

**KENDALI KECEPATAN MOTOR PADA ELECTRIC POWER
STEERING (EPS) BERBASIS MIKROKONTROLER ATMEGA328**

LAPORAN TUGAS AKHIR

LAPORAN INI DISUSUN UNTUK MEMENUHI SALAH SATU SYARAT
MEMPEROLEH GELAR S1 PADA PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG SEMARANG



DISUSUN OLEH:

SAMSUL WAHAB

NIM 30601800039

**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG
SEMARANG**

2022

**MOTOR SPEED CONTROL IN ELECTRIC POWER STEERING
(EPS) BASED ON ATMEGA328 MICROCONTROLLER**

FINAL PROJECT REPORT

THIS REPORT WAS PREPARED TO MEET ONE OF THE REQUIREMENTS
FOR OBTAINING AN S1 DEGREE IN THE ELECTRICAL ENGINEERING
STUDY PROGRAM, FACULTY OF INDUSTRIAL TECHNOLOGY
SULTAN AGUNG ISLAMIC UNIVERSITY SEMARANG



COMPILED BY:

SAMSUL WAHAB

NIM 30601800039

**ELECTRICAL ENGINEERING STUDY PROGRAM
FACULTY OF INDUSTRIAL TECHNOLOGY
SULTAN AGUNG ISLAMIC UNIVERSITY
SEMARANG**

2022

LEMBAR PENGESAHAN PEMBIMBING

Laporan Tugas Akhir dengan judul “KENDALI KECEPATAN MOTOR PADA ELECTRIC POWER STEERING (EPS) BERBASIS MIKROKONTROLER ATMEGA328” ini disusun oleh:

Nama : Samsul Wahab
NIM : 30601800039
Program Studi : Teknik Elektro

Telah disahkan oleh dosen pembimbing pada

Hari : Rabu
Tanggal : 11 Januari 2023

Pembimbing I

Pembimbing II


Dr. Bustanul Arifin, ST., MT.

NIDN. 0614117701


Dr. Ir. Agus Adhi Nugroho, MT.

NIDN. 0628086501

Mengetahui

Ketua Program Studi Teknik Elektro


Dr. Puji Hastuti, ST., MT.

NIDN. 0607018501

LEMBAR PENGESAHAN PENGUJI

Laporan Tugas Akhir dengan judul “KENDALI KECEPATAN MOTOR PADA ELECTRIC POWER STEERING (EPS) BERBASIS MIKROKONTROLER ATMEGA328” ini telah dipertahankan di depan dosen penguji pada:

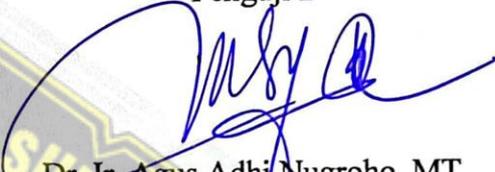
Hari : Rabu

Tanggal : 28 Desember 2022

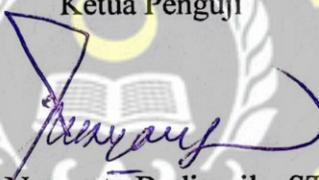
Penguji 1


Dr. Bustanul Arifin, ST., MT.
NIDN. 0614117701

Penguji 2


Dr. Ir. Agus Adhi Nugroho, MT.
NIDN. 0628086501

Ketua Penguji


Dr. Eka Nuryanto Budisusila, ST., MT.
NIDN. 0619107301

UNISSULA
جامعة سلطان أبو جوح الإسلامية

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini

Nama : Samsul Wahab

Nim : 30601800039

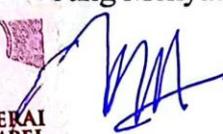
Judul Tugas Akhir : **KENDALI KECEPATAN MOTOR PADA ELECTRIC
POWER STEERING (EPS) BERBASIS
MIKROKONTROLER ATMEGA328**

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat untuk menyelesaikan Pendidikan Strata Satu (S1) Teknik Elektro tersebut adalah asli dan belum pernah diangkat, ditulis maupun dipublikasikan oleh siapapun baik keseluruhan, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila dikemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, Januari 2023

Yang Menyatakan




Samsul Wahab

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Samsul Wahab

NIM : 30601800039

Program Studi : Teknik Elektro

Fakultas : Teknologi Industri

Dengan ini menyatakan Karya Ilmiah berupa Tugas Akhir dengan judul:
**KENDALI KECEPATAN MOTOR PADA ELECTRIC POWER STEERING
(EPS) BERBASIS MIKROKONTROLER ATMEGA328**

Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan hak bebas royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan di internet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran hak cipta/plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan Agung.

Semarang, Januari 2023

Yang Menyatakan


Samsul Wahab

1000
TTL
20
METRBA1
TEMPEL
9Dj191AKX204515567

PERSEMBAHAN

Pertama,

Kepada Allah SWT yang telah memberikan rahmat, hidayah dan karunianya serta kesehatan, sehingga dapat diselesaikannya tugas akhir ini.

Kedua,

Kepada kedua orang tua dan adik – adik saya yang telah memberikan dukungan moral maupun materi serta selalu mendo'akan disetiap shalatnya.

Ketiga,

Kepada sahabat – sahabat saya yang selalu memberikan dukungan dan motivasi dalam pembuatan tugas akhir serta penyusunan laporan tugas akhir.



MOTTO

“Jangan terlalu ambil hati dengan ucapan seseorang, kadang manusia punya mulut tapi belum tentu punya pikiran”

-Albert Einstein

“Dunia tetap sama selama ratusan tahun, namun hidup manusia bagaikan embun pagi atau gelembung air. Hanya bersifat sementara”

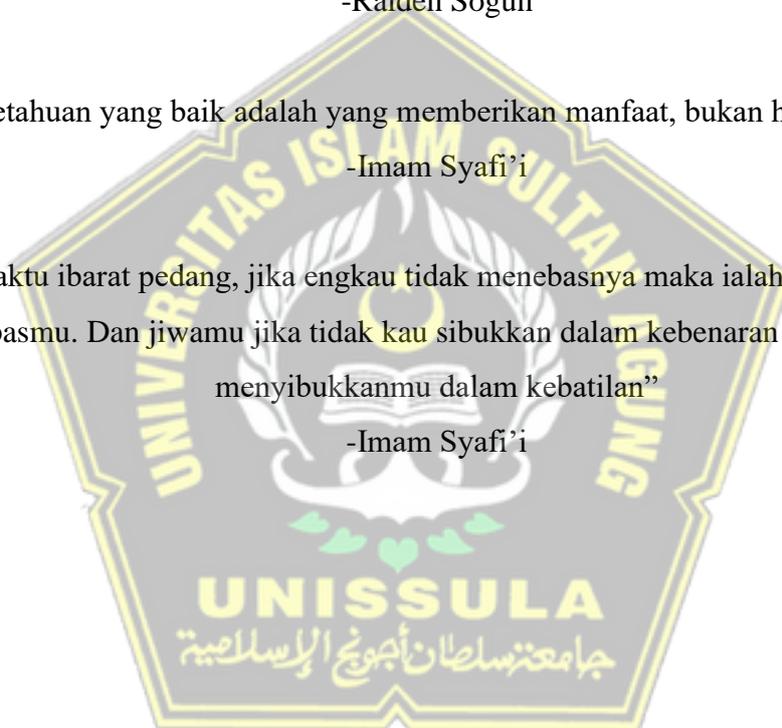
-Raiden Sogun

“Pengetahuan yang baik adalah yang memberikan manfaat, bukan hanya diingat”

-Imam Syafi'i

“Waktu ibarat pedang, jika engkau tidak menebasnya maka ialah yang akan menebasmu. Dan jiwamu jika tidak kau sibukkan dalam kebenaran maka ia akan menyibukkanmu dalam kebatilan”

-Imam Syafi'i



KATA PENGANTAR

Bismillahirrahmanirrahim

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah puja dan puji syukur kehadiran Allah SWT yang telah memberikan nikmat dan rahmat-Nya sehingga masih memberi penulis kesempatan untuk menuntut ilmu dalam keadaan sehat wal'afiat. Shalawat serta Salam tercurahkan kepada baginda Rasulullah Muhammad SAW, semoga kelak kita mendapatkan syafaatnya. Aamiin Yaa Robbaalalamin.

Penyusun Tugas Akhir ini adalah salah satu syarat untuk memperoleh gelar Sarjana pada Fakultas Teknologi Industri di Universitas Islam Sultan Agung Semarang. Dalam penulisan Tugas Akhir ini, tentunya banyak pihak yang telah memberikan bantuan baik dari segi moril maupun materil. Oleh karena itu penulis ingin menyampaikan ucapan terimakasih yang tiada hingganya kepada:

1. Allah SWT yang telah memberikan rahmat dan ridhonya serta memberikan ketabahan, kesabaran dan kelapangan hati serta pikiran dalam menimba ilmu.
2. Kedua orang tua, yakni bapak Suwawi dan ibu Sawiatun yang telah memberikan dukungan baik materil maupun non materil dan tidak pernah berhenti mendo'akan disetiap sholatnya.
3. Bapak Prof. Dr. Gunarto S.H., M.Hum. selaku Rektor Universitas Islam Sultan Agung Semarang.
4. Ibu Dr. Novi Marlyana, S.T., M.T. sebagai Dekan Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang.
5. Ibu Jenny Putri Hapsari, S.T., M.T. selaku ketua jurusan Prodi Teknik Elektro Fakultas Teknologi Industri Univesitas Islam Sultan Agung Semarang.
6. Bapak Munaf Ismail S.T., M.T. selaku koordinator Tugas Akhir jurusan Teknik Elektro Fakultas Teknologi Industri Univesitas Islam Sultan Agung Semarang.
7. Bapak Dr. Bustanul Arifin, ST., MT. dan Bapak Dr. Ir. Agus Adhi Nugroho, MT. selaku dosen pembimbing I dan dosen pembimbing II yang telah

memberikan bimbingan dan dukungan dalam penyusunan Tugas Akhir ini.

8. Bapak Munaf Ismail S.T., M.T. selaku wali dosen jurusan Teknik Elektro Angkatan 2018 Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang.
9. Seluruh dosen dan karyawan Program Studi Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Sultan Agung Semarang atas ilmu, bimbingan, dan dukungan dalam penyusunan tugas akhir ini.
10. Kepada sahabat saya Riyan Fauzi Ashari, M. Singgih Raharjo, M. Shodig Burhannudin. dan Nur Mu'alim yang telah menjadi teman dalam membantu terselesainya tugas akhir ini.
11. Semua pihak yang tidak dapat penulis sebutkan satu per satu atas segala dukungan, semangat, ilmu dalam menyelesaikan tugas akhir ini.

Penulis juga menyadari bahwa dalam penyusunan Tugas Akhir masih banyak kekurangan, baik segi materi maupun penyajiannya. Penulis meminta maaf dan membutuhkan kritik dan saran yang membangun dari berbagai pihak, sehingga kedepannya laporan ini dapat menjadi lebih baik. Akhirnya penulis sangat berharap semoga laporan ini dapat memberi manfaat dan menambah wawasan bagi para pembaca dan khususnya bagi penulis juga.

Wassalamu'alaikum Warahmatullahi' Wabarakatuh

Semarang, 29 November 2022

Samsul Wahab

DAFTAR ISI

LAPORAN TUGAS AKHIR	i
LEMBAR PENGESAHAN PEMBIMBING	iii
LEMBAR PENGESAHAN PENGUJI	iv
SURAT PERNYATAAN KEASLIAN TUGAS AKHIR	v
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	vi
PERSEMBAHAN	vii
MOTTO	viii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvii
ABSTRAK	xviii
<i>ABSTRACT</i>	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI	5

2.1	Tinjauan Pustaka	5
2.2	Landasan Teori	6
2.2.1	Electric Power Steering (EPS)	6
2.2.2	Arduino UNO	10
2.2.3	Motor Listrik DC	13
2.2.4	Tipe Data	23
BAB III METODE PENELITIAN		27
3.1	Model Penelitian	27
3.2	Lokasi dan Objek Penelitian	31
3.3	Perancangan Program Kontrol PID	31
3.3.1	Inisialisasi Program	32
3.3.2	Program Rumus Kecepatan	34
3.3.3	Program Motor	34
3.3.4	Program Eksekusi	34
3.3.5	Monitoring Pada LabVIEW	36
3.4	Perancangan Program Simulasi Sistem	37
3.4.1	Perancangan Rangkaian Simulasi Sistem	37
3.4.2	Perancangan Program Arduino	38
3.4.3	Perancangan Monitoring pada LabVIEW	40
3.5	Perancangan Sistem Pada EPS	42
3.5.1	Motor EPS	43
3.5.2	Sensor Torsi EPS	43
3.5.3	Vehicle Speed Sensor (VSS)	46
3.5.4	Rangkaian Sistem Kemudi	46
3.5.5	Program Arduino	48
3.5.6	Monitoring Pada LabVIEW	51
BAB IV DATA DAN ANALISA		53

4.1 Pengujian Kontrol PID	53
4.2 Pengujian Program Simulasi	55
4.3 Pengujian Program Kendali Motor EPS	58
BAB V PENUTUP	74
5.1 Kesimpulan	74
5.2 Saran	74
DAFTAR PUSTAKA	75
LAMPIRAN	78

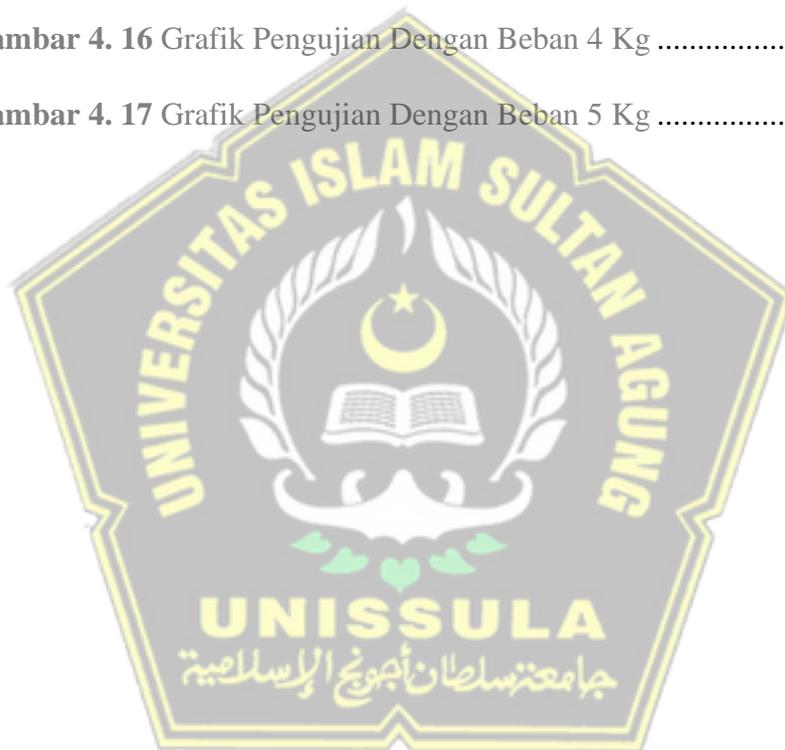


DAFTAR GAMBAR

Gambar 2. 1 Modul EPS Mobil Toyota [9].....	7
Gambar 2. 2 Gambar Kurva EPS	9
Gambar 2. 3 Pinout Arduino UNO R3 [13]	13
Gambar 2. 4 Konstruksi Motor DC bagian Stator dan Rotor	14
Gambar 2. 5 Bentuk Motor DC [15]	14
Gambar 2. 6 Prinsip Kerja Motor DC	17
Gambar 2. 7 Rangkaian Motor DC jenis Penguat Terpisah	18
Gambar 2. 8 Karakteristik Motor DC jenis Shunt.....	18
Gambar 2. 9 Rangkaian Motor DC jenis Shunt.....	19
Gambar 2. 10 Karakteristik Motor DC jenis Seri.....	19
Gambar 2. 11 Rangkaian Motor DC jenis Seri.....	19
Gambar 2. 12 Karakteristik Motor DC jenis Kompon	20
Gambar 2. 13 Rangkaian Motor DC jenis Kompon Panjang	20
Gambar 2. 14 Rangkaian Motor DC jenis Kompon Pendek	21
Gambar 2. 15 Kurva Karakteristik Torsi dan Arus Jangkar	21
Gambar 2. 16 Kurva Karakteristik Putaran Dan Arus Jangkar	22
Gambar 2. 17 Kurva Karakteristik Putaran dan Torsi.....	22
Gambar 3. 1 Blok Diagram Sistem Simulasi.....	27
Gambar 3. 2 Flowchart Sistem Simulasi	28
Gambar 3. 3 Blok Diagram Sistem Kemudi Utama	29

Gambar 3. 4 Flowchart Pembuatan Alat	30
Gambar 3. 5 Block Diagram Monitoring PID	36
Gambar 3. 6 Front Panel Monitoring PID	36
Gambar 3. 7 Rangkaian Simulasi	37
Gambar 3. 8 Tampilan Front Panel Rangkaian Simulasi	41
Gambar 3. 9 Block Diagram Rangkaian Simulasi.....	41
Gambar 3. 10 Motor EPS	43
Gambar 3. 11 Sensor Torsi EPS Avanza Xenia	44
Gambar 3. 12 Pinout Sensor Torsi	44
Gambar 3. 13 Rangkaian Kendali EPS.....	47
Gambar 3. 14 Block Diagram Sistem.....	51
Gambar 3. 15 Front Panel Sistem.....	52
Gambar 4. 1 Nilai Masukan dan Keluaran yang diperoleh	53
Gambar 4. 2 Grafik nilai masukan dan keluaran program PID.....	54
Gambar 4. 3 Potensiometer pada posisi nol	56
Gambar 4. 4 Tampilan LabVIEW saat Potensiometer pada posisi nol	56
Gambar 4. 5 Potensiometer diputar kekiri.....	57
Gambar 4. 6 Tampilan LabVIEW saat Potensiometer diputar kekiri	57
Gambar 4. 7 Potensiometer diputar kekanan.....	58
Gambar 4. 8 Tampilan LabVIEW saat Potensiometer diputar kekanan	58
Gambar 4. 9 Roda depan tanpa beban	59

Gambar 4. 10 Posisi Roda dengan beban 1 kg	59
Gambar 4. 11 Tanpa Beban Kecepatan 0 sampai 10 Putaran Kiri	61
Gambar 4. 12 Grafik Pengujian Tanpa Beban.....	62
Gambar 4. 13 Grafik Pengujian Dengan Beban 1 Kg	64
Gambar 4. 14 Grafik Pengujian Dengan Beban 2 Kg	66
Gambar 4. 15 Grafik Pengujian Dengan Beban 3 Kg	68
Gambar 4. 16 Grafik Pengujian Dengan Beban 4 Kg	70
Gambar 4. 17 Grafik Pengujian Dengan Beban 5 Kg	72



DAFTAR TABEL

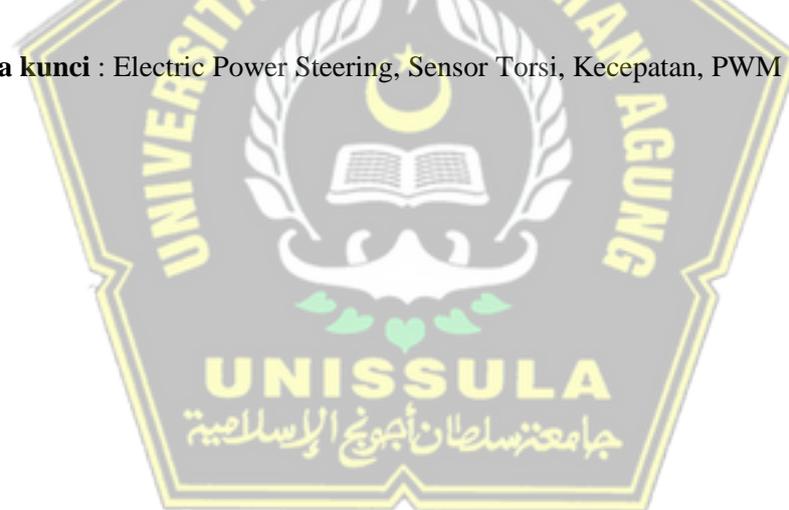
Tabel 2. 1 Spesifikasi Arduino UNO	11
Tabel 4. 1 Tabel nilai masukan dan keluaran program PID	55
Tabel 4. 2 Tabel Nilai PWM berdasarkan nilai kecepatan.....	60
Tabel 4. 3 Tabel Hasil Pengujian Tanpa Beban	62
Tabel 4. 4 Tabel Hasil Pengujian Dengan Beban 1 Kg.....	64
Tabel 4. 5 Tabel Hasil Pengujian Dengan Beban 2 Kg.....	66
Tabel 4. 6 Tabel Hasil Pengujian Dengan Beban 3 Kg.....	68
Tabel 4. 7 Tabel Hasil Pengujian Dengan Beban 4 Kg.....	70
Tabel 4. 8 Tabel Hasil Pengujian Dengan Beban 5 Kg.....	72



ABSTRAK

Pada penelitian ini dilakukan perancangan sistem kendali dengan menggunakan *Electric Power Steering* (EPS) dengan mengatur kecepatan motor pada kemudi tersebut. Program pembacaan nilai torsi, nilai kecepatan dan pengendalian kecepatan motor dibuat dengan Arduino IDE dan monitoring dilakukan lewat aplikasi LabVIEW. Percobaan dilakukan sebanyak 6 kali dimana pada percobaan pertama, roda depan tidak diberi beban dan pada percobaan kedua hingga keenam masing – masing diberi beban 1 kg hingga 5 kg. Kecepatan kendaraan dibagi menjadi 4 tingkatan, yaitu kecepatan sangat rendah, rendah, sedang, dan tinggi, dimana pada setiap nilai kecepatan kendaraan mempengaruhi nilai PWM motor EPS. Pada kecepatan tinggi, Skala PWM pada saat diputar kekanan, kekiri maupun *standby* sama – sama bernilai 0. Pada percobaan tanpa memberi beban PWM maksimal mencapai 102. Pada saat diberi beban 1 Kg, 2 Kg, 3 Kg, dan 5 Kg juga mencapai 102 skala PWM dan saat beban 4 Kg mencapai 127 skala PWM. Nilai tegangan pada sensor torsi yang terbaca saat posisi *standby* pada saat tanpa beban dan beban 1 Kg berada pada nilai 2,5 Volt dan pada beban 2 Kg, 3 Kg, 4 Kg, dan 5 Kg adalah 2,4 Volt. Pada saat diputar ke kiri nilai tegangan pada sensor torsi berada pada nilai rata – rata 15,125, dan pada saat diputar ke kanan bernilai 33,125.

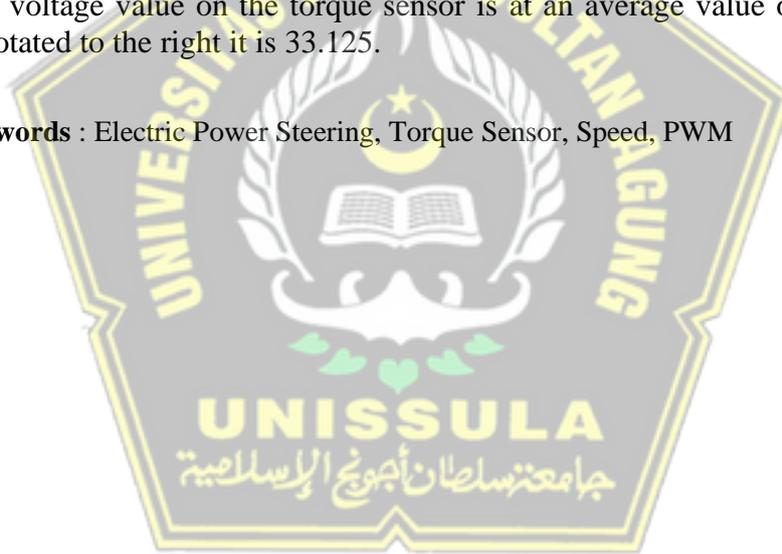
Kata kunci : Electric Power Steering, Sensor Torsi, Kecepatan, PWM



ABSTRACT

In this study, the design of the control system was carried out using *Electric Power Steering* (EPS) by adjusting the speed of the motor on the steering wheel. The program of reading torque values, speed values and controlling motor speed is made with Arduino IDE and monitoring is carried out through the LabVIEW application. The experiment was carried out 6 times where in the first experiment, the front wheels were not given a load and in the second to sixth experiments each was given a load of 1 kg to 5 kg. Vehicle speed is divided into 4 levels, namely very low, low, medium, and high speeds, where each vehicle speed value affects the PWM value of the EPS motor. At high speeds, the PWM Scale when rotated to the right, kekiri or *standby* are equally worth 0. In experiments without giving a maximum PWM load of up to 102. When given a load of 1 Kg, 2 Kg, 3 Kg, and 5 Kg also reached 102 PWM scales and when the load was 4 Kg reached 127 PWM scales. The voltage value on the torque sensor that is read when the *standby* position at no load and a load of 1 Kg is at a value of 2.5 Volts and at loads of 2 Kg, 3 Kg, 4 Kg, and 5 Kg is 2.4 Volts. When rotated to the left the voltage value on the torque sensor is at an average value of 15.125, and when rotated to the right it is 33.125.

Keywords : Electric Power Steering, Torque Sensor, Speed, PWM



BAB I PENDAHULUAN

1.1 Latar Belakang

Pesatnya perkembangan teknologi otomotif menuntut kinerja yang lebih baik dalam segala aspek seperti kenyamanan, keamanan dan stabilitas. Sistem kemudi adalah salah satu aspek terpenting dari desain kendaraan, karena sistem ini memengaruhi respons gerakan kemudi kendaraan terhadap input kemudi pengemudi, yang secara signifikan memengaruhi kenyamanan dan keselamatan. Untuk itu diperlukan rancangan sistem kendali yang nyaman dan aman yang mendukung kesempurnaan kendaraan itu sendiri [1].

Sistem kemudi adalah mekanisme pada kendaraan yang mengontrol arah kendaraan dengan memutar roda depan. Perkembangannya juga meliputi penyetelan arah roda belakang untuk memudahkan belok atau bahkan sebagai kontrol utama saat membelok. Power steering adalah sistem hidraulik (servo-hidraulik) yang bertugas untuk mengurangi gaya yang diperlukan untuk memutar roda, terutama pada kecepatan rendah, dan menyesuaikan pada kecepatan sedang dan tinggi. Pada kecepatan rendah, gesekan antara ban dengan jalan cukup tinggi, apalagi pada ban bertekanan rendah dengan tapak lebar [2].

Untuk memudahkan kemudi, kendaraan biasanya menggunakan power steering yaitu power steering hidraulik. Namun, power steering hidraulik memiliki kelemahan seperti menggunakan media cair, komponen mekanis, deteksi kerusakan yang kurang baik dan kurang ramah lingkungan, serta konsumsi energi yang konstan. Oleh karena itu, power steering hidraulik untuk kendaraan khususnya kendaraan niaga sangat berat dalam hal konsumsi energi. Salah satu cara untuk mengatasi kelemahan power steering hidraulik adalah dengan menggunakan power steering elektrik [3].

Electric power steering (EPS) atau power steering elektrik adalah sistem *power steering* yang banyak digunakan pada mobil modern, termasuk produk terbaru. Berbeda dengan power steering hidraulik, EPS mengandalkan motor

listrik untuk mengurangi ketegangan pada tangan pengemudi saat memutar setir. EPS menggunakan sensor yang membaca input dari pengemudi ke setir kemudian mengirimkan sinyal ke motor listrik untuk memutar setir di dalam rumah setir [4].

Pada penelitian ini dilakukan perancangan sistem kendali dengan menggunakan *Electric Power Steering* (EPS) dengan mengatur kecepatan motor pada kemudi tersebut. Pengaturan kecepatan motor pada EPS dilakukan agar Pengemudi dapat memutar kemudi berdasarkan pada keadaan masukan. Masukan untuk motor EPS berasal dari nilai kecepatan mobil dan juga besaran torsi yang terbaca oleh sensor torsi pada *Electric Power Steering* (EPS). Pengaturan kecepatan motor pada EPS ini akan mempengaruhi seberapa berat atau ringannya kemudi mobil ketika akan berbelok.

Electric Power Steering (EPS) yang digunakan pada penelitian ini diambil dari mobil Toyota Avanza Xenia 2012 dan dikontrol menggunakan mikrokontroler Arduino UNO R3. Penggunaan mikrokontroler Arduino UNO R3 didasari oleh potensi dari mikrokontroler jenis ini. Mikrokontroler Arduino terkenal dengan kompatibilitasnya yang tinggi dengan berbagai perangkat sensor maupun aktuator. Untuk memonitor masukan dan keluaran dari perangkat yang terpasang pada Arduino, digunakan aplikasi LabVIEW.

1.2 Perumusan Masalah

Masalah yang dibahas dalam tugas akhir ini adalah desain kendali kecepatan motor dengan masukan kecepatan mobil dan torsi pada *Electric Power Steering* (EPS) menggunakan mikrokontroler ATmega328 yang ada pada Arduino UNO.

1.3 Batasan Masalah

Dalam penyusunan penelitian ini, telah ditentukan batasan-batasan masalah sebagai berikut:

1. *Plant* yang akan digunakan adalah berupa kolom setir dan motor EPS Avanza Xenia 2012.
2. *Vehicle Speed Sensor* untuk mengukur kecepatan kendaraan sebagai salah satu sumber masukan.
3. *Torque sensor* EPS Avanza Xenia 2012 sebagai pengukur besaran torsi kendaraan sebagai salah satu sumber masukan.
4. Mikrokontroler Arduino UNO dengan ATmega328 sebagai pusat kendali sistem.
5. *Driver* motor BTS7960 sebagai penghubung kendali dari mikrokontroler Arduino ke motor EPS.
6. *Software* Arduino IDE sebagai perangkat lunak pemrograman utama.
7. *Software* LabVIEW sebagai perangkat lunak *monitoring* sistem utama.
8. Pengujian kontrol kecepatan motor pada EPS berdasarkan nilai torsi kemudi dilakukan dengan memberi beban sampai 5 kg pada roda depan dengan posisi seluruh roda kendaraan tidak menyentuh lantai.

1.4 Tujuan

Tujuan penelitian ini adalah merancang pengendalian motor pada EPS berdasarkan nilai kecepatan kendaraan mobil dan nilai torsi putaran motor EPS.

1.5 Manfaat

Adapun manfaat yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Meningkatkan keamanan dan kenyamanan dalam berkendara.
2. Kendali Otomatis kendaraan dengan sistem EPS yang sudah diprogram.

1.6 Sistematika Penulisan

Dalam penyusunan Tugas Akhir ini, sistematika penulisan yang digunakan yaitu sebagai berikut :

BAB I : PENDAHULUAN

Pada bab ini dibahas tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat tugas akhir dan sistematika penyusunan laporan akhir.

BAB II : TINJAUAN PUSTAKA DAN LANDASAN TEORI

Pada bab ini berisi tentang teori – teori dasar yang berhubungan dengan *Electric Power Steering (EPS)*, Motor DC dan Mikrokontroler Arduino UNO ATmega 328 secara umum dan rinci.

BAB III : METODE PENELITIAN

Bab ini memberikan gambaran tentang tujuan penelitian, data penelitian, metode/langkah penelitian, dan metode penelitian yang digunakan untuk membuat sistem pengatur kecepatan motor electric power steering (EPS) berbasis mikrokontroler Atmega328.

BAB IV : DATA DAN ANALISA

Bab ini berisi tentang semua hasil penelitian yang dilakukan.

BAB V : PENUTUP

Bab ini membahas kesimpulan hasil penelitian yang telah dilakukan dan saran – saran yang di berikan peneliti berdasarkan kesimpulan.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Pada tugas akhir sebelumnya oleh Liga Kusuma Pahlawan dan Avin Maulida Dwi H. [3] Sebagai pengembangan lebih lanjut dari sistem kemudi yang ada untuk mobil kancil, sistem power steering listrik untuk mobil kancil dirancang dengan mempertimbangkan perubahan sudut kemudi dan kecepatan yang berbeda. Sebuah model skala 1:90 berfungsi sebagai alat uji sebagai alternatif dari power steering listrik dan mobil kancil.

Pada penelitian oleh Khoirudin Fathoni dan Alfa Faridh Suni [5] yang menggunakan metode *root locus* untuk mengatur kendali kecepatan pada motor dengan Arus searah. Perancangan PID menggunakan posisi *root model* sistem memiliki parameter respon (waktu solusi, waktu naik, tumpang tindih maksimum, rasio dumping, dll) dan analisis stabilitas yang lebih jelas daripada menentukan parameter PID tanpa model sistem.

Pada penelitian oleh Untung Rahardja, Asep Saefullah, dan M. Ramdani [6] yang secara otomatis mensimulasikan sistem *cruise control* mobil dengan memasang pengontrol yang dapat mendeteksi jarak antara mobil dan benda-benda di sekitarnya. Simulasi sistem *automatic cruise control* menjadi salah satu alternatif untuk mengatasi kecelakaan lalu lintas akibat tabrakan mobil.

Pada penelitian oleh Anthoinete P.Y.Waroh [7] mengenai Analisa dan simulasi sistem pengendalian motor DC. Berisi tentang hasil Analisa dan simulasi berdasarkan perhitungan – perhitungan yang ada. Epyk Sunarno, Soebagio, dan Mauridhi Heri Purnomo tentang Pengaturan Kecepatan Motor Induksi Tanpa Sensor Kecepatan dengan *Metoda Direct Torque Control* Menggunakan *Observer Recurrent Neural Network* [8].

Pada penelitian ini menggunakan EPS Toyota Avanza Xenia 2012 yang dipasang pada mobil golf yang telah dimodifikasi. Perangkat yang digunakan untuk mengontrol EPS adalah mikrokontroler Arduino UNO R3. Salah satu kelebihan dari mikrokontroler Arduino adalah dapat dipasangkan dengan banyak jenis sensor maupun aktuator. Pada penelitian ini, Arduino mendapat masukan dari sensor torsi yang terpasang pada batang setir EPS, potensiometer, dan *rotary encoder* yang terpasang pada motor EPS. Masukan tersebut kemudian diolah dengan program pada Arduino IDE untuk mengatur keluaran motor EPS.

2.2 Landasan Teori

2.2.1 Electric Power Steering (EPS)

1. Pengertian EPS

Electronic power steering (EPS) adalah sistem modifikasi alur kerja *power steering* yang menghubungkan sistem hidrolis dengan sistem kelistrikan. Sistem *electronic power steering* (EPS) bekerja sedemikian rupa sehingga saat kunci diputar ke posisi ON, modul kontrol menerima tenaga listrik ke mode siaga, yang juga menerangi lampu indikator EPS di panel instrumen. Saat mesin dihidupkan, peredam segera memberi sinyal ke unit kontrol untuk mengaktifkan motor listrik, dan kopling menghubungkan motor langsung ke lengan kemudi. Salah satu sensor pada *steering rack* bertugas memberi tahu *steering module* kapan setir akan mulai berputar. Disebut sensor torsi, ini mengirimkan informasi tentang berapa lama setir diputar dan seberapa cepat putarannya. Dengan dua informasi ini, unit kontrol segera mengirimkan arus yang diperlukan ke motor listrik untuk memutar setir. Ini membuatnya lebih mudah untuk memutar setir. Sensor kecepatan kendaraan dihidupkan segera setelah mobil mulai bergerak. Sensor ini menyediakan unit kontrol dengan informasi tentang kecepatan kendaraan. Pada kecepatan tinggi, biasanya 80 km/jam, unit kontrol mematikan motor listrik. Gambar Modul EPS Toyota dapat dilihat pada Gambar 2.1 berikut.



Gambar 2. 1 Modul EPS Mobil Toyota [9]

Ini membuat setir lebih berat dan meningkatkan keselamatan. Oleh karena itu sistem EPS ini hanya mengatur besarnya arus listrik yang dialirkan ke motor listrik sesuai kebutuhan. Selain mengatur pengoperasian motor listrik berdasarkan data dari sensor, modul kontrol juga mendeteksi jika ada kerusakan pada sistem EPS. Lampu indikator EPS di dasbor akan menyala dengan kedipan tertentu saat terjadi kegagalan fungsi. Selain itu, unit kontrol menonaktifkan motor listrik dan sakelar memutus motor dari kolom kemudi. Namun, karena sistem kemudi yang dilengkapi EPS masih tersambung ke roda kemudi dengan batang baja, maka mobil tetap dapat dikemudikan. Meski memutar setir terasa berat, seperti setir tanpa power steering [10].

2. Komponen EPS

Electric Power Steering (EPS) umumnya menggunakan beberapa perangkat elektronik seperti:

- a. Modul control (*Control Module*) seperti komputer untuk mengelola pekerjaan EPS.
- b. Sebuah motor listrik yang fungsinya untuk memperlancar secara langsung putaran roda kemudi.
- c. Sensor kecepatan kendaraan, terletak di *gearbox* dan berperan untuk menginformasikan unit kontrol tentang kecepatan kendaraan.
- d. Torque sensor pada *steering column* yang fungsinya memberikan informasi ke *control module* saat pengemudi mulai memutar setir.

- e. Clutch/Sakelar, ini terletak di antara mesin dan roda kemudi. Tugasnya menyambung dan melepas motor ke kolom setir sesuai kondisi.
- f. *Noise Suppressor*, Bertindak sebagai sensor yang mendeteksi apakah mesin hidup atau tidak.
- g. *On-board Diagnostic Display*, berupa lampu indikator di dashboard yang menyala saat terjadi masalah pada sistem EPS.

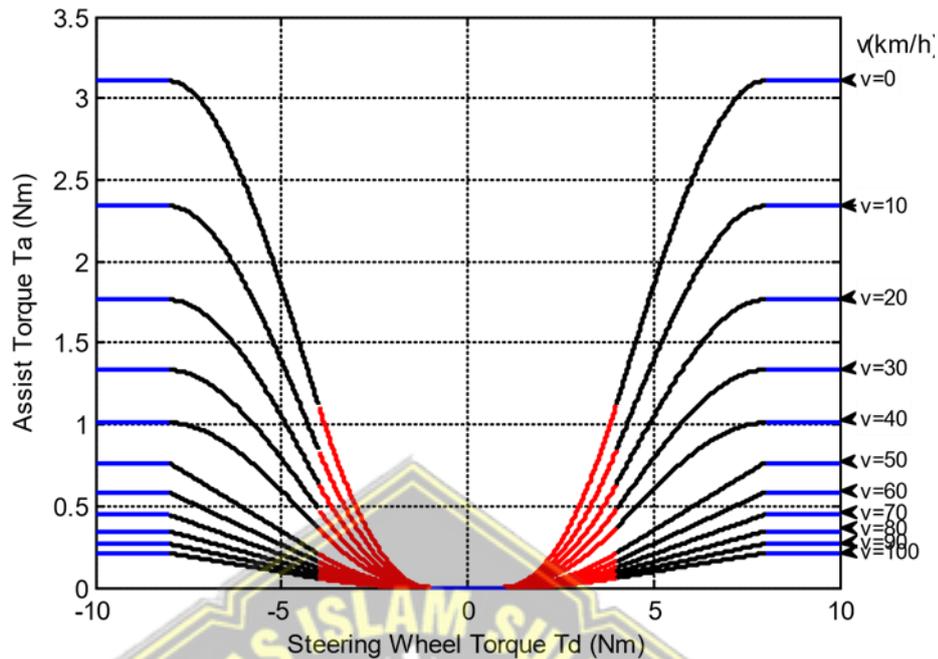
3. Cara Kerja EPS

Cara kerja *Electric Power Steering* adalah sebagai berikut.

- a. Saat pengemudi memutar roda kemudi, sensor torsi kemudi menerima sinyal posisi dan kecepatan putaran roda kemudi.
- b. Sinyal *steering torque* sensor dikirim ke EPS ECU (Electronic Control Unit) bersama dengan sinyal lain dari ECM (Electronic Control Module) seperti kecepatan kendaraan dan RPM mesin.
- c. Unit kontrol EPS mengontrol motor listrik untuk memberikan tenaga tambahan yang dibutuhkan ke rak kemudi.
- d. Tenaga belok ekstra diarahkan ke roda, membuat kemudi lebih responsif.

4. Kurva Electric Power Steering

Kurva karakteristik *power assist* merupakan kurva fungsi torsi kemudi pada kecepatan dan torsi roda kemudi. Ada tiga jenis kurva karakteristik power assist, seperti garis lurus, garis lipat dan tipe kurva. Menganalisis tiga kurva fleksibilitas kemudi dan perasaan jalan, dapat dilihat bahwa karakteristik pendukung tipe kurva lebih baik daripada tipe kurva, dan tipe kurva lebih baik daripada garis lurus dari perasaan jalan, sedangkan model garis lurus adalah lebih unggul dari tipe garis lengkung, tipe garis lebih baik daripada tipe stang manuver tipe lengkung magnetik [11]. Gambar Kurva Electric Power Steering dapat dilihat pada Gambar 2.2.



Gambar 2. 2 Gambar Kurva EPS

5. Sensor Torsi Pada EPS

Dua jenis sensor torsi utama yang ditemukan dalam sistem power steering elektronik adalah teknologi kontak dan non-kontak. Kedua jenis melakukan fungsi yang sama untuk mengambil informasi kemudi input dan mengirimkan data tersebut sebagai output kembali ke sistem EPS, tetapi ada beberapa perbedaan fungsional antara kedua kategori tersebut.

a. Sensor Torsi Non-Kontak

Mungkin yang lebih disukai dan lebih maju dari 2 jenis sensor torsi di EPS, sensor torsi non-kontak (termasuk sensor tipe Hall) mengukur perubahan magnetisasi untuk membaca perubahan arah kemudi dan masukan serta mengukur torsi. Di dalam sensor non-kontak, sebuah rotor magnetik dengan potongan kutub memvariasikan voltase keluaran tergantung pada posisi roda. Spline input dan slot output (dengan batang torsi di antaranya) mengubah posisi relatif terhadap jumlah torsi, dan informasi tersebut diteruskan ke modul kontrol melalui tegangan magnet. Meskipun potongan-potongan canggih ini membuat sensor non-kontak lebih mahal, mereka mengandung lebih sedikit bagian yang bergerak dan tahan aus.

b. Sensor Torsi Kontak

Inti dari sensor torsi kontak adalah batang torsi puntir yang terhubung ke jembatan putar yang mengukur jumlah puntiran, atau torsi, yang dialami batang torsi. Sikat kontak di jembatan terhubung ke rumahan sensor dan memfasilitasi transfer informasi voltase ke modul kontrol. Dibandingkan dengan sensor non-kontak, sensor torsi kontak menggabungkan lebih banyak bagian bergerak yang lebih rentan terhadap keausan, pelapukan, dan elemen seperti cairan, kotoran, dan kontaminan. Namun, sensor kontak telah menjadi jauh lebih canggih selama bertahun-tahun, dan umumnya memberikan pilihan biaya yang lebih rendah dibandingkan sensor non-kontak yang lebih mahal [12].

2.2.2 Arduino UNO

Arduino UNO adalah papan mikrokontroler berbasis ATmega328P. Mikrokontroler ini memiliki 14 pin input/output digital (6 di antaranya dapat digunakan sebagai output PWM), 6 input analog, resonator keramik 16 MHz, koneksi USB, colokan listrik, header ICSP, dan tombol reset. Mikrokontroler ini berisi semua yang diperlukan untuk mendukung mikrokontroler; cukup sambungkan ke komputer dengan kabel USB atau nyalakan dengan adaptor AC-ke-DC atau baterai untuk dapat digunakan. Anda dapat bermain-main dengan UNO Anda tanpa terlalu khawatir melakukan sesuatu yang salah, skenario terburuk Anda dapat mengganti chip dengan harga yang murah dan memulai dari awal lagi. Chip ATmega328P dapat dengan mudah diganti, karena tidak disolder ke papan. ATmega328P juga dilengkapi EEPROM 1kb, memori yang tidak terhapus saat dimatikan [13]. Spesifikasi lengkap Arduino UNO dapat dilihat pada Tabel 2.1, dan *pinout* Arduino UNO dapat dilihat pada Gambar 2.3.

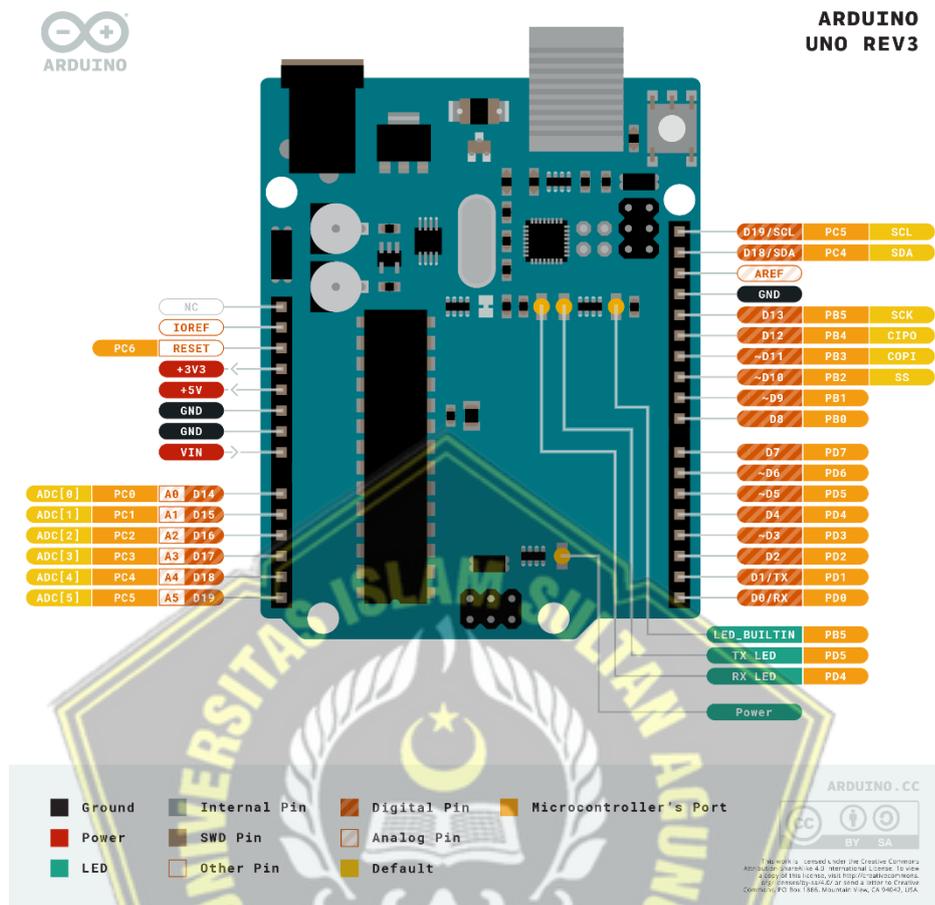
Tabel 2. 1 Spesifikasi Arduino UNO

Microcontroller	ATmega328P	
USB connector	USB-B	
Pins	Built-in LED Pin	13
	Digital I/O Pins	14
	Analog input pins	6
	PWM pins	6
Communication	UART	Yes
	I2C	Yes
	SPI	Yes
Power	I/O Voltage	5V
	Input voltage (nominal)	7-12V
	DC Current per I/O Pin	20 mA
	Power Supply Connector	Barrel Plug
Clock speed	Main Processor	ATmega328P 16 MHz
	USB-Serial Processor	ATmega16U2 16 MHz
Memory	ATmega328P	2KB SRAM, 32KB FLASH, 1KB EEPROM
Dimensions	Weight	25 g
	Width	53.4 mm

Length

68.6 mm



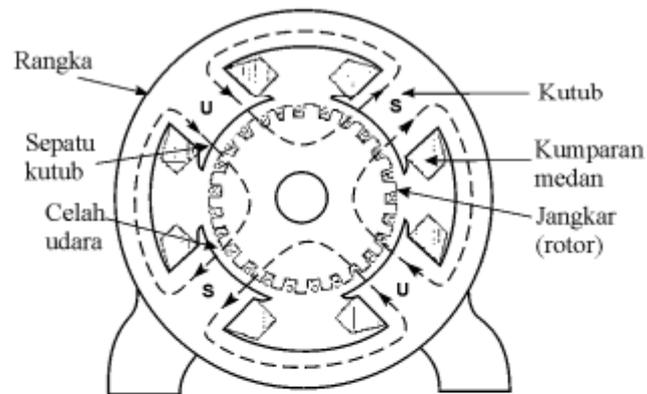


Gambar 2. 3 Pinout Arduino UNO R3 [13]

2.2.3 Motor Listrik DC

1. Pengertian Motor Listrik DC

Motor listrik adalah alat yang mengubah energi listrik arus searah menjadi energi kinetik atau gerak dalam bentuk putaran. Secara umum motor listrik terdiri dari dua bagian yaitu bagian diam (stator) dan bagian berputar (rotor). Bagian stator merupakan tempat diletakkannya kumparan medan yang dapat berfungsi untuk menghasilkan fluks magnet sedangkan pada bagian rotor ditempati oleh rangkaian seperti kumparan jangkar yang berfungsi menjadi tempat terbentuknya gaya gerak listrik (ggl E), sikat dan komutator. Apabila arus pada kumparan jangkar terhubung medan magnet, maka akan menghasilkan torsi (T) yang dapat memutar motor [14].



Gambar 2. 4 Konstruksi Motor DC bagian Stator dan Rotor

Motor DC adalah motor yang beroperasi dengan tegangan masukan berupa tegangan DC. Semakin tinggi tegangan maka semakin cepat motor DC akan berputar menggunakan satuan kecepatan RPM (*Rotation Per Minute*). Namun tegangan yang digunakan tidak boleh melebihi tegangan maksimum yang tertera pada karakter motor karena hal ini dapat merusak motor atau membakar belitan karena tidak dapat menahan besarnya arus yang dihasilkan. Contoh bentuk Motor DC dapat dilihat pada Gambar 2.3 berikut.



Gambar 2. 5 Bentuk Motor DC [15]

2. Komponen Motor DC

Berikut adalah tiga komponen penting yang umumnya terdapat di dalam motor listrik DC:

a. Kutub Medan Magnet

Secara sederhana dapat dipahami bahwa interaksi dua kutub magnet akan menyebabkan perputaran pada motor DC. Motor DC memiliki kutub medan yang stasioner dan kumparan motor DC yang menggerakkan bearing pada ruang diantara kutub medan. Motor DC sederhana memiliki dua kutub medan: kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi bukaan diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet. Elektromagnet menerima listrik dari sumber daya dari luar sebagai penyedia struktur medan.

b. Kumparan Motor Listrik DC

Bila arus masuk menuju kumparan motor DC, maka arus ini akan menjadi elektromagnet. kumparan motor DC yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, kumparan motor DC berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi. Jika hal ini terjadi, arusnya berbalik untuk merubah kutub-kutub utara dan selatan kumparan motor DC.

c. Komutator Motor Listrik DC

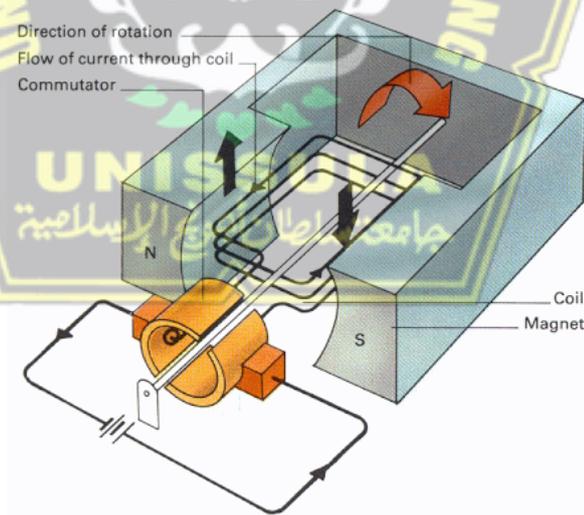
Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk membalikan arah arus listrik dalam kumparan motor DC. Komutator juga membantu dalam transmisi arus antara kumparan motor DC dan sumber daya [16].



3. Prinsip Kerja Motor DC

Gerak atau putaran yang dihasilkan oleh motor arus searah diperoleh dari interaksi dua buah medan yang dihasilkan oleh bagian jangkar (*armature*) dan bagian medan (*field*) dari motor arus searah. Pada gambar dibawah, bagian medan berbentuk suatu kumparan yang terhubung ke sumber arus searah. Sedangkan bagian jangkar ditunjukkan sebagai magnet permanen (U-S), bagian jangkar ini tidak harus berbentuk magnet permanen, bisa juga berbentuk belitan yang akan menjadi elektromagnet apabila mendapatkan sumber arus searah.

Sehingga apabila motor arus searah berjenis jangkar belitan, maka kita harus menyediakan dua sumber arus searah, satu untuk bagian jangkar, satu lagi untuk bagian medan. Bagian lain yang tidak kalah penting pada motor arus searah adalah adanya komutator yang berpasangan dengan cincin belah (*slip rings*). Pasangan ini menjadikannya suatu konverter mekanik yang membuat arus dari sumber mengalir pada arah yang tetap walaupun belitan medan berputar [14].



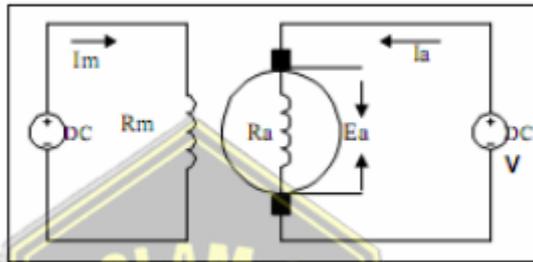
Gambar 2. 6 Prinsip Kerja Motor DC

4. Jenis – Jenis Motor DC

Ada berbagai macam jenis motor DC yang dapat ditemui, berikut adalah beberapa jenis motor DC pada umumnya.

a. Motor DC penguat terpisah / *Separately Excited*

Motor DC yang menggunakan arus penguat magnet (I_f) yang diperoleh dari sumber arus searah di luar motor.



Gambar 2. 7 Rangkaian Motor DC jenis Penguat Terpisah

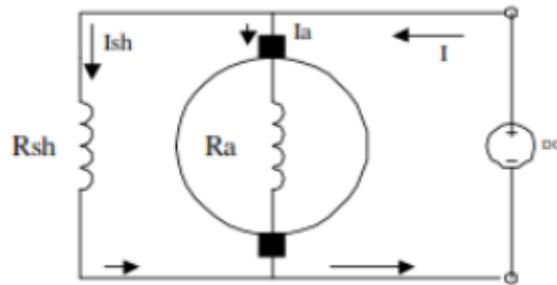
b. Motor DC penguat sendiri / *Self Excited: Motor Shunt*

Motor DC yang menggunakan arus penguat magnet berasal dari motor DC itu sendiri. Pada motor shunt, lilitan penguat magnet disambungkan secara paralel dengan lilitan jangkar (A) seperti diperlihatkan dalam Gambar 2.7. Oleh karena itu total arus dalam jalur merupakan penjumlahan arus medan dan arus jangkar.



Gambar 2. 8 Karakteristik Motor DC jenis Shunt

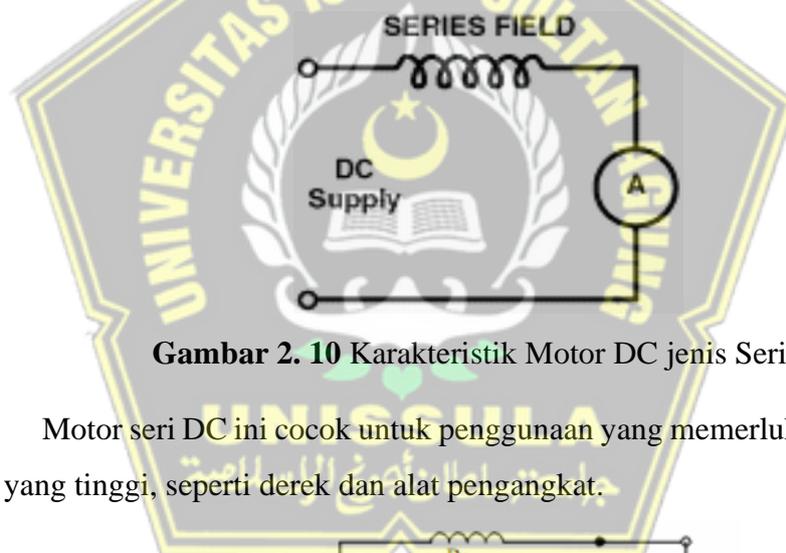
Kecepatan pada prakteknya konstan tidak tergantung pada beban dan oleh karena itu cocok untuk penggunaan komersial dengan beban awal yang rendah, seperti peralatan mesin. Kecepatannya dapat dikendalikan dengan cara memasang tahanan dalam susunan seri dengan dinamo (untuk kecepatan berkurang) atau dengan memasang tahanan pada arus medan (untuk kecepatan bertambah).



Gambar 2. 9 Rangkaian Motor DC jenis Shunt

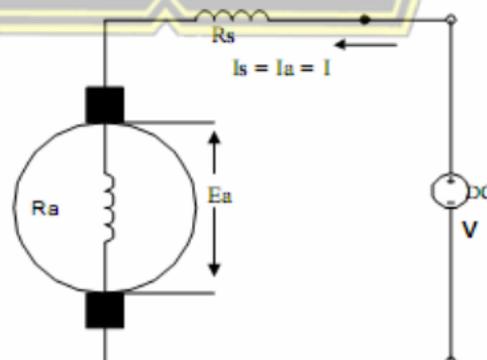
c. Motor DC penguat sendiri / Self Excited: Motor Seri

Motor DC penguat sendiri yang menggunakan lilitan penguat magnet dihubungkan seri dengan lilitan jangkar. Oleh karena itu nilai arus medan sama dengan arus jangkar.



Gambar 2. 10 Karakteristik Motor DC jenis Seri

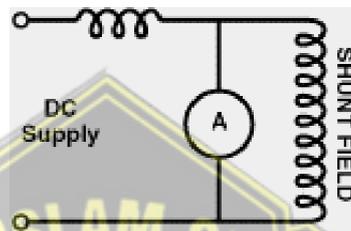
Motor seri DC ini cocok untuk penggunaan yang memerlukan Torsi awal yang tinggi, seperti derek dan alat pengangkat.



Gambar 2. 11 Rangkaian Motor DC jenis Seri

d. Motor DC Kompon/Gabungan

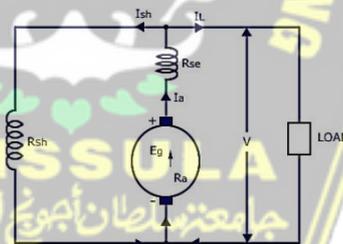
Pada motor kompon, lilitan penguat magnet dihubungkan secara paralel dan seri dengan lilitan jangkar. Sehingga, motor kompon memiliki torque penyalaan awal yang bagus dan kecepatan yang stabil. Makin tinggi persentase penggabungan (yakni persentase gulungan medan yang dihubungkan secara seri), makin tinggi pula torsi penyalaan awal yang dapat ditangani oleh motor ini.



Gambar 2. 12 Karakteristik Motor DC jenis Kompon

1. Motor DC Kompon Panjang

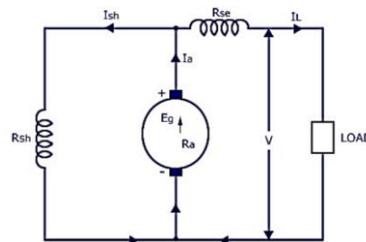
Kumparan penguat medan tambahan diletakkan secara seri antara kumparan jangkar dan kumparan penguat medan pada motor DC shunt.



Gambar 2. 13 Rangkaian Motor DC jenis Kompon Panjang

2. Motor DC Kompon Pendek

Dimana tata letak kumparan penguat medan tambahan diletakkan seri dengan kumparan medan pada motor DC shunt.



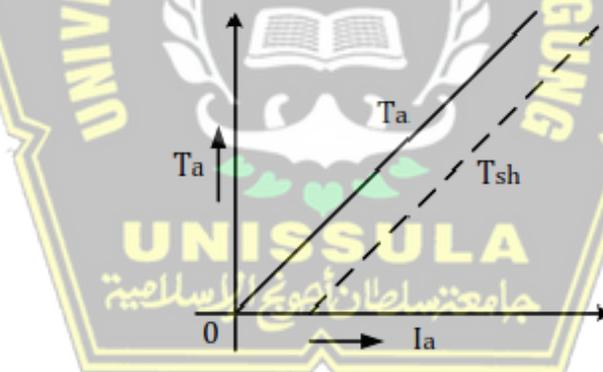
Gambar 2. 14 Rangkaian Motor DC jenis Kompon Pendek

5. Karakteristik Motor DC

Kinerja dari motor DC dapat ditentukan berdasarkan kurva karakteristiknya yang disebut karakteristik. Berikut adalah tiga karakteristik penting pada motor DC:

a. Karakteristik Torsi dan Arus Jangkar (T_a/I_a)

Asumsikan ϕ bernilai konstan maka dapat diketahui bahwa nilai T_a sebanding nilai I_a maka besar torsi akan bertambah secara linear dengan bertambahnya I_a . Untuk lebih jelasnya, bentuk kurva karakteristik torsi dan arus jangkar dapat dilihat pada Gambar 2.15 berikut.

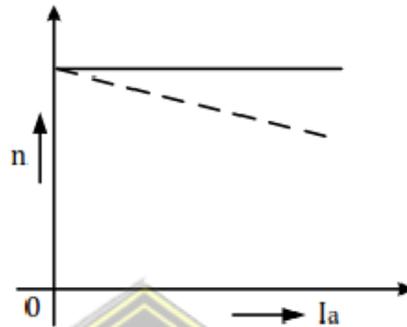


Gambar 2. 15 Kurva Karakteristik Torsi dan Arus Jangkar

b. Karakteristik Putaran dan Arus Jangkar (n/I_a)

Jika ϕ dianggap bernilai konstan, maka n sebanding dengan E_a . Fluksi ϕ dan ggl lawan E_a di dalam motor DC shunt akan berkurang dengan penambahan beban. Bagaimanapun juga E_a akan berkurang lebih banyak daripada ϕ secara keseluruhan, ada beberapa pengurangan kecepatan. Drop bervariasi dari 5 sampai 15% kecepatan beban penuh, tergantung saturasi, reaksi jangkar dan posisi sikat. Oleh karena itu, kurva kecepatan akan

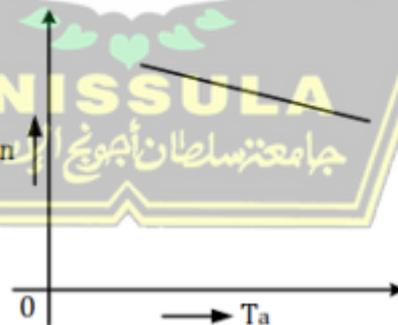
berkurang ditunjukkan oleh garis putus putus. Tetapi secara umum motor shunt memiliki kecepatan putaran yang konstan. Untuk lebih jelasnya bentuk kurva putaran – arus jangkar dapat dilihat pada Gambar 2.16 berikut.



Gambar 2. 16 Kurva Karakteristik Putaran Dan Arus Jangkar

c. Karakteristik Putaran dan Torsi (n/T_a)

Karakteristik putaran dan torsi diperoleh dengan menggambarkan nilai n dan T_a untuk berbagai arus jangkar (I_a). Dari kurva dapat dilihat bahwa kecepatan sedikit menurun seiring dengan penambahan beban. Untuk lebih jelasnya karakteristik putaran dan torsi dapat dilihat pada Gambar 2.17 berikut [14].



Gambar 2. 17 Kurva Karakteristik Putaran dan Torsi

6. Daya Keluaran Motor DC

Daya yang dihasilkan pada motor dapat dihitung melalui rumus berikut

$$P = E_a \times I_a = T \times \omega \dots\dots\dots (2.1)$$

Dimana:

$$E_a = \text{Tegangan (V)}$$

- I_a = Arus Jangkar (A)
 P = Daya Motor (W)
 T = Torsi Motor (nM)
 ω = *Angular Speed* (rad/s)

7. Torsi

Secara umum khususnya dalam bidang fisika, momen gaya alias torsi adalah penyebab suatu benda berputar. Torsi dapat memperlihatkan seberapa besar gaya yang membuat sebuah benda berotasi. Nilai dari torsi bergantung pada gaya yang dikeluarkan dan jarak antara sumbu putaran dengan letak gaya.

Dalam dunia otomotif, torsi adalah besaran untuk menyatakan seberapa besar gaya dorong yang bisa diproduksi oleh suatu mesin. Namun torsi ini berbeda dengan tenaga. Torsi adalah sistem mobil yang lebih berfokus pada urusan akselerasi atau percepatan. Sementara tenaga berhubungan dengan sisi kecepatan.

Rumus torsi sendiri merupakan hasil perkalian antara gaya dengan lengan momennya. Secara matematis rumus torsi ditulis sebagai berikut.

$$\tau = I \times F \dots\dots\dots (2.2)$$

Jika lengan gaya (F) dan lengan gaya (l) tidak tegak lurus, maka rumusnya menjadi sebagai berikut.

$$\tau = I \times F \sin \alpha \dots\dots\dots (2.3)$$

Dimana:

τ = Momen Gaya (Nm)

I = Lengan Gaya (m)

F = Gaya (N)

α = Sudut antara Lengan Gaya (I) dan Gaya (F) [17]

2.2.4 Tipe Data

Berikut adalah beberapa jenis tipe data yang umumnya dipakai dalam pemrograman terutama pada Arduino.

1. Integer (int)

Integer atau Bilangan bulat adalah tipe data utama Anda untuk

penyimpanan angka. Di perangkat Arduino Uno atau *board* arduino berbasis ATmega lainnya, int menyimpan nilai 16-bit (2-byte). Ini menghasilkan kisaran -32.768 hingga 32.767 (nilai minimum -2^{15} dan nilai maksimum $(2^{15}) - 1$). Pada *board* berbasis Arduino Due dan SAMD seperti MKR1000 dan Zero, sebuah int menyimpan nilai 32-bit (4-byte). Ini menghasilkan kisaran -2.147.483.648 hingga 2.147.483.647 (nilai minimum -2^{31} dan nilai maksimum $(2^{31}) - 1$).

int menyimpan bilangan negatif dengan teknik yang disebut (2's *complement math*). Bit tertinggi, terkadang disebut sebagai bit "tanda", menandai angka sebagai angka negatif. Sisa bit dibalik dan 1 ditambahkan.

2. Float

Tipe data untuk *floating-point numbers*, angka yang memiliki titik desimal. *Floating-point number* sering digunakan untuk mengaproksimasi nilai analog dan kontinu karena memiliki resolusi yang lebih besar daripada bilangan (*integer*). Angka floating-point bisa sebesar 3,4028235E+38 dan serendah -3,4028235E+38. Mereka disimpan sebagai 32 bit (4 byte) informasi.

3. Long

Variabel *long* adalah variabel ukuran yang diperluas untuk penyimpanan angka, dan menyimpan 32 bit (4 byte), dari -2.147.483.648 hingga 2.147.483.647.

4. Short

Short adalah jenis tipe data 16-bit. Pada semua Arduino (berbasis ATmega dan ARM), *short* menyimpan nilai 16-bit (2-byte). Ini menghasilkan kisaran -32.768 hingga 32.767 (nilai minimum -2^{15} dan nilai maksimum $(2^{15}) - 1$).

5. Double

Double precision floating point number pada Arduino Uno dan *board* berbasis ATMEGA lainnya, ini menempati 4 byte. Artinya, implementasi

double persis sama dengan *float*, tanpa presisi. Pada *board* Arduino Due, *double* memiliki presisi 8-byte (64 bit).



6. Char

Tipe data yang digunakan untuk menyimpan nilai karakter. Literal karakter ditulis dalam tanda kutip tunggal, seperti 'A' dan untuk banyak karakter *string* gunakan tanda kutip ganda seperti "ABC".

Ukuran tipe data *char* minimal 8 bit. Disarankan untuk hanya menggunakan *char* untuk menyimpan karakter. Untuk tipe data *unsigned*, satu byte (8 bit) lebih baik menggunakan tipe data *byte* [18].

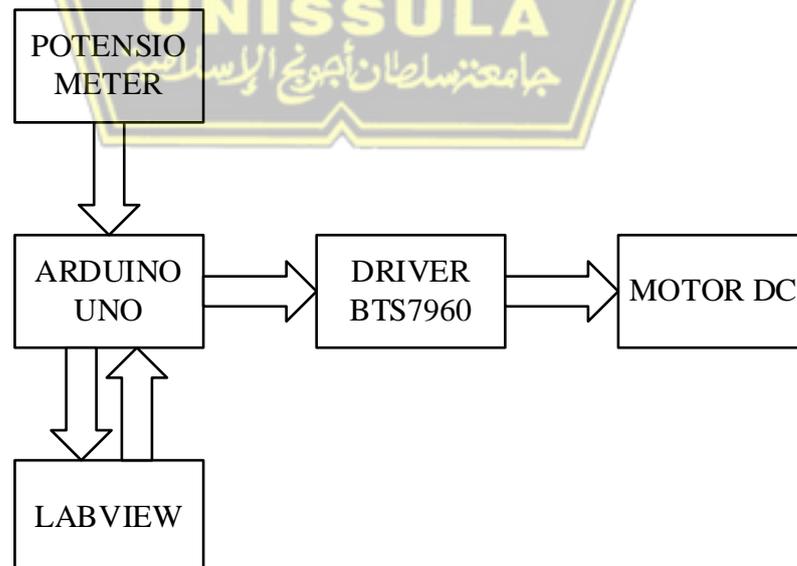


BAB III METODE PENELITIAN

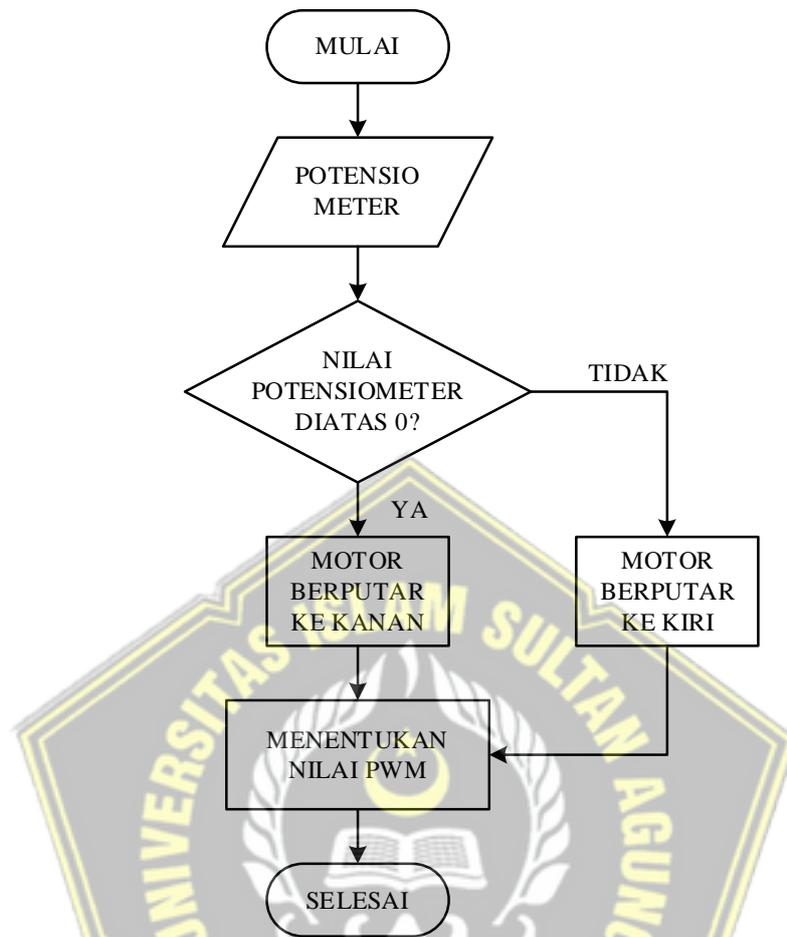
3.1 Model Penelitian

Penelitian ini diawali dengan membuat rangkaian simulasi menggunakan aplikasi *NI LabVIEW*. Simulasi dilakukan untuk melihat gambaran awal dari sistem yang akan dibuat nantinya. Komponen yang digunakan dalam simulasi adalah motor DC, Arduino UNO, driver motor BTS7960, dan potensiometer. Pada simulasi ini, potensiometer digunakan untuk mengatur besar PWM dan arah pergerakan motor DC.

Pada aplikasi NI LabView dibutuhkan aplikasi tambahan (*add on*) untuk dapat membaca masukan dari Arduino atau memberi masukan ke Arduino. Aplikasi tambahan tersebut adalah NI VISA dimana isinya adalah komponen untuk membaca port yang terhubung dengan Arduino, memberi masukan dengan cara komunikasi serial dan menerima keluaran serial dari Arduino. Sebelumnya, program dibuat dengan aplikasi Arduino IDE seperti program pembacaan pin analog, *rotary encoder*, kontrol kecepatan motor, dan pengaturan algoritma pemrograman lainnya. Setelah program berjalan dengan baik selanjutnya data dikirim ke LabVIEW melalui program serial.



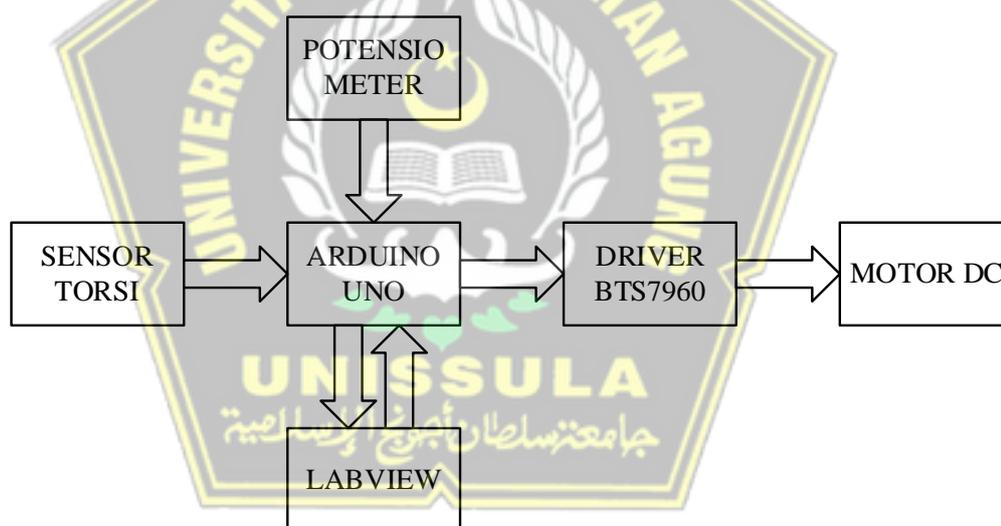
Gambar 3. 1 Blok Diagram Sistem Simulasi



Gambar 3. 2 Flowchart Sistem Simulasi

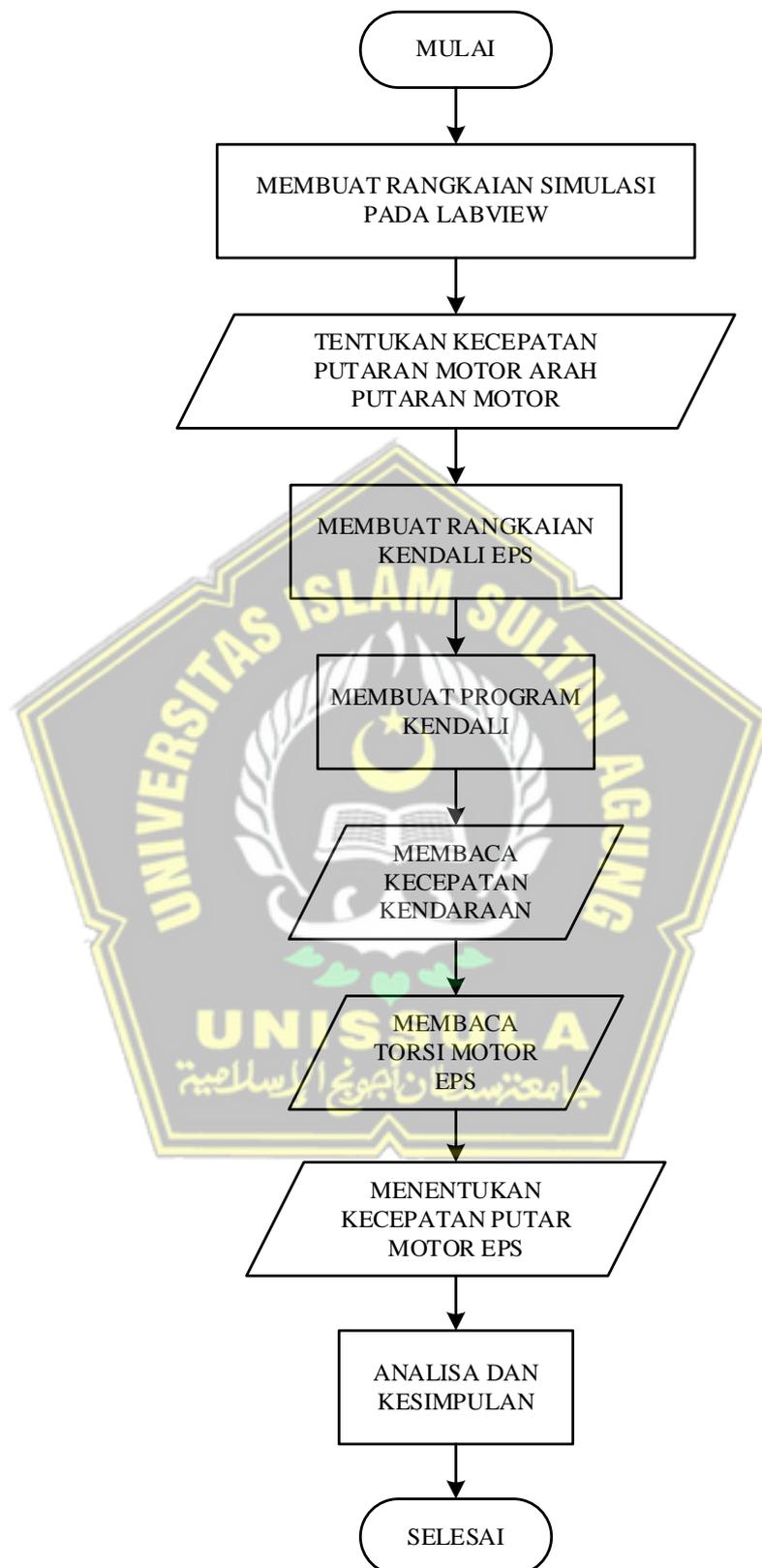
Gambar 3.1 dan Gambar 3.2 menjelaskan alur kerja dari sistem simulasi kendali kecepatan motor dengan cara mengatur nilai PWM nya berdasarkan nilai masukan dari Potensiometer. Nilai potensiometer yang dibaca oleh aplikasi *NI LabVIEW* adalah -255 sampai dengan 255. Ketika nilai potensio berada dibawah nilai 0 maka putaran motornya adalah ke arah kiri dan nilai PWM nya adalah 0 ketika nilai potensio 0 dan 255. ketika nilai potensio di angka -255. Ketika nilai potensio berada diatas nilai 0 maka putaran motornya adalah ke arah kanan dan nilai PWM nya adalah 0 ketika nilai potensio 0 dan 255 ketika nilai potensio di angka 255.

Setelah dilakukan simulasi, tahap selanjutnya adalah merancang sistem pengendalian yang sesungguhnya menggunakan *Electric Power Steering*. Perangkat *Electric Power Steering* ini memiliki dua masukan atau *input* data, yaitu *Torque Sensor EPS* dan *Vehicle Speed Sensor*. Kedua masukan tersebut akan mengirim data ke mikrokontroler untuk dapat diolah datanya. Mikrokontroler kemudian memberi perintah kepada motor yang terdapat di dalam *Electric Power Steering* sesuai dengan data masukan. Perintah tersebut berupa nilai kecepatan putaran motor pada *Electric Power Steering* tersebut. Motor *Electric Power Steering* yang bergerak akan menggerakkan *Rack and Pinion* pada mobil. Untuk membaca seberapa cepat setir mobil dibelokkan, diperlukan rotary encoder yang dihubungkan dengan perangkat *Electric Power Steering*.



Gambar 3. 3 Blok Diagram Sistem Kemudi Utama

Gambar 3.3 menjelaskan tentang cara kerja dari sistem yang akan dibuat. Mikrokontroler utama yang digunakan adalah Arduino UNO menerima data dari Potensiometer dan Sensor Torsi pada perangkat EPS serta bertukar data dengan aplikasi LabVIEW. Data yang diterima oleh perangkat Arduino tersebut kemudian digunakan untuk menentukan besaran nilai yang akan dikirim ke Driver Motor BTS7960 yang kemudian mengatur besaran kecepatan Motor DC.



Gambar 3. 4 Flowchart Pembuatan Alat

Gambar 3.4 menjelaskan alur pembuatan sistem Kendali Kecepatan Motor Pada Electric Power Steering (Eps) Berbasis Mikrokontroler Atmega328. Dimulai dengan pembuatan simulasi sistem dengan mengontrol arah putaran dan kecepatan motor DC. Dilanjut dengan pembuatan sistem kendali utama yang mana pada sistem ini akan membaca 2 masukan yaitu nilai kecepatan kendaraan dan nilai torsi EPS. Dari masukan tersebut digunakan untuk mengatur nilai kecepatan motor EPS. Setelah dilakukan pengambilan data maka didapatkan beberapa hasil Analisa dan Kesimpulan.

3.2 Lokasi dan Objek Penelitian

Dalam tugas akhir ini, tempat penelitian dilakukan di Laboratorium Teknik Elektro Fakultas Teknik Industri Universitas Islam Sultan Agung Semarang (UNISSULA).

Subjek pekerjaan adalah H-Molex *Electric Power Steering* mobil dari Fakultas Teknik Industri Universitas Islam Sultan Agung (UNISSULA).

3.3 Perancangan Program Kontrol PID

Sistem kendali PID (*Proportional Integral Derivative*) merupakan suatu perhitungan untuk menentukan nilai akurasi dari sistem dengan menggunakan umpan balik dari sistem yang akan dirancang. Suatu sistem kontrol PID terdiri dari tiga jenis kontrol yaitu P (Proporsional), I (Integral) dan D (Derivatif) yang masing-masing memiliki kelebihan dan kekurangan.

Dalam peng-aplikasiannya, masing-masing kontrol tersebut dapat bekerja sendiri maupun bersamaan. Perancangan sistem kendali PID yang perlu dilakukan adalah mengatur parameter P, I atau D agar tanggapan sinyal keluaran sistem terhadap masukan tertentu sebagaimana yang diinginkan [19]. Pada dasarnya sistem kendali PID ini sendiri dibuat untuk meminimalisir nilai *error* pada sebuah sistem.

Pada penelitian ini program menghitung PID dan kontrol kecepatan motor berdasarkan dari rotari dilakukan menggunakan Arduino IDE. Sedangkan aplikasi LabVIEW hanya digunakan untuk memonitor atau melihat nilai

keluaran dari program serial arduino.

3.3.1 Inisialisasi Program

Berisi deklarasi pin arduino yang digunakan, penamaan satuan yang akan digunakan dalam rumus perhitungan serta nilai awal dari beberapa parameter seperti nilai Kp, Ki, Kd dan besar *Pulses Per Revolution* (PPR) pada *rotary encoder*.

```
#include <Encoder.h>
#include <PID_v1.h>

#define EN_A 3
#define EN_B 2
Encoder E  ygt nc(EN_A, EN_B);

#define motora1 11
#define motora2 10

double Setpoint, Input, Output;
double Kp=2, Ki=5, Kd=1;
PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);
const int PPR = 100;

long previousMillis = 0;
long currentMillis = 0;

volatile long currentEncoder;
volatile long previousEncoder = 0;
volatile long oldPosition = 0;
volatile long newPosition;

float old_rot_speed=0;
```

```
float rot_speed;
```

Pada program ini menggunakan 2 *library* tambahan, yaitu Encoder.h yang berfungsi untuk membaca masukan dari *rotary encoder* dan PID_v1.h untuk perhitungan PID. Nilai Kp, Ki, dan Kd telah ditentukan di awal program lalu nilai *Set Point* didapat dari hasil perhitungan ketiga nilai tersebut.



3.3.2 Program Rumus Kecepatan

Berisi rumus penghitung kecepatan berdasarkan nilai keluaran dari *rotary encoder*.

```
float read_speed(void) {
    const int PPR = 100;
    currentEncoder = Enc.read();
    const int interval = 1000;
    currentMillis = millis();

    if ((currentMillis - previousMillis) > interval){

        previousMillis = currentMillis;
        rot_speed = ((currentEncoder - previousEncoder)*60)/PPR;
        previousEncoder = currentEncoder;
        return rot_speed;
    }
}
```

Pada program ini dilakukan perhitungan nilai keluaran berupa jumlah pulsa yang dihasilkan dari putaran motor pada EPS.

3.3.3 Program Motor

Berisi program untuk menjalankan motor secara otomatis.

```
void steer(){
    analogWrite(motora2, Output);
    delay(3000);
    analogWrite(motora1, 0-Output);
    delay(4000);
}
```

3.3.4 Program Eksekusi

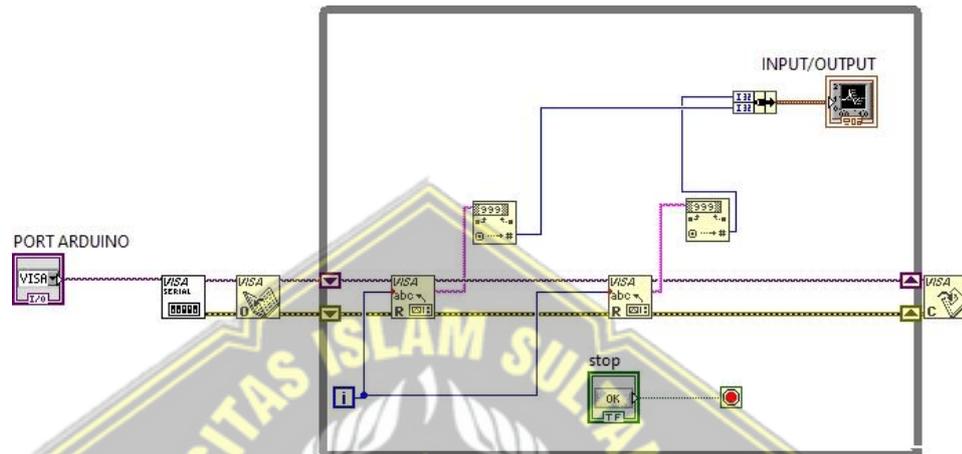
Berisi pemanggilan program yang sudah dibuat sebelumnya agar dapat bekerja dan mengirim data ke serial untuk dibaca oleh program LabVIEW.

```
void loop() {  
  Input = read_speed();  
  myPID.Compute();  
  steer();  
  
  Serial.println(Input);  
  Serial.println(Output);  
}
```



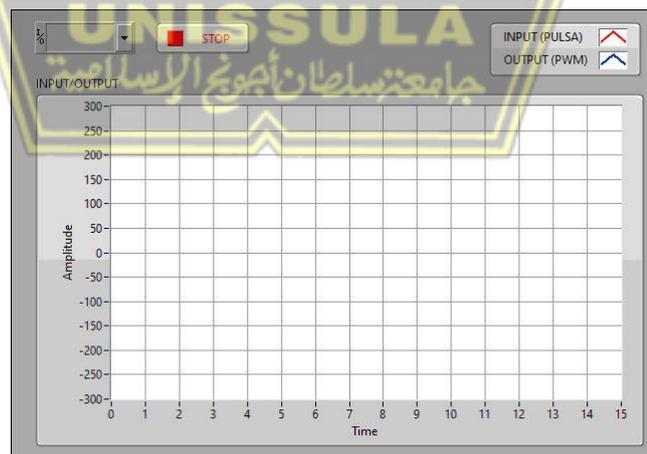
3.3.5 Monitoring Pada LabVIEW

Rangkaian pembaca keluaran serial dari Arduino yang sebelumnya sudah dibuat. Nilai keluaran serial yang akan ditampilkan adalah nilai *input rotary encoder* dan nilai *output* berupa besaran PWM. Nilai keluaran tersebut direpresentasikan menjadi sinyal gelombang pada LabVIEW.



Gambar 3. 5 Block Diagram Monitoring PID

Gambar 3.5 adalah *Block Diagram* yang berisi rangkaian pembacaan keluaran serial dari Arduino yang kedua hasil tersebut disatukan kedalam satu *Waveform Chart*. Hasil dari rangkaian tersebut lalu ditampilkan pada *Front Panel* pada Gambar 3.6 berikut.



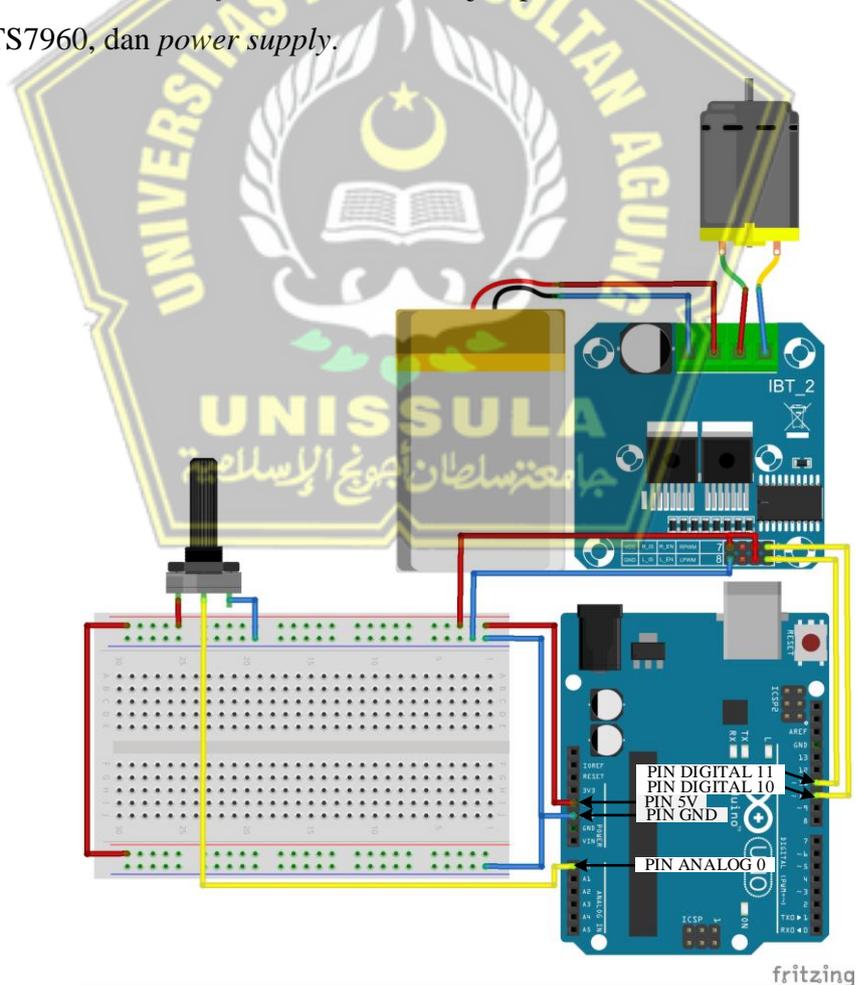
Gambar 3. 6 Front Panel Monitoring PID

3.4 Perancangan Program Simulasi Sistem

Pada tahap perancangan hardware ini yang akan dibahas adalah pembuatan simulasi alat menggunakan NI LabVIEW yang dihubungkan dengan *Board* Arduino UNO ATmega 328P. Lalu setelah itu dibahas pembuatan sistem kendali secara langsung menggunakan *Electric Power Steering* yang dikendalikan dengan mikrokontroler Arduino UNO ATmega 328P.

3.4.1 Perancangan Rangkaian Simulasi Sistem

Pada tahap perancangan simulasi sistem kerja *electric power steering* menggunakan aplikasi NI LabVIEW dilakukan perangkaian perangkat elektronik. Perangkat elektronik yang dibutuhkan adalah *board* Arduino UNO, Potensiometer, *Project Board*, kabel jumper, motor DC, driver motor DC BTS7960, dan *power supply*.



Gambar 3. 7 Rangkaian Simulasi

Pada rangkaian Gambar 3.7, potensiometer berfungsi sebagai masukan (*input*) data dan keluarannya (*output*) adalah motor DC. Pin *output* pada potensiometer dihubungkan dengan pin A0 (Analog 0) arduino, pin *input* dihubungkan dengan pin 5V Arduino, dan pin GND (*ground*) dihubungkan dengan pin GND Arduino. Pin VCC, R_EN, dan L_EN driver motor BTS7960 dihubungkan dengan pin VCC Arduino, tujuannya adalah supaya pin R_EN dan L_EN selalu mendapat masukan HIGH. Pin GND driver motor BTS7960 dihubungkan dengan pin GND Arduino. Pin LPWM dan RPWM pada driver motor BTS7960 masing – masing dihubungkan dengan pin PWM pada Arduino, yaitu dihubungkan dengan pin Digital 10 dan 11.

3.4.2 Perancangan Program Arduino

Untuk membaca nilai masukan yang dihasilkan dari putaran potensiometer digunakan program pembacaan pin analog. Nilai keluaran potensiometer adalah 0 sampai 1023, untuk mempermudah pengaplikasiannya pada motor maka nilai tersebut diubah menjadi -255 sampai 255 menggunakan map. Ketika nilai potensio diatas 0 maka motor akan berputar kekanan dan Ketika dibawah 0 maka akan berputar kekiri.

Berikut adalah program map untuk mengubah nilai masukan potensiometer.

```
void loop() {
    pot = analogRead(A1);

    pot = map(pot, 0, 1023, -255, 255);

    Serial.println(pot);

    motor();

    control();
}
```

Pada program diatas, nilai *default* potensiometer yang sebelumnya adalah

0 sampai 1023 diubah dengan program “map” menjadi -255 sampai dengan 255. Nilai keluaran tersebut adalah nilai PWM motor yang mana Ketika nilai dibawah 0 motor beputar ke kiri dan berputar ke kanan Ketika di atas 0.

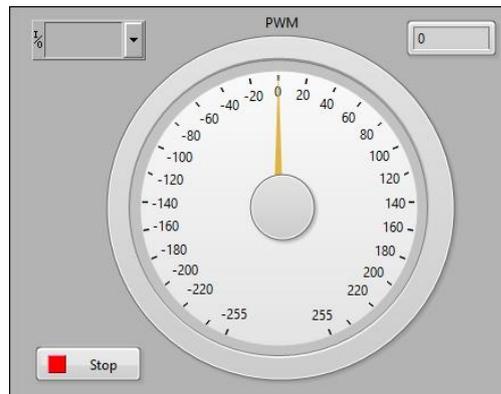


Kemudian program untuk membuat motor berputar kekanan Ketika nilai potensio diatas 0 dan kekiri Ketika dibawah 0 adalah sebagai berikut.

```
void control() {
  if (pot > 0) {
    cw = pot;
    ccw = 0;
  }
  else if (pot < 0) {
    cw = 0;
    ccw = 0-pot;
  }
  else {
    cw = 0;
    ccw = 0;
  }
}
```

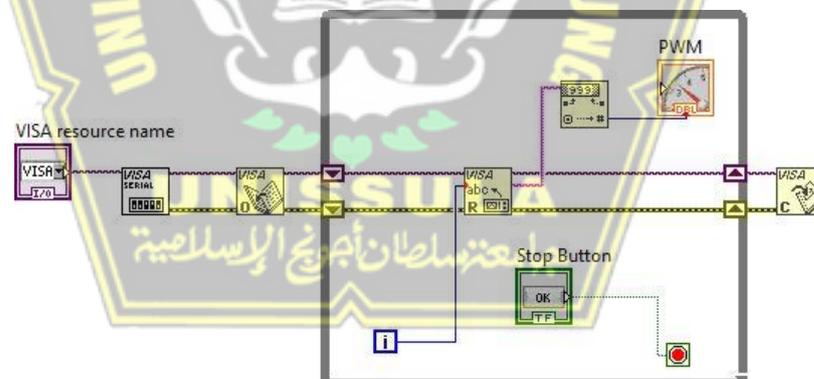
Program ini mengatur nilai PWM berdasarkan nilai dari potensiometer. Program pertama adalah Ketika nilai potensiometer berada diatas 0, maka PWM cw (*clockwise*) atau PWM arah putaran kanan bernilai sama dengan nilai potensiometer, dan PWM ccw (*counter clockwise*) atau PWM arah putaran kiri bernilai 0. Program selanjutnya Ketika nilai potensiometer dibawah angka 0, maka nilai PWM cw (*clockwise*) atau PWM arah putaran kanan adalah 0 dan PWM ccw (*counter clockwise*) atau PWM arah putaran kiri bernilai $0 - (pot)$ atau 0 dikurang nilai potensiometer. Contohnya Ketika masukannya bernilai -255 maka 0 dikurang -255 bernilai 255 pada PWM ccw (*counter clockwise*). Terakhir Ketika potensiometer bernilai 0 maka nilai PWM adalah 0 untuk cw (*clockwise*) maupun ccw (*counter clockwise*).

3.4.3 Perancangan Monitoring pada LabVIEW



Gambar 3. 8 Tampilan Front Panel Rangkaian Simulasi

Gambar 3.8 adalah *Front Panel* dari system simulasi pada LabVIEW yang berisi indikator arah putaran dan nilai PWM motor. Pada rangkaian ini, terdapat indikator berupa *dial* yang merepresentasikan putaran potensiometer sesuai dengan besaran PWM yaitu 0 sampai 255 untuk putaran kekanan dan 0 sampai -255 untuk putaran ke kiri. Disini juga terdapat tampilan kontrol, yaitu pemilih *serial port* yang digunakan oleh Arduino dan juga kontrol untuk mematikan *loop* sistem.



Gambar 3. 9 Block Diagram Rangkaian Simulasi

Diagram blok pada Gambar 3.9 berisi kode sumber grafis dari program LabVIEW. Konsep diagram blok adalah untuk memisahkan kode sumber grafis dari antarmuka pengguna dengan cara yang logis dan sederhana. Objek *front panel* muncul sebagai terminal pada diagram blok. Terminal pada diagram blok mencerminkan perubahan yang dibuat pada objek *front panel* yang sesuai dan sebaliknya.

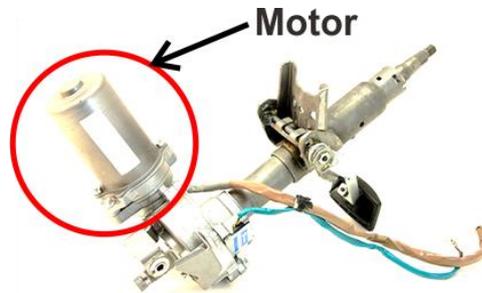
3.5 Perancangan Sistem Pada EPS

Pada tahap ini, mikrokontroler akan dihubungkan langsung dengan komponen yang ada pada modul EPS. Rangkaian sistem ini tidak jauh berbeda rangkaian simulasi sebelumnya. Motor EPS disambungkan dengan driver motor BTS 7960 dan diberi tegangan sebesar 12 Volt. Sensor torsi disambungkan pada pin analog sama seperti potensiometer pada simulasi.



3.5.1 Motor EPS

Posisi motor pada EPS dapat dilihat pada Gambar 3.10 berikut.



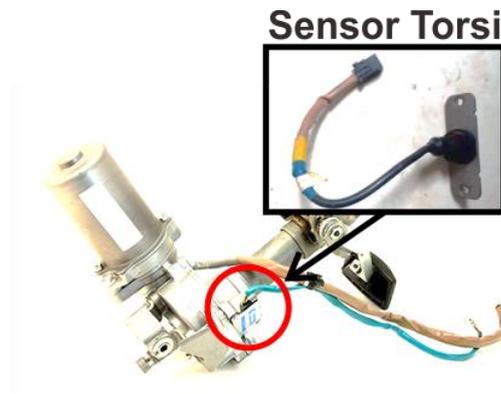
Gambar 3. 10 Motor EPS

EPS menggunakan motor listrik untuk membantu pengemudi kendaraan, tidak seperti sistem tradisional yang bekerja pada tekanan hidrolik yang diberikan melalui pompa yang digerakkan oleh mesin kendaraan. Pompa ini terus bekerja, baik setir sedang diputar atau tidak. Itu terus menempatkan beban pada mesin, mempengaruhi konsumsi bahan bakar kendaraan [20].

Motor yang biasa digunakan pada modul EPS adalah motor listrik DC. Untuk dapat dikendalikan dengan mikrokontroler Arduino, motor listrik DC perlu dihubungkan dengan driver motor. Pengontrol motor yang digunakan dalam penelitian ini adalah pengontrol motor BTS7960. Pengontrol motor ini dapat digunakan dengan motor dengan tegangan input maksimum 27 Volt DC dan arus 43 ampere.

3.5.2 Sensor Torsi EPS

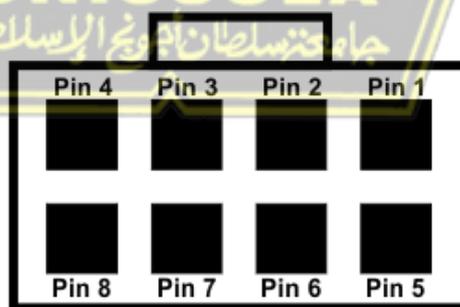
Bentuk sensor torsi dapat dilihat pada Gambar 3.11 berikut.



Gambar 3. 11 Sensor Torsi EPS Avanza Xenia

Torsi adalah gaya putaran terhadap suatu sumbu, yang dapat diterapkan searah jarum jam atau berlawanan arah jarum jam. Salah satu contoh penggunaannya adalah pada mekanisme kemudi kendaraan untuk berbelok di tikungan, pengemudi menerapkan gaya ke roda kemudi yang menerapkan torsi langsung ke kolom kemudi.

Sensor torsi biasanya terletak pada kolom kemudi untuk memberikan informasi kepada *control module*. Sensor torsi memberikan informasi tentang seberapa sejauh setir diputar dan seberapa cepat putarannya. Pada penelitian ini, *control module* diganti dengan mikrokontroler Arduino UNO dengan memberikan keluaran analog yang akan dibaca oleh Arduino untuk kemudian melakukan perintah ke motor.



Gambar 3. 12 Pinout Sensor Torsi

Dari Gambar 3.12, pin yang akan digunakan sebagai masukan dari sensor torsi ke Arduino adalah pin 5, pin 6, pin 7, dan pin 8 dengan rincian sebagai berikut.

Pin 5 : Pin *Torque Sensor* 1

Pin 6 : Pin VCC
Pin 7 : Pin *Torque Sensor 2*
Pin 8 : Pin GND



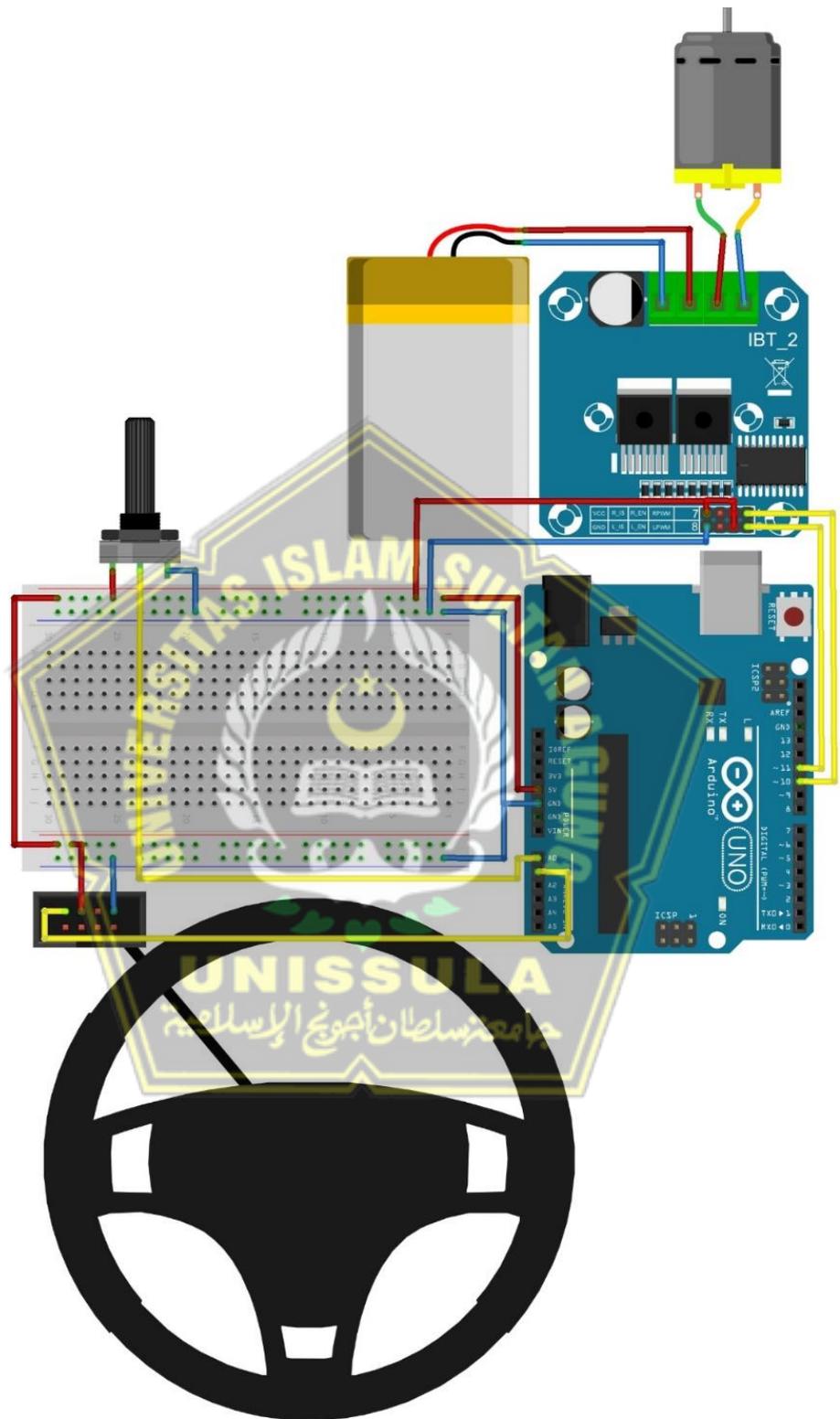
3.5.3 Vehicle Speed Sensor (VSS)

Vehicle Speed Sensor atau sensor kecepatan kendaraan adalah sebuah sensor yang berfungsi sebagai pembaca kecepatan sebuah kendaraan. Memiliki cara kerja yang hampir sama dengan *tachometer* yaitu membaca putaran roda kendaraan. Pada percobaan ini pembacaan kecepatan kendaraan belum dapat dilakukan dikarenakan mobil belum dapat dijalankan. Sebagai penggantinya menggunakan potensiometer untuk men-simulasikan besaran kecepatan kendaraan. Kecepatan kendaraan pada percobaan kali ini, nilai kecepatan kendaraan dibuat dengan nilai 0 km/jam sampai 40 km/jam.

3.5.4 Rangkaian Sistem Kemudi

Rangkain kemudi EPS menggunakan Arduino UNO ini tidak berbeda jauh dengan rangkaian simulasi sistem kemudi yang sebelumnya sudah dibuat sebelum menerapkannya pada EPS secara langsung. Motor EPS disambungkan dengan driver motor BTS 7960 agar dapat dikendalikan besar PWM-nya dengan Arduino. Pin LPWM dan RPWM pada driver motor BTS 7960 masing – masing dihubungkan dengan pin digital 10 dan 11 mikrokontroler yang juga merupakan pin keluaran PWM.

Sensor torsi dihubungkan langsung pada Arduino dimana pin VCC dan GND sensor torsi dihubungkan ke pin VCC dan GND Arduino. Sedangkan pin keluaran sensor torsi dihubungkan dengan pin masukan analog Arduino. Potensiometer yang digunakan untuk men-simulasikan *Vehicle Speed Sensor* (VSS) juga dihubungkan dengan pin masukan analog Arduino. Rangkaian Sistem kendali dapat dilihat pada Gambar 3.13.



fritzing

Gambar 3. 13 Rangkaian Kendali EPS

3.5.5 Program Arduino

Inti dari program ini adalah mengubah nilai masukan dari potensiometer dari 0 sampai 1023 menjadi 0 sampai 40 yang merepresentasikan nilai kecepatan kendaraan. Lalu nilai masukan dari sensor torsi diubah menjadi 0 sampai 50 yang mana nilai tersebut diambil dari nilai tegangan yang terukur pada sensor torsi yaitu 0 sampai 5 volt. Untuk mempermudah dalam pemrograman, nilai tegangan tersebut didefinisikan sebagai 0 untuk tegangan 0 dan 50 untuk 5 volt. Ketika berada pada kecepatan terendah dan besar torsi maksimal maka motor DC akan berputar pada kecepatan maksimal, sedangkan pada kecepatan tertinggi, motor tidak mendapat tegangan atau sama dengan diputar dengan tenaga pengemudi sendiri. Berikut adalah program kendali kecepatan motor EPS berdasarkan nilai masukan sensor torsi dan kecepatan kendaraan.

```
void loop() {
  Torsi = analogRead(PinTS);
  Torsi = map(Torsi, 0, 1023, 0, 50);
  Kecepatan = analogRead(Pot);
  Kecepatan = map(Kecepatan, 0, 1023, 0, 40);
  Serial.println(Torsi);
  Serial.println(Kecepatan);
  motor();

  TingkatKecepatan();

  Serial.println(cw);
  Serial.println(ccw);
}
```

Pada program tersebut dilakukan *mapping* atau penggantian pada nilai masukan sensor torsi yang pada awalnya terbaca bernilai minimal 0 dan nilai maksimal 1023 yang kemudian diubah ke nilai minimal 0 sampai maksimal 50. Pada saat *idle* atau posisi setir diam nilai *default* yang terbaca adalah antara 510

sampai 514 dan Ketika sudah dirubah nilai yang terbaca adalah antara 24 sampai 26 atau pada tegangan 2,4 sampai 2,6 volt. Nilai masukan dari potensiometer yang pada awalnya juga bernilai 0 sampai dengan 1023 dirubah menjadi 0 sampai dengan 40. Nilai ini digunakan untuk menunjukkan besaran nilai kecepatan kendaraan yaitu 0 sampai 40 km/jam.

Selain itu, terdapat program pengiriman data ke serial yaitu nilai Torsi dan nilai Kecepatan yang nantinya akan dibaca oleh program LabVIEW. Pada program ini juga dilakukan pemanggilan program void pembagian kecepatan putaran motor EPS berdasarkan tingkat kecepatan kendaraan. Untuk penerapan nilai nilai tersebut adalah sebagai berikut.

```
void TingkatKecepatan() {
    // Kecepatan Sedang
    if (Kecepatan >= 20 && Kecepatan <30) {
        if(Torsi <= 50 && Torsi > 45) {
            cw = 85;
            ccw = 0;
        }
        else if(Torsi <= 45 && Torsi > 40) {
            cw = 68;
            ccw = 0;
        }
        else if(Torsi <= 40 && Torsi > 35) {
            cw = 51;
            ccw = 0;
        }
        else if(Torsi <= 35 && Torsi > 30) {
            cw = 34;
            ccw = 0;
        }
        else if(Torsi <= 30 && Torsi > 28) {
            cw = 17;
```

```

    ccw = 0;
}
else if(Torsi >= 20 && Torsi < 22) {
    cw = 0;
    ccw = 17;
}
else if(Torsi >= 15 && Torsi < 20) {
    cw = 0;
    ccw = 34;
}
else if(Torsi >= 10 && Torsi < 15) {
    cw = 0;
    ccw = 51;
}
else if(Torsi >= 5 && Torsi < 10) {
    cw = 0;
    ccw = 68;
}
else if(Torsi >= 0 && Torsi < 5) {
    cw = 0;
    ccw = 85;
}
else {
    cw = 0;
    ccw = 0;
}
}
}

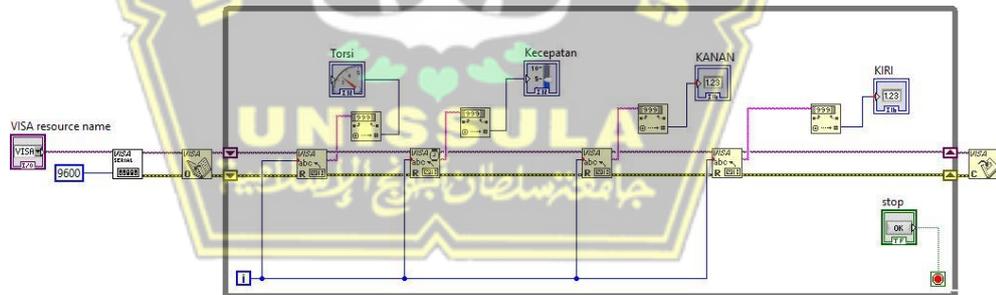
```

Pada program tersebut tingkat kecepatan dibuat menjadi antara 0 sampai dengan 40, yaitu Ketika kecepatan lebih besar atau sama dengan 10 dan lebih kecil dari 20 untuk kecepatan rendah, kecepatan lebih besar atau sama dengan

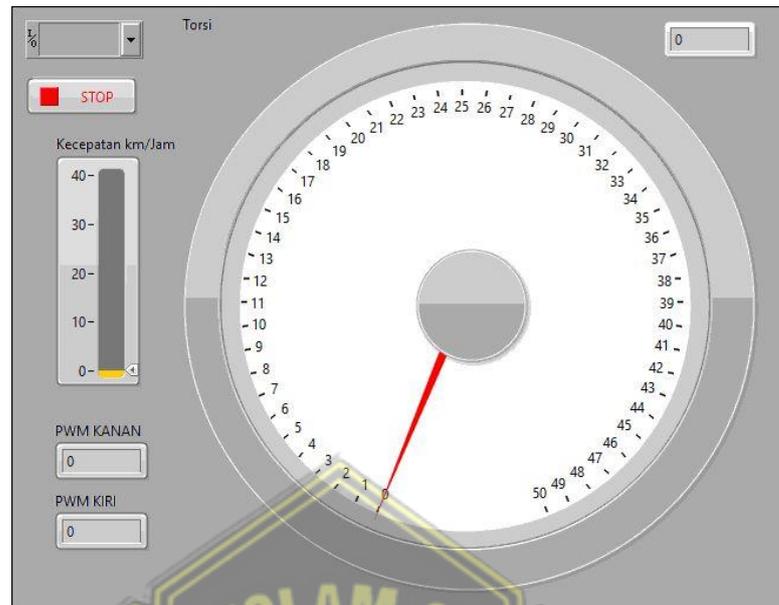
20 dan lebih kecil dari 30 untuk kecepatan sedang, kecepatan lebih besar atau sama dengan 30 dan lebih kecil dari 40 untuk kecepatan tinggi. Besar PWM untuk motor EPS juga dibatasi sesuai dengan tingkat kecepatan kendaraan, yaitu Ketika kecepatan rendah besar PWM maksimal adalah $\frac{2}{3}$ besar total PWM yaitu 170, Ketika kecepatan sedang besar PWM maksimal adalah $\frac{1}{3}$ besar total PWM yaitu 85 dan Ketika kecepatan tinggi besar PWM adalah 0 atau tidak ada nilai PWM-nya. Ketika tingkat kecepataannya dibawah 10 atau pada program ditulis dengan program “else” maka besar PWM adalah besaran maksimal dari PWM atau maksimalnya adalah 255.

3.5.6 Monitoring Pada LabVIEW

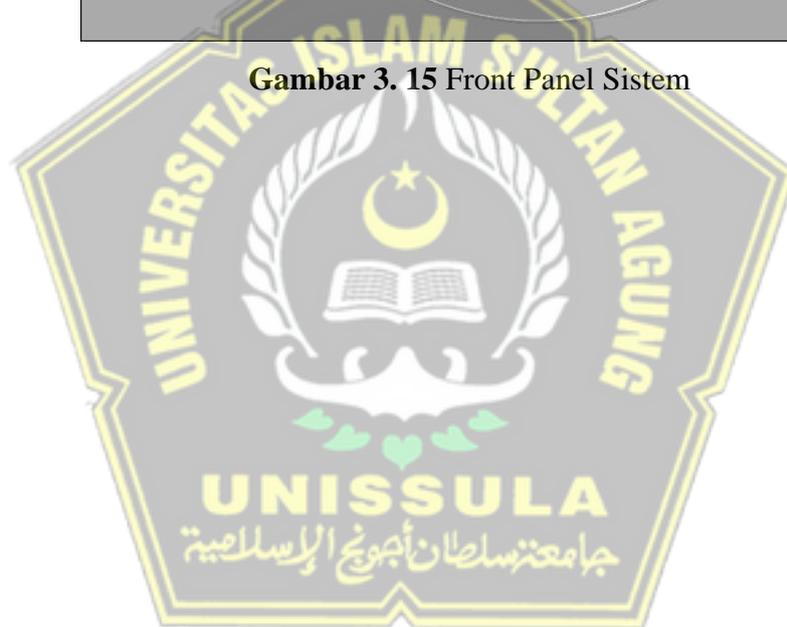
Pada percobaan ini, nilai yang akan ditampilkan pada LabVIEW adalah nilai tingkat kecepatan yang direpresentasikan kedalam bentuk *vertical pointer slide*, putaran kemudi yang direpresentasikan kedalam bentuk *gauge*. Selain dua hal tersebut, disini juga diberikan tampilan indicator kecepatan putaran motor EPS sesuai dengan arahnya. *Block Diagram* dan *Front Panel* dari program ini dapat dilihat pada Gambar 3.14 dan Gambar 3.15 berikut.



Gambar 3. 14 Block Diagram Sistem



Gambar 3. 15 Front Panel Sistem

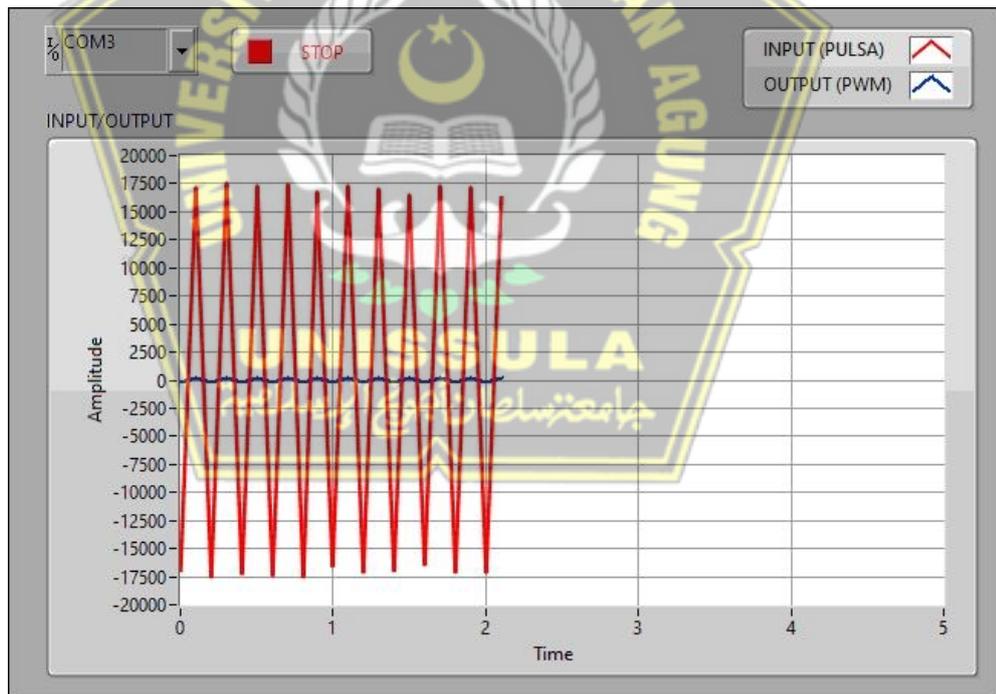


BAB IV DATA DAN ANALISA

Pada bagian ini berisi hasil percobaan yang telah dilakukan mengenai kendali kecepatan motor EPS dengan mikrokontroler Arduino UNO ATmega328. Data disajikan dalam gambar yang ditampilkan pada LabVIEW, Tampilan *Serial plotter* Arduino IDE dan data grafik dari LabVIEW yang di ekspor kedalam excel.

4.1 Pengujian Kontrol PID

Pengujian yang pertama dilakukan adalah pengujian kontrol PID dengan program arduino dan monitoring dengan LabVIEW. Pada tahap pengujian kontrol PID menggunakan program Arduino yang telah dilakukan, diperoleh hasil yang ditampilkan pada Gambar 4.1 berikut.

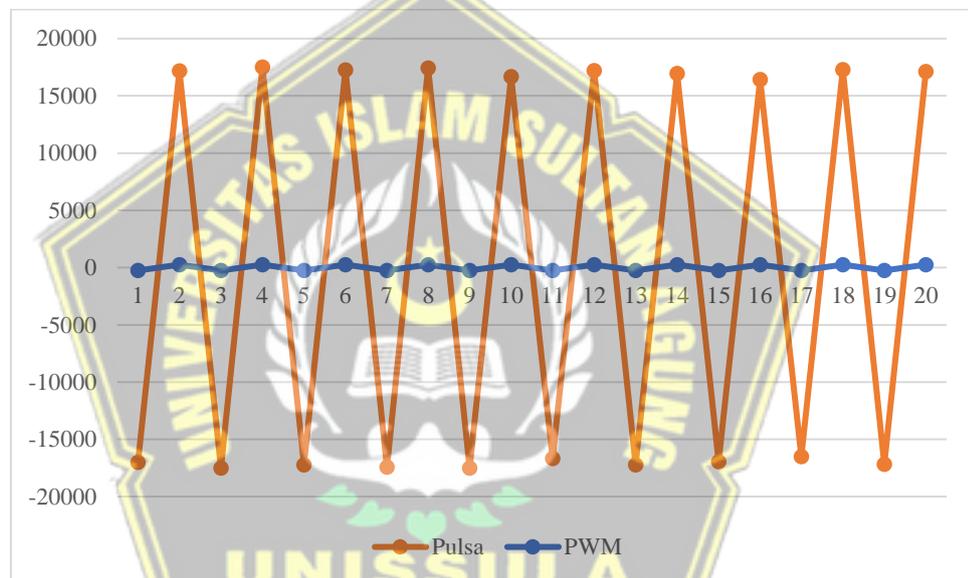


Gambar 4. 1 Nilai Masukan dan Keluaran yang diperoleh

Nilai yang ditampilkan pada *front panel* LabVIEW tersebut adalah nilai yang terbaca oleh *rotary encoder* yang dideskripsikan sebagai *Input* dan juga nilai keluaran berupa nilai PWM yang akan digunakan untuk mengontrol

kecepatan motor. Nilai masukan yang dihasilkan yang dihasilkan selalu berubah – ubah, akan tetapi nilai keluarannya antara -255 sampai dengan 255. Hal ini terjadi karena pada program telah dibatasi untuk nilai keluaran minimal adalah -255 dan nilai keluarannya maksimal adalah 255. Pembatasan nilai keluaran tersebut menyesuaikan dengan fungsinya yaitu untuk mengontrol nilai PWM atau tingkat kecepatan putaran motor pada EPS.

Diagram yang ditampilkan dalam *front panel* tersebut kemudian diekspor ke dalam program excel untuk dilihat perbandingan nilainya secara lebih mendetail. Berikut adalah grafik nilai masukan dan keluaran dari Gambar 4.1.



Gambar 4. 2 Grafik nilai masukan dan keluaran program PID

Dari Gambar 4.2 dapat dilihat grafik nilai masukan yaitu besaran pulsa yang dihasilkan setiap putaran setir kemudi ke kanan dan ke kiri dengan garis berwarna oranye dan nilai keluaran control PWM yang berwarna biru. Rincian nilai pada grafik dapat dilihat pada **Tabel 4.1** berikut.

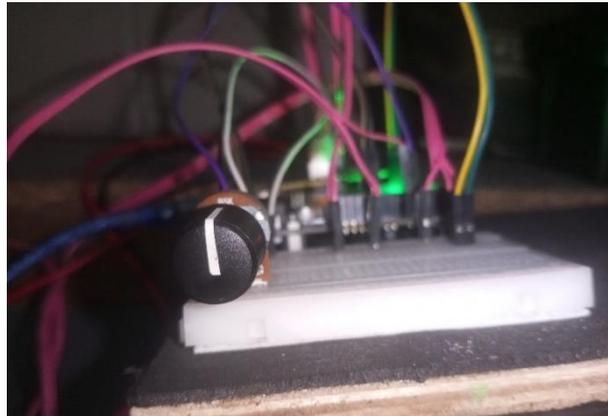
Tabel 4. 1 Tabel nilai masukan dan keluaran program PID

No.	Putaran Kanan		Putaran Kiri	
	Pulsa (Input)	PWM (Output)	Pulsa (Input)	PWM (Output)
1	-16993	-255	17181	255
2	-17509	-255	17511	255
3	-17222	-255	17266	255
4	-17426	-255	17445	255
5	-17507	-255	16678	255
6	-16661	-255	17224	255
7	-17218	-255	16955	255
8	-16953	-255	16435	255
9	-16508	-255	17293	255
10	-17169	-255	17123	255
Rata-Rata:	-17116,6	-255	17111,1	255

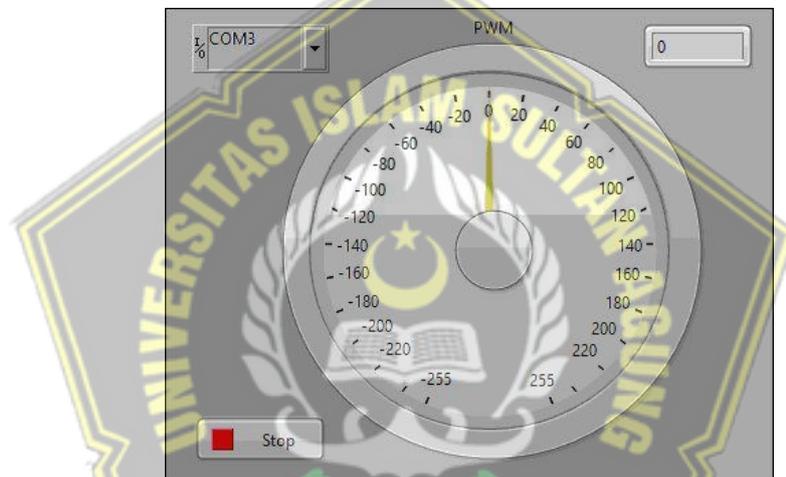
Pada Tabel 4.1 dapat dilihat bahwa nilai masukan terus mengalami perubahan dengan nilai rata – rata untuk putaran ke kanan adalah -17116,6 dan nilai rata – rata saat berputar ke arah kiri adalah 17111,1 dengan nilai pulsa terendah yang tercapai Ketika berputar ke kanan adalah -17509 dan nilai tertinggi Ketika berputar ke kiri adalah 17511 yang terjadi pada percobaan ke 2. Nilai rata – rata keluaran saat berputar ke arah kanan adalah -255 dan nilai rata – rata keluaran saat berputar ke arah kiri adalah 255.

4.2 Pengujian Program Simulasi

Pada pengujian program simulasi ini dilakukan percobaan apakah keluaran dari potensiometer sesuai dengan tampilan yang terlihat pada LabVIEW dan juga dapat menentukan arah putaran motor sesuai nilainya. Berikut pengujian Ketika potensiometer berada pada posisi 0 atau posisi tengah, posisi putaran kekiri dengan nilai maksimal yaitu -255 dan posisi putaran kekanan dengan nilai maksimal 255 beserta tampilan pada LabVIEW. Hasil pengujian dapat dilihat pada Gambar 4.3 dan Gambar 4.4 untuk nilai potensio pada posisi tengah.



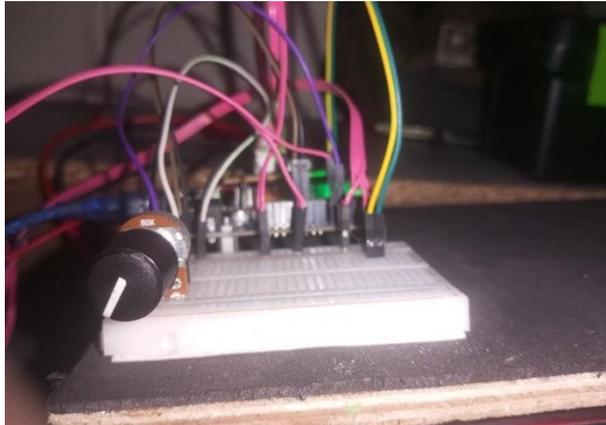
Gambar 4. 3 Potensiometer pada posisi nol



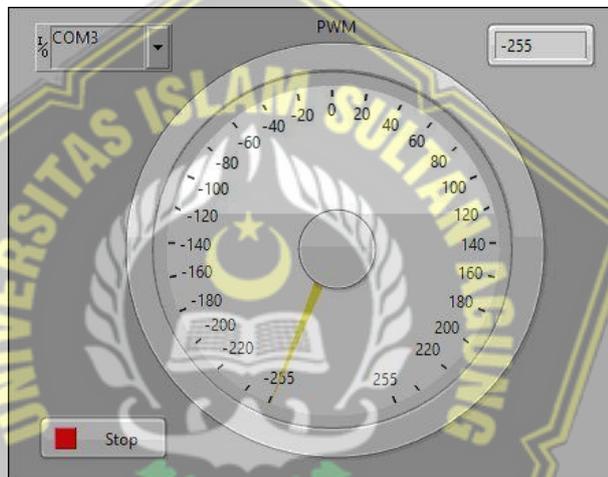
Gambar 4. 4 Tampilan LabVIEW saat Potensiometer pada posisi nol

Pada saat potensiometer berada pada posisi tengah atau pada nilai 0, maka nilai PWM dari motor DC juga akan bernilai 0 atau tidak berputar sama sekali.

Hasil pengujian potensiometer Ketika diputar kekiri dapat dilihat pada Gambar 4.5 dan Gambar 4.6 berikut.



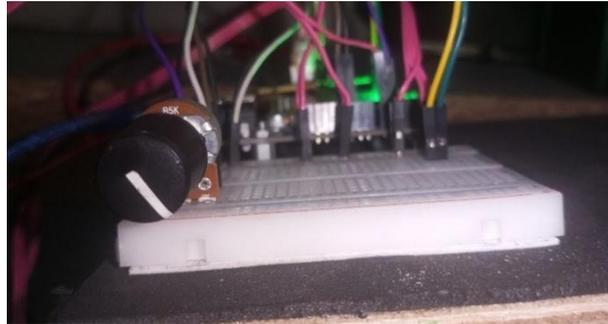
Gambar 4. 5 Potensiometer diputar kekiri



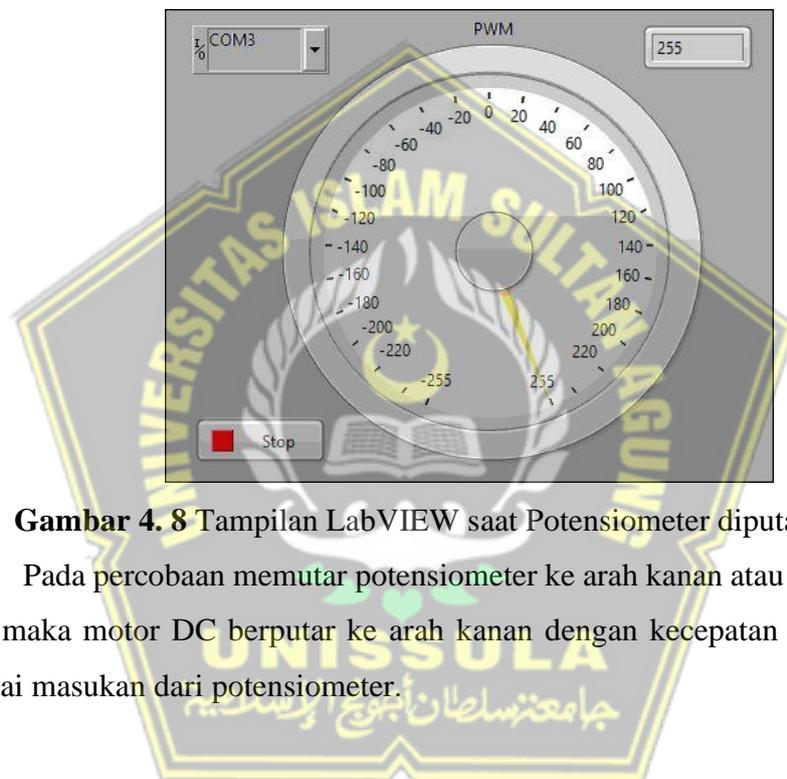
Gambar 4. 6 Tampilan LabVIEW saat Potensiometer diputar kekiri

Pada percobaan memutar potensiometer ke arah kiri atau bernilai dibawah 0, maka motor DC berputar ke arah kiri dengan kecepatan sesuai dengan nilai masukan dari potensiometer.

Hasil pengujian potensiometer Ketika diputar kekanan dapat dilihat pada Gambar 4.7 dan Gambar 4.8.



Gambar 4. 7 Potensiometer diputar kekanan



Gambar 4. 8 Tampilan LabVIEW saat Potensiometer diputar kekanan

Pada percobaan memutar potensiometer ke arah kanan atau bernilai diatas 0, maka motor DC berputar ke arah kanan dengan kecepatan sesuai dengan nilai masukan dari potensiometer.

4.3 Pengujian Program Kendali Motor EPS

Pada tahap ini pengujian dilakukan dengan memutar gagang setir mobil H-Molex yang mana akan memicu sensor torsi dan memberikan masukan ke program Arduino. Tingkat kecepatan dibagi menjadi 4, yaitu kecepatan nol sampai dengan 10 atau dapat diasumsikan kendaraan berhenti, kecepatan 10 sampai dengan 20 untuk kecepatan rendah, kecepatan 20 sampai dengan 30 untuk kecepatan sedang dan kecepatan 30 sampai dengan 40 untuk kecepatan tinggi.

Pembagian nilai PWM berdasarkan tingkat kecepatannya telah diatur oleh

program yang sudah ditulis pada BAB sebelumnya. Pada tampilan *front panel* nilai PWM dapat dilihat pada bagian bawah tingkat kecepatan atau pada bagian kiri bawah dan dapat dilihat sesuai arah putaran setir mobil.

Pengujian dilakukan 6 kali dimana pada pengujian pertama adalah pengujian tanpa beban dan pengujian kedua hingga keenam adalah pengujian dengan beban 1kg hingga 5 kg. Beban yang dipasang berupa *barble* yang berat satuannya adalah 0,5 kg dan dipasang pada sisi kiri dan kanan. Contoh posisi pemasangan pemberat pada roda pada saat tanpa beban dan berbeban dapat dilihat pada Gambar 4.9 pada saat tanpa beban dan Gambar 4.10 saat posisi roda diberi beban 1 kg.



Gambar 4. 9 Roda depan tanpa beban



Gambar 4. 10 Posisi Roda dengan beban 1 kg

Nilai PWM dari setiap kecepatan diatur sesuai dengan program yang telah tercantum pada bab sebelumnya. Rincian nilai PWM berdasarkan tingkat kecepatannya dan besar tegangan sensor torsi yang terbaca dapat dilihat pada Tabel 4.2 berikut.

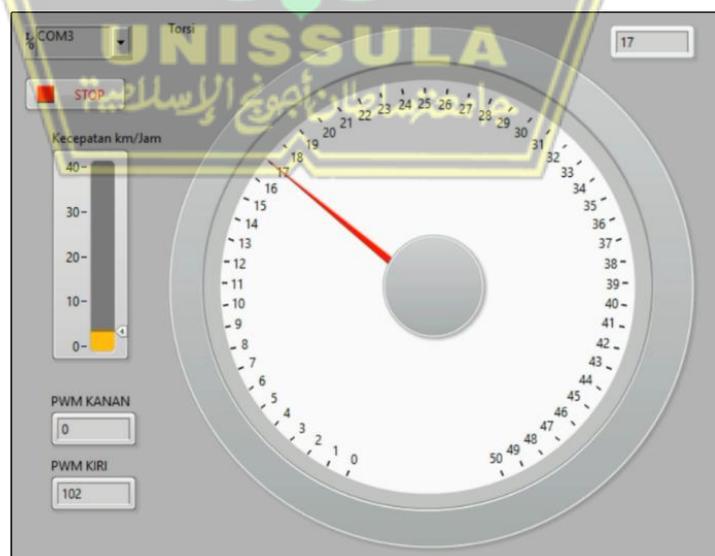
Tabel 4. 2 Tabel Nilai PWM berdasarkan nilai kecepatan

Kecepatan	Nilai Tegangan Sensor Torsi	Skala PWM Putaran Kanan	Skala PWM Putaran Kiri
0 - 10	47 – 50	255	0
	45 - 47	229	0
	42 – 45	204	0
	40 - 42	178	0
	37 – 40	153	0
	35 - 37	127	0
	32 – 35	102	0
	30 - 32	76	0
	28 – 30	51	0
	22 – 28	0	0
	20 – 28	0	51
	18 - 20	0	76
	15 – 18	0	102
	13 - 15	0	127
	10 – 13	0	153
	7 - 10	0	178
	5 – 7	0	204
	3 - 5	0	229
0 – 3	0	255	
10 - 20	45 – 50	170	0
	40 – 45	136	0
	35 – 40	102	0
	30 – 35	68	0
	28 – 30	34	0
	22 – 28	0	0
	20 – 28	0	34
	15 – 20	0	68
	10 – 15	0	102
	5 – 10	0	136
	0 – 5	0	170

20 - 30	45 – 50	85	0
	40 – 45	68	0
	35 – 40	51	0
	30 – 35	34	0
	28 – 30	17	0
	22 – 28	0	0
	20 – 28	0	17
	15 – 20	0	34
	10 – 15	0	51
	5 – 10	0	68
	0 – 5	0	85
30 - 40	0 - 50	0	0

Nilai pada Tabel 4.2 adalah nilai yang ditentukan pada program pengendalian yang telah dibuat sebelumnya. Nilai – niali tersebut dibagi berdasarkan tingkat kecepatan dan pada setiap tingkat kecepatan nilai PWM dibagi berdasarkan nilai masukan dari sensor torsi EPS.

Gambar sampel hasil pengujian program kendali kecepatan motor EPS berdasarkan nilai kecepatan kendaraan dan nilai torsi dari EPS dapat dilihat pada Gambar 4.11 berikut.

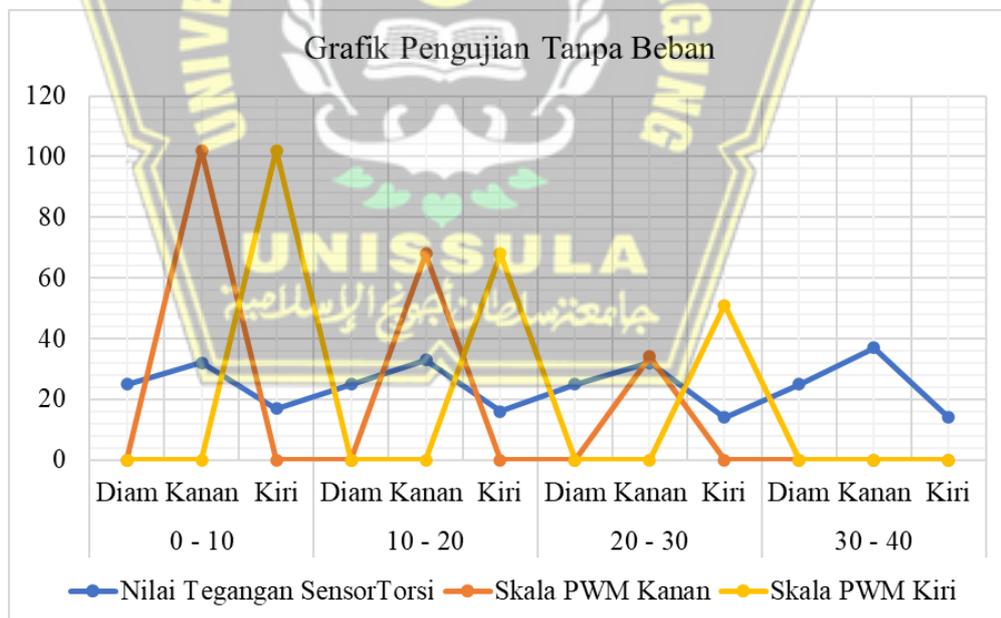


Gambar 4. 11 Tanpa Beban Kecepatan 0 sampai 10 Putaran Kiri

Data hasil percobaan Ketika roda depan tidak diberi beban dapat dilihat pada Tabel 4.3 dan grafik pada Gambar 4.12 berikut.

Tabel 4. 3 Tabel Hasil Pengujian Tanpa Beban

Kecepatan (km/jam)	Putaran Setir	Nilai Tegangan Sensor Torsi	Skala PWM Kanan	Skala PWM Kiri
0 - 10	Diam	25	0	0
	Kanan	32	102	0
	Kiri	17	0	102
10 - 20	Diam	25	0	0
	Kanan	33	68	0
	Kiri	16	0	68
20 - 30	Diam	25	0	0
	Kanan	32	34	0
	Kiri	14	0	51
30 - 40	Diam	25	0	0
	Kanan	37	0	0
	Kiri	14	0	0



Gambar 4. 12 Grafik Pengujian Tanpa Beban

Dari hasil pengujian tanpa beban pada Tabel 4.3 dan Gambar 4.12 didapatkan beberapa hasil. Pada Kecepatan terendah, yaitu 0 sampai 10 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 17 atau 1,7 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 102 dan 0 ketika posisi *Standby*.

Pada Kecepatan rendah, yaitu 10 sampai 20 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 33 atau 3,3 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 68 dan 0 ketika posisi *Standby*.

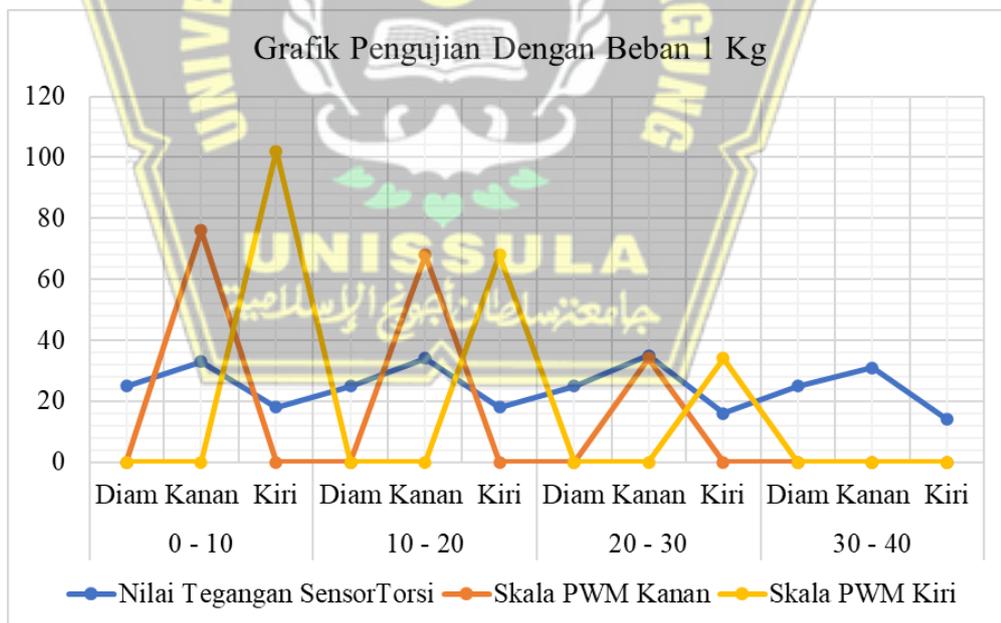
Pada Kecepatan sedang, yaitu 20 sampai 30 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 14 atau 1,4 Volt. Skala PWM pada saat diputar kekanan bernilai 34, saat diputar kekiri bernilai 51 dan 0 ketika posisi *Standby*.

Pada Kecepatan tinggi, yaitu 30 sampai 40 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 37 atau 3,7 Volt dan Ketika diputar ke arah kiri nilainya adalah 14 atau 1,4 Volt. Skala PWM pada saat diputar kekanan, kekiri maupun *standby* sama – sama bernilai 0.

Data hasil percobaan Ketika roda depan diberi beban 1 Kg dapat dilihat pada Tabel 4.4 dan grafik pada Gambar 4.13 berikut.

Tabel 4. 4 Tabel Hasil Pengujian Dengan Beban 1 Kg

Kecepatan (km/jam)	Putaran Setir	Nilai Tegangan Sensor Torsi	Skala PWM Kanan	Skala PWM Kiri
0 - 10	Diam	25	0	0
	Kanan	33	76	0
	Kiri	18	0	102
10 - 20	Diam	25	0	0
	Kanan	34	68	0
	Kiri	18	0	68
20 - 30	Diam	25	0	0
	Kanan	35	34	0
	Kiri	16	0	34
30 - 40	Diam	25	0	0
	Kanan	31	0	0
	Kiri	14	0	0



Gambar 4. 13 Grafik Pengujian Dengan Beban 1 Kg

Dari hasil pengujian dengan beban 1 Kg pada Tabel 4.4 dan Gambar 4.13 didapatkan beberapa hasil. Pada Kecepatan terendah, yaitu 0 sampai 10 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 33 atau 3,3 Volt dan Ketika diputar ke arah kiri nilainya adalah 18 atau 1,8 Volt. Skala PWM pada saat diputar kekanan bernilai 76, saat diputar kekiri bernilai 102 dan 0 ketika posisi *Standby*.

Pada Kecepatan rendah, yaitu 10 sampai 20 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 34 atau 3,4 Volt dan Ketika diputar ke arah kiri nilainya adalah 18 atau 1,8 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 68 dan 0 ketika posisi *Standby*.

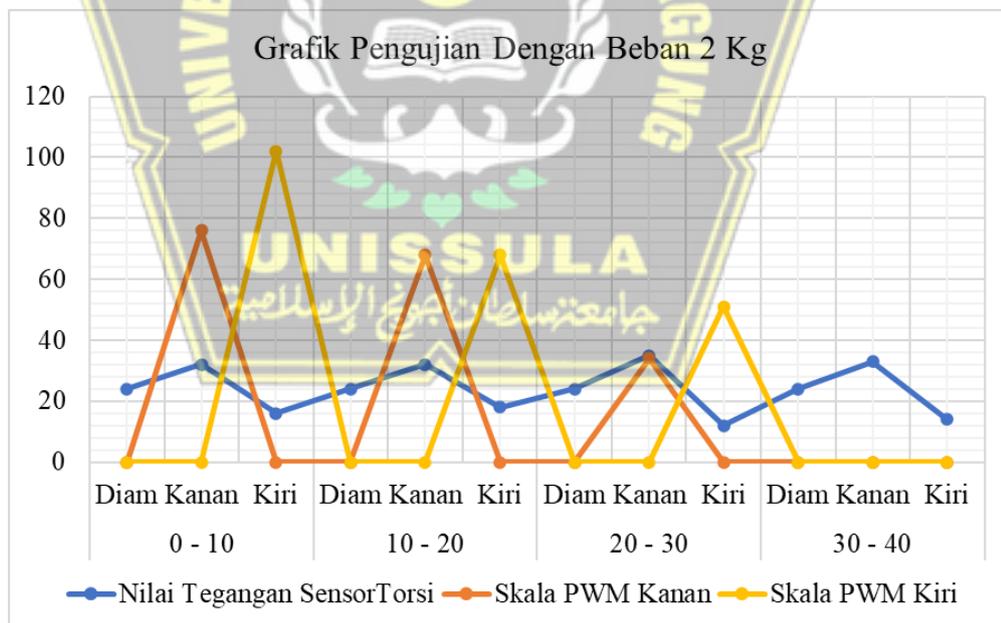
Pada Kecepatan sedang, yaitu 20 sampai 30 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 35 atau 3,5 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 34 dan 0 ketika posisi *Standby*.

Pada Kecepatan tinggi, yaitu 30 sampai 40 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 25 atau 2,5 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 31 atau 3,1 Volt dan Ketika diputar ke arah kiri nilainya adalah 14 atau 1,4 Volt. Skala PWM pada saat diputar kekanan, kekiri maupun *standby* sama – sama bernilai 0.

Data hasil percobaan Ketika roda depan diberi beban 2 Kg dapat dilihat pada Tabel 4.5 dan grafik pada Gambar 4.14 berikut.

Tabel 4. 5 Tabel Hasil Pengujian Dengan Beban 2 Kg

Kecepatan (km/jam)	Putaran Setir	Nilai Tegangan Sensor Torsi	Skala PWM Kanan	Skala PWM Kiri
0 - 10	Diam	24	0	0
	Kanan	32	76	0
	Kiri	16	0	102
10 - 20	Diam	24	0	0
	Kanan	32	68	0
	Kiri	18	0	68
20 - 30	Diam	24	0	0
	Kanan	35	34	0
	Kiri	12	0	51
30 - 40	Diam	24	0	0
	Kanan	33	0	0
	Kiri	14	0	0



Gambar 4. 14 Grafik Pengujian Dengan Beban 2 Kg

Dari hasil pengujian dengan beban 2 Kg pada Tabel 4.5 dan Gambar 4.14 didapatkan beberapa hasil. Pada Kecepatan terendah, yaitu 0 sampai 10 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan bernilai 76, saat diputar kekiri bernilai 102 dan 0 ketika posisi *Standby*.

Pada Kecepatan rendah, yaitu 10 sampai 20 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 18 atau 1,8 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 68 dan 0 ketika posisi *Standby*.

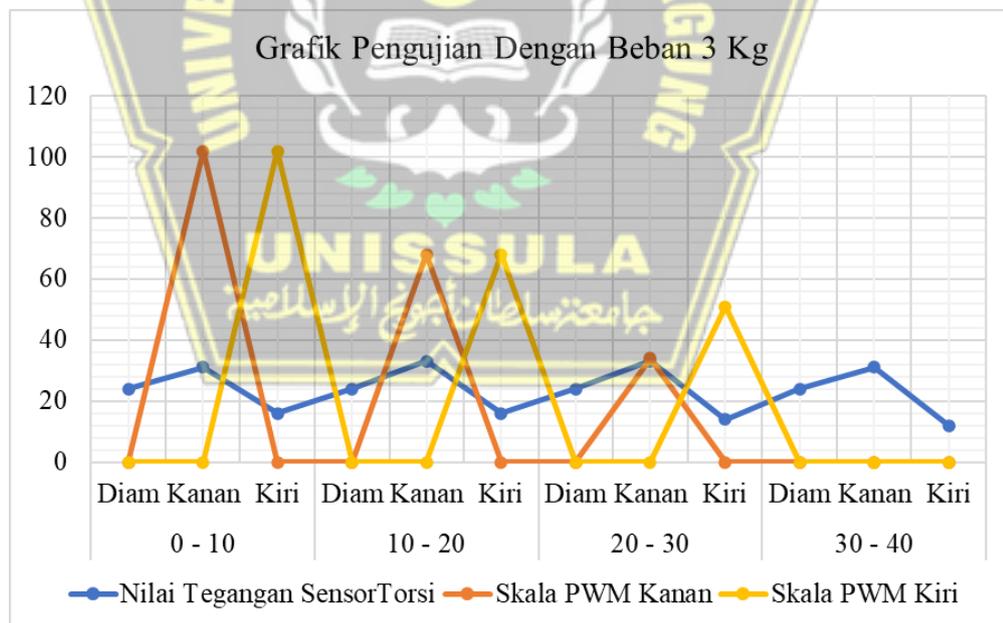
Pada Kecepatan sedang, yaitu 20 sampai 30 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 35 atau 3,5 Volt dan Ketika diputar ke arah kiri nilainya adalah 12 atau 1,2 Volt. Skala PWM pada saat diputar kekanan bernilai 34, saat diputar kekiri bernilai 51 dan 0 ketika posisi *Standby*.

Pada Kecepatan tinggi, yaitu 30 sampai 40 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 33 atau 3,3 Volt dan Ketika diputar ke arah kiri nilainya adalah 14 atau 1,4 Volt. Skala PWM pada saat diputar kekanan, kekiri maupun *standby* sama – sama bernilai 0.

Data hasil percobaan Ketika roda depan diberi beban 3 Kg dapat dilihat pada Tabel 4.6 dan grafik pada Gambar 4.15 berikut.

Tabel 4. 6 Tabel Hasil Pengujian Dengan Beban 3 Kg

Kecepatan (km/jam)	Putaran Setir	Nilai Tegangan Sensor Torsi	Skala PWM Kanan	Skala PWM Kiri
0 - 10	Diam	24	0	0
	Kanan	31	102	0
	Kiri	16	0	102
10 - 20	Diam	24	0	0
	Kanan	33	68	0
	Kiri	16	0	68
20 - 30	Diam	24	0	0
	Kanan	33	34	0
	Kiri	14	0	51
30 - 40	Diam	24	0	0
	Kanan	31	0	0
	Kiri	12	0	0



Gambar 4. 15 Grafik Pengujian Dengan Beban 3 Kg

Dari hasil pengujian dengan beban 3 Kg pada Tabel 4.6 dan Gambar 4.15 didapatkan beberapa hasil. Pada Kecepatan terendah, yaitu 0 sampai 10 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 31 atau 3,1 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 102 dan 0 ketika posisi *Standby*.

Pada Kecepatan rendah, yaitu 10 sampai 20 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 33 atau 3,3 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 68 dan 0 ketika posisi *Standby*.

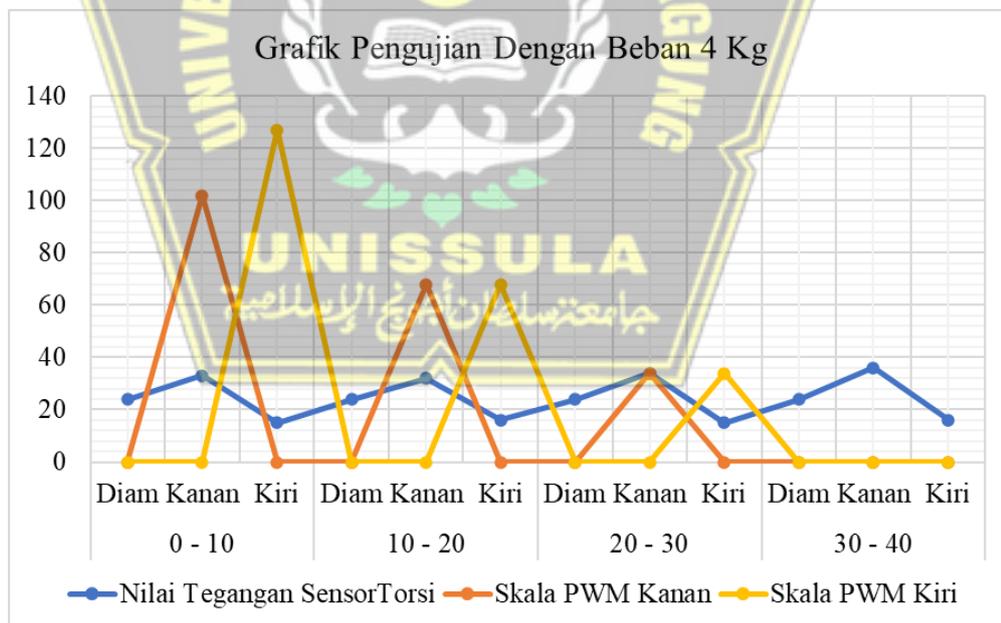
Pada Kecepatan sedang, yaitu 20 sampai 30 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 33 atau 3,3 Volt dan Ketika diputar ke arah kiri nilainya adalah 14 atau 1,4 Volt. Skala PWM pada saat diputar kekanan bernilai 34, saat diputar kekiri bernilai 51 dan 0 ketika posisi *Standby*.

Pada Kecepatan tinggi, yaitu 30 sampai 40 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 31 atau 3,1 Volt dan Ketika diputar ke arah kiri nilainya adalah 12 atau 1,2 Volt. Skala PWM pada saat diputar kekanan, kekiri maupun *standby* sama – sama bernilai 0.

Data hasil percobaan Ketika roda depan diberi beban 4 Kg dapat dilihat pada Tabel 4.7 dan grafik pada Gambar 4.16 berikut.

Tabel 4. 7 Tabel Hasil Pengujian Dengan Beban 4 Kg

Kecepatan (km/jam)	Putaran Setir	Nilai Tegangan Sensor Torsi	Skala PWM Kanan	Skala PWM Kiri
0 - 10	Diam	24	0	0
	Kanan	33	102	0
	Kiri	15	0	127
10 - 20	Diam	24	0	0
	Kanan	32	68	0
	Kiri	16	0	68
20 - 30	Diam	24	0	0
	Kanan	34	34	0
	Kiri	15	0	34
30 - 40	Diam	24	0	0
	Kanan	36	0	0
	Kiri	16	0	0



Gambar 4. 16 Grafik Pengujian Dengan Beban 4 Kg

Dari hasil pengujian dengan beban 4 Kg pada Tabel 4.7 dan Gambar 4.16 didapatkan beberapa hasil. Pada Kecepatan terendah, yaitu 0 sampai 10 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 33 atau 3,3 Volt dan Ketika diputar ke arah kiri nilainya adalah 15 atau 1,5 Volt. Skala PWM pada saat diputar kekanan bernilai 102, saat diputar ke kiri bernilai 127 dan 0 ketika posisi *Standby*.

Pada Kecepatan rendah, yaitu 10 sampai 20 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan maupun ke kiri sama – sama bernilai 68 dan 0 ketika posisi *Standby*.

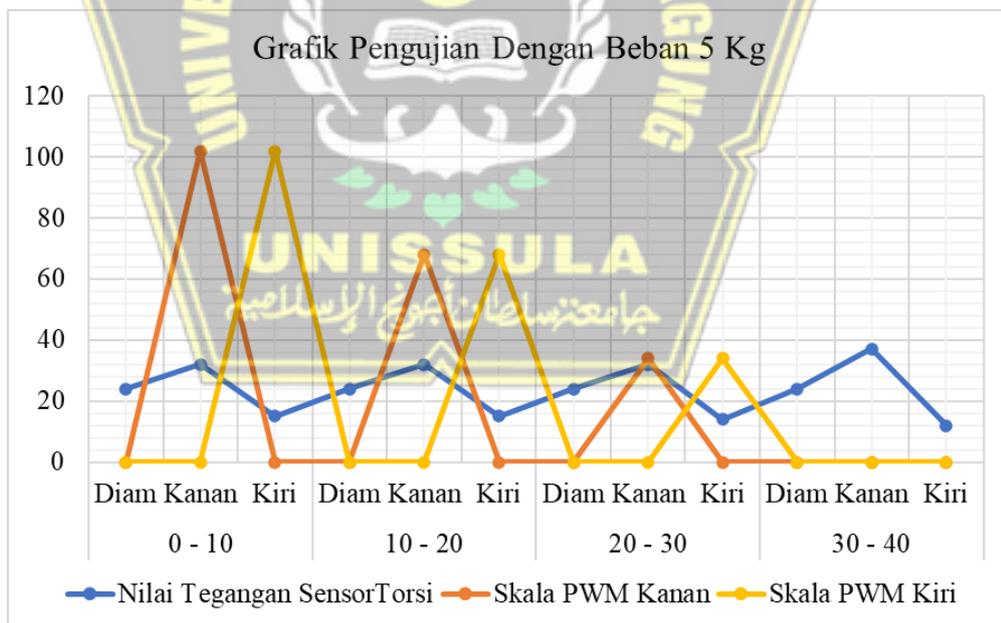
Pada Kecepatan sedang, yaitu 20 sampai 30 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 34 atau 3,4 Volt dan Ketika diputar ke arah kiri nilainya adalah 15 atau 1,5 Volt. Skala PWM pada saat diputar kekanan maupun ke kiri sama – sama bernilai 34 dan 0 ketika posisi *Standby*.

Pada Kecepatan tinggi, yaitu 30 sampai 40 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 36 atau 3,6 Volt dan Ketika diputar ke arah kiri nilainya adalah 16 atau 1,6 Volt. Skala PWM pada saat diputar kekanan, ke kiri maupun *standby* sama – sama bernilai 0.

Data hasil percobaan Ketika roda depan diberi beban 5 Kg dapat dilihat pada Tabel 4.8 dan grafik pada Gambar 4.17 berikut.

Tabel 4. 8 Tabel Hasil Pengujian Dengan Beban 5 Kg

Kecepatan (km/jam)	Putaran Setir	Nilai Tegangan Sensor Torsi	Skala PWM Kanan	Skala PWM Kiri
0 - 10	Diam	24	0	0
	Kanan	32	102	0
	Kiri	15	0	102
10 - 20	Diam	24	0	0
	Kanan	32	68	0
	Kiri	15	0	68
20 - 30	Diam	24	0	0
	Kanan	32	34	0
	Kiri	14	0	34
30 - 40	Diam	24	0	0
	Kanan	37	0	0
	Kiri	12	0	0



Gambar 4. 17 Grafik Pengujian Dengan Beban 5 Kg

Dari hasil pengujian dengan beban 4 Kg pada Tabel 4.7 dan Gambar 4.16 didapatkan beberapa hasil. Pada Kecepatan terendah, yaitu 0 sampai 10 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 15 atau 1,5 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 102 dan 0 ketika posisi *Standby*.

Pada Kecepatan rendah, yaitu 10 sampai 20 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 15 atau 1,5 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 68 dan 0 ketika posisi *Standby*.

Pada Kecepatan sedang, yaitu 20 sampai 30 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 32 atau 3,2 Volt dan Ketika diputar ke arah kiri nilainya adalah 14 atau 1,4 Volt. Skala PWM pada saat diputar kekanan maupun kekiri sama – sama bernilai 34 dan 0 ketika posisi *Standby*.

Pada Kecepatan tinggi, yaitu 30 sampai 40 km/jam menunjukkan nilai tegangan pada sensor torsi bernilai 24 atau 2,4 Volt Ketika sensor torsi pada batang setir tidak mendeteksi masukan atau *standby*, kemudian Ketika diputar ke arah kanan nilainya adalah 37 atau 3,7 Volt dan Ketika diputar ke arah kiri nilainya adalah 12 atau 1,2 Volt. Skala PWM pada saat diputar kekanan, kekiri maupun *standby* sama – sama bernilai 0.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan beberapa percobaan yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut.

1. Pembuatan simulasi sistem kendali kecepatan motor dengan potensiometer dapat dilakukan untuk melihat gambaran awal dari sistem yang sebenarnya.
2. Dari percobaan penentuan nilai PWM berdasarkan masukan dari *rotary encoder* dapat diperoleh hasil bahwa nilai putaran setir kea rah kiri dan kanan memiliki nilai yang berubah – ubah tetapi tidak signifikan.
3. Dari percobaan sistem kendali kecepatan motor EPS berdasarkan dari nilai kecepatan kendaraan dan besaran torsi yang masuk dengan memberi beban pada roda depan, diperoleh hasil bahwa penambahan beban roda depan mobil dapat mempengaruhi besaran torsi yang masuk, akan tetapi perubahannya tidak signifikan.

5.2 Saran

1. Pembuatan simulasi sistem kendali dapat juga dilakukan dengan menggunakan *software* simulasi seperti proteus.
2. Program kontrol PID dapat pula digunakan untuk membuat sistem kemudi kendaraan autonomus dengan menambahkan menambahkan beberapa sensor yang dapat membantu melihat keadaan sekitar kendaraan.

DAFTAR PUSTAKA

- [1] D. I. Fajar, *Analisa Sistem Kemudi Mobil Listrik Brajawahana Its Terhadap Konsdisi*. 2015.
- [2] Anonim, “POWER STEERING, WHEEL ALIGNMENT & BALANCE,” 2012.
- [3] L. Kusuma Pahlawan and A. Maulida Dwi H., “PERMODELAN SKALA ASSIST ELECTRIC POWER STEERING BERBASIS SENSOR SUDUT KEMUDI DAN KECEPATAN PADA KENDARAAN KANCIL,” 2018.
- [4] IndraKeren, “Prinsip dan Cara Kerja Power Steering Elektrik | PT. Toyota Astra Motor | Mobil Terbaik Keluarga Indonesia,” May 21, 2020. <https://www.toyota.astra.co.id/toyota-connect/news/prinsip-dan-cara-kerja-power-steering-elektrik> (accessed Jul. 11, 2022).
- [5] K. Fathoni and F. Suni, “Perancangan Kendali Kecepatan Motor Arus Searah Menggunakan Metode Root Locus,” *Jurnal Teknik Elektro Unnes*, vol. 8, no. 2, pp. 39–42, 2016, doi: 10.15294/jte.v8i2.8793.
- [6] U. Rahardja, A. Saefullah, and M. Ramdani, “Simulasi Sistem Kendali Kecepatan Mobil Secara Otomatis,” vol. 2, no. 40, pp. 130–141, 2009.
- [7] Anthoinete P.Y. Waroh, “Analisa dan simulasi sistem pengendali motor dc,” *Jurnal Ilmiah Sains*, vol. 14, no. 2, p. 80, 2014.
- [8] E. Sunarno and M. H. Purnomo, “Pengaturan Kecepatan Motor Induksi Tanpa Sensor Kecepatan dengan Metoda,” *Jurnal Teknik Elektro*, 2016.

- [9] IndraKeren, “4 Komponen Power Steering Elektrik Yang Punya Potensi Rusak | PT. Toyota Astra Motor | Mobil Terbaik Keluarga Indonesia,” May 21, 2020. <https://www.toyota.astra.co.id/toyota-connect/news/4-komponen-power-steering-elektrik-yang-punya-potensi-rusak> (accessed Aug. 16, 2022).
- [10] L. Toyota *et al.*, “Electric Power Steering (EPS), Sistem Kemudi Lebih Responsif”.
- [11] Q. Liu, W. Kong, and T. Li, “The Assist Curve Design for Electric Power Steering System,” 2016.
- [12] Admin, “2 Types of Torque Sensors in Electronic Power Steering Systems,” 2018. <https://blog.transtec.com/2-types-of-torque-sensors-in-electronic-power-steering-systems> (accessed Jan. 05, 2023).
- [13] Admin, “UNO R3 | Arduino Documentation | Arduino Documentation.” <https://docs.arduino.cc/hardware/uno-rev3> (accessed Jul. 12, 2022).
- [14] M. A. Sulaiman, D. Pembimbing, D. Teknik, S. Perkapalan, and F. T. Kelautan, “Skripsi - me 141501,” 2018.
- [15] Admin, “DC Gear motor dengan encoder 45mm peredam Planet bermotor torsi Besar PG45|motor sauna|motor finmotor 220v - AliExpress.” <https://id.aliexpress.com/item/32727631977.html> (accessed Aug. 16, 2022).
- [16] S. Muttaqin, “Analisa Karakteristik Generator dan Motor DC,” *syaqi*, no. 21060112130034, pp. 1–11, 2013.
- [17] Admin, “Yuk, Cari Tahu Apa Itu Torsi dan Bedanya dengan Tenaga!,” Jun. 15, 2022. <https://www.hyundai.com/id/id/hyundai-story/articles/yuk,-cari-tahu-apa-itu-torsi-dan-bedanya-dengan-tenaga!-0000000104#none> (accessed Jan. 05, 2023).

- [18] Admin, "Arduino Reference - Arduino Reference."
<https://www.arduino.cc/reference/en/> (accessed Jan. 10, 2023).
- [19] A. A. Z. A'laa, "Perancangan Sistem Three Element Control Pada Steam Drum Menggunakan Controller PID Berbasis Neural Network di PG. Modjopangoong," 2021.
- [20] B. Dyson, "How it works: Electric Power Steering (EPS) - carsales.com.au," Aug. 26, 2016.
<https://www.carsales.com.au/editorial/details/how-it-works-electric-power-steering-eps-103538/> (accessed Aug. 30, 2022).

