

PERANCANGAN DAN IMPLEMENTASI ROBOT KESEIMBANGAN BERODA DUA BERBASIS MIKROKONTROLER

DESIGN AND IMPLEMENTATION OF BALANCE TWO-WHEELED ROBOT BASED MICROCONTROLLER

¹Grace Bobby, ² Erwin Susanto, ³ Fiky Yosep Suratman

^{1,2,3}Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom

Jalan Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia

¹gracebobbysidauruk@students.telkomuniversity.ac.id, ²erwinelektro@telkomuniversity.ac.id,

³fysuratman@telkomuniversity.ac.id

ABSTRAK

Perkembangan dunia robot berkembang pesat dari tahun ke tahun. Salah satu contohnya ialah Segway Personal Transporter. Variasi teknik dalam pergerakan robot pada lingkungan yang dinamik pun semakin banyak, diantaranya Pole-Placement Controller, Fuzzy Logic, Proportional Integrated Derivative Controller (Kontrol PID). Pada tugas akhir ini Fuzzy Logic akan digunakan sebagai pengontrol robot keseimbangan ini, karena dengan kontrol ini, tingkat kestabilan dan controllability yang lebih baik. Pada sistem ini, digunakan dua sensor (accelerometer dan gyroscope) untuk mendapatkan pembacaan data yang stabil dan handal. Dari hasil percobaan kalman filter, diperoleh nilai parameter kalman filter yang optimal adalah $Q_{\text{accelerometer}} = 0,001$; $Q_{\text{gyroscope}} = 0,003$ dan $R_{\text{pengukuran}} = 0,03$.

Kata Kunci : Fuzzy Logic, Kalman Filter, Self-balancing Control

ABSTRACT

The development of robots is growing rapidly from year to year. One example is the Segway Personal Transporter. A variety of techniques in the movement of the robot in the dynamic environment became more numerous, including Pole-Placement Controller, Fuzzy Logic, Proportional Integrated Derivative Controller (PID control). In this final project, Fuzzy Logic will be used as an balancing robot controller, because with this control, the level of stability and controllability better. In this system, used two sensors (accelerometer and gyroscope) to obtain data readout is stable and reliable. From the experimental of Kalman filter, obtained the optimal parameter values of Kalman filter are $Q_{\text{accelerometer}} = 0.001$; $Q_{\text{gyroscope}} = 0.003$ and $R_{\text{pengukuran}} = 0.03$.

Keywords : Fuzzy Logic, Kalman Filter, Self-balancing Control

1. Pendahuluan

Di dunia modern saat ini. Penelitian mengenai robot sangatlah banyak dan berkembang pesat, disertai dengan kemajuan teknologi yang terus menerus berkembang. Salah satu yang membuat penulis ingin untuk mengembangkan pengetahuannya mengenai robot dan sejenisnya. Robot yang akan dibahas disini adalah robot yang akan dapat menyeimbangkan dirinya hanya dengan dua roda. Dimana pada saat ini kita sudah mengenal adanya kendaraan Segway Personal Transporter. Kendaraan yang memiliki dua roda, dikanan dan dikiri sebagai penyeimbangnya.

Di sini kita akan merancang prototip dari Segway. Seperti yang sudah dibahas sebelumnya, ini juga merupakan pengembangan dari pendulum terbalik yang di letakan diatas kereta beroda. Untuk dapat menyeimbangkannya dibutuhkan metode kontrol yang baik dan handal untuk mempertahankan posisi robot dalam keadaan tegak lurus terhadap permukaan bumi tanpa memerlukan pengendali dari luar dan menerapkan *Self-balancing Control*. Maka dari itu penulis memilih kontrol Fuzzy Logic sebagai kontrol yang dapat menciptakan kontrol sesuai dengan yang diharapkan penulis.

Pada Tugas Akhir ini digunakan Arduino uno,serta memakai sensor *accelerometer* dan *gyroscope* yang akan mengendalikan motor DC. Sehingga robot ini dapat mempertahankan

posisinya tegak lurus dengan seimbang terhadap permukaan bumi pada bidang datar. Dan dapat menerapkan *Self-balancing Control* pada robot.

2. Dasar Teori

2.1 Balancing Robot

Balancing robot beroda dua merupakan suatu robot mobile yang memiliki dua roda di kedua sisinya dan tidak akan seimbang tanpa sebuah metode kontrol yang baik. Saat balancing robot beroda dua condong ke depan atau miring ke belakang, maka yang perlu dilakukan adalah motor akan memutar searah jarum jam sehingga balancing robot beroda dua akan berputar ke arah depan. Gaya yang digunakan untuk menyeimbangkan dihasilkan dari putaran roda.

2.2 Fuzzy Logic

Logika Fuzzy merupakan logika dalam pengambilan keputusan yang digunakan untuk memecahkan masalah dengan sistem yang sulit untuk dimodelkan, teori tentang *fuzzy set* pertama kali diperkenalkan dan dikembangkan oleh Dr. Lotfi A. Zadeh dari Universitas California, Berkeley pada tahun 1965. Dalam kehidupan banyak masalah dengan informasi yang sulit direpresentasikan ke dalam sebuah model rumus atau angka yang pasti karena informasi tersebut bersifat kualitatif (tidak bisa dihitung secara kuantitatif). Fuzzy logic dibagi menjadi tiga bagian proses, yaitu:

1. Fuzzyfication

Fuzzyfication merupakan proses pengubahan data masukan yang berupa nilai kebenaran bersifat pasti (*crisp input*) menjadi masukkan *fuzzy* yang berupa nilai linguistik dengan cara pemetaan *crisp input* pada himpunan *fuzzy*.

2. Inference

Suatu aturan *fuzzy* dituliskan sebagai: *IF antecedent THEN consequent*. Dalam suatu sistem berbasis aturan fuzzy, proses *inference* memperhitungkan semua aturan yang ada dalam basis pengetahuan. Hasil dari proses *inference* direpresentasikan oleh suatu *fuzzy set* untuk setiap untuk setiap *variable* bebas (pada *consequent*). Derajat keanggotaan untuk setiap nilai *variable* tidak bebas menyatakan ukuran kompatibilitas terhadap *variable* bebas (pada *antecedent*).

3. Defuzzification

Terdapat berbagai metode *defuzzification* yang dapat diaplikasikan untuk berbagai macam masalah. Metode *Weighted Average* mengambil nilai rata-rata dengan menggunakan pembobotan berupa derajat keanggotaan. Sehingga y^* didefinisikan sebagai :

$$y^* = \frac{\sum \mu(y) \cdot y}{\sum \mu(y)} \quad (2.1)$$

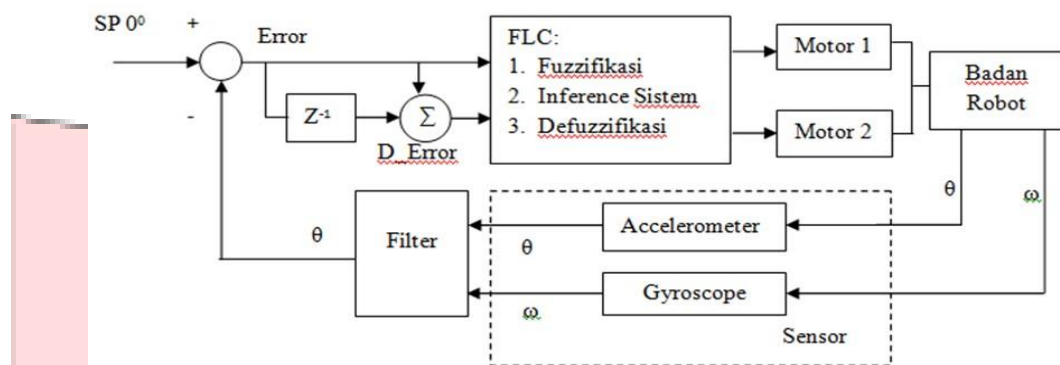
Dimana y adalah nilai *crisp*, $\mu(y)$ adalah derajat keanggotaan dari nilai *crisp* y . Kelemahan dari metode ini hanya bisa digunakan bila fungsi keanggotaan dari keluaran *fuzzy* memiliki bentuk yang sama.

2.4 Inertial Measurement Unit (IMU)

6-axis IMU yang penulis pakai adalah MPU-6050 yang mana sensor ini terhubung secara I2C. IMU digital ini memiliki 3-axis *accelerometer* dan 3-axis *gyroscope*, dan ketika dikombinasikan dengan filter, akan menampilkan pembacaan kemiringan yang sangat stabil.

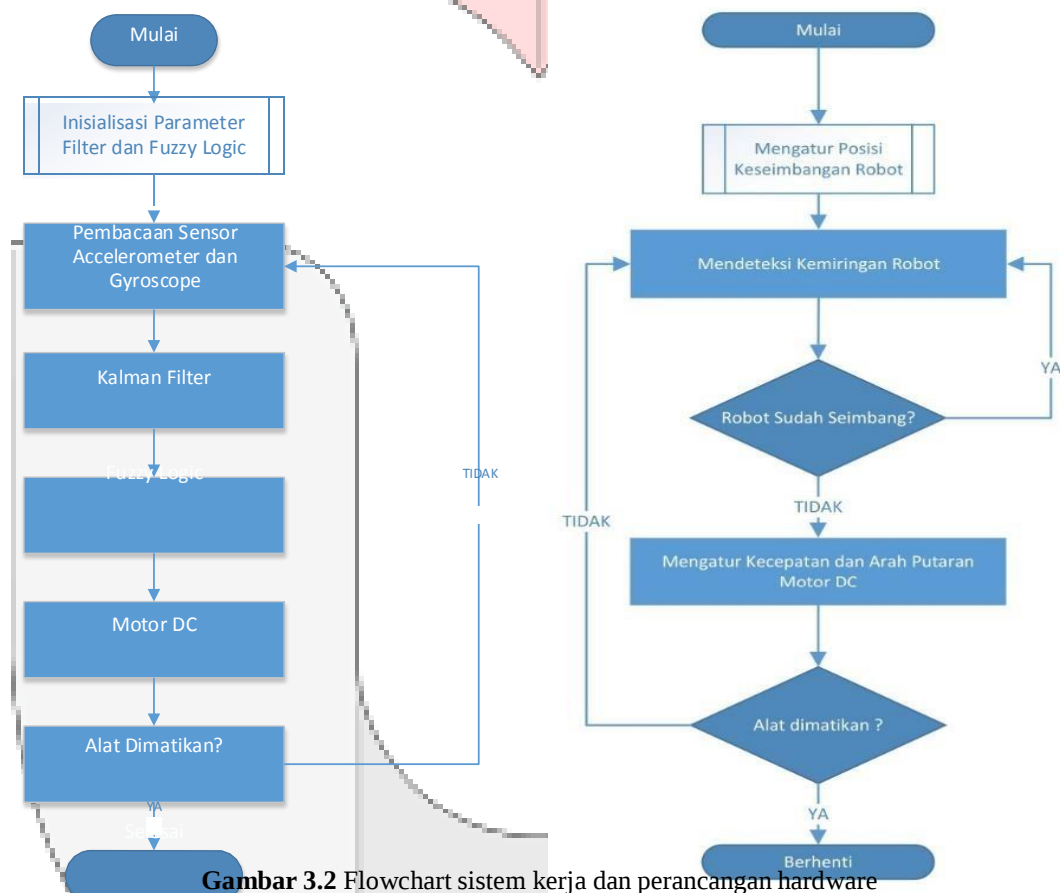
3. Perancangan Sistem

Perancangan robot keseimbangan terdiri dari beberapa blok sistem yang diintegrasikan menjadi satu sistem utuh. Pembagian blok sistem dibagi menjadi blok sensor, blok pengolahan data, dan blok keluaran. Pada blok sensor, robot menggunakan sensor IMU sebagai masukan. Pertama adalah gabungan sensor *accelerometer* dan *gyroscope* yang berfungsi untuk mengukur derajat robot yang berporos pada pusat bumi. Pada bagian keluaran merupakan pengaturan arah pergerakan dan kecepatan dua motor DC yang dipasang secara diferensial dengan menggunakan *driver* motor. Data yang dihasilkan dari bagian pengolahan akan digunakan untuk mengatur arah gerak dari kedua motor serta mengatur kecepatannya menggunakan *duty cycle*. Berikut ini adalah diagram blok dari sistem yang digunakan dalam pengerjaan tugas akhir:

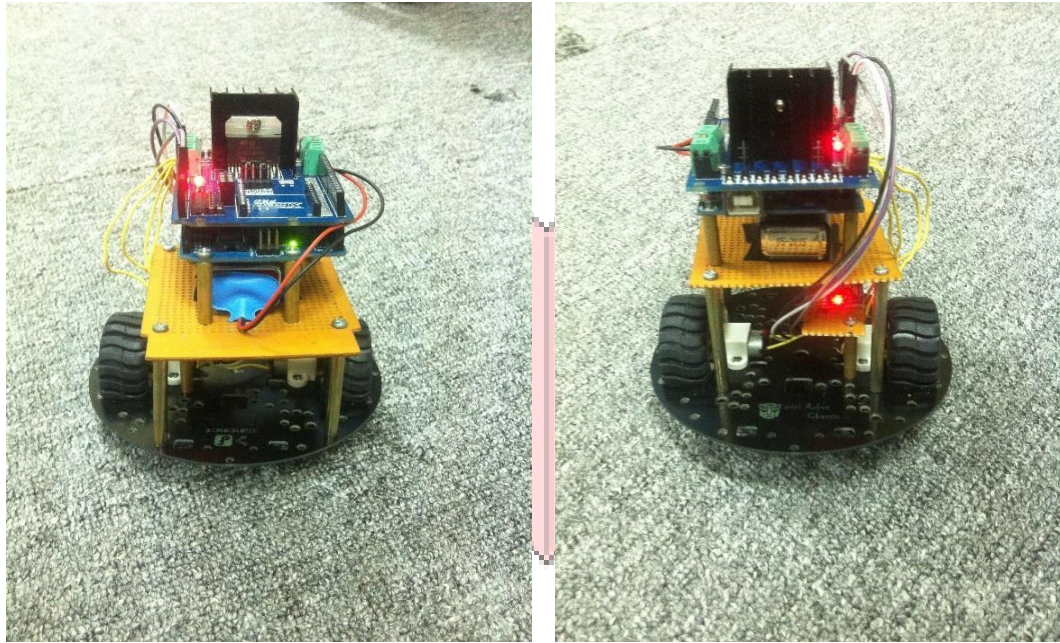


Gambar 3.1 Diagram Blok Sistem^[2]

Sementara untuk *flowchart* sistem yang digunakan oleh mikrokontroler dapat dilihat pada gambar berikut ini :



Gambar 3.2 Flowchart sistem kerja dan perancangan hardware



Gambar 3.3 Bentuk Fisik Robot tampak Depan dan Belakang

3.1 Perancangan Fuzzy Logic

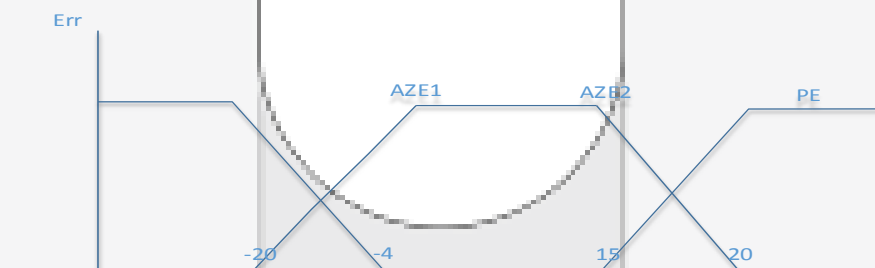
Logika fuzzy digunakan sebagai pengolah data masukan dari sensor IMU. Hal ini bertujuan agar proses penyeimbangan berjalan lancar.

3.1.1 Fuzzyfication

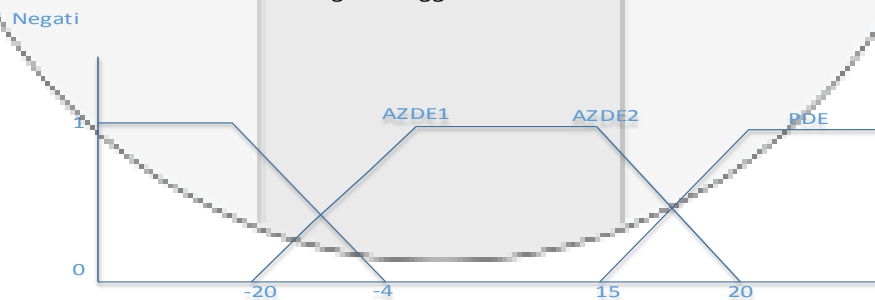
Data hasil pembacaan sensor IMU merupakan data masukan bernilai tegas (*crisp*) kemudian diubah menjadi himpunan fuzzy menurut fungsi keanggotaannya.

Proses awal dari fuzzyfikasi adalah membuat *membership function* (fungsi keanggotaan) dari masukan, serta menentukan banyaknya *variable linguistic* dalam fungsi keanggotaan tersebut. Dari fungsi keanggotaan yang dibuat akan diketahui nilai derajat keanggotaan dari masing-masing *variable* dalam himpunan fuzzy berdasarkan masukan tegas (*crisp*).

Pada sistem ini terdapat masukan dari sensor IMU. Masukan dari sensor IMU ini memiliki masing-masing empat nilai linguistik untuk masing-masing sensor **NE**, **AZE1**, **AZE2** dan **PE**. Masukan kedua dari delta error **NDE**, **AZDE1**, **AZDE2** dan **PDE** dengan fungsi keanggotaan bahu trapesium.



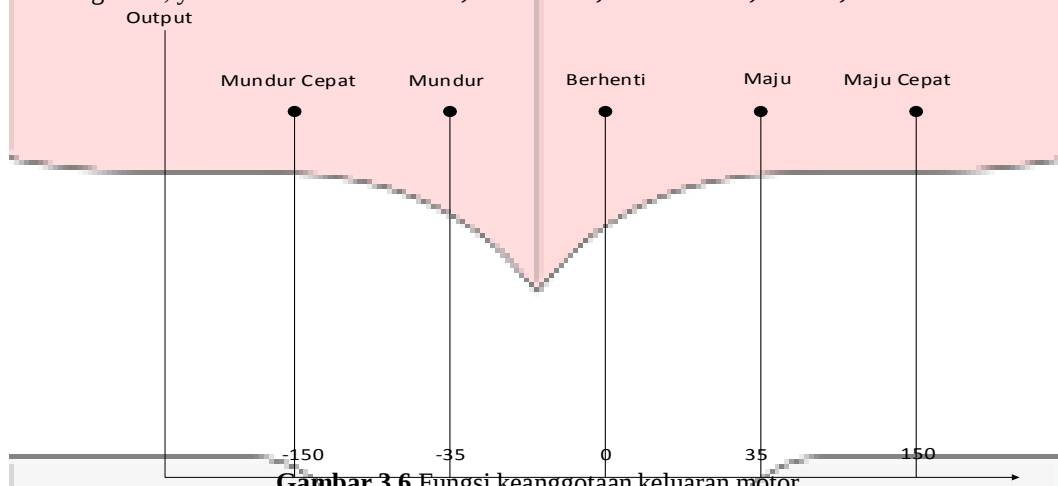
Gambar 3.4 Fungsi keanggotaan masukan sensor error



Gambar 3.5 Fungsi keanggotaan masukan sensor negatif error

Pada keluaran sistem menggunakan model sugeno, pembentukan fungsi keluaran pada model sugeno memiliki fungsi yang lebih sederhana dengan respon lebih cepat dari model yang lain. Bentuk keluaran fungsi keanggotaan pada model sugeno mempunyai bentuk *singletone*, bentuk dengan derajat keanggotaan satu pada suatu nilai *crisp* tunggal dan nilai nol pada suatu *crisp* yang lain.

Karena keluaran dalam bentuk *singletone* maka fungsi pada setiap nilai linguistik bernilai satu dan nol pada diluar nilai linguistik. Keluaran pada sistem yang dibuat ada dua, yaitu PWM pengontrol motor dc kanan dan kiri. Untuk keluaran sistem yang berupa kecepatan memiliki lima nilai linguistik, yaitu: **MUNDUR CEPAT, MUNDUR, BERHENTI, MAJU, MAJU CEPAT**.



Gambar 3.6 Fungsi keanggotaan keluaran motor

3.1.2 Rule Inference

Pada *rule inference*, terjadi proses pengolahan data masukan fuzzyfikasi dengan hasil keluaran yang dikehendaki dengan aturan-aturan tertentu. Dari aturan-aturan yang dibentuk inilah yang nantinya akan menentukan respon dari sistem terhadap berbagai kondisi *set point* dan gangguan yang terjadi pada sistem yang akan dibuat. Rule inference sistem tertulis pada tabel 3.1

Tabel 3.1 Rule Inference motor

Error	NE	AZE1	AZE2	PE
Delta Errpr				
NDE	MC-	MC-	M+	M+
AZDE1	M-	M-	B	M+
AZDE2	MC-	M-	B	MC+
PDE	M-	B	MC+	MC+

Keterangan : MC-=Mundur Cepat, M-= Mundur, B=Berhenti, M+= Maju, MC+= Maju Cepat.. Berdasarkan tabel, maka sistem mempunyai 16 aturan fuzzy, yaitu:

- 1 Jika Error = **NE** dan Delta Error = **NDE** maka Motor DC = **MUNDUR CEPAT**
- 2 Jika Error = **NE** dan Delta Error = **AZDE1** maka Motor DC = **MUNDUR** .
- 3 Jika Error = **NE** dan Delta Error = **AZDE2** maka Motor DC = **MUNDUR CEPAT**
- .
- .
- .
- 25 Jika Error = **PE** dan Delta Error = **PDE** maka Motor DC = **MAJU CEPAT** .

3.1.3 Defuzzification

Defuzzifikasi merupakan pemetaan bagi nilai-nilai *fuzzy* keluaran yang dihasilkan pada tahap *rules inference* ke nilai-nilai keluaran kuantitatif. Pada perancangan robot mobil ini proses

defuzzifikasi menggunakan metode *Weigth Average* dan keluaran dari proses defuzzifikasi berupa nilai PWM yang nantinya digunakan untuk mengontrol kecepatan motor DC.

3.2 Kalman Filter Diskrit

Kalman filter merupakan filter digital rekursif yang dapat mengestimasi proses dengan sangat efektif. Kalman filter dapat mengurangi noise pada pengukuran sensor sebelum masuk ke dalam sistem kontrol. Oleh karena itu kalman filter sering digunakan pada sistem kontrol yang sensitif terhadap noise karena dapat meminimalkan *square error*. Pada implementasinya kalman filter tidak memerlukan data yang harus disimpan. Setiap data pengukuran yang masuk akan diproses ulang. Sehingga kalman filter akan lebih mudah untuk diimplementasikan pada mikrokontroler. Sebelum kalman filter digunakan untuk membaca data dari sensor, proses atau sistem harus dimodelkan dalam sistem linear.

$$\hat{x}_k = \hat{x}_{k-1} + \Delta t \cdot \hat{u}_{k-1} \quad (3.1)$$

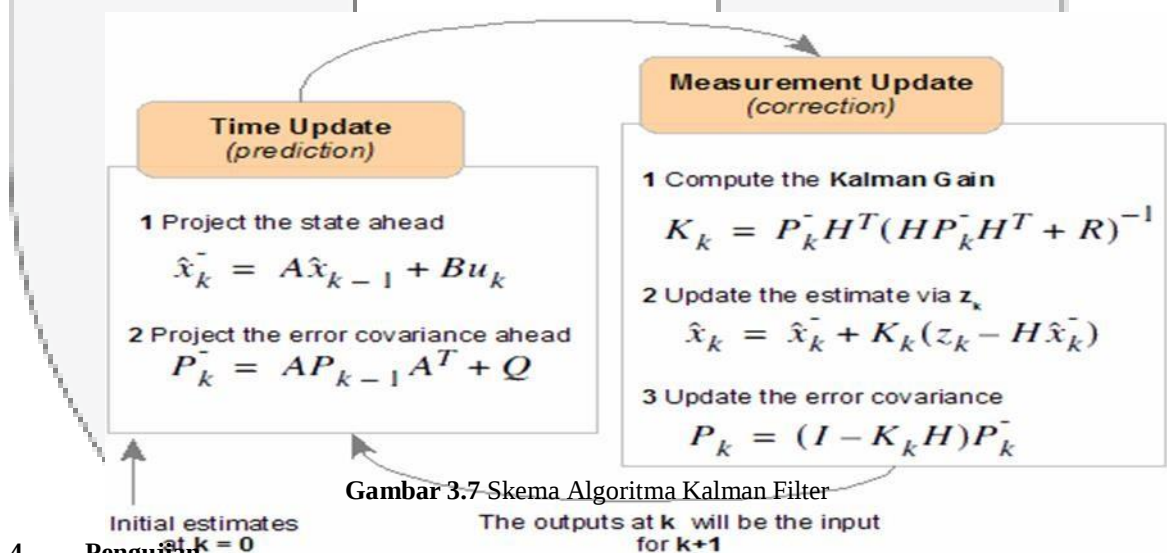
$$z_k = H\hat{x}_k + v_k \quad (3.2)$$

Persamaan 3.1 merepresentasikan persamaan state proses. $\hat{x}_k$ merupakan kombinasi dari nilai sebelumnya dengan sinyal $\hat{u}_{k-1}$ dan Δt yang merupakan *process noise*. Persamaan 3.2 menyatakan *output state* yang merupakan hasil dari pengukuran sensor. v_k menyatakan *noise* pengukuran yang merupakan *independent random noise variables*. Tapi keduanya dapat diuraikan dengan matriks kovarian Q dan R . Sehingga dapat dituliskan pada persamaan 3.3 dan 3.2.

$$Q = E \{ v_k v_k^T \} \quad (3.3)$$

$$R = E \{ v_k v_k^T \} \quad (3.4)$$

Dari persamaan 3.3 dan 3.4, T adalah *matrix transpose* dan E merupakan nilai estimasinya. *Process noise covariance* (Q) matrix dan *measurement noise covariance* (R) merupakan komponen penting dalam kualitas hasil dari keluaran kalman filter. Kedua matriks kovarian ini dapat diperoleh dengan menggunakan persamaan statistic. Namun, pada tugas akhir ini penulis menggunakan metode *tunning*. Yaitu dengan *trial and error* sampai mendapat nilai yang paling baik untuk pembacaan sensor yang digunakan. Kalman filter memperkirakan keluaran sensor dengan skema umpan balik. Pertama, kalman filter akan memperkirakan keadaan pada suatu waktu. Lalu mengambil nilai pengukuran yang ber-noise dalam bentuk umpan balik. Maka perhitungan dari kalman filter akan terbagi menjadi dua bagian, yaitu *Time Update* (kondisi prediksi) dan *Measurement Update* (kondisi koreksi). Dua persamaan inilah yang menjadi dasar dari kalman filter.



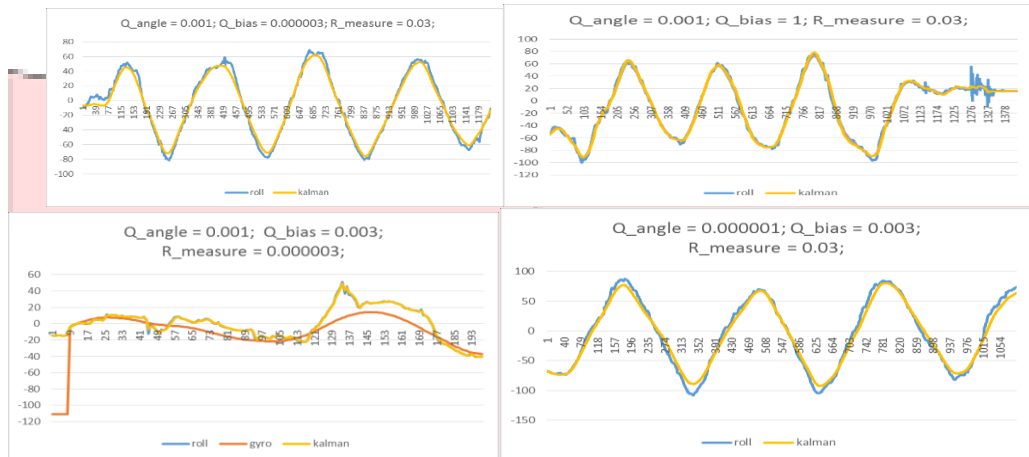
Gambar 3.7 Skema Algoritma Kalman Filter

4. Pengujian

4.1 Pengujian Kalman Filter

Seperti yang pernah dijelaskan, tujuan utama dari Kalman filter adalah menghilangkan *noise* pengukuran pada *sensor accelerometer* serta mengurangi *drift* dari *gyroscope*. Kalman filter memberikan prediksi sudut yang lebih baik dari data asli dari *accelerometer*. Agar filter dapat

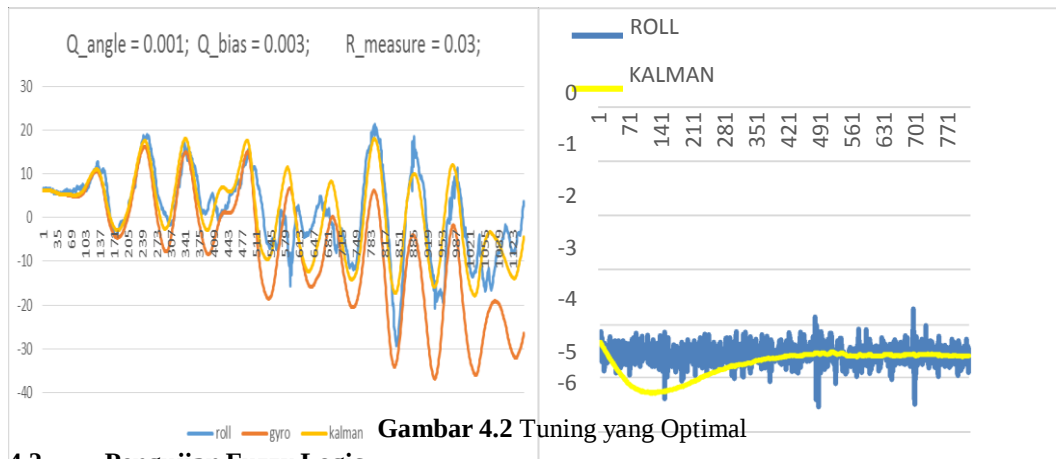
berjalan dengan lancar, maka perlu dilakukan *tuning* manual. Parameter yang perlu dituning adalah Q_w dan R_v . Matriks Q_w terdiri dari *noise* proses sedangkan R_v merupakan *noise* pengukuran.



Gambar 4.1 Tuning yang tidak baik

Setelah melakukan beberapa percobaan untuk mencari nilai Q_w dan R_v yang optimal. Didapatkan nilai parameter kalman filter yang optimal adalah:

- $Q_{\text{angle}} (Q_{\text{accelerometer}}) = 0.001$
- $Q_{\text{bias}} (Q_{\text{gyroscope}}) = 0.003$
- $R_{\text{measure}} = 0.03$

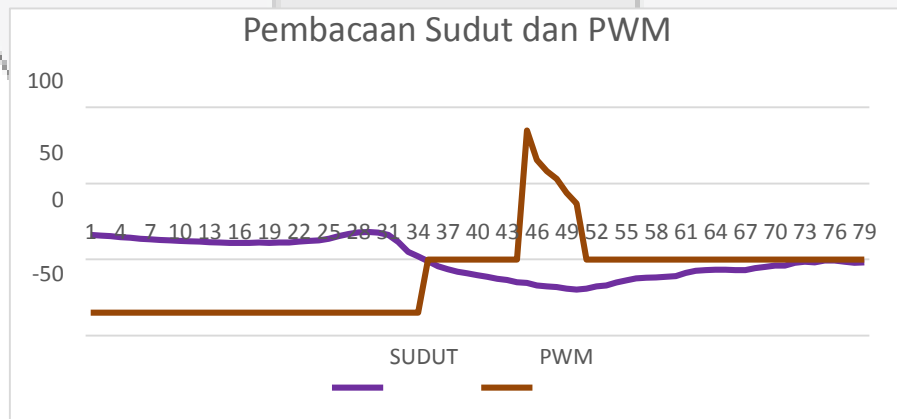


Gambar 4.2 Tuning yang Optimal

4.2 Pengujian Fuzzy Logic

Tujuan dari Fuzzy Logic adalah kestabilan. Terlihat robot menjaga kestabilannya di sudut 0° . Pada gambar terlihat bahwa pada awal inisialisasi terdapat sebuah overshoot yang disebabkan inisialisasi prediksi sudut kalman, akan tetapi setelah itu overshoot itu hilang.

Parameter Fuzzy ialah : $-20^\circ, -4^\circ, 15^\circ, 20^\circ$.



Gambar 4.3 Pembacaan Sudut dan PWM

5. Kesimpulan

Dengan menggunakan 2 buah input pada proses fuzzifikasi dan menggunakan 16 rules, robot mampu mencapai kesetimbangannya kembali (*steady state*) setelah mendapatkan gangguan dari luar. Pengujian robot keseimbangan dengan menggunakan kendali fuzzy logic dapat membuat robot berdiri tegak pada pusat bumi..

6. Daftar Pustaka

- [1] Wibowo, B.C., dan Iqbal. M., 2013. Implementasi Metode Logika Fuzzy pada Kontrol Keseimbangan Robot Mobil Beroda Dua. Jurnal Universitas Muria. Kudus.
- [2] Pamungkas, A.P. Balancing Robot Beroda Dua Menggunakan Metode Fuzzy Logic Berbasis Mikrokontroler Arduino. Fakultas Teknik Elektro, Universitas Dian Nuswantoro.
- [3] Handry, K., dan Purwanto. D., Keseimbangan Robot Beroda Dua Menggunakan Metode Fuzzy Logic, *paper* Hal 1 – 10.
- [4] Royyan, M., 2015. *Implementation of Kalman Filter and PID Controller for Inverted Pendulum Robot*. Telkom University.
- [5] Putra, N.W., 2015. *Design And Implementation ADCS (Attitude Determination And Control System) For Nano Satellite Using Reaction Wheel*. Telkom University
- [6] <http://www.arduino.cc/en/Main/ArduinoBoardUno>