



UNIVERSITAS
Dinamika

**PROTOTYPE SELF BALANCING DENGAN IMU SENSOR
MENGUNAKAN METODE INTERPOLASI LINEAR**



TUGAS AKHIR

**Program Studi
S1 TEKNIK KOMPUTER**

UNIVERSITAS
Dinamika

Oleh:

MUHAMMAD NUR GAHUTSUL AZHAM

18410200040

FAKULTAS TEKNOLOGI DAN INFORMATIKA

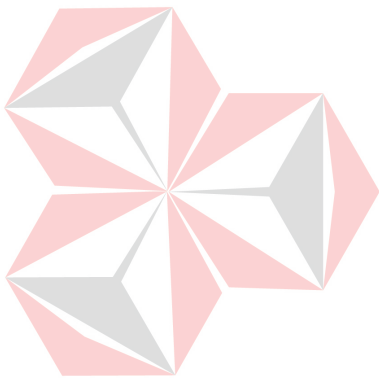
UNIVERSITAS DINAMIKA

2023

**PROTOTYPE SELF BALANCING DENGAN IMU SENSOR
MENGUNAKAN METODE INTERPOLASI LINEAR**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana Teknik**



UNIVERSITAS
Dinamika

Disusun Oleh:

Nama : Muhammad Nur Gahutsul Azham

NIM : 18410200040

Program Studi : S1 Teknik Komputer

FAKULTAS TEKNOLOGI DAN INFORMATIKA

UNIVERSITAS DINAMIKA

2023

TUGAS AKHIR

PROTOTYPE SELF BALANCING DENGAN IMU SENSOR MENGUNAKAN METODE INTERPOLASI LINEAR

Dipersiapkan dan disusun oleh:

Muhammad Nur Gahutsul Azham
18410200040

Telah diperiksa, dibahas dan disetujui oleh Dewan Pembahas


Pada: 1 Februari 2023

Susunan Dewan Pembahas

Pembimbing


- I. Pauladie Susanto, S.Kom., M.T.
NIDN 0729047501
- II. Weny Indah Kusumawati, S.Kom., M.MT.
NIDN 0721047201


Universitas Dinamika
2023.02.02 08:29:05
+07'00'


Universitas
Dinamika
2023.02.01
15:24:37 +07'00'

Pembahas

- I. Hariato, S.Kom., M.Eng.
NIDN 0722087701


cn=Harianto Harianto,
o=Universitas Dinamika,
ou=Prodi S1 Teknik Komputer,
email=hari@dinamika.ac.id,
c=ID
2023.02.02 08:54:10 +07'00'

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana



Digitally signed by
Universitas Dinamika
Date: 2023.02.07
07:22:19 +07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN. 0731017601


Dekan Fakultas Teknologi dan Informatika
UNIVERSITAS DINAMIKA

**“Segala sesuatu yang negatif – tekanan, tantangan – adalah kesempatan
bagiku untuk bangkit.”**

~ Kobe Bryant ~



UNIVERSITAS
Dinamika



Dipersembahkan kepada seluruh Keluarga Besar saya, terutama Ayah dan Ibu atas dukungan, motivasi yang tinggi dan doa yang diberikan kepada saya. Beserta orang yang telah membantu, mendukung, memberi masukan, dan memberi motivasi agar tetap berusaha dan belajar untuk menjadi yang lebih baik.

**SURAT PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH**

Sebagai mahasiswa Universitas Dinamika, saya:

Nama : Muhammad Nur Gahutsul Azham

NIM : 18410200040

Program Studi : S1 Teknik Komputer

Fakultas : Fakultas Teknologi dan Informatika

Jenis Karya : Laporan Tugas Akhir

Judul Karya : **PROTOTYPE SELF BALANCING DENGAN
IMU SENSOR MENGGUNAKAN METODE
INTERPOLASI LINEAR**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau sebagai pemilik pencipta dan Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kejarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 13 Desember 2022

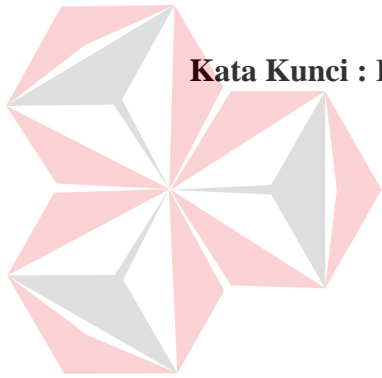
Yang menyatakan



Muhammad Nur Gahutsul Azham
NIM : 18410200040

ABSTRAK

Self balancing merupakan metode kontrol yang digunakan untuk mempertahankan posisi dari suatu objek. IMU sensor sebagai input pembacaan sudut dan percepatan yang di proses dalam rumus Interpolasi Linear. Peran IMU sensor pada *prototype self balancing* ini adalah sebagai input yang berfungsi untuk membantu keputusan gerak dari Motor DC agar dapat mempertahankan pada posisi seimbang melalui proses perhitungan interpolasi linear yang menghasilkan nilai PWM. Penelitian ini sebagai pembandingan dari hasil penelitian yang sudah ada sebelumnya yang menggunakan roda dua. Dengan adanya penelitian ini diharapkan penulis dapat menyelesaikan *Prototype Self Balancing* menggunakan sensor Serial 6 Axis Dfrobot. Pengujian sensor Serial 6 Axis Dfrobot yang digunakan pada sistem menghasilkan persentase *error* sensor sebesar 4.52%. Pengujian PWM menggunakan osiloskop untuk menentukan hasil *duty cycle* yang dihasilkan menghasilkan persentase *error* sebesar 1.72%. Pada pengujian keseluruhan sistem yang telah dilakukan berhasil berjalan dengan baik sesuai dengan algoritma yang dibuat dengan presentase *error* 0.16%.



Kata Kunci : Interpolasi Linear, Serial 6 Axis Dfrobot, Motor DC.

UNIVERSITAS
Dinamika

KATA PENGANTAR

Syukur alhamdulillah, penulis ucapkan atas rahmad dan ridha Allah Yang Maha Esa, karena atas nikmat dan kuasanya penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul ”*Prototype Self Balancing Dengan IMU Sensor Menggunakan Metode Interpolasi Linear*”, sebagai sebuah syarat untuk lulus dan mendapatkan gelar Sarjana Teknik (S.T.).

Pada kesempatan ini, penulis mengucapkan terima kasih pada berbagai pihak yang membimbing dan memberi dukungan selama proses mengerjakan Tugas Akhir dan menyusun laporan ini:

1. Orang Tua, yang telah memberikan dukungan besar hingga penulis dapat menyelesaikan Laporan Tugas Akhir ini.
2. Ibu Tri Sagirani, S.Kom., M.MT., selaku Dekan Fakultas Teknologi dan Informatika (FTI) Universitas Dinamika.
3. Bapak Pauladie Susanto, S.Kom., M.T., selaku Ketua Program Studi S1 Teknik Komputer, dan selaku Dosen Pembimbing I yang selalu memberi waktu dan bimbingan dalam menyelesaikan Tugas Akhir beserta laporan ini.
4. Ibu Weny Indah Kusumawati, S.Kom., M.MT., selaku Dosen Pembimbing II yang juga selalu memberi waktu dan bimbingan dalam menyelesaikan Tugas Akhir beserta laporan ini.
5. Bapak Harianto, S.Kom., M.Eng., selaku Dosen Pembahas yang selalu memberi waktu dan bimbingan dalam menyelesaikan laporan Tugas Akhir ini.
6. Seluruh teman-teman S1 Teknik Komputer angkatan 2018 yang memberikan semangat pantang menyerah dan selalu menemani selama proses mengerjakan Tugas Akhir ini.

Penulis menyadari, masih banyak kesalahan yang terdapat dalam penulisan laporan Tugas Akhir ini, suatu kehormatan bila pembaca dapat memberikan saran dan kritik, sehingga dapat memperbaiki kekurangan dan berusaha untuk lebih baik lagi. Penulis juga berharap, semoga laporan ini berguna, bermanfaat, serta menambah wawasan bagi pembacanya. Terimakasih.

Surabaya, 1 Februari 2023

Penulis

DAFTAR ISI

	Halaman
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Manfaat	2
BAB II LANDASAN TEORI	3
2.1 Metode Interpolasi	3
2.1.1 Interpolasi Linear	4
2.1.2 Interpolasi Kuadrat	4
2.1.3 Interpolasi Kubik	5
2.2 Motor Driver BTS7960	5
2.3 Motor DC 775 24V	6
2.4 Dfrobot Serial 6-axis Accelerometer	7
2.5 Teensy 4.1	8
BAB III METODOLOGI PENELITIAN	9
3.1 Perancangan <i>Hardware</i> (Blok Diagram)	9
3.2 Perancangan Rangkaian Skematik	11
3.2.1 Rangkaian Skematik <i>Input</i>	11
3.2.2 Rangkaian skematik <i>output</i>	12
3.3 <i>Flowchart</i>	14
3.3.1 Kontrol Sistem	14
3.3.2 Interpolasi Linear	15

3.3.3 Model Perancangan Alat Simulasi	18
BAB IV HASIL DAN PEMBAHASAN	21
4.1 Pengujian Dfrobot Serial 6 Axis Accelerometer.....	21
4.1.1 Alat Dan Bahan Yang Digunakan Pengujian Dfrobot Serial 6 Axis Accelerometer.....	21
4.1.2 Prosedur Pengujian Dfrobot Serial 6 Axis Accelerometer.....	21
4.2 Pengujian <i>Duty Cycle</i> PWM	22
4.2.1 Alat Dan Bahan Yang Digunakan Pengujian <i>Duty Cycle</i> PWM...	23
4.2.2 Prosedur Pengujian.....	23
4.3 Pengujian Keseluruhan Sistem.....	26
4.3.1 Alat Dan Bahan Yang Digunakan Pengujian Keseluruhan Sistem	26
4.3.2 Prosedur Pengujian Keseluruhan Sistem.....	26
BAB V PENUTUP.....	28
5.1 Kesimpulan.....	28
5.2 Saran.....	28
DAFTAR PUSTAKA	29
LAMPIRAN.....	30



DAFTAR GAMBAR

	Halaman
Gambar 2.1 Metode Interpolasi.....	3
Gambar 2.2 Interpolasi Linear	4
Gambar 2.3 Motor Driver BTS7960	5
Gambar 2.4 Motor DC 24V	6
Gambar 2.5 Dfrobot Serial 6-axis Accelerometer.....	7
Gambar 2.6 Teensy 4.1	8
Gambar 3.1 Blok diagram.....	9
Gambar 3.2 Rangkaian skematik Input.....	11
Gambar 3.3 Rangkaian skematik Output	12
Gambar 3.4 Flowchart kontrol sistem.....	14
Gambar 3.5 Flowchart Interpolasi Linear	15
Gambar 3.6 Flowchart Motor DC maju	16
Gambar 3.7 Flowchart Motor DC mundur.....	17
Gambar 3.8 Flowchart Motor DC berhenti.....	18
Gambar 3.9 Tampak bagian dalam	19
Gambar 3.10 Tampak belakang	19
Gambar 3.11 Gambar alat keseluruhan.....	20
Gambar 4.1 Hasil Duty Cycle 10%	24
Gambar 4.2 Hasil Duty Cycle 50%	25
Gambar 4.3 Hasil Duty Cycle 100%	25

DAFTAR TABEL

	Halaman
Tabel 4.1 Hasil perbandingan IMU sensor dengan Inclinometer	21
Tabel 4.2 Hasil pengujian <i>Duty Cycle</i> PWM.....	24
Tabel 4.3 Algoritma Motor DC.....	26
Tabel 4.4 Pengujian keseluruhan sistem	27



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 <i>Source Code</i> Pengujian Dfrobot <i>Serial 6-Axis Accelerometer</i>	30
Lampiran 2 <i>Source Code</i> Pengujian <i>Duty Cycle</i>	32
Lampiran 3 <i>Source Code</i> Keseluruhan	33
Lampiran 4 Rangkaian Skematik Keseluruhan.....	36
Lampiran 5 Tabel Koneksi Pin.....	36
Lampiran 6 Dokumentasi Pengujian <i>Duty Cycle</i> pada Osiloskop.....	37
Lampiran 7 Hasil Plagiasi	42



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Self balancing merupakan suatu metode kontrol yang digunakan untuk mempertahankan posisi agar tetap seimbang, dibantu dengan IMU sensor sebagai input pembacaan sudut dan percepatan yang nantinya diolah pada mikrokontroller. Perancangan *prototype self balancing* ini bertujuan untuk mengumpulkan data uji *self balancing* sebelum nantinya dapat diterapkan langsung pada proyek yang lebih besar. Adanya metode kontrol *self balancing* untuk membantu objek agar tetap seimbang dan meminimalisir resiko terjatuh (NAFI', 2017).

Inertial Measurement Unit (IMU) merupakan suatu sensor yang memiliki komponen penyusun seperti *accelerometer* dan *gyroscope*. *Accelerometer* berfungsi untuk mengukur percepatan dari suatu objek terhadap waktu, sedangkan cara kerja dari *gyroscope* yaitu mendeteksi sudut gerakan dari objek (Firman, 2016). Peran IMU sensor pada *prototype self balancing* ini adalah sebagai input yang berfungsi untuk membantu keputusan gerak dari Motor DC agar dapat melakukan percepatan putaran yang seimbang antara derajat kemiringan dengan *output PWM* pada motor DC. Penelitian kali ini penulis menerapkan metode Interpolasi Linear pada *prototype self balancing* sebagai pembanding dari hasil penelitian yang sudah ada sebelumnya.

Dalam penelitian ini penulis menggunakan dinamo motor DC yang telah terpasang baling-baling sebagai simulasi *self balancing*, sebelum nantinya dapat diuji pada proyek yang lebih besar. Dengan adanya baling-baling pada motor DC maka percepatan ke arah kanan atau ke arah kiri dapat dilihat pada arah putaran baling-baling ke kanan atau ke kiri dari Motor DC.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat dirumuskan masalah pada Tugas Akhir ini adalah bagaimana merancang *prototype self balancing* menggunakan IMU Sensor?

1.3 Batasan Masalah

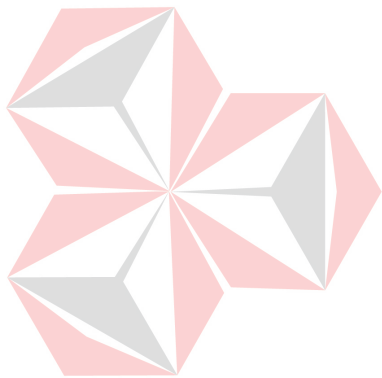
Dalam pembuatan Tugas Akhir ini, batasan masalah dibatasi pada, *Prototype self balancing* hanya dirancang untuk satu arah sumbu.

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah diatas mendapatkan tujuan pada tugas akhir ini adalah merancang *prototype self balancing* menggunakan sensor IMU sensor dengan metode interpolasi linear.

1.5 Manfaat

Adapun dari Tugas Akhir ini dapat diperoleh manfaat adalah menambah ilmu pengetahuan terkait penerapan *Self Balancing* menggunakan IMU Sensor.

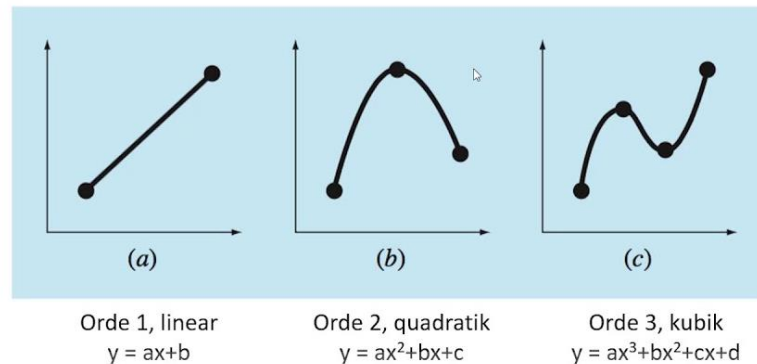


UNIVERSITAS
Dinamika

BAB II LANDASAN TEORI

2.1 Metode Interpolasi

Interpolasi menggunakan fungsi polinomial



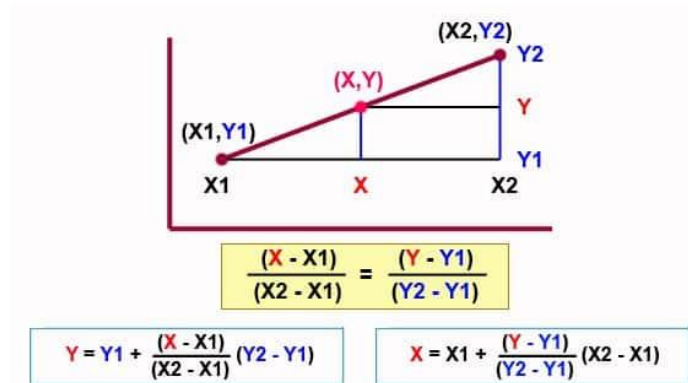
Gambar 2.1 Metode Interpolasi
(Sumber: (Hartomo, 2006))

Secara sederhana interpolasi yaitu metode yang digunakan untuk menentukan fungsi yang sesuai dari titik-titik yang diberikan. Secara umum, interpolasi adalah suatu proses menyusun suatu fungsi atau persamaan dengan menggunakan beberapa titik yang diketahui untuk memprediksi dan estimasi nilai pada titik lain yang belum diketahui nilainya. Interpolasi juga dapat digunakan untuk menyederhanakan fungsi yang kompleks menjadi fungsi yang lebih sederhana dengan menggunakan titik-titik sampel data yang diinterpolasikan. Dengan interpolasi, titik-titik yang diketahui dan lainnya dapat digambarkan menjadi suatu kurva.

Berikut beberapa jenis interpolasi, yaitu:

1. Interpolasi Linear
2. Interpolasi Kuadrat
3. Interpolasi Kubik

2.1.1 Interpolasi Linear



Gambar 2.2 Interpolasi Linear
(Sumber: (Ukuransatuan.com, 2016))

Pengertian nilai-nilai yang terdapat pada gambar 2.2 adalah Metode Interpolasi Linear merupakan suatu metode yang digunakan untuk mengetahui nilai dengan cara menghubungkan dua titik yang mengapit daerah yang dicari interpolasinya. Dimana y , x merupakan nilai yang dicari di antara titik pertama dan titik kedua. x_1 , y_1 merupakan titik pertama dan x_2, y_2 titik kedua. Contoh rumus Interpolasi Linear dua titik, yaitu:

$$f(y) = \left(\frac{(x - x_1)}{(x_2 - x_1)} (y_2 - y_1) \right) + y_1 \quad (1)$$

Penerapan metode interpolasi linear pada penelitian kali ini yaitu untuk mendapatkan output nilai pwm yang disesuaikan dengan rentang derajat bacaan sensor *gyroscope*.

2.1.2 Interpolasi Kuadrat

Interpolasi kuadrat digunakan untuk penggunaan lebih dari 2 titik. Interpolasi kuadrat dapat dilakukan dengan menggunakan masing-masing titik ke dalam rumus interpolasi kuadrat, Interpolasi Kuadrat muncul dari interpolasi linear yang menunjukkan adanya *error* estimasi. *Error* pada interpolasi linear tersebut adalah konsekuensi bahwa titik yang ada hanya sampai dua.

Jika diberikan tiga titik maka memungkinkan untuk dibuat interpolasi polinomial order 2. Dengan rumus umum yang sering digunakan seperti berikut:

$$y1 = y1 + ((y2 * x1) + (y3 * x1)) \quad (2)$$

$$y2 = y1 + ((y2 * x2) + (y3 * x2)) \quad (3)$$

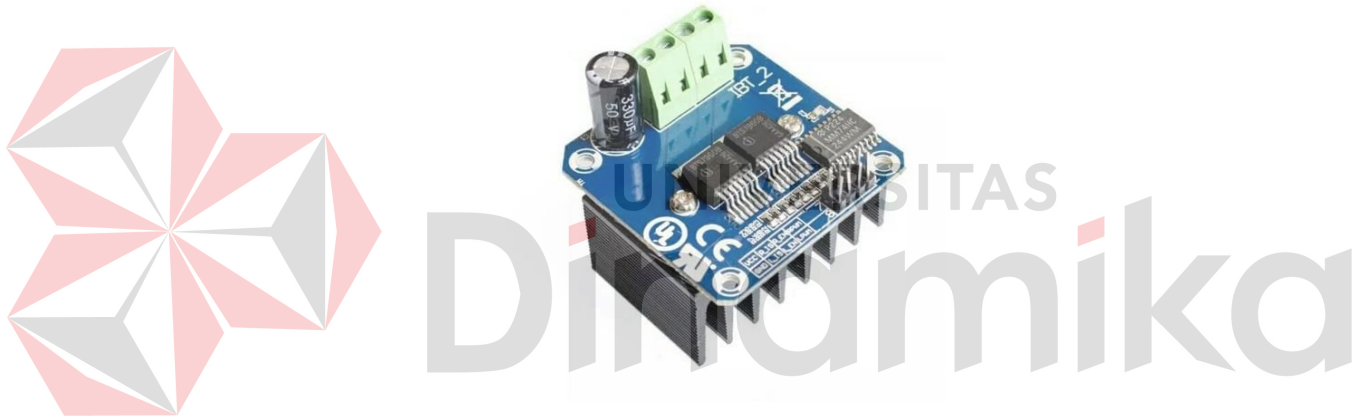
$$y3 = y1 + ((y2 * x3) + (y3 * x3)) \quad (4)$$

2.1.3 Interpolasi Kubik

Interpolasi kubik dalam penyelesaiannya menggunakan 4 titik data. Hasil yang dicapai pada interpolasi kubik menunjukkan hasil lebih baik dari interpolasi linear dan juga interpolasi kuadrat. Berikut bentuk rumus umum dari interpolasi kubik:

$$P3(x) = a0 + a1(x - x0) + a2(x - x0 + a3(x - x0)(x - x1)(x - x2)) \quad (5)$$

2.2 Motor Driver BTS7960



Gambar 2.3 Motor Driver BTS7960
(Sumber: <https://teramitra.co.id/part/bts-7960/>)

Driver motor DC ini mampu menerima beban hingga 43A dengan maksimal tegangan 27Vdc secara kontinu. Motor driver ini juga memiliki fungsi PWM dengan tegangan input level antara 3.3V-5Vdc. Motor Driver BTS7960 memiliki 8 pin input dari mikrokontroller. Motor driver BTS7960 menggunakan full H-bridge dengan IC BTS7960 yang memiliki perlindungan saat terjadi panas atau arus berlebih. Adanya IC BTS7960 juga sangat baik untuk mendrive motor DC dengan dua arah, yaitu counter clock wise (CCW) dan clock wise (CW).

2.3 Motor DC 775 24V



Gambar 2.4 Motor DC 24V
(Sumber: <https://www.tokopedia.com>)

Motor DC adalah jenis motor arus searah yang mengubah energi listrik menjadi energi mekanis berupa putaran. Motor DC terdiri dari stator yang terdiri dari badan motor, inti stator, bearing dan terminal box. Lalu ada rotor yang terdiri dari lilitan pada bagian kutub magnet dan berfungsi menciptakan gaya tarik menarik yang menghasilkan putaran sebagai penggerak.

Keuntungan utama dari motor DC ini yaitu sebagai aktuator yang paling umum digunakan untuk menghasilkan gerakan terus menerus dimana kecepatan putarannya dapat dengan mudah dikontrol, hal inilah yang membuat motor DC menjadi sangat ideal untuk digunakan dalam aplikasi pengaturan kontrol kecepatan, kontrol tipe servo, dan / atau positioning.

2.4 Dfrobot Serial 6-axis Accelerometer



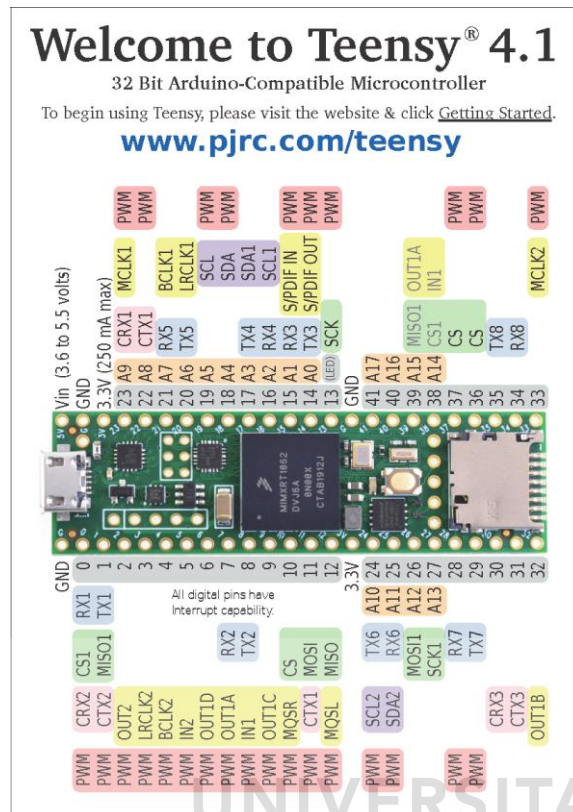
Gambar 2. 5 Dfrobot Serial 6-axis Accelerometer
(Sumber: <https://www.dfrobot.com/product-2200.html>)

Modul ini mengintegrasikan giroskop presisi tinggi, akselerometer, mikro-prosesor performa tinggi dan penyelesaian dinamika canggih, serta algoritma filter Kalman yang bertujuan untuk dengan cepat menyelesaikan pergerakan sikap modul saat ini secara real-time. Penggunaan teknologi penyaringan digital canggih dapat secara efektif mengurangi kebisingan pengukuran dan meningkatkan akurasi.

Modul ini dilengkapi dengan pemecah gerakan bawaan yang dapat mendapatkan sikap akurat dalam lingkungan dinamis yang digabungkan dengan algoritma filter Kalman dinamis. Akurasi pengukuran statisnya mencapai 0.05 derajat (dinamis 0.1) dengan stabilitas tinggi, yang dapat menghasilkan kinerja yang lebih baik bahkan daripada beberapa Inclinometer lain yang ada di pasaran. Modul ini dapat juga disebut IMU sensor karena memiliki kegunaan yang sama, yaitu untuk mengetahui sebuah derajat kemiringan.

Ada rangkaian penstabil tegangan di dalam modul. Produk harus dioperasikan pada 3.3-5V dan level pinnya kompatibel dengan sistem tertanam 3.3/5V. Ini menggunakan antarmuka TTL untuk koneksi. Selain itu, modul ini mendukung baud rate 2400bps-921600bps yang dapat disesuaikan dan output data 0.1Hz~200Hz.

2.5 Teensy 4.1

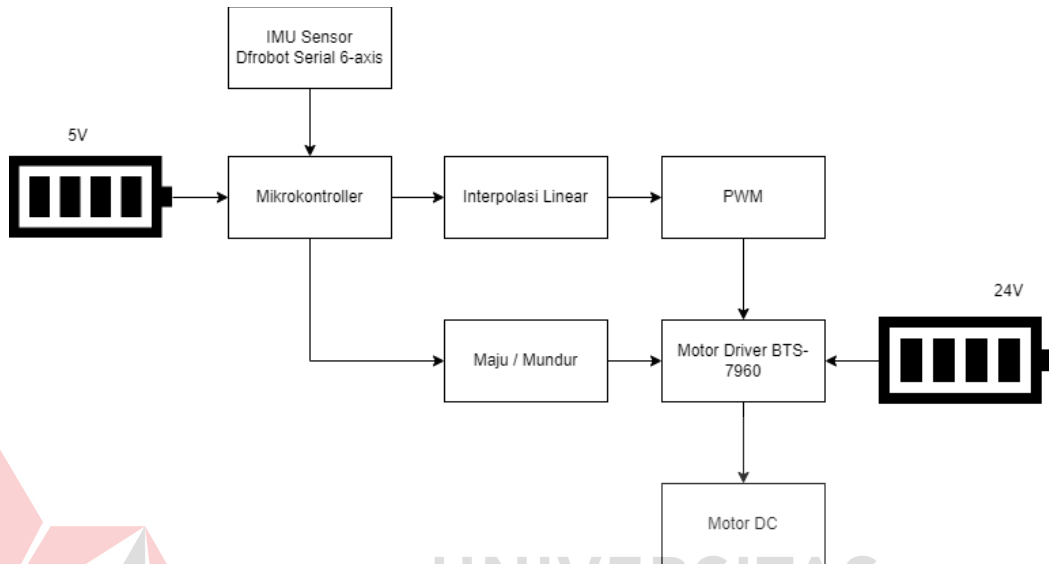


Gambar 2.6 Teensy 4.1
(Sumber: <https://www.pjrc.com/store/teensy41.html>)

Teensy merupakan mikrokontroler berbasis prosesor ARM Cortex-M7 dengan menggunakan chip NXP iMXRT1062. NXP Imxrt1062 merupakan prosesor ‘cross-over’ yang memiliki fungsi mikrokontroler dengan banyak flash, RAM dan dapat mengelola banyak data. Teensy 4.1 juga dibekali dengan Ethernet PHY 100MB sebagai koneksi nirkabel dan socket kartu SD. Konsumsi arus yang dibutuhkan teensy 4.1 ketika berjalan pada 600MHz mengkonsumsi arus sekitar 100mA. Berbeda dengan mikrokontroler lainnya, dimana ketika kecepatan jam diubah dapat menyebabkan baud rate yang salah. Dukungan perangkat keras Teensy 4.1 dan perangkat lunak Teensyduino dirancang untuk memungkinkan perubahan kecepatan secara dinamis pada fungsi pengaturan waktu, sehingga laju baud rate tidak mengalami gangguan. Teensy dapat dijalankan pada software arduino ide dengan menambahkan library Teensy dan instal software Teensyduino untuk mengcompile program pada aplikasi arduino IDE. *reset*. Hal tersebut adalah semua yang diperlukan untuk mendukung sebuah rangkaian mikrokontroler.

BAB III METODOLOGI PENELITIAN

3.1 Perancangan *Hardware* (Blok Diagram)



Gambar 3.1 Blok diagram

Berikut penjelasan dari blok diagram atau perancangan perangkat keras:

A. Input

a. Sensor Gyroscope (Serial 6 Axis Dfrobot)

Berfungsi sebagai *input* untuk membaca derajat kemiringan dari *prototype self balancing* secara kontinu menggunakan komunikasi serial yang terhubung dengan mikrokontroller teensy 4.1. Nilai bacaan derajat dari IMU sensor masuk ke dalam perhitungan interpolasi linear untuk mendapatkan nilai PWM. Keluaran dari sensor *gyroscope* berupa nilai derajat.

B. Proses

a. Mikrokontroller

Mikrokontroller berfungsi sebagai mini komputer dalam bentuk kumpulan chip IC yang bertugas menjalankan sistem operasi rangkaian elektronik. Fungsi mikrokontroler yang digunakan pada penelitian kali ini sebagai

penentuan gerak ke kanan atau ke kiri dan juga menghitung nilai *input* derajat dari IMU Sensor untuk menghasilkan *output* PWM melalui perhitungan Interpolasi Linear.

b. Rumus Interpolasi Linear

Hasil perhitungan interpolasi linear berfungsi untuk mendapatkan nilai PWM pada rentang derajat yang telah ditentukan dari hasil pembacaan derajat sensor *gyroscope*.

$$f(x) = ((x - x_1) / (x_2 - x_1)) (y_2 - y_1) + y_1 \quad (6)$$

Nilai input dari *gyroscope* yang berupa satuan derajat masuk ke dalam nilai 'x', lalu untuk 'x1' sebagai titik derajat terendah, 'x2' sebagai titik tertinggi derajat, 'y2' sebagai titik tertinggi nilai PWM, dan 'y1' sebagai titik terendah nilai PWM. Berikut disertakan contoh proses perhitungan interpolasi linear yang mendapat nilai input derajat dari IMU sensor dan diolah untuk mendapatkan nilai output PWM:

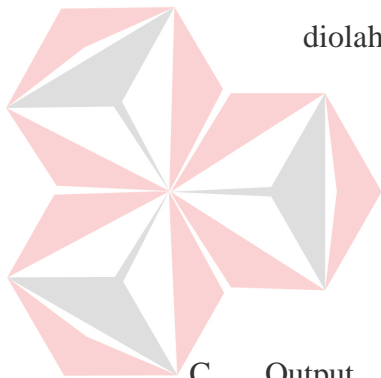
$$f(10) = ((10 - 1) / (15 - 1)) (255 - 0) + 0 \quad (7)$$

$$f(10) = ((9) / (14)) (255 - 0) + 0 \quad (8)$$

$$f(10) = (0,643) \cdot (255) + 0 \quad (9)$$

$$f(10) = 163.929 + 0 \quad (10)$$

$$f(10) = 163.929 \quad (11)$$



C. Output

a. Motor Driver BTS7960

Meneruskan daya pada baterai ke motor DC dan mengubah output sinyal PWM dari hasil perhitungan Interpolasi Linear agar dapat diterima oleh motor DC. Motor Driver BTS7960 juga menjadi kontrol untuk bergerak ke arah kanan atau kiri menyesuaikan nilai output RPWM atau LPWM yang didapat dari hasil perhitungan interpolasi linear.

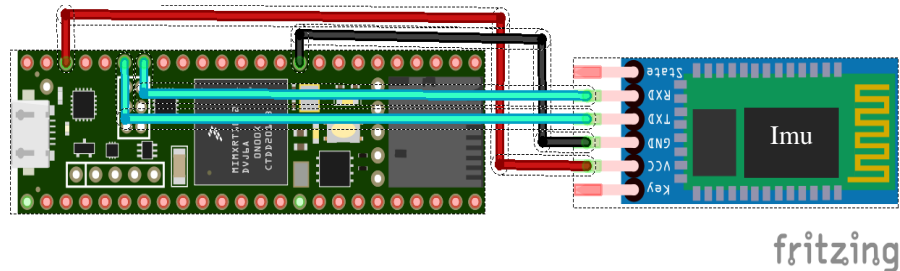
b. Motor DC 775 24V

Mesin penggerak dari *Prototype Self Balancing* yang terdiri dari rotor atau bagian yang berputar karena adanya gaya elektromagnetik dari stator. Stator dengan fungsi sebagai medan putar motor untuk memberikan gaya elektromagnetik pada rotor. Motor DC akan bergerak mengikuti perintah dari hasil perhitungan mikro melalui motor driver. Motor DC mengalami

percepatan dan perubahan arah perputaran menyesuaikan dengan logika yang diterapkan.

3.2 Perancangan Rangkaian Skematik

3.2.1 Rangkaian Skematik *Input*



Gambar 3.2 Rangkaian skematik input

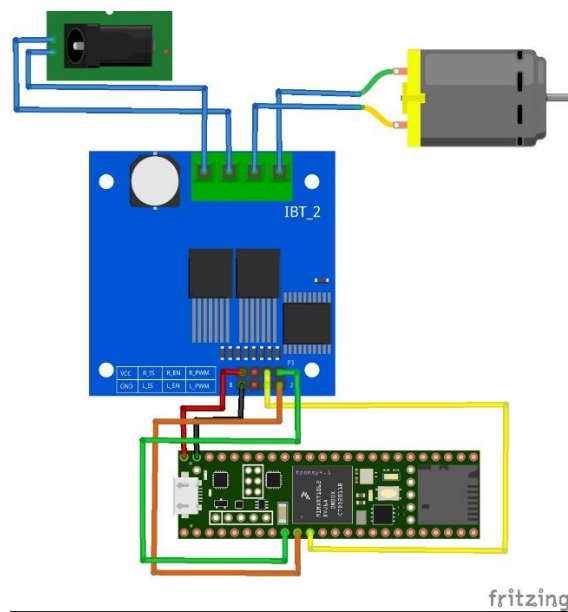
Gambar 3.2 menunjukkan rangkaian skematik *input* yang digunakan pada Tugas Akhir ini sebagai *Prototype Self Balancing* pada motor DC, Rangkaian skematik keseluruhan dapat dilihat pada Lampiran Rangkaian Keseluruhan. Rangkaian skematik yang digunakan terdiri dari beberapa komponen, diantaranya adalah:

- a. IMU Sensor (DFRobot Serial 6-Axis Accelerometer)
- b. Teensy 4.1

Penjelasan rangkaian skematik IMU Sensor dengan Teensy 4.1 adalah IMU Sensor memiliki 4 pin yang perlu dihubungkan ke Teensy dengan pemasangan pin sebagai berikut:

- a. Pin VCC IMU Sensor dihubungkan ke pin VCC pada Teensy.
- b. Pin GND IMU Sensor dihubungkan ke pin GND pada Teensy.
- c. Pin Rx IMU Sensor dihubungkan ke pin analog D21 (Tx) pada Teensy.
- d. Pin Tx IMU Sensor dihubungkan ke pin analog D20 (Rx) pada Teensy

3.2.2 Rangkaian skematik *output*



Gambar 3.3 Rangkaian skematik *output*

Gambar 3.3 menunjukkan rangkaian skematik *output* yang digunakan pada Tugas Akhir ini sebagai *Prototype Self Balancing* pada motor DC. Rangkaian skematik yang digunakan terdiri dari beberapa komponen, diantaranya adalah:

- a. Motor Driver BTS7960
- b. Teensy 4.1
- c. Motor DC 24V
- d. Power Supply 24V

Keterangan komponen pada gambar 3.3:

1. Motor Driver BTS-7960 dengan Teensy 4.1

Motor Driver BTS-7960 memiliki 8 pin yang dapat dihubungkan semua atau Sebagian dari pinnya, berikut pemasangan pin Motor Driver ke Teensy:

Pin daya untuk rangkaian Motor Driver:

- a. Pin VCC Motor Driver dihubungkan ke pin VCC pada Teensy.
- b. Pin GND Motor Driver dihubungkan ke pin DNG pada Teensy.

Pin digital kontrol kanan dan kiri dihubungkan secara seri:

Pin R_EN dan L_EN Motor driver dihubungkan secara seri ke pin D11 Teensy yang dapat digunakan untuk *output* PWM.

Pin digital output pwm dari teensy ke motor driver:

- a. Pin RPWM Motor Driver dihubungkan pada pin Digital 9 Teensy yang digunakan untuk *output* PWM ke arah kanan.
- b. Pin LPWM Motor Driver dihubungkan pada pin Digital 10 Teensy yang digunakan sebagai output PWM ke arah kiri.

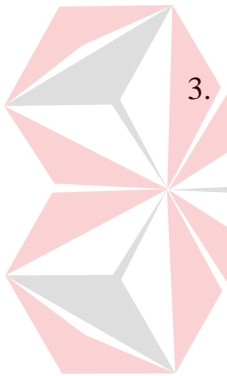
2. *Power Supply* 24v dengan Motor Driver BTS7960

Power Supply digunakan untuk mengubah sumber tegangan AC menjadi sumber tegangan DC agar dapat diteruskan ke motor driver sebagai sumber daya ke motor DC yang bekerja sebagai output. Berikut cara pemasangan *Power Supply* ke motor driver:

- a. Menghubungkan kutub positif *Power Supply* ke terminal block B+ pada motor driver.
- b. Menghubungkan kutub negatif *Power Supply* ke Terminal Block B- pada motor driver.

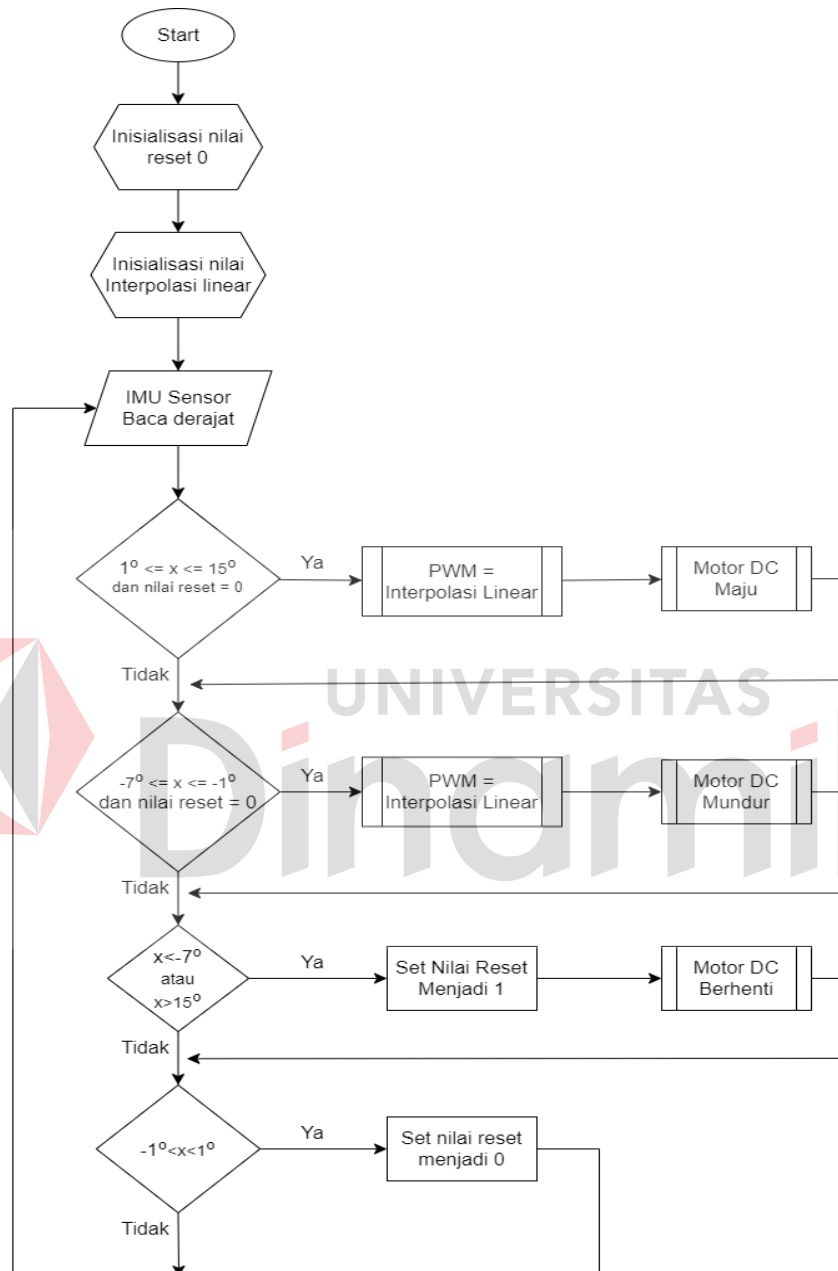
3. Motor DC 775 24v dengan Motor Driver BTS7960

VCC dan GND Motor DC dihubungkan ke terminal blok motor driver (M+ dan M-). Putaran motor DC 24v tipe 775 dapat dijalankan searah jarum jam dan berlawanan arah jarum jam. Memberikan tegangan DC pada motor DC tanpa logika dapat menggerakkan motor searah jarum jam atau berlawanan arah jarum jam tergantung pemasangan pada kaki vcc dan ground.



3.3 Flowchart

3.3.1 Kontrol Sistem



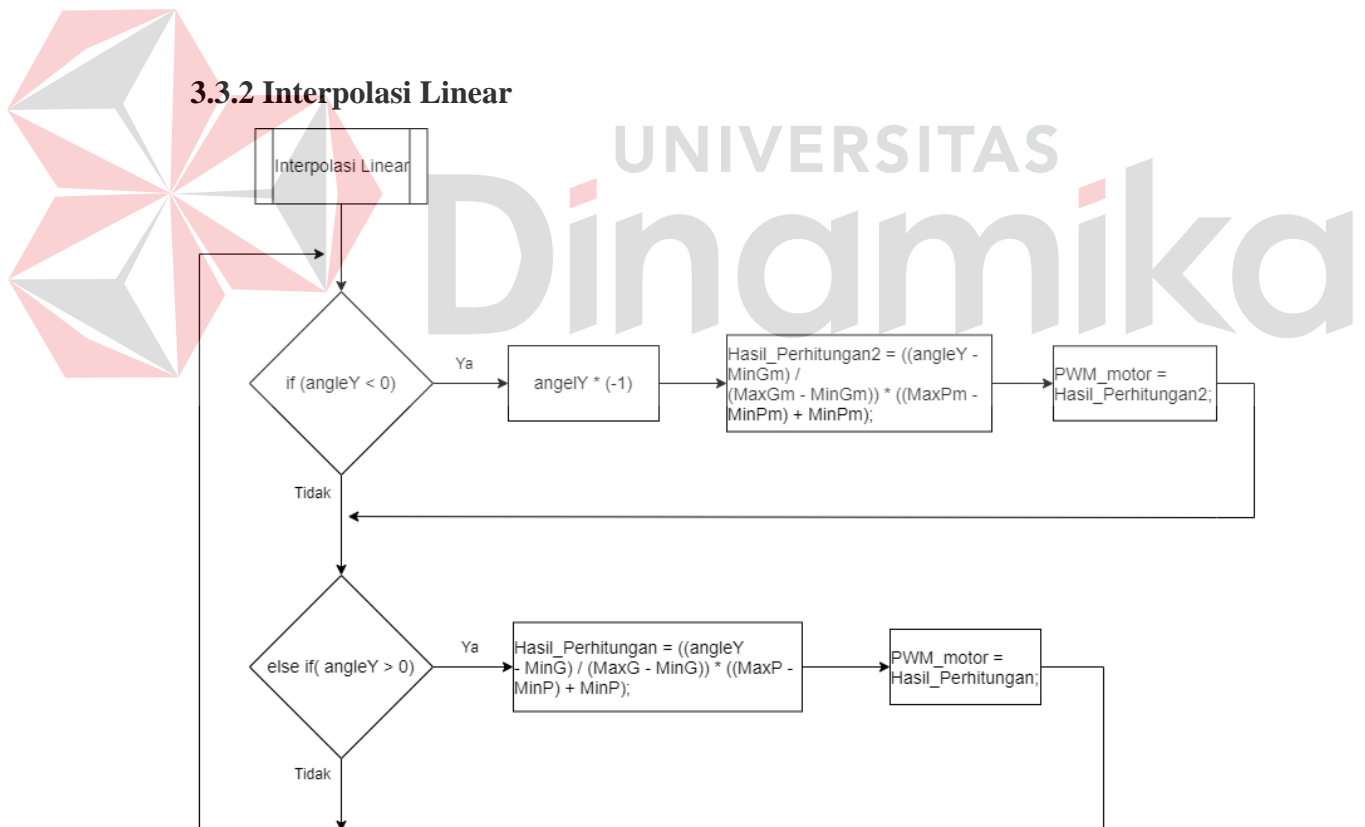
Gambar 3.4 Flowchart kontrol sistem

Tahap pertama dari sistem yaitu sensor gyroscope membaca nilai derajat kemiringan, dimana ketika nilai bacaan sensor *gyroscope* bernilai positif, maka dapat diasumsikan putaran motor dc ke arah kanan (*Clock wise*) dengan batas kemiringan maksimal 15 derajat, yang jika melebihi kemiringan tersebut, maka

nilai PWM menjadi 0 atau motor berhenti dan nilai reset menjadi 1. Putaran motor DC akan tetap berhenti hingga nilai reset kembali 0. Ketika nilai kemiringan kembali pada posisi tegak atau kurang dari 1 dan lebih dari -1 derajat maka nilai reset kembali menjadi 0. Ketika nilai bacaan sensor *gyroscope* bernilai negatif, maka dapat diasumsikan putaran motor DC ke arah kiri (*Counter clock wise*) dengan batas kemiringan maksimal -7. Rentang keluaran nilai PWM untuk gerakan ke arah kanan yaitu 0-255 dan 0-127 untuk bergerak ke arah kiri.

Setelah mengambil nilai dari sensor *gyroscope* proses selanjutnya yaitu masuk di dalam perhitungan interpolasi linear untuk mendapatkan nilai PWM. Konsep dari *self balancing* ini yaitu motor DC akan bergerak ke arah kanan atau kiri dengan percepatan yang berdasarkan keluaran nilai PWM dari hasil perhitungan di dalam rumus interpolasi linear agar dapat menghasilkan nilai yang seimbang antara derajat kemiringan dengan percepatan putaran motor DC.

3.3.2 Interpolasi Linear

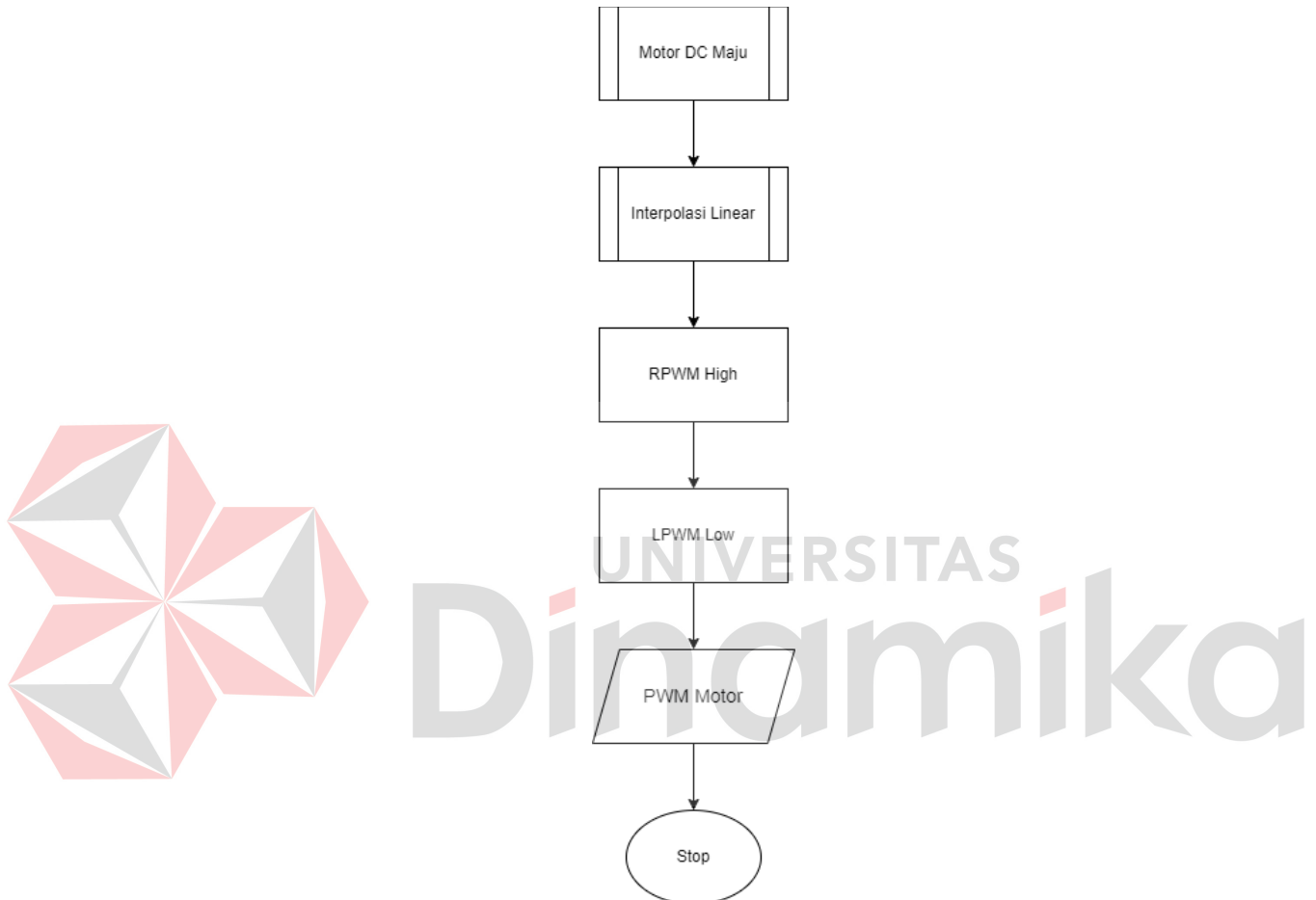


Gambar 3.5 Flowchart Interpolasi Linear

Alur dari Interpolasi Linear menentukan arah dari motor melalui sudut kemiringan, setelah sudut kemiringan ditentukan dilakukan proses perhitungan

mencari nilai PWM dengan rumus perhitungan Interpolasi Linear kemudian hasil perhitungan menjadi nilai PWM yang disimpan pada PWM_Motor untuk motor DC.

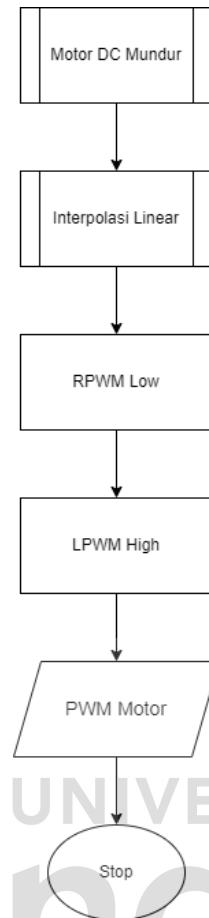
Motor DC Maju



Gambar 3.6 Flowchart Motor DC maju

Alur proses dari motor DC maju memanggil perhitungan Interpolasi Linear untuk menentukan kondisi maju atau RPWM High dan membuat LPRM Low. Setelah menentukan arah motor, nilai PWM diambil dari hasil perhitungan Interpolasi Linear untuk menentukan percepatan motor DC.

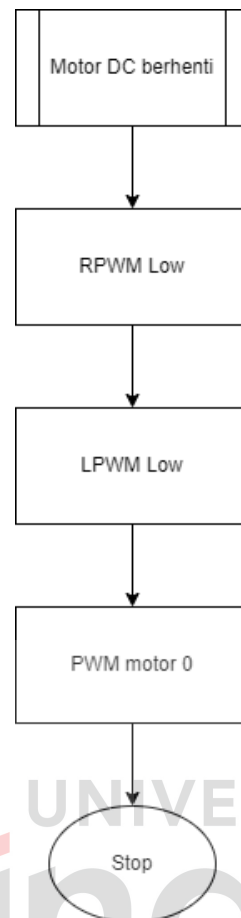
Motor DC Mundur



Gambar 3.7 Flowchart Motor DC mundur

Alur proses dari motor DC mundur memanggil perhitungan Interpolasi Linear untuk menentukan kondisi mundur atau RPWM Low dan membuat LPRM High. Setelah menentukan arah motor, nilai PWM diambil dari hasil perhitungan Interpolasi Linear untuk menentukan percepatan motor DC.

Motor DC Berhenti



Gambar 3.8 Flowchart Motor DC berhenti

Alur proses dari motor berhenti adalah membuat nilai percepatan PWM menjadi 0 sehingga RPWM dan LPWM menjadi Low atau berhenti bergerak.

3.3.3 Model Perancangan Alat Simulasi

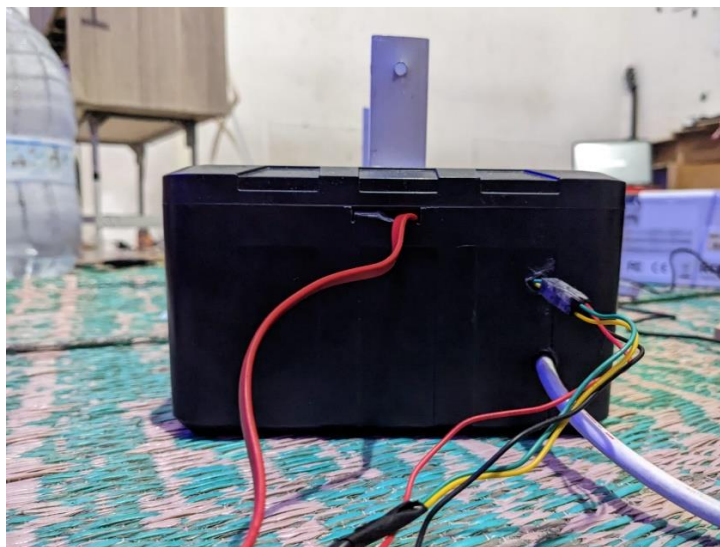
Gambar model perancangan pada Tugas Akhir ini dapat dilihat pada gambar 3.9 dibawah ini. Terdapat 1 box komponen yang berisi beberapa komponen seperti Teensy, motor driver, *power supply* 24v, dan juga motor dc 775 24V. Dimana pada boxnya sendiri terdapat sedikit modifikasi agar as dari motor DC dapat berada di luar dan akan digunakan untuk menggerakkan baling-baling yang berukuran 10 inci. Ada beberapa komponen yang berada di luar box hitam dengan bentuk yang menyerupai remot yaitu, sensor *gyroscope* dan juga inclinometer digital sebagai pembanding dari sensor *gyroscope*.



Gambar 3.9 Tampak bagian dalam

Pada bagian dalam box terdapat komponen diantaranya:

1. Power supply 24V
2. Motor DC 24V
3. Motor Driver BTS7960
4. Mikrokontroler Teensy 4.1



Gambar 3.10 Tampak belakang

Pada bagian belakang box diberikan lubang sebagai jalur kabel untuk *Power supply* ke sumber daya AC dan IMU Sensor yang terhubung ke pin mikro yang berada di bagian dalam box.



Gambar 3.11 Gambar alat keseluruhan

BAB IV HASIL DAN PEMBAHASAN

4.1 Pengujian Dfrobot Serial 6 Axis Accelerometer

Pengujian hasil baca *angle* Dfrobot Serial 6 Axis atau IMU sensor dengan inclinometer untuk mengetahui selisih dari nilai *angle* yang di dapat. Perbandingan tingkat akurasi IMU sensor dengan inclinometer dilakukan untuk mendapatkan rata-rata nilai error dari hasil pembacaan angle dengan menggunakan rumus:

$$error = \left(\left| \frac{\text{Inclinometer} - \text{IMU sensor}}{\text{Inclinometer}} \right| \right) \times 100\% \quad (12)$$

4.1.1 Alat Dan Bahan Yang Digunakan Pengujian Dfrobot Serial 6 Axis Accelerometer

Berikut merupakan alat dan bahan yang digunakan dalam proses pengujian pembacaan *angle* IMU sensor.

1. IMU Sensor (Dfrobot Sesial 6-Axis Accelerometer)
2. Inclinometer
3. Teensy 4.1
4. Laptop

4.1.2 Prosedur Pengujian Dfrobot Serial 6 Axis Accelerometer

Berikut merupakan tata cara melakukan proses pengujian pembacaan *angle* IMU sensor.

1. Mengbungkan pin Rx dan Tx sensor pada Teensy 4.1.
2. Mengunggah program lampiran 1 pada arduino IDE ke mikro Teensy 4.1.
3. Meletakkan sensor *gyroscope* dan inclinometer pada bidang yang datar.
4. Menekan tombol *zero* di inclinometer saat telah sejajar dengan IMU sensor.
5. Membandingkan hasil pembacaan sensor *gyroscope* dan inclinometer.
6. Mencatat hasil pengujian kemudian ulangi Langkah 3 dengan kemiringan yang berbeda.

Tabel 4.1 Hasil perbandingan IMU sensor dengan Inclinometer

No.	IMU sensor (°)	Inclinometer (°)	Selisih	Error (%)
1	6.58	6.95	0.37	5.32

No.	IMU sensor (°)	Inclinometer (°)	Selisih	Error (%)
2	8.31	8.8	0.49	5.56
3	9.52	9.95	0.43	4.32
4	3.13	3.32	0.19	5.72
5	2.83	3.01	0.18	5.98
6	2.52	2.73	0.21	7.69
7	3.22	3.43	0.21	6.12
8	4.11	4.32	0.21	4.86
9	12.08	12.43	0.35	2.81
10	14.26	14.83	0.57	3.84
11	7.38	7.06	0.32	4.53
12	6.88	6.73	0.15	2.22
13	6.63	6.42	0.21	3.27
14	3.38	3.09	0.29	9.38
15	5.47	5.23	0.24	4.58
16	5.95	5.78	0.17	2.94
17	5.72	5.51	0.21	3.81
18	8.03	7.79	0.24	3.08
19	2.83	2.64	0.19	7.19
20	1.72	1.65	0.07	4.24
21	0.82	0.87	0.05	5.74
22	2.18	2.35	0.17	7.23
23	11.3	11.62	0.32	2.75
24	13.1	13.49	0.39	2.89
25	13.95	14.29	0.34	2.37
26	8.72	9.04	0.32	3.53
27	9.8	10.25	0.45	4.39
28	15.2	15.58	0.38	2.43
29	13.15	13.62	0.47	3.45
30	10.25	10.62	0.37	3.48
Rata-rata				4.52%.

Tabel 4.1 menunjukkan data yang dihasilkan pada pengujian sensor IMU sensor. Dari data yang dihasilkan menunjukkan rata-rata nilai *error* sebesar 4.52%.

4.2 Pengujian *Duty Cycle* PWM

Pada pengujian ini digunakan untuk mengetahui pengaruh sinyal PWM terhadap perubahan kecepatan motor DC. Pengujian PWM bertujuan untuk mengetahui apakah program pengatur kecepatan motor DC yang telah dibuat agar dapat mengatur gerak dari motor DC. Metode pengujian dilakukan dengan cara memberikan nilai *duty cycle* PWM secara bertahap dengan kelipatan 10% dan mengamati hasilnya secara langsung dan menentukan nilai keluaran sesuai dengan nilai *duty cycle* yang diberikan. Pengujian dilakukan dengan cara membandingkan hasil pengukuran langsung pada pin D11 dan ground pada mikro menggunakan osiloskop yang dibandingkan dengan hasil perhitungan. Pengujian dilakukan sebanyak

10 kali dengan *duty cycle* PWM 10%. 20%. 30% sampai 100%. ‘Hasil Pengukuran’ di dapat dari pin D11 mikro. Sedangkan ‘Hasil Perhitungan’ di dapat dari perhitungan nilai PWM dibagi dengan nilai *duty cycle* yang ditentukan dengan menggunakan perhitungan manual dengan rumus:

$$D = \left(\frac{t_{ON}}{t_{ON} + t_{OFF}} \right) \times 100\% \quad (13)$$

1. D Merupakan notasi yang digunakan untuk *Duty Cycle*.
2. tON Merupakan notasi yang digunakan untuk waktu ON.
3. tON Merupakan notasi yang digunakan untuk waktu OFF.

4.2.1 Alat Dan Bahan Yang Digunakan Pengujian *Duty Cycle* PWM

Berikut merupakan alat dan bahan yang digunakan dalam proses pengujian *duty cycle*.

1. Motor DC 24V
2. Teensy 4.1
3. Laptop
4. Osiloskop

4.2.2 Prosedur Pengujian

Berikut merupakan tata cara dalam proses pengujian tegangan *duty cycle*.

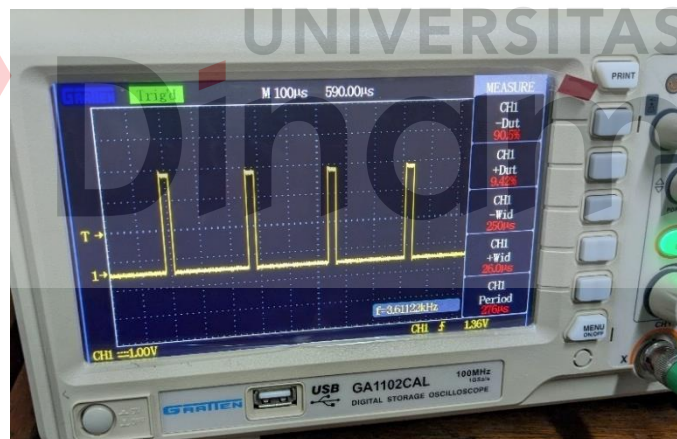
1. Menghubungkan motor driver dengan Teensy 4.1.
2. Menghubungkan Teensy 4.1 dengan laptop melalui kabel data.
3. Mengunggah program lampiran 2 pada Arduino IDE ke Teensy 4.1.
4. Menyalakan osiloskop.
5. Menghubungkan kabel sinyal (merah) probe osiloskop dengan pin 11 Teensy 4.1.
6. Menghubungkan kabel *ground* (hitam) probe osiloskop dengan pin GND Teensy 4.1.
7. Membandingkan nilai *Duty Cycle* yang tampil pada layar osiloskop dengan hasil perhitungan manual, kemudian catat hasil pengujiannya.

$$Error = \left(\frac{|(Duty\ Cycle - osiloskop)|}{Duty\ Cycle} \right) \times 100\% \quad (14)$$

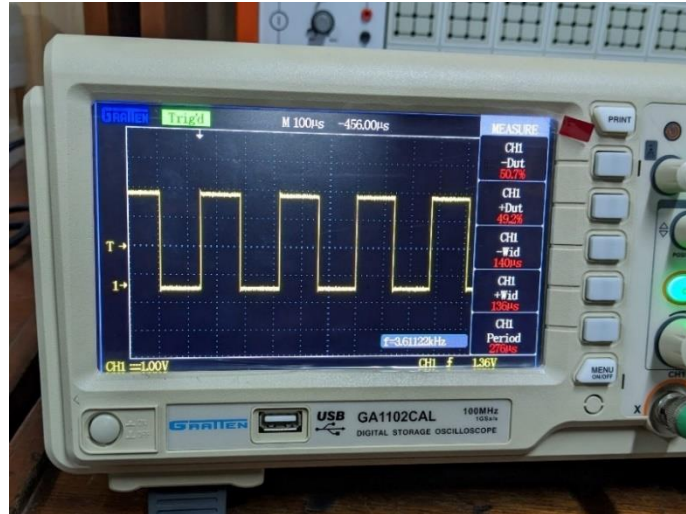
Tabel 4.2 Hasil pengujian *Duty Cycle* PWM

No.	PWM	<i>Duty Cycle</i> (%)	Osiloskop (%)	Selisih	<i>Error</i> (%)
1	26	10	9.42	0.58	5.80
2	51	20	19.56	0.44	2.20
3	77	30	29.71	0.29	0.97
4	102	40	39.13	0.87	2.17
5	128	50	49.27	0.73	1.46
6	153	60	59.42	0.58	0.97
7	179	70	69.56	0.44	0.63
8	204	80	78.98	1.02	1.28
9	230	90	89.13	0.87	0.97
10	255	100	99.27	0.73	0.73
Rata-rata				0.655	1.72

Tabel 4.2 menunjukkan hasil pengujian *Duty Cycle* PWM menggunakan osiloskop yang diuji sebanyak 10 kali yang terlampir pada Lampiran Dokumentasi Osiloskop. Sistem yang telah dibuat berhasil berjalan dengan baik dan menghasilkan rata-rata nilai *error duty cycle* PWM sebesar 1.72%.

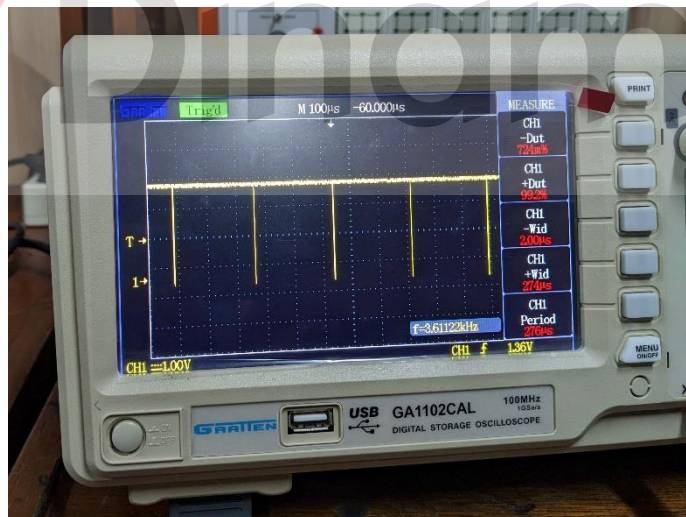
Gambar 4.1 Hasil *Duty Cycle* 10%

Ini merupakan dokumentasi hasil pengujian *duty cycle* dengan *input* nilai PWM 26 atau 10% dari maksimal PWM 255. Maksud dari *duty* 10% adalah waktu aktif dari tegangan yang dialirkan ke Motor DC. Driver mengirimkan tegangan 24V utuh selama 10% dari periode satu gelombang.



Gambar 4.2 Hasil Duty Cycle 50%

Ini merupakan dokumentasi hasil pengujian *duty cycle* dengan *input* nilai PWM 128 atau 50% dari maksimal PWM 255. Maksud dari duty 50% adalah waktu aktif dari tegangan yang dialirkan ke Motor DC. Driver mengirimkan tegangan 24V utuh selama 50% dari periode satu gelombang.



Gambar 4.3 Hasil Duty Cycle 100%

Ini merupakan dokumentasi hasil pengujian *duty cycle* dengan *input* nilai PWM 255 atau 100% dari maksimal PWM 255. Maksud dari duty 100% adalah waktu aktif dari tegangan yang dialirkan ke Motor DC. Driver mengirimkan tegangan 24V utuh selama 100% dari periode satu gelombang.

4.3 Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem dilakukan untuk memastikan sistem yang telah dibuat berjalan dengan baik dan berjalan sesuai dengan algoritma yang telah dibuat. Berikut algoritma yang digunakan pada sistem yang digunakan:

Tabel 4.3 Algoritma Motor DC

Kemiringan Sensor	Aksi Motor DC
Jika sensor kurang dari 1 dan lebih dari -1	Berhenti dan Reset 0
Jika sensor lebih dari 1 dan kurang dari 15	Maju
Jika sensor kurang dari -1 dan lebih dari -7	Mundur
Jika sensor kurang dari -7 dan lebih dari 15	Berhenti

4.3.1 Alat Dan Bahan Yang Digunakan Pengujian Keseluruhan Sistem

Berikut merupakan alat dan bahan yang digunakan dalam proses pengujian keseluruhan sistem.

1. IMU Sensor
2. Motor driver BTS7960
3. Teensy 4.1
4. Motor DC 24V
5. Power Supply 24V
6. Laptop

4.3.2 Prosedur Pengujian Keseluruhan Sistem

Berikut merupakan tata cara dalam proses pengujian keseluruhan sistem.

1. Merangkai alat seperti Gambar 3.2.
2. Menghubungkan laptop dengan Teensy 4.1.
3. Mengunggah program lampiran 3 pada Arduino IDE ke Teensy 4.1.
4. Membuka serial monitor pada software Arduino IDE.
5. Mengamati perputaran baling-baling dengan memiringkan sensor sesuai arah sumbu Y.
6. Membandingkan nilai PWM dengan perhitungan manual interpolasi linear.

$$error = \left(\left| \frac{PWM(\text{motor dc}) - PWM(\text{perhitungan})}{PWM(\text{perhitungan})} \right| \right) \times 100\% \quad (15)$$

Tabel 4.4 Pengujian keseluruhan sistem

IMU Sensor	PWM		Selisih	Error (%)	Gerak Motor
	Motor DC	Perhitungan			
1.18	3.30	3.28	0.02	0.61%	Maju
2.59	29.01	28.96	0.05	0.17%	
3.37	43.32	43.17	0.15	0.35%	
4.75	68.23	68.30	0.07	0.10%	
5.31	78.47	78.50	0.03	0.04%	
6.30	96.62	96.54	0.08	0.08%	
7.20	112.86	112.93	0.07	0.06%	
8.40	134.68	134.79	0.11	0.08%	
9.15	148.34	148.45	0.11	0.07%	
10.25	168.37	168.48	0.11	0.07%	
11.50	191.17	191.25	0.08	0.04%	
12.43	208.26	208.19	0.07	0.03%	
13.29	223.54	223.85	0.31	0.14%	
14.91	253.34	253.36	0.02	0.01%	
15.23	259.19	258.99	0.2	0.08%	
0.37	-11.23	-11.48	0.25	2.18%	Berhenti
2.46	26.51	26.59	0.08	0.30%	Maju
-1.51	-10.65	-10.80	0.15	1.39%	Mundur
-2.24	-26.14	-26.25	0.11	0.42%	
-3.08	-43.94	-44.03	0.09	0.20%	
-4.21	-68.12	-67.95	0.17	0.25%	
-5.43	-93.63	-93.77	0.14	0.15%	
-6.50	-116.5	-116.42	0.08	0.07%	
Rata – rata			0.00304	0.16%	

Tabel 4.4 menunjukkan hasil dari pengujian keseluruhan yang telah dilakukan. Dari hasil *angle* yang diterima melalui sensor *gyroscope* kemudian masuk ke dalam rumus interpolasi linear hingga menghasilkan nilai PWM. Adanya nilai PWM negatif pada tabel hanya untuk menunjukkan kondisi motor berjalan ke arah kiri atau IMU sensor pada kemiringan berlawanan arah jarum jam. Nilai PWM yang ditampilkan pada serial monitor dibandingkan dengan perhitungan manual, dan sistem berjalan sesuai dengan algoritma yang telah ditentukan. Nilai *error* yang di dapat dari hasil pengujian keseluruhan sistem yaitu 0.16%.

BAB V PENUTUP

5.1 Kesimpulan

Adapun hasil kesimpulan dari Tugas Akhir ini adalah sebagai berikut:

1. *Prototype self balancing* dirancang dengan menggunakan Sensor *Gyroscope* sebagai alat input. nilai sensor *gyroscope* diproses oleh mikrokontroler Teensy dengan menggunakan formula Interpolasi Linear. Hasil perhitungan interpolasi merupakan nilai PWM yang akan dikirimkan kepada driver motor. Nilai positif untuk percepatan ke arah kanan dan nilai negatif untuk percepatan ke arah kiri.
2. Pengujian sensor Dfrobot Serial *6-Axis Accelerometer* yang digunakan pada sistem yang dibuat menghasilkan persentase nilai *error* sensor sebesar 4.52%.
3. Pengujian PWM menggunakan osiloskop untuk menentukan hasil keluaran tegangan, menghasilkan persentase nilai *error* sebesar 1.72%.
4. Pengujian keseluruhan sistem yang telah dilakukan berhasil berjalan dengan baik sesuai dengan algoritma yang dibuat menghasilkan persentase nilai *error* sebesar 0.16%.

5.2 Saran

Saran untuk mengembangkan dan menyempurnakan dari alat Tugas Akhir ini, agar benar-benar dapat diterapkan adalah sebagai berikut:

1. Dapat menggunakan sensor tambahan seperti rotary sensor untuk melakukan pengecekan putaran Motor DC.
2. Dapat dikembangkan pada skala yang lebih besar.
3. *Prototype Self Balancing* pada motor DC dengan metode interpolasi linear dapat dikembangkan menjadi *One Wheel Scooter*.

DAFTAR PUSTAKA

- Firman. B. (2016. Agustus). Implementasi Sensor Imu Mpu6050 Berbasis Serial I2c Pada Self-Balancing Robot. *Jurnal Teknologi Technoscientia*. Vol. 9 No. 1. 18-24.
- Hartomo. K. D. (2006). Implementasi Metode Interpolasi Linear Untuk Pembesaran Resolusi Citra. *TEKNOIN*. Vol. 11. No 3. 219-232.
- Ira Puspasari. S. M. (2017). *Otomasi Sistem Hidroponik Wick Pada Pembibitan Tanaman Tomat Ceri*. Stikom Surabaya. Surabaya.
- Jayanti. A. V. (2007. maret 30). Perbandingan Interpolasi Dalam metode Spline. Yogyakarta: Fakultas Matematika dan Ilmu pengetahuan Alam Universitas Sanata Dharma.
- Kho. D. (n.d.). *Pengertian Motor DC dan Prinsip Kerjanya*. Retrieved from Teknik Elektronika: <https://teknikelektronika.com/pengertian-motor-dc-prinsip-kerja-dc-motor/>
- NAFI'. M. Y. (2017). *Rancang Bangun Self-Balancing Pada Inverted Pendulum Menggunakan Control Moment Dengan Double Gyroscope Dan*. Institut Teknologi Sepuluh Nopember. Departemen Teknik Mesin. Surabaya.
- Sahid. (2004-2012). *Pengantar Komunikasi Numerik*.
- Satryo. M. I. (2022). *DETEKSI KECELAKAAN MOBIL LISTRIK MENGGUNAKAN GYROSCOPE SENSOR*. Universitas Dinamika. Surabaya.
- Ukuransatuan.com. (2016. Agustus 20). *Cara Menentukan Nilai antara Dua Data dengan Interpolasi Linear (Interpolasi Garis Lurus)*. Retrieved from Ukuran dan satuan: <https://ukurandansatuan.com/cara-menentukan-nilai-antara-dua-data-dengan-interpolasi-linear.html/>
- Wibowo. Y. C.. & Riyadi. S. (2018. Desember). Analisa Pembebanan pada Motor Brushless DC (BLDC). *Seminar Nasional Instrumentasi, Kontrol dan Otomasi (SNIKO) 2018*.