

**RANCANG BANGUN KENDALI DAN *MONITORING* PADA
KECEPATAN MOTOR BLDC DENGAN SENSOR HALL**

SKRIPSI

Diajukan sebagai salah satu syarat
untuk memperoleh gelar
Sarjana Teknik



Oleh:

MUSYAFFA' AHMAD

I0716026

PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS SEBELAS MARET
SURAKARTA
2020

HALAMAN PENUGASAN

SURAT PERNYATAAN INTEGRITAS PENULIS

Yang bertanda tangan dibawah ini, saya mahasiswa Program Studi Sarjana Teknik Elektro Universitas Sebelas Maret atas nama:

Nama : Musyaffa' Ahmad
NIM : I0716026
Judul Tugas Akhir : Rancang Bangun Kendali dan *Monitoring* Pada Kecepatan Motor BLDC Dengan Sensor Hall

Dengan ini menyatakan bahwa tugas akhir yang saya susun tidak mencontoh atau melakukan plagiat dari karya tulis orang lain. Jika terbukti tugas akhir yang saya susun tersebut merupakan hasil plagiat dari karya orang lain maka tugas akhir yang saya susun tersebut dinyatakan batal dan gelar sarjana yang saya peroleh dengan sendirinya dibatalkan atau dicabut.

Demikian surat pernyataan ini saya buat dengan sebenarnya dan apabila dikemudian hari terbukti melakukan kebohongan maka saya sanggup menanggung segala konsekuensinya.

Surakarta, 18 Juli 2020

Musyaffa' Ahmad

NIM. I0716026

**RANCANG BANGUN KENDALI DAN *MONITORING* PADA
KECEPATAN MOTOR BLDC DENGAN SENSOR HALL**

Disusun oleh

MUSYAFFA' AHMAD

NIM. I0716026

Dosen Pembimbing I

Dr.Ir.Augustinus Sujono, M.T.

NIP. 1951100120161001

Dosen Pembimbing II

Hari Maghfiroh, S.T., M.Eng

NIP. 199104132018031001

Telah dipertahankan di hadapan Tim Dosen Penguji pada hari, tanggal 2020

1. Dr.Ir.Augustinus Sujono, M.T

NIP. 1951100120161001
.....

2. Hari Maghfiroh, S.T., M.Eng

NIP. 199104132018031001
.....

3. Ir. Chico Hermanu B. A., S.T., M.Eng

NIP. 198804162015041002
.....

4. Joko Slamet Saputro S.Pd., M.T

NIP. 196801161999031001
.....

Mengetahui,

Kepala Prodi Teknik Elektro

Koordinator Tugas Akhir

Feri Adriyanto, Ph.D.

NIP. 196801161999031001

M. Hamka Ibrahim, S.T., M.Eng

NIP. 19680116199903100

RANCANG BANGUN KENDALI DAN *MONITORING* PADA KECEPATAN MOTOR BLDC DENGAN SENSOR HALL

Musyaffa' Ahmad¹⁾

¹Program Studi Teknik Elektro, Fakultas Teknik, Universitas Sebelas Maret

Email : 26.musyaffa@gmail.com

ABSTRAK

Dewasa ini, kendaraan berbahan bakar fosil telah banyak menimbulkan masalah lingkungan, sehingga kedepannya kendaraan listrik akan menjadi pengganti utama sebagai alat transportasi, terutama untuk transportasi ringan jarak dekat. Dengan demikian pengembangan kendaraan sepeda listrik sangat relevan untuk dikembangkan. Namun pada kendaraan listrik di pasaran, sistem kendali yang digunakan kurang optimal untuk merespon beban yang diberikan pada kendaraan listrik, sehingga mengakibatkan terjadinya penurunan performa dan pemborosan energi listrik. Dalam skripsi ini dikembangkan kontrol motor listrik BLDC yang memberikan unjuk kerja yang optimal, efisien dan aman. Sistem kontrol yang digunakan adalah sistem NIGD (*Non Isolated Gate Driver*) yang menggunakan driver IC IR2104 dan power switching MOSFET IRFZ44N. Sebagai prototipe, digunakan Motor BLDC 9,6 W, dilengkapi dengan sensor hall guna deteksi kecepatan putar yang dikendalikan dengan mikrokontroler Arduino dan dimonitor dengan *Blynk*. Hasil yang dicapai adalah bahwa sistem ini mampu mengendalikan putaran motor dan menstabilkan saat terjadi gangguan perubahan beban, rata rata dalam 30 detik. Efisiensi sistem tanpa beban mencapai 88,45 % dan pada kondisi pembebanan sangat berat mencapai 82,84 %.

Kata Kunci : Motor BLDC, Kendaraan Listrik, Efisiensi.

**DEVELOPMENT OF CONTROL AND MONITORING SYSTEM IN BLDC
MOTOR SPEED WITH HALL SENSOR**

Musyaffa' Ahmad¹⁾

¹Program Studi Teknik Elektro, Fakultas Teknik, Universitas Sebelas Maret
Email : 26.musyaffa@gmail.com

ABSTRACT

Nowadays, fossil-fueled vehicles have caused many environmental problems, so that in the future electric vehicles will become the main substitute as a means of transportation, especially for short-distance light transportation. Thus the development of electric bicycle vehicles is very relevant to be developed. However, on electric vehicles on the market, the control system used is less than optimal to respond to the load given to electric vehicles, resulting in a decrease in performance and waste of electrical energy. In this thesis BLDC electric motor control is developed which provides optimal, efficient and safe performance. The control system used is the NIGD (Non Isolated Gate Driver) system that uses the IR2104 IC driver and IRFZ44N MOSFET switching power. As a prototype, a 9.6 W BLDC motor was used, equipped with a hall sensor for detection of rotational speed controlled by an Arduino microcontroller and monitored by Blynk. The results achieved are that the system is able to control the rotation of the motor and stabilize when there is a change in load disturbance, averaged in 30 seconds. The efficiency of the no-load system reaches 88.45% and in very heavy loading conditions it reaches 82.84%.

Keyword : BLDC Motor, Electric Vehicle, Efficiency.

DAFTAR ISI

HALAMAN PENUGASAN	ii
SURAT PERNYATAAN INTEGRITAS PENULIS	iii
HALAMAN PENGESAHAN.....	iv
ABSTRAK.....	v
DAFTAR ISI.....	vii
DAFTAR SINGKATAN	xiii
DAFTAR NOTASI.....	xiv
KATA PENGANTAR	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	4
1.5 Sistematika Penulisan	5
BAB II KAJIAN PUSTAKA	6
2.1 Penelitian Terdahulu	6
2.2 Sistem Kendali Kecepatan Motor BLDC	9
2.3 Pulse Width Modulation (PWM)	10
2.4 <i>Blynk</i>	11
2.5 <i>Driver Motor BLDC</i>	12
2.5.1 Rancangan Sistem Kendali Kecepatan Motor BLDC	12
2.5.2 MOSFET IRFZ44N.....	14
2.5.3 IC MOSFET <i>Driver</i> IR2104.....	14
2.5.4 Regulator Tegangan	15
2.5.5 Sensor Tegangan (<i>Voltage Divider</i>)	15
2.5.6 Motor Dc Tanpa Sikat (BLDC)	16
2.5.7 Mikrokontroler Arduino	18
2.5.8 <i>Hall Effect Sensor</i>	19
2.5.9 Sensor Arus	19
2.6 Parameter Performa	20
BAB III METODOLOGI PENELITIAN	21
3.1 Alat dan Bahan	21
3.1.1 Instrumen dan komponen:	21
3.1.2 Perangkat kerja:.....	21

3.1.3 Bahan:	21
3.2 Gambaran Alat yang Dibuat	22
3.3 Alur Penelitian.....	23
3.4 Perancangan dan Pembuatan Perangkat Keras (<i>Hardware</i>)	24
3.4.1 Perancangan Pengatur PWM.....	26
3.4.2 Perancangan <i>Driver</i> Mosfet 3 Fasa.....	26
3.4.3 Perancangan Sensor Arus	27
3.4.1 Perancangan Sensor Tegangan	28
3.5 Perancangan dan Pembuatan Perangkat Lunak (<i>Software</i>)	28
3.5.1 Pembuatan Algoritma Pembacaan Sensor Hall	29
3.5.2 Pembuatan Algoritma Kendali Motor.....	30
3.5.3 Perancangan Aplikasi Monitor dengan <i>Blynk</i>	31
BAB IV HASIL DAN PEMBAHASAN	33
4.1 Hasil Simulasi.....	33
4.1.1 Gelombang keluaran PWM	33
4.1.2 Gelombang keluaran rangkaian	33
4.2 Pembuatan Alat.....	34
4.3 Pengujian Gelombang Keluaran Rangkaian <i>Driver</i>	37
4.3.1 Gelombang PWM.....	38
4.3.2 Gelombang Keluaran Fasa UVW Tanpa Beban	38
4.3.3 Gelombang Keluaran Fasa UVW Pembebanan Ringan.....	40
4.3.4 Gelombang Keluaran Fasa UVW Pembebanan Sedang	40
4.3.5 Gelombang Keluaran Fasa UVW Pembebanan Berat.....	41
4.3.6 Gelombang Keluaran Fasa UVW Pembebanan Sangat Berat	42
4.4 Pengujian Pembacaan nilai PWM	43
4.4.1 Tujuan Utama Pengujian Pembacaan Nilai PWM	43
4.4.2 Hasil Pembacaan Nilai PWM	44
4.5 Pengujian Pembacaan Sensor Arus dan Sensor Tegangan	45
4.5.1 Tujuan Utama Pembacaan Nilai Arus dan Tegangan	45
4.5.2 Hasil Pembacaan Nilai Arus dan Tegangan.....	45
4.6 Pengujian Pembacaan Kecepatan Motor BLDC	47
4.6.1 Tujuan Pengujian Pembacaan Nilai Kecepatan Motor BLDC.....	47
4.6.2 Hasil Pembacaan Kecepatan Motor pada Variasi Potensiometer.....	47
4.6.3 Grafik Regresi Perubahan PWM dan Nilai Kecepatan Motor	48
4.6.4 Pengujian Pelacakan Kecepatan pada Variasi <i>Setpoint</i>	49
4.7 Pengujian Respon Motor BLDC pada <i>Setpoint</i> Kecepatan	49

4.8 Pengujian Variasi Pembebanan pada Motor BLDC	50
4.8.1 Perbandingan Kendali Otomatis & Sistem <i>Open Loop</i> pada Pembebanan	51
4.8.2 Pengujian <i>Driver</i> pada Pembebanan Ringan (10,6 gram).....	52
4.8.3 Pengujian <i>Driver</i> pada Pembebanan Sedang (21,2 gram).....	53
4.8.4 Pengujian <i>Driver</i> pada Pembebanan Berat (31,8 gram).....	54
4.8.5 Pengujian <i>Driver</i> pada Pembebanan Sangat Berat (42,4 gram)	54
4.8.6 Perbandingan Respon Kecepatan pada Variasi Pembebanan.....	55
4.9 Pengujian Monitoring <i>Blynk</i>	56
4.10 Analisa Efisiensi Sistem yang Dirancang	57
4.10.1Perhitungan Kinerja Tanpa Beban.....	58
4.10.2Perhitungan Kinerja Beban Ringan	58
4.10.3Perhitungan Kinerja Beban Sedang	58
4.10.4Perhitungan Kinerja Beban Berat	59
4.10.5Perhitungan Kinerja Beban Sangat Berat	59
BAB V KESIMPULAN DAN SARAN	61
5.1 Kesimpulan.....	61
5.2 Saran.....	61
DAFTAR PUSTAKA	62
LAMPIRAN.....	65

DAFTAR GAMBAR

Gambar 2. 1 Sistem Kendali Open Loop pada motor BLDC	9
Gambar 2. 2 Sistem Kendali Closed Loop pada Motor BLDC.....	10
Gambar 2. 3 PWM [23].....	11
Gambar 2. 4 Aplikasi Blynk.....	11
Gambar 2. 5 Rancangan Sistem Kendali Kecepatan Motor BLDC	13
Gambar 2. 6 IRFZ44N [16].....	14
Gambar 2. 7 Rangkaian MOSFET Driver IR2104 [17].....	15
Gambar 2. 8 Aplikasi sirkuit regulator tegangan.....	15
Gambar 2. 9 Rangkaian Pembagi Tegangan [27]	16
Gambar 2. 10 Konstruksi Motor BLDC [19].....	17
Gambar 2. 11 Motor BLDC In-runner dan Out-runner [20].....	17
Gambar 2. 12 Arduino Nano dan Arduino Uno[7]	18
Gambar 2. 13 Sensor Hall Efek.....	19
Gambar 2. 14 Sensor ACS712 [28].....	20
Gambar 3. 1 Blok Diagram Alat	22
Gambar 3. 2 Alur Penelitian.....	23
Gambar 3. 3 Alur Pembuatan Hardware	25
Gambar 3. 4 Rangkaian Pengatur PWM	26
Gambar 3. 5 Rangkaian Driver Mosfet 3 Fasa	27
Gambar 3. 6 Perancangan Sensor Arus ACS712.....	27
Gambar 3. 7 Rangkaian Sensor Tegangan.....	28
Gambar 3. 8 Diagram Alir pembuatan software.....	29
Gambar 3. 9 Diagram Alir Pembacaan nilai sensor hall.....	30
Gambar 3. 10 Diagram Alir Proses Pengendalian Motor	31
Gambar 3. 11 Algoritma Sistem Monitoring dengan Blynk.....	32
Gambar 4. 1 Grafik Pengujian Simulasi PWM	33
Gambar 4. 2 Grafik Pengujian Simulasi Keluaran Driver	34
Gambar 4. 3 Desain Alat pada Proteus.....	35
Gambar 4. 4 Layout PCB	36
Gambar 4. 5 Desain 3D rangkaian driver	36
Gambar 4. 6 Hardware yang telah dibuat	37
Gambar 4. 7 Gelombang PWM 50%.....	38
Gambar 4. 8 Gelombang keluaran fasa UVW pada kecepatan 5.375 rpm tanpa beban	39
Gambar 4. 9 Gelombang fasa UVW kecepatan 5.375 rpm beban ringan	40

Gambar 4. 10 Gelombang fasa UVW kecepatan 5.375 rpm beban sedang.....	41
Gambar 4. 11 Gelombang fasa UVW kecepatan 5.375 rpm beban berat.....	42
Gambar 4. 12 Gelombang fasa UVW kecepatan 5.375 rpm beban sangat berat.....	43
Gambar 4. 13 Grafik hubungan Potensio dan Nilai PWM	44
Gambar 4. 14 Hubungan nilai PWM dan tegangan 3 fasa	46
Gambar 4. 15 Hubungan nilai PWM dan Arus Input	46
Gambar 4. 16 Grafik Regresi Kecepatan Motor BLDC	48
Gambar 4. 17 Pelacakan kecepatan pada motor BLDC.....	49
Gambar 4. 18 Grafik Transien Respon Time.....	50
Gambar 4. 19 Variasi Pembebanan	51
Gambar 4. 20 Perbandingan respon kecepatan Closed Loop dan sistem Open Loop	52
Gambar 4. 21 Grafik Respon Pembebanan Ringan	53
Gambar 4. 22 Grafik Respon Pembebanan Sedang	53
Gambar 4. 23 Grafik Respon Pembebanan Berat	54
Gambar 4. 24 Grafik Respon Pembebanan Sangat Berat	55
Gambar 4. 25 Respon Kecepatan pada Variasi Pembebanan	55
Gambar 4. 26 Grafik variasi beban vs waktu steady state	56
Gambar 4. 27 Sistem Kendali Motor BLDC dengan Monitoring Blynk	57

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	6
Tabel 2. 2 Perbandingan jenis motor BLDC [11].....	17
Tabel 2. 3 Spesifikasi Mikrokontroler Arduino Nano dan Arduino Uno [24]	18
Tabel 3. 1 Konfigurasi Sensor Hall pada Motor BLDC [11].....	30
Tabel 4. 1 Pengukuran driver tanpa beban	39
Tabel 4. 2 Pengukuran driver beban ringan.....	40
Tabel 4. 3 Pengukuran driver beban sedang.....	41
Tabel 4. 4 Pengukuran driver beban berat	42
Tabel 4. 5 Pengukuran driver beban sangat berat	43
Tabel 4. 6 Respon PWM terhadap perubahan bukaan potensiometer.....	44
Tabel 4. 7 Hasil Pengujian Sensor Arus dan Tegangan.....	45
Tabel 4. 8 Perubahan nilai PWM terhadap kecepatan putar motor.....	48
Tabel 4. 9 Pengujian Driver dengan Beban Ringan.....	53
Tabel 4. 10 Pengujian Driver dengan Beban Sedang	53
Tabel 4. 11 Pengujian Driver dengan Beban Berat.....	54
Tabel 4. 12 Pengujian Driver dengan Beban Sangat Berat.....	55
Tabel 4. 13 Arus dan Tegangan pada Variasi Pembebanan.....	57

DAFTAR SINGKATAN

CD	: <i>Direct Current</i>
BLDC	: <i>Brushless Direct Current</i>
NIGD	: <i>Non Isolated Gate Driver</i>
PWM	: <i>Pulses Width Modulation</i>
VDC	: <i>Volt Direct Current</i>
IC	: <i>Integrated Circuit</i>
RPM	: <i>Revolutions Per Minute</i>
CD ROM	: <i>Compact Disk – Read Only Memory</i>
MOSFET	: <i>Metal Oxide Semiconductor Field Effect Transistor</i>
HO	: <i>High Output</i>
LO	: <i>Low Output</i>
PCB	: <i>Printed Circuit Board</i>
HC	: <i>Hydrocarbon</i>
CO	: <i>Carbon Monoxide</i>
CO ₂	: <i>Carbon Dioxide</i>
O ₂	: <i>Oxygen</i>
Nox	: <i>Nitrogen Oxide</i>
IoT	: <i>Internet of Things</i>

DAFTAR NOTASI

V	: <i>Volt</i>
A	: <i>Ampere</i>
W	: <i>Watt</i>
C	: <i>Celcius</i>
Hz	: <i>Hertz</i>

KATA PENGANTAR

Puji syukur kehadirat Allah *Subhanahu Wa Ta'ala* yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Rancang Bangun Kendali dan *Monitoring* pada Kecepatan Motor DC Tanpa Sikat (BLDC)”. Skripsi ini disusun untuk memenuhi salah satu syarat dalam memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro Universitas Sebelas Maret Surakarta.

Penulis tidak akan sampai pada tahap ini dan skripsi ini tidak akan pernah selesai jika penulis tidak mendapat bantuan dan dukungan dari berbagai pihak. Penulis hendak menyampaikan terima kasih kepada:

1. Bapak Dr.Ir.Augustinus Sujono, M.T. Selaku Pembimbing I dan Bapak Hari Maghfiroh, S.T., M.Eng selaku Pembimbing II yang selalu memberikan pengarahan, ide, saran, ilmu, bimbingan dan motivasi selama perkuliahan di Teknik Elektro sampai pengerjaan skripsi ini.
2. Bapak Ir. Chico Hermanu Brillianto Apribowo, S.T., M.Eng. dan Bapak Joko Slamet Saputro S.Pd., M.T., selaku Dosen Pengaji yang telah memberikan dukungan, ide, dan motivasi selama pengerjaan skripsi ini.
3. Bapak Muhammad Hamka Ibrahim, S.T., M.Eng. selaku Pembimbing Akademik yang selalu memberikan bimbingan dan pemahaman yang baik mengenai perkuliahan.
4. Seluruh Dosen Program Studi Teknik Elektro yang telah memberikan ilmu yang bermanfaat, motivasi, dan inspirasi selama masa perkuliahan penulis.
5. Rekan saya Salman Al Farisi dan Henry Probo Santoso yang membantu memecahkan permasalahan pada saat pengerjaan hardware tugas akhir ini.
6. Teman-teman Teknik Elektro angkatan 2016 yang tidak bisa penulis sebutkan satu persatu, yang telah bersama-sama berjuang dari nol selama 4 tahun.
7. Kakak-kakak angkatan 2014 dan 2015 yang selalu memberi contoh, arahan, dukungan dan *tips* selama penulis berkuliahan di Teknik Elektro serta adik-adik 2017, 2018 dan 2019 yang selalu memberikan dukungan selama perkuliahan.
8. Keluarga tercinta, Bapak Warjo dan Ibu Sun Muslikhah Ahmad yang

selalu sabar memberikan semangat, nasehat serta doa restu sehingga penulis dapat menyelesaikan pengerajan tugas akhir ini.

9. Dan semua pihak yang telah memberikan bantuan, yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa skripsi ini masih belum sempurna. Penulis juga memohon maaf apabila terdapat kesalahan dalam penulisan skripsi ini.

Boyolali, 18 Juli 2020

Penulis

BAB I PENDAHULUAN

1.1 Latar Belakang

Dewasa ini, sistem transportasi menjadi salah satu kebutuhan pokok dalam segala aktivitas manusia. Sistem transportasi terus dituntut untuk meningkatkan kualitas dan keandalannya dalam mendukung mobilitas masyarakat. Sistem transportasi berbasis motor bakar semakin menjamur di masyarakat Indonesia. Berdasarkan data dari Badan Pusat Statistik, pada tahun 2017, jumlah kendaraan bermotor di Indonesia adalah sebanyak 138.556.669 unit [1]. Hal ini tentu saja perlu menjadi perhatian khusus, selain karena peningkatan kemacetan yang terjadi, emisi yang dihasilkan ikut meningkat mengikuti bertambahnya jumlah kendaraan bermotor. Menurut Peraturan Menteri Negara Lingkungan Hidup Republik Indonesia Nomor 10 Tahun 2012, emisi gas buang yang dihasilkan oleh kendaraan bermotor diantaranya adalah hidrokarbon (HC), karbon monoksida (CO), karbon dioksida (CO₂), oksigen (O₂) dan senyawa nitrogen oksida (NO_x) [2]. Hampir seluruh gas buang ini merupakan senyawa yang berbahaya bagi lingkungan.

Untuk mengatasi permasalahan pencemaran lingkungan tersebut, diperlukan suatu alternatif sistem transportasi yang ramah lingkungan, namun tetap handal digunakan sebagai alat transportasi. Sistem transportasi alternatif yang sedang ramai dikembangkan saat ini adalah kendaraan listrik, yang dinilai sangat efektif dan tidak memberikan kontribusi emisi bagi lingkungan. Untuk dapat beroperasi, kendaraan listrik menggunakan sumber energi listrik yang berasal dari baterai dan terhubung dengan sistem kendali elektronik yang mengatur putaran motor listrik. Berbeda dengan kendaraan bermotor yang menggunakan motor bakar, kendaraan listrik menggunakan motor listrik sebagai penggerak. Motor listrik merupakan piranti yang mampu mengubah energi listrik menjadi energi mekanik, dimana pengubahan tersebut umumnya terjadi melalui media elektromagnet [3].

Salah satu jenis motor listrik yang digunakan sebagai penggerak pada kendaraan listrik adalah motor DC tanpa sikat (*Brushless Direct Current / BLDC*). Motor BLDC merupakan motor DC yang mempunyai karakteristik berupa torsi yang besar dan daya yang kecil [4]. Jenis motor DC ini tidak memiliki sikat dan komutator, sehingga motor BLDC hanya membutuhkan sedikit perawatan dan dapat beroperasi lebih tenang dibanding motor DC lainnya [5]. Sifat inilah yang

membuat motor BLDC menjadi sangat populer digunakan sebagai penggerak pada kendaraan listrik. Motor BLDC terbagi menjadi 2 jenis, yaitu *sensored* dan *sensorless*. Perbedaan keduanya terletak pada sensor yang digunakan, dimana BLDC *sensored* menggunakan sensor *Hall effect* untuk memantau posisi rotor. Sedangkan pada jenis BLDC *sensorless*, tidak menggunakan sensor untuk memantau posisi rotor, namun menggunakan deteksi *back EMF* (*Back Electromotive Force*). Pada kendaraan listrik, jenis motor BLDC yang digunakan adalah jenis motor BLDC *sensored*, karena dibutuhkan pembacaan kecepatan yang akurat menggunakan motor BLDC.

Agar dapat beroperasi, Motor BLDC membutuhkan sebuah *driver* untuk dapat menggerakkan dan melakukan pengendalian kecepatan. *Driver* motor umumnya terdiri dari kumpulan komponen elektronika seperti IC, transistor dan dioda. *Driver* motor ini merupakan rangkaian yang memodifikasi tegangan DC menggunakan metode PWM, sehingga menghasilkan tegangan DC 3 fasa berbentuk kotak/trapesium. *Driver* motor BLDC konvensional, kebanyakan menggunakan 6 inverter *switch* untuk pergantian secara elektronik dan sensor hall untuk mendapatkan posisi rotor [6]. Untuk dapat melakukan *switching* pada mosfet, rangkaian *driver* motor BLDC menggunakan IC *driver* mosfet, salah satunya adalah IR2104. IR2104 merupakan IC yang digunakan untuk mengontrol mosfet, dengan cara memberikan input HIGH atau LOW ke mosfet. *Driver* motor ini dapat beroperasi dengan tegangan input antara 0-20 V, dan mampu menghidupkan mosfet yang dialiri tegangan hingga 600 V [7].

Disamping kelebihan yang dimiliki, pada aplikasinya, kendaraan listrik memiliki beberapa kelemahan, yaitu pengembangan teknologinya tidak berkelanjutan, dan masing-masing berdiri sendiri, baik desain maupun konstruksinya, membutuhkan banyak baterai, suplai sistem pengisian belum optimal, kecepatan masih rendah, dan biaya produksi relatif mahal [8]. Selain itu, sistem kendali yang digunakan pada kendaraan listrik di pasaran bersifat *fixed*, atau tidak bisa diprogram ulang. Hal ini menyebabkan sistem kendali tersebut masih kurang optimal, ketika kendaraan listrik diberikan beban yang besar atau kondisi medan yang ekstrem. Pemberian beban dan kondisi medan yang ekstrem pada kendaraan listrik akan mengakibatkan penurunan kecepatan putar pada motor

BLDC sehingga mengakibatkan torsi yang dihasilkan sangat besar. Motor BLDC akan melakukan respon dengan menarik arus yang tinggi dari inverter, untuk mencapai kecepatan *steady state*. Arus yang tinggi ini akan menyebabkan konsumsi energi yang besar ke motor BLDC. Sistem kendali pada kendaraan listrik yang ada di pasaran umumnya memberikan respon yang cukup lama untuk mencapai kecepatan *steady state*, sehingga terjadi pemborosan energi listrik selama waktu untuk mencapai kecepatan *steady state* tersebut.

Melihat permasalahan diatas, maka pada penelitian ini dilakukan rancang bangun untuk menghasilkan sebuah sistem kendali kecepatan motor BLDC pada kendaraan listrik. Adapun metode yang digunakan pada penelitian ini adalah dengan melakukan rancang bangun sebuah *driver* motor BLDC. Sistem kendali yang akan dirancang merupakan sistem kendali *programable*, dimana sistem tersebut nantinya dapat diprogram ulang sesuai dengan keadaan medan dan pembebahan. Perancangan sistem ini menggunakan 2 buah mikrokontroler, yaitu Arduino Nano sebagai otak dari rangkaian kendali kecepatan dan Arduino Uno sebagai penerima *feedback* pembacaan sensor. Rangkaian ini memanfaatkan IC *driver* mosfet IR2104 menjadi NIGD untuk mengatur *switching* high/low pada mosfet. NIGD menerima input HIGH/LOW dari mikrokontroler Arduino yang kemudian diteruskan ke inverter. Pengendalian melalui mikrokontroler Arduino tersebut dilakukan dengan menggunakan prinsip PWM (*Pulse Width Modulation*). Parameter dari kondisi sistem dipantau dengan menggunakan pembacaan nilai kecepatan dan 2 buah sensor sensor, yaitu sensor arus dan sensor tegangan. Data informasi tersebut kemudian diteruskan ke mikrokontroler Arduino sebagai *feedback* untuk diolah dan dikirimkan ke aplikasi *blynk* serta memberikan *feedback* ke Arduino Nano.

Pada sistem ini dilakukan juga pengujian kendali kecepatan otomatis dengan menggunakan eror pembacaan antara kecepatan *setpoint* dari potensiometer dan kecepatan motor sebenarnya, akan diolah untuk ditambahkan ke mikrokontroler sebagai nilai PWM yang dimasukkan ke *driver*. Sehingga ketika terjadi pembebahan, nilai kecepatan dapa dipertahankan , sehingga sistem menjadi lebih handal. Pada sistem ini juga ditambahkan proteksi untuk motor BLDC, dimana tegangan dan arus yang masuk ke motor dibatasi sebesar 15 V dan 1 A. Hal ini

ditujukan untuk menjaga motor agar tidak mengalami kerusakan karena *overvoltage* dan *overcurrent* yang disebabkan pembebanan berlebih pada motor. Ketika nilai batas tersebut telah dicapai, maka arduino akan memproses untuk menghentikan switching pada motor BLDC, sehingga motor BLDC tidak menerima arus dan tegangan dari sumber.

Penggunaan *driver* motor yang dapat diprogram memungkinkan sistem menggunakan algoritma yang lebih handal dan efisien untuk memperoleh hasil penghematan energi listrik yang lebih tinggi. Pembacaan sensor dan informasi seperti tegangan tiap fasa, arus input, kecepatan putar (RPM), dan bukaan potensiometer akan diolah oleh mikrokontroler Arduino Uno yang kemudian akan dikirimkan ke database *blynk* untuk dapat ditampilkan menggunakan aplikasi *blynk* pada smarphone. Dengan sistem ini, diharapkan dapat meningkatkan efisiensi dan performa dari kendaraan listrik sekaligus mengatasi kelemahan yang sering dijumpai pada kendaraan listrik.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan sebelumnya, maka dapat ditentukan rumusan masalah yaitu bagaimana mengembangkan sistem kontrol kecepatan putar motor BLDC dengan PWM *driver* pada Arduino dan sensor hall menggunakan metode kendali otomatis, yang mampu menstabilkan putaran pada perubahan beban dan termonitor dengan *Blynk*?

Sedangkan ruang lingkup permasalahan yang diangkat pada penelitian ini akan diberi batasan pada pengendalian kecepatan pada motor BLDC jenis *Sensored*.

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah dapat ditentukan tujuan dari penelitian ini adalah mengimplementasikan sistem kontrol kecepatan putar otomatis pada motor BLDC dengan PWM driver pada Arduino dan sensor hall yang mampu menstabilkan putaran pada perubahan beban dan termonitor dengan *Blynk*.

1.4 Manfaat Penelitian

Adapun manfaat yang diharapkan dari dilaksanakannya penelitian ini adalah sebagai berikut:

- a. Dihasilkan sebuah drive motor BLDC yang dapat menghasilkan gelombang keluaran DC 3 fasa berbentuk trapesium yang dengan perbedaan fasa 120^0 yang dapat dilakukan pemrograman ulang sesuai dengan keinginan pengguna untuk meningkatkan efisiensi motor BLDC.
- b. Dihasilkan sistem monitoring arus, tegangan, bukaan potensiometer dan kecepatan (rpm) dari sistem kendali yang telah dirancang.
- c. Menjadi bahan rujukan informasi terhadap pengembangan drive motor BLDC.

1.5 Sistematika Penulisan

Susunan dari pembahasan yang direncanakan pada penulisan Skripsi ini adalah sebagai berikut:

BAB I Pendahuluan

Bab ini terdiri dari latar belakang, rumusan masalah, tujuan dan manfaat penelitian, serta sistematika penulisan laporan.

BAB II Tinjauan Pustaka

Bab ini berisi hasil studi penulis dari beberapa literatur tentang teori-teori yang berkaitan dengan topik pembahasan. Pada penelitian ini, fokus utama pembahasan adalah perancangan *driver* motor BLDC

BAB III Metodologi Penelitian

Bab ini mencakup penjelasan tentang metode yang digunakan dalam penelitian beserta tahapan pengerjaan dari perancangan *driver* motor BLDC

BAB IV Hasil dan Pembahasan

Bab ini menyajikan uraian dari data, serta analisis pembahasan mengenai hasil pengujian dari *driver* motor BLDC yang telah dirancang

BAB V Kesimpulan dan Saran

Bab ini berisi tentang hasil akhir penelitian meliputi kesimpulan dari keseluruhan pembahasan yang dilakukan dan saran yang diperuntukan untuk perkembangan yang berhubungan dengan penelitian ini

BAB II KAJIAN PUSTAKA

2.1 Penelitian Terdahulu

Pada tulisan ini, disajikan beberapa hasil penelitian terdahulu yang akan digunakan sebagai perbandingan terhadap penelitian yang dilakukan penulis. Hasil penelitian tersebut dapat diamati pada tabel berikut ini:

Tabel 2. 1 Penelitian Terdahulu

No.	Pengarang	Judul	Metoda	Hasil
1	Mohammad A. [9]	Implementation of a Three Phase Inverter for BLDC Motor Drive	Simulasi menggunakan software PSIM dan Rancang bangun sistem kendali kecepatan motor BLDC menggunakan 3 sensor <i>hall effect</i> berbasis PWM	Hasil dari simulasi dan uji coba hardware sesuai, dimana semua line tegangan dan fasa dalam kondisi seimbang dengan beda fasa 120^0 . Sistem yang diusulkan dapat menghidupkan motor BLDC dan dapat melakukan pengendalian pada kecepatan dengan baik.
2	Ekmen, S [10]	A BLDC Motor Drive with Four Switch Three Phase Inverter	Simulasi menggunakan software MATLAB/Simulink dan rancang bangun drive motor BLDC menggunakan 4 saklar (MOSFET).	Pengujian hardware motor drive dengan sistem yang diusulkan, dapat berjalan baik dengan pengaturan <i>duty ratio</i> menggunakan kontroler PI. Hasil pengujian hardware dapat membuktikan hasil dari simulasi melalui Simulink
3	Aspalli, M.S. [4]	Speed control of BLDC Motor with Four Switch Three Phase Inverter using Digital Signal Controller	Simulasi menggunakan software MATLAB/Simulink dan rancang bangun sistem kendali motor BLDC menggunakan inverter 3 fasa 4 saklar dan kendali sinyal digital yang rendah biaya.	Hardware yang dirancang dapat beroperasi dengan baik, dimana nilai hasil pengujian mendekati nilai hasil simulasi. Penggunaan 4 saklar pada inverter dapat mengurangi biaya yang dibutuhkan tanpa menurunkan performa sistem secara signifikan.

No.	Pengarang	Judul	Metoda	Hasil
4	Ebadpour, M. [5]	Modeling and synchronize d control of dual parallel brushless direct current motors with single inverter	Simulasi sistem kendali 2 motor BLDC dengan satu buah inverter 3 fasa, menggunakan software MATLAB/Simulink.	Sistem yang diusulkan dapat beroperasi dengan baik dan mengurangi berat, volume dan biaya dari sistem kendali motor BLDC pada kendaraan listrik. Sistem ini memiliki respon terhadap perubahan torsi yang baik, stabilitas dan sinkronisasi terhadap kondisi jalan yang baik, dan meningkatkan performa pada variasi medan.
5	Karnap, S. [11]	Developmen t of a Motor Controller for Electric Bicycles	Simulasi menggunakan software Proteus dan rancang bangun 2 jenis sistem drive motor BLDC, <i>sensored</i> dan <i>sensorless</i> (optocoupler) menggunakan IC IR2110.	Hardware yang dirancang dapat menghidupkan motor BLDC dan dapat melakukan pengendalian dengan baik. Namun durasi pengoperasian masih terbatas, karena pemilihan spesifikasi MOSFET yang kurang sesuai sehingga mengakibatkan panas berlebih.
6	Chaterine, P.J. [12]	Simulation Analysis of Power Quality Improvemen t in BLDC Motor Drive Using Type III CUK Converter	Desain dan simulasi drive motor BLDC dengan konverter CUK tipe III menggunakan software MATLAB/Simulink untuk meningkatkan kualitas daya dan koreksi PF pada motor BLDC	Hasil simulasi menunjukkan bahwa sistem yang diusulkan, beroperasi dengan baik dan dapat meningkatkan PF pada daya AC dengan kendali kecepatan yang baik menggunakan Logika Fuzzy.
7	Thirunavuk karasu, [13]	Performance Analysis of BLDC Motor Drive for Feed Drives	Desain dan simulasi drive motor BLDC dengan kendali PI menggunakan software MATLAB/Simulink pada beberapa variasi beban di mesin CNC.	Simulasi motor drive BLDC pada mesin CNC dapat bekerja dengan baik, dimana memberikan rentang kecepatan yang tinggi dan puncak <i>overshoot</i> kurang dari 2%.

No.	Pengarang	Judul	Metoda	Hasil
8	Pindoriya, R.M. [14]	FPGA Based Digital Control Technique for BLDC Motor Drive	Simulasi menggunakan software MATLAB/Simulink dan rancang bangun sistem kendali kecepatan pada drive motor BLDC menggunakan <i>Spartan 3AN field programmable gate arrays</i> (FPGA).	Hasil pengujian hardware memberikan nilai yang sesuai dengan hasil simulasi pada MATLAB/Simulink, termasuk respon kecepatan dari motor. Pengendalian menggunakan PWM memungkinkan sistem untuk melakukan pengendalian kecepatan tinggi dengan komputasi yang akurat. Penggunaan FPGA semakin meningkatkan efektivitas pengendalian kecepatan motor.
9	Musyaffa, A.	Rancang Bangun Kendali dan <i>Monitoring</i> pada Kecepatan Motor BLDC dengan sensor Hall	Rancang bangun sistem kendali kecepatan pada motor BLDC menggunakan 12 Mosfet dan <i>monitoring</i> dengan aplikasi <i>Blynk</i> .	Hasil pengujian menunjukkan bahwa <i>driver</i> motor BLDC menggunakan IC IR2104 dan 12 Mosfet IRFZ44N dapat melakukan pengendalian dan mempertahankan kecepatan motor BLDC baik pada kondisi tanpa beban dan pemberian variasi pembebahan pada motorserta juga berhasil dilakukan perancangan sistem monitoring menggunakan aplikasi <i>Blynk</i> .

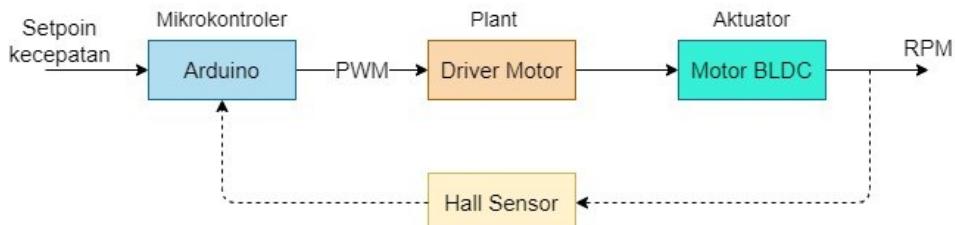
Berdasarkan beberapa penelitian yang telah dilakukan, penulis mencoba merancang sebuah sistem kendali otomatis untuk memperbaiki respon kecepatan terhadap variasi pembebahan yang diberikan ke motor BLDC. Pengujian sistem dilakukan dengan memberikan *setpoint* kecepatan ke motor BLDC, yang kemudian diberikan variasi pembebahan, yang akan menurunkan kecepatan dari motor BLDC. Motor BLDC akan dipacu untuk mengejar *setpoint* yang telah diberikan, dengan menambahkan nilai pwm ke sistem drive motor.

Selain itu juga ditambahkan algoritma proteksi untuk membatasi arus dan tegangan yang masuk ke dalam motor, agar tidak terjadi *overvoltage* maupun

overcurrent yang akan merusak motor BLDC. Nilai tegangan yang dibatasi adalah 12V dan arus sebesar 1 A. Perbedaan yang lain dibandingkan sistem yang sudah ada, adalah pada bagian monitoring, dimana pada sistem ini dihubungkan dengan *Blynk* yang berperan menampilkan kondisi sistem berupa kecepatan, pwm, bukaan potensiometer dan arus pada motor.

2.2 Sistem Kendali Kecepatan Motor BLDC

Salah satu perbedaan dari motor BLDC dan motor DC lainnya adalah pada sistem kendali yang digunakan. Sistem kendali pada motor BLDC cukup rumit dengan menggunakan beberapa komponen elektronika yang disusun untuk dapat melakukan *switching* pada ketiga fasa motor secara bergantian dengan presisi. Sistem kendali tersebut tersusun dari beberapa bagian yang saling terintegrasi untuk menghasilkan kinerja pengendalian yang baik. Salah satu jenis sistem kendali tersebut adalah sistem kendali kecepatan. Pada sistem ini, kecepatan motor BLDC diatur melalui bukaan dari potensiometer / *throttle gas*. Nilai ini kemudian diolah menjadi nilai PWM oleh mikrokontroler untuk dikirimkan ke rangkaian *driver* sebagai pengatur nilai kecepatan motor BLDC.

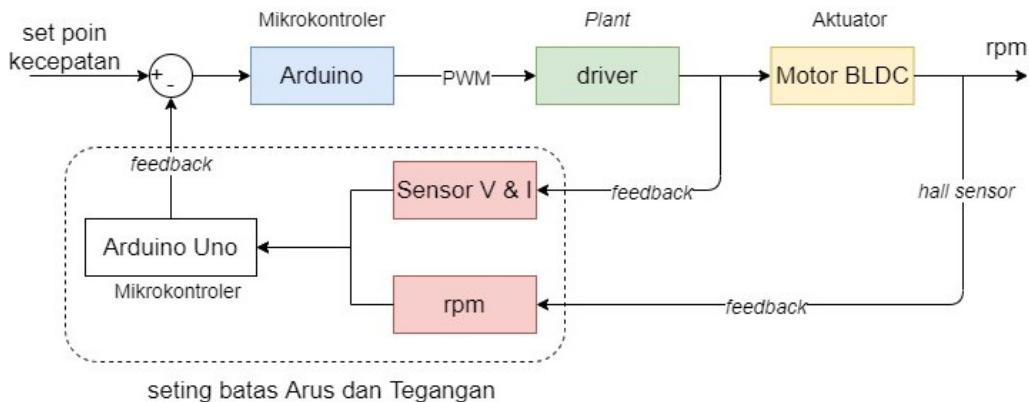


Gambar 2. 1 Sistem Kendali *Open Loop* pada motor BLDC

Terdapat dua jenis sistem pengendalian kecepatan, yaitu sistem *Open Loop* dan sistem *Closed Loop*. Sistem *Open Loop* merupakan sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolan dan tidak ada umpan balik (*feedback*) ke mikrokontroler [30]. Pada jenis sistem kendali ini, nilai kecepatan motor murni diatur oleh potensiometer tanpa ada *feedback* eror ke mikrokontroler untuk diolah. Sistem ini cukup sederhana, dan biasanya diterapkan pada kendaraan listrik. Jenis sistem *Open Loop* memiliki konstruksi yang sederhana dan mudah dipahami, namun memiliki beberapa kelemahan seperti performa yang kurang maksimal dan terjadi penurunan efisiensi pada kondisi medan yang ekstrem atau

pembebatan berlebih. Blok diagram dari sistem *Open Loop* dapat diamati pada gambar 2.1.

Pada jenis *Closed Loop*, sinyal keluarannya mempunyai pengaruh langsung pada aksi pengontrolan yang memiliki *feedback* eror untuk diumpulkan ke kontroler untuk memperkecil kesalahan [30]. Sistem ini biasanya menggunakan metode kendali tertentu untuk meningkatkan akurasi dan presisi dari sistem. Jenis sistem ini memiliki konstruksi yang lebih kompleks dibandingkan jenis *Open Loop*, sehingga perancangan sistem lebih rumit. Namun penggunaan sistem *Closed Loop* ini dapat meningkatkan performa dan efisiensi dari sistem kendali yang dirancang. Pada sistem *Closed Loop* motor BLDC, nilai kecepatan motor merupakan hasil dari pengaturan *setpoint* pada potensiometer ditambah hasil pengolahan *feedback* eror kecepatan yang kemudian dikirimkan ke mikrokontroler. Blok diagram dari sistem kendali *Closed Loop* dapat diamati pada gambar 2.2 berikut:



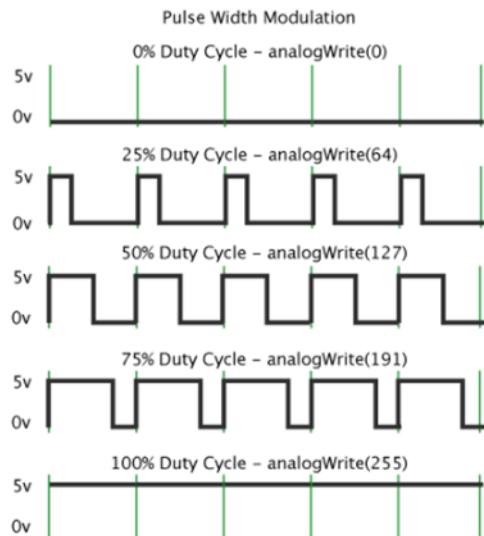
Gambar 2. 2 Sistem Kendali *Closed Loop* pada Motor BLDC

2.3 Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM) merupakan teknik modulasi sinyal dari suatu gelombang dengan memodifikasi lebar pulsa gelombang (*Duty Cycle*) dimana nilai frekuensi dan amplitudo dijaga agar bernilai tetap [15]. *Duty Cycle* (D) sendiri merupakan persentase dari waktu sinyal positif/HIGH (T_{on}) dalam satu periode (T) [7]. Nilai rata-rata dari tegangan yang akan diteruskan ke beban, dikontrol melalui penyaklaran diantara sumber dan beban, dengan pergantian on/off yang cepat. Rentang dari PWM sendiri bernilai antara 0% sampai 100%. Perhitungan nilai *duty cycle* dapat dilakukan dengan menggunakan rumus dibawah ini [7]:

$$D = \frac{T_{on}}{T} \quad (1)$$

PWM digunakan untuk mengeliminasi efek dari lonjakan arus tinggi hingga *throttle* maksimum [11]. Pada penelitian ini, digunakan teknik PWM untuk menghasilkan sinyal modulasi 3 fasa untuk diaplikasikan ke drive motor BLDC, yang tersusun dari rangkaian inverter 3 fasa dan motor BLDC [11]. Nilai PWM dapat diatur menggunakan potensiometer yang terhubung ke pin analog arduino. Nilai pembacaan PWM ini akan mengatur bukaan pada mosfet, sehingga mempengaruhi kecepatan putar motor BLDC. Contoh bentuk gelombang PWM dapat diamati pada gambar 2.5 berikut:



Gambar 2. 3 PWM [23]

2.4 *Blynk*

Blynk adalah perusahaan teknologi yang mengembangkan infrastruktur untuk *Internet of Things* (IoT) [31]. *Blynk* banyak digunakan pada proyek yang membutuhkan monitor melalui IoT pada perangkat smartphone. Aplikasi ini cukup sederhana dan dapat melakukan monitoring maupun pengendalian jarak jauh pada sistem yang diintegrasikan ke *blynk* melalui modul Wifi. Pada penelitian ini, aplikasi *Blynk* digunakan untuk monitoring pada arus, tegangan, kecepatan putar dan bukaan potensiometer. Gambar 2.6 merupakan logo dari aplikasi *Blynk*.



Gambar 2. 4 Aplikasi *Blynk*

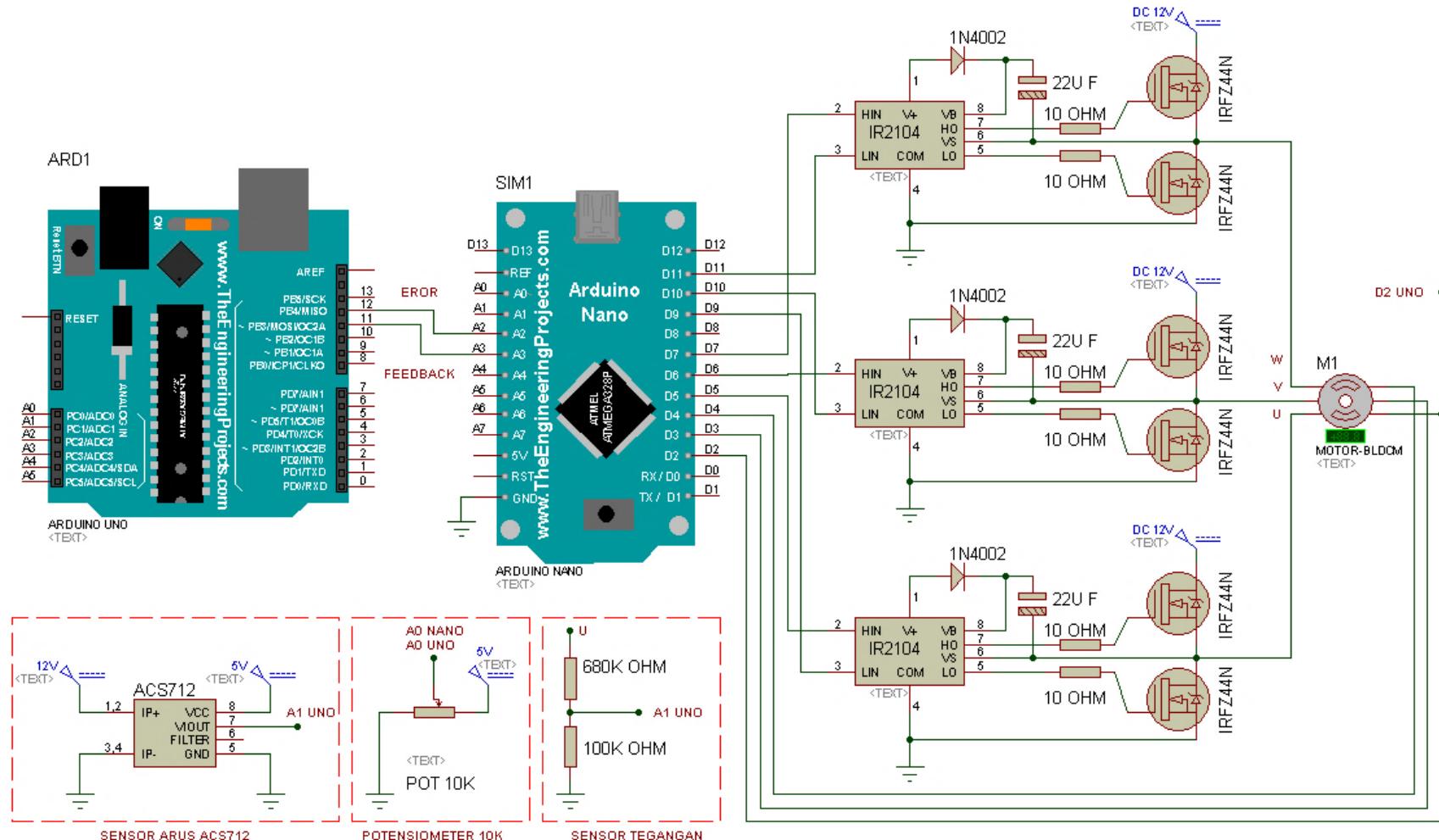
2.5 *Driver* Motor BLDC

Driver motor BLDC merupakan sebuah rangkaian kendali yang tersusun oleh beberapa komponen elektronika yang saling terintegrasi untuk dapat melakukan *switching* pada MOSFET sehingga motor BLDC dapat berputar dengan baik. Rangkaian ini tersusun dari komponen utama yaitu mikrokontroler, IC Driver, Mosfet dan komponen lain. Seluruh komponen tersebut diintegrasikan untuk dapat melakukan *switching* sehingga dapat memutar motor BLDC.

2.5.1 Rancangan Sistem Kendali Kecepatan Motor BLDC

Pada penelitian ini, dirancang sebuah sistem kendali kecepatan otomatis pada motor BLDC seperti dilihat pada gambar 2.7. Pengendalian dilakukan dengan memberikan variasi sinyal PWM ke rangkaian *driver*. Nilai PWM diambil dari nilai bukaan potensiometer yang dikonversi ke nilai PWM. Pada sistem ini diberikan *feedback* ke mikrokontroler Arduino Uno berupa selisih kecepatan yang dihasilkan dengan kecepatan *setpoint* dari potensiometer. Selisih ini kemudian diolah dan dikirimkan melalui pin PWM dari Arduino Uno ke pin Analog Arduino Nano untuk memperbaiki nilai PWM yang masuk ke rangkaian *driver*. Nilai selisih yang diterima Arduino Nano melalui pin Analog kemudian ditambahkan pada nilai PWM yang masuk ke dalam rangkaian *driver*. Dengan metode ini, maka ketika terjadi pembebanan, maka motor akan dapat menjaga kecepatan *setpoint* yang diberikan. Hal ini akan meningkatkan keandalan dan efisiensi dari sistem yang dibuat.

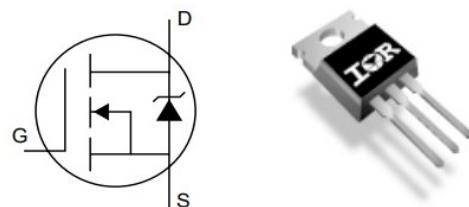
Rangkaian ini menggunakan beberapa komponen utama antara lain IC IR2104 dan MOSFET IRFZ44N. 3 buah rangkaian *driver* MOSFET IR2104 disusun menjadi rangkaian *driver* motor 3 fasa, dengan memanfaatkan 12 MOSFET IRFZ44N untuk melakukan *switching* pada tiap fasa berdasarkan pengolahan informasi dari ketiga sensor hall. Pada bagian monitoring, digunakan modul sensor arus dan rangkaian pembagi tegangan sebagai sensor tegangan yang kemudian dihubungkan ke Arduino Uno untuk dikirimkan ke server *blynk* sehingga dapat ditampilkan pada aplikasi *blynk*.



Gambar 2. 5 Rancangan Sistem Kendali Kecepatan Motor BLDC

2.5.2 MOSFET IRFZ44N

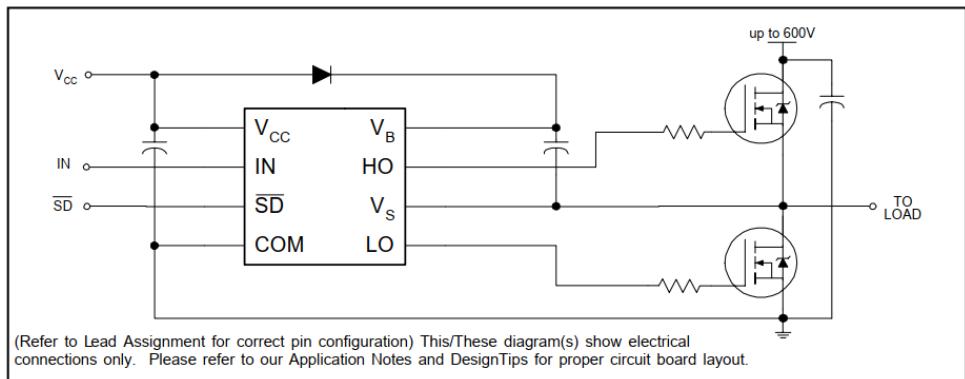
Metal Oxide Semiconductor Field Effect Transistor (MOSFET) merupakan komponen elektronika daya yang berfungsi sebagai saklar elektrik dalam suatu rangkaian elektronik. Selain digunakan sebagai saklar elektrik, MOSFET juga sering digunakan sebagai penguat sinyal pada perangkat elektronik. Pada penelitian ini, digunakan MOSFET tipe-N seri IRFZ44N yang memiliki kelebihan pada kecepatan penyaklaran dan dapat mengalirkan daya yang tinggi. Mosfet ini dapat dialiri arus hingga 49 Ampere, tegangan hingga 55VDC dan memiliki ketahanan suhu hingga 175°C [16]. Pada pengujian ini, digunakan 4 mosfet pada setiap fasa, dimana pada masing-masing keluaran HO dan LO dari IC IR2104 dihubungkan dengan 2 mosfet yang disusun secara paralel. Penyusunan ini dilakukan untuk mengurangi resistansi internal dari masing-masing mosfet, sehingga dapat mengurangi panas akibar arus yang besar pada motor [29]. Konfigurasi dari MOSFET IRFZ44N dapat diamati pada gambar 2.8 berikut:



Gambar 2. 6 IRFZ44N [16]

2.5.3 IC MOSFET Driver IR2104

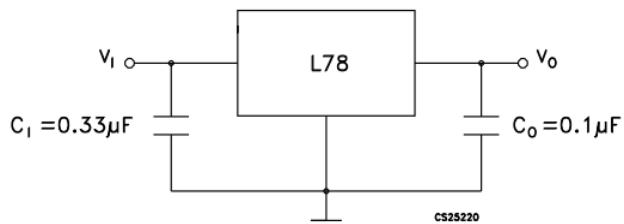
IR2104 merupakan salah satu *driver* MOSFET daya tegangan kecepatan tinggi yang bergantung pada sisi keluaran HIGH dan LOW [17]. IC ini dapat melakukan penyaklaran mosfet yang dialiri tegangan hingga 600VDC. IR2104 memiliki 8 kaki dengan konfigurasi 2 kaki untuk keluaran HIGH dan LOW (HO dan LO), kaki *Shutdown* (SD) , satu kaki input logika (IN), dan kaki Vcc yang membutuhkan suplai tegangan -0,3 hingga 20V [17]. Pada rangkaian drive motor BLDC, IR2104 digunakan untuk menghidupkan motor BLDC melalui penyaklaran MOSFET berdasarkan data HIGH/LOW yang diterima dari mikrokontroler. Konfigurasi koneksi dari IC IR2104 seperti ditampilkan pada gambar 2.7.



Gambar 2. 7 Rangkaian MOSFET Driver IR2104 [17]

2.5.4 Regulator Tegangan

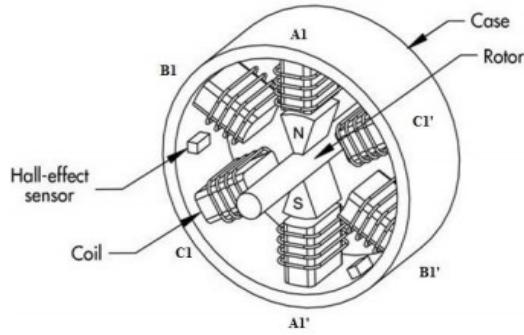
Regulator tegangan merupakan rangkaian elektronika daya yang berfungsi mempertahankan tegangan keluaran agar berada pada nilai tertentu secara otomatis. Rangkaian regulator tegangan umumnya digunakan untuk menurunkan tegangan dari sumber ke level tegangan kerja mikrokontroler [11]. Pada penelitian ini, digunakan rangkaian regulator untuk menurunkan tegangan DC dari tegangan suplai *driver* 12 VDC ke tegangan 5V untuk menyuplai Arduino. Komponen utama pada rangkaian ini yaitu: IC regulator, kapasitor dan resistor. IC regulator yang digunakan adalah tipe LM7805 yang memiliki tegangan keluaran 5V dan arus hingga 1,5A. IC tipe ini dipilih karena memiliki beberapa kelebihan dibanding seri lain, yaitu tidak memerlukan komponen tambahan untuk menyediakan sumber pengaturan konstan, mudah untuk digunakan, serta ekonomis, dan juga menggunakan sirkuit *board* yang efisien dan nyata. Konfigurasi dari regulator tegangan dapat diamati pada gambar 2.8 berikut:



Gambar 2. 8 Aplikasi sirkuit regulator tegangan

2.5.5 Sensor Tegangan (*Voltage Divider*)

Sensor tegangan digunakan untuk membaca nilai tegangan di suatu titik pada sebuah rangkaian elektronika. Sensor tegangan bekerja dengan melakukan

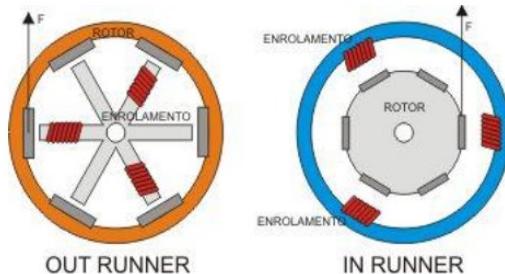


Gambar 2. 10 Konstruksi Motor BLDC [19]

Terdapat 2 jenis motor BLDC, yaitu *In-Runner Motor* dan *Out-Runner Motor* [11]. Seperti dilihat pada gambar 2.11, perbedaan keduanya terletak pada lokasi magent permanen, dimana motor *In-Runner* memiliki permanen magnet yang terletak di dalam terminal elektromagnetik, sedangkan motor *Out-Runner* terletak di bagian luar. Kedua jenis motor BLDC ini memiliki karakteristik masing-masing seperti terlihat pada tabel berikut ini:

Tabel 2. 2 Perbandingan jenis motor BLDC [11].

No	<i>In-Runner Motor</i>	<i>Out-Runner Motor</i>
1	RPM tinggi, torsi rendah	RPM rendah, torsi tinggi
2	Lebih efisien karena kecepatan rotasi tinggi	Efisiensi lebih rendah dari <i>In-Runner</i>
3	Membutuhkan gearbox	Tidak membutuhkan gearbox
4	Agak berisik	Tak ada suara



Gambar 2. 11 Motor BLDC *In-runner* dan *Out-runner* [20]

Motor BLDC banyak digunakan pada berbagai komponen elektronik, terutama pada kendaraan listrik. Hal ini dikarenakan motor BLDC memiliki berbagai kelebihan dibanding motor lain, yaitu *range* torsi lebar, kecepatan tinggi, efisiensi tinggi, respon dinamis yang baik, kuat, tak ada slip, dan lain-lain [18]. Motor BLDC umumnya dikontrol menggunakan inverter 3 fasa, dimana membutuhkan elemen sensor posisi rotor untuk starting dan menyuplai urutan komutator untuk kemudian mengontrol inverter [11].

Pada penelitian ini, digunakan motor BLDC yang diambil dari CD ROM sebagai motor untuk pengujian, yang kemudian dimodifikasi agar pengkabelannya sama dengan motor BLDC pada kenaraan listrik. Spesifikasi dari motor ini adalah 9,6 W dengan tegangan maksimal sebesar 12V.

2.5.7 Mikrokontroler Arduino



Gambar 2. 12 Arduino Nano dan Arduino Uno[7]

Arduino merupakan papan rangkaian elektronik yang bersifat open-source yang dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang dengan komponen utama berupa prosesor jenis AVR dari perusahaan Atmel [21]. Pada penelitian ini digunakan Arduino Uno dan Arduino Nano seperti pada gambar 2.12, yang memiliki spesifikasi sebagai berikut:

Tabel 2. 3 Spesifikasi Mikrokontroler Arduino Nano dan Arduino Uno [24]

Spesifikasi	Arduino Nano	Arduino Uno
Mikrokontroler	Atmega328	Atmega328
Operasi Tegangan	5 V	5 V
Input Tegangan	7-12 V	7-12 V
Input tegangan (limit)	6-20 V	6-20 V
Digital I/O	22 Pin (6 pin output PWM)	14 Pin (6 pin output PWM)
Analog Input	8 Pin	6 Pin
DC Currents per I/O Pin	20 mA	20 mA
DC currents untuk 3.3 v Pin	50 mA	50 mA
Flash Memory	32 KB	32 KB
SRAM	2 KB	2 KB
EEPROM	1 KB	1 KB
Clock Speed	16 MHz	16 MHz
Panjang	45 mm	68.6 mm
Lebar	18 mm	53.4 mm
Massa	7 gram	25 g

Pada penelitian ini, arduino nano digunakan sebagai mikrokontroler untuk mengatur *switching driver* motor dan Arduino Uno sebagai mikrokontroler untuk pembacaan sensor.

2.5.8 Hall Effect Sensor

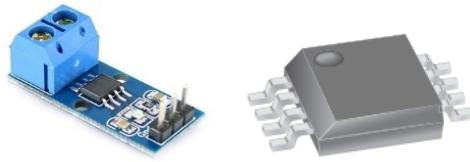
Pada jenis motor BLDC *sensored*, digunakan Sensor Hall Efek untuk mengetahui posisi dari rotor. Sensor Hall Efek seperti pada gambar 2.13 merupakan komponen tranduser yang dapat mengubah suatu bentuk energi magnetik menjadi energi listrik. Sensor hall merupakan jenis sensor medan magnet yang prinsip kerjanya banyak diaplikasikan pada banyak perangkat deteksi, seperti arus, suhu, tekanan dan posisi [25]. Sensor Hall dapat digunakan pada perangkat yang memiliki frekuensi tinggi, hingga 100 kHz [25]. Pada penelitian kali ini, hall sensor digunakan untuk membaca posisi dari rotor pada motor BLDC *sensored*. Informasi pembacaan sensor ini kemudian dikirimkan ke mikrokontroler arduino pada port digital. Nilai pembacaan tersebut adalah HIGH (5V) dan LOW (0V) berperan sebagai feedback yang kemudian akan mengatur *switching* dari mosfet. Tabel konfigurasi sensor hall pada motor BLDC adalah sebagai berikut:



Gambar 2. 13 Sensor Hall Efek

2.5.9 Sensor Arus

Untuk dapat memonitor arus digunakan modul ACS712 sebagai sensor arus yang bekerja berdasarkan prinsip *hall effect*. Modul sensor ini terdiri dari IC ACS712, kapasitor, resistor dan dioda. ACS712 merupakan sensor yang memiliki sedikit *noise*, dengan resistansi konduktor internal sebesar 1,2 mOhm dan membutuhkan suplai tegangan operasi sebesar 5VDC [28]. Modul sensor ini dipilih karena memiliki sensitifitas sebesar 66-185 Mv/a dan memiliki eror pembacaan yang kecil yaitu 1,5% pada suhu 25⁰ C dan 4% pada suhu -40⁰ hingga 85⁰ C. Pada penelitian ini digunakan modul sensor arus ACS712 5 Ampere seperti pada gambar 2.14, untuk membaca arus input dari power suplai.



Gambar 2. 14 Sensor ACS712 [28]

2.6 Parameter Performa

1. Efisiensi

Efisiensi suatu alat merupakan perbandingan antara daya keluaran dengan daya input. Nilai efisiensi biasanya dituliskan dalam bentuk persen. Efisiensi yang baik berkisar diatas 80 atau 90 % [7]. Rumus dari efisiensi adalah sebagai berikut:

$$\eta = \frac{P_{out}}{P_{in}} \times 100\% \quad \dots \dots \dots \quad (3)$$

2. Konsumsi Daya

Daya listrik yang digunakan oleh suatu alat dapat digunakan untuk mengetahui efisiensi dan performa dari alat yang digunakan, serta untuk mengetahui seberapa besar sumber listrik yang dibutuhkan untuk dapat mengoperasikan peralatan tersebut. Rumus dari daya listrik dapat dihitung menggunakan rumus:

$$P = V \times I \quad \dots \dots \dots \quad (4)$$

BAB III METODOLOGI PENELITIAN

Pada penelitian ini dilakukan beberapa tahapan, yang dimulai dengan perancangan alat, perancangan program dan kemudian dilanjutkan dengan pembuatan *hardware*. Setelah hardware selesai dibuat, dilakukan pengecekan apakah rangkaian tersebut dapat berjalan dengan baik dan sesuai dengan rancangan awal. Apabila rangkaian telah selesai dan memberikan kinerja yang baik, maka dilanjutkan dengan pengujian dan pengambilan data yang kemudian akan dilakukan analisa terhadap performa rangkaian.

3.1 Alat dan Bahan

3.1.1 Instrumen dan komponen:

- | | |
|--------------------------------|---------------------------|
| a) Osiloskop | b) Resistor |
| c) Multimeter | d) Kapasitor |
| e) LM7805 | f) Sensor Arus ACS712 20A |
| g) Mosfet <i>Driver</i> IR2104 | h) Dioda |
| i) Mosfet IRFZ44N | j) Arduino Nano |
| k) Esp8266 | l) Potensiometer 10k Ohm |

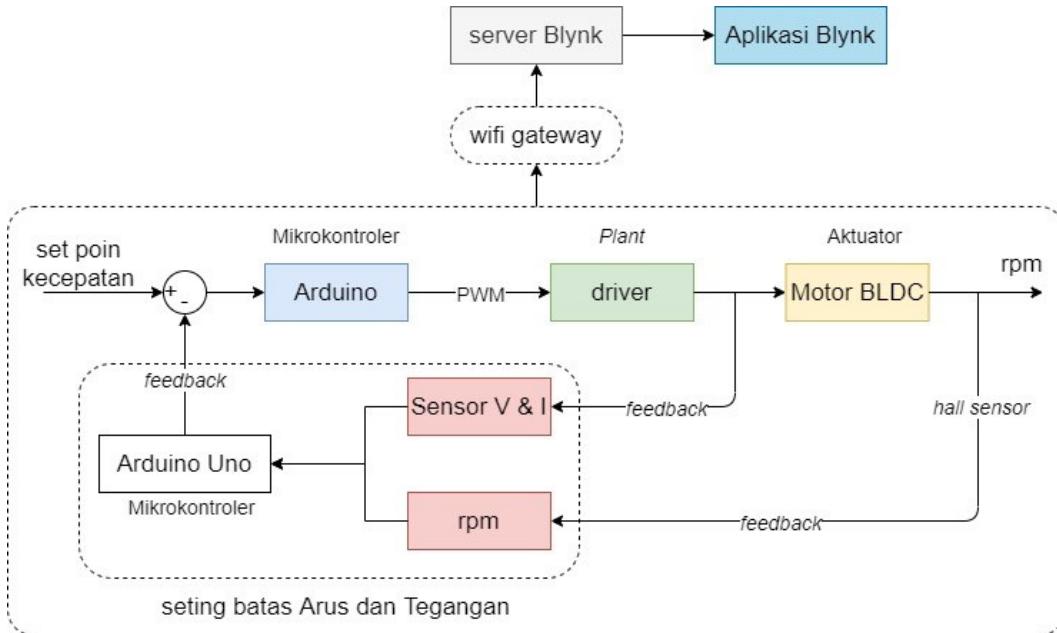
3.1.2 Perangkat kerja:

- | | |
|--------------------------------|------------------------|
| a) Komputer/Laptop | b) Solder |
| c) Perangkat Lunak Proteus | d) Tang Potong |
| e) Perangkat Lunak Arduino IDE | f) Tang Pengupas Kabel |
| g) Kabel penghubung | h) Bor Tangan |

3.1.3 Bahan:

- | |
|-----------------------------|
| a) PCB |
| b) Pelarut PCB (Feriklorit) |
| c) Heatsink |
| d) Timah solder |
| e) Papan Kayu |
| f) Papan Akrilik |

3.2 Gambaran Alat yang Dibuat



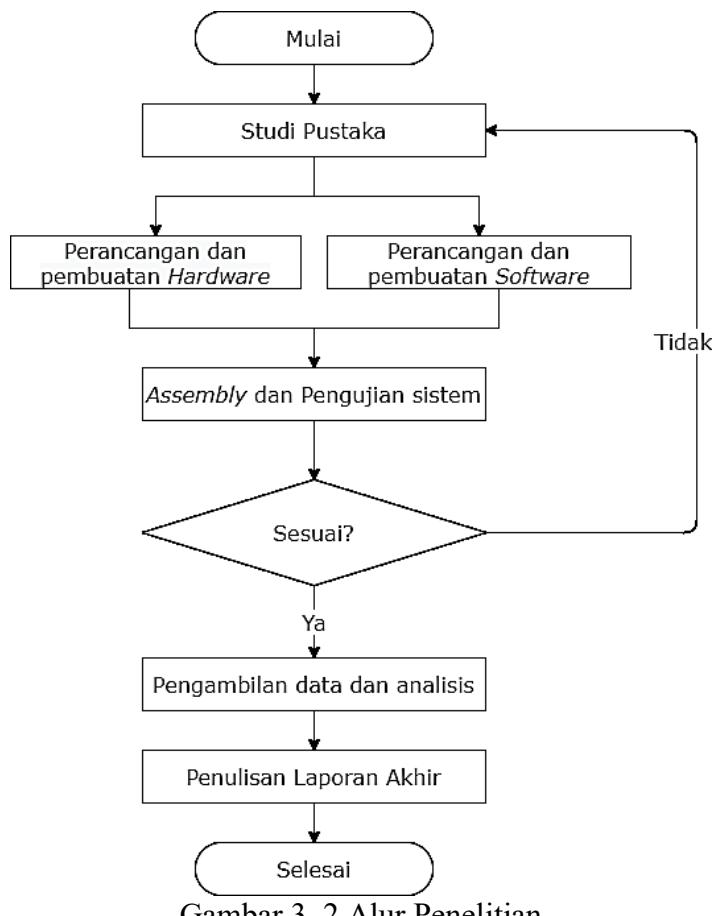
Gambar 3. 1 Blok Diagram Alat

Rancangan alat yang akan dibuat terdiri dari rangkaian kendali dan rangkaian monitoring seperti ditampilkan pada gambar 3.1 diatas. Pada *driver* motor BLDC ini, digunakan 2 mikokontroler yaitu Arduino Uno sebagai sebagai mikrokontroler pembacaan sensor dan Arduino Nano yang digunakan untuk mengatur *switching driver* yang memutar motor BLDC. Arduino Nano bekerja dengan melakukan pembacaan nilai sensor hall dan nilai PWM yang merupakan pengolahan nilai potensiometer ditambah dengan *feedback* nilai eror kecepatan yang diberikan oleh arduino Uno.

Sedangkan Arduino Uno bekerja sebagai monitor dari nilai sensor yang digunakan. Pembacaan yang dilakukan oleh mikrokontroler ini berupa pembacaan nilai sensor arus, sensor tegangan, nilai bukaan potensiometer dan nilai kecepatan (rpm) dari pembacaan sensor hall. Pada metode kendali otomatis, selisih nilai kecepatan antara nilai *setpointt* dari bukaan potensiometer dengan nilai pembacaan kecepatan, akan dijadikan *feedback* ke mikrokontroler Arduino Nano. *Feedback* ini akan melakukan perbaikan kecepatan yang dihasilkan oleh motor BLDC. Hasil pembacaan arus, tegangan, kecepatan dan bukaan potensiometer ini kemudian dikirimkan ke server *Blynk* melalui ESP8266 untuk ditampilkan pada Aplikasi *Blynk*.

3.3 Alur Penelitian

Seperti ditampilkan pada gambar 3.2, penelitian ini diawali dengan melakukan studi pustaka dari beberapa referensi seperti *paper*, laporan penelitian, skripsi, *datasheet*, buku hasil percobaan yang sesua dengan topik yang diusulkan. Setelah diperoleh referensi dan alur kerja alat yang akan dibuat, kemudian dilakukan perancangan sistem dan pembuatan *hardware* dan *software*. Pembuatan *hardware* dilakukan melalui beberapa tahapan, seperti pada gambar 3.3 pada subbab 3.4 dibawah. *Hardware* tersebut kemudian memasuki tahapan pengecekan kinerja pada masing-masing komponen untuk memastikan rangkaian dapat berjalan dengan baik.



Gambar 3. 2 Alur Penelitian

Bersamaan dengan pembuatan *hardware* juga dilakukan pembuatan *software* yang nantinya akan diaplikasikan ke alat yang dibuat. Pembuatan *software* berupa program untuk mikrokontroler arduino yang dibuat dengan menggunakan Arduino IDE. Alur pembuatan *software* dapat diamati pada gambar 3.8.

Setelah rangkaian *hardware* dan *software* selesai dibuat, kemudian dilakukan *assembly* atau penyatuan antara *hardware* dan *software*. *Software* yang dibuat kemudian diupload ke mikrokontroler Arduino Nano. Setelah proses *upload* selesai dilakukan, *driver* yang telah dibuat kemudian diujikan pada motor BLDC. Pada pengujian ini, dilakukan pengecekan apakah terdapat eror pada rangkaian. Apabila *hardware* dapat bekerja dengan baik, maka tahapan selanjutnya adalah pengambilan data pada sistem. Pengambilan data dilakukan melalui Arduino Uno yang digunakan sebagai mikrokontroler pembacaan. Data yang diambil berupa arus, tegangan tiap fasa, kecepatan dalam rpm, nilai pwm dan nilai bukaan pada potensiometer. Data yang telah diambil kemudian dilakukan analisa keandalan dari sistem yang telah dibuat. Hasil dan analisa tersebut kemudian dituliskan pada laporan sesuai dengan data yang diperoleh.

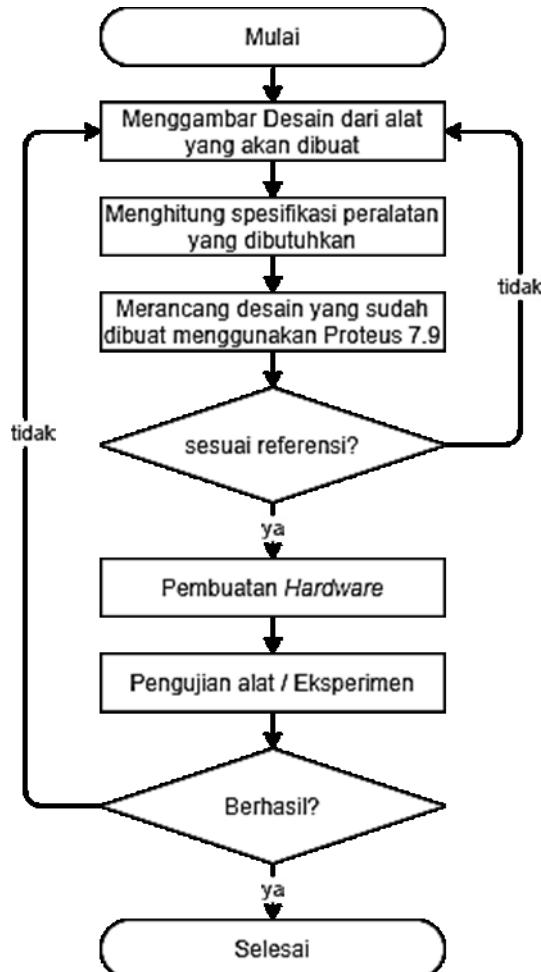
Waktu penelitian dilakukan pada bulan Maret 2020 – Juli 2020 yang mulanya berlokasi di Laboratorium Teknik Elektro Universitas Sebelas Maret Surakarta, dan kemudian dilanjutkan di rumah penulis dikarenakan sedang terjadi pandemi Covid-19.

3.4 Perancangan dan Pembuatan Perangkat Keras (*Hardware*)

Seperti ditampilkan pada gambar 3.3, pembuatan *hardware* dimulai dengan membuat gambaran mengenai alat yang akan dibuat. Gambaran tersebut merupakan rancangan awal berupa rangkaian rangkaian kontrol PWM, *driver* mosfet 3 fasa yang dihubungkan dengan rangkaian sensor dan rangkaian komunikasi ke *blynk* menggunakan ESP 8266. Setelah tahapan desain selesai, dilanjutkan dengan melakukan perhitungan mengenai spesifikasi dan jumlah komponen yang digunakan seperti nilai resistor yang digunakan sebagai rangkaian pembagi tegangan, spesifikasi sensor arus yang digunakan, dan jenis mosfet yang dipakai.

Tahapan berikutnya adalah melakukan perancangan dari alat yang akan dibuat. Perancangan dilakukan dengan menggunakan *software* Proteus 7.9. Perancangan ini dibuat berdasarkan desain dan perhitungan yang telah dilakukan. Pada tahapan perancangan ini, juga dilakukan simulasi untuk mengetahui keluaran gelombang yang dihasilkan pada rangkaian. Apabila hasil keluaran sesuai dengan referensi, maka dibuat desain layout PCB yang kemudian masuk ke tahapan pembuatan *hardware*. Apabila hasil simulasi masih belum memberikan hasil seperti yang

diharapkan, maka perlu mengulangi proses desain alat dan perhitungan spesifikasi yang digunakan.

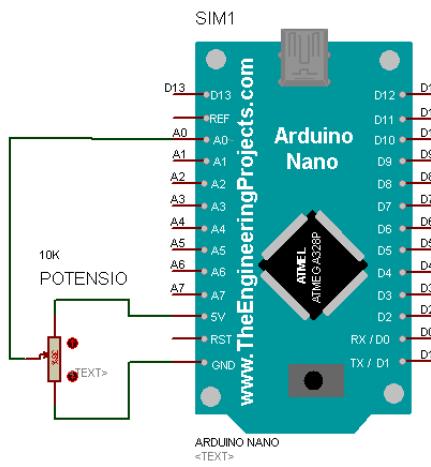


Gambar 3. 3 Alur Pembuatan Hardware

Tahapan terakhir dari proses ini adalah pembuatan *hardware*. Pembuatan *hardware* dimulai dengan memasang semua komponen yang digunakan pada rangkaian ini pada papan PCB. Papan PCB yang digunakan adalah PCB fiber 1 layer. Pemasangan komponen dilakukan menggunakan solder dengan hati-hati agar tidak ada jalur yang terhubung (*short*). Setelah pemasangan komponen selesai, kemudian rangkaian *driver* tersebut dilakukan uji coba untuk mengetahui apakah sistem yang dirancang dapat berjalan dengan baik dan memberikan hasil respon yang sesuai.

3.4.1 Perancangan Pengatur PWM

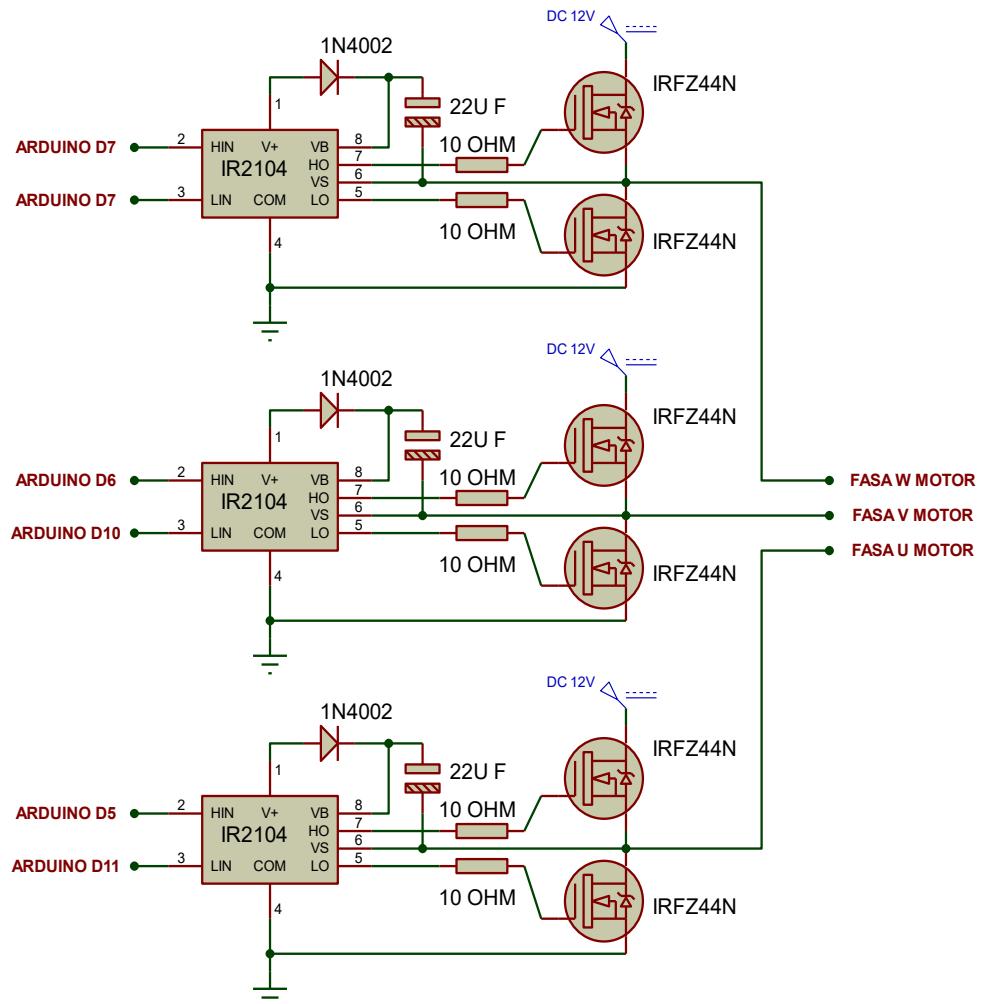
Tahapan ini meliputi perancangan konfigurasi untuk mengatur nilai PWM yang akan mengendalikan kecepatan motor BLDC. Rangkaian ini menggunakan potensiometer 10k Ohm yang dihubungkan dengan Pin Analog 0 dari Arduino Nano. Nilai Putaran potensiometer akan mengatur nilai *duty cycle* yang keluar dari pin digital Arduino ke IC IR2104 pada pin SD. Pengaturan PWM berjalan dengan baik dan mikrokontroler dapat menerima informasi dengan benar. Konfigurasi dari rangkaian ini dapat diamati pada gambar 3.4 dibawah ini:



Gambar 3. 4 Rangkaian Pengatur PWM

3.4.2 Perancangan *Driver* Mosfet 3 Fasa

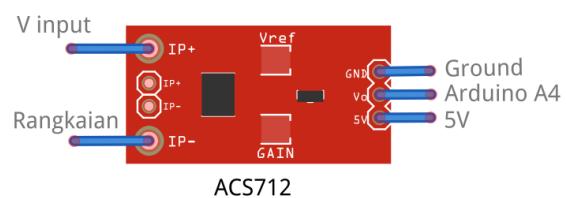
Rangkaian *driver* mosfet 3 fasa seperti gambar 3.5 merupakan rangkaian utama pada penelitian ini, yang menjadi rangkaian kendali motor BLDC. *Driver* mosfet 3 fasa ini terdiri dari 3 buah IC IR2104 yang masing-masing terhubung dengan mosfet pada masing-masing channel keluaran High dan Low. Keluaran dari rangkaian ini adalah 3 kabel U,V dan W yang terhubung ke motor BLDC. IC IR2104 mendapat input logika dari pin digital Arduino Nano, yang kemudian melakukan *switching* pada mosfet sehingga motor BLDC dapat berputar.



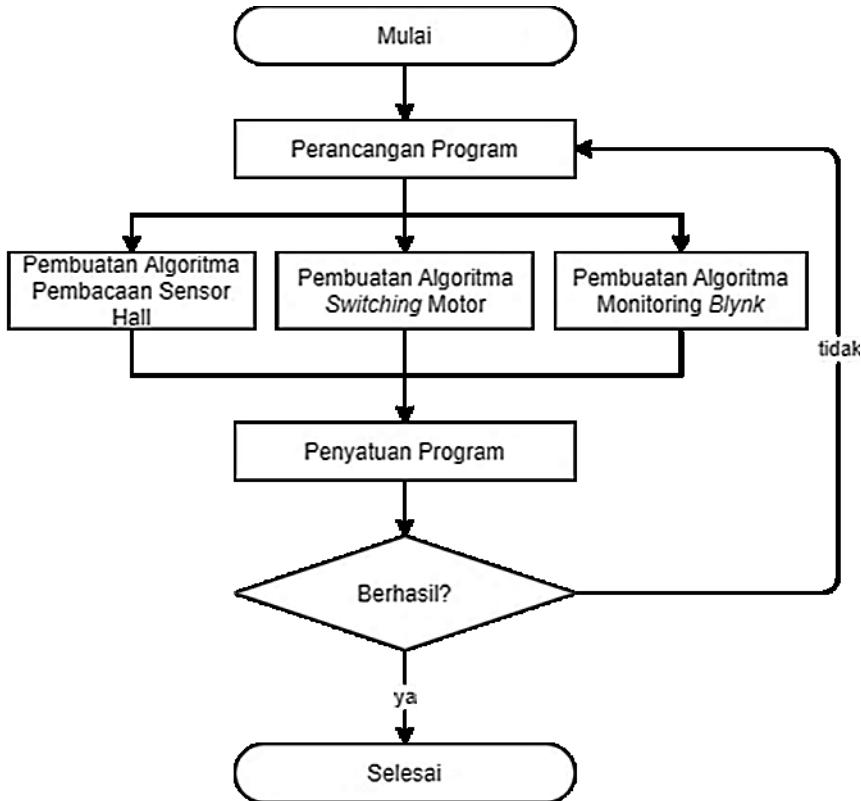
Gambar 3. 5 Rangkaian *Driver Mosfet 3 Fasa*

3.4.3 Perancangan Sensor Arus

Pembacaan nilai arus menggunakan modul sensor ACS712 yang dapat mengukur nilai arus hingga 20 Amper. Modul sensor ini diletakkan di titik suplai rangkaian dan hasil pembacaan sensor dihubungkan ke mikrokontroler port Analog 4 dari Arduino Uno untuk diolah hasil pembacaannya. ACS712 dipilih karena hasil pembacaan yang cukup stabil dan rangkaianya sederhana untuk diaplikasikan. Konfigurasi dari sensor arus ACS712 dapat diamati pada gambar 3.6 dibawah ini:



Gambar 3. 6 Perancangan Sensor Arus ACS712

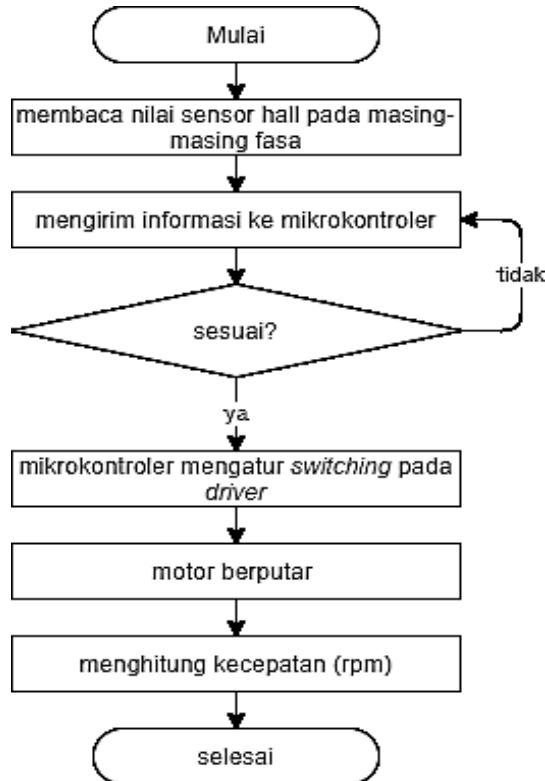


Gambar 3. 8 Diagram Alir pembuatan software

Setelah ketiga program selesai dibuat, tahapan selanjutnya adalah penyatuan program dan upload ke mikrokontroler. Setelah berhasil, kemudian dilakukan pengujian pada *hardware* yang telah dibuat. Apabila *hardware* dapat berjalan dengan baik, maka dilakukan pengujian untuk pengambilan data. Namun jika *hardware* belum bisa berjalan dengan baik, maka perlu dilakukan pengecekan ulang pada rancangan yang telah dibuat.

3.5.1 Pembuatan Algoritma Pembacaan Sensor Hall

Pada tahapan ini, dilakukan pembacaan nilai keluaran sensor hall pada motor BLDC. Sensor ini mengeluarkan nilai 5V ketika hidup dan 0V ketika mati. Informasi ini kemudian diubah ke digital dan menghasilkan nilai HIGH untuk 5V dan LOW untuk 0V. Konfigurasi pembacaan dari 3 sensor hall ini berperan sebagai *feedback* yang kemudian diproses oleh mikrokontroler untuk menghasilkan keputusan terkait pengendalian *switching* pada rangkaian *driver*. Selain digunakan untuk pengaturan *switching*, pembacaan sesnor hall juga digunakan untuk menghitung kecepatan motor BLDC.



Gambar 3. 9 Diagram Alir Pembacaan nilai sensor hall

Untuk dapat melakukan pembacaan nilai sensor hall dan melakukan *switching* dengan benar, maka disusun algoritma pembacaan nilai sensor hall seperti ditunjukkan pada gambar 3.9 diatas, dimana agar dapat melakukan *switching* dengan benar, maka nilai pembacaan harus memenuhi tabel 3.1 berikut ini [11]:

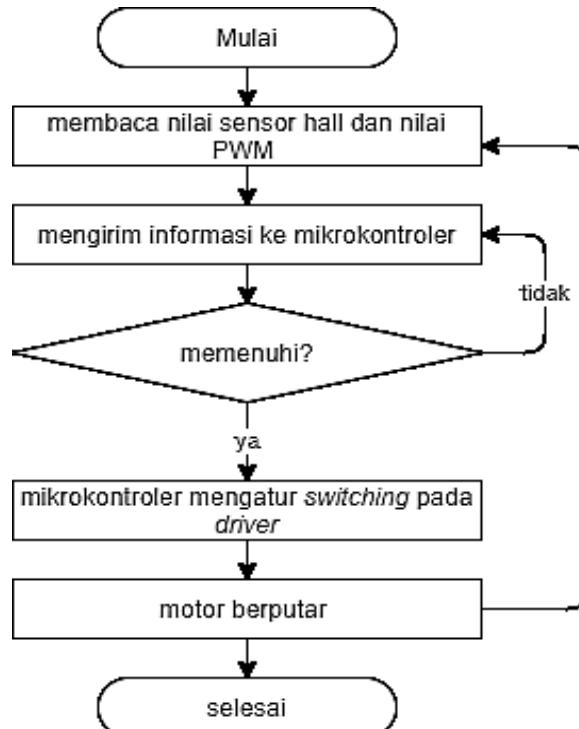
Tabel 3. 1 Konfigurasi Sensor Hall pada Motor BLDC [11]

Step	Hall A	Hall B	Hall C	Code	A (M1)	B (M2)	C (M3)
1	1	0	1	101	+Vm	-Vm	Float
2	1	0	0	100	Float	-Vm	+Vm
3	1	1	0	110	-Vm	Float	+Vm
4	0	1	0	010	-Vm	+Vm	Float
5	0	1	1	011	Float	+Vm	-Vm
6	0	0	1	001	+Vm	Float	-Vm

3.5.2 Pembuatan Algoritma Kendali Motor

Tahapan ini berisi mengenai algoritma pengendalian motor BLDC yang tersusun dari pembacaan nilai PWM yang terhubung ke arduino, kemudian dilakukan pembacaan nilai sensor hall. Apabila kedua persyaratan telah

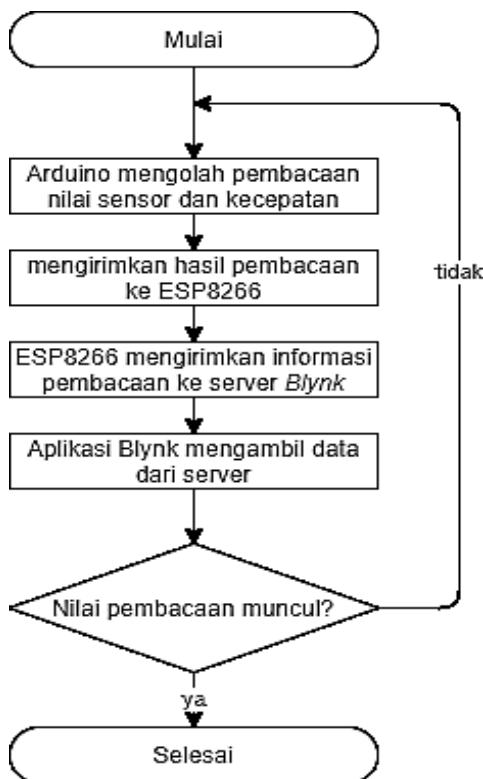
terpenuhi, maka mikrokontroler akan membuat keputusan untuk penyaklaran IC IR2104 yang akan mengatur putaran motor BLDC sesuai dengan tabel konfigurasi. Diagram alir pada proses ini dapat diamati pada gambar 3.10 berikut:



Gambar 3. 10 Diagram Alir Proses Pengendalian Motor

3.5.3 Perancangan Aplikasi Monitor dengan *Blynk*

Gambar 3.11 menunjukkan algoritma dari monitoring pada penelitian ini dilakukan dengan menggunakan aplikasi *blynk* yang dapat diakses melalui smartphone. *Blynk* dipilih karena memiliki sistem yang sederhana dan mudah diakses oleh siapa saja. Sistem monitoring ini menampilkan nilai pembacaan dari sensor tegangan dan sensor arus, nilai kecepatan putar dalam RPM, bukaan potensio dan nilai PWM yang dihasilkan oleh mikrokontroler. Informasi ini kemudian dikirimkan ke *server blynk* melalui internet dan kemudian ditampilkan pada aplikasi *blynk*. Untuk rangkaian ini hanya dikhususkan untuk monitoring saja, karena pengendalian sistem dilakukan pada hardware sendiri.



Gambar 3. 11 Algoritma Sistem Monitoring dengan *Blynk*

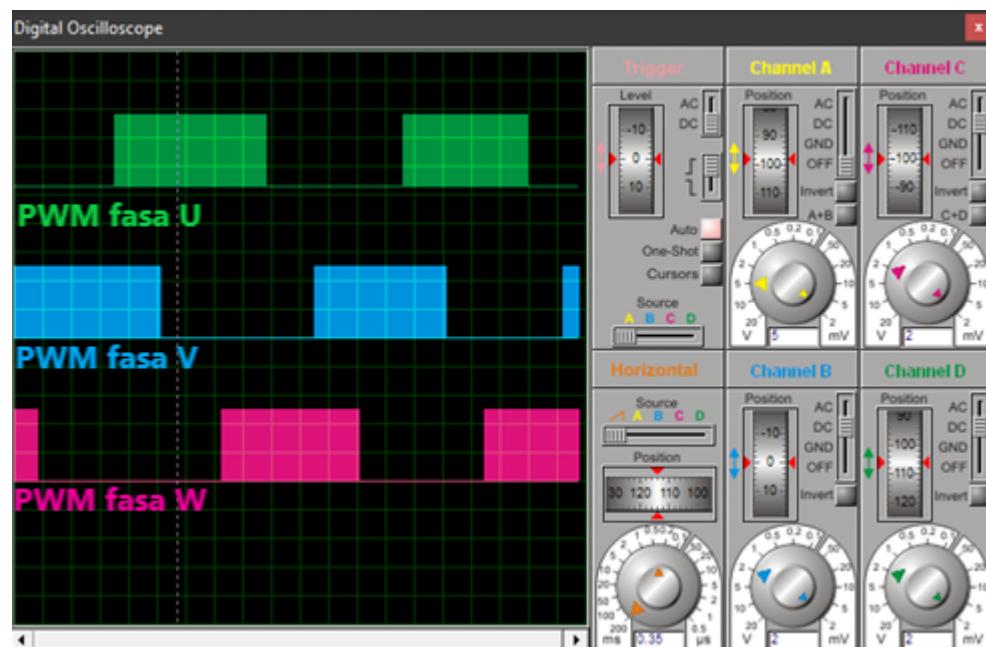
BAB IV HASIL DAN PEMBAHASAN

4.1 Hasil Simulasi

Pengujian simulasi dilakukan dengan menggunakan *software* Proteus. Pengujian yang dilakukan pada tahap ini hanya terbatas pada pengujian pembacaan gelombang PWM dan gelombang keluaran dari rangkaian *driver*, kemudian pengujian tegangan pada fasa dan arus input rangkaian. Pengamatan gelombang keluaran dilakukan untuk mengetahui bentuk gelombang pada masing-masing fasa yang nantinya akan dibandingkan dengan gelombang keluaran dari hardware. Hasil simulasi yang telah dilakukan dapat dilihat pada gambar berikut:

4.1.1 Gelombang keluaran PWM

Hasil simulasi menunjukkan bentuk sinyal keluaran PWM pada mikrokontroler arduino seperti ditampilkan pada gambar 4.1 dibawah ini. Sinyal PWM yang dimatai merupakan sinyal pada ketiga fasa. Hasil menunjukkan bahwa terdapat perbedaan 120° antar masing-masing fasa. Hasil simulasi ini sesuai dengan referensi, dan tidak dijumpai noise pada gelombang hasil pengujian.

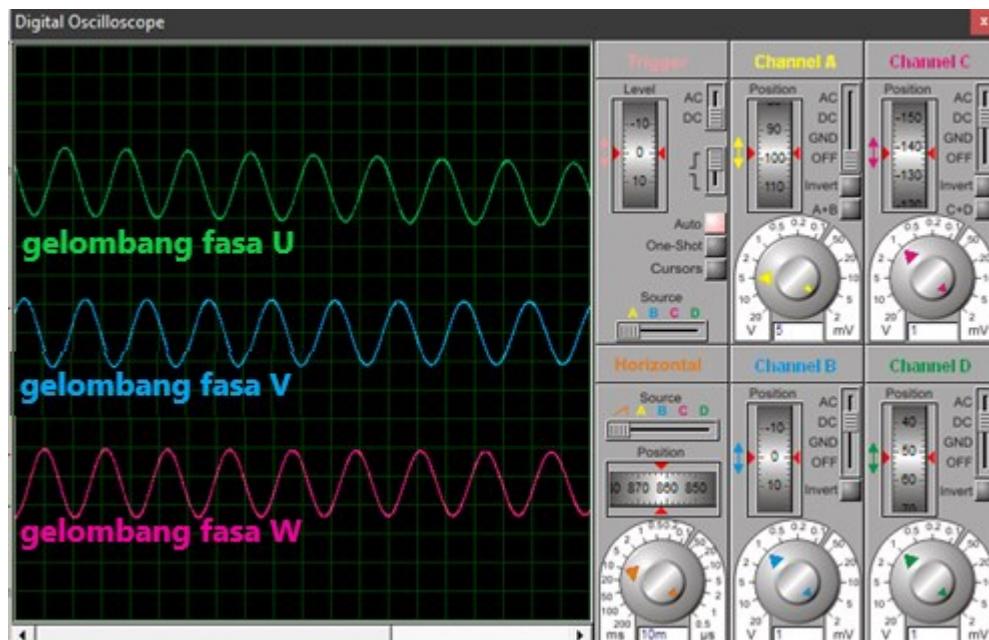


Gambar 4. 1 Grafik Pengujian Simulasi PWM

4.1.2 Gelombang keluaran rangkaian

Pada pengujian ini dilakukan pengamatan gelombang keluaran pada *driver* motor, yaitu hasil keluaran masing-masing fasa yang terhubung ke motor

BLDC. Bentuk gelombang yang dihasilkan cukup sempurna, tanpa ada noise dan memiliki perbedaan fasa sebesar 120^0 . gelombang keluaran pengujian ini dapat diamati pada gambar 4.2.

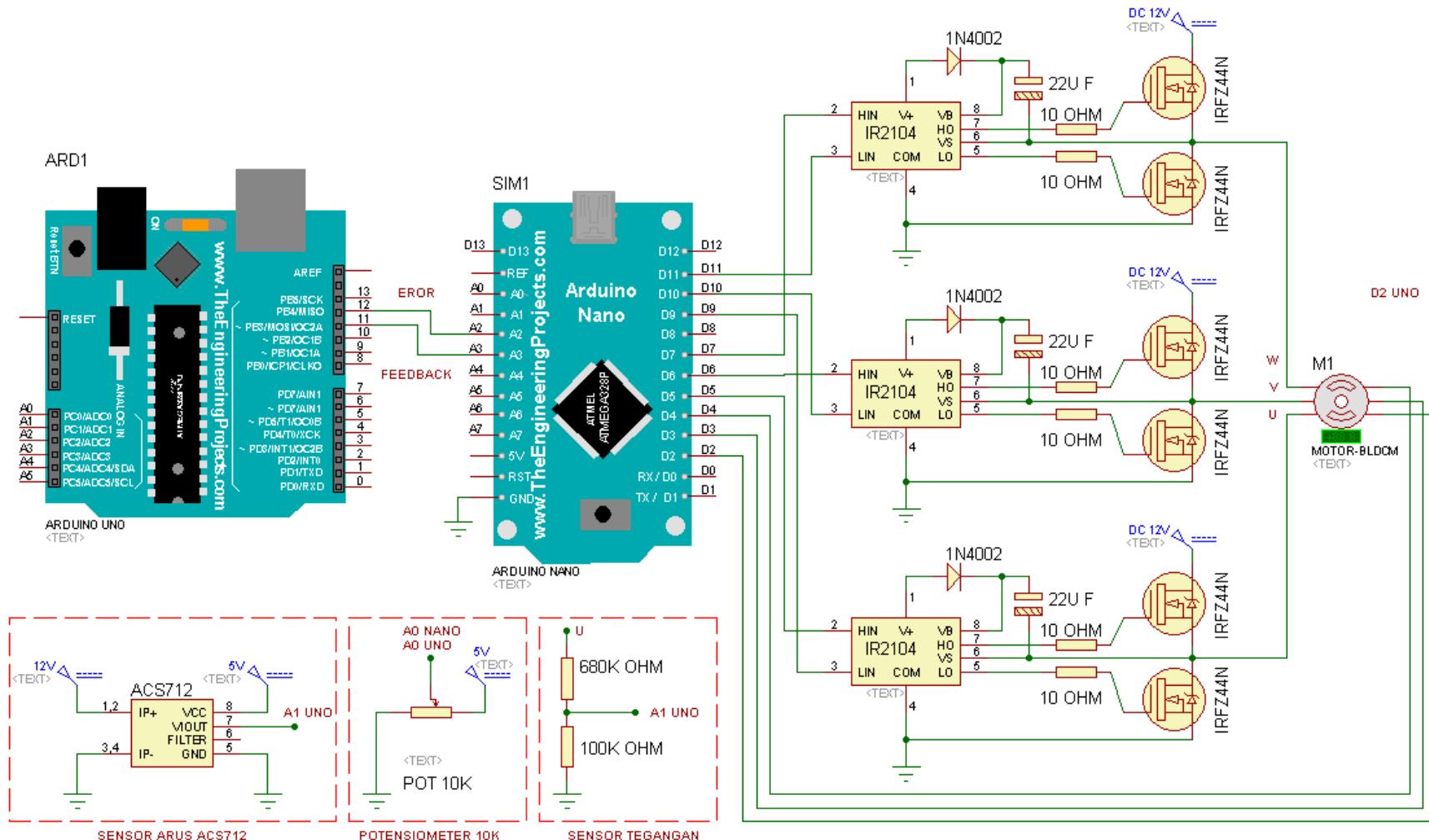


Gambar 4. 2 Grafik Pengujian Simulasi Keluaran *Driver*

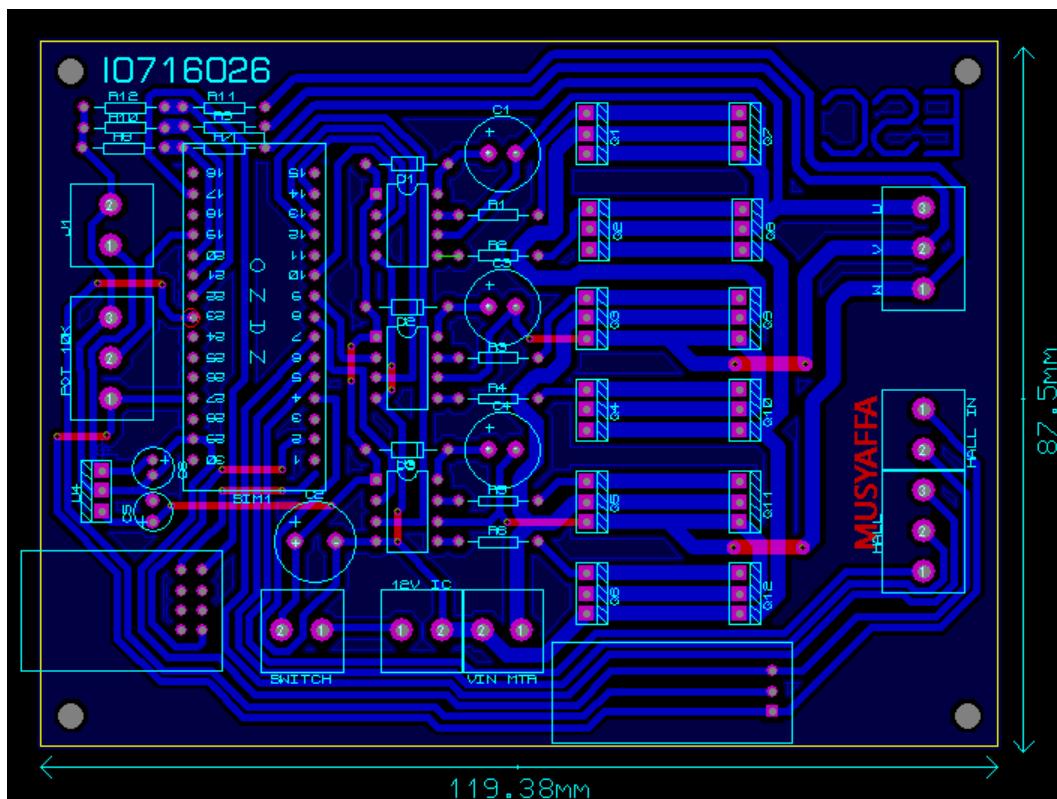
4.2 Pembuatan Alat

Pembuatan alat dilakukan setelah pengujian simulasi berhasil dilakukan dan diperoleh hasil sesuai yang diinginkan. Tahapan ini dimulai dengan membuat layout PCB berdasarkan rangkaian yang telah dibuat di simulasi Proteus. Setelah layout PCB selesai dibuat, dilakukan pengecekan jalur dan kemudian dicetak ke papan PCB. Kemudian dilakukan *assembly* komponen menggunakan solder. Hasil pembuatan alat dapat diamati pada gambar 4.3.

Rangkaian skematik tersebut kemudian diubah menjadi layout PCB menggunakan Proteus-Ares. Susunan komponen dibuat serapi mungkin dengan jalur yang sesuai dan meminimalisir adanya kabel *jumper*. Desain papan PCB menggunakan *single sided layer* dengan ukuran $11,9 \times 8,75$ Cm. Komponen yang digunakan pada rangkaian ini adalah komponen DIP. Desain layout PCB yang telah selesai dibuat dapat diamati pada gambar 4.4.

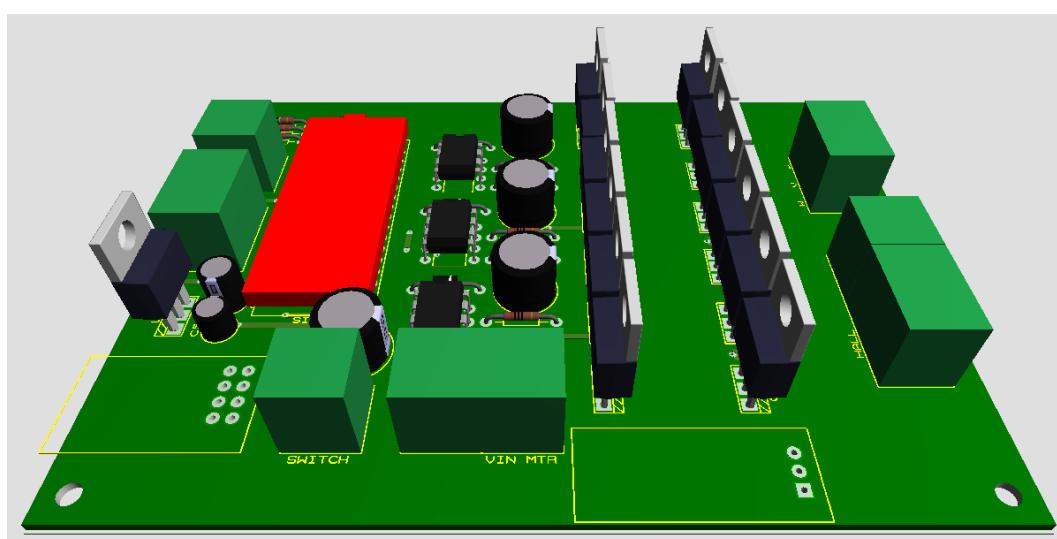


Gambar 4. 3 Desain Alat pada Proteus

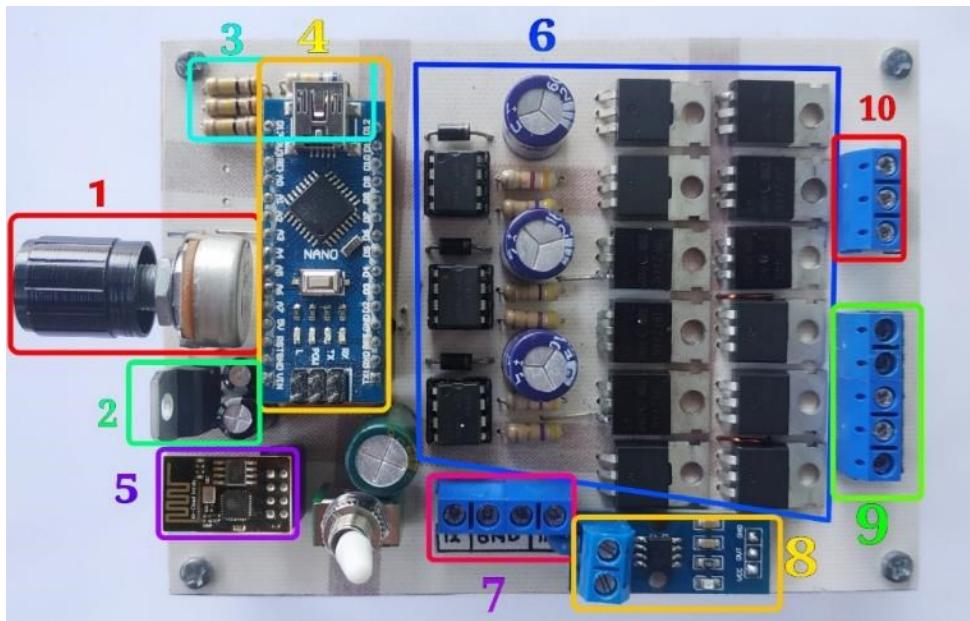


Gambar 4. 4 Layout PCB

Desain skematik yang dirancang merupakan gabungan dari beberapa komponen utama, yaitu mikrokontroler Arduino Nano, rangkaian *driver* mosfet yang disusun menjadi 3 fasa, rangkaian pembagi tegangan yang berperan sebagai sensor tegangan dan potensiometer yang mengatur nilai keluaran PWM. Desain 3 dimensi dari rangkaian yang dirancang dapat diamati pada gambar 4.5 berikut ini:



Gambar 4. 5 Desain 3D rangkaian *driver*



Gambar 4. 6 Hardware yang telah dibuat

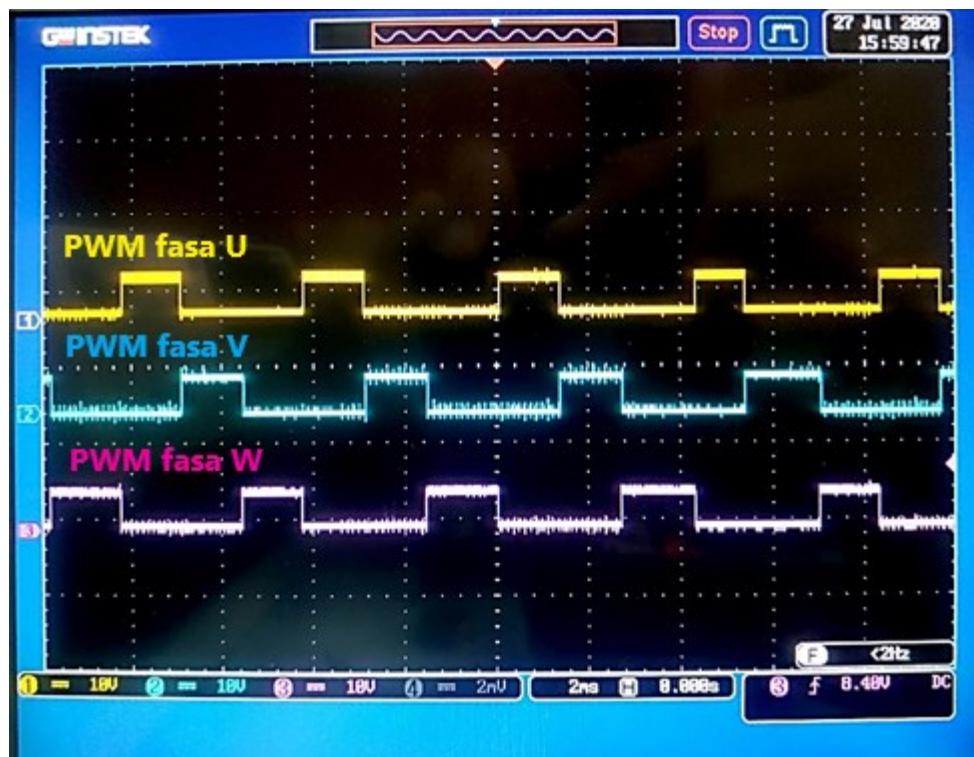
Pada gambar 4.6 diatas dapat diamati hardware yang tersusun dari komponen-komponen sesuai dengan desain yang dibuat. Berikut ini adalah keterangan dari rangkaian hardware yang telah dibuat:

1. Potensiometer 10k Ohm
2. Regulator tegangan AN7805
3. Sensor Tegangan (Rangkaian Pembagi Tegangan)
4. Mikrokontroler Arduino Nano
5. ESP 8266
6. Rangkaian *Driver* motor BLDC
7. Port Input tegangan suplai
8. Sensor Arus ACS712
9. Port sensor Hall
10. Port output *driver* 3 fasa

4.3 Pengujian Gelombang Keluaran Rangkaian *Driver*

Pada tahapan ini, dilakukan pengamatan bentuk gelombang keluaran dari rangkaian *driver* yang telah dibuat. Pengujian dilakukan dengan mengamati bentuk gelombang keluaran dari fasa UVW dan bentuk gelombang PWM. Hasil pengamatan kemudian dibandingkan dengan referensi dan hasil simulasi untuk melihat keakuratan alat.

4.3.1 Gelombang PWM



Gambar 4. 7 Gelombang PWM 50%

Pengujian ini dilakukan untuk mengetahui sinyal yang dihasilkan pada ketiga pin PWM Arduino Nano yang masuk menuju rangkaian driver. Sinyal PWM dari mikrokontroler akan mengatur kecepatan *switching* pada masing-masing fasa sehingga kecepatan putar motor dapat diatur. Pada pengujian ini diambil sampel nilai PWM 127 (*duty cycle* 50%) dimana motor berputar pada kecepatan 5.375 rpm untuk kemudian diamati menggunakan osiloskop seperti pada gambar 4.7. pada kondisi ini, tegangan yang mengalir ke motor adalah sebesar 7,63 V dan arus sebesar 0,28 A. Sinyal PWM yang dihasilkan sesuai dengan referensi, dimana terdapat perbedaan sudut fasa sebesar 120° antar fasa. Level tegangan dari PWM adalah HIGH untuk 5V dan LOW untuk 0V.

4.3.2 Gelombang Keluaran Fasa UVW Tanpa Beban

Pengujian ini dilakukan untuk mengetahui bentuk gelombang yang dihasilkan oleh rangkaian *driver* motor yang telah dibuat. Pada pengujian ini diambil sampel gelombang keluaran *driver* ketika diberi PWM sebesar 127 (*duty cycle* 50%) dan tidak ada pembebanan pada motor. Hasil pengujian dapat diamati pada gambar 4.8, yang menunjukkan bahwa gelombang

keluaran memiliki perbedaan sudut sebesar 120^0 antar fasa-nya. Namun pada pengamatan ini masih ditemukan noise pada sinyal keluaran, hal ini disebabkan oleh kualitas komponen yang digunakan, jenis motor yang diujikan dan pembacaan osiloskop yang kurang sempurna. Seperti diamati pada hasil pengukuran pada tabel 4.1, nilai tegangan keluaran bernilai 7,54 Vrms dimana tegangan maksimal pada jalur sebesar 17,6V dan -3,6V pada nilai minimal serta frekuensi sebesar 891,471 Hz. Nilai frekuensi akan meningkat mengikuti kenaikan dari kecepatan putar motor, dimana frekuensi maksimal yang dapat dihasilkan adalah 1,08 KHz Pada pengujian ini, nilai Vp-p adalah sebesar 21,2 V. Sedangkan arus yang mengalir adalah sebesar 0,18 A.



Gambar 4. 8 Gelombang keluaran fasa UVW pada kecepatan 5.375 rpm tanpa beban

Tabel 4. 1 Pengukuran *driver* tanpa beban

Vp-p	Vmax	Vmin	Vrms	Arus	F (Hz)	Setpoint
21,2 V	17,6V	-3,6 V	7,63 V	0,18 A	891,471	5.375 rpm

4.3.3 Gelombang Keluaran Fasa UVW Pembebanan Ringan



Gambar 4. 9 Gelombang fasa UVW kecepatan 5.375 rpm beban ringan

Pengujian ini dilakukan untuk mengetahui bentuk gelombang yang dihasilkan oleh rangkaian *driver* motor yang telah dibuat pada kondisi pemberian beban ringan berupa 2 buah koin. Pada pengujian ini diambil sampel gelombang keluaran *driver* ketika diberi PWM 127 (*duty cycle* 50%). Hasil pengujian dapat diamati pada gambar 4.9, yang menunjukkan bahwa gelombang keluaran memiliki perbedaan sudut sebesar 120^0 antar fasa-nya. Seperti diamati pada hasil pengukuran pada tabel 4.2, nilai tegangan keluaran bernilai 7,66 Vrms dimana tegangan maksimal pada jalur sebesar 17,2V dan -4V pada nilai minimal serta frekuensi sebesar 836,435 Hz. Pada pengujian ini, nilai Vp-p adalah sebesar 21,2 V dan arus sebesar 0,22 A.

Tabel 4. 2 Pengukuran *driver* beban ringan

Vp-p	Vmax	Vmin	Vrms	Arus	F (Hz)	Setpoint
21,2 V	17,2V	-4 V	7,66 V	0,22 A	836,435	5.375 rpm

4.3.4 Gelombang Keluaran Fasa UVW Pembebanan Sedang

Pada pengujian ini dilakukan untuk mengetahui bentuk gelombang yang dihasilkan oleh rangkaian *driver* motor yang telah dibuat pada kondisi

pemberian beban sedang berupa 3 buah koin. Pada pengujian ini diambil sampel gelombang keluaran *driver* ketika diberi PWM 127 (*duty cycle* 50%). Hasil pengujian dapat diamati pada gambar 4.10, yang menunjukkan bahwa gelombang keluaran memiliki perbedaan sudut sebesar 120^0 antar fasa-nya. Seperti diamati pada hasil pengukuran pada tabel 4.3 Seperti diamati pada hasil pengukuran pada tabel 4.3, nilai tegangan keluaran bernilai 7,88 Vrms dimana tegangan maksimal pada jalur sebesar 17,6V dan -3,6V pada nilai minimal serta frekuensi sebesar 740,358 Hz. Pada pengujian ini, nilai Vp-p adalah sebesar 21,2 V dan arus sebesar 0,26 A.



Gambar 4. 10 Gelombang fasa UVW kecepatan 5.375 rpm beban sedang

Tabel 4. 3 Pengukuran *driver* beban sedang

Vp-p	Vmax	Vmin	Vrms	Arus	F (Hz)	Setpoint
21,2 V	17,6V	-3,6 V	7,88 V	0,26 A	740,358	5.375 rpm

4.3.5 Gelombang Keluaran Fasa UVW Pembebanan Berat

Pengujian keempat ini dilakukan dengan memberikan 4 buah koin. Pada pengujian ini diambil sampel gelombang keluaran *driver* ketika diberi PWM 127 (*duty cycle* 50%). Hasil pengujian dapat diamati pada gambar 4.11, yang menunjukkan bahwa gelombang keluaran memiliki perbedaan sudut sebesar 120^0 antar fasa-nya. Seperti diamati pada hasil pengukuran pada tabel 4.4, nilai tegangan keluaran bernilai 7,67 Vrms dimana tegangan maksimal pada jalur sebesar 17,2V dan -4V pada nilai minimal serta frekuensi sebesar

698,102 Hz. Pada pengujian ini, nilai V_{p-p} adalah sebesar 21,2 V. Sedangkan arus yang mengalir sebesar 0,28 A.



Gambar 4. 11 Gelombang fasa UVW kecepatan 5.375 rpm beban berat

Tabel 4. 4 Pengukuran *driver* beban berat

V _{p-p}	V _{max}	V _{min}	V _{rms}	Arus	F (Hz)	Setpoint
21,2 V	17,2V	-4 V	7,67 V	0,28 A	698,102	5.375 rpm

4.3.6 Gelombang Keluaran Fasa UVW Pembebatan Sangat Berat

Pengujian terakhir dilakukan dengan memberikan 8 buah koin. Pada pengujian ini diambil sampel gelombang keluaran *driver* ketika diberi PWM 127 (*duty cycle* 50%). Hasil pengujian dapat diamati pada gambar 4.12, yang menunjukkan bahwa gelombang keluaran memiliki perbedaan sudut sebesar 120° antar fasa-nya. Seperti diamati pada hasil pengukuran pada tabel 4.5, nilai tegangan keluaran *driver* bernilai 7,98 Vrms dimana tegangan maksimal pada jalur sebesar 18 V dan -4V pada nilai minimal serta frekuensi sebesar 772,468 Hz. Pada pengujian ini, nilai V_{p-p} adalah sebesar 22 V. Sedangkan arus yang mengalir adalah sebesar 0,31 A.



Gambar 4. 12 Gelombang fasa UVW kecepatan 5.375 rpm beban sangat berat

Tabel 4. 5 Pengukuran *driver* beban sangat berat

Vp-p	Vmax	Vmin	Vrms	Arus	F (Hz)	Setpoint
22 V	18 V	-4 V	7,98 V	0,31 A	772,468	5.375 rpm

4.4 Pengujian Pembacaan nilai PWM

Pengujian pembacaan nilai PWM dilakukan dengan membaca nilai keluaran dari potensiometer yang terhubung ke arduino port analog 0. Nilai potensiometer ini memiliki range tegangan dari 0 sampai 5V yang kemudian dikonversi menjadi 0-1023 oleh arduino dan dikonversi lagi menjadi nilai keluaran PWM dengan rentang 0-255.

4.4.1 Tujuan Utama Pengujian Pembacaan Nilai PWM

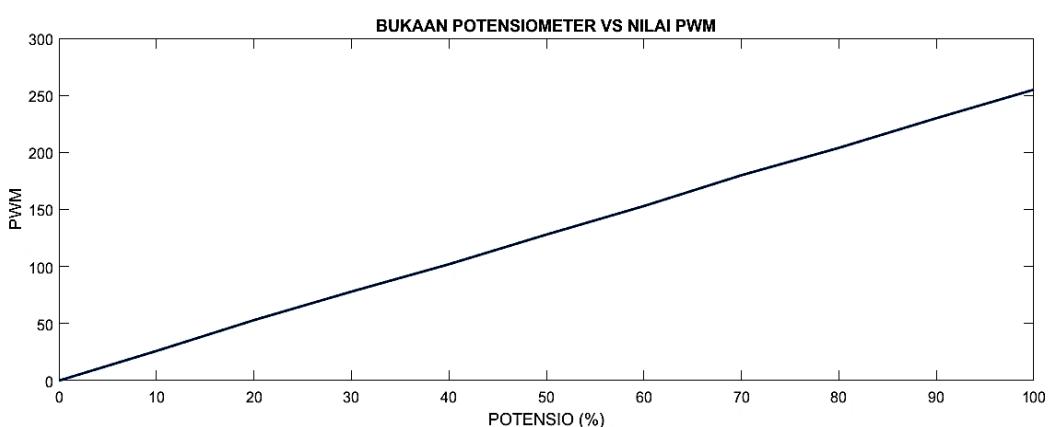
Pengujian ini dilakukan untuk membaca nilai keluaran PWM pada arduino berdasarkan bukaan potensiometer. Nilai PWM yang keluar bernilai minimal 0 dan maksimal 255. Perubahan PWM bernilai konstan dengan mengikuti putaran potensiometer yang terhubung ke arduino port A0 yang akan mengatur kecepatan putar motor BLDC. Nilai keduanya ditampilkan pada serial monitor Arduino IDE yang kemudian diolah menjadi grafik untuk menentukan akurasi dari potensiometer yang digunakan terhadap perubahan nilai PWM.

4.4.2 Hasil Pembacaan Nilai PWM

Pembacaan nilai PWM dilakukan melalui pengamatan pada serial monitor Arduino, dimana nilai potensiometer diubah-ubah sehingga nilai dari PWM yang dihasilkan ikut berubah. Perubahan potensiometer akan menghasilkan tegangan bervariasi dengan rentang dari 0V hingga 5V yang diteruskan pada port analog 0 arduino. Nilai input yang terbaca dari port tersebut kemudian diolah menjadi nilai PWM dengan rentang dari 0 hingga 255. Nilai PWM ini akan mempengaruhi besar bukaan dari *output* IC, yang akan mempengaruhi kecepatan putar motor BLDC. Nilai PWM yang dihasilkan oleh alat ini dapat diamati pada tabel 4.6 berikut ini:

Tabel 4. 6 Respon PWM terhadap perubahan bukaan potensiometer

Bukaan Potensio (%)	Nilai PWM
0%	0
10 %	26
20 %	53
30 %	78
40 %	102
50 %	128
60 %	153
70 %	180
80 %	204
90 %	230
100 %	255



Gambar 4. 13 Grafik hubungan Potensio dan Nilai PWM

Nilai pembacaan PWM tersebut kemudian ditampilkan dalam grafik pada gambar 4.13 untuk mengetahui hubungan perubahan bukaan pitensiometer dengan nilai PWM.

4.5 Pengujian Pembacaan Sensor Arus dan Sensor Tegangan

Pengujian sensor ini dilakukan dengan membaca nilai keluaran dari sensor tegangan dan sensor arus pada rangkaian. Sensor yang terhubung pada port analog 1 hingga analog 3 akan menghasilkan tegangan sebesar 0-5V yang kemudian dikonversi oleh arduino menjadi 0-36 VDC. Sedangkan nilai pembacaan dari sensor arus ACS712 akan diteruskan ke arduino melalui port analog 4. Nilai keluaran dari sensor arus ini adalah sebesar 0-5A yang diolah menjadi 0-20A oleh mikrokontroler.

4.5.1 Tujuan Utama Pembacaan Nilai Arus dan Tegangan

Tujuan dari pembacaan nilai sensor ini adalah untuk mengetahui seberapa besar tegangan dan arus yang digunakan oleh rangkaian ini, pada setiap perubahan nilai PWM. Nilai pembacaan tegangan adalah dalam bentuk Vrms. Hasil pembacaan yang benar adalah nilai tegangan dan arus ikut berubah mengikuti perubahan nilai *duty cycle* yang dihasilkan oleh mikrokontroler.

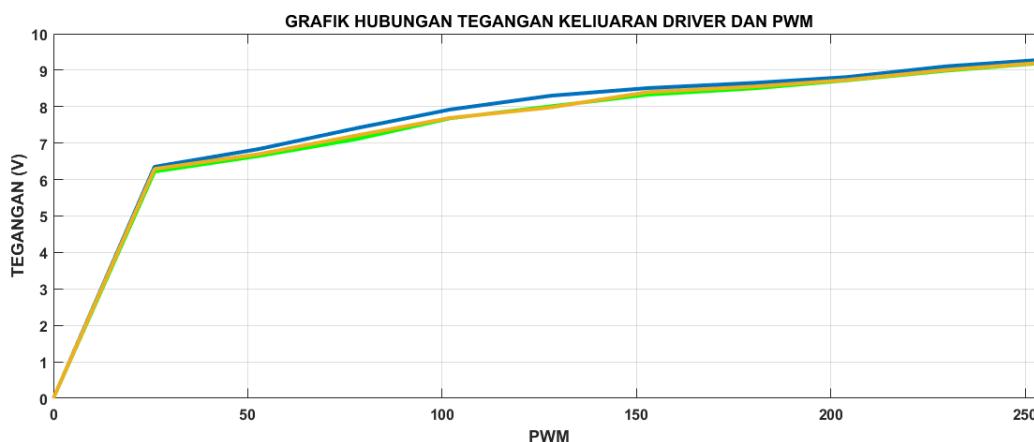
4.5.2 Hasil Pembacaan Nilai Arus dan Tegangan

Tabel 4. 7 Hasil Pengujian Sensor Arus dan Tegangan

Potensio (%)	PWM	Tegangan U (V)	Tegangan V (V)	Tegangan W (V)	Arus (A)
0	0	0	0	0	0,01
10	26	6,35	6,22	6,29	0,05
20	53	6,84	6,65	6,70	0,12
30	78	7,41	7,11	7,21	0,14
40	102	7,92	7,68	7,69	0,15
50	128	8,30	8,01	7,98	0,18
60	153	8,51	8,33	8,40	0,20
70	180	8,65	8,50	8,55	0,25
80	204	8,81	8,72	8,73	0,29
90	230	9,11	8,99	9,01	0,34
100	255	9,29	9,20	9,21	0,44

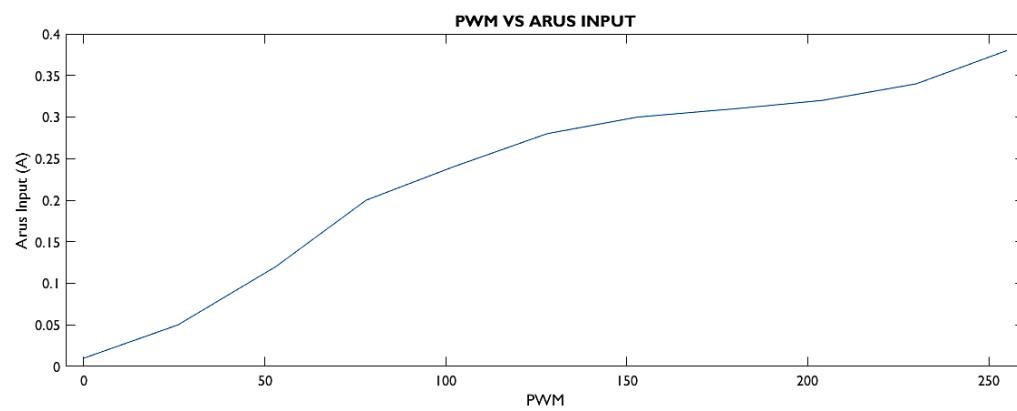
Hasil pembacaan dari sensor arus menggunakan ACS712 dan sensor tegangan menggunakan rangkaian pembagi tegangan menunjukkan respon perubahan arus input dan tegangan pada ketiga fasa. Perubahan nilai pembacaan sensor mengikuti perubahan nilai pwm yang diberikan ke rangkaian. Nilai maksimal tegangan tanpa beban yang digunakan oleh motor adalah sebesar 9,29 VDC dan 0 VDC ketika motor berhenti berputar.

Nilai tegangan pada ketiga fasa terdapat sedikit eror karena perbedaan kualitas komponen yang digunakan serta kecepatan pembacaan Arduino terhadap *switching* motor BLDC. Nilai pembacaan sensor dapat diamati pada tabel 4.7. Tabel tersebut kemudian diolah menjadi grafik hubungan pwm dan tegangan 3 fasa serta hubungan pwm dan arus input yang ditampilkan pada gambar 4.14 berikut:



Gambar 4. 14 Hubungan nilai PWM dan tegangan 3 fasa

Grafik diatas menunjukkan hubungan perubahan nilai tegangan pada masing-masing fasa terhadap perubahan nilai potensiometer. Grafik diatas menunjukkan hubungan yang baik antara keduanya. Nilai tegangan masing-masing fasa mengikuti perubahan nilai PWM yang diberikan oleh mikrokontroler.



Gambar 4. 15 Hubungan nilai PWM dan Arus Input

Sedangkan pada gambar 4.15, dapat diamati mengenai hubungan perubahan nilai PWM terhadap nilai arus input yang masuk ke rangkaian. Grafik diatas

menunjukkan bahwa perubahan arus mengikuti perubahan nilai PWM yang diberikan oleh mikrokontroler.

4.6 Pengujian Pembacaan Kecepatan Motor BLDC

Pembacaan nilai kecepatan motor dilakukan dengan pembacaan kondisi salah satu sensor hall yan terhubung pada pin *interrupt* arduino nano dan kemudian diolah tiap satu detik menjadi nilai kecepatan putar dalam RPM. Pada rangkaian yang dibuat, pembacaan nilai hall sensor dilakukan pada port digital 2 yang terhubung dengan keluaran sensor hall 1. Pembacaan tersebut kemudian diolah oleh mikrokontroler untuk ditampilkan menjadi nilai RPM pada serial monitor Arduino. Pada penelitian ini, dikarenakan sedang terjadi pandemi Covid-19, maka motor yang diujikan adalah motor BLDC yang diambil dari CD ROM, yang kemudian dimodifikasi agar pengkabelannya sama dengan motor BLDC pada sepeda listrik. Spesifikasi dari motor ini adalah 2,76W dengan tegangan maksimal sebesar 12V dan arus nominal sebesar 0,23A.

4.6.1 Tujuan Pengujian Pembacaan Nilai Kecepatan Motor BLDC

Pengujian ini dilakukan untuk mengetahui perubahan nilai kecepatan putar (rpm) motor BLDC. Perubahan nilai kecepatan tersebut dipengaruhi oleh nilai PWM yang mempengaruhi *duty cycle* tegangan yang masuk ke *driver*. Pembacaan nilai kecepatan ini dilakukan pada beberapa kondisi, yaitu ketika motor tanpa diberi beban dan pemberian variasi beban.

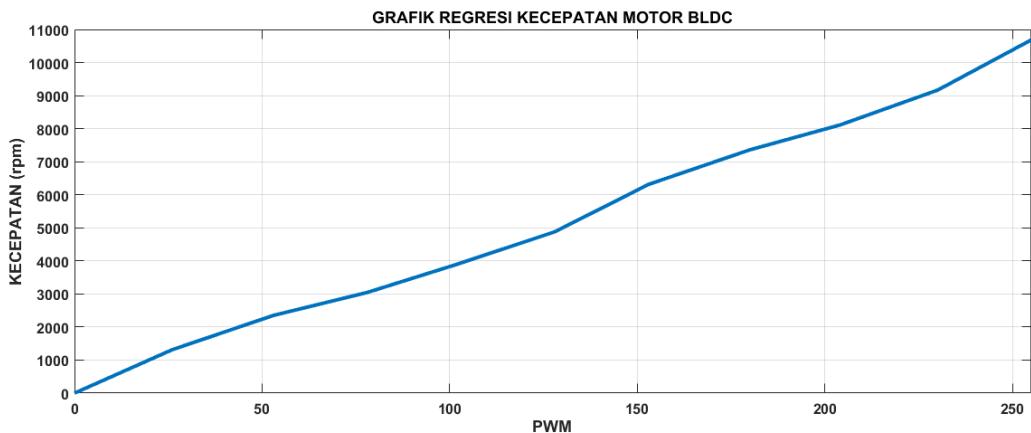
4.6.2 Hasil Pembacaan Kecepatan Motor pada Variasi Potensiometer

Hasil pembacaan nilai kecepatan motor BLDC oleh mikrokontroler arduino nano menunjukkan bahwa kecepatan putar motor dipengaruhi oleh nilai PWM yang masuk ke motor BLDC. Nilai kecepatan maksimal dari motor bldc ini adalah 9785 rpm pada bukaan potensiometer 100% dan motor akan berhenti ketika bukaan potensiometer tertutup sempurna. Nilai ini hampir mendekati spesifikasi motor BLDC yaitu 10.000 rpm. Pengujian ini dilakukan pada kondisi tanpa beban, untuk mengetahui performa maksimal dari motor BLDC yang diujikan. Hasil pengujian dapat diamati pada tabel 4.8 dibawah ini:

Tabel 4. 8 Perubahan nilai PWM terhadap kecepatan putar motor

Potensio (%)	PWM	Tanpa Beban (rpm)
0	0	0
10	26	1.312
20	53	2.351
30	78	3.045
40	102	3.900
50	128	4.880
60	153	6.315
70	180	7.360
80	204	8.111
90	230	9.164
100	255	10.685

4.6.3 Grafik Regresi Perubahan PWM dan Nilai Kecepatan Motor



Gambar 4. 16 Grafik Regresi Kecepatan Motor BLDC

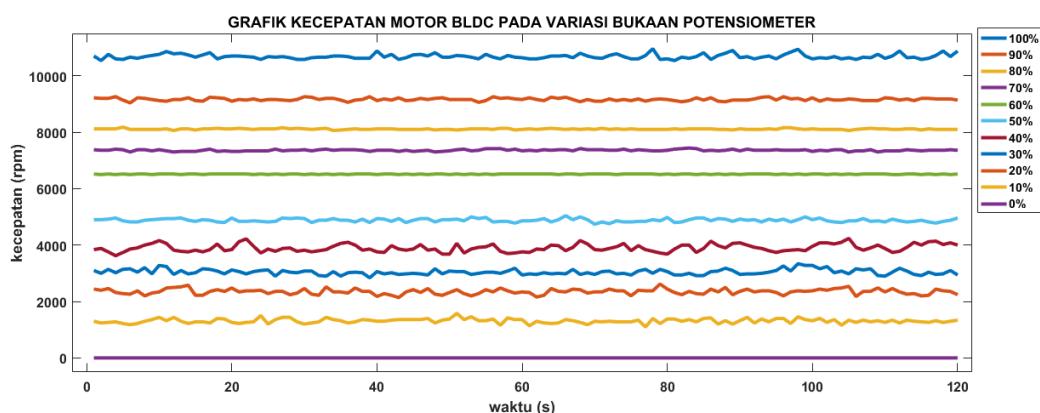
Hasil pengujian kecepatan putaran motor BLDC diatas kemudian diolah menjadi grafik regresi seperti pada gambar 4.16 untuk mengetahui persamaan dari alat yang dibuat. Dari grafik tersebut dapat diamati bahwa kenaikan kecepatan putaran motor berlangsung cukup stabil. Kecepatan maksimal dapat dicapai dengan memberikan nilai PWM sebesar 255 ke rangkaian *driver*. Dari grafik tersebut dapat diketahui persamaan regresi sebagai berikut:

$$y = 40,431x + 14,8 \quad (6)$$

Persamaan regresi diatas merupakan persamaan yang menunjukkan hubungan antar nilai pwm dan nilai kecepatan putar motor. Y menunjukkan nilai kecepatan putar dan x merupakan nilai PWM. Dari persamaan tersebut dapat digunakan untuk mencari nilai kecepatan motor pada PWM tertentu.

4.6.4 Pengujian Pelacakan Kecepatan pada Variasi Setpoint

Pengujian kecepatan motor juga dilakukan pada variasi nilai bukaan potensiometer. Pengujian ini dilakukan untuk mengetahui tingkat presisi dari putaran motor terhadap nilai *setpoint* yang diberikan. Nilai *setpoint* tersebut diberikan melalui bukaan potensiometer dari 0% , 10%, 20% dan seterusnya hingga bukaan potensiometer maksimal atau 100%. Pembacaan dilakukan pada masing-masing variasi selama 2 menit dan menghasilkan 120 data untuk diolah dan dibandingkan. Grafik pengujian ini dapat diamati pada gambar 4.17 berikut ini:



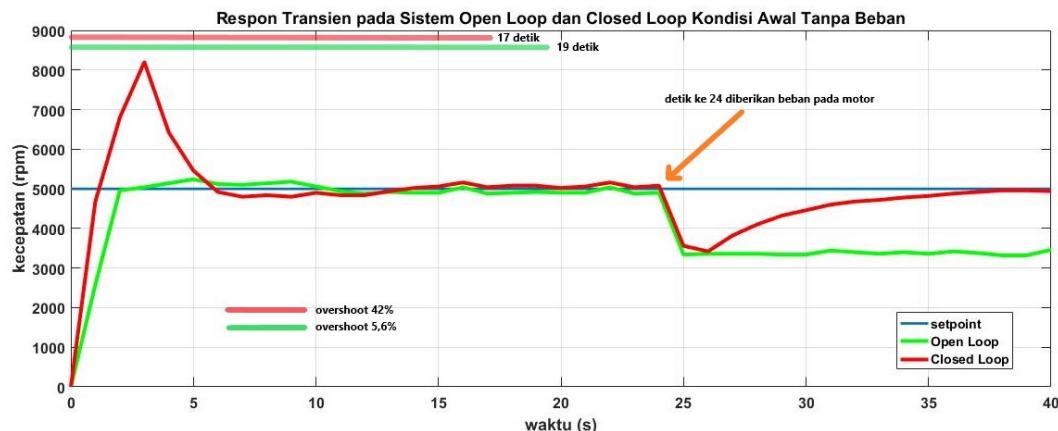
Gambar 4. 17 Pelacakan kecepatan pada motor BLDC

Berdasarkan grafik diatas, masing-masing variasi bukaan potensiometer memberikan nilai *setpoint* yang berbeda. Rangkaian *driver* dapat merespon cukup baik dan menghasilkan putaran motor yang cukup stabil. Namun pada bukaan potensiometer dibawah 50%, kecepatan motor yang dihasilkan kurang stabil. Hal ini dikarenakan jenis motor yang digunakan memiliki interferensi sehingga kecepatannya tidak dapat stabil pada nilai putaran rendah.

4.7 Pengujian Respon Motor BLDC pada Setpoint Kecepatan

Pada pengujian ini, dilakukan penyetelan nilai *setpoint* kecepatan untuk motor BLDC, dengan tujuan mengamati respon transien dari motor BLDC untuk

mencapai kecepatan *steady state* baik dengan metode *open loop* maupun metode *closed loop*. Pengujian dilakukan dengan mengatur nilai *setpoint* kecepatan sebesar 5000 rpm. Hasil pengujian menunjukkan bahwa dengan menggunakan sistem *closed loop*, kecepatan motor mengalami kenaikan dan mencapai kecepatan steady state pada detik ke 17. Nilai *overshoot* kecepatan adalah sebesar 42 %. Sedangkan pada metode *open loop*, *overshoot* kecepatan cukup kecil yaitu 5,6% namun membutuhkan waktu 19 detik untuk dapat mencapai kecepatan *steady state* pada *setpoint* yang diberikan. Meskipun nilai *overshoot* yang dihasilkan oleh sistem *closed loop* jauh lebih besar, namun ketika diberikan pembebanan pada detik ke 24, kecepatan motor yang menggunakan sistem *closed loop* akan turun dan kemudian segera naik mengejar kecepatan setpoint yang diberikan. Sedangkan pada sistem *open loop*, setelah diberikan beban, kecepatan motor jatuh dan tidak mampu mengejar nilai setpoint lagi. Hal ini menunjukkan bahwa penggunaan sistem *closed loop* dinilai lebih baik dan handal dibandingkan sistem *open loop*. Grafik pengujian ini dapat diamati pada gambar 4.18 berikut:



Gambar 4. 18 Grafik Transien Respon Time

4.8 Pengujian Variasi Pembebanan pada Motor BLDC

Pada pengujian ini, dilakukan penambahan variasi beban ke motor BLDC untuk melihat performa dari sistem yang dibuat. Pengujian dilakukan dengan menambahkan variasi beban berupa uang koin 500 rupiah yang berwarna kuning, dengan berat 5,3 gram. Variasi pembebanan dapat diamati pada gambar 4.19 berikut:



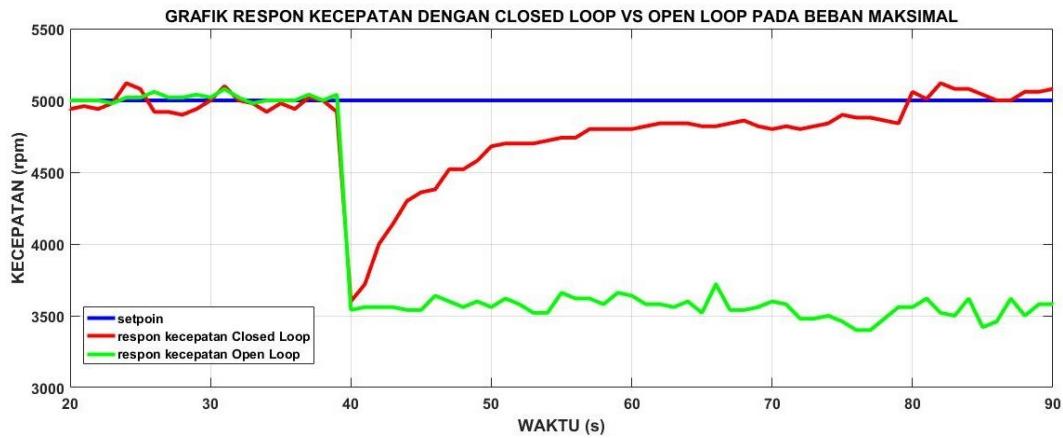
Gambar 4. 19 Variasi Pembebanan

Pengujian dilakukan dengan menentukan satu set poin kecepatan dan kemudian diberikan variasi beban, untuk melihat respon sistem. Pada kondisi pembebanan ini, nilai tegangan dan arus yang masuk ke motor dibatasi, yaitu sebesar 15 V dan 1 A, dengan tujuan sebagai proteksi agar motor BLDC yang digunakan tidak mengalami *overvoltage* maupun *overcurrent* yang dapat membakar motor.

4.8.1 Perbandingan Sistem *Closed Loop* dan *Open Loop* pada Pembebanan

Pada pengujian ini akan diamati perbandingan respon kecepatan ketika menggunakan metode sistem kendali *closed loop* (otomatis) dan *open loop*. Pengujian dilakukan dengan memberikan variasi pembebanan sangat berat berupa 8 buah koin dengan berat 42,4 gram. Pemberian beban dilakukan ketika motor berputar pada kecepatan *setpoint* yang diberikan, yaitu 5000 rpm, lalu diamati perubahan kecepatan dari motor BLDC. Pada gambar 4.20, ditunjukkan hasil pengujian bahwa ketika diberikan pembebanan, kecepatan motor akan menurun, dimana pada penggunaan metode kendali otomatis, kecepatan turun dari 5000 rpm menjadi 3520 rpm yang 40 detik kemudian mengejar kecepatan *setpoint* 5000 rpm dan *steady state* setelah 5 detik kemudian. Namun pada pengujian *open loop*, nilai kecepatan ketika terjadi pembebanan turun menjadi 3500 rpm dan tidak dapat mengejar kecepatan *setpoint* yang diberikan dan tetap pada kecepatan 3500 rpm hingga akhir pengujian. Pada pengujian ini dapat diamati bahwa penggunaan metode kendali otomatis dapat memberikan respon yang bagus untuk mempertahankan kecepatan motor ketika terjadi pembebanan, namun pada pengujian ini masih ditemui eror berupa ketidakstabilan kecepatan dan respon untuk mencapai kecepatan *setpoint* yang agak lama karena keterbatasan mikroprosesor dan algoritma kendali yang digunakan. Sehingga dari pengujian ini, akan dilakukan

pengujian selanjutnya pada pemberian variasi pembebahan menggunakan metode kendali otomatis.



Gambar 4. 20 Perbandingan respon kecepatan *Closed Loop* dan sistem *Open Loop*

4.8.2 Pengujian *Driver* pada Pembebahan Ringan (10,6 gram)

Pada pengujian ini, dilakukan penambahan beban ringan sebesar 21,2 gram ke motor untuk mengetahui respon yang diberikan oleh motor dengan menggunakan metode kendali otomatis. *Setpoint* kecepatan diatur pada nilai 5000 rpm. Pada penambahan beban ini, nilai kecepatan motor turun menjadi 4380 rpm motor mengejar kecepatan *setpoint* dan stabil selama 25 detik. Nilai tegangan pada kondisi ini seperti ditampilkan pada tabel 4.9 sebesar 7,66 V pada tiap fasa dan frekuensi 856,435 Hz. Arus yang digunakan oleh *driver* ini naik menjadi 0,34 A. Sedangkan arus yang ditarik oleh motor BLDC sebesar 0,21 A. Hal ini dikarenakan pada pembebahan ini, motor berupaya mempertahankan nilai *setpoint* yang diberikan dengan menarik arus lebih besar dari sumber, untuk meningkatkan torsi motor. Grafik pengujian pembebahan ringan dapat diamati pada gambar 4.21.



Gambar 4. 21 Grafik Respon Pembebanan Ringan

Tabel 4. 9 Pengujian *Driver* dengan Beban Ringan

<i>Setpoint</i> <i>t</i> (RPM)	Tegangan U (V)	Tegangan V (V)	Tegangan W (V)	Arus <i>drive</i> <i>r</i> (A)	Arus Motor (A)	Frekuensi (Hz)
5000	7,66	7,66	7,66	0,34	0,21	856,435

4.8.3 Pengujian *Driver* pada Pembebanan Sedang (21,2 gram)

Tabel 4. 10 Pengujian *Driver* dengan Beban Sedang

<i>Setpoint</i> <i>t</i> (RPM)	Tegangan U (V)	Tegangan V (V)	Tegangan W (V)	Arus <i>drive</i> <i>r</i> (A)	Arus Motor (A)	Frekuensi (Hz)
5000	7,88	7,88	7,88	0,37	0,25	740,358

Pada pengujian pembebanan sedang, dilakukan penambahan beban sedang sebesar 21,2 gram ke motor untuk mengetahui respon yang diberikan oleh motor. *Setpoint* kecepatan diatur pada nilai 5000 rpm. ada penambahan beban ini, nilai kecepatan motor turun menjadi 3920 rpm yang kemudian selama 26 detik, kecepatan motor BLDC akan dikejar hingga mencapai *setpoint* kembali. Seperti ditampilkan pada tabel 4.10, nilai tegangan pada kondisi ini sebesar 8,91 V pada tiap fasa dan frekuensi 740,358 Hz. Arus yang digunakan oleh *driver* ini naik menjadi 0,37 A. Sedangkan arus yang ditarik oleh motor BLDC sebesar 0,25 A. Hal ini dikarenakan pada pembebanan ini, motor berupaya mempertahankan nilai *setpoint* yang diberikan dengan menarik arus lebih besar dari sumber, untuk meningkatkan torsi motor. Grafik pengujian pembebanan sedang dapat diamati pada gambar 4.22.



Gambar 4. 22 Grafik Respon Pembebanan Sedang

4.8.4 Pengujian *Driver* pada Pembebanan Berat (31,8 gram)



Gambar 4. 23 Grafik Respon Pembebanan Berat

Pengujian ketiga dilakukan dengan memberikan penambahan beban berat sebesar 31,8 gram ke motor untuk mengetahui respon yang diberikan oleh motor. *Setpoint* kecepatan diatur pada nilai 5000 rpm. Pada penambahan beban ini, nilai kecepatan motor turun menjadi 3720 rpm dan kemudian naik menjadi kecepatan *setpoint* 5000 rpm selama 30 detik. Seperti ditampilkan pada tabel 4.11, nilai tegangan pada kondisi ini sebesar 7,93 V pada tiap fasa dan frekuensi 698,182 Hz. Arus yang digunakan oleh *driver* ini naik menjadi 0,40 A. Sedangkan arus yang ditarik oleh motor BLDC sebesar 0,27 A. Grafik pengujian pembebanan berat dapat diamati pada gambar 4.23.

Tabel 4. 11 Pengujian *Driver* dengan Beban Berat

<i>Setpoint</i> <i>t</i> (RPM)	Tegangan U (V)	Tegangan V (V)	Tegangan W (V)	Arus <i>drive</i> <i>r</i> (A)	Arus Motor (A)	Frekuensi (Hz)
5000	7,93	7,93	7,93	0,40	0,27	698,182

4.8.5 Pengujian *Driver* pada Pembebanan Sangat Berat (42,4 gram)

Pengujian terakhir adalah pemberian penambahan beban sangat berat sebesar 42,4 gram ke motor untuk mengetahui respon yang diberikan oleh motor. Pada penambahan beban ini, nilai kecepatan motor turun menjadi 3600 rpm. Motor kemudian mengejar kecepatan hingga mencapai *steady state* di 5000 rpm selama 8 detik. Seperti ditampilkan pada tabel 4.12, nilai tegangan

pada kondisi ini sebesar 7,98 V pada tiap fasa dan frekuensi 772,468 Hz. Arus yang digunakan oleh *driver* ini naik menjadi 0,45 A. Sedangkan arus yang ditarik oleh motor BLDC sebesar 0,3 A. Grafik pengujian pembebanan sangat dapat diamati pada gambar 4.24.

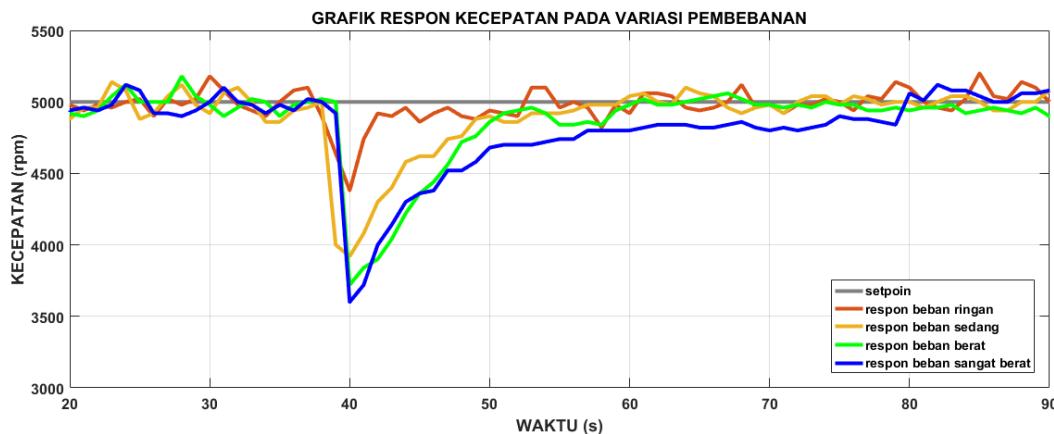


Gambar 4. 24 Grafik Respon Pembebanan Sangat Berat

Tabel 4. 12 Pengujian *Driver* dengan Beban Sangat Berat

<i>Setpoint</i> <i>t</i> (RPM)	Tegangan U (V)	Tegangan V (V)	Tegangan W (V)	Arus <i>driver</i> (A)	Arus Motor (A)	Frekuensi (Hz)
5000	7,98	7,98	7,98	0,45	0,3	772,468

4.8.6 Perbandingan Respon Kecepatan pada Variasi Pembebanan



Gambar 4. 25 Respon Kecepatan pada Variasi Pembebanan

Dari pengujian yang telah dilakukan, diperoleh hasil bahwa penggunaan metode kendali otomatis dapat memperbaiki eror kecepatan yang dihasilkan oleh pemberian variasi pembebanan sehingga sistem yang dirancang dapat optimal. Hasil perbandingan respon variasi pembebanan dapat diamati pada gambar 4.25. Dimana penurunan kecepatan mengikuti pertambahan nilai

pembebanan yang dilakukan. Dengan bantuan kendali otomatis, kecepatan motor BLDC akan kembali naik menuju titik *setpoint* dengan waktu rata-rata selama 30 detik. Pada pengujian ini masih ditemukan beberapa eror kecepatan yang disebabkan oleh kualitas komponen yang digunakan, interferensi motor dan keterbatasan mikroprosesor untuk mengirimkan *feedback error* secara *real time*.



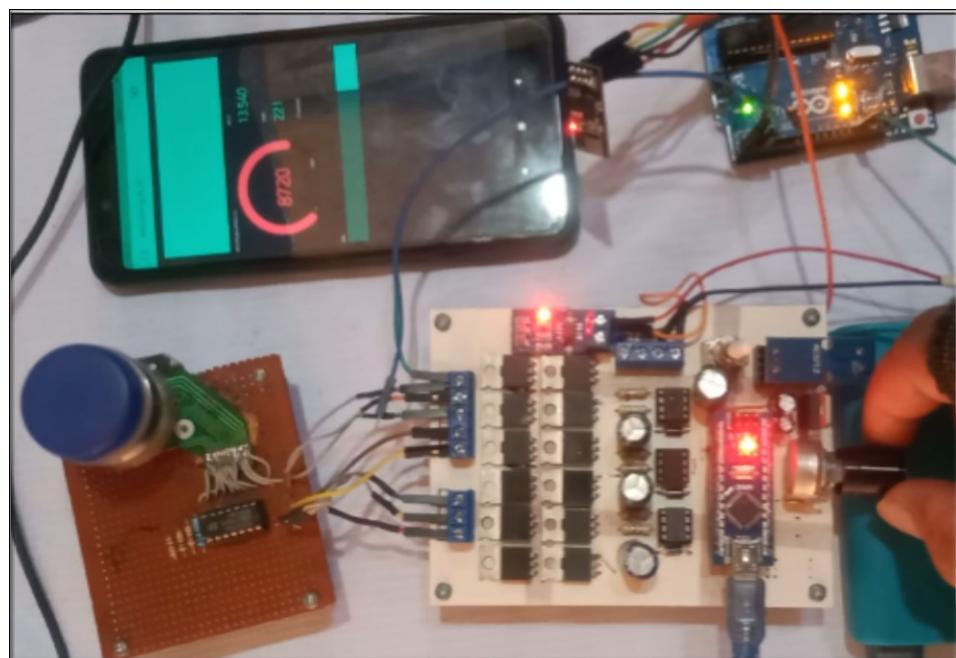
Gambar 4. 26 Grafik variasi beban vs waktu steady state

Gambar 4.26 diatas menunjukkan hubungan antara variasi pembebanan dengan waktu respon motor untuk kembali ke kecepatan awal. Dengan menggunakan metode kendali otomatis, kecepatan dari motor dapat dipertahankan ketika diberikan pembebanan, namun membutuhkan waktu yang agak lama, yang sebanding dengan besarnya variasi pembebanan yang diberikan. Semakin besar nilai pembebanan yang diberikan, maka semakin lama waktu yang dibutuhkan untuk mencapai dan stabil pada kecepatan awal.

4.9 Pengujian Monitoring *Blynk*

Pengujian terakhir yang dilakukan adalah menguji sistem monitoring dari aplikasi *Blynk* yang ditampilkan pada smartphone. Pengujian dilakukan dengan memberikan sampel putaran kecepatan motor, yang kemudian direspon oleh *Blynk* dengan menampilkan data kecepatan, arus, tegangan dan bukaan potensiometer. *Delay* pembacaan dari aplikasi *blynk* diatur sebesar 1 detik untuk menjaga

pengiriman data dari ESP8266 tetap stabil. Hasil pengujian dapat diamati pada gambar 4.27 berikut:



Gambar 4. 27 Sistem Kendali Motor BLDC dengan Monitoring *Blynk*

4.10 Analisa Efisiensi Sistem yang Dirancang

Sebelum melakukan perhitungan efisiensi, terlebih dahulu dilakukan pengujian variasi pembebahan pada motor, dengan keadaan PWM = 255 atau motor berada pada kecepatan maksimal. Parameter yang diamati pada pengujian ni dapat dilihat pada tabel 4.13. Analisa ini dilakukan untuk mengetahui performa dari drive motor BLDC pada kondisi tanpa beban dan kondisi dengan variasi pembebahan. Data pada tabel 4.13 kemudian diolah untuk menentukan efisiensi pada masing-masing kondisi pembebahan motor.

Tabel 4. 13 Arus dan Tegangan pada Variasi Pembebahan

Kondisi	Driver			Motor		
	V	I	P	V	I	P
Tanpa Beban	10,27	0,45	4,6215	9,29	0,44	4,0876
Beban Ringan	10,55	0,54	5,697	9,32	0,53	4,9396
Beban Sedang	10,76	0,57	6,1332	9,54	0,57	5,4378
Beban Berat	11,5	0,59	6,785	9,33	0,64	5,9712
Beban Sangat Berat	12	0,64	7,68	9,64	0,66	6,3624

4.10.1 Perhitungan Kinerja Tanpa Beban

Pada pengujian tanpa pembebahan, daya listrik yang digunakan oleh *driver* adalah sebesar 4,6215 W dan digunakan oleh motor sebesar 4,0876 W. Sehingga dapat dicari nilai efisiensi rangkaian *driver* dari kondisi tanpa pembebahan adalah sebagai berikut:

$$\eta = \frac{P_o}{P_i} \times 100\%$$

$$\eta = \frac{4,0876 \text{ W}}{4,6215 \text{ W}} \times 100\%$$

$$\eta = 88,45 \%$$

Berdasarkan perhitungan diatas, dapat diketahui bahwa pada kondisi tanpa pembebahan dan kecepatan putar maksimal, nilai efisiensi dari rangkaian *driver* mencapai 88,45 %.

4.10.2 Perhitungan Kinerja Beban Ringan

Pada pengujian pembebahan ringan, daya listrik yang digunakan oleh *driver* adalah sebesar 5,697 W dan digunakan oleh motor sebesar 4,9396 W. Sehingga dapat dicari nilai efisiensi rangkaian *driver* dari kondisi tanpa pembebahan adalah sebagai berikut:

$$\eta = \frac{P_o}{P_i} \times 100\%$$

$$\eta = \frac{4,9396 \text{ W}}{5,697 \text{ W}} \times 100\%$$

$$\eta = 88,7 \%$$

Berdasarkan perhitungan diatas, dapat diketahui bahwa pada kondisi tanpa pembebahan dan kecepatan putar maksimal, nilai efisiensi dari rangkaian *driver* mencapai 88,7 %.

4.10.3 Perhitungan Kinerja Beban Sedang

Pada pengujian pembebahan sedang, daya listrik yang digunakan oleh *driver* adalah sebesar 6,1332 W dan digunakan oleh motor sebesar 5,4378 W. Sehingga dapat dicari nilai efisiensi rangkaian *driver* dari kondisi tanpa pembebahan adalah sebagai berikut:

$$\eta = \frac{P_o}{P_i} \times 100\%$$

$$\eta = \frac{5,4378 \text{ W}}{6,1332 \text{ W}} \times 100\%$$

$$\eta = 88,67 \%$$

Berdasarkan perhitungan diatas, dapat diketahui bahwa pada kondisi tanpa pembebahan dan kecepatan putar maksimal, nilai efisiensi dari rangkaian driver mencapai 88,67 %.

4.10.4 Perhitungan Kinerja Beban Berat

Pada pengujian pembebahan berat, daya listrik yang digunakan oleh *driver* adalah sebesar 6,785 W dan digunakan oleh motor sebesar 5,9712 W. Sehingga dapat dicari nilai efisiensi rangkaian *driver* dari kondisi tanpa pembebahan adalah sebagai berikut:

$$\eta = \frac{P_o}{P_i} \times 100\%$$

$$\eta = \frac{5,9712 \text{ W}}{6,785 \text{ W}} \times 100\%$$

$$\eta = 88 \%$$

Berdasarkan perhitungan diatas, dapat diketahui bahwa pada kondisi tanpa pembebahan dan kecepatan putar maksimal, nilai efisiensi dari rangkaian driver mencapai 88 %.

4.10.5 Perhitungan Kinerja Beban Sangat Berat

Pada pengujian pembebahan sangat berat, daya listrik yang digunakan oleh *driver* adalah sebesar 7,68 W dan digunakan oleh motor sebesar 6,3624 W. Sehingga dapat dicari nilai efisiensi rangkaian *driver* dari kondisi tanpa pembebahan adalah sebagai berikut:

$$\eta = \frac{P_o}{P_i} \times 100\%$$

$$\eta = \frac{6,3624 \text{ W}}{7,68 \text{ W}} \times 100\%$$

$$\eta = 82,84 \%$$

Berdasarkan perhitungan diatas, dapat diketahui bahwa pada kondisi tanpa pembebahan dan kecepatan putar maksimal, nilai efisiensi dari rangkaian driver mencapai 82,84 %.

Hasil perhitungan efisiensi menunjukkan nilai yang cukup baik, mengingat terdapat banyak eror pengujian, baik dari kualitas komponen yang digunakan maupun pembacaan dari sensor yang kurang akurat karena *switching* motor yang sangat cepat. Sehingga dari dapat diketahui bahwa untuk dapat mengoperasikan motor BLDC, *driver* motor mengonsumsi daya sebesar 4,62 W pada kondisi tanpa beban dan 7,68 W pada kondisi pembebahan maksimal yaitu 42,4 gram. Efisiensi dari rangkaian ini cukup baik dimana pada kondis tanpa pembebahan efisiensi rangkaian *driver* sebesar 88,45% dan pada kondisi pembebahan maksimal efisiensinya sebesar 82,84 %. Pengujian ini memperoleh hasil yang baik dimana motor dapat berputar dengan konsumsi daya yang relatif kecil dan memiliki nilai kecepatan yang cukup stabil.

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Pada penelitian ini telah dilakukan pembuatan drive motor BLDC dengan menggunakan IC IR2104, dengan menggunakan 12 Mosfet untuk melakukan *switching*. Pengujian dilakukan pada motor BLDC CD ROM dengan kondisi tanpa beban dan variasi pembebahan. Setelah dilakukan pengujian, dapat ditarik kesimpulan, yaitu pada penelitian ini telah berhasil dikembangkan sistem kontrol motor listrik BLCD dengan menggunakan metode kendali otomatis yang memberikan unjuk kerja yang optimal, efisien dan aman, dengan menggunakan motor BLDC 9,6 W yang dilengkapi dengan sensor hall serta dikendalikan dengan mikrokontroler Arduino dan dimonitor dengan *Blynk*. Prestasi yang dicapai adalah bahwa dengan penggunaan metode kendali otomatis, sistem ini mampu mengendalikan putaran motor dan menstabilkan kecepatan saat terjadi gangguan perubahan beban, rata rata dalam 30 detik hingga mencapai kecepatan *steady state*. Efisiensi daya sistem tanpa beban mencapai 88,45 % dan pada kondisi pembebahan sangat berat efisiensinya mencapai 82,84 %.

5.2 Saran

Berdasarkan hasil pengujian yang telah dilakukan, terdapat beberapa saran yang dapat diambil, antara lain:

1. Penggunaan komponen seharusnya menggunakan komponen dengan kondisi dan kualitas yang sama, agar hasil keluaran dari drive motor dapat sesuai dan stabil.
2. Variasi pembebahan sebaiknya dipilih dengan saksama agar hasil pengujian dapat lebih akurat.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik. 2017. Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis, 1949-2017. BPS Indonesia. Jakarta.
- [2] Syamsudin, A. 2012. Peraturan Menteri Negara Lingkungan Hidup Republik Indonesia Nomor 10 Tahun 2012. Berita Negara Republik Indonesia. No.788 (Revisi 2018).
- [3] Prabu, M.J., Poongodi, P., and Premkumar, K. 2016. Fuzzy supervised online coactive neuro-fuzzy inference system-based rotor position control of brushless DC motor. *IET Power Electronics Journal*. pp. 1-11
- [4] Aspalli, M.S. Farhat, M.M. Savitri, L.M. 2015. Speed control of BLDC Motor with Four Switch Three Phase Inverter using Digital Signal Controller. *2015 International Conference on Power and Advanced Control Engineering (ICPACE)*. Bangalore, India.
- [5] Ebadipour, M. Mohammad, B.B.S. Ebrahim, B. 2017. Modeling and synchronized control of dual parallel brushless direct current motors with single inverter. *Journal of Computers and Electrical Engineering*. 1-14
- [6] Rashid, M.H. 2004. *Power Electronics Circuit, Devices, And Applications 3rd Edition*, Pearson Education Inc, University of West Florida.
- [7] Sola E. B. 2016 “Design And Simulation Of A Single-Phase Inverter With Digital PWM”. *Bachelor Final Thesis*. Hochschule Niederrhein University of Applied Science. Germany
- [8] Nag, T., Santra, S.B., Chatterjee, A., and Chatterjee, D. 2016. Fuzzy logic-based loss minimisation scheme for brushless DC motor drive system. *IET Power Electronics Journal*. Vol. 9. Iss 8, pp 1581-1589.
- [9] Mohammad, A., Abedin, Md.A., and Khan, Md.Z.R. 2016. Implementation of a Three Phase Inverter for BLDC Motor Drive. *9th International Conference on Electrical and Computer Engineering*. Dhaka, Bangladesh
- [10] Ekmen, S. Bekir, F. dan Murat, I. 2016. A BLDC Motor Drive with Four Switch Three Phase Inverter. *5th International Conference on Renewable Energy Research and Application*. Birmingham, UK.
- [11] Karnap, S. 2018. Development of a Motor Controller for Electric Bicycles. *Project Advisor*. Instituto Superior de Engharia de Coimbra. Portugal
- [12] Catherine, P.J., Bala, M.R. 2015. Simulation Analysis of Power Quality Improvement in BLDC Motor Drive Using Type III CUK Derived Converter. *IEEE SPONSORED 2ND INTERNATIONAL CONFERENCE ON ELECTRONICS AND COMMUNICATION SYSTEM (ICECS 2015)*
- [13] Thirunavukkarasu, S. 2018. Performance Analysis of BLDC Motor Drive for Feed Drives. Proc. *IEEE Conference on Emerging Devices and Smart Systems (ICEDSS 2018)*.
- [14] Pindoriya, R.M. 2018. FPGA Based Digital Control Technique for BLDC

Motor Drive. *IEEE Journal.* 2/18

- [15] Affandy. 2019. DESAIN INVERTER SATU FASA DENGAN MONITORING DAYA MENGGUNAKAN ARDUINO. *Proposal Skripsi.* Porgram Studi Teknik Elektro, Fakultas Teknik, Universitas Sebelas Maret.
- [16] International Rectifier, “IRFZ44N Datasheet,” [Online]. Available: <https://www.alldatasheet.com/datasheet-pdf/pdf/68619/IRF/IRFZ44N.html> [Diakses 15 Juni 2020].
- [17] International Rectifier, “IR2104 Datasheet,” [Online]. Available: <https://www.alldatasheet.com/datasheet-pdf/pdf/86635/IRF/IR2104.html>. [Diakses 15 Juni 2020].
- [18] Ashok, R.B., and Mahesh, B.K. 2017. Comparative Analysis of BLDC motor for different control topology. *1st International Conference on Power Engineering, Computing and Control, PECCON.* VIT University. Chennai Campus
- [19] Deshpande, P. 2019. Power factor correction and power quality improvement in BLDC motor drive using SEPIC converter. *IEEE JournalI.* 9/19
- [20] Hendertshot, J. R., Miller, T. J. E. 1994. Design of Brushless Permanent Magnet Motors-Magna Physics Publications.
- [21] I. Efendi, “it-jurnal.com,” [Online]. Available: <https://www.it-jurnal.com/pengertian-dan-kelebihan-arduino/>. [Diakses 14 Mei 2019].
- [22] Martfuadi, R. 2019. Pengontrolan Motor DC Berbasis Arduino dengan Metode Hibrid Fuzzy Logic-PID pada Mini Conveyor. *Proposal Skripsi.* Porgram Studi Teknik Elektro, Fakultas Teknik, Universitas Sebelas Maret.
- [23] PWM, [Online]. Available: <https://arduino.cc/en/tutorial/PWM>. [Diakses 15 April 2020].
- [24] Arduino, [Online]. Available: <https://store.arduino.cc/usa/arduino>. [Diakses 20 April 2020]
- [25] Honeywell Inc., Hall Effect Sensing and Application, MICRO SWITCH Sensing and Control
- [26] Wilutomo, R.M.M. and Teguh, Y. 2017. Rancang Bangun Memonitor Arus dan Tegangan Serta Kecepatan Motor Induksi 3 Fasa Menggunakan Web Berbasis Arduino Due, GEMA TEKNOLOGI Vol. 19 No. 3
- [27] Kuphaldt, T.R., 2020, Series Circuit and Voltage Divider, Modular Electronics Learning (MODEL) Project
- [28] ACS712 Datasheet, 2006, Allegro MicroSystems Inc., Worcester, Massachusetts, USA.
- [29] Forsythe, J.B., Paralleling Of Power MOSFETs For Higher Power Output, *International Rectifier*, E1 Segundo, California.
- [30] Ogata, Katsuhiko. 1995. Terjemahan Teknik Kontrol Automatik (Sistem

Pengaturan) Jilid 1. Erlangga, Jakarta
[31] Blynk [Online]. Available: <https://blynk.io> [Diakses 20 Mei 2020].

LAMPIRAN

1. Program Arduino Kontrol Motor (Arduino 1)

```
-----KENDALI MOTOR BLDC-----
//                                         Tugas Akhir
//                                         Musyaffa' Ahmad / I0716026
//-----


//Kendali Kecepatan Motor BLDC dengan PWM
//PIN 2,3,4 terhung ke sensor hall
//PIN 9,10,11 terhubung ke kaki SD IC IR2104
//PIN 5,6,7 terhubung ke kaki IN IC IR2104
//PIN A0 terhubug ke output Potensiometer
//PIN A7 terhubung ke Arduino Uno D11

//FEEDBACK EROR=====
int feedback;
int setel=5000;
#define Proteksi A7

//Pendefinisian Pin=====
int sd1=5;
int in1=9;
int sd2=6;
int in2=10;
int sd3=11;
int in3=7;
//hall pin
int h1=4;
int h2=3;
int h3=2;
int hA,hB,hC;
int n=0; //step awal
int Pot,pwm,pwm1;

void setup() {
    Serial.begin(115200);

    //PENDEFINISIAN
    PIN=====
    pinMode(in1,OUTPUT);
    pinMode(sd1,OUTPUT);
    pinMode(in2,OUTPUT);
    pinMode(sd2,OUTPUT);
    pinMode(in3,OUTPUT);
    pinMode(sd3,OUTPUT);
    pinMode(h1,INPUT);
    pinMode(h2,INPUT);
    pinMode(h3,INPUT);
    digitalWrite(in1,LOW);
    digitalWrite(sd1,LOW);
    digitalWrite(in2,LOW);
    digitalWrite(sd2,LOW);
    digitalWrite(in3,LOW);
    digitalWrite(sd3,LOW);
}

void loop() {
//RUN
PROGRAM=====
```

```

hall_read();
hall_logic();
motor_run();
pot();

//SETEL SINYAL FEEDBACK EROR KECEPATAN=====
    feedback=analogRead(A7);
}

//PENDEFINISIAN PIN HALL=====
void hall_read(){
    hA=digitalRead(h1);
    hB=digitalRead(h2);
    hC=digitalRead(h3);
}

//LOGIKA PEMBACAAN HALL=====
void hall_logic(){
    if(hA==1&&hB==0&&hC==0){
        n=1;
    }
    else if(hA==0&&hB==1&&hC==0){
        n=2;
    }
    else if(hA==1&&hB==1&&hC==0){
        n=3;
    }
    else if(hA==0&&hB==0&&hC==1){
        n=4;
    }
    else if(hA==1&&hB==0&&hC==1){
        n=5;
    }
    else if(hA==0&&hB==1&&hC==1){
        n=6;
    };
    else if(Proteksi >= 1000 ){
        n=default;
    }
}

//STEP STEP SWITCHING=====
void motor_run(){
    switch(n){
        case 1:
            analogWrite(sd1,pwm1);//pwm
            digitalWrite(in1,HIGH);//1
            digitalWrite(sd2,LOW);//0
            digitalWrite(in2,LOW);//0
            digitalWrite(sd3,HIGH);//1
            digitalWrite(in3,LOW);//0
            break;
        case 2:
            digitalWrite(sd1,HIGH);//1
            digitalWrite(in1,LOW);//0
            analogWrite(sd2,pwm1);//pwm
            digitalWrite(in2,HIGH);//1
            digitalWrite(sd3,LOW);//0
            digitalWrite(in3,LOW);//0
            break;
        case 3:
            digitalWrite(sd1,LOW);//0
            digitalWrite(in1,LOW);//0
            analogWrite(sd2,pwm1);//pwm
}

```

```

digitalWrite(in2,HIGH);//1
digitalWrite(sd3,HIGH);//1
digitalWrite(in3,LOW);//0
break;
case 4:
digitalWrite(sd1,LOW);//0
digitalWrite(in1,LOW);//0
digitalWrite(sd2,HIGH);//1
digitalWrite(in2,LOW);//0
analogWrite(sd3,pwm1);//pwm
digitalWrite(in3,HIGH);//1
break;
case 5:
analogWrite(sd1,pwm1);//pwm
digitalWrite(in1,HIGH);//1
digitalWrite(sd2,HIGH);//1
digitalWrite(in2,LOW);//0
digitalWrite(sd3,LOW);//0
digitalWrite(in3,LOW);//0
break;
case 6:
digitalWrite(sd1,HIGH);//1
digitalWrite(in1,LOW);//0
digitalWrite(sd2,LOW);//0
digitalWrite(in2,LOW);//0
analogWrite(sd3,pwm1);//pwm
digitalWrite(in3,HIGH);//1
break;
default:
digitalWrite(in1,LOW);
digitalWrite(sd1,LOW);
digitalWrite(in2,LOW);
digitalWrite(sd2,LOW);
digitalWrite(in3,LOW);
digitalWrite(sd3,LOW);
break;
}
}

//PEMBACAAN POTENSIOMETER
void pot(){
//Jika pakai potensiometer
Pot=analogRead(A0);
pwm=map(Pot,0,1024,0,255);

//Jika pakai setpoint
//pwm = map(setel,0,10600,0,255);

pwm1 = pwm+feedback;

if (pwm1>=255){
pwm1=255;
}
}

```

2. Program Arduino Monitor Motor (Arduino 2)

```
//-----MONITORING KECEPATAN, ARUS DAN TEGANGAN-----  
  
//Dipasang di Arduino ke 2 (khusus pembacaan)  
//Hall sensor (satu aja) dihubungkan ke pin D3  
//Output sensor arus terhubung ke pin A4 Arduino 2  
//Output tegangan dihubungkan ke A1,A2 dan A3  
//GND Arduino 1 dihubungkan ke GND Arduino 2  
  
//FEEDBACK ERROR=====  
float pwmMotor,motorPwm2,error,error1,error2;  
#define feedback 11  
float umpanbalik;  
#define Proteksi 8  
  
//BLYNK=====  
#include <ArduinoJson.h>  
  
//BACA ARUS=====  
#include "ACS712.h"  
ACS712 sensor(ACS712_05B, A1);  
float Amps;  
  
//BACA TEGANGAN=====  
double vin;  
double Volt;  
double adc;  
#define tegangan A2  
  
//KONTROL KECEPATAN=====  
int setel; //disable jika pakai Setpoint kecepatan. Enable jika pakai potensio  
//int setel = 5000; //enable jika pakai Setpoint kecepatan. Disable jika pakai potensio  
int gas;  
#define ENC_COUNT_REV 3 //jumlah pulsa per rotasi  
#define ENC_IN 3 //pin hall yang akan diukur untuk kecepatan  
#define PWM 10 //pin pwm  
#define DIR 12 //pin dir  
int speedcontrol = 0; //pin analog potensio  
volatile long encoderValue = 0;// Pulse count from encoder  
int interval = 1000;//interval pengukuran (ms)  
// Counters for milliseconds during interval  
long previousMillis = 0;  
long currentMillis = 0;  
int rpm = 0; //nilai awal rpm  
int motorPwm = 0; //nilai awal pwm  
//=====  
void setup()  
{  
    Serial.begin(115200); // Setup Serial Monitor  
//FEEDBACK KE NANO=====  
    pinMode(feedback,OUTPUT);  
//KALIBRASI ARUS=====  
    sensor.calibrate();  
    delay(1000);  
    sensor.calibrate();  
//BACA TEGANGAN=====  
    pinMode (tegangan, INPUT);  
//PROTEKSI=====  
    pinMode (Proteksi, OUTPUT);  
//MONITOR KECEPATAN MOTOR=====  
    pinMode(ENC_IN, INPUT_PULLUP); //setel pin hall yang akan dimonitor  
    pinMode(PWM, OUTPUT); //setel pin pwm
```

```

pinMode(DIR, OUTPUT); //setel pin dir
attachInterrupt(digitalPinToInterrupt(ENC_IN), updateEncoder, RISING); //interrupt
previousMillis = millis(); // Setup initial values for timer
}

void loop()
{
//PI=====
kendali();
analogWrite(feedback, umpanbalik);
//BLYNK=====
// blynk();
//BACA ARUS=====
Arus();
//BACA TEGANGAN=====
tegangan();
//PROTEKSI=====
If (Volt == 15 && Amps==1){digitalwrite(Prpteksi,HIGH);}
//KONTROL KECEPATAN=====
//Control motor with setpoint. Enable jika pakai Setpoint kecepatan. Disable jika pakai potensio
//motorPwm =map(setel,0,10600,0,255); //Konversi RPM setpoint ke PWM.

//Control motor with potentiometer. //disable jika pakai Setpoint kecepatan. enable jika pakai potensio
motorPwm = map(analogRead(speedcontrol), 0, 1024, 0, 255); //disable jika pakai Setpoint kecepatan. enable jika pakai
potensio
setel = map(analogRead(speedcontrol), 0, 1024, 0, 11000); //disable jika pakai Setpoint kecepatan. enable jika pakai
potensio

gas = map (analogRead(motorPwm), 0, 255, 0, 100);
analogWrite(PWM, motorPwm); // Write PWM to controller

//Update RPM value every second
currentMillis = millis();
if (currentMillis - previousMillis > interval) {
previousMillis = currentMillis;

//Calculate RPM
rpm = ((float)(encoderValue * 60 / ENC_COUNT_REV));
motorPwm2= map(rpm,0,10600,0,255);

// Only update display when there is a reading
if (motorPwm > 0 || rpm > 0) {

//SERIAL MONITOR=====
// Serial.print("Gas : ");Serial.print(gas);Serial.println(" %");
// Serial.print("PWM : ");Serial.print(motorPwm);Serial.print(" ");
// Serial.print("Kecepatan : ");Serial.print(rpm);Serial.println(" rpm");
// Serial.print("Arus : ");Serial.println(Amps); //Serial.println(" A");
// Serial.print("Pi : ");Serial.println(Pi);
// Serial.print("Eror : ");Serial.println(error);
// Serial.print("setel : ");Serial.print(setel);Serial.print(" ");
// Serial.print("asli : ");Serial.println(rpm); // Serial.print(" ");
// Serial.print("tegangan: ");Serial.print(Volt); Serial.print(" ");

}

encoderValue = 0;
}
}

void updateEncoder()
{
// Increment value for each pulse from encoder
encoderValue++;
}

```

```

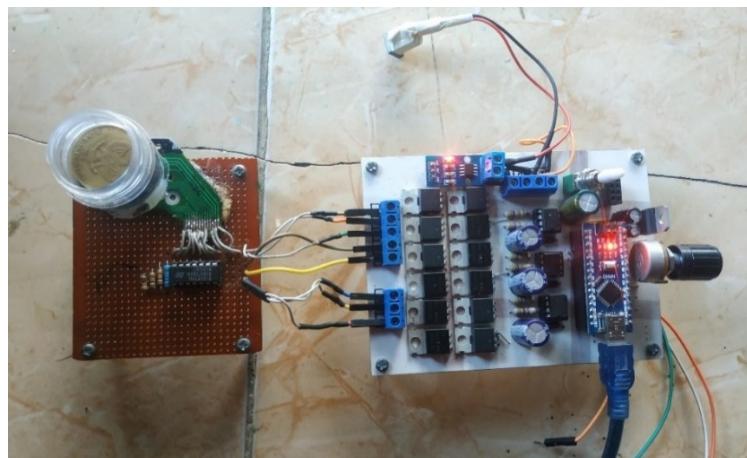
//BACA ARUS=====
void Arus(){
    // Get current from sensor
    Amps = sensor.getCurrentDC();
}
//BLYNK=====
void blynk(){
    StaticJsonDocument<200> doc;
    doc["gas"] = gas;
    doc["pwm"] = motorPwm;
    doc["kecepatan"] = rpm;
    doc["arus"] = Amps;
    doc["tegangan"] = Volt;
    serializeJson(doc, Serial);
}
//FEEDBACK=====
void kendali(){
    error1 = setel-rpm ;
    error = map(error1,0,10600,0,255);
    error2+=error;

    umpanbalik = error+error2;

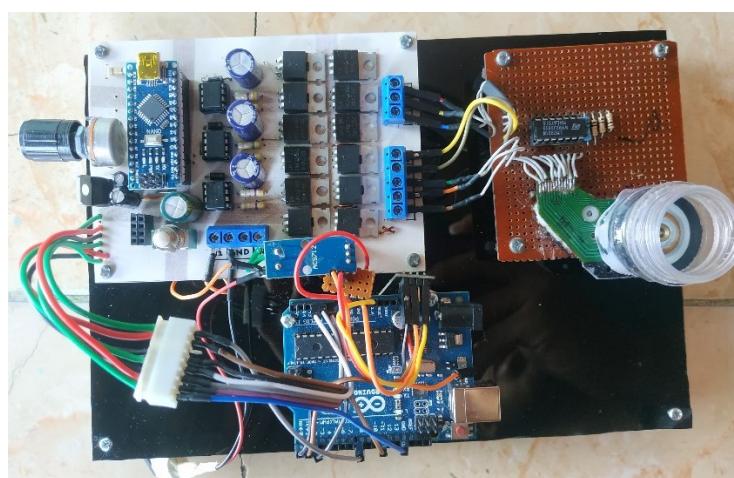
    if (umpanbalik <= 0){ umpanbalik=0; } //pembatasan nilai feedback minimal
    if(umpanbalik >= 255){ umpanbalik=255;} //pembatasan nilai feedback maksimal
}
//BACA TEGANGAN=====
void tegangan1(){
    adc = analogRead(tegangan);
    vin = adc/1023 * 5;
    Volt = vin*3;
}

```

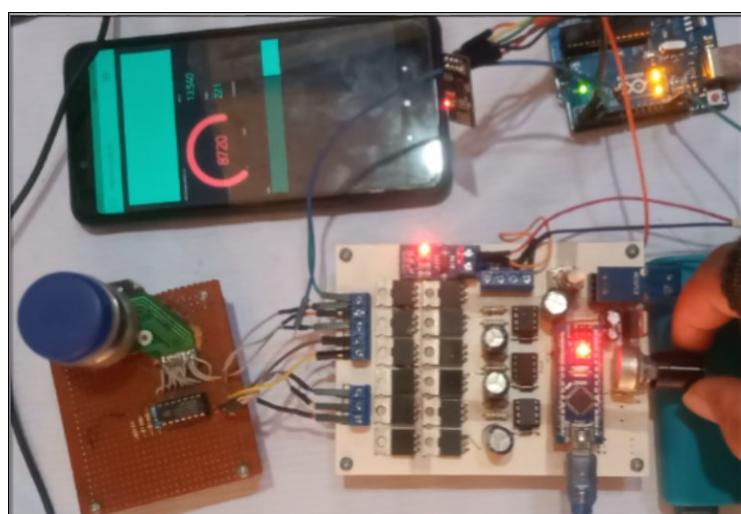
3. Dokumentasi Alat



Sistem Kendali *Open Loop* pada Motor BLDC



Sistem Kendali *Closed Loop* pada Motor BLDC



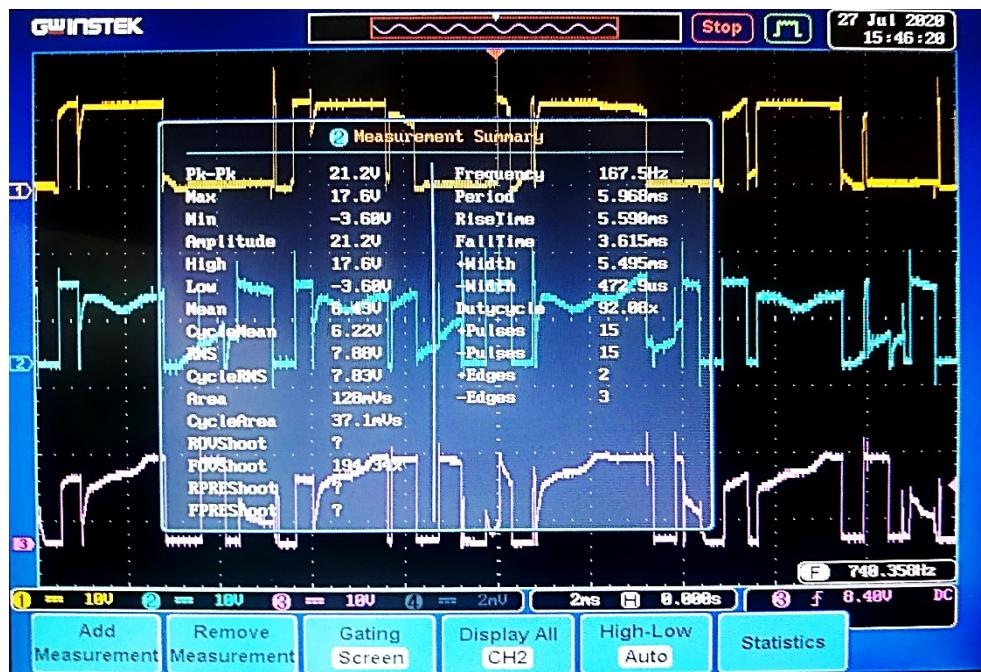
Pengujian *driver* Motor BLDC dengan monitoring *Blynk*



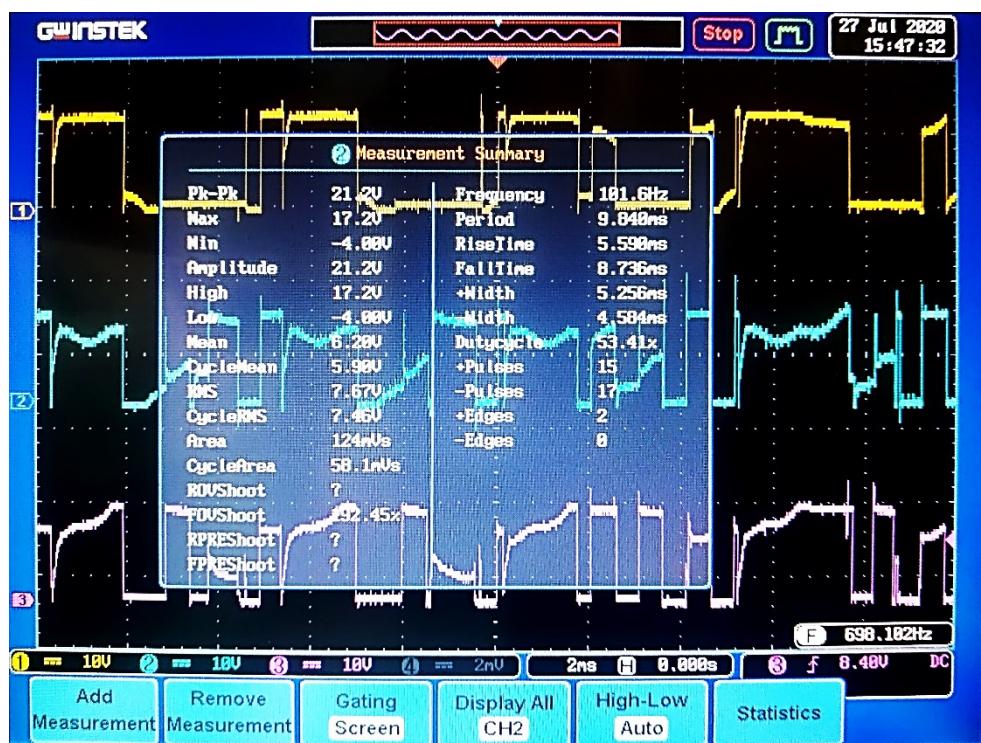
Pengukuran keluaran driver pada kecepatan 5.375 rpm tanpa beban



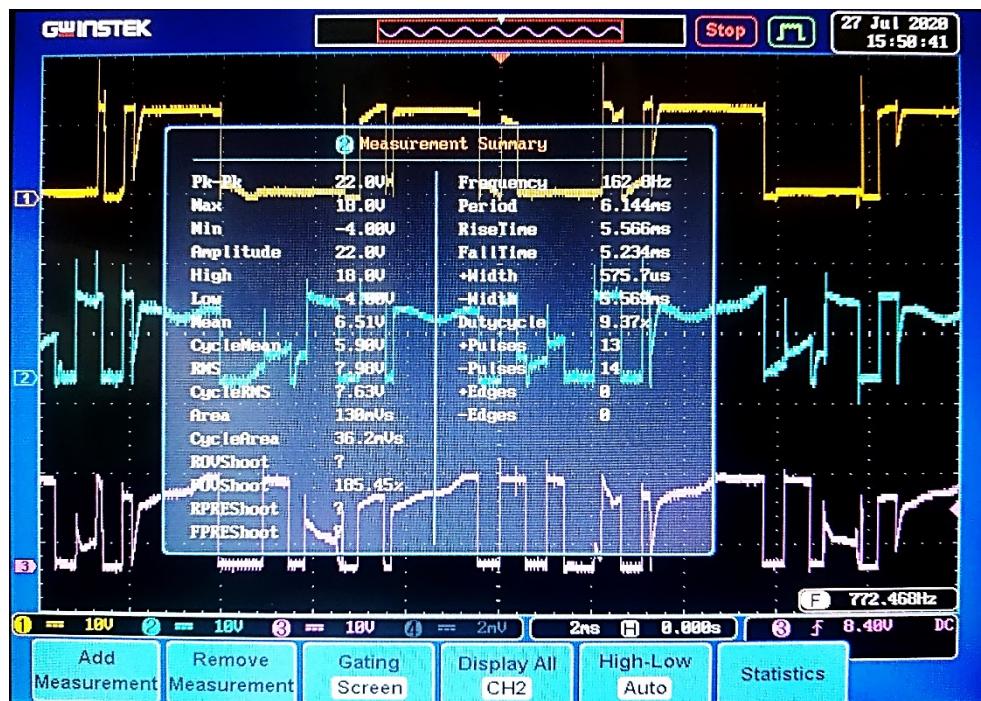
Pengukuran keluaran driver kecepatan 5.375 rpm beban ringan



Pengukuran keluaran driver kecepatan 5.375 rpm beban sedang



Pengukuran keluaran driver kecepatan 5.375 rpm beban berat



Pengukuran keluaran *driver* kecepatan 5.375 rpm beban sangat berat