



PROYEK AKHIR - VE 180626

MONITORING KECEPATAN MOTOR BLDC DENGAN BEBAN REM MAGNETIK

Bela Lorenza
NRP 10311500010003

Dosen Pembimbing
Ir. Josaphat Pramudijanto, M.Eng

Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2019

-----Halaman ini sengaja dikosongkan-----



FINAL PROJECT - VE 180626

***BLDC SPEED MONITORING USING MAGNETIC
BRAKE LOAD***

Bela Lorenza
NRP 10311500010003

Supervisor
Ir. Josaphat Pramudijanto, M.Eng

*Automation Electrical Engineering Department
Faculty of Vocation
Institut Teknologi Sepuluh Nopember
Surabaya 2019*

-----Halaman ini sengaja dikosongkan-----

PERNYATAAN KEASLIAN PROYEK AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya dengan judul **“Monitoring Kecepatan Motor BLDC dengan Beban Rem Magnetik”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Januari
2019

Bela Lorenza
NRP 10311500010003

-----Halaman ini sengaja dikosongkan-----

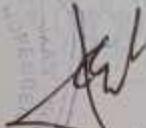
**MONITORING KECEPATAN MOTOR BLDC DENGAN
BEBAN REM MAGNETIK**

PROYEK AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Memperoleh Gelar Ahli Madya Teknik
Pada
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing



Ir. Josaphat Pramudijanto M.Eng
NIP. 19621005 199003 1 003

**SURABAYA
JANUARI, 2019**

-----Halaman ini sengaja dikosongkan-----

Monitoring Kecepatan Motor BLDC dengan Beban Rem Magnetik

Nama Mahasiswa : Bela Lorenza
NRP : 10311500010003
Pembimbing I : Ir. Josaphat Pramudijanto, M.Eng.
NIP : 19621005 199003 1 003

ABSTRAK

Saat ini dalam proses pembelajaran untuk mengetahui kecepatan dari motor *brushless* DC diperlukan sebuah alat ukur yang dapat memberikan data secara tepat. Untuk memonitoring kecepatan dari motor *brushless* DC diperlukan sebuah modul. Akan tetapi modul kontrol tersebut tidak mudah untuk dibuat.

Oleh karena itu untuk mempermudah proses pembelajaran, diperlukan suatu modul yang praktis dan efisien. Sehingga muncul ide untuk membuat suatu modul yang digunakan untuk memonitoring kecepatan motor *brushless* DC dengan mikrokontroler dan menggunakan metode *open loop*. Modul ini terdiri dari 4 komponen utama, yaitu arduino sebagai mikrokontroler, motor *brushless* DC, beban (rem magnetik), dan sensor kecepatan.

Pada Proyek akhir ini, dihasilkan bahwa motor *brushless* DC dapat dimonitoring kecepatannya dengan menggunakan *Rotary encoder* yang hasilnya ditampilkan pada LCD dan LabView. Dari hasil monitoring tersebut didapatkan hasil pengujian kecepatan dan didapatkan nilai pembacaan nilai *error* pada sensor *Rotary encoder* sebesar 22 %.

Kata Kunci : Motor *Brushless* DC, Rem Magnetik

-----Halaman ini sengaja dikosongkan-----

***BLDC Speed Control Monitoring Using Magnetic
Brake Load***

Student's Name : Bela Lorenza
Registration Number : 10311500010003
Supervisor I : Ir. Josaphat Pramudijanto, M.Eng.
ID Number : 19621005 199003 1 003

ABSTRACT

Currently in the learning process to find out the speed of a DC brushless motor, a measuring device is needed that can provide data correctly To monitor the speed of a DC brushless motor, a module is needed. However, the control module is not easy to make.

Therefore to facilitate the learning process, a module that is practical and efficient is needed. So the idea arises to make a module that is used to monitor the brushless DC motor speed with a microcontroller and use the open loop method. This module consists of 4 main components, namely Arduino as a microcontroller, brushless DC motor, load (magnetic brake), and speed sensor.

In this final project, the DC brushless speed can be monitored by using a Rotary encoder whose results are displayed on the LCD and Labview. From the results of the monitoring, the speed test results are obtained and the error value of the Rotary encoder is 22% .

Keywords : Brushless DC Motor, Magnetic Brake

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga Proyek Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Proyek Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Program Studi Elektronika Industri, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul

Monitoring Beban Kecepatan Motor BLDC dengan Beban Rem Magnetik

Proyek Akhir ini merupakan syarat untuk menyelesaikan mata kuliah dan memperoleh nilai pada Proyek Akhir. Dengan selesainya Proyek Akhir ini penulis menyampaikan terimakasih yang sebesar-besarnya kepada :

1. Kedua orangtua yang selalu memberi dukungan, doa, dan memberi perhatian kepada penulis.
2. Bapak Joko Susila, ST., MT selaku Ketua Departemen Teknik Elektro Otomasi, FV-ITS.
3. Bapak Ir. Josaphat Pramudijanto, M.Eng. selaku dosen pembimbing
4. Bapak Onie Meiyanto, S.Pd. selaku dosen pembimbing dari BLK
5. Semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Proyek Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Proyek Akhir ini. Akhir kata, semoga Proyek Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 2019

Penulis

-----Halaman ini sengaja dikosongkan-----

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan.....	1
1.3 Batasan Masalah.....	1
1.4 Tujuan	2
1.5 Sistematika Penulisan.....	2
1.6 Relevansi	3
BAB II TEORI PENUNJANG	5
2.1 Motor <i>Brushless</i> DC.....	5
2.1.1 Cara Kerja Motor <i>Brushless</i> DC	6
2.1.2 Konstruksi Motor <i>Brushless</i> DC	7
2.1.3 ESC (<i>Electronic Speed Controller</i>)	9
2.2 Sensor <i>Rotary Encoder</i>	11
2.3 Mikrokontroler Arduino MEGA 2560	12
BAB III PERANCANGAN DAN PEMBUATAN ALAT	15
3.1 Blok Fungsional Sistem	15
3.2 Perancangan Perangkat Keras (<i>Hardware</i>)	16
3.2.1 Perancangan Mekanik	16
3.2.2 Desain alat keseluruhan	16
3.2.3 Perancangan Elektrik	17
3.2.4 Rangkaian Sensor <i>Rotary Encoder</i>	17
3.2.5 Perancangan Pulley <i>Rotary Encoder</i>	18
3.3 Perancangan Perangkat Lunak (<i>Software</i>).....	20
3.3.1 Pemrograman <i>Software</i> Arduino	19
3.3.2 Perancangan Perangkat Lunak Labview	22
BAB IV PENGUJIAN DAN ANALISA	25
4.1 Pengujian Korelasi Sensor <i>Rotary Encoder</i>	25

4.1.1 Pengujian Kecepatan Motor 20% Pengereman	27
4.1.2 Pengujian Kecepatan Motor 40% pengereman	39
4.1.3 Pengujian Kecepatan Motor 60% pengereman	30
4.1.4 Pengujian Kecepatan Motor 80% pengereman	32
4.1.5 Pengujian Kecepatan Motor 90% pengereman	34
BAB V PENUTUP	37
5.1 Kesimpulan	37
5.2 Saran.....	37
DAFTAR PUSTAKA	39
LAMPIRAN A <i>LISTING</i> PROGRAM	A-1
LAMPIRAN B PROGRAM.....	B-1
DAFTAR RIWAYAT HIDUP	D-9

DAFTAR GAMBAR

Gambar 2.1	Salah Satu Contoh BLDC	5
Gambar 2.2	Skema Kerja Motor BLDC	6
Gambar 2.3	<i>Outer Construction</i>	7
Gambar 2.4	<i>Inner Construction</i>	8
Gambar 2.5	ESC	9
Gambar 2.6	Sinyal <i>Output</i> ESC 50%	10
Gambar 2.7	Sinyal <i>Output</i> ESC 75%	10
Gambar 2.8	Sinyal <i>Output</i> ESC 99%	10
Gambar 2.9	Sensor <i>Rotary Encoder</i>	11
Gambar 2.10	Arduino MEGA 2560.....	13
Gambar 3.1	Diagram Fungsional Alat	15
Gambar 3.2	Desain Modul Kontrol kecepatan Motor BLDC	16
Gambar 3.3	Rangkaian <i>Rotary Encoder</i>	17
Gambar 3.4	Dimensi <i>Rotary Encoder</i>	18
Gambar 3.5	<i>Pulley</i> Tampak Depan dan Tampak Samping	19
Gambar 3.6	<i>Flowchart</i> Arduino 2560.....	21
Gambar 3.7	<i>Flowchart</i> LabView	22
Gambar 3.8	Tampilan LabView	23
Gambar 3.9	Blok Diagram LabView.....	24
Gambar 4.1	Rangkaian Pengujian <i>Rotary Encoder</i>	25
Gambar 4.2	Hubungan Tegangan dan Kecepatan	26
Gambar 4.3	Hubungan 20% Pengereman	28
Gambar 4.4	Hubungan 40% Pengereman	30
Gambar 4.5	Hubungan 60% Pengereman	31
Gambar 4.6	Hubungan 80% Pengereman	33
Gambar 4.7	Hubungan 90% Pengereman	35

-----Halaman ini sengaja dikosongkan-----

DAFTAR TABEL

Tabel 2.1	Spesifikasi Motor <i>Brushless</i> DC Tipe <i>Outrunner</i> RCTimer A2830-13 850KV.....	6
Tabel 2.2	Deskripsi Arduino MEGA 2560	12
Tabel 4.1	Pengujian Sensor <i>Rotary Encoder</i>	26
Tabel 4.2	Pengujian Kecepatan Motor 20%	28
Tabel 4.3	Pengujian Kecepatan Motor 40%	30
Tabel 4.4	Pengujian Kecepatan Motor 60%	31
Tabel 4.5	Pengujian Kecepatan Motor 80%	33
Tabel 4.6	Pengujian Kecepatan Motor 90%	34

-----Halaman ini sengaja dikosongkan-----

BAB I

PENDAHULUAN

1.1. Latar Belakang

Aplikasi monitoring kecepatan banyak digunakan pada peralatan elektronik, dimana kecepatan dari peralatan elektronik tersebut harus dimonitoring sesuai dengan keinginan. Beberapa monitoring kecepatan banyak menggunakan motor salah satunya motor BLDC.

Selama ini monitoring mengenai motor *brushless* DC hanya sebatas monitoring kecepatan putaran motor yang dilakukan dengan berbagai metode kontrol. Namun masih belum ada alat atau modul monitoring kecepatan motor *brushless* DC yang khusus diperuntukkan untuk memonitoring kecepatan peralatan elektronik secara spesifik. Berdasarkan penelitian yang dilakukan oleh Lukman Hakim dkk, untuk memonitoring motor BLDC dibutuhkan sensor *Rotary Encoder* yang memiliki nilai yang akurat sehingga kecepatan yang akan di monitoring memiliki nilai yang tepat.

Oleh karena itu pada Proyek Akhir ini akan dibuat sebuah modul monitoring kecepatan motor *brushless* DC dengan menambahkan *Rotary encoder* Autonics agar kecepatan yang dihasilkan oleh motor dapat di monitoring dengan baik. Selain untuk mempermudah monitoring kecepatan diharapkan modul monitoring ini dapat digunakan sebagai sarana pembelajaran seperti praktikum.

1.2. Permasalahan

Dikarenakan belum banyak peralatan yang bisa digunakan untuk mengatur kontrol kecepatan motor BLDC.

1.3. Batasan Masalah

Dari perumusan masalah di atas, maka batasan masalah dari tugas akhir ini adalah :

1. Sensor yang digunakan sebagai pembaca kecepatan motor BLDC adalah sensor *Rotary encoder* sebanyak 1 buah sensor.
2. monitoring yang dilakukan hanya terbatas pada pengukuran kecepatan motor BLDC saja.

Dengan adanya batasan masalah ini diharapkan hasil akhir atau tujuan dari Proyek Akhir ini dapat dicapai dengan baik.

1.4 Tujuan

Tujuan dari proyek akhir ini adalah untuk membuat alat yang berfungsi untuk memonitoring kecepatan motor BLDC dengan menggunakan *Rotary encoder* yang mana kecepatan dari motor BLDC akan ditampilkan pada LCD dan LabView.

1.5 Sistematika Laporan

Pembahasan Proyek Akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, sistematika laporan, dan relevansi.

Bab II Teori Penunjang

Meliputi pembahasan dan teori-teori penunjang yang digunakan dalam mendukung terselesaikannya Proyek Akhir. Teori-teori yang dipakai dalam pembuatan proyek Tugas Akhir ini antara lain : motor *brushless* DC, sensor *rotary encoder*, pengereman magnetik, sensor arus, arduino MEGA 2560, *interfacing*, serta teori penunjang lainnya

Bab III Perancangan dan Pembuatan Alat

Pada Bab ini Membahas tentang tahap-tahap perancangan mekanik dan perancangan sistem kontrolernya.

Bab IV Pengujian dan Analisa

Bab ini membahas tentang pengukuran, pengujian, serta analisa terhadap prinsip kerja dan proses dari suatu alat yang dibuat.

Bab V Penutup

Bab ini menjelaskan tentang kesimpulan dari Priyek Akhir dan saran – saran untuk pengembangan alat ini lebih lanjut.

1.6 Relevansi

Proyek akhir ini diharapkan dapat memberikan manfaat bagi dunia pendidikan dan masyarakat terutama dalam pengembangansistem kontrol kecepatan motor BLDC.

-----Halaman ini sengaja dikosongkan-----

BAB II

TEORI PENUNJANG

2.1. Motor *Brushless* DC[1]

Salah satu jenis motor listrik yang paling banyak digunakan belakangan ini adalah motor *brushless* DC dimana motor DC ini tidak menggunakan brush (sikat) untuk proses komutasi. Motor *brushless* DC sangat cocok untuk diaplikasikan pada produk yang menuntut reliabilitas dan efisiensi yang tinggi. Dapat dikatakan bahwa motor *brushless* DC dapat menghasilkan torsi yang besar dan memiliki range RPM yang tinggi. *Brushless* DC Motor termasuk kedalam jenis motor sinkron. Artinya medan magnet yang dihasilkan oleh stator dan medan magnet yang dihasilkan oleh rotor berputar pada frekuensi yang sama. Motor BLDC tidak mengalami slip seperti yang terjadi pada motor induksi biasa. Motor jenis ini mempunyai magnet permanen pada bagian rotor dan elektromagnet pada bagian stator. Setelah itu, dengan menggunakan sebuah rangkaian sederhana (*simple computer system*), maka kita dapat merubah arus di elektromagnet ketika bagian rotornya berputar. Untuk lebih jelasnya mengenai motor ini dapat dilihat pada Gambar 2.1 dan Tabel 2.1.

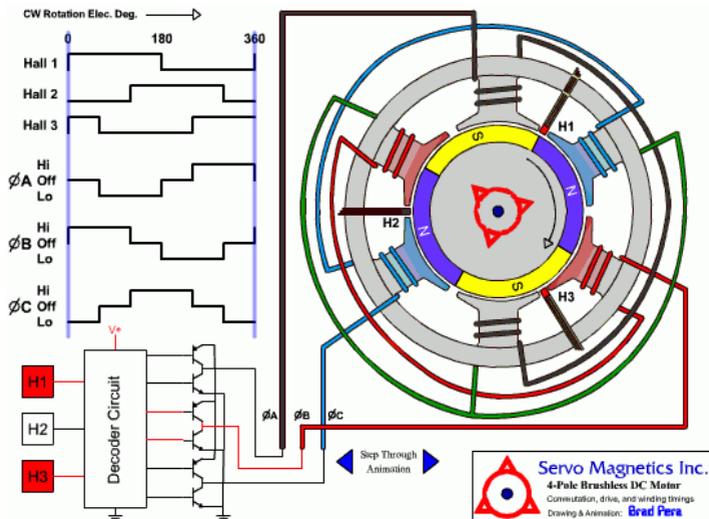


Gambar 2.1 Salah Satu Contoh BLDC[1]

Tabel 2.1 spesifikasi Motor Brushless DC Tipe A2212/13T
1000KV *Outrunner Brushless*

No	parameter	Nilai	
1.	Kecepatan putar	12580 rpm	
2.	Berat putar	52 gram	
3.	KV	1000 rpm/volt	
4.	Tegangan	Tegangan Minimal	7,40 volt
		Tegangan Maksimal	14,80 volt
5.	<i>Input</i> Arus	Arus Nominal	13 Ampere
		Arus Maksimal	25 Ampere
6.	<i>Maximal power</i>	187 watt	
7.	<i>Input</i> Baterai	Lithium polimer 2S-4S	

2.1.1 Cara Kerja *Brushless* DC [1]



Gambar 2.2 Skema Kerja Motor *Brushless* DC[1]

Cara kerja pada motor BLDC cukup sederhana, yaitu magnet yang berada pada poros motor akan tertarik dan terdorong oleh gaya elektromagnetik yang diatur oleh driver pada motor BLDC. Hal ini membedakan motor BLDC dengan motor DC yang menggunakan sikat mekanis yang berada pada komutator untuk mengatur waktu dan memberikan medan magnet pada lilitan. Motor BLDC ini juga berbeda dengan motor AC yang pada umumnya menggunakan siklus tenaga sendiri untuk mengatur waktu dan memberi daya pada lilitan. BLDC dapat memberikan rasio daya dan beban yang lebih tinggi secara signifikan dan memberikan efisiensi yang lebih baik dibandingkan motor tanpa sikat tradisional. Pada Gambar 2.3 adalah gambar gelombang dari motor BLDC.

2.1.2 Kontruksi Motor *Brushless* DC[1]

Setiap motor *brushless* DC memiliki dua bagian utama, rotor (bagian berputar) dan stator (bagian stasioner). Bagian penting lainnya dari motor adalah gulungan stator dan magnet rotor. Rotor adalah bagian pada motor yang berputar karena adanya gaya elektromagnetik dari stator, dimana pada motor *brushless* DC bagian rotornya berbeda dengan rotor pada motor DC konvensional yang hanya tersusun dari satu buah elektromagnet yang berada diantara *brushless* (sikat) yang terhubung pada dua buah motor hingga delapan pasang kutub magnet permanen berbentuk persegi panjang yang saling direkatkan menggunakan semacam “*epoxy*” dan tidak ada *brushes*-nya, seperti pada Gambar 2.3 .



Gambar 2.3 *Outer construction*[1]



Gambar 2.4 Inner construction[1]

Pada Gambar 2.4 adalah gambar stator. Stator adalah bagian pada motor yang diam/statis dimana fungsinya adalah sebagai medan putar motor untuk memberikan gaya elektromagnetik pada rotor sehingga motor dapat berputar. Pada motor *brushless* DC statornya terdiri dari 12 belitan (elektromagnet) yang bekerja secara elektromagnetik dimana statornya terhubung dengan tiga buah kabel untuk disambungkan pada rangkaian kontrol sedangkan pada motor DC konvensional statornya terdiri dari dua buah kutub magnet permanen. Belitan stator pada motor *brushless* DC terdiri dari dua jenis, yaitu belitan stator jenis trapezoidal dan jenis sinusoidal. Yang menjadi dasar perbedaan kedua jenis belitan stator tersebut terletak pada hubungan antara koil dan belitan stator yang bertujuan untuk memberikan EMF (*Electro Motive Force*) balik yang berbeda. EMF balik sendiri adalah tegangan balik yang dihasilkan oleh belitan motor *brushless* DC ketika motor tersebut berputar yang memiliki polaritas tegangan berlawanan arahnya dengan tegangan sumber yang dibangkitkan, sehingga besarnya EMF balik dapat dihitung dengan persamaan :

$$EMF \text{ balik} = B \cdot N \cdot l \cdot r \cdot \omega \quad \dots\dots\dots (2.1)$$

- di mana :
- B : kerapatan medan magnet yang dihasilkan rotor (Tesla)
- N : banyaknya lilitan pada belitan stator per fasa
- l : panjangnya batang rotor (m)
- R : jari-jari dalam motor (m)
- Ω : kecepatan sudut putaran motor (rad)

Perubahan besarnya EMF balik ini mempengaruhi torsi motor *brushless* DC, sebagaimana rumus torsi pada motor *brushless* DC

menurut persamaan diatas bahwa besarnya torsi yang dihasilkan motor *brushless* DC dapat dihitung dengan :

$$T = K_{rms} \cdot \Phi \cdot I \text{ (Nm)} \dots\dots\dots (2.2)$$

- di mana :
- K_{rms} : tegangan rata-rata konstan (Volt)
 - Φ : besarnya fluks magnet (Tesla)
 - I : besarnya arus (Ampere)
- (dimana $\omega=2\pi f$)

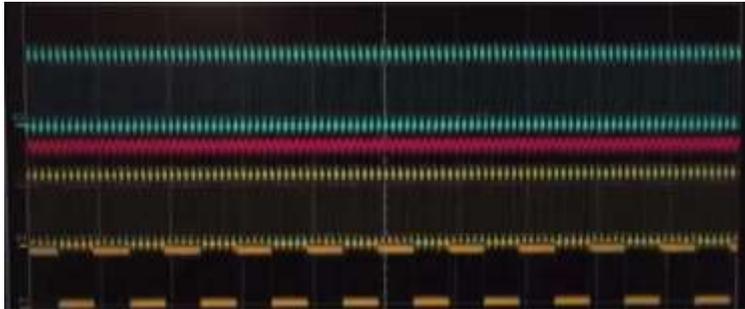
2.1.3 ESC (*Electronic Speed Controller*) [1]



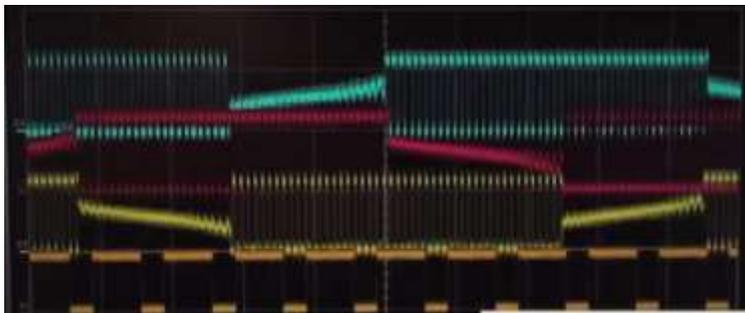
Gambar 2.5 ESC (*Electronic Speed Controller*)[1]

ESC adalah rangkaian elektronik yang berfungsi sebagai pengatur kecepatan putaran motor pada pesawat RC atau helikopter RC, cara kerjanya yaitu dengan cara menterjemahkan sinyal yang diterima *receiver* dari *transmitter*. Di pasaran terdapat berbagai merk ESC dengan kekuatan arus (*current rating*) dan kekuatan voltase (*voltage rating*) serta *feature* yang ditawarkan.

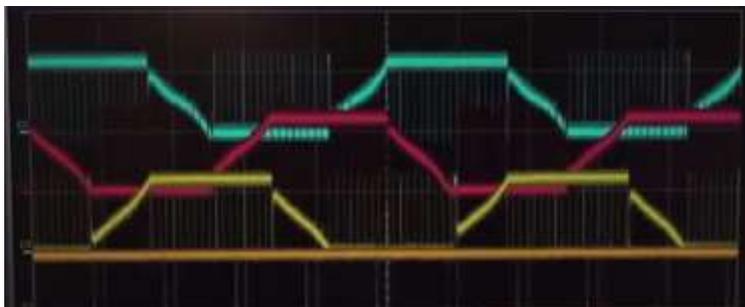
Untuk menentukan ESC yang akan kita gunakan sangatlah penting untuk mengetahui kekuatan (*peak current*) dari motor. Pilihlah ESC yang kekuatannya melebihi kekuatan motor. Misalnya, dari data kita dapatkan kekuatan motor adalah 12A (sesuai dengan *datasheet* motor) pada saat *throttle* terbuka penuh. sebaiknya ESC yang akan kita gunakan adalah ESC yang berkekuatan 18A atau 20A. Jika kita paksakan menggunakan ESC 10A kemungkinan pada saat *throttle* dibuka penuh, ESC akan panas bahkan terbakar.



Gambar 2.6 Sinyal *Output* ESC Ketika *Duty Cycle* 50% [1]



Gambar 2.7 Sinyal *Output* ESC Ketika *Duty Cycle* 75% [1]



Gambar 2.8 Sinyal *Output* ESC Ketika *Duty Cycle* 99% [1]

Pada Gambar 2.6, Gambar 2.7, dan Gambar 2.8 adalah gambar sinyal dari ESC. Sinyal warna biru, merah, dan kuning adalah sinyal *output* ESC. Sinyal warna oranye adalah sinyal *input* ESC. Terlihat perbedaan dari *Input* 50%, 75%, dan 99%, yaitu ketika input PWM dinaikkan maka sinyal *output* ESC semakin rapat, dan sebaliknya.

2.2 Sensor *Rotary Encoder* [2]



Gambar 2.9 *Rotary encoder*

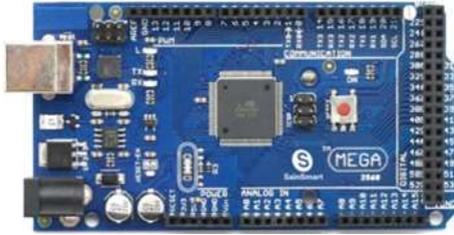
Rotary encoder digunakan untuk memonitor secara elektronik posisi poros yang berputar dengan mengubah rotasi poros menjadi pulsa elektronik. Pulsa *output encoder* dihitung dan dievaluasi oleh unit kontrol untuk menentukan posisi dan kecepatan mesin, yang memberikan akurasi dan fleksibilitas luar biasa saat mengendalikan gerakan. Kecepatan operasi dan kecepatan sistem yang tinggi mengurangi jumlah siklus dan meningkatkan produktivitas secara keseluruhan di lingkungan manufaktur. Autonics *rotary encoder* dalam berbagai ukuran dan tipe untuk menawarkan kualitas tinggi dan pengukuran yang tepat dari posisi operasi dan kecepatan poros berputar, dll untuk kontrol yang lebih tepat dari objek.

2.3 Arduino MEGA 2560[2]

Arduino MEGA 2560 adalah *board* arduino yang merupakan perbaikan dari *board* arduino MEGA sebelumnya. Arduino MEGA awalnya memakai *chip* ATmega1280 dan kemudian diganti dengan *chip* ATmega2560, oleh karena itu namanya diganti menjadi arduino MEGA 2560. Tabel 2.2 merupakan spesifikasi arduino MEGA 2560.

Tabel 2.2 Deskripsi Arduino MEGA 2560

Mikrokontroler	RISC ATmega 2560
<i>Operating Voltage</i>	5 V
<i>Input Voltage (recommended)</i>	7 – 12 V
<i>Input Voltage (limit)</i>	6 – 20 V
<i>Digital I/O Pins</i>	54 (15 diantaranya <i>input</i> PWM)
<i>Analog Input Pins</i>	16
<i>DC Current per I/O Pin</i>	40 mA
<i>DC Current for 3,3V Pin</i>	50 mA
<i>Flash Memory</i>	256 KB (8 KB sebagai <i>bootloader</i>)
SRAM	8 KB
EEPROM	4 KB
<i>Clock Speed</i>	16 MHz
<i>USB Host Chip</i>	MAX3421E
<i>Length</i>	101,52 mm
<i>Width</i>	53,3 mm
<i>Weight</i>	37 g



Gambar 2.10 Arduino MEGA 2560[2]

-----Halaman ini sengaja dikosongkan-----

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

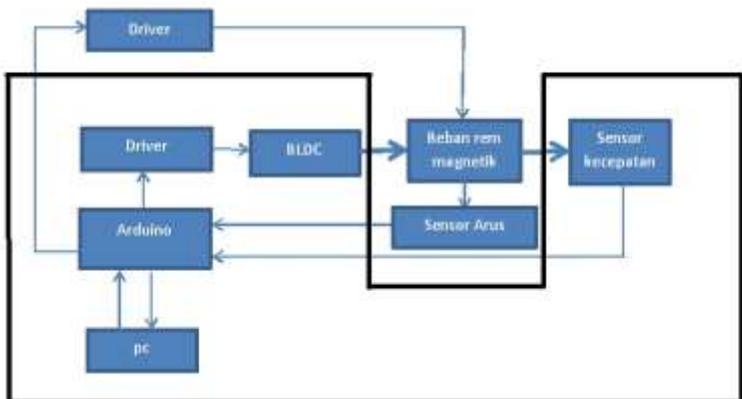
Pada bab ini akan dibahas mengenai perancangan dan pembuatan alat “Monitoring Kecepatan Motor BLDC dengan Beban Rem Magnetik”. Penjelasan diawali dengan penjelasan diagram fungsional sistem secara keseluruhan, kemudian perancangan perangkat keras dan diakhiri perencanaan perangkat lunak. Perancangan perangkat keras (*hardware*). ”. Perancangan alat akan dibahas perbagian yang disertai dengan gambar skematik.

Untuk memudahkan dalam pembahasan bab ini akan dibagi menjadi 3 yaitu:

1. Blok fungsional system
2. Perancangan perangkat keras yang terdiri dari perancangan mekanik dan elektrik, yaitu:
 - a. Perancangan mekanik meliputi :
 1. Desain Alat Secara Keseluruhan
 - b. Perancangan elektrik meliputi :
 1. Rangkaian Sensor *Rotary Encoder*

3.1. Diagram Fungsional Sistem

Perancangan sistem secara keseluruhan dalam pembuatan alat yang digunakan dapat diperlihatkan pada Gambar 3.1.



Gambar 3.1 Diagram Fungsional Alat Keseluruhan

Dari Gambar 3.1 alat ini dirancang dengan komponen-komponen sebagai berikut ; Arduino, *driver*, BLDC, sensor kecepatan,dan LCD.Sistem kerja dari alat ini adalah dari arduino akan memberikan informasi pada *driver* dan *driver* akan menggerakkan BLDC setelah itu dari BLDC akan menuju pada sensor kecepatan,sensor kecepatan akan mendeteksi kecepatan pada BLDC setelah itu masuk pada arduino dan akan ditampilkan pada LCD. Pengukuran kecepatan dapat langsung dikontrol dan dilihat melalui LCD.

3.2 Perancangan Perangkat Keras (*Hardware*)

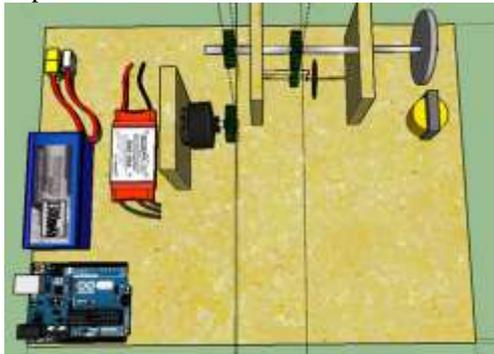
Pada perancangan hardware pada tugas akhir ini yang dibahas terdiri perancangan perangkat keras, prosesnya dibagi menjadi 2 bagian, yaitu perancangan mekanik dan elektrik. Masing-masing perancangan tersebut selanjutnya akan dibahas lebih mendalam pada sub bab berikutnya

3.2.1 Perancangan Mekanik

Perancangan mekanik ini terdiri dari desain alat secara keseluruhan, motor *brushless* DC, rem magnetik, *rotary encoder*, dan *driver* motor *brushless* DC.

3.2.2 Desain Alat Secara Keseluruhan

Gambar 3.2 akan menyatakan bagaimana desain modul kontrol kecepatan motor *brushless* DC secara keseluruhan.



Gambar 3.2 Desain Modul Kontrol Kecepatan Motor *Brushless* DC

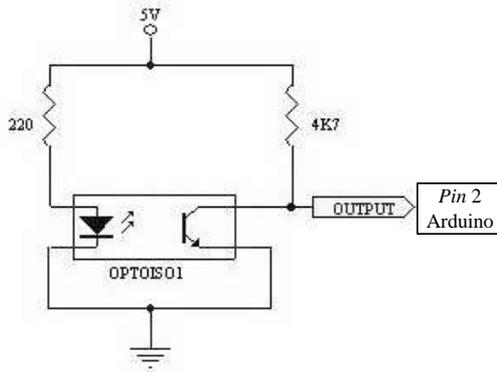
3.2.3 Perancangan Elektrik

Perancangan elektronik ini meliputi desain layout PCB serta pengkabelan. Rangkaian elektrik pada *plant* ini meliputi rangkaian sensor *rotary encoder*, *driver* rem magnetik, dan arduino MEGA 2560.

3.2.4 Rangkaian Sensor *Rotary Encoder*[2]

Rotary encoder pada tugas akhir ini akan digunakan untuk menentukan banyaknya putaran poros tiap menit (RPM) yang kemudian akan menghasilkan gelombang kotak yang frekuensinya akan bertambah bila kecepatan putar poros bertambah. *Rotary encoder* ini diletakkan pada poros yang sudah dikopel dengan motor *brushless DC*.

Pulsa *output encoder* dihitung dan dievaluasi oleh unit kontrol untuk menentukan posisi dan kecepatan mesin, yang memberikan akurasi dan fleksibilitas luar biasa saat mengendalikan gerakan. Kecepatan operasi dan kecepatan sistem yang tinggi mengurangi jumlah siklus dan meningkatkan produktivitas secara keseluruhan di lingkungan manufaktur. Autonics *rotary encoder* dalam berbagai ukuran dan tipe untuk menawarkan kualitas tinggi dan pengukuran yang tepat dari posisi operasi dan kecepatan poros berputar, dll untuk kontrol yang lebih tepat dari objek

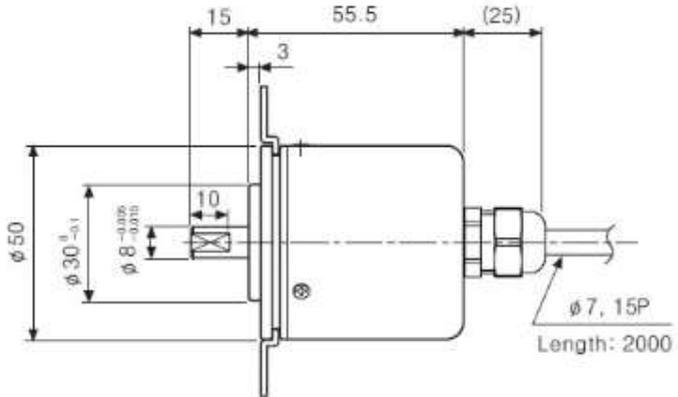


Gambar 3.3 Rangkaian *Rotary Encoder*[2]

3.2.5 Perancangan *Pulley Rotary Encoder*

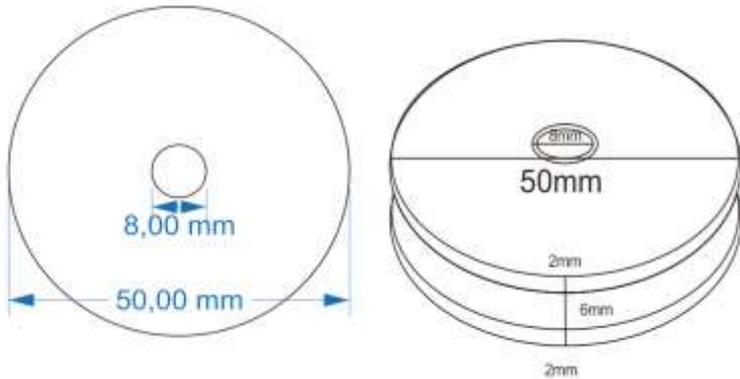
Pada perancangan *pulley rotary encoder*, *Rotary Encoder* yang digunakan memiliki tipe *shaft incremental* (lihat Gambar 3.5), sehingga dibutuhkan *pulley* sebagai alat bantu, dimana *pulley* tersebut akan dipasang pada *van belt* dengan poros tabung. Hal ini bertujuan untuk mendeteksi putaran (rpm) pada beban. Dikarenakan dimensi *shaft rotary encoder* yang kecil (dengan diameter 8 mm) maka diperlukan perancangan *pulley* yang sesuai dengan *shaft* tersebut.

Pulley yang dipakai, terbuat dari plastik jenis *nylon 6*. Hal ini dikarenakan *nylon 6* termasuk *plastic tipe thermoplast* yang memiliki sifat keras (tahan terhadap pukulan tinggi), bersifat isolasi, dan tahan terhadap suhu tinggi pada pengoperasian yang kontinyu.



Gambar 3.4 Dimensi *Rotary Encoder Tipe Shaft Incremental*
Tampak Samping

Dengan *shaft rotary encoder* sebesar 8 mm, maka diperlukan *pulley* untuk diletakkan di *shaft* sensor. Perancangan *pulley* untuk *rotary encoder* akan terlihat seperti Gambar 3.5.



Gambar 3.5 *Pulley* Tampak Depan dan Tampak Samping

3.3 Perancangan Perangkat Lunak (*Software*)

Agar rem elektromagnetik dapat dikontrol dan hasil putaran dapat dilihat di LCD maka perlu dirancang di sebuah *software* yang mampu mengelola dan mengontrol data terhadap kinerja peralatan rem elektromagnetik. *Software* merupakan program berisi perintah-perintah yang dieksekusi oleh arduino MEGA 2560 sehingga sistem dapat bekerja sesuai dengan alur dan tujuan.

3.3.1 Pemrograman *Software* Arduino

Dalam perancangan program pada *software* arduino dengan fungsi terkait yang dibutuhkan diperlukan beberapa tahapan yang harus dilakukan terlebih dahulu. Tahapan tersebut adalah membuat algoritma dari alat yang sudah kita jalankan.

Pembuatan algoritma ini dilakukan setelah membuat *flowchart*, dari algoritma kita ini maka diharapkan alat yang akan dibuat ini dapat terlebih lebih sederhana. Setelah tahapan tersebut terselesaikan barulah kita memprogram fungsi terkait yang dikodingkan dalam bahasa C.

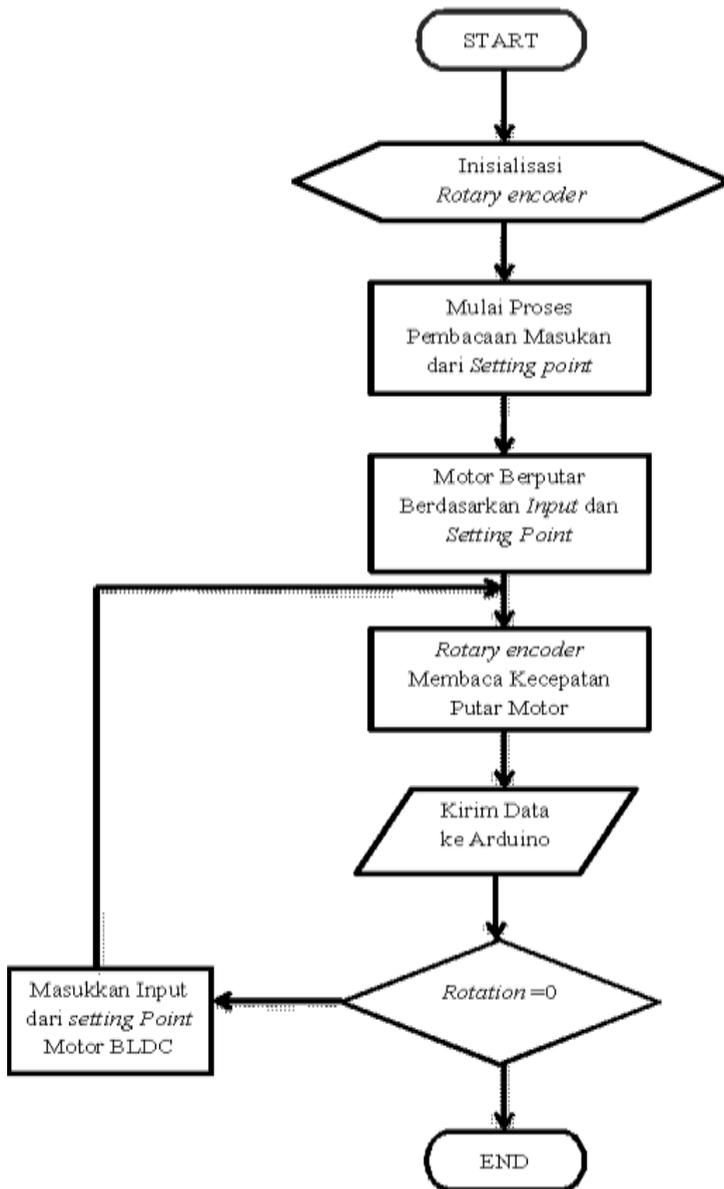
Berikut ini algoritma program utama dari rem elektromagnetik:

1. Modul dapat dioperasikan setelah sistem terpasang dengan benar seperti motor *brushless* DC beserta rem magnetik, *driver* motor *brushless* DC, *driver* rem magnetik, catu daya dan *potensiometer*.

2. Modul dapat bekerja jika rangkaian kontrol sudah dijalankan dan sudah terpasang.
3. *Potentiometer* digunakan untuk mengatur *range* yang akan di suplai ke *driver* rem elektromagnetik yang berupa tegangan dan akan di ubah menjadi digital melalui ADC Arduino.
4. *Driver* rem elektromagnetik akan mengatur tegangan yang masuk ke rem magnetik yang berfungsi untuk memperlambat putaran motor

Pada Tugas Akhir ini *software* yang digunakan adalah program arduino MEGA 2560 untuk membuat dan merencanakan program dalam bahasa C. Pemrograman *software* arduino dirancang dengan menggunakan *software* yang bernama Arduino IDE dengan menggunakan bahasa pemrograman C.

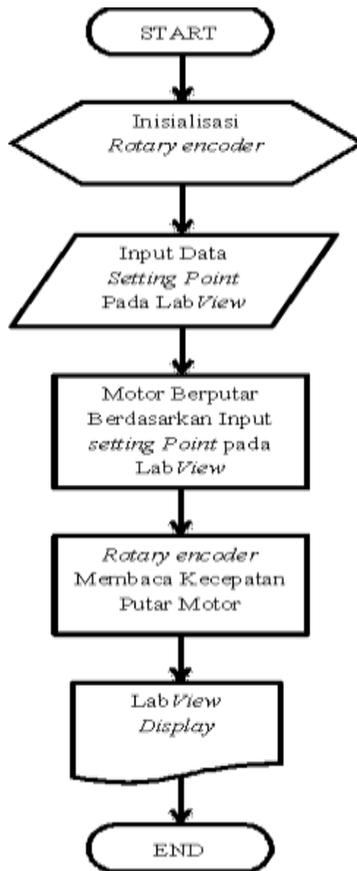
Arduino sangatlah berbeda sekali dengan mikrokontroler. Arduino merupakan sebuah *kit* mikrokontroler AVR yang dibuat dalam sebuah *board* (papan PCB). Dikembangkan di Italia sejak tahun 2005. Dalam 1 *board* sudah terdapat mikrokontroler lengkap dengan *pin/port* untuk koneksi serta sudah dilengkapi dengan *downloader*. Dalam segi bahasa pemrograman, arduino memiliki bahasa pemrograman yang lebih mudah dan sederhana terutama bagi pemula, alasan bahasa pemrograman arduino lebih mudah dan sederhana adalah karena didalam arduino sudah terdapat beberapa *library* yang dapat digunakan untuk merancang pemrograman yang diinginkan. Pada Gambar 3.6 berikut menunjukkan *flowchart* yang digunakan pada Arduino MEGA 2560 pada Proyek akhir ini.



Gambar 3.6 Flowchart Arduino MEGA 2560

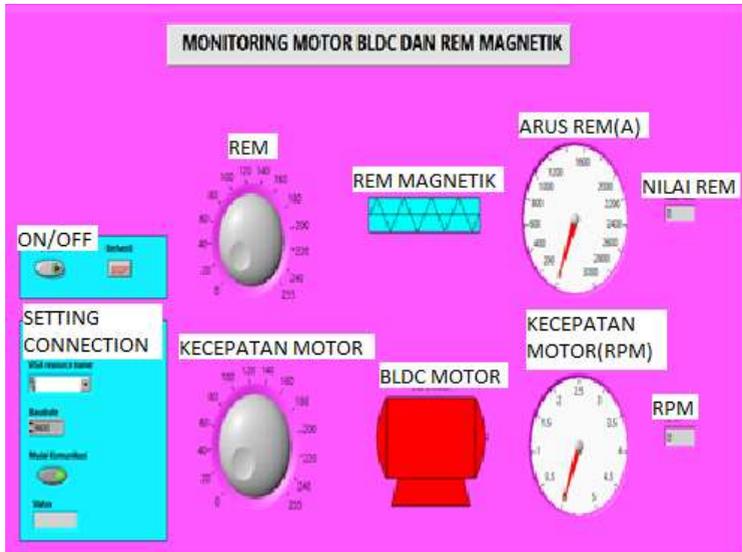
3.3.2 Perancangan Perangkat Lunak LabView

Ketika awal memulai *software* LabView, tahapan yang dilakukan adalah dengan membuka *File Menu* kemudian klik *New VI*. *Untitled VI* akan mempunyai dua bagian yang disebut *Front Panel* dan *Block Diagram*. *Front Panel Window* merupakan *user interface window* pada saat VI dijalankan, sedangkan *Block Diagram Window* merupakan bagian yang akan melakukan kalkulasi dan mengeksekusi algoritma. Gambar 3.7 menunjukkan *state diagram* yang difungsikan pada LabView untuk pengaturan kecepatan pada motor *brushless DC*.



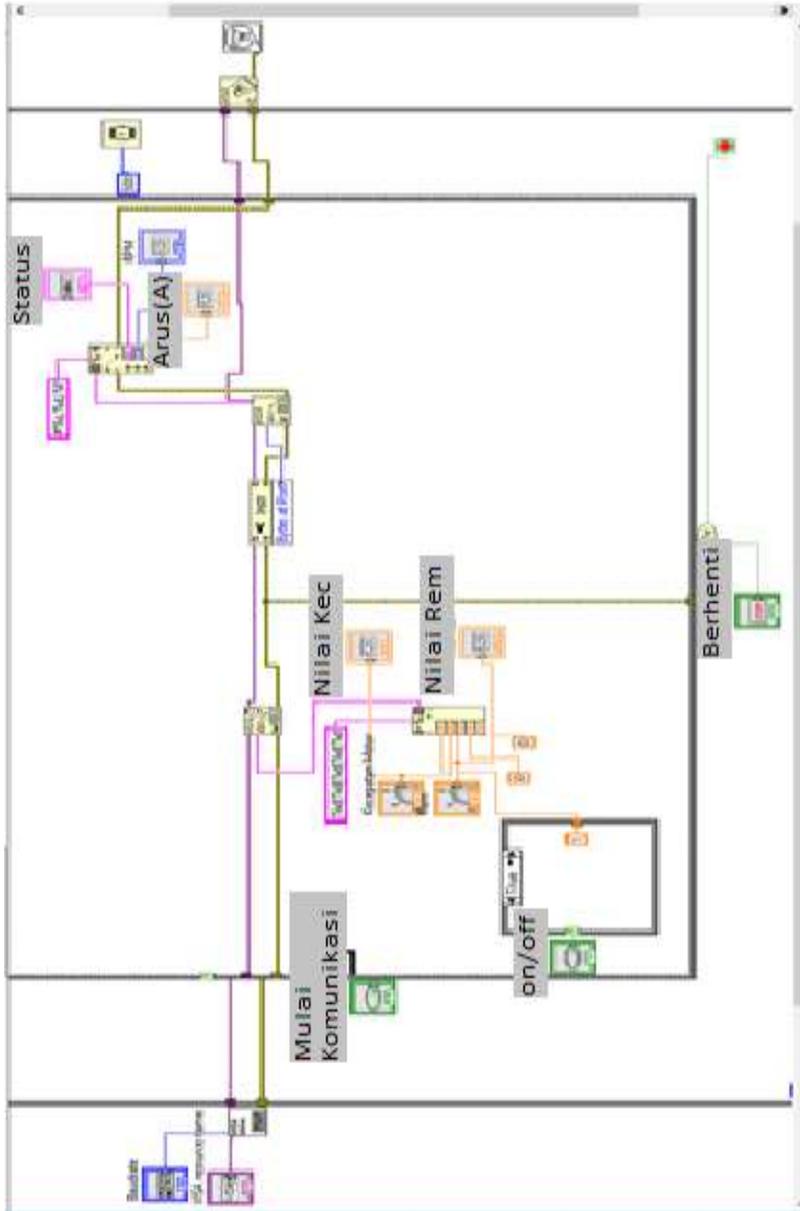
Gambar 3.7 Flowchart LabView

Gambar 3.7 menjelaskan bagaimana proses dari LabView bekerja pada sistem monitoring motor *brushless* DC. Pada LabView terdapat tampilan yang berfungsi untuk memonitoring kecepatan motor *brushless* berikut ini pada Gambar 3.8 adalah contoh dari tampilan LabView yang digunakan untuk memonitoring kecepatan motor *brushless* DC.



Gambar 3.8 Tampilan pada LabView

Pada tampilan Gambar 3.6 terdapat potensiometer kecepatan motor dan rem yang berfungsi untuk mengatur kecepatan dari motor *brushless* DC dan rem magnetik, untuk membuat tampilan LabView seperti diatas harus terlebih dahulu dibuat block diagram LabView seperti pada Gambar 3.9 .

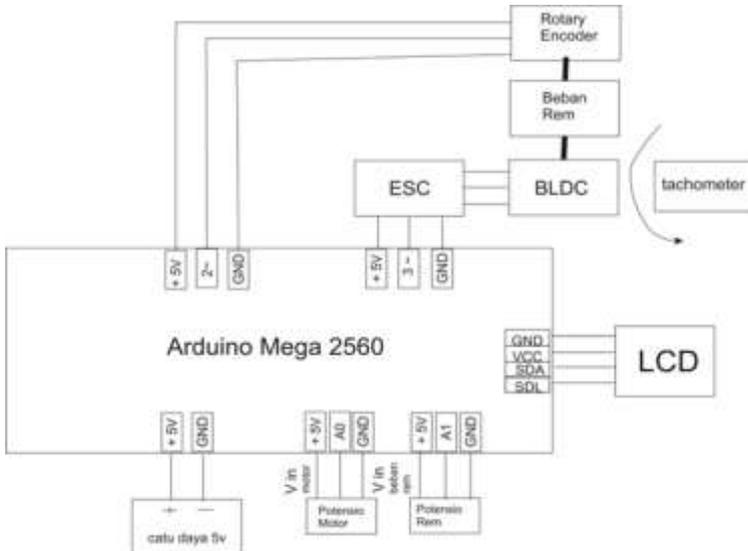


Gambar 3.9 Block Diagram LabView

BAB IV PENGUJIAN DAN ANALISA

4.1 Pengujian Korelasi Sensor *Rotary Encoder* dan *Tachometer*

Pada pengujian sensor *rotary encoder* dilakukan dengan perbedaan pembacaan antara kecepatan yang diukur oleh sensor *rotary encoder* dari program yang dibuat dengan *digital tachometer* buatan pabrik. Hal ini dilakukan untuk mengkalibrasi sensor *rotary encoder*. Untuk pengujian sensor, caranya yaitu dengan menjalankan motor *brushless* DC yang sudah terpasang dengan ESC yang diputar menggunakan potensiometer dengan sensor *rotary encoder* yang sudah terpasang dan terkopel dengan motor *Brushless* DC. Kemudian membandingkan hasil pembacaan kecepatan yang terbaca oleh sensor dengan hasil yang terbaca pada *digital tachometer*. Program dari pengujian ini menggunakan Arduino yang telah diprogram. Dalam pembacaan sensor *Rotary encoder* juga dibutuhkan rangkaian uji. Gambar 4.1 adalah rangkaian pengujian sensor *Rotary encoder*.

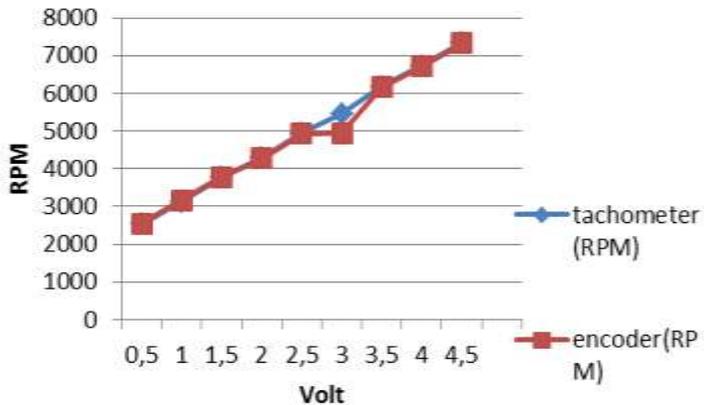


Gambar 4.1 Rangkaian Pengujian Sensor *Rotary Encoder*

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.1 maka dapat dihasilkan Tabel 4.1 dengan hasil sebagai berikut.

Tabel 4.1 Pengujian Sensor *Rotary Encoder*

No	Vin Motor Volt	Tachometer RPM	Encoder RPM
1	0,5	2522	2549
2	1	3136	3168
3	1,5	3758	3782
4	2	4297	4282
5	2,5	4953	4932
6	3	5460	4949
7	3,5	6174	6156
8	4	6729	6706
9	4,5	7352	7335
10	5	8976	8964



Gambar 4.2 Hubungan Antara Tegangan dan Kecepatan

Pada pengujian yang tertera pada Tabel 4.1 dapat dilihat bahwa *output* hasil pembacaan pada sensor *rotary encoder* dan *digital tachometer* memiliki perbedaan. Nilai korelasi antara *tachometer* dan nilai *encoder* yang telah dihitung nilai korelasinya adalah 0,9 dan nilai korelasi tersebut sangat kuat. Perbedaan

kecepatan ini semakin besar pada saat kecepatan tinggi. Hal ini disebabkan karena getaran dari poros motor yang sudah tidak lurus lagi yang menyebabkan getaran rotary *encoder* saat diputar pada kecepatan tinggi.

4.1.1 Pengujian Kecepatan Motor Dengan 20% Pengereman

Pada pengujian kecepatan motor dengan 20% pengereman, pengukuran kecepatan motor *brushless* DC dikonfigurasi dengan beban motor (rem magnetik). Pengujian ini dilakukan ketika rem magnetik berada dalam kondisi 20% Pengereman. 20% pengereman didapatkan dari $\frac{1}{4}$ putaran dari potensiometer.

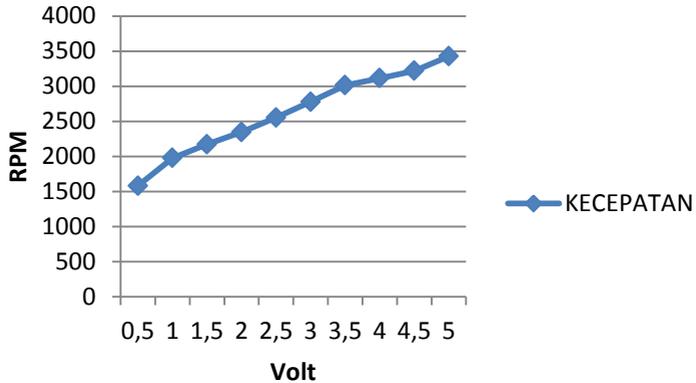
Pengujian ini dilakukan dengan menggunakan *input* dari potensiometer yang nilainya di *mapping* menjadi nilai *throttle* dari tegangan 0,5 V sampai 5 V.

Cara yang digunakan agar pengereman bisa mencapai 20% pengereman adalah dengan memutar potensiometer yang telah dipasang dan diprogram pada *driver* rem magnetik sebesar 20%. Program dari pengujian ini menggunakan Arduino yang telah diprogram, selain menggunakan arduino yang sudah diprogram pada pengujian ini juga menggunakan *tachometer digital* yang digunakan untuk mengukur kecepatan motor secara manual agar hasilnya dapat dibandingkan dengan hasil pengujian kecepatan motor yang dihasilkan oleh *Rotary encoder*, kecepatan motor yang dihasilkan oleh perhitungan *Rotary encoder* yang telah diprogram di arduino akan ditampilkan pada LCD. Gambar potongan program inisialisasi arduino terdapat pada Gambar 4.6 dan gambar rangkaian pengujian kecepatan motor terdapat pada Gambar 4.8 .

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.2 maka dapat dihasilkan Tabel 4.2 dengan hasil sebagai berikut.

Tabel 4.2 Pengujian Kecepatan Motor (20% Pengereman)

No	Vin motor Volt	Encoder RPM
1	0,5	1580
2	1	1977
3	1,5	2170
4	2	2346
5	2,5	2552
6	3	2777
7	3,5	3012
8	4	3114
9	4,5	3219
10	5	3426



Gambar 4.3 Hubungan Antara Tegangan dan Kecepatan pengereman 20%

Berdasarkan pengujian yang telah dilakukan dengan hasil yang didapat sesuai Tabel 4.3 diatas, maka didapatkan kesimpulan bahwa pengaturan kecepatan motor *brushless* DC dapat dilakukan dengan mengatur besar tegangan input potensiometer dengan nilai perbedaan ketika motor tanpa beban rem magnetik dan dengan pengereman motor 20% didapatkan persentase rata-rata nilai error kecepatan motor adalah 27% .

4.1.2 Pengujian Kecepatan Motor Dengan 40% Pengereman

Pada pengujian kecepatan motor dengan 40% pengereman, pengukuran kecepatan motor *brushless* DC dikonfigurasi dengan beban motor (rem magnetik). Pengujian ini dilakukan ketika rem magnetik berada dalam kondisi 40% Pengereman. 40% pengereman didapatkan dari 2/5 putaran dari potensiometer.

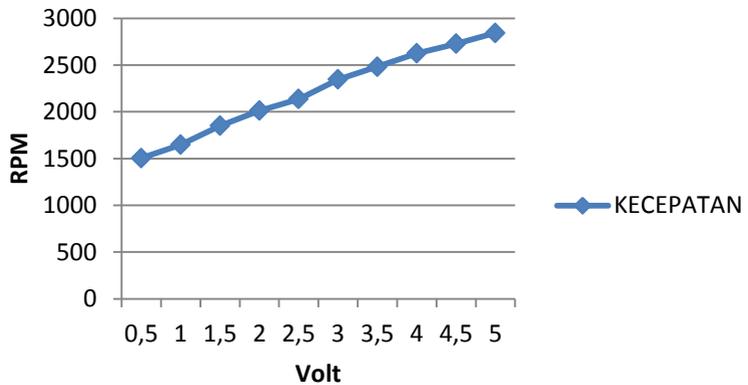
Pengujian ini dilakukan dengan menggunakan *input* dari potensiometer yang nilainya di *mapping* menjadi nilai *throttle* dari tegangan 0,5 V sampai 5 V.

Cara yang digunakan agar pengereman bisa mencapai 40% pengereman adalah dengan memutar potensiometer yang telah dipasang dan diprogram pada *driver* rem magnetik sebesar 40%. Program dari pengujian ini menggunakan Arduino yang telah diprogram, selain menggunakan arduino yang sudah diprogram pada pengujian ini juga menggunakan *tachometer digital* yang digunakan untuk mengukur kecepatan motor secara manual agar hasilnya dapat dibandingkan dengan hasil pengujian kecepatan motor yang dihasilkan oleh *Rotary encoder*, kecepatan motor yang dihasilkan oleh perhitungan *Rotary encoder* yang telah diprogram di arduino akan ditampilkan pada LCD. Gambar potongan program inisialisasi arduino terdapat pada Gambar 4.6 dan gambar rangkaian pengujian kecepatan motor terdapat pada Gambar 4.8 .

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.2 maka dapat dihasilkan Tabel 4.3 dengan hasil sebagai berikut.

Tabel 4.3 Pengujian Kecepatan Motor (40% Pengereman)

No	Vin motor Volt	Encoder RPM
1	0,5	1504
2	1	1648
3	1,5	1850
4	2	2011
5	2,5	2135
6	3	2346
7	3,5	2481
8	4	2624
9	4,5	2727
10	5	2842



Gambar 4.4 Hubungan Antara Tegangan dan Kecepatan pengereman 40%

Berdasarkan pengujian yang telah dilakukan dengan hasil yang didapat sesuai Tabel 4.3 diatas, maka didapatkan kesimpulan bahwa pengaturan kecepatan motor *brushless* DC dapat dilakukan dengan mengatur besar tegangan input potensiometer dengan nilai perbedaan ketika motor tanpa beban rem magnetik dan dengan pengereman motor 40% didapatkan hasil didapatkan persentase rata-rata nilai error kecepatan motor adalah 31% .

4.1.3 Pengujian Kecepatan Motor Dengan 60% Pengereman

Pada pengujian kecepatan motor dengan 60% pengereman, pengukuran kecepatan motor *brushless* DC dikonfigurasi dengan beban motor (rem magnetik). Pengujian ini dilakukan ketika rem magnetik berada dalam kondisi 60% pengereman. Pengereman 60% didapatkan dari 1,5 putaran dari potensiometer.

Pengujian ini dilakukan dengan menggunakan *input* dari potensiometer yang nilainya di *mapping* menjadi nilai *throttle* dari tegangan 0,5 V sampai 5 V.

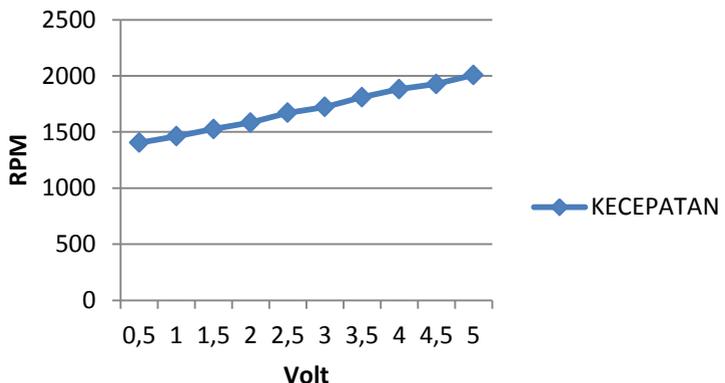
Cara yang digunakan agar pengereman bisa mencapai 60% pengereman adalah dengan memutar potensiometer yang telah dipasang pada *driver* rem magnetik sebesar 60%. Program dari pengujian ini menggunakan Arduino yang telah diprogram, selain menggunakan arduino yang sudah diprogram pada pengujian ini juga menggunakan *tachometer digital* yang digunakan untuk mengukur kecepatan motor secara manual agar hasilnya dapat dibandingkan

dengan hasil pengujian kecepatan motor yang dihasilkan oleh *Rotary encoder*. kecepatan motor yang dihasilkan oleh perhitungan *Rotary encoder* yang telah diprogram di arduino akan ditampilkan pada LCD. Gambar potongan program inisialisasi arduino terdapat pada Gambar 4.6 dan gambar rangkaian pengujian kecepatan motor terdapat pada Gambar 4.8

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.2 maka dapat dihasilkan Tabel 4.4 dengan hasil sebagai berikut.

Tabel 4.4 Pengujian Kecepatan Motor (Dengan 60% Pengereman)

No	Vin Motor Volt	Encoder RPM
1	0,5	1404
2	1	1461
3	1,5	1526
4	2	1583
5	2,5	1671
6	3	1722
7	3,5	1810
8	4	1881
9	4,5	1928
10	5	2007



Gambar 4.5 Hubungan Antara Tegangan dan Kecepatan pengereman 60%

Berdasarkan pengujian yang telah dilakukan dengan hasil yang didapat sesuai Tabel 4.4 diatas, maka didapatkan kesimpulan bahwa pengaturan kecepatan motor *brushless* DC dapat dilakukan dengan mengatur besar tegangan input potensiometer dengan nilai perbedaan ketika motor tanpa beban rem magnetik dan dengan pengereman motor 60% didapatkan hasil persentase rata-rata nilai error kecepatan motor adalah 37% .

4.1.4 Pengujian Kecepatan Motor Dengan 80% Pengereman

Pada pengujian kecepatan motor dengan 80% pengereman, pengukuran kecepatan motor *brushless* DC dikonfigurasi dengan beban motor (rem magnetik). Pengujian ini dilakukan ketika rem magnetik berada dalam kondisi 80% pengereman. .Pengereman 80% didapatkan dari $\frac{3}{4}$ putaran dari potensiometer.

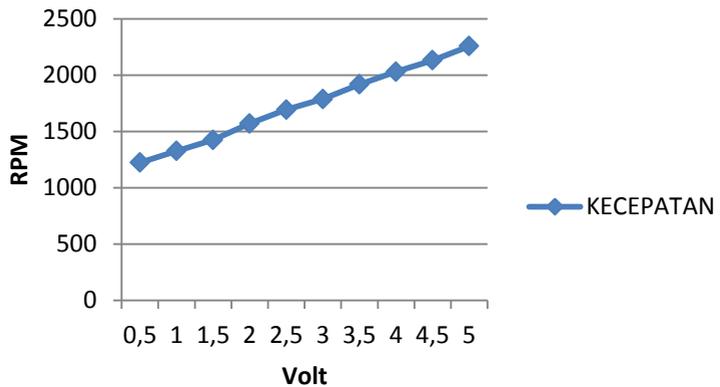
Pengujian ini dilakukan dengan menggunakan *input* dari potensiometer yang nilainya di *mapping* menjadi nilai *throttle* dari tegangan 0,5 V sampai 5 V.

Cara yang digunakan agar pengereman bisa mencapai 80% pengereman adalah dengan memutar potensiometer yang telah dipasang pada *driver* rem magnetik sebesar 80%. Program dari pengujian ini menggunakan Arduino yang telah diprogram, selain menggunakan arduino yang sudah diprogram pada pengujian ini juga menggunakan *tachometer digital* yang digunakan untuk mengukur kecepatan motor secara manual agar hasilnya dapat dibandingkan dengan hasil pengujian kecepatan motor yang dihasilkan oleh *Rotary encoder*. kecepatan motor yang dihasilkan oleh perhitungan *Rotary encoder* yang telah diprogram di arduino akan ditampilkan pada LCD. Gambar potongan program inisialisasi arduino terdapat pada Gambar 4.6 dan gambar rangkaian pengujian kecepatan motor terdapat pada Gambar 4.8 .

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.2 maka dapat dihasilkan Tabel 4.5 dengan hasil sebagai berikut.

Tabel 4.5 Pengujian Kecepatan Motor (Dengan 80% Pengereman)

No	Vin Motor Volt	Encoder RPM
1	0,5	1222
2	1	1326
3	1,5	1424
4	2	1570
5	2,5	1693
6	3	1787
7	3,5	1917
8	4	2030
9	4,5	2131
10	5	2256



Gambar 4.6 Hubungan Antara Tegangan dan Kecepatan pengereman 80%

Berdasarkan pengujian yang telah dilakukan dengan hasil yang didapat sesuai Tabel 4.5 diatas, maka didapatkan kesimpulan bahwa pengaturan kecepatan motor *brushless* DC dapat dilakukan dengan mengatur besar tegangan input potensiometer dengan nilai perbedaan ketika motor tanpa beban rem magnetik dan dengan pengereman motor 80% didapatkan hasil persentase rata-rata nilai error kecepatan motor adalah 43% .

4.1.5 Pengujian Kecepatan Motor Dengan 90% Pengereman

Pada pengujian kecepatan motor dengan 90% pengereman, pengukuran kecepatan motor *brushless* DC dikonfigurasi dengan beban motor (rem magnetik). Pengujian ini dilakukan ketika rem magnetik berada dalam kondisi 90% pengereman. Pengereman 90% didapatkan dari hampir satu putaran dari potensiometer.

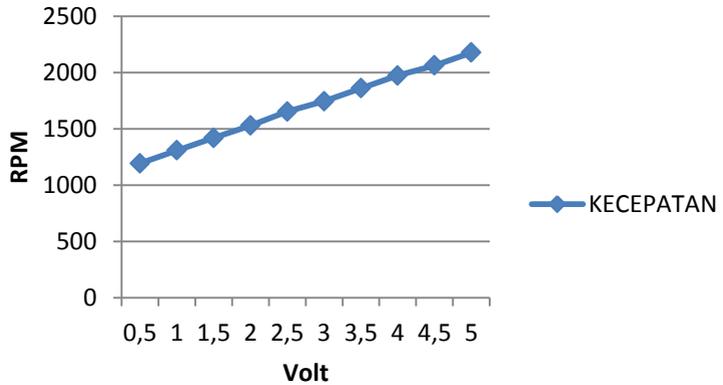
Pengujian ini dilakukan dengan menggunakan *input* dari potensiometer yang nilainya di *mapping* menjadi nilai *throttle* dari tegangan 0,5 V sampai 5 V.

Cara yang digunakan agar pengereman bisa mencapai 90% pengereman adalah dengan memutar potensiometer yang telah dipasang pada *driver* rem magnetik sebesar 90%. Program dari pengujian ini menggunakan Arduino yang telah diprogram, selain menggunakan arduino yang sudah diprogram pada pengujian ini juga menggunakan *tachometer digital* yang digunakan untuk mengukur kecepatan motor secara manual agar hasilnya dapat dibandingkan dengan hasil pengujian kecepatan motor yang dihasilkan oleh *Rotary encoder*. Kecepatan motor yang dihasilkan oleh perhitungan *Rotary encoder* yang telah diprogram di arduino akan ditampilkan pada LCD. Gambar potongan program inisialisasi arduino terdapat pada Gambar 4.6 dan gambar rangkaian pengujian kecepatan motor terdapat pada Gambar 4.2 .

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.8 maka dapat dihasilkan Tabel 4.6 dengan hasil sebagai berikut.

Tabel 4.6 Pengujian Kecepatan Motor (Dengan 90% Pengereman)

No	Vin Motor Volt	Encoder RPM
1	0,5	1192
2	1	1309
3	1,5	1419
4	2	1528
5	2,5	1652
6	3	1744
7	3,5	1860
8	4	1973
9	4,5	2062
10	5	2177



Gambar 4.7 Hubungan Antara Tegangan dan Kecepatan pengereman 90%

Berdasarkan pengujian yang telah dilakukan dengan hasil yang didapat sesuai Tabel 4.6 diatas, maka didapatkan kesimpulan bahwa pengaturan kecepatan motor *brushless* DC dapat dilakukan dengan mengatur besar tegangan input potensiometer dengan nilai perbedaan ketika motor tanpa beban rem magnetik dan dengan pengereman motor 90% didapatkan hasil persentase rata-rata nilai *error* kecepatan motor adalah 53% .

Dengan Hasil analisa yang didapat sesuai dengan Tabel 4.1- Tabel 4.6 dapat disimpulkan bahwa didapatkan nilai *error* pada sensor Rotary encoder sebesar 22%.

-----Halaman ini sengaja dikosongkan-----

BAB V

PENUTUP

Setelah melakukan perancangan dan pembuatan alat serta pengujian dan analisa, maka dapat ditarik kesimpulan dan saran dari kegiatan yang telah dilakukan untuk pengembangan Proyek Akhir ini.

5.1. Kesimpulan

Dari seluruh tahapan yang sudah dilaksanakan pada penyusunan Tugas Akhir ini, mulai dari studi literatur, perancangan dan pembuatan sampai pada pengujiannya maka dapat disimpulkan bahwa :

1. Motor BLDC (*brushless* DC) dapat dimonitoring kecepataannya dengan menggunakan *Rotary encoder* yang hasilnya ditampilkan pada LCD dan *LabView*.
2. Dari pembacaan sensor *Rotary encoder* didapatkan nilai error pada sensor *Rotary encoder* sebesar 22%.

5.2. Saran

Untuk lebih memperbaiki dan menyempurnakan kinerja dari alat ini, maka perlu disarankan :

1. Agar pembacaan sensor maksimal dan nilainya lebih presisi dibutuhkan sensor yang sering digunakan sehingga memudahkan dalam penggunaan dan pembuatan program untuk sensor itu sendiri.
2. Keseluruhan alat baik penyangga maupun tatanan lebih baik dibuat dari bahan yang kuat dan tahan terhadap getaran, agar sewaktu dijalankan tidak mengganggu kecepatan putar

-----Halaman ini sengaja dikosongkan-----

DAFTAR PUSTAKA

- [1] Djohar,Syamsi, "Pemanfaatan Motor BLDC", *Jurnal*, Pusat Penelitian Bandung, 2012.
- [2] Intan Nur Robi Annisa dan Zaka Perwira, "Pembuatan Modul Kontrol Kecepatan Motor Brushless DC dengan Mikrokontroler", *Tugas Akhir*, Program D3 Teknik Elektro FTI-ITS, Surabaya, 2016.
- [3] Hakim,Lukman, "Implementasi Perhitungan Posisi Robot Dengan FPGA Menggunakan Rotary Encoder ", *Tugas Akhir*, Jurusan Teknik Elektro,Institut Teknologi Sepuluh Nopember, Surabaya,2012
- [4] Suhartono, "Optimasi Parameter Kontroler PID Berbasiskan Particle Swarm Optimization untuk Pengendalian Kecepatan Motor Induksi 3 Fase", *Tugas Akhir*, Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember,Surabaya,2012.
- [5]*Pengertian Motor BLDC dan Fungsinya*, URL: <http://teknikelektronika.com/pengertian-BLDC-fungsi-BLDC/>, Diakses Pada Tanggal 18 April 2018
- [6]*Posisi Kontrol Motor BLDC*, URL: <http://wangready.com/kontrol>, Diakses Pada Tanggal 20 April 2018
- [7]*Produk Arduino Mega 2560*, URL: <http://famosastudio.com/2013>, Diakses Pada Tanggal 25 April 2018

-----Halaman ini sengaja dikosongkan-----

LAMPIRAN A PROGRAM

```
#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>

Servo esc;
int PWM;
int nilai1;
int nilai;
int persent;
double arus_temporary=0.0;
float adc_Volt, cal_value,temp;
unsigned long waktu_kalibrasi=0, kalibrasi=600;
boolean calibration=false;
LiquidCrystal_I2C lcd(0x27,2,1,0,4,5,6,7);
byte potPin = 0;

byte graph [8][8] =
{
  { 0x00, 0x03, 0x05, 0x19, 0x11, 0x19, 0x05, 0x03 }, // simbol
    speaker
  { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0x00 }, // 1 bar
  { 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00 }, // 2 bar
  { 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00 }, // 3 bar
  { 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00 }, // 4 bar
  { 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00 }, // 5 bar
  { 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00 }, // 6 bar
  { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00 } // 7 bar
};

void setup ()
{
  esc.attach(3);
```

```

pinMode (9, OUTPUT);
lcd.begin(20, 4);
lcd.setBacklightPin(3, POSITIVE);
lcd.setBacklight(HIGH);
Serial.println("sistem dimulai");
Serial.print("waktu kalibrasi:");
Serial.println(kalibrasi);
Serial.begin(9600);
for (int a = 0; a < 8; a++)
    lcd.createChar(a, graph[a]); // pembuatan karakter khusus bar
    volume
}

void loop ()
{
    data_olah();
    delay(10);
    int readADC = analogRead(potPin);
    int bar = map(readADC, 0, 1023, 1, 8);           // perbandingan
        readADC dengan jumlah bar
    int vol = map(readADC, 0, 1023, 0, 100);
    int PWM = analogRead(potPin);
    PWM = map(PWM, 0,1023,0,255);
    esc.write(PWM);
    for(int b = 0; b<bar; b++)                       // untuk 'b' kurang dari
        'bar', 'b' akan ditambah terus
    {

        lcd.write(byte(b));                         // tulis karakter graph sesuai
            nilai 'b' (b =jumlah barisnya)
    }
    if(vol==0)                                       // jika vol = 0
    {
        lcd.setCursor(1,0);
        lcd.print("x");                             // maka print 'x'
    }
}

```

```

lcd.print(" Mute");           // dan "Mute"
}
lcd.setCursor(8,0);
lcd.print(vol);              // cetak nilai 'vol'
lcd.print(" %");            // persentase
lcd.setCursor(0,1);
lcd.print("A = ");
lcd.print(readADC);
lcd.setCursor(9,1);
lcd.print("D = ");
lcd.print(PWM);

```

```

nilai1 = analogRead (A1);
nilai = map(nilai1, 0, 1023, 0, 255);
analogWrite (9, nilai);
persent = map (nilai, 0, 255, 0, 100);
Serial.print(persent);
Serial.println("%");
lcd.setCursor(0, 2);
lcd.print("Duty cycle = ");
lcd.print(persent);
lcd.print("%");
lcd.setCursor(0,3);
lcd.print("D = ");
lcd.print(nilai);

```

```

}

```

```

void data_olah() {
temp = analogRead(A1) * (5.0 / 1023.0); //konversi tegangan
analog menjadi digital
adc_Volt = abs(temp - 2.50); //mengambil selisih tegangan
adc_Volt /= 0.185; //Arus dalam A

```

```

adc_Volt *= 1000; //merubah arus A ke mA

if (waktu_kalibrasi < kalibrasi) {
waktu_kalibrasi++;
Serial.print("waktu kalibrasi :");
Serial.println(waktu_kalibrasi);
arus_temporary += adc_Volt; //penjumlahan arus output sensor
calibration = true;
}
else if (calibration == true) {
cal_value = arus_temporary / kalibrasi; //pembagian nilai
keseluruhan dengan waktu
calibration = false;
}
if (calibration == false) {
adc_Volt -= cal_value;
adc_Volt = abs(adc_Volt);
Serial.print("Arus= ");
Serial.print(adc_Volt);
Serial.println(" mA");
//delay(1000);
adc_Volt /= 1000;
Serial.print(adc_Volt);
Serial.print(" A");
Serial.println(" ");
delay(1000);
}
}

```

LAMPIRAN B PROGRAM

B.1 Pemrograman Keseluruhan

```
#include <Wire.h>

#include <Adafruit_MCP4725.h>

Adafruit_MCP4725 dac;

volatile unsigned int counter = 0;

unsigned long lastTime = 0;

const long interval = 1000;

int data;

int a;

int b;

void setup() {

    pinMode(2, INPUT);

    digitalWrite(2, HIGH);

    attachInterrupt(0, ai0, FALLING);

    Serial.begin(9600);

    dac.begin(0x62);

}
```

```
void loop() {  
  kirim();  
  if (timeIntervall()) {  
    sendData();  
    resetSampling();  
  
  }  
  
  uint32_t counter;  
  
  }  
  void ai0() {  
    counter++;  
  
  }  
  
  void kirim() {  
    if (Serial.available() > 0)  
      a = Serial.parseInt();  
      b = map(a, 0, 805, 0, 1600);  
      dac.setVoltage(b, true);  
      delay(50);  
  }  
}
```

B.4 Pemrograman Pembacaan Sensor di Arduino

```
/*  
*****  
*****  
*/
```

```
/*!
```

Rotary Encode Speed

Calculate the speed from pulses by an rotary encoder or other pulsesensor.

<https://github.com/BenTommyE/BenRotaryEncoder/>
This example code is in the public domain.

modified 27 Mars 2017

by Ben-Tommy Eriksen

```
*/
```

```
volatile unsigned int counter = 0;  
unsigned long lastTime = 0;  
const long interval = 1000;
```

```
void setup() {  
  Serial.begin (9600);
```

```
  pinMode(2, INPUT);  
  digitalWrite(2, HIGH);
```

```
  attachInterrupt(0, ai0, RISING);
```

```
}
```

```
void loop() {
```

```
  if(timeIntervall()) {  
    sendData();  
    resetSampling();
```

```
  }
```

```
}
```

```
void ai0() {
```

```

counter++;
}

void sendData() {

int c2 = map(counter, 0, 100, 0, 60);

int speedOut = c2 ;

int rpm = ((0.978 * speedOut) - 11.52 ) ;

Serial.println(rpm);

}

void resetSampling() {
counter = 0;

}

boolean timeIntervall() {
unsigned long currentTime = millis();

if (currentTime - lastTime >= interval) {
lastTime = lastTime + interval;

return true;
} else if (currentTime < lastTime) {

} else {
return false;
}
}

```

DAFTAR RIWAYAT HIDUP Nama : Bela Lorenza
TTL : Kediri, 22 Juni 1997
Jenis Kelamin : Perempuan
Agama : Islam
Alamat : Jl.Raya Kediri-Blitar
RT 03,RW 02 Kab.Kediri
Telp/Hp : 082231690091
E-mail :Lorenzabella22@gmail.com



RIWAYAT PENDIDIKAN

1. 2003 – 2009 : SD Negeri 030 Balikpapan Selatan
2. 2009 – 2012 : SMP Negeri 5 Balikpapan Selatan
3. 2012 – 2015 : SMA Negeri 1 Kandat Kab.Kediri
4. 2015 – 2019 : Departemen Teknik Elektro otomasi, Program Studi Teknik Elektro Industri - FV Institut Teknologi Sepuluh Nopember (ITS)

PENGALAMAN KERJA

1. Kerja Praktek di PT. PLN Kediri Rayon Pare
2. Kerja Praktek di PT. Airnav

-----Halaman ini sengaja dikosongkan-----