



PROYEK AKHIR - VE180626

**PERANCANGAN DAN PENGATURAN PENEREMAN REGENERATIF
BRUSHLESS DC SEBAGAI MODUL PEMBELAJARAN**

Muhammad Dani
NRP 10311500010020

Dosen Pembimbing
Ir. Joko Susila. MT.
Ciptian Wieried Priananda S.ST. MT.

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2020



FINAL PROJECT - VE180626

**DESIGN BRUSHLESS DC REGENERATIVE BRAKING AS A LEARNING
MODULE**

Muhammad Dani
NRP 10311500010020

Supervisor
Ir. Joko Susila MT.
Ciptian Wieried Priananda S.ST. MT

***DEPARTMENT ELECTRICAL AND AUTOMATION ENGINEERING
Vocational Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2020***

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

PERNYATAAN KEASLIAN

TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **“Perancangan dan Pengaturan Pengereman Regeneratif Motor Brushless DC sebagai Modul Pembelajaran”** 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, 30 Juni 2020

Muhammad Dani
NRP 10311500010020

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

**PERANCANGAN DAN PENGATURAN Pengereman
REGENERATIF BRUSHLESS DC SEBAGAI MODUL
PEMBELAJARAN**

TUGAS AKHIR

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

Menyetujui:

Pembimbing 1

Pembimbing 2



I. Joko Susila, MT.

NIP. 19621005 199003 1 003

Ciptian Weried Priananda S.ST. MT

NPP. 1990201711060

**SURABAYA
JUNI, 2020**

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

PERANCANGAN DAN PENGATURAN PENGGEREMAN REGENERATIF BRUSHLESS DC SEBAGAI MODUL PEMBELAJARAN

Nama : Muhammad Dani
NRP : 10311500010020
Pembimbing 1 : Ir. Joko Susila, MT.
Pembimbing 2 : Ciptian Weried Priyananda S.ST. MT.

ABSTRAK

Motor listrik pada saat ini memiliki peran yang penting dalam dunia industri. Dalam hal ini, motor brushless DC merupakan penggerak yang sering di pakai dunia industri di bandingkan motor jenis lainnya. Kelebihan dari penggerak ini adalah lebih efisien dalam mengkonversi listrik menjadi tenaga mekanik, tidak adanya sikat yang dapat mengurangi kerugian akibat gesekan, dan memiliki kecepatan yang tinggi. Dalam hal ini, pengereman regeneratif dijadikan suatu solusi agar gesekan pada saat pengereman dapat di konversikan menjadi energi listrik, kemudian menyimpannya untuk digunakan di lain waktu dan tidak terbuang begitu saja saat tidak digunakan.

Untuk membuat sebuah pengereman regeneratif diperlukan sebuah controller untuk mengatur pengoperasian motor dan drive system yang berfungsi untuk mentransfer energi mekanik menjadi energi listrik. Oleh karena itu untuk mempermudah proses pembelajaran, diperlukan suatu driver yang praktis dan efisien. Sehingga muncul ide untuk membuat suatu driver pengereman regenerative motor brushless DC dengan Arduino.

Hasil tugas akhir berupa driver pengereman regeneratif pada modul motor brushless DC dapat digunakan sebagai modul praktikum untuk adik-adik kelas di Departemen Teknik Elektro Otomasi maupun jurusan lain, dan menunjukkan bahwasannya pengereman yang mendapatkan kecepatan awal 2500 RPM dapat turun menjadi 1670 RPM dan mendapatkan tegangan rata-rata 8V.

**Kata Kunci : Sepeda Listrik, Pengereman Regenerative, dan
*Brushless DC***

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

DESIGN BRUSHLESS DC REGENERATIVE BRAKING AS A LEARNING MODULE

Name : Muhammad Dani
NOR : 10311500010020
Supervisor 1: Ir. Joko Susila, MT.
Supervisor 2: Ciptian Weriend Priyananda S.ST. MT.

ABSTRACT

Electric motors at this time played an important role in the industrial world. In this case, DC brushless motor is a frequent driver in the use of the industrial world in comparison to other types of motors. The advantages of this drive are more efficient in converting electricity into mechanical power, the absence of a brush that can reduce the loss caused by friction, and has a high speed. In this case, regenerative braking is used as a solution for friction when braking can be converted into electrical energy, then storing it for use at a later time and not being wasted when not in use.

To make a regenerative braking required a controller to regulate the operation of the motor and drive system that serves to transfer mechanical energy into electrical energy. Therefore, to facilitate the learning process, a driver is required to be practical and efficient. It appears that the idea to make a regenerative motor braking driver with the Arduino brushless motors.

The final task of the regenerative braking driver of the DC Brushless motor module can be used as a practicum module for the brothers of the class in the Department of Electrical Engineering and other majors, and shows that braking is getting an initial speed of 2500 RPM can drop to 1670 RPM and get an average voltage of 8V.

Keywords : Electric Motors, Regenerative Braking, and Bruhsless DC

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

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa karena atas limpahan rahmat dan karunia-Nya, penulis dapat menyelesaikan tugas akhir ini dengan judul :

“PERANCANGAN DAN PENGATURAN Pengereman REGENERATIF BRUSHLESS DC SEBAGAI MODUL PEMBELAJARAN”

Tugas Akhir ini merupakan sebagian syarat untuk menyelesaikan mata kuliah dan memperoleh nilai pada Tugas Akhir.

Dengan selesainya Tugas Akhir ini penulis menyampaikan terima kasih sebesar-besarnya kepada:

1. Orang Tua atas limpahan doa, kasih sayang, dukungan dan dorongan baik berupa moril atau materil bagi penulis.
2. Bapak Imam Arifin, ST., MT. selaku Ketua Program Teknik Elektro Otomasi Bidang Studi Teknik Listrik, FV-ITS Surabaya.
3. Bapak Ir. Joko Susila, MT ,dan Ciptian Wieried Priananda S.ST. MT selaku Dosen Pembimbing.
4. Seluruh staf pengajar dan administrasi Prodi Teknik Elektro Otomasi FV-ITS.
5. Seluruh Mahasiswa Teknik Elektro Otomasi khususnya angkatan 2015.
6. Semua pihak yang telah banyak membantu untuk menyelesaikan Tugas Akhir ini yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari masih banyak kekurangan dalam Tugas Akhir ini. Kritik dan saran untuk perbaikan tugas ini sangat diperlukan. Akhir kata semoga tugas ini dapat bermanfaat bagi kita semua.

Surabaya, 30 Juni 2020

Penulis

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

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR	xii
DAFTAR ISI	xiv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Sistematika Penulisan	3
1.6 Relevansi	3
BAB II TEORI PENUNJANG	5
2.1 Motor <i>Brushless</i> DC	5
2.2 Motor DC sebagai Generator	7
2.3 Pengereman Regeneratif	9
2.4 Flywheel	9
2.5 Mikrokontroler Arduino Mega 2560	10
2.6 Sensor Tegangan DC	13
2.7 Rotary Encoder	13
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
3.2.1.1 Motor <i>Brushless</i> DC	16
3.2.1.2 <i>Generator</i> dengan Motor DC	16
3.2.1.3 Pengereman regeneratif	17
3.2.1.4 Sensor Tegangan	17
3.2.1.3 Flywheel	18
3.3.2 Perancangan Elektrik	19

3.3.2.1 Rangkain Arduino Mega 2560.....	19
3.2.2.2 rangkaian sensor rotary encoder	20
3.3 Perancangan Perangkat Lunak (<i>Software</i>)	21
3.4.1 Pemrograman <i>Software</i> Arduino	22
BAB IV PENGUJIAN DAN ANALISA	25
4.2 Pengujian Sensor Tegangan	25
4.3 Pengujian Pengereman Regeneratif	26
BAB V PENUTUP	29
5.1 Kesimpulan	29
5.2 Saran.....	29
DAFTAR PUSTAKA	31
LAMPIRAN A LISTING PROGRAM	A-1
LAMPIRAN B DATASHEET	B-1
LAMPIRAN C DOKUMENTASI	C-1
DAFTAR RIWAYAT HIDUP	D-1

DAFTAR GAMBAR

Gambar 2.1	Salah Satu Contoh BLDC	5
Gambar 2.2	Motor DC <i>Gearbox</i>	8
Gambar 2.3	Arduino Mega 2560	11
Gambar 2.4	<i>Setting Serial Port</i>	12
Gambar 2.5	<i>Rotary Encoder</i>	13
Gambar 3.1	Blok Fungsional Perancangan.....	15
Gambar 3.2	Blok Diagram Sistem	16
Gambar 3.3	Rangkaian Pembagi Tegangan	17
Gambar 3.4	Beban Flywheel tampak Samping.....	18
Gambar 3.5	Beban Flywheel Tampak Depan	18
Gambar 3.6	<i>Ilustrasi bentuk Flywheel</i>	19
Gambar 3.7	<i>Layout Arduino Mega 2560</i>	20
Gambar 3.8	Skematik Arduino Mega 2560 dengan Komponen....	20
Gambar 3.9	Rangkaian Rotary Encoder.....	21
Gambar 3.10	Flowchart Arduino Mega 2560	23
Gambar 4.1	Grafik Perbandingan Nilai ADC dengan Tegangan Voltmeter.....	26

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

DAFTAR TABEL

Tabel 2.1 Spesifikasi Motor <i>Brushless</i> DC Tipe <i>Outrunner</i> <i>RCTimer</i> BL2830-11 1000KV	6
Tabel 2.2 Spesifikasi Motor <i>Gearbox</i>	8
Tabel 2.3 Spesifikasi Arduino Mega 2560	11
Tabel 4.1 Pengujian Kalibrasi Sensor Tegangan	25
Tabel 4.2 Generator dan Sensor Tegangan Saat Kecepatan Motor 1500 Rpm	26
Tabel 4.3 Generator dan Sensor Tegangan Saat Kecepatan Motor 2000 Rpm	27
Tabel 4.4 Generator dan Sensor Tegangan Saat Kecepatan Motor 2500 Rpm	27
Tabel 4.5 Generator dan Sensor Tegangan Saat Kecepatan Motor 3000 Rpm	27
Tabel 4.6 Generator dan Sensor Tegangan Saat Kecepatan Motor 3500 Rpm	27

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

BAB I

PENDAHULUAN

1.1 Latar Belakang

Minyak bumi merupakan sumber daya alam yang tidak dapat diperbaharui. Karena jumlahnya yang semakin sedikit, maka manusia dituntut untuk mencari alternatif energi yang dapat menggantikan minyak bumi. Pada perkembangan dunia teknologi, khususnya otomotif, mulai bermunculan kendaraan-kendaraan dengan sumber energi alternatif, salah satunya adalah sepeda listrik.

Sepeda listrik adalah rangkaian sepeda yang dikombinasikan dengan sebuah motor yang digerakkan dengan sebuah baterai, sehingga mudah digunakan oleh siapapun bahkan para manula. Penggunaan alat transportasi dengan menggunakan baterai yang dapat di-recharge ini mulai berkembang di Indonesia. Meskipun kesadaran masyarakat untuk menggunakan sepeda sebagai salah satu alat transportasi makin meningkat, sepeda belum bisa dijadikan alat transportasi utama seperti di negara maju pada umumnya. Hal ini wajar saja, mengingat kondisi jalan-jalan di Indonesia, khususnya kota-kota besar yang dipenuhi kendaraan bermotor.

Kontur jalan di berbagai tempat di Indonesia begitu banyak yang naik dan turun. Pada saat jalanan turun, kecepatan kendaraan kita akan meningkat dan membutuhkan pengereman untuk menurunkannya. Pada suatu sistem pengereman kendaraan, pengereman dapat dilakukan ketika menarik/menginjak tuas rem yang membuat 2 buah kampas rem akan menjepit rotor pada roda. Akibat gesekan pada kampas rem dan bidang rotor yang berputar menyebabkan perlambatan pada kendaraan dan terjadi perubahan energi dari energi mekanik menjadi energi panas.

Sistem pengereman ini kurang efisien jika digunakan dalam sepeda listrik yang digerakkan oleh daya yang disimpan dalam baterai. Dengan motor listrik yang dapat melakukan pengereman regeneratif, motor listrik ini nantinya akan mendapatkan tambahan daya dikarenakan sistem pengereman regeneratif yang dapat mengubah energi mekanik menjadi energi listrik, sehingga sistem pengereman ini membuat sepeda listrik lebih efisien dibandingkan dengan sistem pengereman mekanik biasa.

Pengereman regeneratif membutuhkan sebuah generator agar dapat menghasilkan tegangan saat terjadi pengereman. Generator yang

digunakan adalah motor DC dvd. Generator dan motor DC memiliki komponen-komponen yang sama persis. Meskipun demikian, fungsi dari keduanya saling bertolak belakang. Motor listrik berfungsi untuk mengubah energi listrik menjadi energi mekanis, sedangkan generator berfungsi untuk mengubah energi mekanis menjadi energi listrik, dan apabila arah putaran rotor pada motor DC di balik, maka terjadilah pembangkitan listrik atau berubahnya motor DC menjadi generator DC.

Salah satu cara yang memungkinkan untuk melakukan kontrol pada sistem regenerative braking ini adalah dengan menggunakan arduino yang dikemas dalam bentuk modul pembelajaran dan dikontrol dengan menggunakan suatu metode kontrol.

1.2 Pemasalahan

Permasalahan yang diangkat pada proyek akhir ini adalah ;

1. Belum adanya modul pembelajaran praktis guna mempermudah proses pembelajaran mengenai pengereman regeneratif motor *brushless* DC.
2. Pentingnya efisiensi energi tak terbarukan dan mencegah pemanasan global

1.3 Batasan Masalah

Pembatasan pada tugas akhir ini meliputi :

1. Kontrol kecepatan motor dan output dari sistem pengereman regeneratif yang hanya di tampilkan pada *LCD Graphic*
2. Sistem pengereman yang bersifat manual

1.4 Tujuan

Tujuan proyek akhir ini adalah:

1. Merancang sistem pengereman regeneratif motor *brushless* DC.
2. Membuat modul pembelajaran praktis yang mudah digunakan sebagai sarana pembelajaran pengereman regeneratif motor *brushless* DC.
3. Mengimplementasikan pengereman regeneratif motor *brushless* DC.

1.5 Sistematika Penulisan

Dari proses pembuatan alat proyek akhir ini dimulai dari studi literatur, menentukan gambar desain alat, membuat alat, membuat program, menguji alat secara keseluruhan, analisa data, serta dapat menyusun laporan akhir dengan sistematika penulisan yaitu pendahuluan, teori penunjang, perancangan alat, pengukuran dan analisa serta penutup.

BAB I PENDAHULUAN

Berisi latar belakang pembuatan alat, permasalahan, pembatasan masalah, tujuan, metodologi, sistematika penulisan serta manfaat Proyek Akhir.

BAB II TEORI PENUNJANG

Meliputi pembahasan dan teori-teori penunjang yang digunakan dalam mendukung terselesaikannya Proyek Akhir. Teori-teori yang dipakai dalam pembuatan Proyek Akhir ini antara lain : motor DC, Arduino, *flywheel*, motor DC sebagai Generator, serta teori penunjang lainnya

BAB III PERANCANGAN ALAT

Membahas tentang tahap-tahap perancangan mekanik dan perancangan sistem kontrolernya

BAB IV PENGUKURAN DAN ANALISA

Membahas tentang pengukuran dan pengujian alat atau rangkaian yang digunakan pada proyek akhir ini.

BAB V PENUTUP

Berisikan tentang kesimpulan atas hasil yang diperoleh serta saran-saran atas kekurangan dan kelemahan proyek akhir ini.

1.6 Relevansi

Hasil yang diperoleh dari Perancangan dan Pengaturan Pengereman Regeneratif Motor *Brushless* DC sebagai Modul Pembelajaran akan digunakan untuk mempermudah pembelajaran dalam pengereman regeneratif motor *brushless* DC.

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

BAB II

TEORI PENUNJANG

Sistem kendali atau sistem kontrol adalah suatu alat (kumpulan alat) untuk mengendalikan, memerintah, dan mengatur keadaan dari suatu sistem. Banyak contoh dalam bidang sistem kendali/kontrol salah satunya adalah sistem pengereman regenerative pada motor brushless DC.

Pada bab ini akan dibahas mengenai teori-teori yang berkaitan dengan peralatan yang akan dirancang. Teori yang mendukung penyelesaian tugas akhir ini diantaranya adalah mengenai motor brushless dc, pengereman regeneratif, boost converter, baterai, dan sensor arus.

2.1 Motor Brushless DC

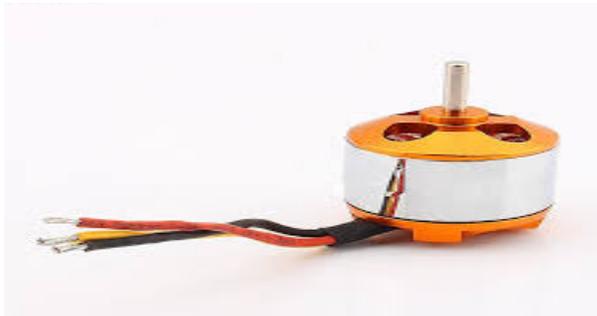
Brushless DC (BLDC) motor merupakan pilihan ideal untuk aplikasi yang membutuhkan keandalan yang tinggi, efisiensi tinggi, dan rasio *power-to-volume* yang tinggi. Secara umum, motor BLDC dianggap sebagai motor dengan performa tinggi yang mampu menghasilkan torsi yang besar pada rentang kecepatan yang besar. BLDC motor adalah turunan dari motor DC yang paling umum digunakan, yaitu motor DC dengan sikat, dan mereka memiliki kurva karakteristik torsi dan kecepatan yang sama. Perbedaan utama antara keduanya adalah penggunaan sikat. Motor BLDC tidak memiliki sikat dan harus terkomutasi secara elektronik.

Motor BLDC adalah tipe dari motor sinkron. Hal ini berarti medan magnetik yang dihasilkan oleh stator dan medan magnetik yang dihasilkan oleh rotor berputar pada frekuensi yang sama. Motor BLDC tidak mengalami slip yang biasanya terjadi pada motor induksi. Motor BLDC terdapat dalam konfigurasi 1 fasa, 2 fasa, dan 3 fasa. Berdasarkan tipe stator yang memiliki jumlah kumparan yang sama terdapat dua tipe dari kumparan stator motor BLDC yaitu trapezoidal dan sinusoidal perbedaan ini dibuat berdasarkan dari hubungan kumparan stator yang menghasilkan tipe GGL balik yang berbeda.

Seperti namanya motor BLDC tidak menggunakan sikat untuk melakukan komutasi tetapi menggunakan komutasi elektris, untuk dapat memutar motor BLDC maka kumparan stator dari motor

harus diberi tegangan sesuai urutan komutasi, Supaya dapat melakukan hal tersebut maka posisi rotor harus dapat diketahui. Posisi rotor ini dapat diketahui dengan menggunakan sensor *hall effect* atau dengan teknik *sensorless* yaitu dengan mendeteksi GGL balik pada kumparan stator. Pada umumnya motor BLDC memiliki tiga sensor *hall effect* yang terpasang dekat stator dan memiliki kontroler elektronik dalam motor untuk mengatur komutasi daya yang mengalir dalam kumparan stator.

Pada proyek akhir kali ini motor yang kami gunakan adalah Motor *Brushless* DC Tipe *Outrunner* RC Timer A2212 1400RPM yang biasa digunakan untuk balang baling quadcopter



Gambar 2.1 Motor *Brushless* DC Tipe *Outrunner* RC Timer A2212 1400 RPM

Tabel 2.1 Spesifikasi Motor *Brushless* DC Tipe *Outrunner* RC Timer A2212 1400RPM

No.	Parameter	Nilai	
1.	Berat Motor	47 gram	
2.	KV	1400 RPM/Volt	
3.	Tegangan	Tegangan Minimal	7 Volt
		Tegangan Maksimal	15 Volt
4.	<i>Input</i> Arus	Arus Nominal	4 Ampere
		Arus Maksimal	18 Ampere
5.	<i>Maximal Power</i>	210 Watt	
6.	<i>Input</i> Baterai	Lithium Polimer 2S-4S	

2.2 Motor DC sebagai Generator

Mesin DC (Direct Current) merupakan salah satu jenis mesin listrik, dimana mesin ini digunakan untuk mengkonversi energi listrik arus searah menjadi energi mekanik, atau sebaliknya. Ada dua macam mesin DC, yakni: Motor DC dan Generator DC. Pada pengoperasiannya, motor DC dapat mengkonversi energi listrik arus searah menjadi energi mekanik, sedangkan generator DC sebaliknya.

Berdasarkan konstruksinya, ada tiga komponen penting dari mesin listrik, yakni: stator, rotor, dan air gap. Stator merupakan bagian mesin yang statis, rotor merupakan bagian mesin yang bergerak/ berputar, sedangkan air gap merupakan celah antara stator dan rotor yang berfungsi untuk mencegah terjadi friksi antara stator dengan rotor, serta mempermudah rotor untuk berputar pada porosnya.

Motor dan generator DC memiliki beberapa perbedaan dasar, diantaranya terkait dengan pencatutan pada komponen mesin tersebut, yakni stator dan rotor. Pada motor DC, bagian stator dan rotornya diberi *supply* tegangan arus searah (VDC). Karena rangkaian stator dan rotor merupakan *close loop*, serta memiliki resistansi dalam (kawat penghantar), maka timbulah arus searah (DC) pada kedua komponen. Kemudian, berdasarkan hukum oersted, dimana saat sebuah kawat penghantar dialiri arus, akan dihasilkan medan magnet di sekeliling kawat penghantar. Maka, pada kedua komponen tersebut dihasilkan medan magnet. Medan magnet pada stator merupakan medan magnet utama, dimana nantinya akan mendominasi dalam penentuan arah perputaran rotor. Sedangkan medan magnet pada rotor seringkali disebut sebagai medan magnet jangkar, karena dalam prakteknya rotor seringkali disebut dengan kumparan jangkar (*armature winding*). Medan magnet merupakan besaran vector (besaran yang memiliki satuan dan arah). Bila ada dua medan magnet berdekatan, maka akan dihasilkan medan magnet resultan yang merupakan hasil interaksi antara magnet pada stator dengan Magnet pada rotor.

Karena rotor dialiri arus searah dan terdapat medan magnet resultan, maka timbulah gaya Lorentz (F). Kemudian, pada rotor dihasilkan torsi (T) yang merupakan hasil kali vector antara lengan gaya (direpresentasikan dengan jari-jari rotor) dengan gaya yang

timbul pada rotor. Pada akhirnya, torsi yang bekerja pada rotor inilah yang membuat rotor dapat berputar. Dalam hal ini, peran komutator dan brush diperlukan, yakni untuk menjaga arah putaran rotor supaya tetap satu arah, baik *clock wise* maupun *counter clock wise*.

Berdasarkan penjelasan singkat di atas, dapat diketahui bahwa pada pengoperasiannya motor DC dapat mengkonversi energi listrik, yakni energi yang disupply ke stator dan rotor, menjadi energi gerak, yakni energi yang menyebabkan rotor berputar pada porosnya.

Berbeda dengan motor DC. Pada *generator* DC, komponen mesin yang diberi supply tegangan arus searah (VDC) hanyalah stator, sedangkan rotor dikopel (dihubungkan secara fisik) dengan prime mover atau alat penggerak, tanpa diberi *supply*.

Motor DC yang digunakan pada proyek akhir ini adalah motor DC *gearbox* 3-6V



Gambar 2.2 Motor DC *Gearbox*

Tabel 2.2 Spesifikasi Motor DC *Gearbox*

Tegangan	DC 3V	DC 5V	DC 6V
Arus	100 ma	100ma	120ma
Rasio Reduksi	48:1		
RPM	100	190	240
Berat Motor (g)	29		
Ukuran Motor	70mm*22mm*18mm		

2.3 Pengereman regeneratif

Sistem pengereman regeneratif adalah sistem pengereman yang canggih dimana digunakan bersamaan dengan sistem pengereman konvensional pada kendaraan listrik ataupun kendaraan hybrid.

Pada rem konvensional, bekerja dengan cara merubah energi mekanik menjadi energi panas. Namun ternyata panas yang dihasilkan dari gesekan antara sepatu rem dengan piringan / tromol menjadi sia-sia. Dengan adanya teknologi regeneratif rem, maka energi panas yang sia-sia dapat dipergunakan kembali menjadi energi listrik yang bermanfaat. Dimana sistem akan menangkap dan mengubah menjadi energi listrik untuk mengisi daya baterai.

Sistem pengereman regeneratif pada mobil listrik ataupun hybrid memiliki sebuah motor penggerak. Motor ini mempunyai fungsi ganda yaitu sebagai penggerak untuk menggerakkan kendaraan dan sebagai generator untuk mengisi daya pada baterai.

Motor akan bekerja sebagai penggerak pada satu putaran tertentu, dan menjadi generator untuk mengisi baterai pada putaran berlawanan. Ketika bekerja sebagai motor penggerak, motor mengubah energi listrik menjadi energi mekanik dan menggerakkan roda. Nah, ketika kendaraan melakukan pengereman, arah putaran motor menjadi berlawanan.

Sederhananya adalah jika menjadi motor penggerak, motor berputar searah jarum jam. Maka ketika menjadi generator berputar berlawanan arah jarum jam. Pada prinsip mobil listrik atau hybrid, bahwa putaran sebaliknya yaitu pada mode generator, dengan demikian hal ini dapat memperlambat roda.

2.4 *Flywheel*

Flywheel adalah perangkat mekanik berputar yang digunakan untuk menyimpan energi rotasi. *Flywheel* memiliki momen inersia yang signifikan, dan dengan demikian menahan perubahan kecepatan rotasi.

Jumlah energi yang tersimpan dalam *flywheel* adalah sebanding dengan kuadrat kecepatan rotasi. Energi ditransfer ke *flywheel* dengan menggunakan torsi, sehingga meningkatkan kecepatan rotasi, dan karenanya energi dapat tersimpan. Sebaliknya, *flywheel*

melepaskan energi yang tersimpan dengan melakukan torsi ke beban mekanik, sehingga mengurangi kecepatan rotasi.

Penggunaan umum dari roda gila meliputi:

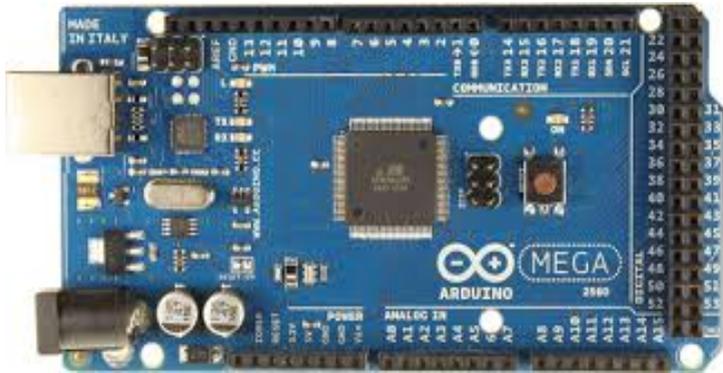
- Menyediakan energi yang terus menerus ketika sumber energi terputus. Misalnya, *flywheel* yang digunakan dalam mesin piston (*piston engine / reciprocating engine*), karena sumber energi berupa torsi dari mesin, berselang (tidak konstan).
- Memberikan energi pada tingkat di luar kemampuan sumber energi yang terus menerus. Hal ini dicapai dengan mengumpulkan energi dalam *flywheel* dari waktu ke waktu dan kemudian melepaskan energi dengan cepat, dengan tingkat yang melebihi kemampuan sumber energi.
- Mengontrol orientasi dari sebuah sistem mekanik. Dalam aplikasi tersebut, momentum sudut dari *flywheel* sengaja ditransfer ke beban ketika energi ditransfer ke atau dari *flywheel*.

Flywheel biasanya terbuat dari baja dan berputar pada bantalan (*bearings*) konvensional, dan ini umumnya terbatas pada tingkat revolusi kurang dari 1000 RPM. Beberapa *flywheel* modern terbuat dari bahan serat karbon dan menggunakan bantalan magnet, memungkinkan *flywheel* untuk berputar pada kecepatan sampai 60.000 RPM.

Akan tetapi pada proyek yang kami buat ini terbuat dari *Acrylic* yang di cutting dan disusun menyerupai beban pada *flywheel*, dan juga *bearing*, as besi yang menjadi satu dengan pulley BLDC

2.5 Mikrokontroler Arduino Mega 2560

Arduino Mega 2560 adalah papan pengembangan (*development board*) mikrokontroler yang berbasis chip Atmega2560. Disebut sebagai papan pengembangan karena *board* ini memang berfungsi sebagai arena *prototyping* sirkuit mikrokontroler. Dengan menggunakan papan pengembangan, anda akan lebih mudah merangkai rangkaian elektronika mikrokontroler dibanding jika anda memulai merakit ATmega2560 dari awal di *breadboard*. Tabel 2.3 merupakan spesifikasi arduino mega 2560



Gambar 2.3 Arduino mega 2560

Tabel 2.3 Spesifikasi Arduino Mega 2560

Mikrokontroler	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 (15diantaranya menyediakan PWM)
<i>Analog Input Pins</i>	16
<i>DC Current per I/O Pin</i>	20 mA
<i>DC Current for 3,3V Pin</i>	50 mA
<i>Flash Memory</i>	256 KB (Atmega2560) (8 KB sebagai bootloader)
SRAM	8 KB (Atmega2560)
EEPROM	4 KB (Atmega2560)
<i>Clock Speed</i>	16 MHz
LED_BUILTIN	13
<i>Length</i>	101,52 mm
<i>Width</i>	53,3 mm
<i>Weight</i>	37 g

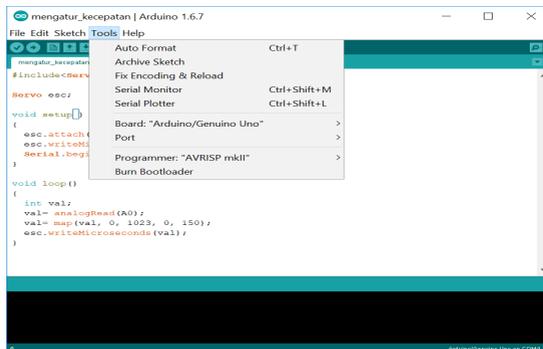
Untuk memprogram arduino juga harus dilakukan beberapa tahapan sebagai berikut :

1. *Setting Board Arduino.*

Dalam memprograman *software* arduino harus di *setting* terlebih dahulu *board* arduino agar penggunaan arduino cocok. Dalam purwarupa kali ini arduino menggunakan arduino Mega 2560. Untuk *setting board* arduino bisa masuk ke *tools – board* – setelah itu pilihlah *board* arduino yang sesuai.

2. *Setting Serial.*

Serial ini merupakan kabel arduino yang dihubungkan kepada komputer atau laptop. *Serial* ini mempunyai dua fungsi yang bisa digunakan. Pertama *serial port* digunakan untuk mendownload program dari arduino yang kedua *serial* digunakan sebagai komunikasi *serial* pada arduino dengan komputer. *Setting serial* bisa masuk *tools – serial* - lali pilih COM yang sesuai dengan arduino yang terpasang. Untuk lebih jelasnya dapat dilihat pada Gambar 2.4.



Gambar 2.4 *Setting Serial Port*

Apabila *program* tidak dapat di *download* karena *serial port*, maka cek terlebih dahulu *serial* yang benar pada *device manager*. Lalu dalam *software* arduino untuk memilih *serial portnya* samakan dengan *serial port* untuk arduino dalam *device manager* tersebut. Untuk masuk ke *device manager* dapat masuk start windows – lalu ketika *device manager* klik dua kali dan masuk ke dan COM.

2.6 Sensor Tegangan DC

Tegangan (dan arus) merupakan parameter dasar dalam dunia elektro baik digital maupun analog. Tegangan ini merupakan besaran analog, jadi dalam elektro analog, tegangan dapat langsung diolah, diproses atau dikonversi dalam bentuk atau level lainnya. Sedangkan dalam dunia digital, tegangan akan dikonversi versi diskritnya dengan ADC (Analog to Digital Converter) atau jika dibalik dari digital ke tegangan analog harus menggunakan teknik DAC (*Digital to Analog Converter*).

Demikian juga dengan Arduino. Pada dasarnya Arduino dapat membaca nilai tegangan dengan memanfaatkan pin analog. Jika range tegangan yang dibaca diantara 0-5 V bisa langsung menggunakan pin analog, sedangkan jika range tegangan yang dibaca $>5V$ harus menggunakan rangkaian tambahan yakni pembagi tegangan karena pin arduino bekerja pada max 5 v.

2.7 Sensor Rotary Encoder



Gambar 2.5 Rotary Encoder

Rotary encoder adalah suatu komponen elektro mekanis yang memiliki fungsi untuk memonitoring posisi angular pada suatu poros yang berputar. Dari perputaran benda tersebut data yang termonitoring akan diubah ke dalam bentuk data digital oleh rotary encoder berupa lebar pulsa kemudian akan dihubungkan ke mikrokontroler.

Konstruksi rotary encoder berupa piringan tipis yang biasanya di kopel dengan poros yang berputar. Piringan tipis tersebut terdapat lubang di sepanjang pinggir lingkarannya. Di bagian sisi-sisi piringan terdapat sebuah led dan phototransistor di bagian bersebrangan. Fungsi

dari lubang-lubang yang berada di sepanjang pinggir lingkaran tersebut akan menghantarkan cahaya led ke phototransistor, sebaliknya jika cahaya led tidak menembus lubang piringan maka cahaya akan tertahan. Piringan tersebut akan berputar sesuai dengan kecepatan putaran motor sehingga phototransistor akan saturasi ketika cahaya LED menembus lubang-lubangnya.

Pada saat saturasi phototransistor akan menghasilkan pulsa dengan range +0,5 V s/d +5 V.

BAB III

PERANCANGAN DAN IMPLEMENTASI

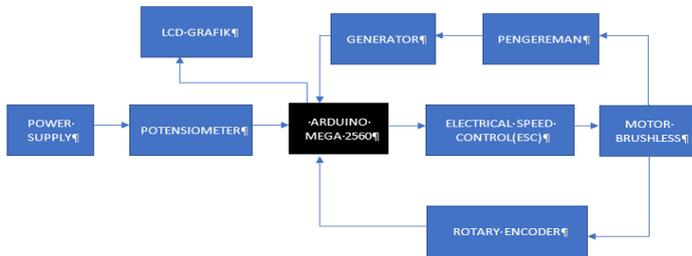
Pada bab ini akan di bahas mengenai perancangan alat yang terdiri dari perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.1 Blok Fungsional Sistem

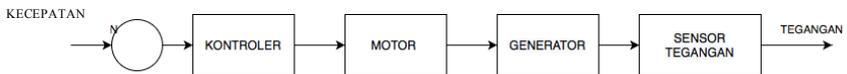
Perancangan sistem dalam pembuatan alat ini secara garis besar disertai urutan dan cara kerja alat ini di ilustrasikan pada Gambar 3.1 dan Gambar 3.2.

Dari Gambar 3.1 dapat dilihat bahwa sistem tersebut terdiri dari beberapa blok fungsional yaitu;

1. Power Supply, perangkat yang digunakan sebagai sumber untuk proyek akhir.
2. Potensiometer, sebagai pengatur arus masukan terhadap driver motor.
3. Arduino mega 2560, merupakan mikrokontroler yang berfungsi sebagai *interface* dari perangkat elektronik dan dapat menyimpan program didalamnya.
4. *Driver* motor yaitu ESC berfungsi sebagai penggerak motor dan sensor.
5. Motor *brushless* DC, digunakan sebagai objek yang akan dikontrol kecepatannya.
6. Sensor kecepatan (*rotary encoder*), digunakan untuk mengetahui sampai seberapa kecepatan motor yang terjadi.
7. *Generator*, digunakan sebagai objek yang akan di gunakan untuk menghasilkan tegangan
8. Pengereman, sebuah alat yang digunakan untuk memperlambat laju motor dan membantu generator menghasilkan tegangan
9. Sensor tegangan, digunakan untuk mengetahui berapa tegangan yang dihasilkan dari generator
10. Tampilan (Display), untuk tampilan digunakan LCD graphic, LCD graphic ini di gunakan untuk menampilkan hasil tegangan dan sensor kecepatan



Gambar 3.1 Blok Fungsional Perancangan



Gambar 3.2 Blok Diagram Sistem

Masukan yang diterima oleh pengereman adalah kecepatan yang di dapatkan dari gerak motor *brushless* DC, yang dimana pengereman ini akan mendorong generator hingga menempel pada *pulley* dari motor *brushless*, sehingga menghasilkan perlambatan kecepatan dan menghasilkan tegangan yang akan di terima oleh arduino mega 2560 untuk ditampilkan pada LCD grafik

3.2 Perancangan Perangkat Keras (hardware)

Perancangan yang dilakukan dalam perancangan perangkat keras ada dua jenis, yaitu perancangan mekanik dan elektronik. Perancangan mekanik merupakan perancangan komponen utama pada plant, yaitu berupa flywheel serta generator DC. Sedangkan perancangan elektronik merupakan perancangan untuk kontroler dan rangkaian sensor.

3.2.1 Perancangan mekanik

Perancangan mekanik ini terdiri dari desain alat secara keseluruhan, genererator dengan motor DC, rotary encoder, dan flywheel.

3.2.1.1 Motor *Brushless* DC

Motor BLDC dipilih karena memiliki berbagai kelebihan, yaitu, dapat menggerakkan dengan daya yang besar, suaranya halus, torsi besar, efisiensi tinggi, mudah di control, dan perawatan yang rendah. Motor BLDC yang digunakan merupakan motor BLDC yang biasa digunakan untuk *aeromodelling* sehingga ukurannya lebih kecil. Untuk lebih jelasnya mengenai motor ini dapat dilihat pada Gambar 2.1 dan Tabel 2.1.

3.2.1.2 Generator dengan Motor DC

Motor DC merupakan mesin listrik yang dapat mengubah energi listrik menjadi energi mekanik dan sebaliknya. Motor DC yang digunakan pada proyek akhir ini adalah motor DC *gearbox* seperti pada gambar 2.2, motor ini memiliki spesifikasi yang dapat menghasilkan tahanan lebih banyak dari pada motor DC tanpa *gearbox*. Motor DC ini disusun menjadi 3 yang saling menghimpit pulley dari *flywheel*. Oleh karena itu motor DC yang kami gunakan ini sangat penting dalam hal me-regeneratif pengereman.

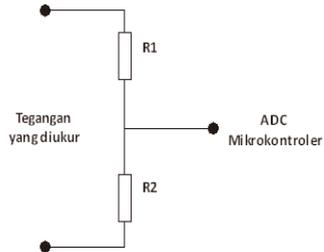
3.2.1.3 Pengereman Regeneratif

Pengereman regeneratif pada plant ini berguna sebagai pendukung dari komponen generator. Komponen pada pengereman regeneratif ini terdapat Generator (Motor DC), dan tuas yang tergabung dengan generator. Terdapat pengunci pada tuas yang digunakan sebagai pengeremannya, lalu gerak mekanik yang dihasilkan dari tuas ini adalah generator yang akan bergesekan dengan sumber mekanik dari percepatan motor BLDC. Adapun sumber mekanik dari Motor BLDC ini adalah *FlyWheel* yang bebannya terbuat dari acrylic.

3.2.1.4 Sensor Tegangan

Sensor tegangan yang digunakan pada proyek akhir ini, menggunakan rangkaian pembagi tegangan yang pin *out*nya terhubung dengan pin A0 pada arduino. dikarenakan range tegangan yang dapat dibaca oleh ar-

duino *max* 5V, maka diperlukan lah rangkaian tambahan yakni pembagi tegangan

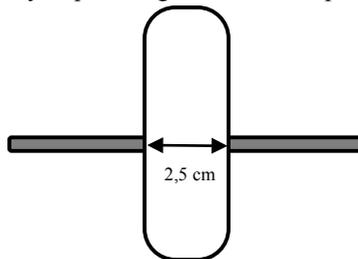


Gambar 3.3 Rangkaian Pembagi tegangan

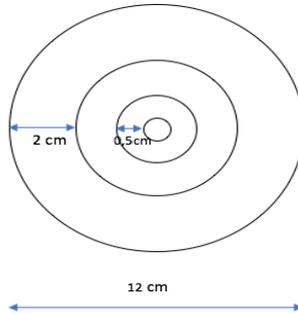
Kemudian hasil nilai ADC yang diterima oleh arduino diubah menjadi satuan Volt melalui program dengan memberikan rumus $y = 0,0245x + 0,0497$ yang didapatkan dari nilai ADC sensor tegangan pada LCD *graphic* dan dibandingkan dengan nilai tegangan yang terukur oleh voltmeter.

3.2.1.5 *Flywheel*

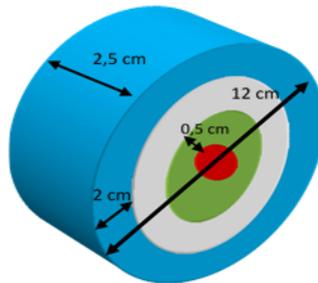
Flywheel (roda gila) merupakan perangkat untuk menyimpan energi rotasi, perangkat ini mendapatkan sumber energi dari percepatan motor BLDC. Dan kemudian perangkat ini meneruskan energi rotasi ke motor generator yang membuat motor generator menghasilkan tegangan. *Flywheel* yang kami gunakan ini berbahan acrylic pada bagian beban dan penyangganya.



Gambar 3.4 Beban *Flywheel* Tampak Samping



Gambar 3.5 Beban *Flywheel* Tampak Depan



Gambar 3.6 Ilustrasi Bentuk *Flywheel*

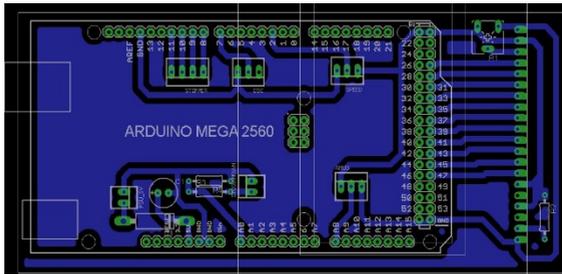
Acrylic yang telah di cutting, disusun dan di rangkai menyerupai flywheel. Gambar 3.4 tebal roda beban yang berasal dari 3 lapis cutting acrylic. Lapisan pertama berbentuk donat dengan diameter 12 cm, 2 cm ketebalan dri sisi luar dan 0,5 cm ketebalan bagian dalam seperti pada gambar 3.5. Lapisan kedua berbentuk jaring-jaring roda. Lapis pertama dan ketiga memiliki bentuk yang sama.

3.2.2. Perancangan Elektrik

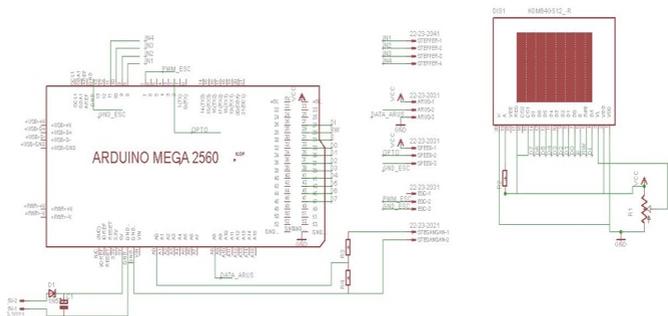
Perancangan elektrik ini meliputi skematik Arduino mega 2560 serta pengkabelan. Rangkaian elektrik pada plant ini meliputi rangkaian sensor tegangan, sensor kecepatan, dan arduino mega 2560.

3.2.2.1. Rangkaian Arduino Mega 2560

Rangkaian Arduino Mega pada Proyek Akhir ini digunakan sebagai core dari segala komponen yang menghubungkan satu komponen dengan komponen lain secara terstruktur. Sehingga setiap sistem dapat bekerja dengan sesuai fungsi dari komponen. Untuk itu dibutuhkan rangkaian yang tepat sehingga menghasilkan nilai yang diinginkan.



Gambar 3.7 Layout Arduino Mega 2560



Gambar 3.8 Skematik Arduino Mega 2560 dengan Komponen

Dari Gambar 3.5 merupakan layout yang menunjukkan pinout dari Arduino Mega. Sedangkan Gambar 3.6 merupakan skematik Arduino dengan beberapa komponen lain, seperti Sensor arus, sensor tegangan, motor stepper, motor brushless dan GLCD. Masing – masing komponen memiliki fungsi sendiri yang saling berkaitan,

sehingga dibutuhkannya penghubung untuk menyinkronkan semua komponen tersebut.

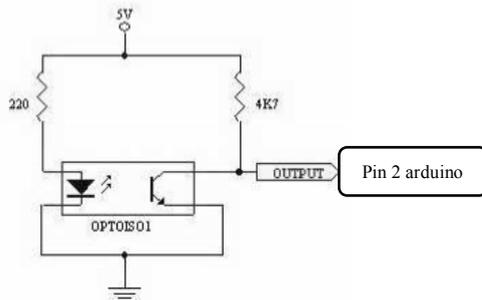
3.2.2.2 Rangkaian Sensor *Rotary Encoder*

Rotary encoder pada Proyek 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*.

Rotary encoder ini tersusun dari suatu piringan tipis yang memiliki lubang-lubang yang terdapat pada piringan tersebut.

Setelah itu akan ditempatkan LED pada salah satu sisi piringan. Hal ini akan membuat cahaya masuk menuju piringan.

Kemudian disisi lain dari piringan ini diletakan *phototransistor* yang bertujuan untuk mendeteksi cahaya LED yang berseberangan. Piringan tipis ini yang nantinya akan dikopel dengan poros motor ataupun perangkat berputar lainnya yang ingin kita ketahui posisinya, hal ini akan membuat piringan berputar ketika motor tersebut berputar. Apabila cahaya yang berasal dari LED mencapai *phototransistor*, maka *phototransistor* itu akan mengalami saturasi dan akan menghasilkan suatu pulsa gelombang persegi.



Gambar 3.9 Rangkaian *Rotary Encoder*

3.3. Perancangan Perangkat Lunak (*Software*)

Agar motor stepper dapat dikontrol dan dapat mendorong generator sehingga terjadi gesekan dengan *flywheel* maka perlu dirancang sebuah *software* yang mampu mengelola dan mengontrol data terhadap kinerja peralatan pengereman regeneratif. *Software* merupakan program berisi perintah-perintah yang dieksekusi oleh arduino mega 2560 sehingga sistem dapat bekerja sesuai dengan alur dan tujuan yang dirancang.

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 sederhana arduino.

Berikut ini algoritma program utama dari kontrol kecepatan dan pengereman regeneratif :

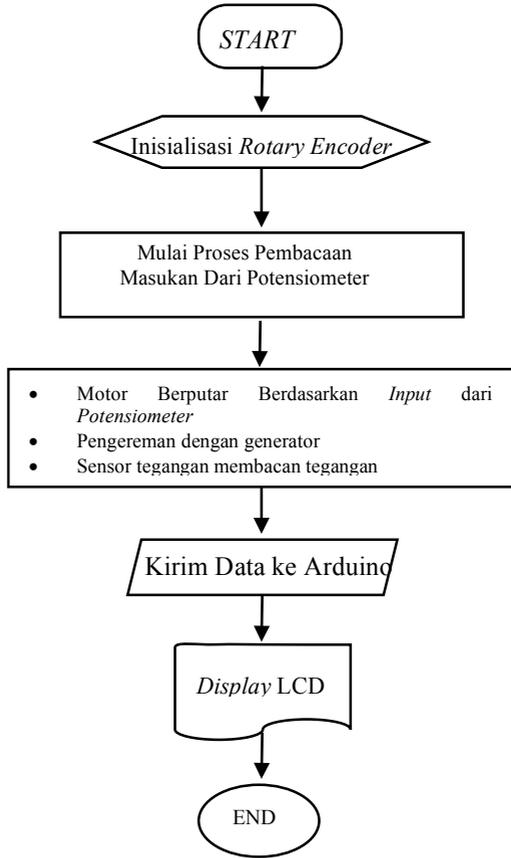
1. Modul dapat di operasikan setelah sistem terpasang dengan benar seperti motor brushless dc, *generator*, *rotary encoder* dan *flywheel*.
2. Modul dapat bekerja jika rangkaian kontrol sudah dijalankan dan sudah terpasang.
3. Pengereman regeneratif yang dilakukan oleh *generator* dc, akan menghasilkan nilai tegangan jika sistem kecepatan brushless bekerja memutar *flywheel* kemudian bergesekan dengan beban (*generator* DC) dan menjadikan energi kinetik menjadi energi listrik.

Pada Proyek Akhir ini *software* yang digunakan adalah program Arduino Mega 2560 untuk membuat dan

merencanakan program dalam bahasa C sederhana arduino. Pemrograman *software* arduino dirancang dengan menggunakan *software* yang bernama Arduino IDE dengan menggunakan bahasa pemrograman C sederhana arduino.

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 3.9 berikut menunjukkan *flowchart* yang digunakan pada Arduino Mega 2560 pada Proyek Akhir ini.



Gambar 3.10 *Flowchart* Arduino Mega

BAB IV

PENGUJIAN DAN ANALISA ALAT

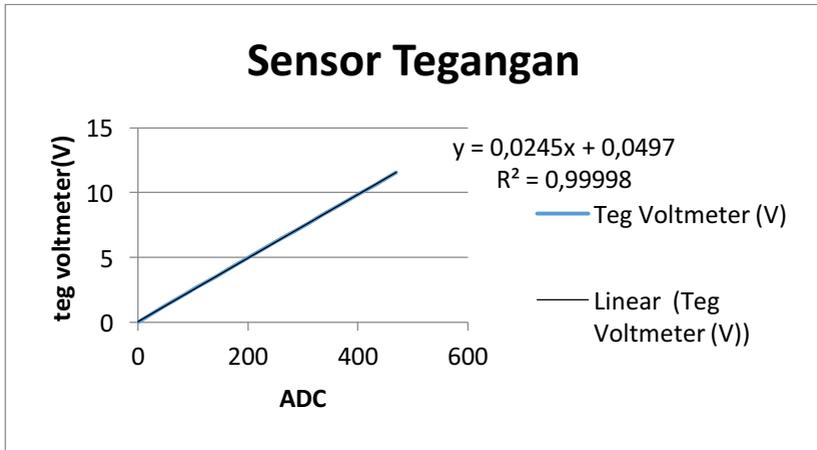
Pada bab ini akan dibahas mengenai pengujian alat dan analisa data dari hasil rancangan alat yang telah dibuat. Pengujian alat ini ditujukan untuk memastikan agar peralatan dapat berfungsi dengan baik.

Pengujian dan analisa dari alat ini meliputi analisa pengujian generator dan sensor tegangan. Setelah melakukan beberapa pengujian tersebut, data yang diperoleh akan dianalisa untuk mengetahui proses kerja dari seluruh sistem alat yang dibuat.

4.1. Pengujian Sensor Tegangan

Tabel 4.1 Pengujian Kalibrasi Sensor Tegangan

No.	ADC	Tegangan Voltmeter (V)
1	0	0
2	47	1,21
3	64	1,63
4	88	2,21
5	104	2,6
6	142	3,54
7	171	4,24
8	223	5,53
9	262	6,47
11	288	7,1
12	312	7,7
13	333	8,21
14	378	9,31
15	398	9,8
16	434	10,65
17	444	10,93
18	470	11,55



gambar 4.1 Grafik Perbandingan Nilai ADC dengan Tegangan Voltmeter

Tabel 4.1 dan gambar 4.1 menunjukkan nilai ADC sensor tegangan yang ditampilkan pada *LCD* dan dibandingkan dengan nilai tegangan yang terukur oleh voltmeter, sehingga didapatlah sebuah rumus $y = 0,0245x + 0,0497$ untuk menentukan nilai adc dalam satuan volt.

4.2. Pengujian Pengereman Regeneratif

Pengujian alat akan dilakukan dengan 3 mode pengereman. Yakni pengereman 1, atau pengereman yang dilakukan dengan 1 generator, pengereman 2, atau yang dilakukan dengan 2 generator secara bersamaan, pengereman 3, atau yang dilakukan dengan 3 generator secara bersamaan.

Tabel 4.2 Generator dan Sensor Tegangan Saat Kecepatan Motor 1500

No	Kecepatan awal (Rpm)	Tegangan Average (V)	Tegangan Max (V)	Kecepatan Akhir (Rpm)	Perlambatan (Rpm)
1	1474	5,49	9,60	838	636
2	1446	4,47	10,19	502	945
3	1454	1,83	6,44	241	1213

Tabel 4.3 Generator dan Sensor Tegangan Saat Kecepatan Motor 2000

No	Kecepatan awal (Rpm)	Tegangan Average (V)	Tegangan Max (V)	Kecepatan Akhir (Rpm)	Perlambatan (Rpm)
1	2015	5,76	9,26	1381	634
2	2005	8,44	11,27	979	1026
3	2022	4,92	10,29	559	1443

Tabel 4.4 Generator dan Sensor Tegangan Saat Kecepatan Motor 2500

No	Kecepatan awal (Rpm)	Tegangan Average (V)	Tegangan Max (V)	Kecepatan Akhir (Rpm)	Perlambatan (Rpm)
1	2494	8,69	16,86	1670	824
2	2471	9,41	13,57	1236	1235
3	2482	7,85	11,21	797	1685

Tabel 4.5 Generator dan Sensor Tegangan Saat Kecepatan Motor 3000

No	Kecepatan awal (Rpm)	Tegangan Average (V)	Tegangan Max (V)	Kecepatan Akhir (Rpm)	Perlambatan (Rpm)
1	3006	9,18	16,07	2357	649
2	2989	11,21	13,65	1957	1031
3	3009	9,68	12,20	1583	1426

Tabel 4.6 Generator dan Sensor Tegangan Saat Kecepatan Motor 3500

No	Kecepatan awal (Rpm)	Tegangan Average (V)	Tegangan Max (V)	Kecepatan Akhir (Rpm)	Perlambatan (Rpm)
1	3508	7,73	16,44	2935	573
2	3501	11,70	14,48	2684	817
3	3494	12,85	15,63	2030	1464

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

BAB V PENUTUP

Bab penutup ini berisi tentang kesimpulan yang diperoleh selama proses pembuatan modul kontrol kecepatan dan pengereman regeneratif motor *brushless* DC, kesimpulan dari hasil pengujian dan analisa data, serta saran untuk modul kontrol kecepatan dan pengereman regeneratif motor *brushless* DC ini kedepannya.

5.1. Kesimpulan

1. Pengereman yang dilakukan dengan 3 mode pengereman dapat menghasilkan perlambatan yang berbeda-beda dan tegangan yang relatif sama.
2. Pengereman yang dilakukan antara pengereman 1, pengereman 2, dan pengereman 3 terjadi perlambatan yang konstan.
3. Perlambatan yang didapatkan adalah antara 400 hingga 600 RPM dikarenakan 1 generator yang memiliki 100 hingga 240 RPM dan gesekan antara generator dengan flywheel.
4. Tegangan puncak (max) yang bisa didapatkan bisa mencapai 16,86V dan perlambatan max yang bisa mencapai 1685 RPM
5. Mode pengereman 2 dapat menghasilkan tegangan *average* lebih tinggi dari pada mode pengereman yang lainnya dikarenakan mendapatkan kecepatan yang lebih tinggi dari pada mode pengereman 3 ,dan menggunakan generator yang lebih banyak dari pada mode pengereman 1

5.2. Saran

1. Diharapkan kedepannya dapat memilih komponen-komponen yang dibutuhkan dengan lebih tepat. Karena selama perancangan alat yang lalu masih sering terjadi alat berfungsi tidak maksima.
2. Diharapkan menggunakan motor dc generetor yang lebih besar agar hasil tegangan yang didapatkan lebih jelas.
3. Diharapkan sistem pengereman yang otomatis untuk mencegah tabrakan dan lebih memudahkan pengguna.
4. 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] Hudaibiy Hibban. “*Desain Kontroler Fuzzy PID Gain Schedulling Untuk Pengaturan Kecepatan Motor DC Tanpa Sikat*”, **Tugas Akhir**, Jurusan Teknik Elektro FTI-ITS, Surabaya, 2015.
- [2] Bachtiar, Muhammad Fachri, Priyatna, Alif Gigah. “*Perancangan Rem Magnetik Pada Motor DC Dengan Menggunakan Arduino*”, Tugas Akhir, D3 Teknik Elektro, ITS Surabaya, 2015.
- [3] Ogata, Katsuhiko. “*Teknik Kontrol Automatik (Sistem Pengaturan) Jilid 1*”, Penerbit Erlangga, Jakarta, 1991.
- [4] Pramudijanto, Jos. Catatan Kuliah Penulisan Ilmiah, <http://www.ee.its.ac.id/~jos>, Jurusan Teknik Elektro ITS, 10 Januari 2016.
- [5] 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.

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

LAMPIRAN A ***LISTING PROGRAM***

LISTING PROGRAM ARDUINO MEGA 2560

```
#include "ESC.h"
#include "TimerThree.h"
#include <glcd.h>
#include "fonts/Arial14.h"
#include "fonts/SystemFont5x7.h"

#if DISPLAY_HEIGHT < 64
#error ks0108 example requires a display at least 64 pixels tall
#endif
#if DISPLAY_WIDTH < 128
#error ks0108 example requires a display at least 128 pixels wide
#endif

float kec ;
int adcTegangan,count;
float tegangan;
float Vmax,Vavg;
float Vdata[10];

#define POT_PIN (A2)

#define SKC (21)
#define Steg (A0)

double pulsa=0;
float kecepatan;
int x;

void isr(){
```

```

    kecepatan=(float)pulsa*60.0/36.0;
    if(kecepatan>0){
    kecepatan =(0.9866*kecepatan) + 5.8555;
    kecepatan = (kecepatan*1.0039) - 4.0476;}
    pulsa=0;

}

ESC myESC (7, 1000, 2000, 500);    // ESC_Name (PIN, Minimum Value,
Maximum Value, Arm Value)
int val;

void setup() {
    Timer3.initialize(1000000);
    Timer3.attachInterrupt(isr);
    myESC.arm();
    pinMode(SKC , INPUT_PULLUP);
    GLCD.Init(NON_INVERTED);
    GLCD.ClearScreen();

    GLCD.SelectFont(System5x7);
    attachInterrupt(2,rutinInterupsi0,RISING);

    Serial.begin(9600);
}
void loop(){
adcTegangan=analogRead(Steg);
if(adcTegangan>0){tegangan = (adcTegangan*0.0245)+0.0497;
    Vdata[9]=Vdata[8];
    Vdata[8]=Vdata[7];
    Vdata[7]=Vdata[6];
    Vdata[6]=Vdata[5];
    Vdata[5]=Vdata[4];
    Vdata[4]=Vdata[3];
    Vdata[3]=Vdata[2];
    Vdata[2]=Vdata[1];
    Vdata[1]=Vdata[0];
    Vdata[0]=tegangan;
}
}

```

```
Vavg=(Vdata[0]+Vdata[1]+Vdata[2]+Vdata[3]+Vdata[4]+Vdata[5]+Vdata[
6]+Vdata[7]+Vdata[8]+Vdata[9])/10.0;
    }
```

```
else {tegangan=0;}
if (Vmax<tegangan){Vmax=tegangan;}
if(count>4){
    GLCD.CursorTo(0,0);
    GLCD.Puts("ADC :");
    GLCD.Puts((String)adcTegangan);
    GLCD.Puts("      ");
    GLCD.CursorTo(0,1);
    GLCD.Puts("Teg :");
    GLCD.Puts((String)tegangan);
    GLCD.Puts(" Volt      ");
    GLCD.CursorTo(0,2);
    GLCD.Puts("Vmax:");
    GLCD.Puts((String)Vmax);
    GLCD.Puts(" Volt      ");
    GLCD.CursorTo(0,3);
    GLCD.Puts("Vavg:");
    GLCD.Puts((String)Vavg);
    GLCD.Puts(" Volt      ");
    GLCD.CursorTo(0,5);
    GLCD.Puts("S:");
    GLCD.Puts((String)kecepatan);
    GLCD.Puts("rpm      ");
    count=0;
}
val = analogRead(POT_PIN);
val = map(val, 0, 1023, 1000, 2000);
myESC.speed(val);
count++;
delay(100);
}
```

```
void rutinInterupsi0(){ pulsa++;}
```

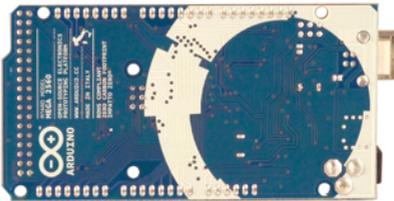
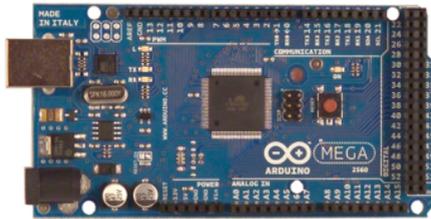
-- Halaman ini sengaja dikosongkan --

LAMPIRAN B DATASHEET

1. *Datasheet* Arduino Mega 2560



Arduino Mega 2560 Datasheet



Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino [Duemilavve](#) or [Diecimila](#).

Schematic & Reference Design

EAGLE files: [arduino-mega2560-reference-design.zip](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1 mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the **5V** and **GND** pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication using the [SPI library](#). The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, [Duesenauer](#) and [Diecimila](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH

- 
- **I²C: 20 (SDA) and 21 (SCL).** Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the [Duemilanove](#) or [Diecimila](#).

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega2560's digital pins.

The ATmega2560 also supports I²C (TWI) and SPI communication. The Arduino software includes a [Wire library](#) to simplify use of the I²C bus; see the [documentation on the Wiring website](#) for details. For SPI communication, use the [SPI library](#).

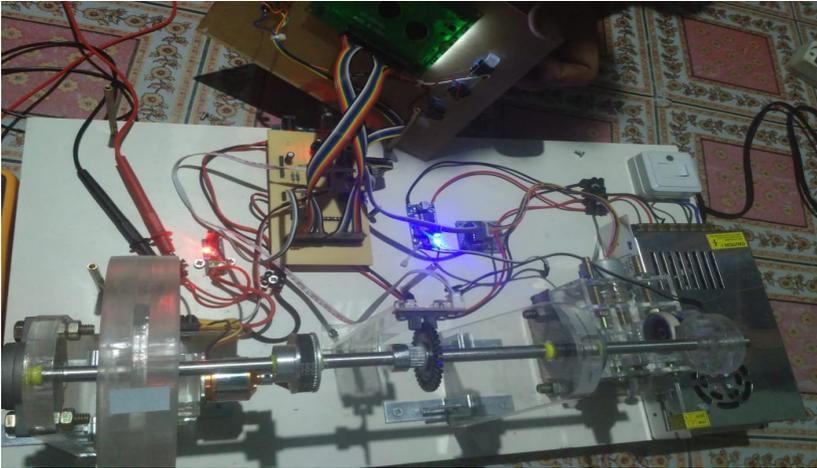
Programming

The Arduino Mega can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes [preburned](#) with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It

-- Halaman ini sengaja dikosongkan --

LAMPIRAN C DOKUMENTASI



-- Halaman ini sengaja dikosongkan --

RIWAYAT PENULIS



Nama : Muhammad Dani
TTL : Gresik, 23 Juni 1997
Jenis Kelamin : Laki - Laki
Agama : Islam
Alamat Asal : JL. Flamboyan no 19 Perum
BP Wetan Gresik
Telp/HP : 081333599133
E-mail : *danimuhammad131@gmail.com*

RIWAYAT PENDIDIKAN

- *2003 – 2009 : SD Muhammadiyah 2 Gresik*
- *2009 – 2012 : MTs PPMI Assalam Sukoharjo*
- *2012 – 2015 : SMA PPMI Assalaam Sukoharjo*
- *2015 – Sekarang: Departemen Teknik Elektro Otomasi Institut Teknologi Sepuluh Nopember.*

PENGALAMAN KERJA

- *Kerja Praktek PT SEMEN INDONESIA TUBAN (Juni 2017 – Juli 2017)*
- *Kerja Praktek PT. PLN (persero) Distribusi Jawa Timur Rayon (Juli 2017 – Agustus 2017)*