesian space* terlebih dahulu. Proses ini membutuhkan data masukan berupa titik koordinat (x,y,z) yang menghasilkan data keluaran berupa nilai sudut tiap joint kaki robot[15]. Pada robot *Hexapod* yang akan dibuat memiliki 3 derajat kebebasan atau DOF (*Degre Of Freedom*) yang terdiri dari bagian *Coxa*, *Tibia*, dan *Femur*.



Gambar 2. 9 Analisa Invers kinematic

$$L1 = \sqrt{x^2 + y^2} \quad (1)$$

$$\gamma = \tan^{-1} \frac{x}{y} \quad (2)$$

$$L = \sqrt{(L1 - cx)^2 + z^2} \quad (3)$$

$$\beta = \cos^{-1}((tb^2 + fm^2 - L^2)/(2 \times tb \times fm)) \quad (4)$$

$$\alpha_1 = \cos^{-1}\left(\frac{z}{L}\right) \quad (5)$$

$$a_2 = \cos^{-1}((fm^2 + L^2 - tb^2)/(2 \times fm \times L)) \quad (6)$$

$$a = a_1 + a_2 \quad (7)$$

Keterangan :

cx = coxa merupakan panjang segmen dari titik gamma sampai titik alpha

fm = femur merupakan panjang segmen dari titik alpha sampai titik beta

tb = tibia merupakan panjang segmen dari titik alpha sampai ujung kaki

γ = sudut gamma kaki

β = sudut beta kaki

α = sudut alpha kaki

Pada Gambar 2.9 merupakan ilustrasi dari analisa *invers kinematic* 3 DOF yang terdiri dari sudut gamma, alpha, dan beta[16]. Untuk perhitungan *invers kinematic* terdapat pada persamaan (1) hingga (7), perhitungan tersebut menghasilkan sudut gamma, alpha, dan beta dengan inputan koordinat (x,y,z) dan konstanta coxa, femur, dan tibia.

2.8 Trayektori Langkah

Trayektori pada kaki robot merupakan lintasan gerak dari suatu titik end-point ke titik end-point lainnya. *Trayektori* ini diperlukan agar perubahan dari suatu titik ke titik lainnya berlangsung sama dan gerakan dari lengan robot menjadi lebih halus. *Trayektori* yang digunakan pada langkah kaki berbentuk kurva parabola yang dihasilkan dari persamaan polinomial. Persamaan polinomial ini menghasilkan titik – titik *trayektori* yang digunakan sebagai lintasan gerak langkah dari kaki robot tersebut. Berikut pada persamaan (8) merupakan persamaan polinomial yang digunakan dalam *trayektori* langkah:

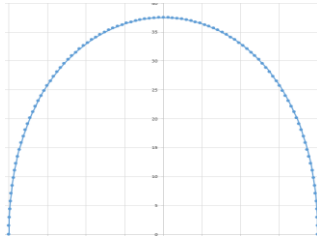
$$P(t)_{x,y,z} = (1 - t)^3 P1_{x,y,z} + 3t(1 - t)^2 P2_{x,y,z} + 3t^2(1 - t) P3_{x,y,z} + t^3 P4_{x,y,z} \quad (8)$$

Keterangan:

P = titik vektor acuan

t = faktor literasi

$P(t)_{x,y,z}$ merupakan *end-point* yang diinginkan pada iterasi ke-t, dan t merupakan faktor iterasi. P1 sampai P4 merupakan titik vektor acuan pada sumbu koordinat x, y, dan z. Titik-titik ini menentukan kurva *trayektori* yang dihasilkan. Nilai t adalah faktor literasi, yang terus meningkat dari nol sampai satu seiring bertambahnya waktu. Nilai t tersebut dinaikkan sebesar 0.1 atau sesuai kebutuhan[17].

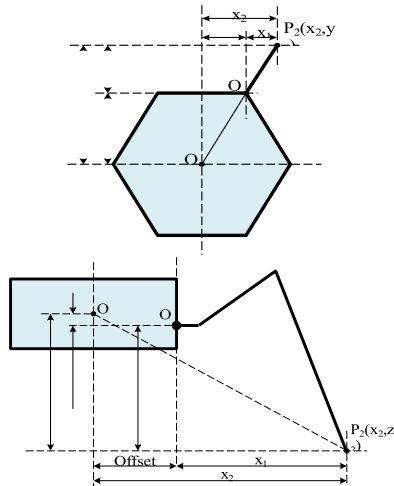


Gambar 2. 10 Kurva *Trayektori* dari persamaan Polinomial

Pada Gambar 2.10 merupakan kurva *trayektori* yang dihasilkan dari persamaan polinomial. Kurva tersebut terbentuk dari banyaknya titik koordinat yang membentuk lintasan parabola, lintasan parabola tersebut yang memungkinkan perpindahan langkah kaki menjadi lebih halus pergerakannya.

2.9 *Body Kinematic*

Body kinematics mengolah data posisi relatif setiap ujung kaki terhadap titik pusat bodi robot, kemudian digunakan untuk mengkoordinasikan setiap ujung kaki robot sehingga bodi robot dapat terintegrasi dengan koordinat bodi robot[17].



Gambar 2. 11 Analisa Bodi Robot pada Bidang XY dan XZ

Pada Gambar 2.11 merupakan ilustrasi dari analisa bodi robot terhadap bidang xy dan xz. Pada analisa bodi robot Titik O_0 merupakan titik pusat bodi robot dan titik O_1 merupakan titik sendi *coxa* kaki robot yaitu sebagai titik pusat *inverse kinematics* kaki. Titik P_1 merupakan posisi ujung kaki terhadap titik *coxa* dan titik P_2 merupakan posisi ujung kaki terhadap bodi robot. Nilai x_1 , y_1 , dan z_1 merupakan koordinat *inverse kinematics* kaki, sedangkan nilai *offset* x , y , dan z didapat melalui proses pengukuran pada bodi robot. Proses perhitungan posisi relatif ujung kaki terhadap bodi robot dijelaskan pada (9) hingga(11)[17].

$$x_2 = x_1 + \text{offset } x \quad (9)$$

$$y_2 = y_1 + \text{offset } y \quad (10)$$

$$z_2 = z_1 + \text{offset } z \quad (11)$$

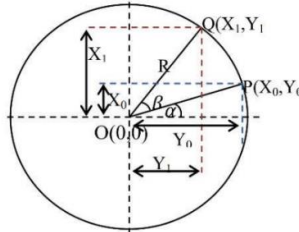
Keterangan:

x_2 : koordinat x berdasarkan bodi robot

y_2 : koordinat y berdasarkan bodi robot

z_2 : koordinat z berdasarkan bodi robot

Pada persamaan (9) hingga (11) digunakan untuk mencari koordinat bodi robot yang nantinya akan digunakan sebagai posisi awal badan robot. *Body kinematic* juga diterapkan pada permodelan geometri badan robot saat terkena rotasi bidang z. Dengan mengetahui posisi awal badan robot, maka koordinat akhir yang harus dituju oleh setiap kaki dapat dihitung dengan Persamaan (12) – (17)[18].



Gambar 2. 12 Permodelan Geometri Rotasi Bidang Z

$$\tan(a) = \frac{x_0}{y_0} \quad (12)$$

$$\tan(a + \beta) = \frac{x_1}{y_1} \quad (13)$$

$$x_1 = R \cos(a + \beta) \quad (14)$$

$$y_1 = R \sin(a + \beta) \quad (15)$$

$$\text{BodyIK}_x = x_1 - x_0 \quad (16)$$

$$\text{BodyIK}_y = y_1 - y_0 \quad (17)$$

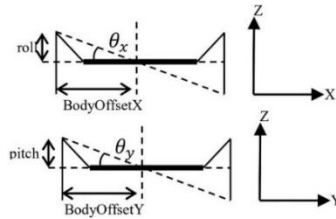
keterangan:

BodyIK_x = *Body kinematic* sumbu x

BodyIK_y = *Body kinematic* sumbu y

BodyIK_z = *Body kinematic* sumbu z

Pada Gambar 2.12 merupakan ilustrasi permodelan geometri rotasi bidang z. *Body kinematic* juga digunakan untuk rotasi pada bidang x dan y (*roll dan pitch*) menyebabkan nilai koordinat z pada tiap kaki berubah. Gambar 2.13 menunjukkan pemodelan geometri badan robot Ketika terjadi rotasi di bidang x dan z[18].



Gambar 2. 13 Permodelan Geometri Bidang Z

$$roll = BodyOffsetY \tan \theta_y \quad (18)$$

$$pitch = BodyOffsetX \tan \theta_x \quad (19)$$

$$BodyIK_z = roll + pitch \quad (20)$$

Pada Gambar 2.13 merupakan ilustrasi permodelan geometri bidang z yang digunakan untuk mencari ketinggian tiap kaki berdasarkan inputan sudut kemiringan. Nilai *roll* dan *pitch* dapat diperoleh dengan Persamaan (18) dan (19). Sedangkan nilai koordinat z dapat diketahui melalui Persamaan(20)[18].

2.10 Sistem Kendali (*Proportional Integral Derivative*)

PID adalah singkatan dari Proportional-Integral-Derivative. Ini adalah metode kontrol yang umum digunakan dalam berbagai sistem kendali untuk mengendalikan sistem dinamis. Penggunaan PID bertujuan untuk mengatur variabel agar tetap mendekati nilai yang diinginkan atau *set point* dengan cara meminimalkan kesalahan atau *error* antara nilai yang diinginkan dan nilai yang aktual. PID sebagai sistem kendali yang akan diimplementasikan pada robot, digunakan sebanyak tiga pengontrol. Pengontrol PID dengan masukan nilai *yaw*, agar robot berjalan lurus, dan pada masukan *pitch* dan *roll* untuk menyeimbangkan posisi robot, dengan *set point* 0 karena posisi awal setelah kalibrasi adalah 0. Perancangan PID didasarkan pada Persamaan (21), dengan nilai $u(t)$ merupakan nilai kendali berasal dari nilai *error* ($e(t)$)[19].

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d de(t) dt \quad (21)$$

Keterangan:

e = error

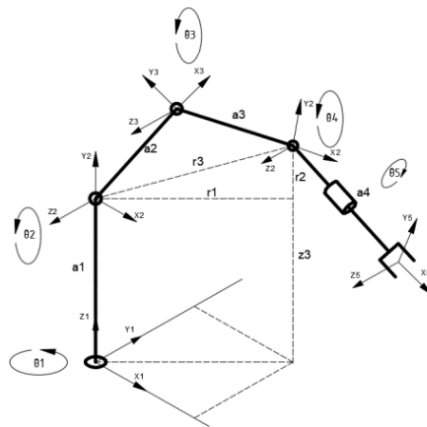
k_p = konstanta *proportional*

k_i = konstanta *integral*

k_d = konstanta *derivatif*

2.11 Invers kinematic Robot ARM 5 DOF

Perhitungan analitis *inverse kinematika* pada robot ARM 5 dof di gunakan untuk menentukan sudut dari setiap joint pada lengan robot. Perhitungan *invers kinematic* menggunakan nilai x , y , dan z (*end point*) yang telah ditentukan sebagai masukan awal dalam perhitungan *invers kinematic*. *End point* adalah perangkat di ujung robot lengan. Robot berkerja dalam ruang 3 dimensi maka sumbu koordinat kartesiusnya terdiri dari sumbu x , sumbu y , dan sumbu z [20].



Gambar 2. 14 Permodelan Robot ARM 5 DOF

manipulator dari koordinat kartesian (x,y,z) diselesaikan dengan rumus trigonometri berikut :

$$r1 = \sqrt{x^2 + y^2} - a4 \cos(\omega) \quad (22)$$

$$r2 = z3 - a1 \quad (23)$$

$$z3 = z - a4 \sin(\omega) \quad (24)$$

$$r3 = \sqrt{r1^2 + r2^2} \quad (25)$$

$$\theta_1 = \tan^{-1}\left(\frac{x}{y}\right) \quad (26)$$

$$\theta_2 = \tan^{-1}\left(\frac{r_2}{r_1}\right) + \cos^{-1}\left(\frac{a_3^2 + a_2^2 - r_3^2}{-2a_2r_3}\right) \quad (27)$$

$$\theta_3 = \cos^{-1}\left(\frac{r_3^2 + a_2^2 - a_3^2}{-2a_2a_3}\right) - 180 \quad (28)$$

$$\theta_4 = \omega - \theta_2 - \theta_3 \quad (29)$$

$$\theta_5 = \cos(\omega) \times \theta_1 - \theta_1 + \varepsilon \quad (30)$$

Keterangan:

θ_1 = Sudut pada Theta 1

θ_2 = Sudut pada Theta 2

θ_3 = Sudut pada Theta 3

θ_4 = Sudut pada Theta 4

θ_5 = Sudut pada Theta 5

ω = orientasi *pitch* pada *end point* (*gripper*)

ε = orientasi *Roll* pada *end point* (*gripper*)

a_1 = panjang Lengan 1

a_2 = panjang Lengan 2

a_3 = panjang Lengan 3

a_4 = panjang *Gripper*

pada Gambar 2.14 merupakan ilustrasi permodelan *invers kinematic* dari robot ARM 5 DOF. Pada persamaan (22) hingga (30) digunakan untuk mencari nilai sudut theta 1 hingga theta 5 dengan inputan koordinat (x,y,z) dan *pitch* dan *roll* pada *end point*.