

**KENDALI MEDAN EKSTASI DENGAN ARDUINO UNO PADA
EMULATOR PEMBANGKIT LISTRIK BIOMASSA**

(Skripsi)

Oleh

AMIRUDIN



FAKULTAS TEKNIK

UNIVERSITAS LAMPUNG

BANDAR LAMPUNG

2019

ABSTRAK

KENDALI MEDAN EKSITASI DENGAN ARDUINO UNO PADA EMULATOR PEMBANGKIT LISTRIK BIOMASSA

Oleh

AMIRUDIN

Kontrol medan eksitasi pada *emulator* pembangkit listrik biomassa merupakan alat yang digunakan untuk mengatur masukan arus DC pada kumparan medan rotor sehingga generator dapat membangkitkan tegangan. Pembangkitan ini diakibatkan karena adanya ggl induksi antara kumparan medan dan kumparan jangkar, selain itu alat ini juga dapat meniru karakteristik respon medan eksitasi pembangkit listrik biomassa.

Penelitian ini bertujuan untuk merancang dan mengimplimentasikan kontrol medan eksitasi dengan menggunakan rangkaian eletronika daya *buck converter* sedangkan untuk pengaturan dan pemantauan kontrol medan eksitasi generator digunakan mikrokontroler arduino uno dengan *PID controller*.

Berdasarkan hasil pengujian, kontrol medan eksitasi yang dibuat dapat menirukan karakteristik dari kontrol medan eksitasi pembangkit listrik biomassa dengan waktu mencapai *steady state* yang sama yaitu detik ke 4 dari *starting* generator kemudian pada pengujian variasi nilai beban yang tersambung pada generator untuk mengetahui respon kontrol medan eksitasi terhadap tegangan keluaran generator. Beban induktif menyebabkan *drop* tegangan lebih besar dibandingkan beban resistif hal ini dikarenakan memiliki faktor daya yang lebih kecil daripada beban resistif sehingga terjadi rugi-rugi daya yang lebih besar.

Kata kunci — *Emulator* Pembangkit Listrik Biomassa , *Eksitasi Generator*, *Arduino Uno*, *Kontrol PID*, *Buck Converter*.

ABSTRACT

EXCITATION FIELD CONTROL WITH ARDUINO UNO IN BIOMASS POWER PLANT EMULATOR

By

AMIRUDIN

Excitation field control on the emulator of the biomass power plant is a tool used to regulate the input of the DC current, on the rotor field winding so that the generator can generate a voltage. This generation is caused by the induction emf between the field winding and the armature winding, besides that this tool can also mimic the response characteristics of the excitation field of the biomass power plant.

The purposes of the research are to design and implement excitation field control using a buck converter while for regulating and monitoring the excitation field control of the generator an Arduino Uno microcontroller is used with a PID controller.

From experiment results, the excitation field control that is mimicking the characteristics of the excitation field control of the biomass power plant with the same steady-state time when the 4th seconds from the starting generator. Then in a test the load value variation which connected to the generator to find out the excitation field control response to generator output voltage. Inductive load causes a voltage drop to be greater than the resistive load because it has a smaller power factor than the resistive load, resulting in greater power losses.

Keywords - Biomass Power Plants Emulator, excitation generator, Arduino Uno, PID control, Buck Converter.

**KENDALI MEDAN EKSITASI DENGAN ARDUINO UNO PADA
EMULATOR PEMBANGKIT LISTRIK BIOMASSA**

**Oleh
AMIRUDIN**

Skripsi

**Sebagai Salah Satu Syarat Mencapai Gelar
SARJANA TEKNIK**

Pada

**Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2019

Judul Skripsi : **KENDALI MEDAN EKSTIASI DENGAN ARDUINO UNO PADA EMULATOR PEMBANGKIT LISTRIK BIOMASSA**

Nama Mahasiswa : Amirudin

Nomor Pokok Mahasiswa : 1415031016

Program Studi : Teknik Elektro

Fakultas : Teknik

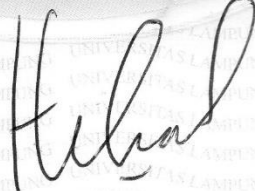
MENYETUJUI

1. Komisi Pembimbing


Khairudin, S.T., M.Sc., Ph.D.Eng
NIP. 197007192000121001


Dr. Eng. Lukmanul Hakim, S.T., M.Sc
NIP. 19720923 200012 1 002


2. Ketua Jurusan Teknik Elektro


Dr. Herman Halomoan Sinaga, S.T., M.T.
NIP. 19711130 199903 1 003

MENGESAHKAN

1. Tim Penguji


Ketua : Khairudin, S.T., M.Sc., Ph.D.Eng. 

Sekretaris : Dr. Eng. Lukmanul Hakim, S.T., M.Sc. 

Penguji Utama : Herri Gusmedi, S.T., M.T. 

2. Dekan Fakultas Teknik



Prof. Drs. Suharno, M.Sc., Ph.D.
NIP. 19620717 198703 1 002 

Tanggal Lulus Ujian Skripsi : 7 Oktober 2019

SURAT PERNYATAAN

Dengan ini saya memberi pernyataan bahwa dalam skripsi ini tidak ada terdapat karya yang pernah dilakukan orang lain dan sepanjang sepengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah ini sebagaimana yang disebutkan di dalam daftar pustaka. Selain itu, saya menyatakan pula bahwa skripsi ini dibuat oleh saya sendiri.

Apabila pernyataan saya tidak benar, maka saya bersedia dikenakan sanksi sesuai hukum yang berlaku.

Bandar Lampung, 18 Oktober 2019



Amirudin
NPM. 1415031016

RIWAYAT HIDUP



Penulis dilahirkan di Lampung Timur, Provinsi Lampung pada tanggal 18 Desember 1996. Penulis merupakan anak keempat dari empat bersaudara dari pasangan Bapak Budiono dan Ibu Runtahati.

Mengenai riwayat pendidikan, penulis lulus Sekolah Dasar di SDN 4 Rajabasa Lama pada tahun 2009, lulus Sekolah Menengah Pertama di SMPN 1 Labuhan Ratu pada tahun 2012, lulus Sekolah Menengah Atas di SMAN 1 Way Jepara pada tahun 2014, dan diterima di jurusan Teknik Elektro Universitas Lampung pada tahun 2014 melalui Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN).

Selama menjadi mahasiswa, penulis aktif diberbagai organisasi internal dan eksternal kampus. Selama 2 kepengurusan menjadi anggota Departemen Pendidikan dan Pengembangan Diri Himpunan HIMATRO Periode 2015-2017, menjadi Sekrearis Umum FOSSI-FT Periode 2015-2016 dan Periode 2016, menjadi Ketua Umum Ikam Lamtim Periode 2016-2017, menjadi Wakil Ketua 1 DPM U KBM Universitas Lampung periode 2018, menjadi anggota Jaringan dan Mitra (Jarmit) Puskomda FSLDK Lampung Periode 2018, Koordinator Humas One Day One Juz (ODOJ) Lamtim Periode 2018-2020 dan Sekretaris

Wilayah Kampus Komite Nasional Rakyat Palestina (KNRP) Provinsi Lampung periode 2018-2020. Pada tanggal 24 Juli – 24 Agustus 2017 penulis melaksanakan Kerja Praktik (KP) di PT. Bukit Asam Tarahan dengan Judul “Sistem Proteksi Transformator Daya 6,3 kV / 20 kV Pada PLTU 2 X 8 MW PT. Bukit Asam (Persero). Tbk., Unit Pelabuhan Tarahan”.

PERSEMBAHAN



Dengan Mengharap Ridho Allah. SWT, Shalawat beriring salam kepada Nabi Muhamad SAW. Kupersembahkan Karya tulis ini sebagai wujud rasa

cinta dan terimakasihku untuk:

Ayah dan Ibuku Tercinta
Budiono dan Runtahati

Kakak-kakak ku tersayang
Imam Yulianto
Siti Insyah
Ahmad Sodikin

Teman-teman dan keluarga besar tercinta

Almamaterku
Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung

Bangsa dan Negaraku
Negara Kesatuan Republik Indonesia

Trimakasih untuk segalanya

Motto

*“Karena sesungguhnya sesudah kesulitan itu ada kemudahan , Sesungguhnya
sesudah kesulitan itu ada kemudahan.”*

(QS. Alam Nasyroh: 5-6)

خَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

“Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia”

(HR. Ahmad, ath-Thabrani, ad-Daruqutni. Hadits ini dihasankan oleh al-Albani di
dalam *Shahihul Jami'* no:3289).

Masa depan adalah milikmu wahai para pemuda, maka didik masa muda mu
dengan hal-hal kebaikan sebagai bekal mu nanti

(Anonim)

SANWACANA

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayahnya kepada penulis, sehingga dapat terselesaikannya tugas akhir ini. Shalawat serta salam disanjungkan kepada Nabi Muhammad SAW yang dinantikan syafaatnya dihari akhir kelak.

Skripsi yang berjudul “**Kendali Medan Eksitasi dengan Arduino Uno pada Emulator Pembangkit Listrik Biomassa**” digunakan sebagai salah satu syarat memperoleh gelar sarjana di jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.

Selama penelitian dan perkuliahan penulis banyak mendapatkan pengalaman yang sangat berharga. Penulis juga telah mendapat bantuan baik moril, materil, bimbingan, petunjuk serta saran dari berbagai pihak baik secara langsung maupun tidak langsung. Untuk itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Ir. Hasriadi Mat Akin, M.P. sebagai Rektor Universitas Lampung.
2. Bapak Prof. Suharno, M.Sc., Ph.D. sebagai Dekan Fakultas Teknik Universitas Lampung

3. Bapak Dr. Herman Halomoan S, S.T., M.T. sebagai Kepala Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.
4. Bapak Khairudin, S.T., M.Sc., Ph.D.Eng. sebagai dosen pembimbing utama. Terimakasih atas kesedian waktunya untuk membimbing dan memberikan ilmu, serta kesabaran selama penulis menjadi mahasiswa Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung selama mengerjakan tugas akhir ini.
5. Bapak Dr. Eng. Lukmanul Hakim, S.T., M.Sc. Sebagai dosen pembimbing kedua. Terimakasih atas waktu, nasihat dan bimbinganya selama penulis menjadi mahasiswa Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung dan selama mengerjakan tugas akhir ini.
6. Bapak Herri Gusmedi, S.T., M.T. sebagai dosen penguji tugas akhir. Terimakasih atas masukan guna membuat tugas akhir ini menjadi lebih baik lagi.
7. Semua Dosen Teknik Elektro, Terimakasih atas bimbingan yang telah diberikan selama menuntut ilmu di Jurusan Teknik Elektro Universitas Lampung.
8. Pak Abdurachman Effendi, S.T., M.T.I. yang telah membantu penulis dalam meminjamkan peralatan untuk melakukan penelitian serta suntikan semangat yang diberikan pada penulis selama di Laboratorium Sistem Tenaga Elektrik Universitas Lampung.
9. Seluruh Staf Jurusan Teknik Elektro Universitas Lampung atas bantuannya dalam hal adminitrasi dan hal lainnya semasa penulis kuliah.

10. Bapak dan Ibu tercinta, bapak Budiono dan Ibu Runtahati dirumah yang selalu berusaha sekuat tenaga untuk memperjuangkan penulis mengenyam pendidikan setinggi mungkin, yang tak pernah lelah mendukung dan mendo'akan penulis.
11. Imam Yulianto, Siti Insyah dan Ahmad Sodikin selaku kakak yang selalu mendukung penulis dalam menempuh pendidikan.
12. Segenap teman-teman Laboratorium Sistem Tenaga Elektrik, Fandi, Fitra, Ferdian, Ruri, Arif, Jeshu, Ebot, Septi, Rafi, Ridwan, Pandji, Yupida, Naufal dan Redo.
13. Kawan-kawan Ikam Lamtim Adi, Dedi, Ambar, Yekti, Fitria, Wahyu, Shintia, Dwi Wahyu, Soni, Fikri, Adit , Widit dan seluruh anggota Ikam Lamtim 2016-2017 yang telah berjuang bersama dalam suka maupun duka berposes di Ikam Lamtim.
14. FOSSI FT periode 2015-2016 dan 2016, Kak rian, mbak endah, Kak Hanif, Kak Agus, Mbak Anggun, Kak Imbron, Mbak Lalili, Kak Triban, Fitra, Dona, Edi, Heru, Farhan, Dendi, Irvan, Ridwan yang telah memberikan banyak ilmu dan semangat persaudaraan menjalani amanah.
15. Rekan-rekan Elite 2014 Edi, Dikson, Cahya, Asep, Nyoman, Andri, Anggi, Bangkit, Arya, Rudi, Iqbal, Hayu, Dimas, Agung yang tak bisa disebutkan satu-persatu atas kebersamaan selama menempuh pendidikan di Jurusan Teknik Elektro Unila dan semoga kawan-kawan diberikan kesuksesan dikemudian hari.
16. Keluarga MPM/DPM U KBM Unila, Herwan, Bagus, Yudha, Dona, Ridho, Azri, Beni, Erssa, Aning, Ambar, Dinati, Arini, Syahrul, Elyas, Abduh atas semangat, ilmu, canda tawa dan persaudaraan.

17. Keluarga besar Masjid Al Awwal kedaton Bandar Lampung, mas Sholihin, pak Khulhaq, Mas Rohim, Pak Aras, Pak Karim, Bang Iwan atas kesediaan tempat tinggal, kebersamaan dan bimbingan selama ini
18. Marbot masjid Al Awwal Redi, Untung dan Juliansyah atas kebersamaan dalam susah atau senang mengurus masjid dan semoga kalian menjadi orang-orang sukses.
19. Semua pihak yang tidak dapat sepenuhnya penulis sebutkan satu persatu yang turut membantu penulis dalam menyelesaikan tugas akhir.

Semoga Allah SWT dapat membalas semua kebaikan pada semua pihak yang telah membantu. Penulis sangat menyadari bahwa skripsi ini masih banyak kesalahan dan jauh dari kesempurnaan, oleh karena itu saran dan kritik yang membangun sangat penulis harapkan demi perbaikan dimasa yang akan datang. Akhirnya, semoga skripsi ini dapat bermanfaat bagi kita semua. Aamiin.

Bandar Lampung, Oktober 2019
Penulis

Amirudin
NPM. 1415031016

DAFTAR ISI

	Halaman
DAFTAR ISI.....	xvi
DAFTAR GAMBAR	xx
DAFTAR TABEL.....	xxiii
BAB 1. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Tujuan.....	2
1.3. Rumusan Masalah	2
1.4. Batasan Masalah.....	2
1.5. Manfaat Penelitian.....	3
1.6. Hipotesis	3
1.7. Sistematika Penulisan.....	3
BAB 2. TINJAUAN PUSTAKA	5
2.1. Pembangkit Listrik Tenaga Biomassa	5
2.2. Buck Converter.....	6
2.2.1. Prinsip Kerja <i>Buck Converter</i>	7
2.2.2. Penentuan LC	8

2.3.	Generator Sinkron	9
2.3.1.	Konstruksi Generator Sinkron	10
2.3.2.	Pinsip Kerja Generator Sinkron	11
2.3.3.	Karakteristik Generator Sinkron	12
2.4.	Sistem Eksitasi	15
2.4.1.	Jenis-Jenis Sistem Eksitasi.....	16
2.5.	AVR (<i>Automatic Voltage Regulator</i>)	18
2.5.1.	Pengertian.....	18
2.5.2.	Prinsip Kerja	18
2.6.	PID <i>Controller</i>	19
2.6.1.	Pengendali <i>Proportional</i> (P)	19
2.6.2.	Pengendali <i>Integral</i> (I)	19
2.6.3.	Pengendali <i>derivative</i> (D)	20
2.6.4.	Pengendali <i>Propotional Integral Derivatif</i> (PID)	21
2.7.	<i>PulseWidth Modulation</i> (PWM).....	22
2.8.	Arduino Uno.....	23
2.9.	Sensor Tegangan	25
2.10.	<i>Gate Driver</i>	26
2.11.	<i>Power Supply</i>	27
BAB 3.	METODE PENELITIAN	28
3.1.	Waktu dan Tempat Penelitian	28

3.2.	Alat dan Bahan	28
3.3.	Tahapan Penelitian	29
3.3.1.	Studi Literatur	30
3.3.2.	Konsep Perancangan Sistem	30
3.3.3.	Perancangan Sistem	31
3.3.4.	Perencanaan Model Sistem	32
3.3.5.	Perancaman kerja sistem	32
3.3.6.	Analisis Data dan Kesimpulan	35
3.3.7.	Penulisan Laporan	35
BAB 4.	HASIL DAN PEMBAHASAN	36
4.1.	Hasil Rancang Kotrol Medan Eksitasi Bangun <i>Emulator</i> Pembangkit Listrik Tenaga Biomass.....	36
4.1.1.	Realisasi Kontrol Medan Eksitasi <i>Emulator</i> Pembangkit Listrik Tenaga Biomassa	36
4.1.2.	Realisasi Perancangan Alat Kontrol Medan Ekstiasi pada <i>Emulator</i> Pemabgnkit Listrik Biomassa	39
4.2.	Hasil Pengujian Perangkat Keras	46
4.2.1.	Hasil Pengujian Rangkaian PWM (<i>Pulse Width Modulation</i>).....	47
4.2.2.	Hasil Pengujian Rangkaian <i>Gate Driver</i>	48
4.2.3.	Pengujaaan Sensor Tegangan AC.....	49
4.2.4.	Hasil Pengujian Perangkat Keras <i>Buck Converter</i>	51

4.3.	Hasil Data Kontrol Medan Ekstasi dengan Arduino Uno pada <i>Emulator</i> Pembangkit Listrik Tenaga Biomassa.....	52
4.3.1.	Perubahan Beban dan Respon pada Sistem Eksitasi.....	53
4.3.2.	Hubungan Daya Beban Dan Tegangan Terminal Pada Eksitasi Generator.....	55
4.3.3.	Pengaruh Eksitasi Generator Terhadap Perubahan Tegangan Terminal	56
4.4	Pembahasan	62
4.4.1	Pola respon eksitasi pembangkit listrik biomassa	62
4.4.2	Keluaran Kontrol Medan Eksitasi pada <i>Emulator</i> Pembangkit Listrik Biomassa	63
BAB 5.	PENUTUP	65
5.1.	Kesimpulan.....	65
5.2.	Saran.....	66
	DAFTAR PUSTAKA	67
	LAMPIRAN.....	69

DAFTAR GAMBAR

Gambar	Halaman
Gambar 2.1 Proses Konversi PLTBm.....	6
Gambar 2.2 Rangkaian ekuivalen Buck Converter.....	6
Gambar 2.3 Rangkaian ekuivalen <i>buck converter</i> pada saat.....	7
Gambar 2.4 Generator Sinkron	10
Gambar 2.5 Kontruksi Generator Sinkron	11
Gambar 2.6 Rangkaian ekuivalen generator sinkron tanpa beban.....	13
Gambar 2.7 Grafik hubungan arus penguat medan (I_f) dan E_0	13
Gambar 2.8 Rangkaian ekuivalen generator sinkron berbeban	14
Gambar 2.9 Karakteristik generator AC Pada berbagai faktor daya.....	15
Gambar 2.10 Sistem Eksitasi Statis	16
Gambar 2.11 Sistem Eksitasi Dinamis.....	17
Gambar 2.12 Sistem Eksitasi Tanpa Menggunakan Sikat	17
Gambar 2.13 Sistem Kerja AVR (<i>Automatic Voltage Regulator</i>)	18
Gambar 2.14 <i>Pulse Width Modulation</i>	23
Gambar 2.15 Arduino Uno R3 (arduino.cc)	24
Gambar 2.16 Sensor tegangan ZMPT101B, (a) Tampilan fisik dan (b) Rangkaian penggunaan sensor	25
Gambar 2.17 Rangkaian <i>gate driver</i> berbasis HCPL3120.....	26

Gambar 2.18 Rangkaian <i>power supply</i>	27
Gambar 3.1 Diagram alir penelitian.....	29
Gambar 3.2 Diagram alir perancangan sistem	31
Gambar 3.3 Blok diagram perencanaan alat	32
Gambar 3.4 Rancangan <i>hardware</i> dari kontrol medan eksitasi pada <i>emulator</i> pembangkit listrik tenaga biomassa	34
Gambar 4.1 Realisasi Kontrol Medan Eksitasi dengan Arduino Uno pada <i>Emulator</i> Pembangkit Listrik Tenaga Biomassa.....	37
Gambar 4.2 Rangkaian <i>gate driver</i>	39
Gambar 4.3 <i>layout PCB gate driver</i>	40
Gambar 4.4 Perangkat keras <i>gate driver</i>	41
Gambar 4.5 sensor tegangan ZMPT10B	41
Gambar 4.6 Grafik pengujian sensor tegangan AC	50
Gambar 4.7 Grafik pengujian <i>buck converter</i>	52
Gambar 4.8 (A) Hubungan beban dan tegangan eksitasi (B) Hubungan beban dengan arus eksitasi.....	54
Gambar 4.9 Hubungan tegangan terminal terhadap penambahan beban.....	55
Gambar 4.10 Pengujian beban induktif dengan menggunakan motor 129 Watt dan $\cos \phi$ 0.6 dan Pengujian induktif dengan menggunakan.....	56
Gambar 4.11 Pengujian Beban lampu pijar 590 Watt.....	58
Gambar 4.12 Pengujian dengan variasi nilai induktor	59
Gambar 4.13 Pengujian dengan menggunakan beban motor 130 Watt dan beban Lampu 630 Watt.....	61
Gambar 4.14 Karakteristik eksitasi Pembangkit Listrik Tenaga Biomassa.....	62

Gambar 4.15 Karakteristik eksitasi generator pada emulator pembangkit listrik biomassa..... 63

Gambar 4.16 Perbandingan karakteristik respon alat kontrol medan eksitasi pada emulator pembangkit listrik biomassa dan Respon eksitasi generator yang ditiru 64

DAFTAR TABEL

Tabel	Halaman
Tabel 2.1 Spesifikasi Arduino Uno R3	24
Tabel 4.1 Spesifikasi Mosfet IRFP460	43
Tabel 4.2 Pengujian sensor tegangan AC	50
Tabel 4.3 Pengujian Buck Converter	51
Tabel 4.4 Hasil respon perubahan beban terhadap sistem eksitasi	53

BAB 1. PENDAHULUAN

1.1. Latar Belakang

Pembangkit listrik tenaga biomassa merupakan salah satu dari pembangkit listrik terbarukan yang berkembang pada saat ini. Penggunaan pembangkit listrik biomassa sebagai salah satu solusi untuk mendapatkan energi listrik dengan memanfaatkan bahan bakar biomassa seperti limbah perkebunan kelapa sawit, perkebunan tebu dan limbah hasil hutan.

Dalam menganalisa kinerja dari pembangkit listrik biomassa perlu dibuat tiruan pembangkit listrik biomassa. Tiruan atau mimik dari pembangkit listrik biomassa ini disebut dengan *emulator* pembangkit listrik biomassa.

Pada pembangkit listrik biomassa disisi generator dibutuhkan sistem eksitasi yang berfungsi untuk memberikan arus penguat (I_f) pada kumparan medan generator dengan cara membangkitkan medan magnetnya dengan bantuan arus searah sehingga adanya medan magnet pada rotor dan ketika rotor diputar maka akan terjadi ggl induksi pada stator yang menyebabkan generator dapat mengeluarkan tegangan. Pada sistem ini diperlukan juga AVR (*Automatic Voltage Regulator*) sebagai alat yang mengatur eksitasi generator akan menstabilkan tegangan keluaran pada generator yang diakibatkan perubahan beban.

Pada tugas akhir ini tiruan dari eksitasi pada pembangkit listrik biomassa dengan meniru karakteristik respon eksitasi dari pembangkit listrik biomassa yaitu dengan komponen elektronika daya yang dikontrol dengan menggunakan mikrokontroler Arduino Uno sehingga alat yang dibangun dapat menyetabilkan tegangan nominal secara otomatis yang diakibatkan perubahan beban.

1.2. Tujuan

Tujuan dari pengerjaan tugas akhir ini yaitu Rancang bangun medan kontrol eksitasi *emulator* pembangkit listrik tenaga biomassa yang menjadi bagian dari *emulator* sistem *smartgrid*.

1.3. Rumusan Masalah

Rumusan masalah pada penelitian tugas akhir ini adalah :

1. Bagaimana pengontrolan medan eksitasi generator sinkron sehingga dapat meniru karakteristik dari eksitasi pembangkit listrik biomassa.
2. Bagaimana masukan tegangan ke arduino uno sehingga mempengaruhi dari keluran medan eksitasi generator.

1.4. Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Pada penelitian ini frekuensi pada tegangan keluaran pembangkit dianggap konstan yaitu 50 Hz
2. Pengendalian menggunakan kontroler PID
3. Pengambilan data menggunakan listrik AC 1 Fasa yaitu 220 V.

1.5. Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk mengetahui kinerja kontrol medan eksitasi pada pembangkit listrik tenaga biomassa yang ditirukan pada kontrol medan eksitasi pada *emulator* pembangkit listrik tenaga biomassa dan mengetahui karakteristik respon pengontrolan eksitasi berdasarkan variasi perubahan nilai beban.

1.6. Hipotesis

Alat ini akan meniru dari kinerja eksitasi pada pembangkit listrik tenaga biomassa dengan tegangan nominal yang stabil pada setiap perubahan nilai beban secara otomatis.

1.7. Sistematika Penulisan

1. BAB 1. PENDAHULUAN

Pada bab ini menjelaskan latar belakang, masalah, tujuan tugas akhir, manfaat tugas akhir, perumusan masalah, batasan masalah, hipotesis, dan sistematika penulisan.

2. BAB 2. TINJAUAN PUSTAKA

Pada bab ini memaparkan beberapa teori pendukung dan referensi materi tugas akhir yang diambil dari berbagai sumber buku dan penelitian ilmiah yang digunakan dalam penulisan laporan tugas akhir ini.

3. BAB 3. METODOLOGI PENELITIAN

Pada bab ini memaparkan waktu dan tempat, alat dan bahan, metode yang digunakan, dan pelaksanaan serta pengamatan dalam pengerjaan tugas akhir.

4. BAB 4. HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan hasil data simulasi dan pembahasan dari keluaran alat tersebut.

5. BAB 5. KESIMPULAN

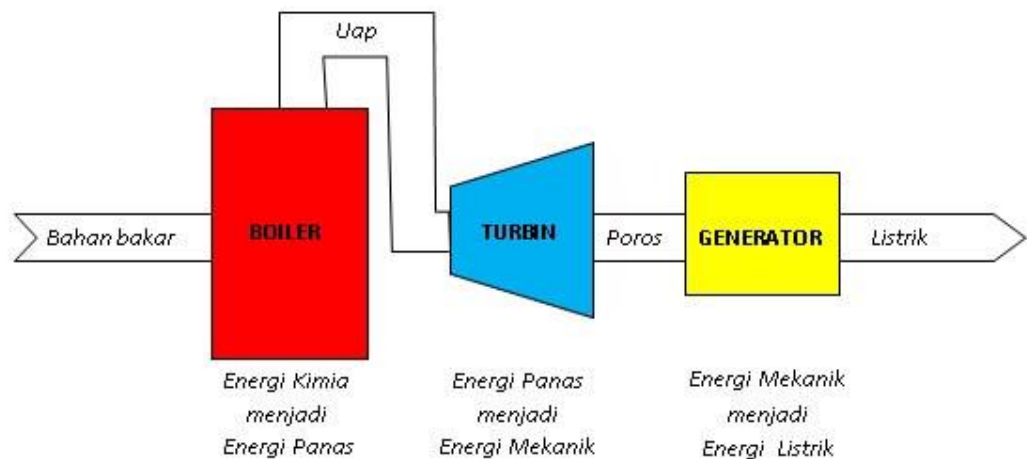
Pada bab ini menjelaskan kesimpulan yang didasarkan pada hasil data dan pembahasan dari kendali medan eksitasi dengan arduino pada *emulator* pembangkit listrik biomassa.

BAB 2. TINJAUAN PUSTAKA

2.1. Pembangkit Listrik Tenaga Biomassa

Pembangkit listrik tenaga biomassa (PLTBm) merupakan pembangkit listrik menggunakan konsep pembangkit listrik tenaga uap (PLTU), pembangkit ini menggunakan bahan bakar untuk proses pembakaran yang kemudian sebagai pemanas air yang berada di boiler.

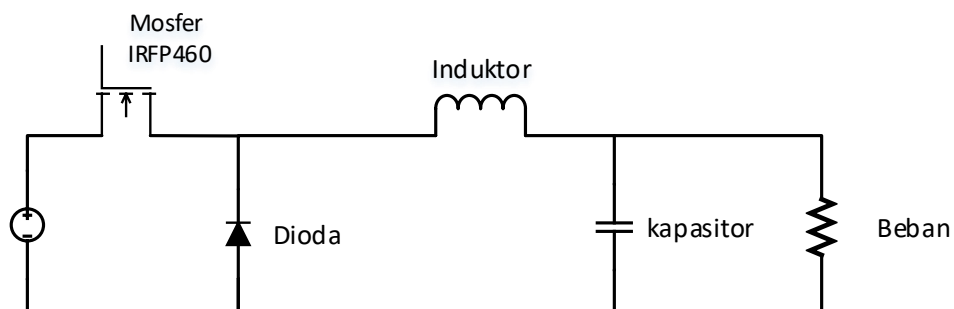
Proses ini berlangsung pada ruang bakar dari ketel uap PLTBm. Energi panas kemudian dipindahkan ke dalam air yang ada didalam pipa ketel untuk mendapatkan uap yang nantinya akan dikumpulkan dalam drum dari ketel. Uap yang berada didalam ketel kemudian dialirkan ke turbin uap, yang akan dikonversikan menjadi energi mekanis penggerak generator sinkron, dari gerakan mekanis dari generator sinkron akan dengan penambahan medan eksitasi untuk membangkitkan medan magnet pada rotor generator sehingga akan terjadi ggl induksi pada yang akan membangkitkan tenaga listrik



Gambar 2.1 Proses Konversi PLTBm

2.2. Buck Converter

Buck Converter merupakan konverter yang memiliki sifat memperkecil tegangan masukannya. Tegangan yang dihasilkan memiliki polaritas yang sama dengan tegangan masukannya. *Buck Converter* biasa disebut dengan *step-down converter*. Skema dengan *buck converter* topologi DC/DC diterapkan untuk sistem eksitasi dari generator sinkron. [1]. Gambar 2.2 menunjukkan rangkaian ekuivalen dari *Buck Converter*.

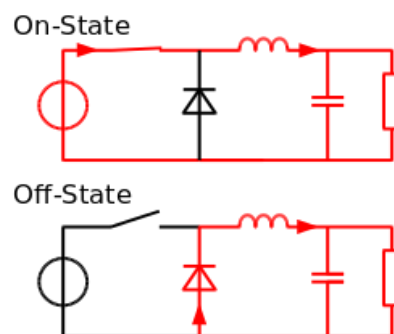


Gambar 2.2 Rangkaian ekuivalen *buck converter*.

Buck Converter merupakan salah satu jenis dari topologi dari *switching power supply* dengan 2 bagian yaitu *power* dan kontrol. Bagian *power* berfungsi untuk konversi tegangan, termasuk komponen-komponen didalamnya, seperti, *switch* dan *filter output*. Bagian kontrol memiliki fungsi untuk kontrol *state* ON-OFF dari *switch* pada rangkaian [2].

2.2.1. Prinsip Kerja *Buck Converter*

Buck converter menggunakan 2 state, yaitu state ON dan state OFF. Pada saat state ON atau saklar pada kondisi ON, arus sumber akan mengalir ke induktor menuju keluaran beban resistor dan kapasitor sampai tegangan keluaran mendekati tegangan masukan. Saat keadaan state OFF atau saklar kondisi OFF, polaritas berbalik yang menyebabkan energi yang tersimpan pada induktor akan mengalir terbalik berdasarkan tegangan yang terdapat pada kapasitor sampai terjadi pengurangan pada kapasitor. Proses pengisian serta pengosongan akan menyebabkan tegangan keluaran selalu lebih rendah dari tegangan masukannya. Berikut adalah state dari rangkaian ekuivalen *buck converter* pada saat state ON dan state OFF:



Gambar 2.3 Rangkaian ekuivalen *buck converter* pada saat state ON dan state OFF

Secara matematis, waktu dari *state ON* dan *state OFF* dapat di rumuskan sebagai berikut :

$$T_{on} = D \times T_s \quad (2.1)$$

$$T_{OFF} = (1 - D) \times T_s \quad (2.2)$$

2.2.1.1. State ON

Pada saat arus dari sumber V_{in} mengalir ke induktor yang disebabkan saklar akan menutup dan arus yang melalui induktor meningkat secara linier karena tegangan menuju ke induktor konstan seperti pada Gambar 2.3 diatas.

2.2.1.2. State OFF

Ketika saklar terbuka yang menyebabkan arus tidak mengalir ke MOSFET, sehingga sumber tegangan keluaran berasal dari induktor dan kapasitor sehingga dioda menjadi aktif. Arus akan mengalir dari induktor ke beban melewati dioda dan akan kembali menuju induktor. Dikarenakan tegangan induktor menjadi lebih kecil dibandingkan dengan state ON dan konstan, sehingga arus akan turun secara linier.

2.2.2. Penentuan LC

Menentukan besarnya kapasitor dan induktor pada rangkaian *buck converter* dapat dilakukan dengan persamaan berikut ini :

- Menentukan besar induktor

$$\Delta I_L = \frac{V_{in}D(1-D)}{fL} \quad (2.3)$$

$$L = \frac{V_{in}D(1-D)}{f\Delta I_L} \quad (2.4)$$

Keterangan :

L = Nilai induktor(H)

V_{in} = Tegangan *input* (V)

D = *Duty cycle*

ΔI_L = *Ripple* arus

f = Frekuensi (Hz)

Tegangan keluaran sistem pada kenyataannya tidak dapat bernilai sama secara sempurna. Hal ini disebabkan kapasitor yang terdapat pada rangkaian akan terus melakukan pengisian dan pelepasan muatan. Pada tegangan keluaran sistem V_o akan terdapat *ripple* tegangan (ΔV). Dalam penentuan besarnya *ripple* tegangan dapat menggunakan persamaan berikut ini :

$$\Delta V = \frac{V_{in}D(1-D)}{8LCf^2} \quad (2.5)$$

Sehingga dalam menentukan besar kapasitor dapat menggunakan persamaan berikut :

$$C = \frac{V_{in}D(1-D)}{8L\Delta Vf^2} \quad (2.6)$$

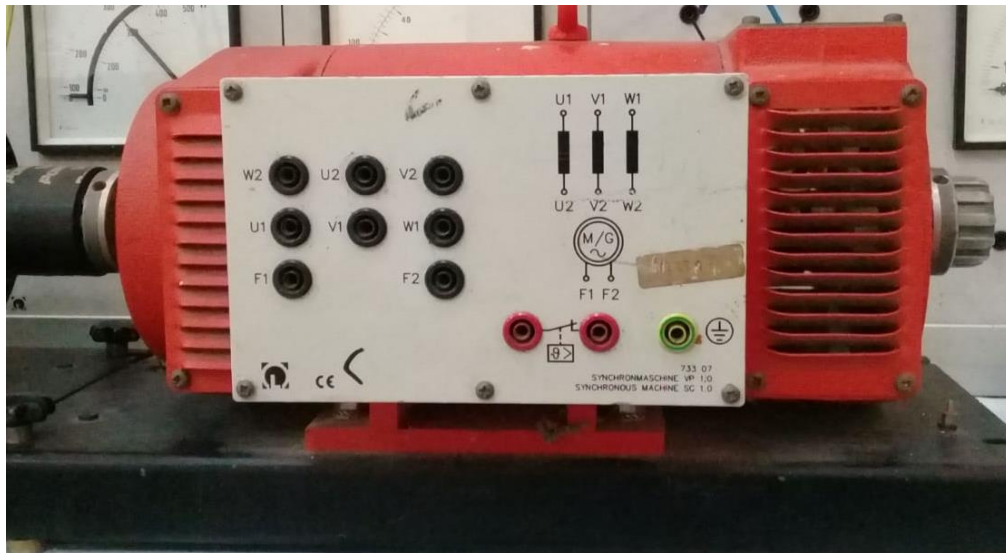
Keterangan :

C = Nilai Kapasitor

ΔV = *Ripple* Tegangan

2.3. Generator Sinkron

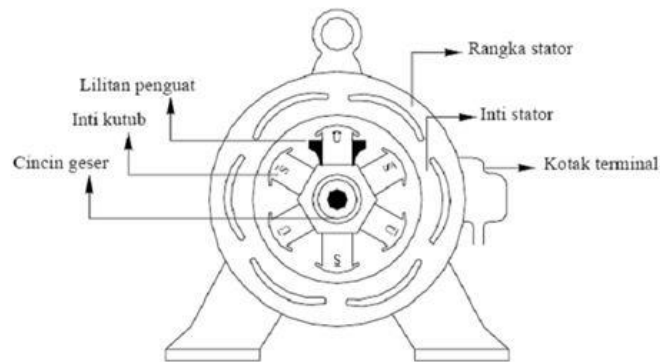
Generator sinkron adalah generator AC yang berfungsi mengubah energi gerak menjadi energi listrik dengan menggunakan induksi medan magnet, perubahan terjadi disebabkan adanya perubahan medan magnet pada kumparan jangkar (stator generator) [4].



Gambar 2.4 Generator Sinkron

2.3.1. Kontruksi Generator Sinkron

Kontruksi generator sinkron terdiri dari rotor (bagian yang bergerak) dan bagian stator (bagian yang diam), keduanya adalah rangkaian magnetik yang berbentuk silindris dan simetris pada generator sinkron juga terdapat celah ruang udara antara rotor dan stator yang berfungsi sebagai ruang terjadi fluksi atau induksi energi listrik dari rotor ke stator [5].



Gambar 2.5 Kontruksi Generator Sinkron

2.3.2. Pinsip Kerja Generator Sinkron

Kumparan medan rotor dihubungkan dengan pembangkit arus searah sebagai sumber eksitasi sehingga membangkitkan fluks yang memiliki besaran tetap terhadap waktu. Pada penggerak mula (*Prime Mover*) yang tergendeng dengan pengoprasian rotor sehingga rotor akan berputar pada kecepatan nominalnya sesuai dengan persamaan berikut:

$$n = \frac{120 \cdot f}{p} \quad (2.7)$$

Keterangan :

n = Kecepatan putar rotor (rpm)

p = Banyaknya kutub rotor

f = frekuensi (Hz)

Pengaturan kuat medan pada rotor dikontrol dengan mengatur arus medan rotor. Besar ggl induksi rata-rata yang dihasilkan kumparan jangkar generator sinkron ini dapat dilihat dalam persamaan berikut ini [4].

$$E = 4,44 \times \phi \cdot \varphi \cdot T \quad (2.8)$$

Jika

$$f = \frac{n.p}{120} \quad (2.9)$$

Maka,

$$E = 4,44 \times \frac{n.p}{120} \times \varphi \times T \quad (2.10)$$

Dengan,

$$C = \frac{44,4 pT}{120} \quad (2.11)$$

Maka,

$$E = Cn\varphi \quad (2.12)$$

Keterangan :

E = GGL induksi (Volt)

p = banyak kutub

n = Putaran (rpm)

φ = Fluks magnetik (weber)

C = Konstanta mesin

T = banyaknya lilitan /fase =1/2

Z = banyak sisi kumparan

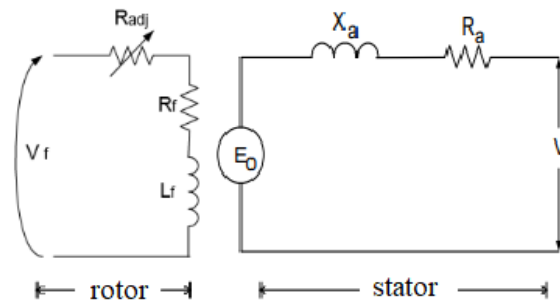
2.3.3. Karakteristik Generator Sinkron

2.3.3.1. Generaton Sinkron Tanpa Beban

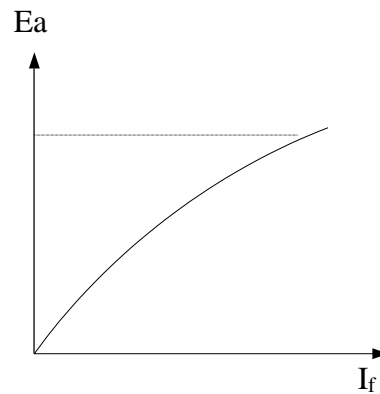
Pada keadaan generator sinkron diputar dalam kondisi kecepatan sinkron dan rotor dialiri arus medan (I_f), maka akan mendapatkan tegangan (E_0) yang terinduksi pada kumparan jangkar di stator. Hubungannya dapat diperlihatkan pada persamaan berikut ini :

$$E_0 = cn\phi \quad (2.13)$$

Pada keadaan tanpa beban arus jangkar tidak mengalir pada stator, sehingga tidak ada pengaruh reaksi jangkar. Fluks hanya dihasilkan oleh arus medan (I_f).



Gambar 2.6 Rangkaian ekuivalen generator sinkron tanpa beban



Gambar 2.7 Grafik hubungan arus penguat medan (I_f) dan E_0

Berikut adalah persamaan untuk besar GGL armatur tanpa beban dengan faktor daya (pf) beban = 1, faktor daya tertinggal dan faktor daya mendahului :

pf = 1

$$E_0 = \sqrt{(V_t + I_a R_a)^2 + (I_a X_s)^2} \quad (2.14)$$

pf = tertinggal

$$E_0 = \sqrt{(V_t \cos \theta + I_a + R_a)^2 + (V_t \sin \theta I_a X_s)^2} \quad (2.15)$$

pf = mendahului

$$E_0 = \sqrt{(V_t \cos \theta + I_a + R_a)^2 + (V_t \sin \theta - I_a X_s)^2} \quad (2.16)$$

Keterangan:

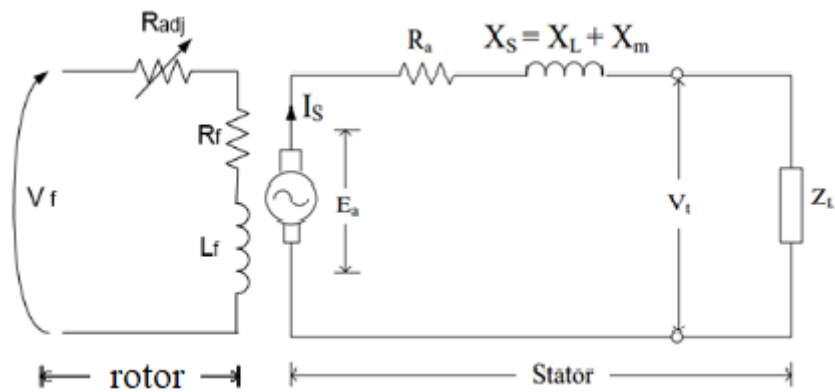
E_0 = GGL armatur tanpa beban

V_t = tegangan terminal output per fasa (Volt)

R_a = resistansi jangkar per fasa (ohm)

X_s = reaktansi sinkron per fasa (ohm).

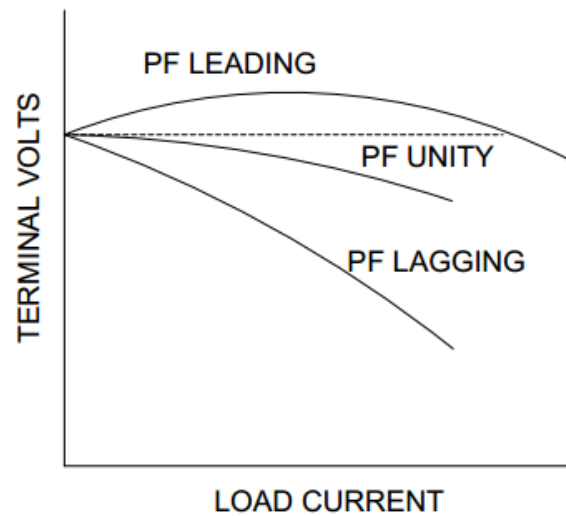
2.3.3.2. Generator sinkron berbeban



Gambar 2.8 Rangkaian ekuivalen generator sinkron berbeban

Pemberian beban yang bervariasi pada generator akan mengakibatkan besarnya tegangan terminal V_t akan berubah-ubah pula. Hal ini dikarenakan terdapat :

- Jatuh tegangan akibat reaktansi jangkar (R_a).
- Jatuh tegangan akibat reaktansi bocor jangkar (X_L).
- Jatuh tegangan akibat reaksi jangkar.



Gambar 2.9 Karakteristik generator AC Pada berbagai faktor daya

Berikut merupakan persamaan untuk mengetahui besar GGL armatur pada PF = 1, PF tertinggal dan PF mendahului :

- pf = 1

$$E_a = \sqrt{(V_t + I_a R_a)^2 + (I_a X_s)^2} \quad (2.17)$$

- pf = tertinggal

$$E_a = \sqrt{(V_t \cos \theta + I_a R_a)^2 + (V_t \sin \theta + I_a X_L)^2} \quad (2.18)$$

- pf = mendahului

$$E_a = \sqrt{(V_t \cos \theta + I_a R_a)^2 + (V_t \sin \theta - I_a X_L)^2} \quad (2.19)$$

Keterangan :

E_a = tegangan induksi pada jangkar per fasa (Volt)

R_a = resistansi jangkar per fasa (ohm)

X_L = reaktansi bocor per fasa (ohm)

2.4. Sistem Eksitasi

Sistem eksitasi ialah suatu sistem pemasokan arus DC sebagai upaya untuk meningkatkan intensitas *fluks* medan magnet yang dihasilkan kumparan medan agar

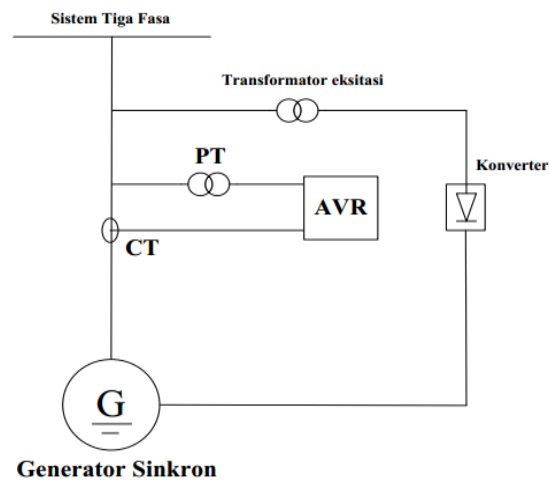
menjaga kesetabilan tegangan generator. Generator yang telah mencapai kecepatan putar nominalnya diberi arus DC di kumparan medannya agar nilai GGL induksi meningkat.

2.4.1. Jenis-Jenis Sistem Eksitasi

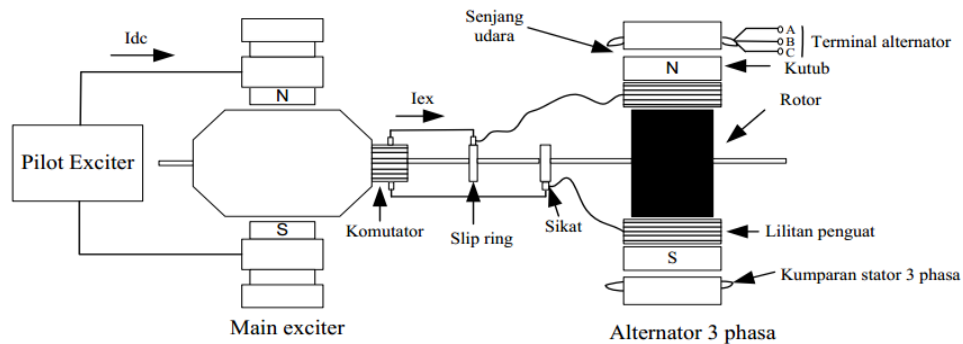
2.4.1.1. Sistem Eksitasi dengan Sikat

Sistem eksitasi ini memiliki sumber eksitasi yang berasal dari generator DC atau generator AC yang disearahkan menggunakan *rectifier* [6].

Sistem eksitasi statis merupakan sistem eksitasi yang peralatan eksitasinya tidak ikut bergerak bersama rotor generatornya. Nama lain dari sistem eksitasi statis adalah *self excitation* karena sumber daya eksitasi merupakan keluaran dari generator sendiri yang diubah ke DC.



Gambar 2.10 Sistem Eksitasi Statis

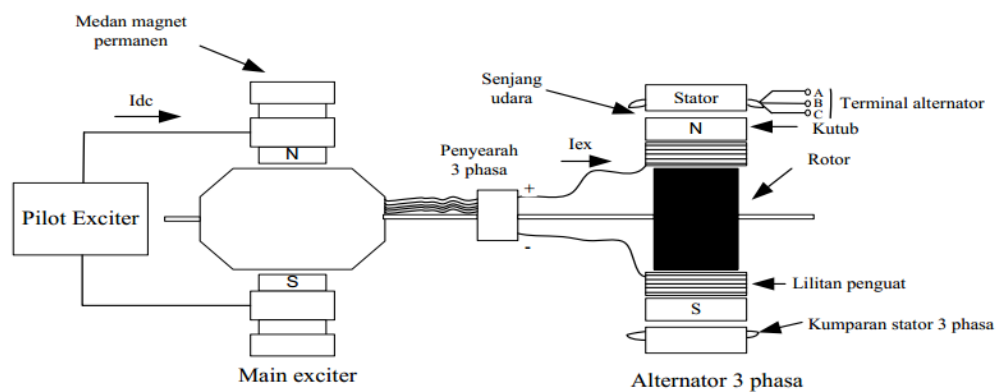


Gambar 2.11 Sistem Eksitasi Dinamis

Sistem eksitasi dinamis merupakan sistem eksitasi yang peralatan eksitasinya bergerak bersama putaran rotor generatornya.. *Pilot exciter* berupa generator AC dengan medan magnet permanen yang tegangan keluarannya disearahkan menggunakan diode penyearah dan disuplai ke *main exciter*.

2.4.1.2. Sistem Eksitasi Tanpa Sikat

Sistem eksitasi tidak menggunakan sikat dan *slip ring* untuk menyalurkan arus eksitasi ke generator utama. Sistem eksitasi ini menggunakan generator. Keluaran dari *main exciter* disearahkan oleh penyearah tiga fasa yang ikut berputar bersama rotor (*rotating rectifier*).



Gambar 2.12 Sistem Eksitasi Tanpa Menggunakan Sikat

2.5. AVR (*Automatic Voltage Regulator*)

2.5.1. Pengertian

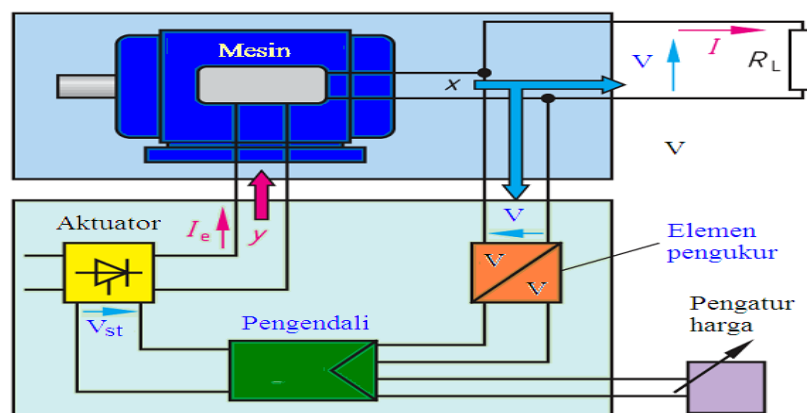
AVR (*Automatic Voltage Regulator*) adalah sebuah sistem kelistrikan yang berfungsi untuk menjaga agar tegangan keluaran generator tetap stabil meskipun nilai beban selalu bervariasi setiap saat [7].

AVR memiliki tiga bagian utama yaitu :

1. Komponen untuk memonitoring tegangan output dari generator
2. Komponen untuk menentukan tegangan *set poin*
3. Komponen untuk membandingkan tegangan output generator dan set poin sehingga menghasilkan *error signal*

2.5.2. Prinsip Kerja

Prinsip kerja dari AVR adalah mengendalikan arus penguatan DC yang diberikan ke kumparan medan generator. Tegangan generator akan tetap stabil karena saat tegangan generator berada dibawah tegangan seharusnya AVR menambah intensitas arus eksitasi dan begitu juga sebaliknya [8] .



Gambar 2.13 Sistem Kerja AVR (*Automatic Voltage Regulator*)

2.6. PID Controller

Kontroler PID (dari singkatan *Proportional-Integral-Derivative*) adalah salah jenis dari pengaturan yang banyak digunakan. PID merupakan jenis kontroler yang mekanismenya menggunakan umpan balik yang biasa dipakai pada sistem kontrol industri [8] :

2.6.1. Pengendali *Proportional* (P)

Pada pengendali *proportional*, memiliki hubungan antara sinyal keluaran $u(t)$ dengan sinyal kesalahan $e(t)$ sebagai berikut :

$$u(t) = K_p e(t) \quad (2.20)$$

Dengan bentuk dari transformasi laplace sebagai berikut :

$$U(s) = K_p e(s) \quad (2.21)$$

Fungsi alih dari kendali *proportional* :

$$\frac{U(s)}{E(s)} = K_p \quad (2.22)$$

Kendali *proportional* merupakan suatu penguatan yang nilainya dapat diubah-ubah, penambahan besar K_p akan menambah penguatan sistem sehingga digunakan untuk menambah kecepatan dari respon dan mengurangi e_{ss} (kesalahan keadaan mantap).

2.6.2. Pengendali *Integral* (I)

Pada pengendali *integral* keluaran sistem $u(t)$ diubah dengan nilai laju berbanding lurus dengan $e(t)$, berikut persamaan dari pengendali interal :

$$\frac{du(t)}{dt} = K_i e(t) \quad (2.23)$$

Sehingga

$$u(t) = \int_0^1 K_i e(t) dt \quad (2.24)$$

Adapun dalam bentuk transformasi laplace adalah sebagai berikut :

$$U(s) = K_i \frac{E(s)}{s} \quad (2.25)$$

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (2.26)$$

Pengendali jenis *Integral* (I) bertugas untuk menghilangkan kesalahan posisi dalam keadaan mantap. Pengendali ini biasa digunakan dengan jenis P + D, tetapi pada keadaan kecepatan respon dan ketidak-stabilan tidak menjadi masalah maka pengendali dengan P + I sudah cukup.

2.6.3. Pengendali *Derivative* (D)

Pengendali *darivative* atau diferensial besar dari keluaran pengendali $U(t)$ merupakan fungsi deiferensial dari sinyal kesalahan $e(t)$:

$$U(t) = K_D \frac{de(t)}{dt} \quad (2.27)$$

Adapun transformasi laplace dari pengendali D sebagai berikut :

$$U(s) = K_D s E(s) \quad (2.28)$$

Bentuk fungsi alihnya adalah sebagai berikut:

$$\frac{U(s)}{E(s)} = K_D s \quad (2.29)$$

Pengendali jenis ini digunakan untuk memperbaiki prestasi respon transien dari sistem kontrol.

2.6.4. Pengendali *Propotional Integral Derivatif* (PID)

Persamaan pengendali PID adalah sebagai berikut :

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{d(t)}{dt} \quad (2.30)$$

Dengan :

$$K_i = \frac{K_P}{T_i} \quad (2.31)$$

$$K_D = K_P \times T_D \quad (2.32)$$

Substitusi persamaan 2.31 dan 2.32 ke persamaan 2.230, maka akan menjadi persamaan PID berikut ini :

$$u(t) = K_p e(t) + \frac{K_P}{T_i} \int_0^t e(\tau) d\tau + K_P \times T_D \frac{d(t)}{dt} \quad (2.33)$$

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_D \frac{d(t)}{dt} \right) \quad (2.34)$$

Adapun bentuk transformasi laplace dari persamaan diatas adalah sebagai berikut :

$$U(s) = K_p E(s) + K_I \frac{E(s)}{s} + K_D s E(s) \quad (2.35)$$

Bentuk fungsi alihnya adalah sebagai berikut :

$$\frac{U(s)}{E(s)} = K_p + \frac{K_I}{s} + K_D s \quad (2.36)$$

$$\frac{U(s)}{E(s)} = \frac{K_D s^2 + K_p s + K_I}{s} \quad (2.37)$$

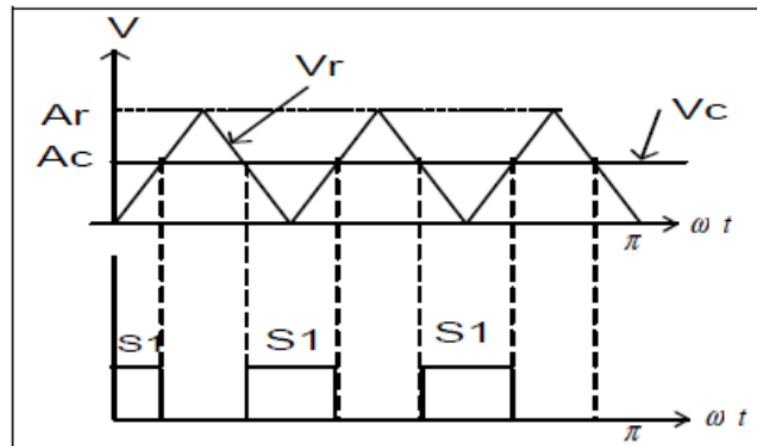
Dengan nilai K_p , K_I , dan K_D merupakan positif yang menandakan koefisien untuk term *proportional*, *integral*, dan *derivatif* secara berurutan (atau P, I dan D). Pada model ini :

- P bertanggung jawab pada nilai kesalahan saat ini, misalnya jika nilai kesalahan besar dan positif, maka keluaran kontrol juga besar dan positif
- I bertanggung jawab pada nilai kesalahan sebelumnya. Misalnya, keluaran saat ini kurang besar, maka kesalahan akan terakumulasi terus menerus, dan kontroler akan merespon dengan keluaran yang tinggi.
- D bertanggung jawab pada kemungkinan nilai kesalahan mendatang, berdasarkan *rate* perubahan pada setiap waktu.

Kontroler PID hanya mengandalkan variabel dari proses terukur, bukan pengetahuan mengenai prosesnya, maka secara luas digunakan. Dengan menyesuaikan dari *tuning* ketiga dari parameter model, kontroler PID dapat memenuhi kebutuhan proses [6].

2.7. Pulse Width Modulation (PWM)

PWM (*Pulse Width Modulation*) atau modulasi lebar pulsa merupakan teknik yang membandingkan sinyal referensi dengan sinyal *carrier* Gambar berikut merupakan hasil perbandingan sinyal referensi dan sinyal *carrier* yang mempunyai nilai *duty cycle* tertentu [10].



Gambar 2.14 *Pulse Width Modulation*

Duty cycle adalah perbandingan antara waktu konduksi dengan total waktu antara kondisi dan titik konduksi dikalikan dengan seratus persen.

$$\text{Duty Cycle} = \frac{t_{on}}{t_{on} + t_{OFF}} \times 100\% \quad (2.57)$$

Duty cycle nantinya akan digunakan untuk memberikan waktu konduksi dan tidak konduksinya komponen semikonduktor. Pada PWM, pulsa penyalaan yang mengendalikan posisi ON dan OFF saklar didapatkan dari perbandingan gelombang;).

2.8. Arduino Uno

Arduino Uno adalah papan pengembangan mikrokontroler yang berbasis chip *input* analog, menggunakan *crystal* 16 MHz antara lain pin A0 sampai A5, koneksi USB, jack listrik, *header* ICSP dan tombol reset [2]. Semua itu diperlukan untuk

mendukung sebuah rangkaian mikrokontroler. Adapun spesifikasi dari arduino uno R3 adalah sebagai berikut :

Tabel 2.1 Spesifikasi Arduino Uno R3

Mikrokontroler	Atmega328
Operasi Tegangan	5 A
Input Tegangan	7-12 V
Pin I/O Digital	14
Pin Analog	6
Arus DC tiap pin I/O	50 Ma
Arus DC ketika 3.3 V	50 mA
Memori Flash	32 KB
SRAM	2 KB



Gambar 2.15 Arduino Uno R3 (arduino.cc)

Arduino memiliki kemampuan untuk terintegrasi terhadap Labview dengan melakukan *upload source code* LIFA (*Labview Interface For Arduino*) Base pada

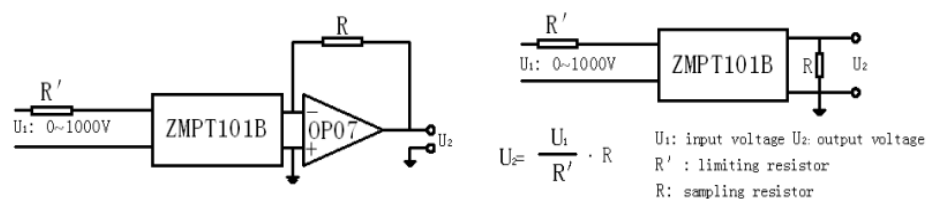
board mikrokontroler Arduino Uno. Alat ini juga dapat digunakan untuk membangkitkan sinyal PWM dengan keluran sinyal PWM 5 Volt DC. Agar dapat digunakan untuk sebagai masukan dari DC-DC konverter pada saklar semikonduktor dibutuhkan tegangan minimal 12-15 VDC. Sehingga perlu rangkain penguat sinyal PWM atau biasa disebut dengan *gate driver*.

2.9. Sensor Tegangan

Sensor tegangan digunakan untuk mengukur tegangan pada sistem, modul sensor yang digunakan yaitu jenis sensor ZMPT101B yang merupakan sensor tegangan yang digunakan sebagai pengganti transformator tegangan untuk arus AC 1 fasa.



(a)



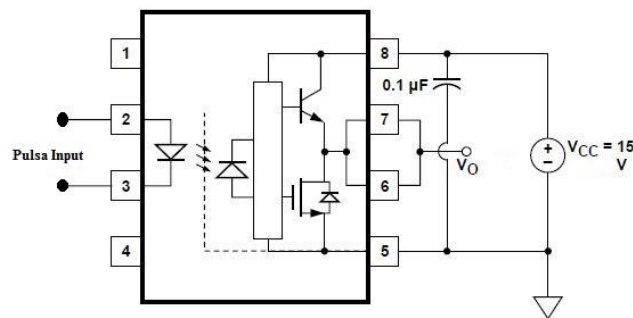
(b)

Gambar 2.16 Sensor tegangan ZMPT101B, (a) Tampilan fisik dan (b) Rangkaian penggunaan sensor

Sensor tegangan ZMPT101B memiliki spesifikasi dengan masukan tegangan sebesar 5 VDC, dan arus sebesar 2 mA, serta range optimal penggunaan 110-250 VAC.

2.10. Gate Driver

Rangkaian *gate driver* ini adalah rangkaian bertugas memindahkan dan memperkuat pulsa kontrol dari mikrokontroler untuk menjalankan *switch* pada mosfet. *Gate driver* ini menggunakan IC HCPL3120 yang memiliki 8 kaki atau pin. Berikut menunjukkan rancangan rangkaian *gate driver* yang digunakan [11].

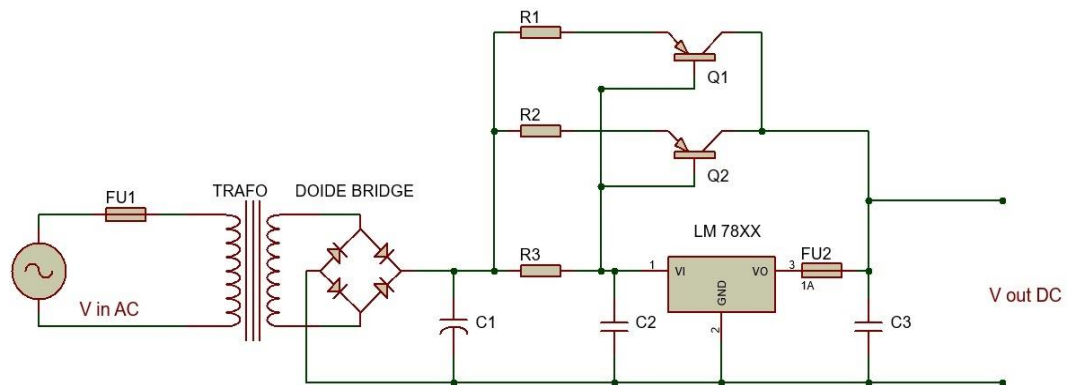


Gambar 2.17 Rangkaian *gate driver* berbasis HCPL3120

Terlihat dari Gambar 2.14 di atas bahwa sisi masukan HCPL3120 yaitu pin 2 dan 3, dihubungkan dengan sisi keluaran dari mikrokontroler penghasil pulsa kontrol. Pada sisi keluaran, pin 8 dihubungkan ke catu tegangan 15 V, pin 5 dihubungkan ke *ground* dari catu tegangan tersebut. Pin 6 dan 7 dengan *ground* dihubungkan ke sisi *gate* MOSFET.

2.11. Power Supply

Rangkaian *power supply* atau catu daya digunakan sebagai suplai tegangan dan arus searah untuk rangkaian lainnya, rangkaian *power supply* terdiri dari trafo *step-down* untuk mendapatkan tegangan yang diinginkan dan jembatan dioda sebagai penyearah gelombang.



Gambar 2.18 Rangkaian *power supply*

BAB 3. METODE PENELITIAN

3.1. Waktu dan Tempat Penelitian

Penelitian ini dilaksanakan pada bulan Agustus 2018 – Juli 2019 di Laboratorium Terpadu Teknik Elektro, Jurusan Teknik Elektro, Universitas Lampung.

3.2. Alat dan Bahan

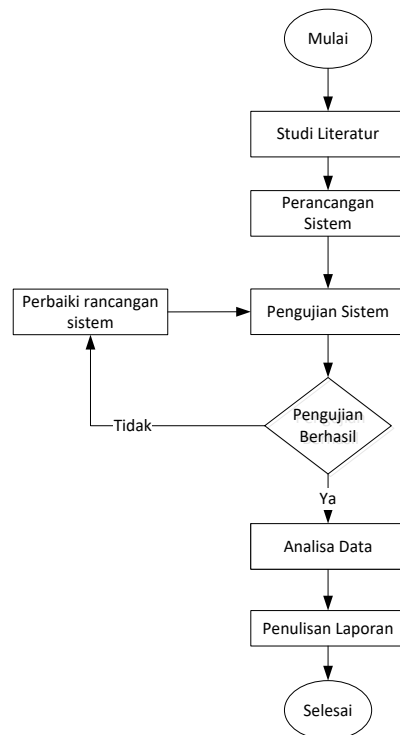
Alat dan bahan yang digunakan dalam penelitian ini adalah sebagai berikut,

1. Laptop ASUS Tipe X435M dengan spesifikasi Intel (R) Celeron (R) CPU N2840 @2.216GHz 2.16Gz, RAM 4,00 GB (3,88 GB Usable), dan sistem operasi Windows 10 Pro 64 bit sebagai media perancangan program dan antarmuka.
2. Rangkaian *buck converter* yang terbangun dari kapasitor, induktor, dioda, MOSFET, *headshink* dan *fan*.
3. Perangkat lunak Arduino 1.6.12 sebagai program untuk *source code* LIFA.
4. Arduino uno sebagai mikrokontroler.
5. Transformator *step down*.
6. *Gate driver* sebagai penguat sinyal PWM dari mikrokontroler..
7. Sensor tegangan ZMPT101B sebagai pembaca nilai tegangan keluaran pada beban.
8. Motor induksi 3 fasa untuk penggerak generator

9. Watt meter untuk mengukur nilai beban.
10. Generator sinkron sebagai mesin yang membangkit energi listrik
11. Multimeter sebagai alat ukur tegangan dan arus eksitasi
12. Kabel jumper untuk menghubungkan *output* alat ke *input* alat yang lain.

3.3. Tahapan Penelitian

Adapun langkah kerja yang dilakukan untuk menyelesaikan penelitian tugas akhir ini digambarkan dalam diagram alir pada gambar 3.1. Hal ini dilakukan untuk memudahkan dalam proses penelitian sehingga dapat dikerjakan secara sistematis dan terperinci, adapun diagram alir pada penelitian ini adalah sebagai berikut :



Gambar 3.1 Diagram alir penelitian

3.3.1. Studi Literatur

Pada studi literatur ini dilakukan pencarian referensi dan informasi yang berkaitan dengan skripsi ini. Diantaranya melalui buku, jurnal ilmiah, internet maupun skripsi-skripsi yang berkaitan. Informasi tersebut antara lain :

- a. Arduino Uno R3.
- b. Dc-dc *converter*.
- c. Generator sinkron dan sistem eksitasinya.
- d. Pengendali PID.
- e. Pembangkit listrik tenaga biomassa.
- f. Sensor tegangan.

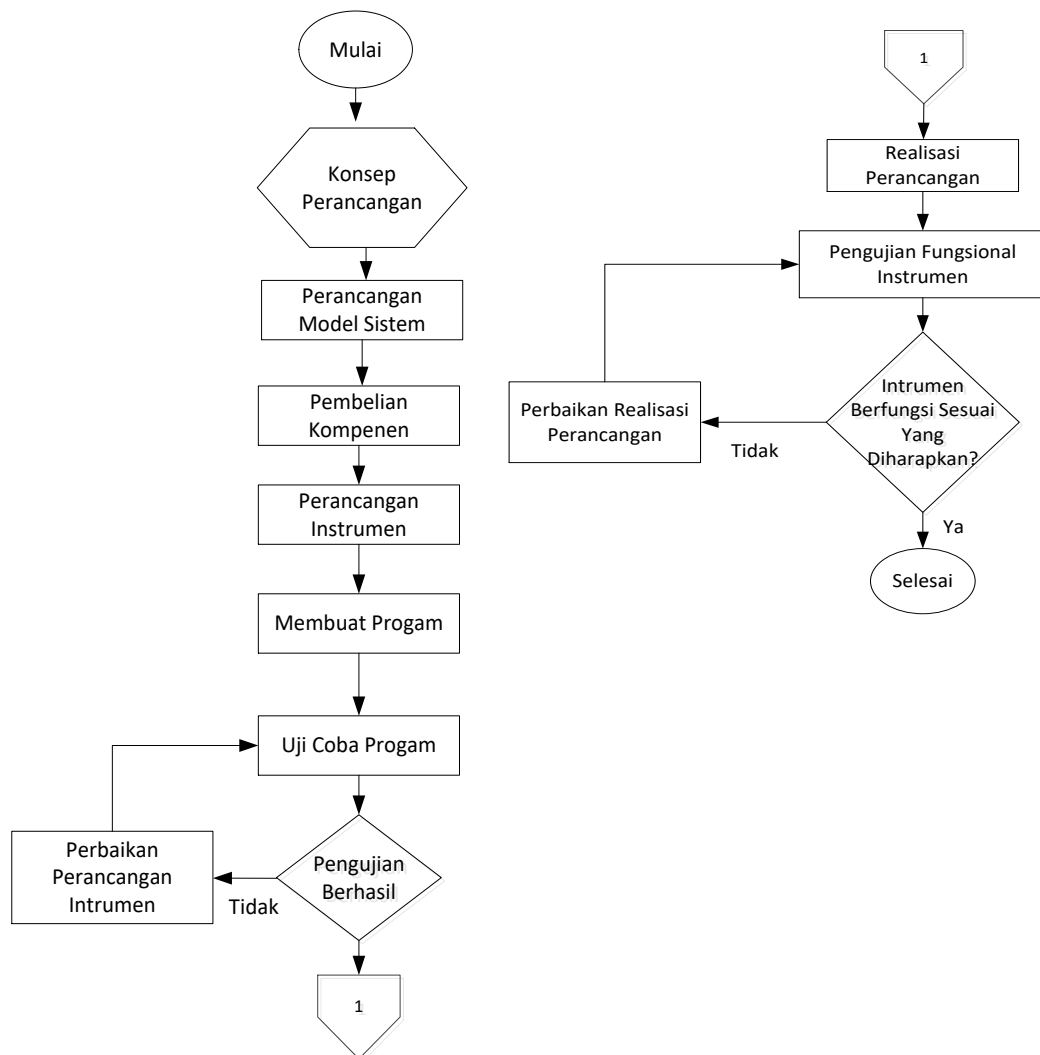
3.3.2. Konsep Perancangan Sistem

Pada tahapan ini dilakukan perencanaan atau konsep mengenai perancangan sistem yang akan dibuat dengan melakukan beberapa hal sebagai berikut :

1. Pemilihan komponen untuk perancangan kontrol medan eksitasi dengan arduino uno pada *emulator* pembangkit listrik tenaga biomassa berupa mikrokontroler Arduino Uno R3, *DC-DC converter*, *gate driver*, *power supply*, dan sensor yang sesuai dengan kebutuhan sistem.
2. Penentuan parameter-parameter dan karakteristik kontrol medan eksitasi yang akan digunakan pada *emulator* pembangkit listrik tenaga biomassa.
3. Referensi untuk *embedded system* untuk pengendalian dan pembacaan sensor pada sistem.

3.3.3. Perancangan Sistem

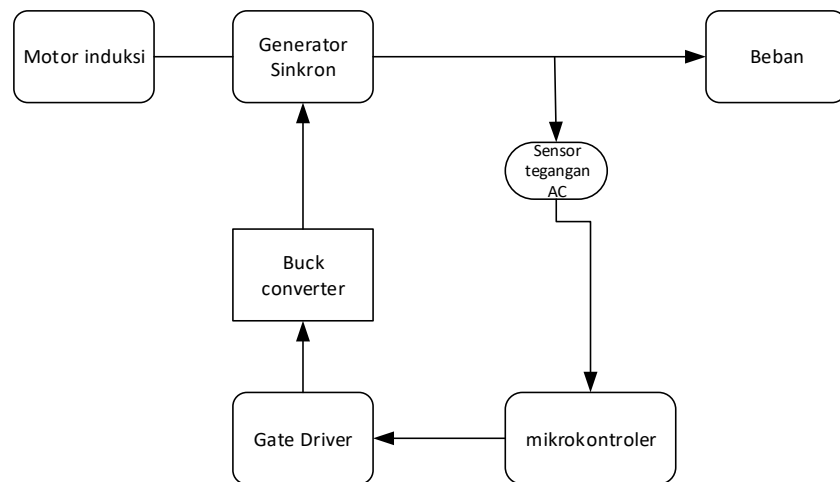
Pada tahap ini penulis melakukan perancangan sistem yang akan digunakan dalam pembuatan alat. Tahapan ini akan dilakukan untuk membuat alat kontrol medan eksitasi *emulator* pembangkit listrik tenaga biomassa ditunjukkan dengan diagram alir perancangan sistem pada Gambar 3.2 hal ini dilakukan untuk memudahkan dalam perancangan sistem sehingga dapat diselesaikan secara sistematis.



Gambar 3.2 Diagram alir perancangan sistem

3.3.4. Perencanaan Model Sistem

Perencanaan model sistem dilakukan dengan pembuatan perangkat sistem yang terdiri dari mikrokontroler arduino uno, *buck converter*, *gate driver*, *power supply*, generator sinkron, motor induksi 3 fasa dan sensor tegangan.



Gambar 3.3 Blok diagram perencanaan alat

3.3.5. Perancangan Kerja Sistem

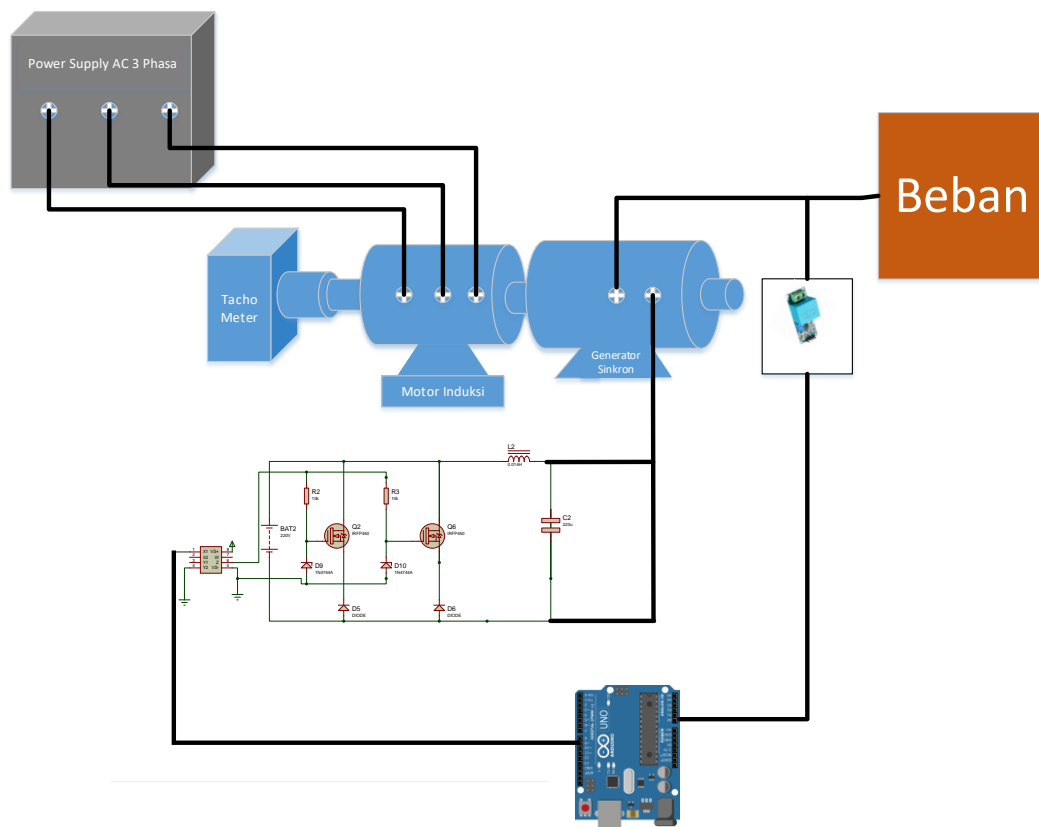
Perancangan kerja dari sistem kontrol medan eksitasi pada *emulator* pembangkit listrik tenaga biomassa secara garis besar yaitu mencari karakteristik medan eksitasi dan menyetabilkan tegangan keluaran generator ketika terjadi perubahan nilai beban listrik. Sehingga pada penelitian ini mengatur medan eksitasi dengan mengatur arus DC dengan menggunakan mikrokontroler arduino uno, pembacaan tegangan dari keluaran generator sinkron dengan sensor tegangan ZMPT101B kemudian akan menjadi masukan ke arduino uno sebagai variabel kontrol sehingga keluaran tegangan generator sama atau mendekati nilai tegangan *setpoint* yaitu 220 V terhadap perubahan nilai beban. Tahapan perencanaanya sebagai berikut :

1. *Power supply* AC 3 fasa sebagai masukan tegangan yang akan memutar motor induksi 3 fasa.
2. Motor induksi yang mendapatkan tegangan dari *power supply* AC 3 fasa akan memutar generator sinkron.
3. Generator sinkron yang berputar belum dapat menghasilkan tegangan sebelum penambahan pasokan listrik DC pada kumparan medan rotor sebagai penguat medan magnet rotor.
4. Eksitasi yang terdiri dari rangkain *buck conveter, gate driver*, catu daya dc, arduino uno dan sensor tegangan. Eksitasi ini akan menghasilkan tegangan DC dan arus DC untuk membangkitkan medan magnet pada generator. Sensor tegangan pada sistem eksitasi ini akan terus membaca keluaran tegangan pada generator sinkron, kemudian hasil dari pembacaan tegangan pada sensor tegangan ini masuk ke arduino, pada pemograman arduino yang sudah ditanamkan pengendalian PID akan merespon dengan mengeluarkan nilai PWM memperbaiki eror dari tegangan *setpoint* dengan tegangan terbaca oleh sensor tegangan yang kemudian keluaran PWM ini akan berbentuk sinyal digital yang akan masuk ke rangkaian gate driver, hal ini dikarenakan nilai tegangan PWM hanya 5 V sehingga perlu dinaikan sampai dengan 15 V oleh gate driver untuk dapat masuk ke kaki gain Mosfet pada rangkaian *buck converter* untuk proses *switching*, proses ini akan terus kontinyu hingga tegangan yang terbaca pada sensor tegangan AC ini mendekati atau sama dengan nilai tegangan *setpoint* yaitu 220 V.
5. Pada penambahan beban maka generator akan merespon dengan keluraran tegangan yang terbaca menurun, dengan progam yang sudah terpasang pada

sistem eksitasi maka proses *switching* hingga tegangan yang terbaca pada sensor tegangan AC mendekati atau sama dengan 220 V.

6. Pada *datalogger* yang terpasang pada arduino uno ini akan menyimpan data tegangan dari generator ke beban yang telah ditanamkan pada pemograman mikrokontroler.

Sehingga nantinya rancangan *hardware* dari emulator pembangkit listrik tenaga biomassa akan seperti Gambar 3.4 dibawah ini :



Gambar 3.4 Rancangan *hardware* dari kontrol medan eksitasi pada *emulator* pembangkit listrik tenaga biomassa

3.3.6. Analisis Data dan Kesimpulan

Setelah pembuatan alat selesai, langkah selanjutnya yaitu pengambilan data dan melakukan analisis data yang didapatkan dari pengujian perangkat sistem dan hasil dari pengaruh pembebanan. Proses analisa yang dilakukan agar mengetahui kemampuan kerja sistem sehingga didapatkan kesimpulan.

3.3.7. Penulisan Laporan

Dalam tahap ini dilakukan penulisan laporan dari data yang diperoleh dan telah didapatkan analisa data serta kesimpulan. Berdasarkan data yang telah dianalisa dan disimpulkan maka didapatkan saran yang berguna untuk penelitian selanjutnya.

BAB 5. PENUTUP

5.1. Kesimpulan

Adapun kesimpulan dari hasil dan pembahasan *emulator* pembangkit listrik tenaga biomassa uap adalah sebagai berikut :

1. Emulator ini dirancang untuk meniru eksitasi pembangkit listrik biomassa.
2. Medan eksitasi maksimal alat ini untuk mencapai tegangan terminal 220 V yaitu pada beban 330 Watt dan arus beban sebesar 1,45 A.
3. Pada daya beban yang sama penambahan beban induktif mengalami *drop* tegangan yang lebih besar dibandingkan dengan penambahan beban resistif hal ini disebabkan faktor daya beban induktif lebih kecil dibandingkan dengan faktor daya beban resistif.
4. Perbandingan antara referensi dari respon eksitasi pada pembangkit listrik tenaga biomassa memiliki waktu yang sama untuk mencapai kondisi *steady state* dengan respon eksitasi dari *emulator* pembangkit listrik tenaga biomassa yaitu pada detik ke-4 pada awal starting dari eksitasi generator.
5. Pada kontrol medan eksitasi *emulator* pembangkit listrik pembangkit biomassa dalam mencapai *steady state* tegangan terminal apabila terjadi perubahan tegangan yaitu dengan mengatur keluaran nilai PWM secara otomatis dengan pemodelan PID *controller*.

5.2. Saran

1. Penambahan pengontrolan pada sisi motor induksi sehingga penelitian ini dapat dikembangkan untuk mendapatkan model emulator pembangkit listrik biomassa uap yang lebih baik lagi
2. Diperlukan penambahan eksitasi pada generator untuk mendapatkan nilai keluaran daya yang lebih maksimal
3. Diperlukan penambahan sensor tegangan DC untuk mengukur tegangan eksitasi dan upayakan tidak mengurangi suplai daya ke beban dalam penelitian ini tidak menggunakan sensor tegangan DC dikarenakan tidak ada sensor tegangan DC yang dapat mengukur tegangan lebih dari 25 volt.

DAFTAR PUSTAKA

- [1] R. I. Sudjoko and P. A. Darwito, "Design and simulation of synchronous generator excitation system using buck converter at motor generator trainer model LEM-MGS," *Proceeding - ICAMIMIA 2017 Int. Conf. Adv. Mechatronics, Intell. Manuf. Ind. Autom.*, pp. 41–45, 2018.
- [2] L. W. Cahyadi, T. Andromeda, and M. Facta, "Kinerja Konverter Arus Searah Tipe Buck Converter Dengan Umpan Balik Tegangan Berbasis TL494," *Transient*, vol. 6, p. 7, 2017.
- [3] H. Buntulayuk and F. A. Samman, "Rancangan DC-DC Converter untuk Penguatan Tegangan," vol. 21, no. 02, pp. 78–82, 2017.
- [4] W. Sunarlik, "Prinsip Kerja Generator," *Prinsip Kerja Gener. Sink.*, p. 6, 2017.
- [5] S. Armansyah, "Pengaruh Penguatan Medan Generator Sinkron Terhadap Tegangan Terminal," *J. Tek. Elektro UISU*, vol. 1, no. 3, pp. 48–55, 2016.
- [6] I. S. Amien, "Sinkron Yang Bekerja Paralel Terhadap," vol. 7, no. 1, pp. 8–15, 2014.
- [7] E. Nw, "Perencanaan Optimal Sistem Kontrol AVR (Automatic Voltage Regulator) Untuk Memperbaiki Kestabilan Tegangan Dengan Menggunakan Algoritma Genetik," pp. 1–11.
- [8] A. Nurdin, A. Azis, and R. A. Rozal, "Peranan Automatic Voltage Regulator

- Sebagai Pengendali Tegangan Generator,” vol. 3, no. 1, pp. 163–173, 2018.
- [9] R. L. Kaufmann, C. F. Matson, A. H. Rowberg, and W. R. Beisel, “Defective lipid disposal mechanisms during bacterial infection in rhesus monkeys,” *Metabolism*, vol. 25, no. 6, pp. 615–624, 1976.
- [10] A. Mz, “Pulse Width Modulation (PWM),” *Inst. Pertan. Bogor*, pp. 1–4, 2015.
- [11] . F. and M. I. Hamid, “Desain Rangkaian Gate-Driver untuk Konverter yang Bekerja dengan Voltage Mode Control,” *J. Nas. Tek. Elektro*, vol. 5, no. 2, p. 175, 2016.

LAMPIRAN

Lampiran A. List Program

1. Program Eksitasi Generator

```

#include <Filters.h> //Easy library to do the calculations
#include <SPI.h> //Libraries for the OLED display
#include <SD.h>
#include <Wire.h>
#include "RTCLib.h"
#include <PWM.h>
#include <PID_v1.h>

#define LOG_INTERVAL 100
#define SYNC_INTERVAL 100 // mills between calls to flush() - to write data to
the card
uint32_t syncTime = 0; // time of last sync()
unsigned int sample = 0 ;
#define ECHO_TO_SERIAL 1 // echo data to serial port
#define WAIT_TO_START 0 // Wait for serial input in setup()
float testFrequency = 50; // test signal frequency (Hz)
float windowLength = 40.0/testFrequency; // how long to average the signal, for
statistist

int Sensor = 0; //Sensor analog input, here it's A0

float intercept = -0.02; // to be adjusted based on calibration testing
float slope = 0.0622; // to be adjusted based on calibration testing
float current_Volts; // Voltage

unsigned long printPeriod = 1000; //Refresh rate
unsigned long previousMillis = 0;
unsigned long sebelumce = 0;

// PWM dan PID
// PWM
int pin = 3;
uint32_t frequency = 10000; //frequency (in Hz)
float D = 255;
// PID
double Setpoint; // nilai konstan
double Input; // voltage sensor
double Output; // PWM

// PID Parameter
float kp = 0.009556 , ki = 1.87892 , kd = 0;

```



```

// create PID instant
PID myPID(&Input, &Output, &Setpoint, kp, ki, kd, DIRECT);
//variabel sensor arus dc
unsigned long now = 0;
unsigned long lastMessage = 0;
int mVperAmp = 66; // use 100 for 20A Module and 66 for 30A Module
int RawValue= 0;
int ACSoffset = 2500;
double Voltage = 0;
double Idc = 0;
int offset =20;
int32_t frequency1 = 40000;
// hitung manual
int bn = 0;
float n;

// variabel variabel-----
-----
// for the data logging shield, we use digital pin 10 for the SD cs line
const int chipSelect = 10;
//variabel sensor arus ac
double VRMS = 0;
double Iac = 0;
// the logging file
File logfile;
void setup()
{
  Serial.begin(250000 ); // start the serial port
  rtc.begin(); // Start the RTC library code
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

// PWM
//initialize all timers except for 0, to save time keeping functions
  InitTimersSafe();
  bool success = SetPinFrequencySafe(pin, frequency);
  if (success)
  {
    pinMode(3, OUTPUT);
  }
// PID
// Code untuk tegangan
  Setpoint = 220;
  // Turn the PID On
  myPID.SetMode (AUTOMATIC);
  // Adjust PID Value
  myPID.SetTunings (kp,ki,kd);

// DATA LOGGER

```

```

Serial.println();

#if WAIT_TO_START
  Serial.println("Type any character to start");
  while (!Serial.available());
#endif //WAIT_TO_START

// initialize the SD card
Serial.print("Initializing SD card...");
// make sure that the default chip select pin is set to
// output, even if you don't use it:
pinMode(10, OUTPUT);

// see if the card is present and can be initialized:
if (!SD.begin(chipSelect)) {
  error("Card failed, or not present");
}
Serial.println("card initialized.");

// create a new file
Createfile();

logfile.println("Time,Tegangan AC,PWM Output,Arus AC,Arus DC");
#if ECHO_TO_SERIAL
Serial.println("Time,Tegagan AC,PWM Output, Arus AC,Arus DC");
#endif //ECHO_TO_SERIAL

}

void loop()
{
  RunningStatistics inputStats;          //Easy life lines, actual calculation of the
RMS requires a load of coding
  inputStats.setWindowSecs( windowLength );
  DateTime now = rtc.now();

  //variabel sensor arus AC
  float result;
  int readValue;
  int maxValue = 0;
  int minValue = 1024;

  while( true ) {
    //sensor tegangan AC
    Sensor = analogRead(A0); // read the analog in value:
    inputStats.input(Sensor); // log to Stats function
    sample = sample + 1;
  }
}

```

```

    current_Volts = intercept + slope * inputStats.sigma(); //Calibartions for offset
and amplitude
    current_Volts = current_Volts*(40.555);           //Further calibrations for the
amplitude
    float Vac = current_Volts;
    Input = current_Volts;
    // PID calculate
    myPID.Compute ();
    pwmWrite (3,Output);
    Serial.print ("n");
    Serial.print (",");
    Serial.print(Vac);
    Serial.print("\n");

//penentuan range untuk sensor arus AC
readValue = analogRead(A1);
if (readValue > maxValue)
    {
        maxValue = readValue;
    }
if (readValue < minValue)
    {
        minValue = readValue;
    }

result = ((maxValue - minValue) * 4.74)/1024.0;

if(((unsigned long)(millis() - previousMillis) >= printPeriod) {

    previousMillis = millis(); // update time every second

    Serial.print( "\n" );

    current_Volts = intercept + slope * inputStats.sigma(); //Calibartions for offset
and amplitude
    current_Volts = current_Volts*(40.555);           //Further calibrations for the
amplitude
    float Vac = current_Volts;

    // read the value from the voltage sensor, analog input : 0 to 1024. We map is to
a value from 0 to 225

    // OUTPUT PWM 3
    int pwmpoint = Output ;
    // PWM pin 3

    Serial.print("sample = ");
    Serial.println(sample);

```

```

bn = bn + 1;

//arus AC
Voltage = result;
VRMS = (Voltage/2.0)*0.5;
Iac = ((VRMS*1000)/mVperAmp)-0.1026;

//RESET
if((unsigned long)(millis() - sebelumce) >= 5000) {

maxValue = 0 ;
minValue = 1024;
sebelumce = millis(); // update time every second
}
//Perintah mecetak
DateTime now = rtc.now();
logfile.print(now.hour(), DEC);
logfile.print(':');
logfile.print(now.minute(), DEC);
logfile.print(':');
logfile.print(now.second(), DEC);
logfile.print(", ");
logfile.print(Vac);
logfile.print(", ");
logfile.print(pwmpoint);
logfile.print(", ");
logfile.println(Iac);

#if ECHO_TO_SERIAL

Serial.println(Vac);
Serial.print(now.hour(), DEC);
Serial.print(':');
Serial.print(now.minute(), DEC);
Serial.print(':');
Serial.print(now.second(), DEC);
Serial.print(", ");
Serial.print(Vac);
Serial.print(", ");
Serial.print(Output);
Serial.print(", ");
Serial.print(Iac);
Serial.print(", ");
Serial.print("bn");
Serial.print (bn);

#endif

```

```

        logfile.flush();
    }
}
}
// Fungsi Membuat File
void Createfile()
{
    char filename[] = "LOGGER00.CSV";
    for (uint8_t i = 0; i < 100; i++) {
        filename[6] = i/10 + '0';
        filename[7] = i%10 + '0';
        if (! SD.exists(filename)) {
            // only open a new file if it doesn't exist
            logfile = SD.open(filename, FILE_WRITE);
            break; // leave the loop!
        }
    }
    if (! logfile) {
        error("couldnt create file");
    }
    Serial.print("Logging to: ");
    Serial.println(filename);
}
// Fungsi error
void error(char *str)
{
    Serial.print("error: ");
    Serial.println(str);

    // red LED indicates error

    while(1);
}

```

7. Progam Skenario Pembebeban Lampu

```

void setup() {
    // put your setup code here, to run once:

    /*-----(RELAY )-----*/
    #define RELAY_ON 0

```

```

#define RELAY_OFF 1
#define Relay_1 4 // Arduino Digital I/O pin number
#define Relay_2 5
#define Relay_3 6
#define Relay_4 7
//-----( Initialize Pins so relays are inactive at reset)----
digitalWrite(Relay_1, RELAY_OFF);
digitalWrite(Relay_2, RELAY_OFF);
digitalWrite(Relay_3, RELAY_OFF);
digitalWrite(Relay_4, RELAY_OFF);

// initialize serial:
Serial.begin(9600);
while(! Serial); // wait until Serial is ready

//---( THEN set pins as outputs )----
pinMode(Relay_1, OUTPUT);
pinMode(Relay_2, OUTPUT);
pinMode(Relay_3, OUTPUT);
pinMode(Relay_4, OUTPUT);

delay(3000); //Check that all relays are inactive at Reset
//skenario 1
Serial.println("Skenario 1");
digitalWrite(Relay_1, RELAY_OFF);
digitalWrite(Relay_2, RELAY_OFF);
digitalWrite(Relay_3, RELAY_OFF);
digitalWrite(Relay_4, RELAY_OFF);
delay (50000);

    digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_ON);
digitalWrite(Relay_3, RELAY_ON);
digitalWrite(Relay_4, RELAY_ON);
delay(50000);

//Skenario 2
Serial.println ("Skenario 2");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_ON);
digitalWrite(Relay_3, RELAY_ON);
digitalWrite(Relay_4, RELAY_OFF);

    delay(50000);

```

```
//Skenario 3
Serial.println("Skenario 3");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_ON);
digitalWrite(Relay_3, RELAY_OFF);
digitalWrite(Relay_4, RELAY_ON);

delay(50000);

//Skenario 4
Serial.println("Skenario 4");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_OFF);
digitalWrite(Relay_3, RELAY_ON);
digitalWrite(Relay_4, RELAY_ON);
delay(50000);

//Skenario 5
Serial.println("Skenario 5");
digitalWrite(Relay_1, RELAY_OFF);
digitalWrite(Relay_2, RELAY_ON);
digitalWrite(Relay_3, RELAY_ON);
digitalWrite(Relay_4, RELAY_ON);

delay(50000);

//Skenario 6
Serial.println("Skenario 6");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_ON);
digitalWrite(Relay_3, RELAY_OFF);
digitalWrite(Relay_4, RELAY_OFF);

delay(50000);

//Skenario 7
Serial.println("Skenario 7");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_OFF);
digitalWrite(Relay_3, RELAY_OFF);
digitalWrite(Relay_4, RELAY_ON);

delay(50000);
```

```
//Skenario 8
Serial.println("Skenario 8");
  digitalWrite(Relay_1, RELAY_OFF);
  digitalWrite(Relay_2, RELAY_OFF);
  digitalWrite(Relay_3, RELAY_ON);
  digitalWrite(Relay_4, RELAY_ON);

  delay(50000);
```

```
// Skenario 9
Serial.println("Skenario 9");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_OFF);
digitalWrite(Relay_3, RELAY_ON);
digitalWrite(Relay_4, RELAY_OFF);
delay(50000);
```

```
//Skenario 10
Serial.println("Skenario 10");
digitalWrite(Relay_1, RELAY_OFF);
digitalWrite(Relay_2, RELAY_ON);
digitalWrite(Relay_3, RELAY_ON);
digitalWrite(Relay_4, RELAY_OFF);

  delay(50000);
```

```
//Skenario 11
Serial.println("Skenario 11");
  digitalWrite(Relay_1, RELAY_OFF);
  digitalWrite(Relay_2, RELAY_ON);
  digitalWrite(Relay_3, RELAY_OFF);
  digitalWrite(Relay_4, RELAY_ON);

  delay(50000);
```

```
//Skenario 12
Serial.println("Skenario 12");
digitalWrite(Relay_1, RELAY_ON);
digitalWrite(Relay_2, RELAY_OFF);
```



```
digitalWrite(Relay_3, RELAY_OFF);  
digitalWrite(Relay_4, RELAY_OFF);  
delay(50000);
```

```
//Skenario 13  
Serial.println("Skenario 13");  
digitalWrite(Relay_1, RELAY_OFF);  
digitalWrite(Relay_2, RELAY_ON);  
digitalWrite(Relay_3, RELAY_OFF);  
digitalWrite(Relay_4, RELAY_OFF);  
  
delay(50000);
```

```
//Skenario 14  
Serial.println("Skenario 14");  
digitalWrite(Relay_1, RELAY_OFF);  
digitalWrite(Relay_2, RELAY_OFF);  
digitalWrite(Relay_3, RELAY_ON);  
digitalWrite(Relay_4, RELAY_OFF);  
  
delay(50000);
```

```
//Skenario 15  
Serial.println("Skenario 50");  
digitalWrite(Relay_1, RELAY_OFF);  
digitalWrite(Relay_2, RELAY_OFF);  
digitalWrite(Relay_3, RELAY_OFF);  
digitalWrite(Relay_4, RELAY_ON);  
  
delay(50000);
```

```
//Skenario 16  
Serial.println("Skenario 16");  
digitalWrite(Relay_1, RELAY_OFF);  
digitalWrite(Relay_2, RELAY_OFF);  
digitalWrite(Relay_3, RELAY_OFF);  
digitalWrite(Relay_4, RELAY_OFF);
```

```
Serial.println("Selesai");  
}
```