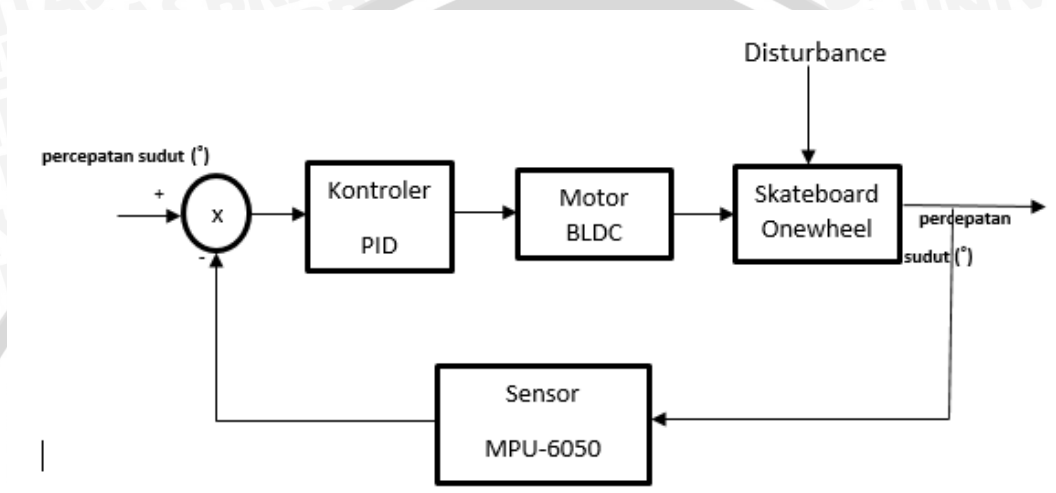


## BAB III METODOLOGI PENELITIAN

### 3.1 Penyusunan Blok Diagram Sistem

Pada perencanaan alat diperlukan perencanaan blok diagram sistem yang dapat menjelaskan sistem secara garis besar dan diharapkan alat dapat bekerja sesuai rencana, berikut blok diagram sistem. Blok diagram dapat dilihat pada Gambar 3.1.



**Gambar 3.1** Diagram Blok Sistem

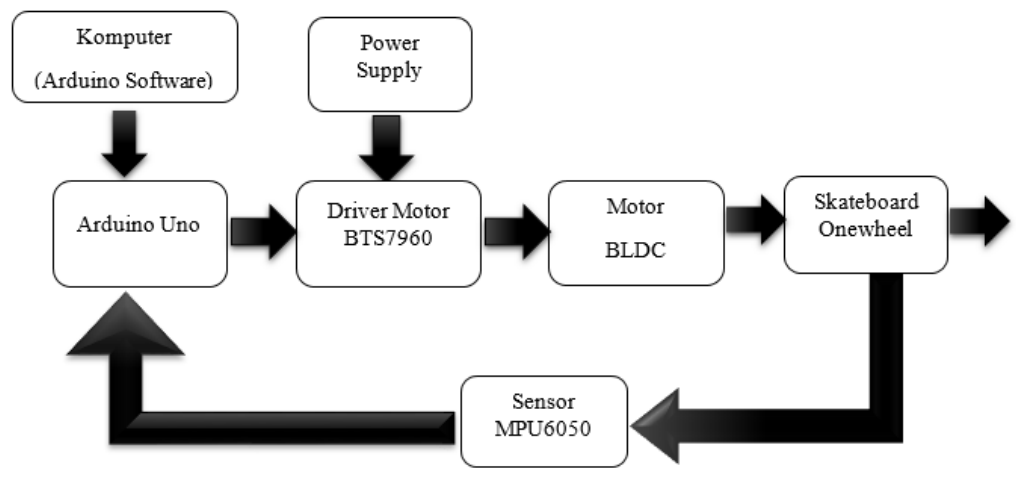
Keterangan dari diagram blok sistem dalam Gambar 3.1 adalah sebagai berikut:

- *Setpoint* sistem adalah nilai keseimbangan dan kecepatan posisi pada skateboard dan kecepatan motor yang dihasilkan oleh karakteristik driver motor dengan tegangan sebesar 24 volt.
- *Skateboard* dengan ukuran 89 cm x 20 cm dan berat 8 kg, sebagai plant pada sistem ini.
- Pusat pengendalian sistem adalah Arduino uno yang memberikan sinyal PWM pada motor BLDC melalui driver mosfet H-Bridge.
- Aktuator menggunakan satu motor BLDC 24V yang berfungsi untuk menyeimbangkan *skateboard* yang mendapat sinyal masukan dari *output* PWM driver mofet H-Bridge.
- Sensor MPU 6050 untuk menentukan kecepatan posisi dan kesimbangan pada skateboard roda satu dengan range yang sudah dilakukan saat penelitian.

- Kontroler yang digunakan adalah Kontrol PID dengan menggunakan Arduino Uno R3 untuk mendapatkan parameter kontroler digunakan metode *hand tuning*.

### 3.2 Pembuatan Perangkat Keras

Dalam perancangan perangkat keras meliputi rangkaian mikrokontroler yang didalamnya terdapat sensor MPU 6050, regulator 7850, dan buzzer. Setelah itu terdapat rangkaian driver BTS7960 arduino. Untuk lebih jelasnya seperti gambar 3.2



**Gambar 3.2** Skema pembuatan perangkat keras

Pada Gambar 3.2 menjelaskan tentang diagram blok perancangan sistem yang dilakukan pada *interface* dan di *skateboard one wheel*. Dalam *skateboard one wheel* terdapat perancangan mulai dari mikrokontroler, penggunaan driver motor, perancangan baterai, perancangan komunikasi yang terjadi pada *interface* matlab dan *skateboard one wheel* yang digunakan untuk melihat data grafik kontroler PID.

Implementasi *Skateboard one wheel* yang digunakan yaitu *microcontroller* atau *Arduino uno*, sensor nya menggunakan MPU 6050, dan dan motor drivernya menggunakan mosfet yang membutuhkan tegangan 24 volt, berikut ini adalah implementasi perangkat keras dapat dilihat pada gambar 3.3.



**Gambar 3.3** *Skateboard one wheel*

Catu daya pada mikrokontroler dan sensor menggunakan daya sebesar 5 volt dan catu daya yang digunakan motor sebesar 24 volt. Dikarenakan motor menggunakan daya yang besar agar bisa bergerak. Agar alat tersebut bisa berjalan ground pada power 5 volt dan 24 volt harus menyambung. Pada sistem ini terdapat dua kondisi, kondisi pertama yaitu inisialisasi push 1. Ketika sensor lebih dari yang diinginkan maka roda pada *skateboard* bergerak maju, jika nilai pembacaan sensor lebih dari nilai standard yang ditetapkan jika sensor lebih dari yang diinginkan maka roda *skateboard* bergerak mundur. Setelah itu kondisi kedua yaitu inisialisasi push 2, *skateboard one wheel* akan bergerak maju ketika sensor tidak lebih dari yang diinginkan.

### 3.3 Spesifikasi Alat

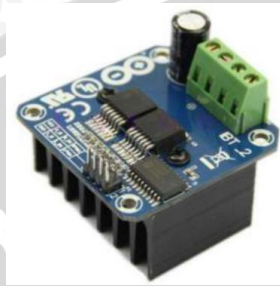
Spesifikasi alat yang meliputi komponen-komponen pendukung pada skripsi ini adalah sebagai berikut:

1. Komputer atau PC yang sudah terinstall *software* Arduino dan MATLAB 2014a.
2. Perangkat kontroler yang digunakan adalah Arduino Uno R3. Mikrokontroler Arduino Uno dapat dilihat pada Gambar 3.4 berfungsi sebagai penyimpan algoritma kontroler sistem, pengendali driver motor, dan menerima serta mengolah data yang dikirimkan sensor ping ultrasonik (sensor jarak). Mikrokontroler menggunakan catu tegangan sebesar 5 Volt yang didapat dari komputer atau PC.



**Gambar 3.4** Mikrokontroler Arduino Uno.

3. Driver yang digunakan adalah modul yang menggunakan BTS7960 yang ditunjukkan pada Gambar 3.6. Driver motor berfungsi sebagai pengali tegangan dimana tegangan masukan dari mikrokontroler Arduino dikali dengan sebuah nilai *gain* atau penguatan tertentu menjadi tegangan keluaran driver motor yang digunakan untuk mencatu kawat nikelin. Catu tegangan driver motor didapat dari *power supply unit* (PSU) dengan tegangan maksimal 40 V dan arus maksimalnya sebesar 43 A.

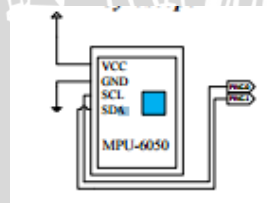


**Gambar 3.5** Driver BTS7960 43A.

4. Sensor MPU-6050 membutuhkan tegangan kerja 3,3V. Tetapi modul ini telah dilengkapi dengan regulator tegangan 3,3V sehingga bisa langsung dihubungkan ke tegangan 5V. Sensor ini mempunyai dua buah keluaran yaitu SCL dan SDA masing-masing dihubungkan ke PC.0 dan PC.1 ditunjukkan pada gambar 3.6 dan gambar 3.7.



**Gambar 3.6** Sensor MPU-6050 3-Axis Accelerometer+ 3-Axis Gyroscope



**Gambar 3.7** Konfigurasi Sensor MPU-6050

5. Motor BLDC

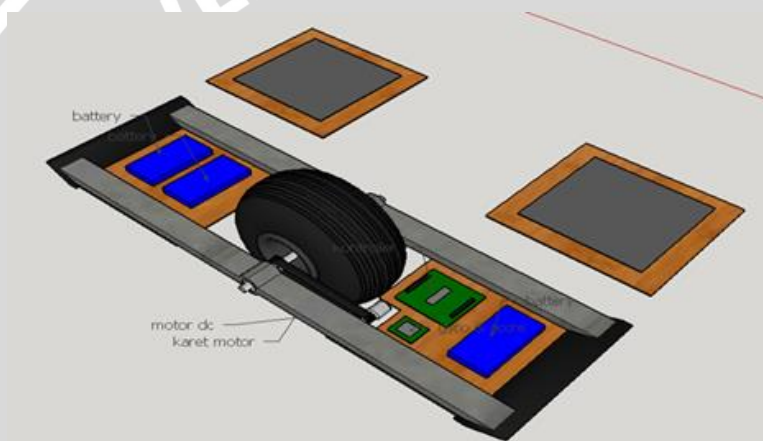
Brushless DC (BLDC) motor adalah pilihan ideal untuk aplikasi yang memerlukan keandalan yang tinggi, efisiensi tinggi, dan tinggi power-to-volume rasio. Secara umum, motor BLDC dianggap motor performa tinggi yang mampu memberikan jumlah besar torsi pada rentang kecepatan yang luas. BLDC motor adalah turunan dari motor DC

yang paling umum digunakan, DC disikat motor, dan mereka berbagi sama torsi dan karakteristik kinerja kecepatan kurva. Perbedaan utama antara keduanya adalah penggunaan kuas. BLDC motor tidak memiliki sikat (maka nama "*brushless DC*") dan harus secara elektronik commutated.



**Gambar 3.8** Motor BLDC

6. Plant berupa skateboard onewheel



**Gambar 3.9** Plant Skateboard One wheel

### 3.4 Prinsip Kerja Sistem

Prinsip kerja sistem adalah sebagai berikut:

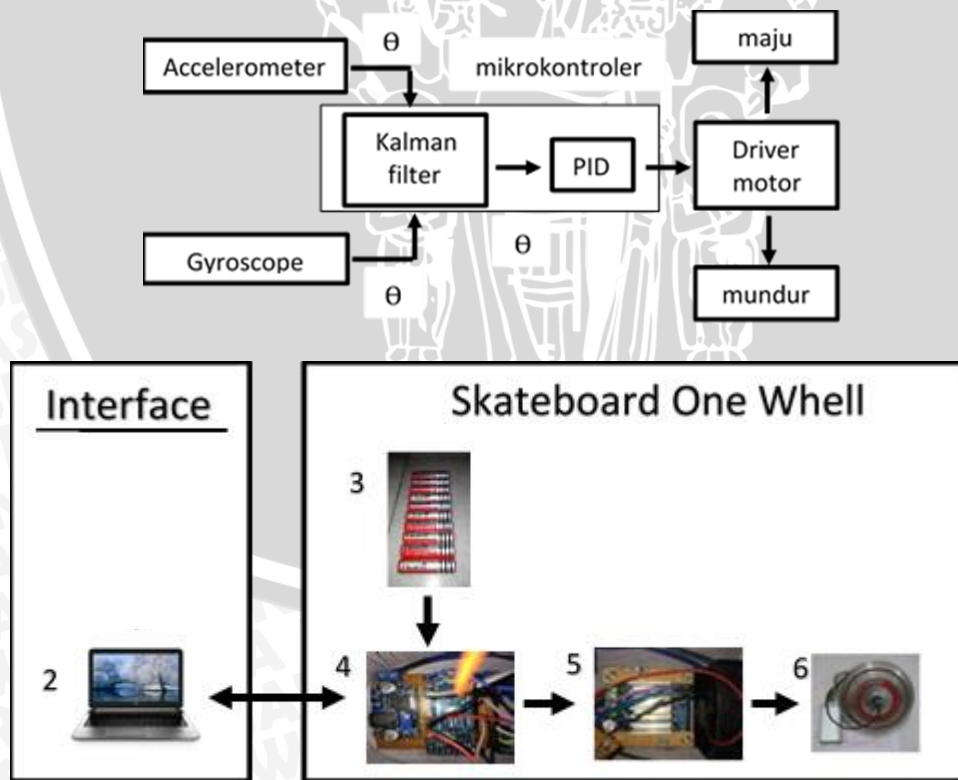
1. Catu daya atau baterai dengan tegangan 24 VDC sebagai suplai rangkaian driver motor dan motor BLDC.
2. Sistem diberi catu daya. Catu daya sebesar 5V dari mikrokontroler digunakan untuk mencatu sensor MPU 6050, dan mikrokontroler arduino uno.
3. Sinyal kontrol dari Arduino uno masuk ke driver motor mosfet H-BRIDGE. Driver motor berfungsi untuk menguatkan sinyal yang dihasilkan mikrokontroler Arduino uno dari 0-5 V menjadi 0-24 V.
4. Sinyal keluaran rangkaian penguat sebagai masukan Arduino uno yang kemudian diproses menggunakan kontroler PID.

5. Keluaran Arduino Uno berupa sinyal pulse width modulation (PWM) *Pulse Width Modulation* diberikan ke driver motor.
6. Menggunakan aktuator berupa motor BLDC yang akan menghasilkan putaran untuk menghasilkan gerakan pada *skateboard one wheel*.
7. Mencari fungsi alih plan dengan menggunakan metode *hand tuning*.
8. Mengimplementasikan hasil desain perancangan pada sistem.

Sistem yang telah direalisasikan ini dapat menyeimbangkan skateboard. Hal ini dapat dilakukan dengan cara mendeteksi kemiringan dari papan *skateboard* kemudian dari data kemiringan ini dikirim ke motor BLDC untuk kembali menyeimbangkan *skateboard*.

Kalman Filter diimplementasikan pada *skateboard*, dan dapat menemukan model yang akurat. Pemodelan didekati dengan metoda linear agar mudah diimplementasi. Kalman filter dapat memanfaatkan data accelerometer untuk menghilangkan drift dari keluaran gyroscope. Dalam proses, noise dari accelerometer juga akan diminimalkan bahkan dihilangkan.

Metode kalman filter dipenelitian ini hanya digunakan sesuai library yang ada pada arduino, tidak sampai mempelajari perhitungan matematik dari penfilteran kalman filter.



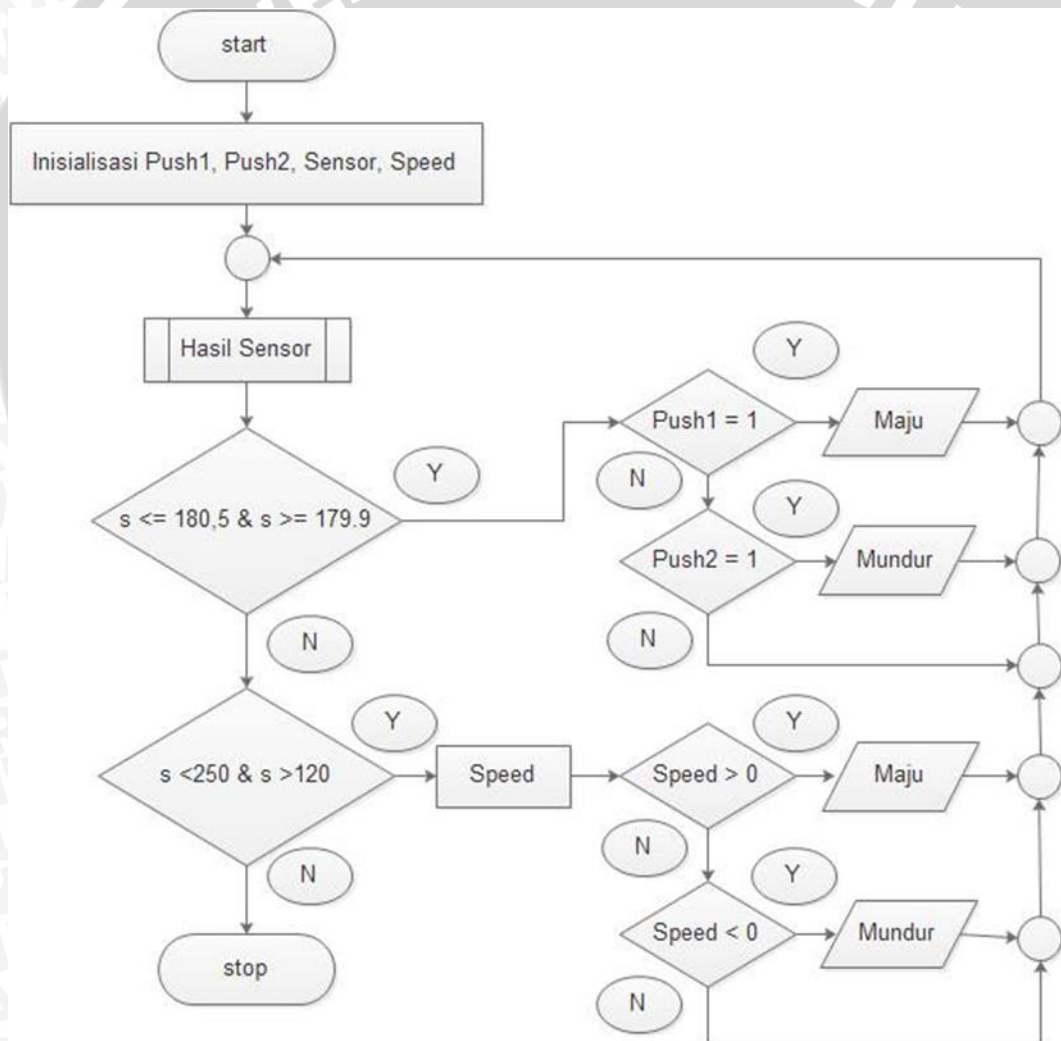
Gambar 3.10 Diagram Blok Sistem *Skatebord One Wheel*

Keterangan:

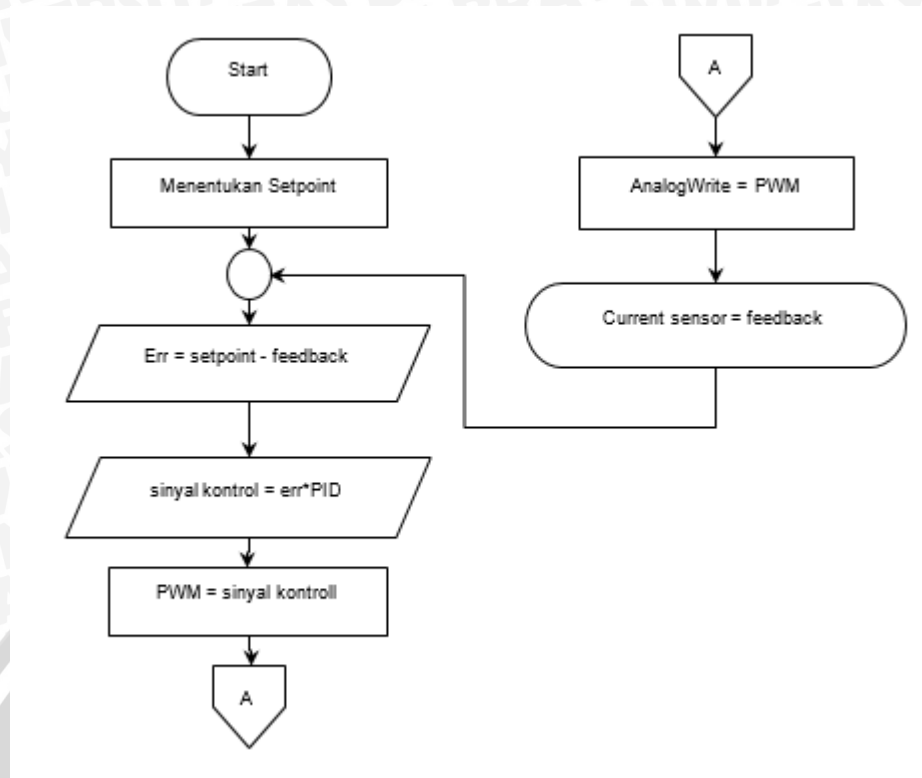
1. Laptop
2. Baterai
3. Mikrokontroler Arduino Uno dan Sensor MPU 6050
4. Driver Motor
5. Motor BLDC

### 3.5 Flowchart

Alur program secara umum atau keseluruhan akan dijelaskan pada gambar flowchart 3.11.



Gambar 3.11 Flowchart sistem



**Gambar 3.12** Flowchart Keseluruhan Sistem dengan Kontroler PID

### 3.6 Pengujian Driver BTS7960 43A

#### a. Tujuan

Mengetahui kinerja dan respon rangkaian driver motor BTS7960 43A dengan membandingkan output tegangan efektif driver dengan masukan *dutycycle* sinyal PWM yang diberikan oleh Arduino Uno.

#### b. Peralatan yang digunakan

1. Power Supply.
2. Driver BTS7960 43A.
3. Arduino Uno.
4. Multimeter.
5. Seperangkat komputer.
6. Kabel penghubung.

#### c. Prosedur pengujian

1. Tulis listing program pada software arduino untuk mengubah *dutycycle*.
2. Hubungkan *output* tegangan *Power Supply Unit* pada input tegangan referensi driver BTS7960 43A.



3. Hubungkan *input* tegangan driver BTS7960 43A dengan pin output PWM di Arduino Uno.
4. Hubungkan *output* tegangan driver BTS7960 43A dengan multimeter.
5. Atur *dutycycle* sinyal PWM pada Arduino Uno dengan nilai 0%-100%.
6. Amati dan catat hasil pembacaan multimeter di setiap kenaikan 5%.

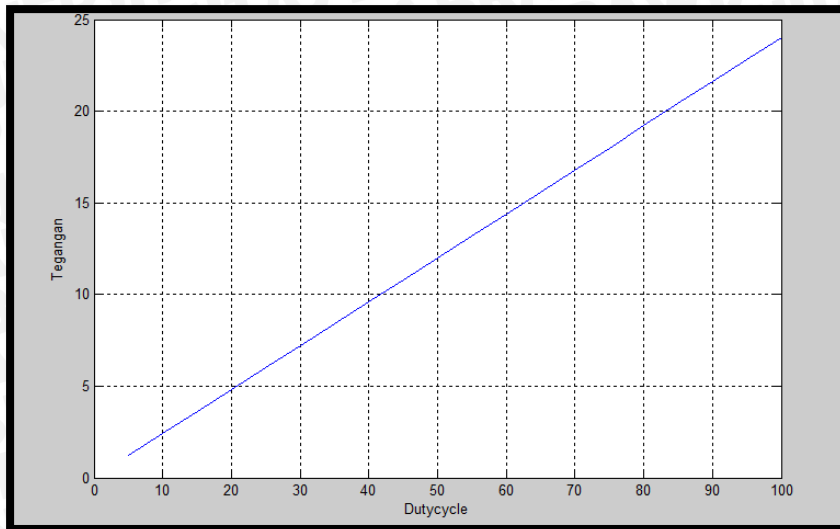
d. Hasil Pengujian

Data pengujian Driver BTS7960 ditunjukkan pada Tabel 3.1.

NO	Dutycycle (%)	PWM	Tegangan (Volt)
1	5	12.75	1.2
2	10	25.5	2.4
3	15	38.25	3.6
4	20	51	4.8
5	25	63.75	6
6	30	76.5	7.2
7	35	89.25	8.4
8	40	102	9.6
9	45	114.75	10.8
10	50	127.5	12
11	55	140.25	13.2
12	60	153	14.4
13	65	165.75	15.6
14	70	178.5	16.8
15	75	191.25	18
16	80	204	19.2
17	85	216.75	20.4
18	90	229.5	21.6
19	95	242.25	22.8
20	100	225	24

**Tabel 3.1** Hasil pengujian Driver BTS7960 43A.

Berdasarkan Tabel 3.1, akan didapatkan kurva *output* tegangan driver (Volt) terhadap masukan *dutycycle* sinyal PWM, seperti yang ditunjukkan pada Gambar 3.13.



**Gambar 3.13** Grafik perubahan tegangan *output* driver BTS7960 43A terhadap *input* *dutycycle*.

### 3.2.2 Karakteristik motor BLDC 24v

#### 1. Tujuan

Mengetahui karakteristik motor BLDC 24v untuk mengetahui kecepatan putaran motor dc dilihat dari perbandingan antara masukan berupa *dutycycle* (%) dengan nilai PWM dari Arduino uno.

#### 2. Peralatan yang digunakan

- Power Supply Unit (PSU)
- Personal Computer (PC)
- Arduino uno
- Motor BLDC
- Kabel jumper
- Driver BTS7960
- Multimeter

#### 3. Prosedur pengujian

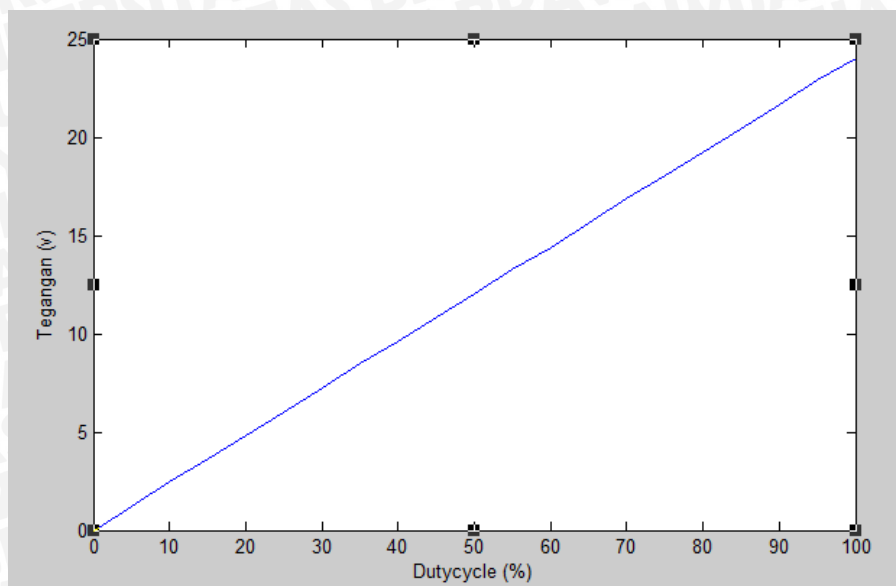
- Hubungkan *output* tegangan *Power Supply Unit* (PSU) pada *input* tegangan referensi *driver* motor BTS 7960.
- Hubungkan *input* tegangan *driver* motor BTS 7960 dengan pin *output* PWM di Arduino uno.
- Hubungkan *output* tegangan *driver* motor BTS 7960 dengan multimeter.
- Hubungkan *output* *driver* BTS7960 dengan motor BLDC.

5. Tulis listing program pada software arduino untuk mengubah *dutycycle*.
  6. Atur *duty cycle* sinyal PWM pada Arduino uno dengan nilai 0%-100%.
  7. Amati dan catat hasil pembacaan multimeter disetiap kenaikan 5%.
4. Hasil pengujian

Hasil pengujian karakteristik motor BLDC 24v dalam dilihat dalam Tabel 3.2 dan Gambar 3.14.

**Tabel 3.2** Hasil pengujian motor BLDC 24v

No.	Duty cycle (%)	PWM	Vout (Data)
1	0	0	0
2	5	12.75	1.22
3	10	25.5	2.44
4	15	38.25	3.67
5	20	51	4.8
6	25	63.75	6.02
7	30	76.5	7.24
8	35	89.25	8.47
9	40	102	9.6
10	45	114.75	10.82
11	50	127.5	12.04
12	55	140.25	13.27
13	60	153	14.4
14	65	165.75	15.62
15	70	178.5	16.84
16	75	191.25	18.07
17	80	204	19.2
18	85	216.75	20.42
19	90	229.5	21.64
20	95	242.25	22.87
21	100	255	24

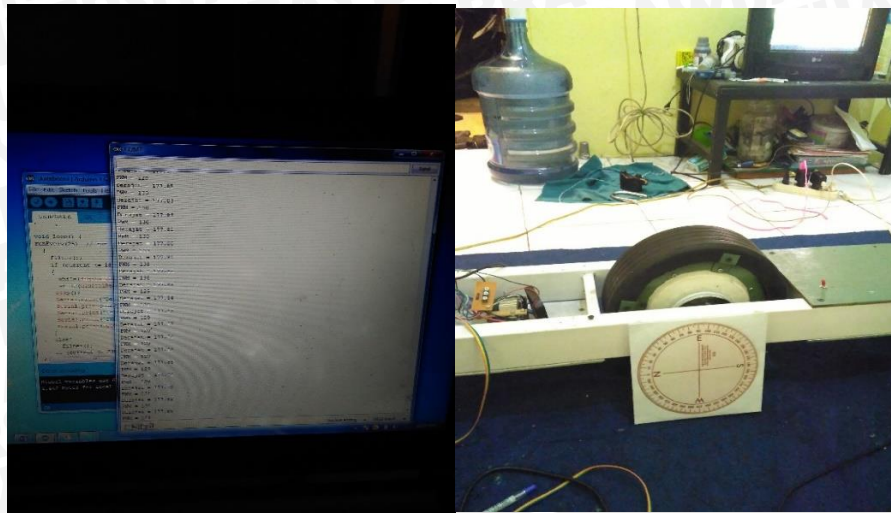


**Gambar 3.14** Grafik perubahan tegangan rata-rata *output* Driver BTS7960 43A terhadap *duty cycle* dengan menggunakan beban motor BLDC.

### 3.7 Pengujian keseluruhan akurasi sensor terhadap PWM dan RPM

Untuk melakukan pengujian sensor yang diuji adalah perubahan sudut *gyroscope* dan *accelerometer* yang ada pada sensor MPU6050. Pengujian *gyroscope* dan *accelerometer* bertujuan untuk mengetahui tingkat keakuratan dari *gyroscope* dan *accelerometer* dalam membaca perubahan sudut skateboard.

Sensor terpasang pada skateboard dan terhubung dengan mikrokontroler arduino nano dan kemudian data dari keluaran sensor akan dibuat input dan keluarannya adalah nilai pwm. Kemiringan skateboard dapat diubah – ubah sesuai dengan papan sudut yang sudah disediakan.



**Gambar 3.15** Cara mendapatkan data sensor MPU6050

Perubahan sudut yang terjadi akan ditampilkan lewat serial monitor arduino uno pada komputer. Cara pengambilan data pada sensor MPU6050. Gambar 3.15 menunjukkan cara pengambilan data pada sensor MPU6050 dan pwm. Pada tabel 3.3 menunjukkan hasil pengujian pada sensor MPU6050.

**Tabel 3.3** Data hasil pengujian keseluruhan

No	Nilai sudut sensor (°)	Nilai sudut busur (°)	Nilai PWM sensor	PWM Terhadap busur	Nilai RPM sensor	RPM Terhadap busur	Akurasi keseluruhan (%)
1	166.37	166	255	254	1000	996	99.6%
2	171.57	172	223	220	874	862	98.62%
3	176.56	177	153	151	600	592	98.66%
4	180.14	180	70	64	274	250	91.24%
5	182.30	182	-141	-138	-552	-541	98.01%
6	185.32	185	-178	-174	-698	-682	97.71%
7	194.36	194	-255	-254	-1000	-996	99.6%
Rata- rata							97.63%

Dari tabel 3.3 terlihat ada dua jenis nilai data yaitu nilai terhadap sensor dan nilai terhadap busur. Yang dimaksud nilai terhadap sensor yaitu nilai keseimbangan pada sensor yang dikeluarkan oleh serial monitor arduino IDE. Sedangkan nilai terhadap busur yaitu pengukuran keseimbangan badan pada *skateboard*. Nilai RPM mengikuti nilai PWM yang dihasil dengan rumus sebagai berikut:

Nilai RPM didapat dari perhitungan  $RPM = \frac{PWM}{255} \times 1000$

Contohnya  $RPM = \frac{153}{255} \times 1000 = 600 \text{ RPM}$

Pada persentase akurasi keseluruhan didapatkan dari nilai RPM terhadap busur dan nilai RPM terhadap sensor dengan rumus sebagai berikut:

Akurasi didapat dari perhitungan  $akurasi = \frac{rpm \text{ terhadap busur}}{rpm \text{ terhadap sensor}} \times 100\%$

Contohnya perhitungan akurasi sensor  $akursi = \frac{682}{698} \times 100\% = 97.71\%$

### 3.6 Penentuan nilai penguatan (Gain)

Fungsi alih plant untuk mendapatkan nilai karakteristik statis gain keadaan mantap, dengan cara plant di rancang secara *open loop* dan diberikan unit step masukan *duty cycle* 0%–100% dan di lihat bagaimana respon keadaan mantapnya. Untuk menemukan parameter kontroler PID digunakan metode *hand tuning*, dimana plant diberi tegangan masukan ke plant dan dilihat bagaimana hasilnya.

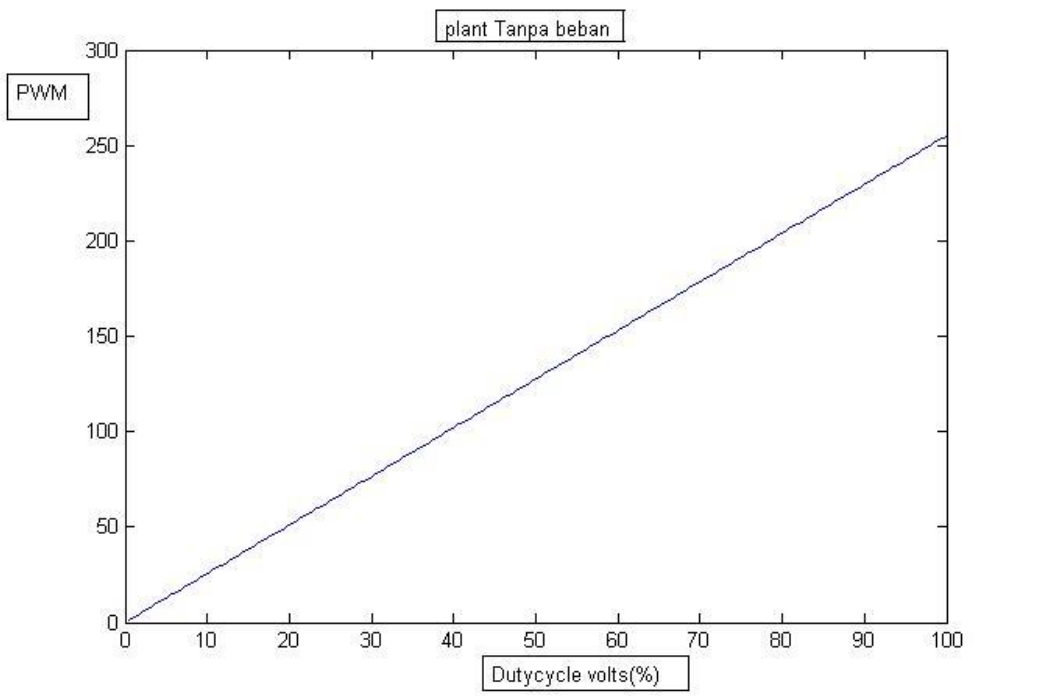
#### a. Peralatan yang digunakan

1. Power battery.
2. Arduino Uno R3.
3. Seperangkat komputer.
4. Kabel penghubung.
5. Driver motor.
6. Motor BLDC 24 volt.

#### b. Prosedur Pengujian

1. Membuat listing program dengan mengatur *dutycycle* 0%-100%.
2. Atur *input* pada driver motor di software Arduino berupa *dutycycle*.
3. Driver motor mendapatkan catu dari *power supply unit* 0-24V.

4. Hubungkan driver motor dengan pin di Arduino Uno R3.
5. Hubungkan *output* driver motor dengan motor BLDC 24V.
6. Plant berupa skateboard untuk mengetahui respon *steadystate* dalam keadaan tanpa beban.



**Gambar 3.15 Grafik Karakteristik statis gain keadaan tanpa beban**

$$k = \frac{\Delta y}{\Delta u} = \frac{y_{70} - y_{40}}{u_{70} - u_{40}} = \frac{274.51 - 156.86}{70 - 40} = 3.92$$

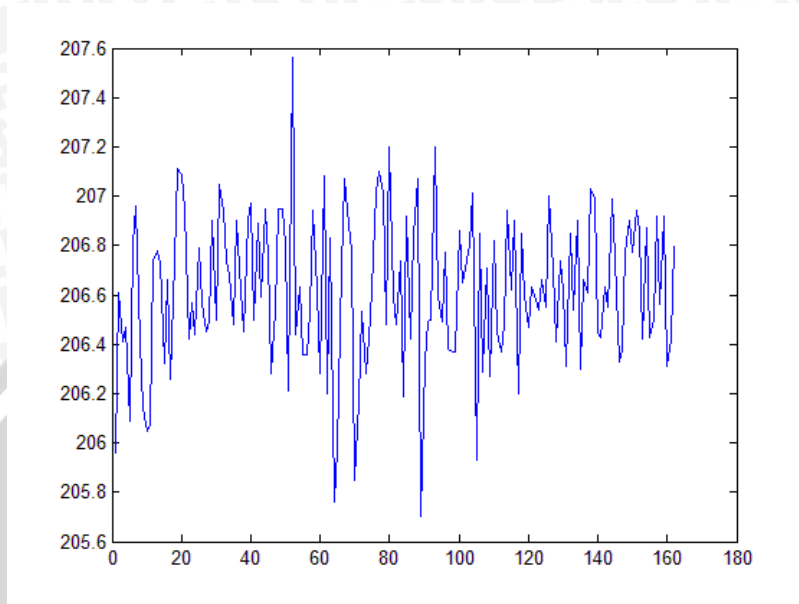
Untuk memenuhi tujuan performansi *loop* yang diinginkan, maka perlu ditambahkan kontroler pada sistem tersebut. Kontroler yang digunakan pada penelitian ini adalah kontroler Proporsional, Integratif dan Derivatif atau disingkat PID. Penelitian ini menggunakan salah satu *tunning* dari PID yaitu *Hard tuning* yaitu dengan cara memasukan nilai proporsional (Pi), integratif (Ki), dan derivatif (Kd). Setiap nilai mempunyai pengaruh yang berbeda seperti nilai Pi mengurangi *error steady state* (waktu awal), menurunkan *rise time* (waktu naik), dan meningkatkan *overshoot* (titik maksimum), selanjutnya nilai ki menghilangkan *error steady state* (waktu awal), menurunkan *rise time* (waktu naik) dan meningkatkan *overshoot* (titik maksimum), dan yang terakhir ki menurunkan *overshoot* (titik maksimum) atau memberi efek redaman.

### 3.6 Pengujian Kalman Filter

Pengujian kalman filter dilakukan untuk mengetahui hasil data dari sensor MPU6050 dengan menggunakan metode kalman filter dan tidak menggunakan metode

kalman filter. Dalam melakukan pengujian, dibandingkan antara keluaran data sudut komputasi kalman filter, nilai komputasi *original* data sensor, dan sudut busur. Untuk lebih jelasnya dapat dilihat pada gambar masing-masing berikut.

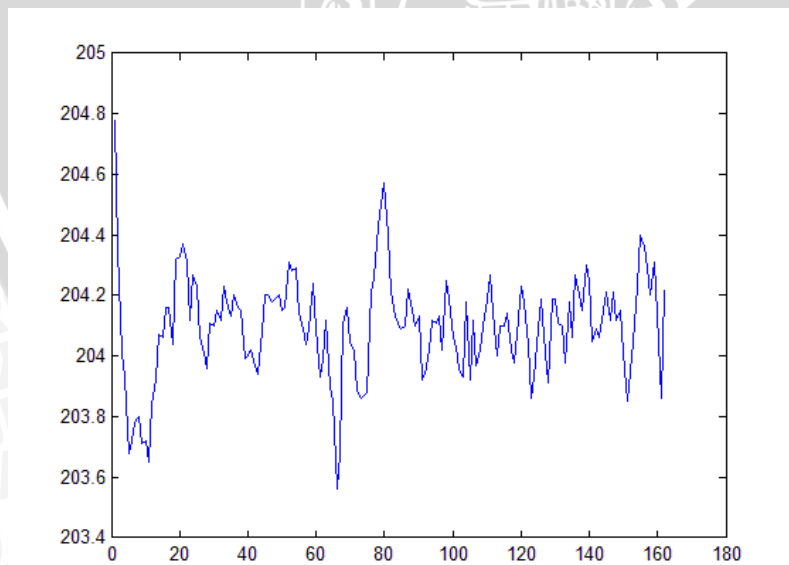
a. Pengujian Data Original terhadap Sensor MPU6050



Gambar 3.16 Grafik Data Original

Pada data ini, grafik ditemukan terlalu banyaknya noise. Terlihat grafikmya seakan naik turun tidak stabil.

b. Pengujian Data Kalman Filter terhadap Sensor MPU6050



Gambar. 3.17 Grafik Data Kalman Filter

Pada data ini, setelah dilakukan pengujian Kalman filter grafik menunjukkan pengurangan noise.