

Implementasi Sensor MPU 6050 untuk Mengukur Kesetimbangan *Self Balancing Robot* Menggunakan Kontrol PID

Oktaf Brillian Kharisma¹, Ahmad Wildan², Auliaullah³, Folkes E. Laumal⁴

^{1,2,3}Jurusan Teknik elektro UIN Sultan Syarif Kasim Riau

⁴Teknik elektro Politeknik Negeri Kupang, NTT

e-mail : brilliankhar@gmail.com, willhasibuan07@gmail.com

Abstrak

Robot keseimbangan merupakan robot mobile yang tidak akan seimbang tanpa adanya pengontrolan yang baik. Robot akan bergerak maju mundur untuk mendapatkan keseimbangan. Supaya robot mampu seimbang dibutuhkan sensor keseimbangan yang memiliki sensitivitas yang bagus. Sensor MPU 6050 merupakan sensor keseimbangan yang sudah dilengkapi dengan sensor Accelerometer dan Gyroscope yang mampu membaca sudut kemiringan objek dengan cakupan tiga dimensi. Algoritma PID akan membantu sensor dalam menyeimbangkan robot dimana komponen Kp, Ki dan Kd secara real time mengirim sinyal kepada sensor. Saat sensor MPU diujicoba pada bidang datar nilai sumbu z stabil pada rentang angka 1.47 sampai -1.24, sedangkan sumbu x dan y mengalami fase naik turun pada sudut kemiringan 60° dan -60°. Pada bidang miring nilai sumbu y stabil pada rentang angka 1.69 sampai -1.76, sedangkan sumbu x dan z mengalami kenaikan disaat sensor diarahkan menuju sudut 90°. Pada saat mengarah ke sudut -90° sumbu z mengalami penurunan sedangkan sumbu x mengalami fase naik turun. Nilai manual tuning ideal pada robot ini adalah Kp 70, Kd 250 dan Ki 1.9.

Kata kunci: robot keseimbangan, MPU 6050, manual tuning, algoritma PID, sudut kemiringan

Abstract

The balancing robot is a mobile robot's that will not be balanced without a control. The robot will move back and forth to get a balance. It is Needed a good sensitivity sensor to make it balance such as MPU 6050 Sensor is balancing sensor which is equipped with an Accelerometer and Gyroscope sensor that is capable of reading the angle of slope of the objects with three-dimensional coverage. PID algorithms would help in balancing robot sensor by sending a signal of the components of the Kp, Ki, and Kd in real time to the sensor. When the MPU6050 sensor was tested on a flat plane the z-axis value will be stable in the range of numbers 1.47 to -1.24, while the x and y-axis are experiencing a phase of up and down at an angle of slope 60° and -60°. On the inclined plane, the value of y-axis steady at 1.69 up to -1.76 range, while the x and z-axis will be increased when the sensor is directed towards angles of 90°. At the pointing to the angle of the -90°, z-axis has decreased while x-axis is experiencing a phase of up and down. Normally, Manual tuning value of the robot is Kp 70 Kd 250 and Ki 1.9.

Keywords: robot balance, MPU 6050, manual tuning, PID algorithm, the angle of the slope

1. Pendahuluan

Sistem robot kesetimbangan 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. Dalam bidang rekayasa kendali robot kesetimbangan merupakan robot yang memiliki prinsip kerja yang hampir mirip dengan sistem pendulum terbalik. Bahkan sekarang ini konsep robot kesetimbangan beroda dua telah digunakan sebagai alat transportasi yang bernama Segway[1].

Self Balancing Robot (robot penyeimbang) beroda dua merupakan suatu robot mobile yang tidak akan seimbang apabila tanpa adanya kontroler yang baik. Balancing robot ini merupakan pengembangan dari model pendulum terbalik (*inverted pendulum*) yang diletakkan di atas kereta beroda. Pada proses kerjanya, robot akan berusaha mengimbangkan dirinya dengan cara maju atau mundur untuk mendapat titik seimbang yang pas. Pada penelitian ini akan digunakan mikrokontroler Arduino Mega 2560, sensor 6050 dan motor driver L298[2],[3].

Hasil deteksi kemiringan yang di keluarkan oleh sensor MPU 6050 masih memiliki gangguan atau noise sehingga dibutuhkan filter digital yang mampu memperbaiki keluaran dari

sensor tersebut. Pada penelitian ini kekurangan tersebut akan ditutupi menggunakan kendali algoritma yang terdiri dari low-pass filter, integral, dan high-pass filter. Metode algoritma tersebut adalah kendali Proporsional Integral Derivatif (PID). Kontroler PID merupakan gabungan dari tiga macam kendali, yaitu kendali proporsional (*proportional controller*), kendali integral (*integral controller*), dan kendali turunan (*derivative controller*)[6].

Tuning kontrol PID ini bertujuan untuk menentukan parameter aksi kontrol agar robot bisa bergiri tegak. Proses ini dapat dilakukan dengan cara *trial and error*. Adapun nilai yang diberikan berupa konstanta P-I-D pada formula PID hingga di peroleh hasil yang di inginkan, dengan mengacu pada karakteristik pada kontroler PID. Penggunaan kontroler PID bertujuan untuk mengolah suatu sinyal kesalahan atau *error*, nilai *error* tersebut diolah dengan formula PID untuk dijadikan suatu sinyal kontrol yang akan diteruskan ke aktuator[7].

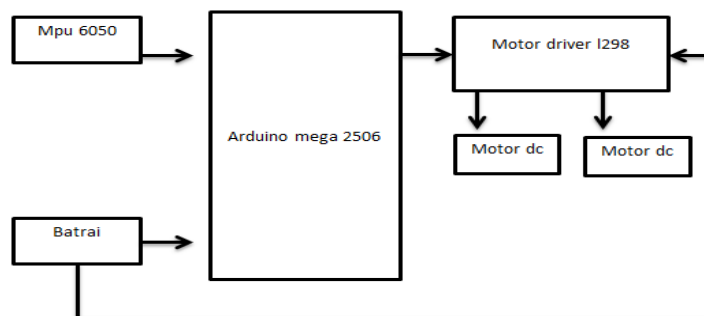
Arduino Mega 2560 adalah papan pengembangan mikrokontroler yang berbasis Arduino dengan menggunakan *chip ATmega2560*. *Board* ini memiliki pin I/O yang cukup banyak, sejumlah 54 buah digital I/O pin (15 pin diantaranya adalah *PWM*), 16 pin analog input, 4 pin UART (*serial port hardware*). *Arduino Mega 2560* dilengkapi dengan sebuah *oscillator 16 Mhz*, sebuah *port USB*, *power jack DC*, *ICSP header*, dan tombol reset. *Board* ini sudah sangat lengkap, sudah memiliki segala sesuatu yang dibutuhkan untuk sebuah mikrokontroler. Dengan penggunaan yang cukup sederhana tinggal menghubungkan *power* dari USB ke PC anda atau melalui *adaptor AC/DC* ke *jack DC* [4].

Sensor MPU6050 adalah sensor yang mampu membaca kemiringan sudut berdasarkan data dari sensor *accelerometer* dan *sensorgyroscop*. Sensor ini juga dilengkapi oleh sensor suhu yang dapat digunakan untuk mengukur suhu di keadaan sekitar. Jalur data yang digunakan pada sensor ini adalah jalur data I2C dan mampu berjalan pada tegangan power supply sebesar 3 – 5v [5].

2. Perancangan

2.1 Perancangan Hardware

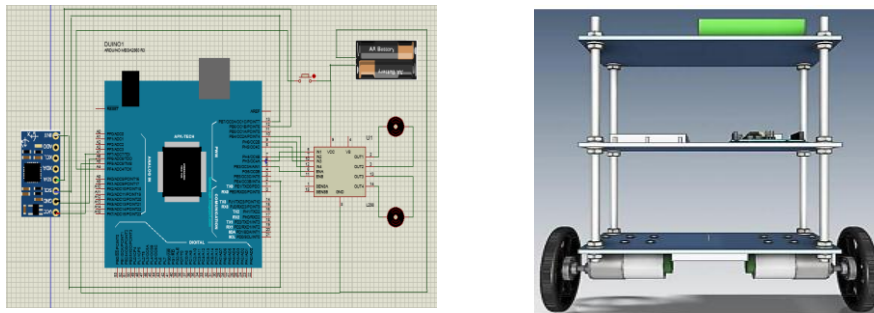
Secara garis besar robot kesetimbangan ini terdiri dari power supply, sensor mpu 6050, mikrokontroler Arduino Mega 2560, Motor Driver L298N, Motor Dc, dan roda. Berikut adalah blog diagram robot kesetimbangan:



Gambar 1. Blog diagram robot

Dari blog diagram diatas dapat disimpulkan cara kerja dari robot, dimana Mpu 6050 merupakan input bagi robot yang mengatur keseimbangan robot, mpu 6050 ini memiliki sensor *Accelerometer* dan *Gyroscope* yang sudah terintegrasi. Selain itu sensor robot dibantu dengan logika PID yang secara *runtime* melakukan pengecekan error dari sensor. Arduino mega 2560 merupakan pusat pengendali dari semua system yang ada di dalam robot ini. Motor driver I298 merupakan pengatur motor dc yang merupakan penggerak dari robot dan terhubung dengan Arduino. Kecepatan motor dc akan diatur didalam Arduino dan hasil dari sensor akan menentukan arah gerak motor dc. Disaat posisi robot miring ke arah depan maka motor akan bergerak kedepan dan begitu sebaliknya dan robot berada pada titik seimbang saat berada pada titik sudut 0^0 .

Berikut adalah rancangan keseluruhan robot dan arsitektur mekanik robot:

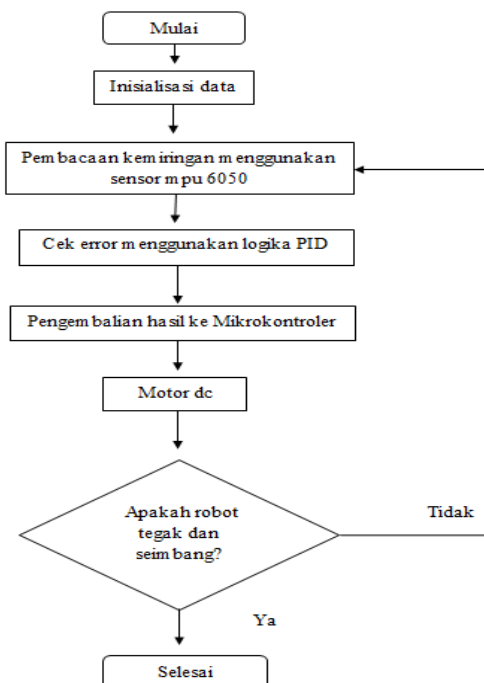


Gambar 2. Rancangan Schematik dan Mekanik Robot [3]

Perancangan mekanik dari robot ini menggunakan bentuk kotak (box) berbahan plastik atau akrilik dengan ketebalan 3mm. Dimensi robot yang digunakan dengan panjang sebesar 23 cm, lebar sebesar 8 cm dan tinggi sebesar 13 cm. Berikut adalah gambar dari perancangan mekanik tersebut.

2.2 Perancangan Software

Untuk membuat program yang baik dan sesuai dengan tujuan penelitian, maka terlebih dahulu dibuat rancangan program dalam suatu diagram alir (*flow chart*) dari program yang dibuat agar alat bekerja berdasarkan program yang diinginkan. Berikut adalah diagram alir dari penelitian ini:



Gambar 3. Flowchart robot

Flowchart diatas diawali dengan Inisialisasi data yang merupakan langkah pertama dilakukan robot setelah daya masuk kedalam Mikrokontroler. Inisialisasi data ini bertujuan untuk menentukan set point dari robot dan toleransi error yang dapat diberikan. Setelah penginisialisasian selesai, sensor mpu 6050 akan membaca kemiringan robot. Sensor akan menguji kemiringan robot dengan menggunakan metode x,y dan z. Robot berada pada titik seimbang pada posisi y, disaat posisi x dan z robot mengalami kemiringan. Disaat robot berada dalam keadaan miring maka sensor akan memberikan intruksi atau berupa output kepada Arduino. Setelah sensor membaca kemiringan robot maka hasil yang dikeluarkan akan diolah menggunakan logika PID. PID ini sendiri berguna untuk melakukan pengecekan eror secara

runtime kemudian akan melakukan perbaikan terhadap eror tersebut. Hasil pembacaan tersebut akan dikirim kepada mikrokontroller untuk mengolah hasil sensor dan mengkonversikannya kepada motor driver L298. Selanjutnya motor driver L298 akan menggerakkan motor dc sesuai hasil yang dikirim oleh mikrokontroller. Dan motor dc akan bergerak maju atau mundur sesuai dengan arah yang di tentukan mikrokontroller. Disaat robot belum mendapat posisi seimbang atau masih dalam posisi miring maka proses akan diulang kembali dari pembacaan kemiringan robot. Proses ini akan terus berulang – ulang selama daya masuk kedalam mikrokontroler.

3. Hasil dan Pembahasan

Setelah semua rangkain software dan hardware selesai maka langkah selanjutnya adalah melakukan uji coba terhadap robot yang sudah dibentuk. Pada uji coba kali ini akan dilakukan dua uji coba pada bidang yang berbeda yaitu, bidang datar dan bidang miring.

3.1 Ujicoba di Bidang datar

Pada percobaan pertama ini dilakukan uji coba dengan memulai posisi sudut robot 0° , 10° , -10° , 20° , -20° , 30° , -30° , 45° , -45° , 60° dan -60° . Pengujian ini dilakukan diatas bidang datar dan sumber tegangan yang diberikan adalah 9 volt. Dan hasil yang didapat setelah uji coba ini adalah:

Tabel 1. Ujicoba di bidang datar

No	Posisi sudut	Keadaan robot
1	-60°	Tidak Seimbang
2	-45°	Tidak Seimbang
3	-30°	Tidak Seimbang
4	-20°	Seimbang
5	-15°	Seimbang
6	-10°	Seimbang
7	0	Seimbang
8	10°	Seimbang
9	15°	Seimbang
10	20°	Seimbang
11	30°	Tidak Seimbang
12	45°	Tidak Seimbang
13	60°	Tidak Seimbang



Gambar 4. Posisi normal

Dari tabel diatas dapat dilihat bahwa *robot balancing* ini mampu berdiri tegak dan baik dibidang datar saat dimulai dari posisi sudut awal 0° , 10° , -10° , 20° , -20° . Tapi robot ini gagal mengimbangkan dirinya sendiri pada posisi sudut awal 30° keatas dan -30° kebawah. Berikut tampilan robot saat di uji coba di bidang datar:

3.2 Ujicoba di Bidang Miring

Pada percobaan kedua, uji coba dilakukan pada bidang dengan kemiringan dari rentang sudut -60° sampai sudut 60° dengan menggunakan acuan variable z, dengan sumber tegangan yang diberikan adalah 9 volt. Berikut adalah hasil yang didapat dari percobaan ke dua ini:

Tabel 2. Ujicoba di bidang miring

No	Posisi sudut	Keadaan robot
1	-60°	Tidak Seimbang
2	-45°	Tidak Seimbang
3	-30°	Seimbang
4	-20°	Seimbang
5	-15°	Seimbang
6	-10°	Seimbang
7	0	Seimbang
8	10°	Seimbang
9	15°	Seimbang
10	20°	Seimbang
11	30°	Seimbang
12	45°	Tidak Seimbang
13	60°	Tidak Seimbang



Gambar 5. Posisi di bidang miring

Dari tabel diatas dapat dilihat hasil yang didapat pada uji coba kedua ini hampir sama dengan uji coba pertama. Robot mampu seimbang pada bidang kemiringan dari sudut -30° sampai pada sudut 30°. Dan robot gagal menyeimbangkan dirinya pada bidang dengan kemiringan -45° kebawah dan 45° keatas. Dan berikut adalah foto uji coba di bidang miring.

3.3 Ujicoba Sensor Mpu 6050

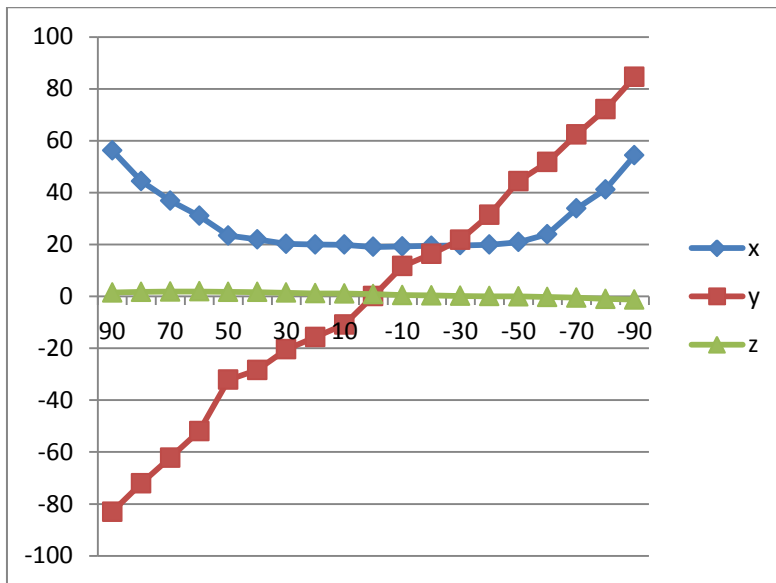
Pada uji coba sensor mpu ini dilakukan dua kali uji coba yaitu dibidang datar dan juga di bidang miring. Pada Uji coba dibidang datar ini dilakukan pada rentang 90° sampai -90° dimana nilai kemiringan sensor mpu ditampilkan ke lcd dan hasil yang di dapatkan adalah sebagai berikut:

3.3.1 Ujicoba di Bidang Datar

Hasil dari uji coba ini menjelaskan bahwa nilai dari sumbu z cukup stabil. Hal ini terjadi karena uji coba dilakukan di bidang datar. Sedangkan nilai sumbu x dan sumbu y mengalami kenaikan sesuai dengan arah kemiringan sensor. Disaat posisi kemiringan mpu berada pada posisi 60 atau -60 derajat maka nilai yang di tampilkan akan meningkat sepuluh poin. Berikut adalah tabel dan nilai grafiknya:

Tabel 3. Pengujian mpu dibidang datar

No	Sudut MPU	X	Y	Z
1	90	56,25	-83,03	1,47
2	80	44,41	-72,06	1,69
3	70	36,86	-62,21	1,84
4	60	30,99	-51,97	1,88
5	50	23,44	-32,15	1,73
6	40	21,95	-28,4	1,65
7	30	20,22	-20,45	1,42
8	20	19,99	-15,71	1,18
9	10	19,89	-10,86	1,07
10	0	19,07	0,1	0,86
11	-10	19,24	11,62	0,5
12	-20	19,5	16,32	0,33
13	-30	19,68	21,81	0,16
14	-40	19,88	31,45	0,03
15	-50	20,91	44,37	-0,01
16	-60	23,93	51,71	-0,28
17	-70	33,87	62,39	-0,48
18	-80	41,18	72,07	-0,98
19	-90	54,35	84,55	-1,24



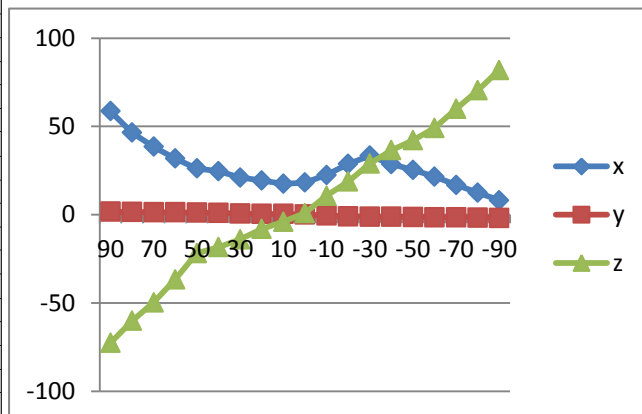
Gambar 6. Grafik Pengujian Sensor MPU 6050 dibidang datar

3.3.2 Ujicoba di Bidang Miring

Berikut adalah tabel dan nilai grafik dari hasil uji coba dibidang miring:

Tabel 4. Pengujian di bidang miring

No	Sudut mpu (°)	X	Y	z
1	90	58.65	1.69	-72.65
2	80	46.44	1.57	-60.10
3	70	38.53	1.48	-49.72
4	60	31.79	1.35	-36.60
5	50	26.21	1.15	-21.73
6	45	24.64	1.06	-18.40
7	30	20.99	0.64	-13.93
8	20	19.41	0.47	-8.10
9	10	17.50	0.45	-4.01
10	0	18.24	0.10	0.86
11	-10	22.45	-0.42	10.70
12	-20	28.70	-1.04	18.72
13	-30	33.43	-1.10	28.91
14	-45	28.67	-1.22	36.51
15	-50	25.34	-1.28	42.11
16	-60	21.56	-1.42	49.07
17	-70	16.80	-1.51	59.86
18	-80	12.51	-1.64	70.45
19	-90	8.13	-1.79	81.94



Gambar 7. Grafik Pengujian dibidang miring

Dari tabel diatas dapat diambil kesimpulan nilai dari sumbu z akan semakin meningkat seiring dengan arah kemiringan sensor. Sedangkan nilai sumbu y stabil, ini dikarenakan posisi sensor diputar ke arah sumbu z. Sedangkan nilai pada sumbu x mengalami kenaikan pada saat diarahkan menuju sudut 90 derajat dan mengalami fase naik turun saat diarahkan ke sudut -90 derajat.

3.3.3 Uji Coba Manual Tuning PID

Setelah uji cobaterhadap robot dan sensor mpu 6050 telah selesai, maka langkah selanjutnya adalah melakukan uji coba kepada *manual tuning* dari logika PID yang digunakan. Uji coba PID kali ini dilakukan dengan mengubah-ubah angka dari Kp, Kd, Ki dari program. Nilai seimbang pada robot ini dari masing variable adalah Kp = 70, Kd = 250 dan Ki = 1.9. Berikut adalah hasil uji coba terhadap *manual tuning* dari PID:

Tabel 5. Uji coba manual tuning PID

No	Kp	Kd	Ki	Keadaan Robot
1	0	0	0	Motor dc tidak bergerak
2	50	0	0	Tidak seimbang tapi motor dc bergerak karena pada percobaan ini hanya Kp yang diberi nilai, dimana nilai Kp berbanding lurus dengan respon sensor
3	0	250	0	Tidak seimbang tapi motor dc bergerak karena pada percobaan ini hanya Kd yang diberi nilai.
4	0	0	1.9	Tidak seimbang tapi motor dc bergerak karena pada percobaan ini hanya Ki yang diberi nilai, dimana nilai Ki merupakan nilai toleransi error dari sensor
5	50	200	1.5	Seimbang tapi respon dari sensor agak lambat
6	70	240	1.9	Seimbang dan stabil
7	100	250	2.0	Seimbang tapi sedikit bergoncang
8	150	350	2.5	Seimbang dan bergoncang

Dari tabel diatas dapat diambil analisa, ketika angka dari Kp, Kd, dan Ki adalah 0 maka motor dc tidak akan bergerak. Disaat salah satu dari ketiga variabel ada yang bernilai 0 maka robot tidak akan seimbang. Posisi keseimbangan ideal robot adalah $Kp = 70$, $Kd = 250$ dan $Ki = 1.9$. Disaat nilai Kp berada diangka 60 kebawah, nilai Kd 200 kebawah dan Ki 1.5 kebawah maka respon sensor agak lambat. Sebaliknya disaat Kp 100 keatas Kd 300 keatas dan Ki 2.5 keatas maka robot akan bergoncang. Tinggi dan bobot dari robot akan berbanding lurus dengan nilai dari manual tuning PID, semakin berat dan tinggi robot maka nilai ideal dari Kp, Kd dan Ki juga semakin bertambah.

4. Kesimpulan

Robotkesetimbangan ini mampu berdiri tegak dan baik saat dimulai dari posisi sudut awal 0° , 10° , -10° , 20° , -20° . Tapi robot ini gagal mengimbangkan dirinya sendiri pada posisi sudut awal 30° keatas dan -30° kebawah. Dimana sumber tegangan yang diberikan adalah 9 volt dan ujicoba ini dilakukan dibidang datar. Robot ini juga mampu berdiri tegak pada bidang miring dari rentang posisi sudut 30° dan -30° . Pada saat percobaan sensor MPU dibidang datar nilai dari sumbu z stabil sedangkan nilai dari sumbu x dan y mengalami peningkatan sesuai dengan kemiringan dari sensor. Saat uji coba pada bidang miring nilai dari sumbu y stabil, sedangkan nilai dari sumbu x mengalami kenaikan pada kemiringan 0 sampai 90 derajat, sedangkan pada kemiringan 0 sampai -90 derajat nilai sumbu x mengalami fase naik turun. Dan nilai z mengalami kenaikan dari sudut -90 sampai pada 90 derajat.

Saat nilai Kp, Kd dan Ki adalah 0 motor dc tidak akan bergerak. Disaat salah satu dari ketiganya bernilai 0, maka robot tidak mendapatkan keseimbangan. Nilai ideal pada keseimbangan robot ini adalah $Kp = 70$, $Kd = 250$ dan $Ki = 1.9$. Pada saat nilai dari ketiga variable berada dibawah nilai ideal, maka respon dari sensor sedikit melambat, sebaliknya saat nilai diatas nilai ideal maka robot mengalami guncangan. Bobot dari robot akan berbanding lurus dengan manual tuning PID, saat bobot robot cukup berat dan cukup tinggi maka nilai ideal dari manual tuning PID bisa berubah.

Daftar Pustaka

- [1] A. P. Gunawan, H. Subagiyo, and R. T. Wahyuni, "Kontrol Kesetimbangan pada Robot Beroda Dua Menggunakan Pengendali PID dan Complementary Filter," p. 11.
- [2] G. Bobby, E. Susanto, and F. Y. Suratman, "Implementasi Robot Keseimbangan Beroda Dua Berbasis Mikrokontroler," *J. Elkomika*, vol. 3, no. 2, Feb. 2016.
- [3] Nazri, Muhammad. (2016). "Perancangan Robot Penyeimbang Menggunakan Sensor Jarak Berbasis Android". Universitas Potensi Utama. Medan
- [4] <http://fritzing.org/projects/working-with-l298n-dc-motor-driver>. Diakses pada tanggal 08 September 2017. Diakses pada tanggal 08 September 2017.

- [5] <http://prasetyoabdi.blogspot.co.id/2014/12/menggunakan-sensor-mpu6050-pada.html>. Diakses pada tanggal 08 September 2017.
- [6] S.N.Cahya, Wahyudi, and Sumardi. (2015). "Perancangan Self-Balancing Robot Menggunakan Logika Fuzzy untuk Tuning Parameter Kendali Proporsional Integral Derivatif". Fakultas Teknik, Universitas Diponegoro. Semarang
- [7] Budiman, Tri Hendrawan. "*Perancangan Kontroler Pid Prediktif Robot Model Segway*" Universitas Bangka Belitung. Pangkalpinang.