

## PERANCANGAN DAN IMPLEMENTASI *FUZZY LOGIC CONTROL* UNTUK PENGATURAN KESTABILAN GERAK PADA *TWO WHEELS SELF BALANCING ROBOT* BERBASIS ARDUINO UNO

**Indira Dwi Rachmawati**

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya  
e-mail: indirarachmawati@mhs.unesa.ac.id

**Puput Wanarti Rusimanto, Endryansyah, Muhammad Syariffuddien Zuhrie**

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya  
e-mail: puputwanarti@unesa.ac.id, endryansyah@gmail.com, zuhrie@unesa.ac.id

### Abstrak

*Two Wheels Self Balancing Robot* merupakan robot yang memiliki prinsip kerja seperti pendulum terbalik yang dapat mempertahankan keseimbangan robot dan tegak lurus terhadap permukaan bumi pada bidang datar. Pada *balancing robot* ini menggunakan pengendali *Fuzzy Logic Controller*. Penelitian ini menggunakan *hardware* MPU-6050 sebagai sensor, Arduino Uno R3 sebagai mikrokontroler, dan motor DC sebagai aktuator. Metode *Fuzzy Logic Control* digunakan untuk mengatur nilai pwm motor agar dapat bergerak maju dan mundur untuk menjaga keseimbangan. Motor bergerak maju dan mundur untuk mempertahankan posisi keseimbangan. Proses konversi nilai *accelerometer* dan *gyroscope* yang tidak stabil mempengaruhi kestabilan robot dalam mempertahankan keseimbangan. Performansi dari sistem *balancing robot* untuk pengaplikasian metode *fuzzy logic* dipengaruhi oleh beberapa parameter seperti, besar nilai dan bentuk dari fungsi keanggotaan masukan dan keluaran sistem, dan *rule-inference* yang ditanamkan pada sistem. Pengujian pada *balancing robot* dilakukan sebanyak empat kali percobaan dengan kemiringan 0° sampai 15°. Hasil penelitian ini menunjukkan bahwa sistem pengendalian *two wheels self balancing robot* berbasis *fuzzy logic controller* dapat menyeimbangkan dengan stabil pada kemiringan di bawah 15 derajat, di atas 15 derajat motor tidak dapat mengembalikan robot pada keadaan tegak berdiri.

**Kata Kunci:** *Two Wheels Self Balancing Robot*, *Fuzzy Logic Controller*, Arduino Uno R3, Sensor MPU-6050, Motor DC.

### Abstract

*Two Wheels Self Balancing Robot* is a robot that has a working principle such as an inverted pendulum that can maintain the robot's balance and is perpendicular to the surface of the earth on a flat plane. In balancing this robot using a *Fuzzy Logic Controller* controller. This research uses MPU-6050 hardware as a sensor, Arduino Uno R3 as a microcontroller, and a DC motor as an actuator. *Fuzzy Logic Control* method is used to adjust the motor PWM value so that it can move forward and backward to maintain balance. The motor moves forward and backward to maintain a balance position. The unstable *accelerometer* and *gyroscope* conversion process affects the stability of the robot in maintaining balance. The performance of the robot balancing system for the application of the *fuzzy logic* method is influenced by several parameters such as the value and shape of the input and output membership functions of the system, and the *rule-inference* implanted in the system. Testing on balancing robots is carried out four times with a slope of 0° to 15°. The results of this study indicate that the *two wheels self balancing robot* control system based on *fuzzy logic controller* can balance stably at slopes below 15 degrees, above 15 degrees the motor cannot return the robot to an upright standing.

**Keywords:** *Two Wheels Self Balancing Robot*, *Fuzzy Logic Controller*, Arduino Uno R3, MPU-6050 Sensor, Motor DC.

### PENDAHULUAN

Perkembangan teknologi robotika beberapa tahun belakangan ini telah membuat kualitas kehidupan manusia semakin tinggi. Saat ini perkembangan teknologi robotika telah mampu meningkatkan kualitas maupun kuantitas berbagai industri. Teknologi robotika juga telah menjangkau pada bidang pendidikan dan hiburan bagi manusia. Pada bidang pendidikan digunakan untuk pembelajaran misalnya perancangan Robot line follower, pengaplikasian lengan robot dan *Self Balancing robot*. Salah satu cara menambah tingkat kecerdasan sebuah robot adalah dengan menambah sensor, metode *control* bahkan memberikan kecerdasan buatan pada robot

tersebut. Salah satunya yang akan dibahas adalah *self balancing robot*.

*Self balancing robot* (robot penyeimbang) merupakan suatu robot yang memiliki dua buah roda di sisi kanan dan kirinya yang tidak akan seimbang apabila tanpa adanya kontroler. *Balancing robot* ini merupakan pengembangan dari model pendulum terbalik (*inverted pendulum*) yang diletakkan di atas kereta beroda. Menyeimbangkan robot beroda dua memerlukan suatu perangkat hardware yang baik dan metode kontrol handal untuk mempertahankan posisi robot dalam keadaan tegak lurus terhadap permukaan bumi. Dan konsep *self balancing robot* ini telah digunakan sebagai alat transportasi sekarang ini yang bernama *segway*.

Sistem *balancing robot* pertama dideklarasikan oleh Dean Kamen tahun 2001 dengan nama *segway* yang kemudian dikenal sebagai *The first Self-balancing, electric powered transportation device*, kemudian diikuti oleh penelitian-penelitian sejenis di berbagai negara. Dean Kamen melalui situsnya juga mengatakan bahwa alat transportasi personal yang beroda dua membutuhkan energi listrik yang lebih hemat dibanding beroda lebih dari dua (Gunawan, 2013:2).

Dalam bidang rekayasa kendali, *balancing robot* merupakan robot yang memiliki prinsip kerja yang hampir mirip dengan sistem pendulum terbalik. Pendulum terbalik sendiri merupakan suatu sistem yang memiliki tingkat ketidaksetimbangan tinggi, non-linier dan multivariabel, sehingga banyak dipakai dalam pengujian metode kendali. Kedepan, hasil *prototype balancing robot* yang digunakan dalam penelitian ini juga dapat digunakan untuk pengujian beberapa metode kendali (Gunawan, 2013:2).

Berbagai macam penelitian yang mendukung tentang perancangan dan implementasi *two wheels self balancing robot* telah dikembangkan dan dipublikasi melalui jurnal. Menurut Bimarta dkk (2015) menjelaskan bahwa 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*. Sedangkan Pada tahun 2017, Raranda melakukan penelitian dengan judul "Implementasi Kontroler PID pada *Two Wheels Self Balancing Robot* Berbasis Arduino UNO". Tujuannya untuk menerapkan kontroler PID pada balancing robot agar robot berdiri seimbang.

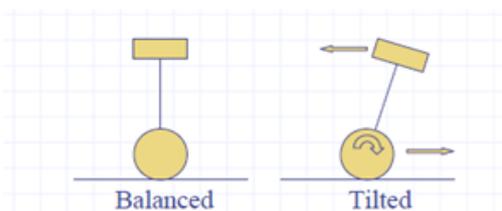
Oleh karena itu dibutuhkan metode kontrol lain untuk menyeimbangkan sistem dari *self balancing robot* beroda dua. Pada penelitian yang penulis buat ini menggunakan sistem kendali fuzzy untuk menyeimbangkan self balancing robot beroda dua. Tujuan penelitian ini adalah membuat dan merancang pengendalian self balancing pada robot beroda dua menggunakan *fuzzy logic controller* untuk menyeimbangkan robot agar seimbang tegak lurus terhadap permukaan datar. Penelitian ini menggunakan hardware MPU-6050 sebagai sensor, arduino sebagai mikrokontroler dan motor DC sebagai aktuator. Berdasarkan permasalahan tersebut penulis mengambil judul "Perancangan Dan Implementasi *Fuzzy Logic Control* Untuk Pengaturan Kestabilan Gerak Pada *Two Wheels Self Balancing Robot* Berbasis Arduino Uno".

## KAJIAN PUSTAKA

### *Inverted Pendulum*

Ide dasar untuk membuat *robot* beroda dua dapat setimbang adalah sangat mudah yaitu dengan cara mengendalikan roda searah dengan arah jatuhnya bagian atas sebuah *robot*. Apabila proses tersebut dapat terlaksana maka *robot* tersebut dapat setimbang. Secara praktis ini membutuhkan dua *sensor* sebagai umpan baliknya yaitu *sensor kemiringan* atau sudut terhadap gaya gravitasi dan *sensor encoder* untuk mengukur posisi robot. Pada Gambar 1. robot dalam keadaan seimbang, gaya yang

digunakan untuk menyeimbangkan robot dihasilkan dari putaran roda. Putaran roda ini diperoleh dari torsi yang dihasilkan oleh motor. Gaya horizontal yang diperoleh *balancing robot* akan sama dengan torsi roda dibagi dengan jari-jari roda. Torsi roda yang diperoleh pada penelitian ini dapat dihasilkan dari pengaturan kecepatan motor DC dengan menggunakan PWM.



Gambar 1. Menunjukkan Cara *Balancing Robot* Menyeimbangkan Diri  
(Sumber: Grace, 2015)

### Algoritma Fuzzy

Satu satu kemungkinan algoritma *fuzzy logic* adalah model mamdani yang memperkirakan karakteristik tertentu yang tersimpan didalam model. Model mamdani adalah salah satu dari algoritma utama dari teori *fuzzy*. Menjadi model dari karakteristik estimasi, yang digunakan dimana angka yang tersedia set input data adalah relatif kecil model sugeno adalah contoh lain dari algoritma *fuzzy logic*. Algoritma tersebut bisa digunakan jika angka dari set input data "cukup". Mekanisme dari *fuzzy inference system* (FIS) dideskripsikan dengan 4 tahap yaitu: *fuzzification, inference, aggregation, defuzzification*.

Tahap pertama (*fuzzification*), variabel input difuzzifikasikan. Jumlah dari *fuzzy set* telah ditentukan dan fungsi keanggotaan juga didefinisikan untuk setiap variabel input, berdasarkan dari total semua informasi yang tersedia.

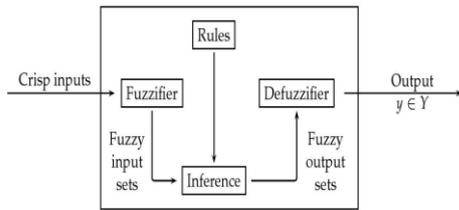
Tahap kedua (*inference*), variabel output juga difuzzifikasikan dalam cara analog, seperti variabel input pada tahap pertama. Tahap ini, *fuzzy rules* ditetapkan. Jumlah dan bentuk tergantung pada kualitas dari informasi yang tersedia. Ini merupakan tahap utama dalam algoritma *fuzzy*.

Pengaplikasian *rules* ini, nilai *crisp* dari semua variabel input untuk menentukan banyak nilai dari fungsi keanggotaan setiap variabel output, sesuai dengan aturan yang telah ditetapkan.

Tahap ketiga (*aggregation*) untuk setiap variabel, nilai dari fungsi keanggotaan yang telah didapatkan di tahap sebelumnya digabungkan sehingga, variabel output mendapat *fuzzy set* beserta fungsi keanggotaan yang telah didefinisikan.

Tahap keempat (*defuzzification*), defuzzifikasi adalah proses mengubah hasil linguistik ke nilai numerik. perubahan dari suatu besaran fuzzy ke suatu besaran numerik, sedangkan fuzzifikasi adalah perubahan dari besaran numerik ke suatu besaran fuzzy. Keluaran proses fuzzy dapat berupa satuan logika dari dua atau lebih fungsi keanggotaan fuzzy dan didefinisikan dalam himpunan semesta keluaran. Berikut Gambar 2. Sistem Fuzzy Logic

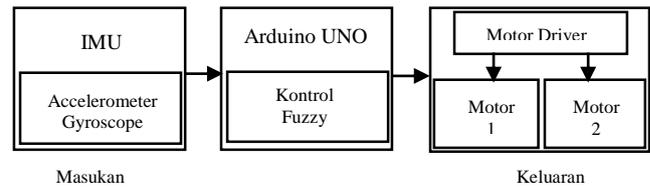
yang terdiri dari 4 tahap diantaranya *fuzzification*, *inference*, *aggregation*, *defuzzification*.



Gambar 2. Sistem *Fuzzy Logic*  
Sumber: (Kumar, 2015)

## 2. Desain Sistem

Desain sistem *Self Balancing Robot* menggunakan *Arduino* yang ditunjukkan pada Gambar 4.

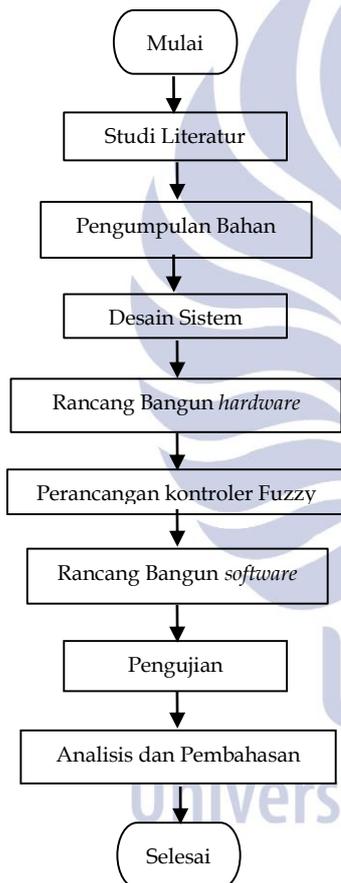


Gambar 4. Model Desain Sistem *Self Balancing Robot*

## METODE PENELITIAN

### Rancangan Penelitian

Tahapan perancangan penelitian ini secara garis besar di jelaskan dalam Gambar 3.



Gambar 3. *Flowchart* Rancangan Penelitian

### Prosedur Penelitian

Dari rancangan penelitian yang sudah dibuat sebelumnya, maka prosedur untuk masing-masing tahapan adalah sebagai berikut :

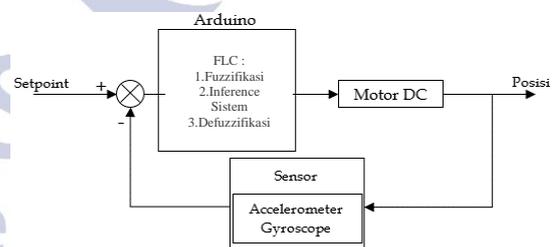
#### 1. Studi Literatur

Studi literatur dilaksanakan dengan mempelajari beberapa jurnal dan *text book* tentang *self balancing robot*, serta bahan referensi lain yang terkait dengan penelitian ini.

Pada Gambar 4. merupakan perancangan *self balancing robot* yang terdiri dari blok sistem yang terintegrasi 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. Sensor *IMU* yang digunakan yaitu *MPU-6050*. 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.

#### 3. Perancangan *Hardware*

Perancangan *Self Balancing Robot* menggunakan kontrol *fuzzy logic* yang ditunjukkan pada Gambar 5.



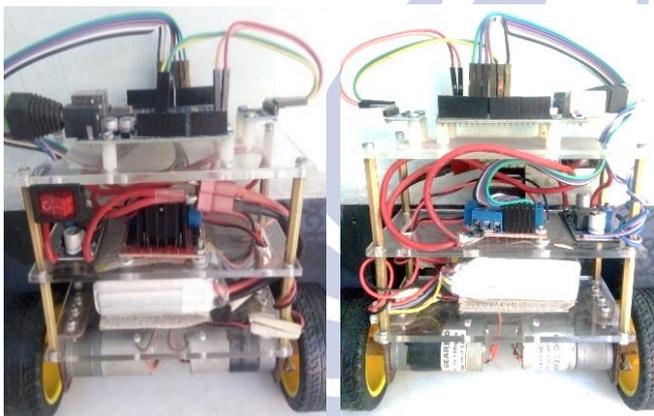
Gambar 5. Skema Sistem Pada *Self Balancing Robot*

Gambar 5 adalah struktur kerja dari perangkat *hardware*. Nilai masukan (*Setpoint*) sistem berupa nilai derajat yang diinginkan ialah  $0^\circ$ . Hasil dari keluaran kontroler berupa tegangan dalam bentuk *Pulse Width Modulation* (*PWM*) untuk mengatur arah putar dan kecepatan motor *DC*. Sensor *accelerometer* dan *gyroscope* akan bekerja untuk mengukur nilai dari derajat kemiringan robot (*Present Value*) dan kemudian menjadikan nilai pembacaan sebagai nilai umpan balik.

Tiap-tiap bagian dari diagram blok sistem di atas dapat dijelaskan sebagai berikut:

1. Arduino UNO berfungsi sebagai pusat pengendalian pada *self balancing robot* beroda dua.
2. Motor DC berfungsi sebagai aktuator yang mengikuti keseimbangan dari *self balancing robot* beroda dua.
3. IMU atau Sensor MPU-6050 Module digunakan untuk pembacaan sudut kemiringan dari *self balancing robot*.

Berikut Gambar 11. Di bawah adalah desain rancangan plant *Self Balancing Robot* beroda dua. Pembuatan mekanik terdiri dari 3 susun akrilik berukuran (P= 13 cm, L= 6 cm), 8 penyangga, dengan 2 roda disamping kanan dan kirinya. Rangka penopang terbuat dari Papan akrilik ketebalan 4 mm dan besi penyangga dengan panjang 4 cm. Robot ini memiliki ketinggian 15 cm dan lebar 19,5.



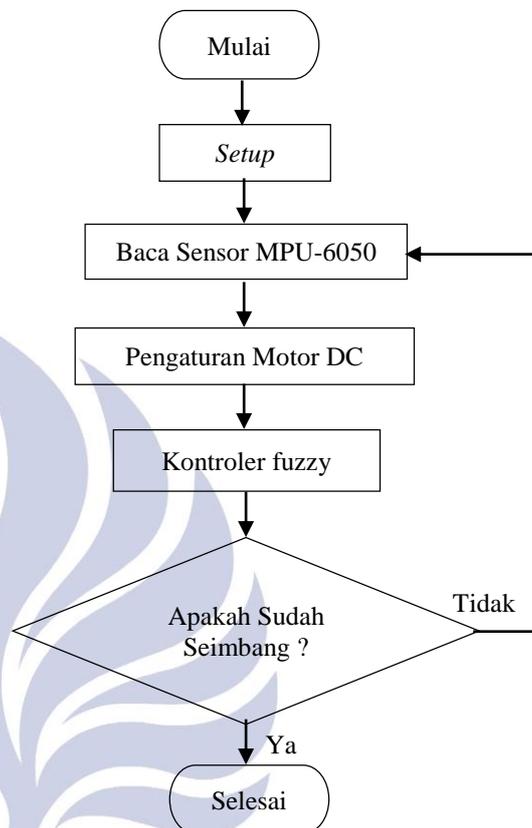
Gambar 11. Desain Mekanik *Two Whells Self Balancing Robot* yang digunakan untuk penelitian

**Perancangan Software**

Perancangan *software* pada *Two Wheels Self Balancing* disini adalah Proses memasukkan program ke dalam mikrokontroler. Pemrograman tersebut dibuat dengan bahasa pemrogram khusus oleh Arduino pada Arduino IDE (*Integrated Development Environment*) versi 1.6.12. Arduino sendiri merupakan bahasa turunan dari C++ sehingga fungsi-fungsi C++ dan C dapat berjalan di Arduino. Pemilihan bahasa Arduino ini dipilih karena untuk melengkapi perhitungan matematis dalam sistem *balancing robot* dan bersifat *open source*. Karena sifatnya yang *open source* ini, maka banyak *library* yang dikembangkan oleh personal atau komunitas di luar *developer* Arduino itu sendiri.

Secara garis besar jalannya program adalah membaca nilai kemiringan dari *balancing robot* dan membandingkannya dengan nilai referensi (*setpoint*), kemudian menentukan kecepatan putar pada motor DC. Kontroler fuzzy digunakan untuk mengontrol kecepatan putar motor DC melalui PWM berdasarkan perbandingan dari nilai sudut referensi yang dikehendaki dengan nilai

sudut yang terukur. Pemrograman dari Arduino akan mencakup deklarasi sensor, kendali fuzzy, kendali untuk motor DC. Untuk garis besar proses jalannya program ditunjukkan melalui diagram alir pada Gambar 6.



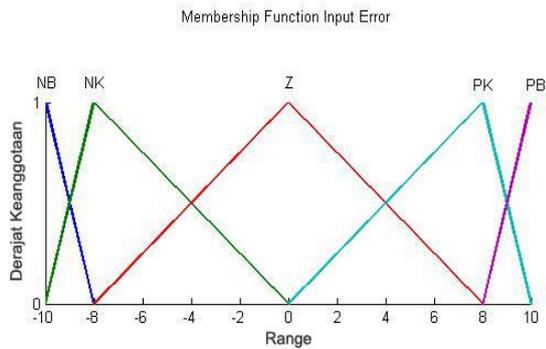
Gambar 6. Diagram Alir *Software*

**Perancangan Pengendali Logika Fuzzy**

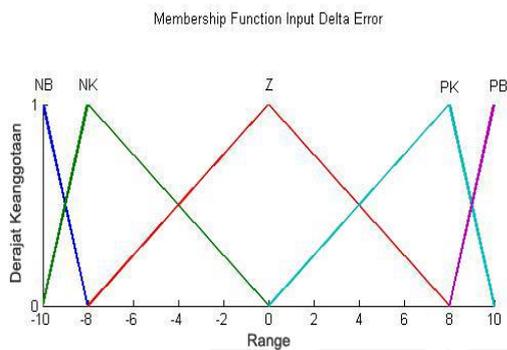
Perancangan pengendali logika fuzzy didasarkan pada pengetahuan dan pengalaman peneliti terhadap sistem. Untuk dapat merancang pengendali logika fuzzy, hal-hal yang harus diperhatikan adalah karakteristik sistem, masukan sistem, dan keluaran sistem.

Pada penelitian ini terdapat 3 variabel yang digunakan dalam pengendali *fuzzy logic*, terdiri dari 2 variabel input yaitu *Input Error* dan *Input Delta\_Error*, sedangkan untuk output terdapat 1 variabel yaitu variable PWM. Variabel *Input Error* dan *Input Delta\_Error* memiliki 5 linguistik, yaitu *Negative Besar* (NB), *Negative Kecil* (NK), *Zero* (Z), *Positive Kecil* (PK), dan *Positive Besar* (PB).

Berikut Gambar 7. Menunjukkan *membership function Input Error* dan Gambar 8. adalah *membership function Input Delta\_Error*.

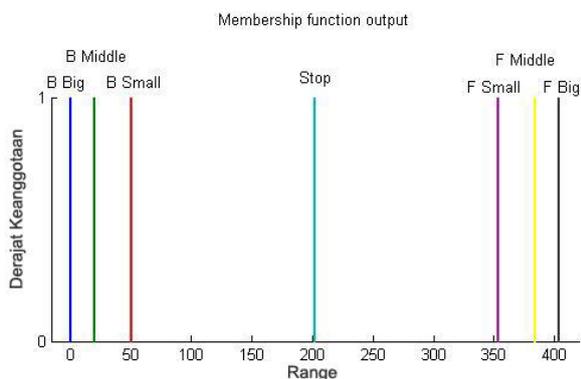


Gambar 7. *Membership function Input Error*



Gambar 8. *Membership function Input Delta\_Error*

Output dari pemodelan fuzzy adalah nilai PWM untuk pengaturan kecepatan motor. Nilai-nilai yang dimasukkan ke dalam perancangan *membership function output* untuk aplikasi pengaturan kecepatan motor dapat dilihat pada tabel dan gambar di bawah ini.



Gambar 9. *Membership function Output*

Pada Gambar 9. terdapat *Membership function output* tersebut dibagi menjadi 2 sisi yaitu sisi *backward* dan sisi *forward*, dan memiliki 7 Label *Membership Function* yaitu B\_Big, B\_Middle, B\_Small, Stop, F\_Small, F\_Middle dan F\_Big.

Berikut ini tabel *rule base self balancing robot* beroda dua dari *fuzzy logic* yang di desain.

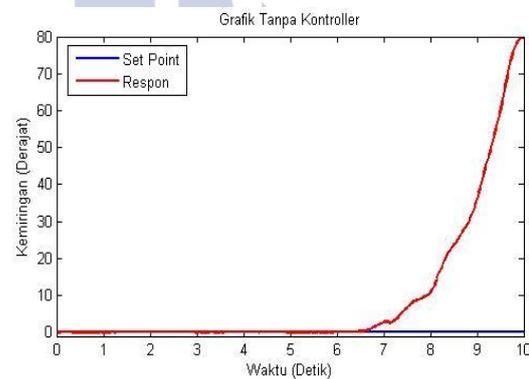
Tabel 1. Matrik *Rule Base Self Balancing Robot* Beroda Dua dengan Pengendali Logika Fuzzy

e/de	NB	NK	Z	PK	PB
NB	B_Big	B_Big	B_Middle	B_Small	Stop
NK	B_Big	B_Middle	B_Small	Stop	F_Small
Z	B_Middle	B_Small	Stop	F_Small	F_Middle
PK	B_Small	Stop	F_Small	F_Middle	F_Big
PB	Stop	F_Small	F_Middle	F_Big	F_Big

## HASIL DAN PEMBAHASAN

### Pengujian *Hardware Robot*

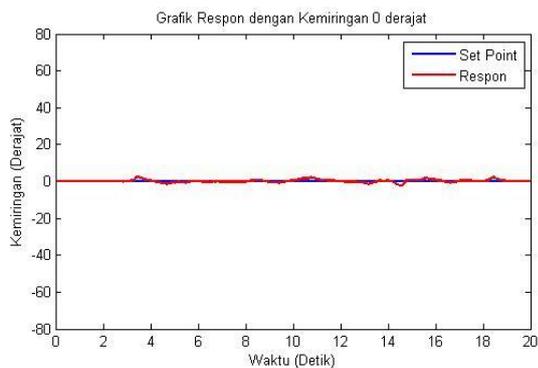
Pada pengujian *hardware robot* yang pertama adalah dilakukan pengujian untuk respon sistem robot sebelum diberikan kontroler.



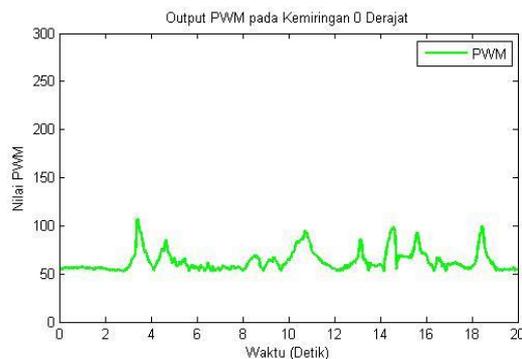
Gambar 10. Grafik Respon Sisten Tanpa Kontroler

Gambar 10. terlihat bahwa respon sistem dari balancing robot tidak bisa mengikuti set point yang diberikan. Respon sistem robot langsung menuju nilai tak hingga dan ini membuktikan bahwa balancing robot tidak bisa mempertahankan posisinya tegak lurus terhadap permukaan bumi sehingga perlu dirancang suatu kontroler yang dapat membuat robot mempertahankan posisinya sesuai *set point*.

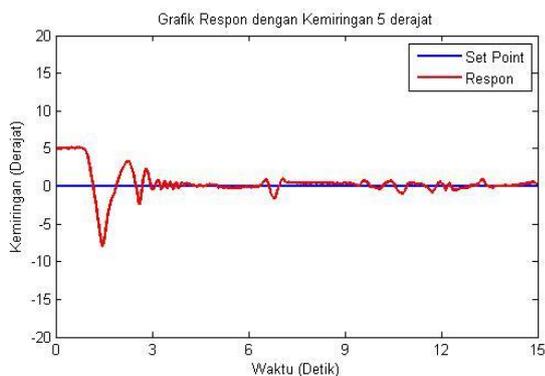
Pada pengujian *hardware robot* selanjutnya dilakukan dengan cara memberikan robot posisi awal kemudian dilihat apakah robot bisa mengatasi kemiringan tersebut dan dapat tetap berdiri dengan seimbang. Dalam pengujian *hardware robot* ini akan dilakukan 4 kali pengujian pada kondisi awal sudut yang berbeda – beda. Kondisi awal pada pengujian meliputi 0 derajat, 5 derajat, 10 derajat, dan 15 derajat. Berikut adalah beberapa grafik respon dari *hardware robot* pada beberapa percobaan kondisi awal kemiringan robot :



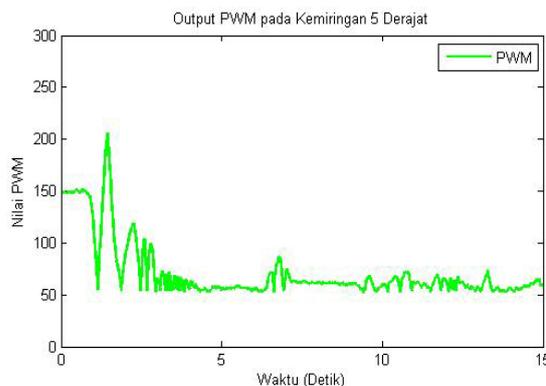
Gambar 11. Grafik Respon dengan Kemiringan 0 derajat



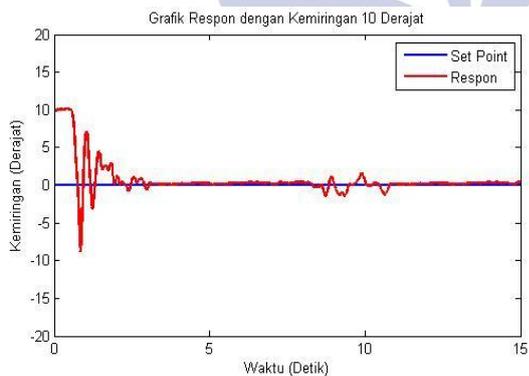
Gambar 15. Grafik Output pada Kemiringan 0 derajat



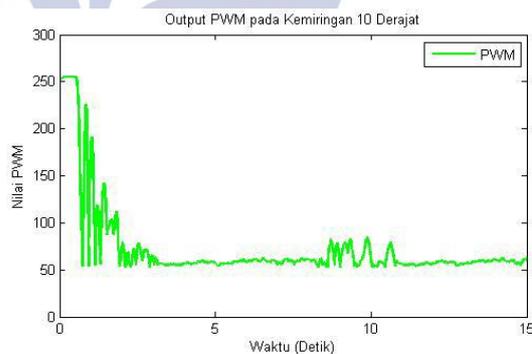
Gambar 12. Grafik Respon dengan Kemiringan 5 derajat



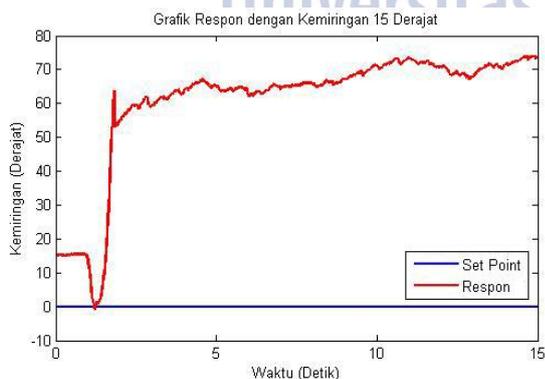
Gambar 16. Grafik Output pada Kemiringan 5 derajat



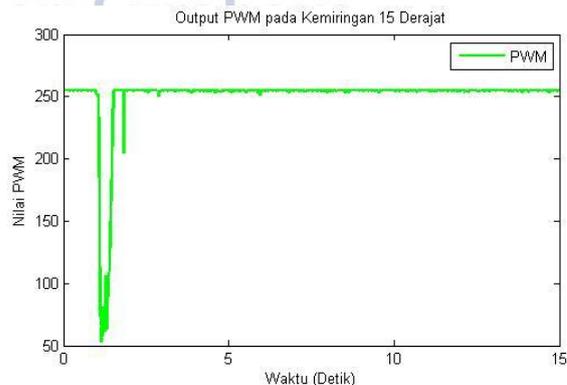
Gambar 13. Grafik Respon dengan Kemiringan 10 derajat



Gambar 17. Grafik Output pada Kemiringan 10 derajat



Gambar 14. Grafik Respon dengan Kemiringan 15 derajat



Gambar 18. Grafik Output pada Kemiringan 15 derajat

Dari percobaan pengujian dengan beberapa kondisi awal tersebut dapat diketahui bahwa robot melakukan *self balancing control* dengan baik ketika kondisi awal yaitu 0 derajat, 5 derajat, dan 10 derajat. Namun ketika *balancing robot* diberikan kondisi awal 15 derajat robot langsung terjatuh dan tidak dapat melakukan *self balancing control*.

Respon posisi robot ketika diberikan gangguan ditunjukkan pada gambar grafik diatas yaitu Gambar 11, Gambar 12, dan Gambar 13. Pada gambar tersebut terlihat robot dapat mempertahankan posisinya namun robot terjatuh ketika diberikan kondisi awal 15 derajat yang grafik respon posisi robot ditunjukkan pada Gambar 14.

Pada kondisi awal kemiringan 0 derajat yang ditunjukkan pada Gambar 11 terlihat bahwa robot dapat mempertahankan posisinya dengan grafik yang relatif stabil yang mendekati *set point*. Dan pada kondisi awal kemiringan 5 derajat dan 10 derajat yang ditunjukkan pada Gambar 12 dan Gambar 13 robot mulai dapat mempertahankan diri pada waktu 3 sampai 5 detik. Kemudian untuk grafik respon Gambar 14 dengan kondisi awal 15 derajat robot tidak dapat mempertahankan posisinya dan langsung terjatuh.

Sedangkan pada Gambar 15, Gambar 16, Gambar 17, dan Gambar 18 adalah *output* pengendalian Fuzzy untuk mengendalikan sudut kemiringan sasis robot. Terlihat *output* pengendalian Fuzzy menghasilkan keluaran yang tidak beraturan dengan rentang waktu naik dan turun yang cepat karena pengendalian sudut kemiringan sasis robot harus agresif agar robot dapat berdiri tegak dengan seimbang. Berdasarkan grafik diatas nilai *Error steady state* yang diperoleh pada kemiringan 0 derajat =0.0254, 5 derajat= 0.0142, dan 10 derajat= 0.0042.

## PENUTUP

### Simpulan

Berdasarkan penelitian dan pengujian yang dilakukan, maka didapatkan simpulan sistem *two wheels self balancing robot* berbasis *fuzzy logic controller* berhasil dibuat dan bekerja dengan baik. Penerapan pengendali logika fuzzy yang telah dirancang memiliki aturan basis 5x5 menghasilkan respon yang mampu menyeimbangkan sistem *two wheels self balancing robot*. Kontroler Fuzzy yang dirancang untuk mengendalikan keseimbangan robot, berhasil mencapai nilai *setpoint* dengan nilai *Ess (Error Steady State)* pada kemiringan 0 derajat adalah 0.0254. Dari hasil pengujian hardware robot bisa mengatasi kemiringan 0 derajat, 5 derajat, dan 10 derajat. Namun robot jatuh ketika kemiringan 15 derajat.

### Saran

Berdasarkan hasil penelitian yang telah dilakukan terdapat beberapa aspek yang dapat diperbaiki dan dikembangkan agar memberikan hasil yang lebih maksimal. Perbaikan dapat dilakukan dengan menggunakan *hardware* yang lebih baik dari *hardware* penulis atau dengan ditambah Bluetooth agar mudah dalam pengujian datanya. Untuk penelitian selanjutnya sistem ini masih dapat dikembangkan dengan menggunakan metode kontroler yang lain seperti *PID-Fuzzy*, *Model Predictive Control*, dan yang lainnya. Dan yang terakhir *self balancing robot* dapat

dikembangkan untuk melakukan *tracking* menggunakan *remote control* secara *wireless* atau dapat digunakan untuk aplikasi *line follower*.

## DAFTAR PUSTAKA

- Arduino. 2017. "Tech Specs Arduino UNO R3".(Online), (<https://store.arduino.cc/arduino-uno-rev3>, diunduh 17 April 2017).
- Bimarta, Rizka, Putra, A. E., & Dharmawan, A. 2015. Balancing Robot Menggunakan Metode Kendali Proporsional Integral Derivatif. *IJEIS-Indonesian Journal of Electronics and Instrumentation Systems*, 5(1), 89-98.
- Bobby, Grace, Susanto, E., & Suratman, F.Y. 2015. Implementasi Robot Keseimbangan Beroda Dua Berbasis Mikrokontroler. *Jurnal Elkomika*, 3(2).
- Firmansyah, Rifqi, Shafiudin, Sofyan. 2016. Pemantauan Ruang Inkubator penetasan telur ayam dengan berbasis telemetri menggunakan Arduino Uno R3. Padang. Universitas Andalas.
- Gunawan, A. P., Subagiyo, H., & Wahyuni, R. T. 2013. Kontrol Kesetimbangan pada Robot Beroda Dua Menggunakan Pengendali PID dan Complementary Filter. Riau: Politeknik Caltex Riau.
- Gutama, B. D. 2016. Desain Dan Implementasi Kontrol Keseimbangan Kendaraan Roda Dua Dengan Kontrol Logika Samar Menggunakan Mikrokontroler. *Teknika*, 1(1).
- Khoswanto, Handry, dkk. 2013. *Semi-Autonomous Telepresence Robot*. Skripsi diterbitkan. Surabaya: Universitas Kristen Petra.
- Kumar, Lakshya; Rawat, Tejender Singh.; Pandey, Manoj; Kumar, Upendra. 2015. "Automatic Control of Fan Speed using Fuzzy Logic", *International Journal of Engineering Technology, Management and Applied Sciences* Vol 3, issue 4.
- Raranda. 2017. Implementasi Kontroler PID Pada *two wheels self balancing Robot* Berbasis Arduino Uno. Skripsi diterbitkan. Surabaya: Universitas Negeri Surabaya.
- Royyan, M. 2015. Implementation of Kalman Filter and PID Controller for Inverted Pendulum Robot. Bandung: *Telkom University*.
- Wardana, I Nyoman Kusuma. 2015. Teknik Antarmuka MATLAB dan Arduino. Denpasar: Vaikutha International Publication.