



**MONITORING DAN PENGONTROLAN
PEMBEBANAN PADA PEMBEBANAN GENERATOR
INDUKSI SATU FASA BERBASIS RASPBERRY PI**

SKRIPSI

Oleh

Marsandi Arfianto

NIM 101910201071

PROGRAM STUDI STRATA 1 TEKNIK

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS JEMBER

2015



**MONITORING DAN PENGONTROLAN
PEMBEBANAN PADA GENERATOR INDUKSI SATU
FASA BERBASIS RASPBERRY PI**

SKRIPSI

Diajukan guna melengkapi tugas akhir dan memenuhi syarat – syarat
untuk menyelesaikan Program Studi Teknik Elektro (S1)
dan mencapai gelar Sarjana Teknik

Oleh:

Marsandi Arfianto

NIM 101910201071

**PROGRAM STUDI STRATA 1 TEKNIK
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER**

2015

PERSEMBAHAN

Alhamdulillah, segala puji bagi Allah SWT, yang telah memberikan limpahan nikmat yang sangat luar biasa kepada penulis, dan tidak lupa juga sholawat dan salam selalu tercurahkan kepada Nabi Muhammad SAW yang telah membawa kita semua menuju peradaban manusia yang lebih baik. Dengan kerendahan hati, penulis mempersembahkan tugas akhir ini untuk:

Ibunda tercinta Libatin, Ayahanda Halil dan seluruh keluarga yang selalu mendoakan dan selalu mendukung baik secara moral dan materi.

Tri Suci Nurwinda Arumingtiyas yang selama ini telah menemani dengan sabar dan tabah

Almamater Jurusan Teknik Elektro Universitas Jember.

MOTTO

“Hanya Engkaulah (Allah) yang kami sembah, dan hanya kepada engkaulah kami meminta pertolongan.”

(QS. Al- Fatihah 5.)

“ Jika semuanya terasa mudah, percayalah ada yang salah dari sesuatu kau lakukan ”

(Om Dhuro)

“Kemewahan yang tersisa dari generasi muda adalah IDEALISME ”

(Marsandi)

“Sesungguhnya selepas kesukaran itu kemudahan”

(QS. Al- Insyirah 5-6)

“Barangsiapa ingin mutiara, harus berani terjun di lautan yang dalam”

(Ir. Soekarno)

PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Marsandi Arfianto

NIM : 101910201071

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “*monitoring dan pengontrolan pembebanan pada generator induksi satu fasa berbasis raspberry pi*” adalah benar – benar hasil karya sendiri, kecuali jika dalam pengutipan subtansi disebutkan sumbernya dan belum pernah diajukan pada institusi manapun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, Maret 2015

Yang menyatakan,

Marsandi Arfianto

NIM 101910201071

SKRIPSI

**MONITORING DAN PENGONTROLAN PEMBEBANAN
PADA GENERATOR INDUKSI SATU FASA BERBASIS
RASPBERRY PI**

Oleh

Marsandi Arfianto

NIM 101910201071

Pembimbing :

Dosen Pembimbing Utama : Samsul Bachri M, S.T.,M.MT.

Dosen Pembimbing Anggota : Dr. Bambang Sri Kaloko, S.T., M.T.

PENGESAHAN

Skripsi dengan judul : “*Monitoring Dan Pengontrolan Pembebanan Generator Induksi Satu Fasa Berbasis Raspberry Pi* “ telah diuji dan disahkan oleh Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember pada :

hari,tanggal : Kamis, 26 Maret 2015

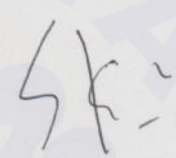
tempat : Fakultas Teknik, Universitas Jember.

Tim Penguji

Pembimbing Utama,

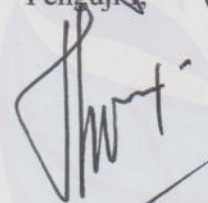
Pembimbing Anggota,

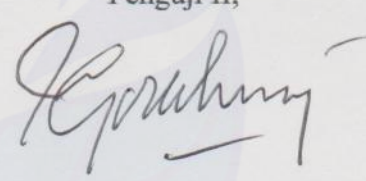

Samsul Bachri, M. S.T., M.MT.
NIP 19640317198021001


Dr. Bambang Sri Kaloko, S.T., M.T.
NIP 197104022003121001

Penguji I,

Penguji II,


Ir. Widyono Hadi, M.T.
NIP 196104141989021001


H.R.B.Moch. Gozali, S.T., M.T.
NIP 196906081999031002

Mengesahkan,
Dekan,

Ir. Widyono Hadi, M.T.
NIP 196104141989021001

**MONITORING DAN PENGONTROLAN PEMBEBANAN
PADA GENERATOR INDUKSI SATU FASA BERBASIS
RASPBERRY PI**

Marsandi Arfianto

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

ABSTRAK

Tenaga listrik sekarang ini sudah menjadi suatu kebutuhan *primer* bagi setiap penduduk, terutama didaerah perkotaan. Saat ini pembangkit listrik bedaya rendah semakin berkembang dengan memanfaatkan sumber daya mikrohidro dan angin sebagai energi penggerak yang ramah lingkungan. Biasanya pada pembangkit listrik berdaya rendah menggunakan generator induksi satu fasa yang memiliki konstruksi yang sederhana. Dalam pengoperasiannya nilai keluaran dari generator mengalami fluktuasi baik pada tegangan, arus maupun frekuaensi dikarenakan nilai beban yang tidak stabil. Oleh karena itu, dibutuhkan suatu sistem yang dapat memonitoring kinerja generator dan mengontrol pembebanannya.

Kata kunci : *Raspberry Pi, Generator, Monitoring, Pengontrolan*

***MONITORING AND CONTROL LOADING OF THE PHASE
INDUCTION GENERATOR BASED ON RASPBERRY PI***

Marsandi Arfianto

Department of Electrical Engineering, Engineering Faculty, University of Jember

ABSTRACT

Electricity today has become a primary needed for every population, especially in urban areas. Currently low-power electricity generation is growing by utilizing micro-hydro and wind resources as the driving force is environmentally friendly. Usually on a low-power electricity generation using a single phase induction generator that has a simple construction. In that operation, the output values of the generator have fluctuated both in voltage, current and frequency due to an unstable load values. Therefore, we need a system that can monitor the performance of the generator and controls the load system.

Keyword :Raspberry Pi, Generator, Monitoring, Controlling.

RINGKASAN

Monitoring Dan Pengontrolan Pembebanan Pada Generator Induksi Satu Fsa Berbasis Raspberry Pi; Marsandi Arfianto; 101910201071; 2015; 50 halaman; Program Studi Strata Satu Teknik, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember.

Tenaga listrik sekarang ini sudah menjadi suatu kebutuhan *primer* bagi setiap penduduk, terutama didaerah perkotaan. Saat ini pembangkit listrik bedaya rendah semakin berkembang dengan memanfaatkan sumber daya mikrohidro dan angin sebagai energi penggerak yang ramah lingkungan. Biasanya pada pembangkit listrik berdaya rendah menggunakan generator induksi satu fasa yang memiliki konstruksi yang sederhana. Dalam pengoprasiaannya nilai keluaran dari generator mengalami fluktuasi baik pada tegangan, arus maupun frekuensi dikarenakan nilai beban yang tidak stabil. Oleh karena itu, dibutuhkan suatu sistem yang dapat memonitoring kinerja generator dan mengontrol pembebanannya.

Alat ini digunakan untuk memonitoring kinerja generator saat beroperasi, mengontrol pembebanannya yang berupa beban lampu yang berfluktuatif dan menyimpan data – data dalam bentuk database. Ketiga fungsi tersebut dilakukan melalui HMI (*Human Machine Interface*) yang berbasis html sehingga user dapat mengakses HMI dari komputer lain secara *realtime* selama masih terhubung satu jaringan dengan *master station* yaitu raspberry pi.

Dalam pembangunan alat ini dibutuhkan beberapa rangkaian dan alat antara lain catu daya 5 V, mikrokontrol Arduino, *mini PC* Raspberry Pi sensor arus, sensor tegangan, sensor frekuensi, generator induksi satu fasa 1 KW, rangkaian kontrol beban, beban lampu, *software* Arduino dan *software* Phyton..

Pada penelitian ini pengujian dilakukan dengan menggunakan beban lampu 60 Watt sebanyak 5 buah. Pada pengujian dilakukan 5 kali pengambilan data, yaitu dengan beban 60 Watt, 120 Watt, 180 Watt, 240 Watt dan 300 Watt. Dari hasil pengujian didapatkan hasil saat beban 60 watt diaktifkan, pada HMI

muncul status tegangan sebesar 220 V, frekuensi sebesar 51 Hz dan Arus sebesar 0.19 A.

Dari hasil penelitian, analisa, dan pembahasan didapatkan beberapa kesimpulan. *Realtime* grafik pada HMI berfungsi dengan baik dalam menampilkan data pada pembebanan generator, *button switch* pada HMI berfungsi dengan baik dalam mengontrol beban lampu dan raspberry pi dapat dikatakan handal dalam penggunaannya.



PRAKATA

Bismillahirrohmanirrohim

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “*Monitoring Dan Pengontrolan Pembebanan Pada Generator Induksi Satu Fasa Berbasis Raspberry Pi*”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan beberapa pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada :

1. Allah SWT yang telah melimpahkan rahmat dan rizki-Nya serta memberi kelancaran dan kemudahan sehingga terselesaikannya skripsi ini.
2. Nabi Muhammad SAW yang telah membawa kita ke peradaban manusia yang lebih baik.
3. Bapak Ir. Widyono Hadi, M.T., selaku Dekan Fakultas Teknik Universitas Jember yang memberikan bimbingan dan pengarahan dalam penyusunan skripsi ini.
4. Bapak Samsul Bachri M, S.T.,M.MT., selaku Dosen Pembimbing Utama dan Bapak Dr. Bambang Sri Kaloko, S.T., M.T. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran guna memberikan bimbingan dan pengarahan dalam penyusunan skripsi ini.
5. Seluruh Dosen yang ada di Fakultas Teknik khususnya Teknik Elektro beserta karyawan.
6. Bunda Libatin, dan Ayah Halil, serta seluaruh keluarga. Terima kasih atas semua dukungan baik secara materi maupun moral dan kasih sayang serta doa restunya yang tulus kepada saya. Kalian semua tetap sabar dan selalu menyelipkan do'a untuk saya disetiap shalat dan perkataan kalian, hingga terselesaikannya studi ini.
7. Teman – teman para “*Jancukers Lab Patrang*” Angga, Budi, Yusqi, Resan, Riki, Bono terima kasih telah membantu dalam penyusunan skripsi ini.

8. Teman-teman *Angkatan 2000-X* Brian, Penyok, Embut, Azu, Oyek, Gerbong, Naser, MbahJimup, terima kasih banyak atas dukungan kalian selama ini. Dengan kalian saya tahu makna kebersamaan, kekompakan, secangkir kopi dan kesederhanaan dalam hidup.
9. Senior” lab Patrang, Cak Feri, Cak sugik, Cak Jazuli. Junior” lab Gunawan, Fajar, Heri, Sujarwo, bebek, Karyo, Hendro, Ibrahim terima kasih telah menemani selama menjadi penghuni Lab.
10. Special thanks to my “Tiyas” terima kasih telah menjadi salah satu motivasi saya untuk segera menyelesaikan studi ini.
11. Pihak - pihak yang tidak dapat saya sebutkan satu persatu, terima kasih atas dukungan dan motivasi kalian dalam penyusunan skripsi ini.

Semoga skripsi ini dapat bermanfaat dalam mengembangkan ilmu pengetahuan khususnya untuk disiplin ilmu teknik elektro. Kritik dan saran yang membangun diharapkan terus mengalir untuk lebih menyempurnakan skripsi ini dan dapat dikembangkan untuk penelitian selanjutnya.

Jember, Maret 2015

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBING	v
HALAMAN PENGESAHAN	vi
ABSTRAK	vii
RINGKASAN	ix
PRAKATA	xi
DAFTAR ISI	xiii
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Batasan Penelitian	2
BAB 2 TINJAUAN PUSTAKA	4
2.2 Generator	4
2.2.1 Generator AC.....	5
2.2.2 Konstruksi Generator AC	5
2.2.3 Prinsip Kerja Generator Induksi Satu fasa	6
2.3 Mikrokontroler Arduino	9
2.3.1 Arduino Uno	10
2.4 Raspberry Pi	11

BAB 3 METODOLOGI PENELITIAN	13
3.1 Prosedur Penelitian	13
3.2 Perancangan Sistem	14
3.4.1 <i>Power Supply</i>	15
3.4.2 Rangkaian Sensor Tegangan	15
3.4.3 Rangkaian Sensor Arus	16
3.4.4 Arduino Uno.....	17
3.4.5 Raspberry Pi	18
3.4.6 Beban Lampu	18
3.4.7 Desain Tampilan HMI.....	19
3.4.8 PC	20
3.3 Flowchart Sistem	20
BAB 4 ANALISIS DATA DAN PEMBAHASAN	25
4.1 Pengujian Sub Sistem	25
4.1.1 Pengujian Sensor Frekuensi.....	25
4.1.2 Pengujian Sensor Tegangan.....	27
4.1.3 Pengujian Sensor Arus.....	29
4.2 Pengujian Alat	31
4.2.1 Pengujian HMI, Respon Grafik dan Database.....	32
4.2.2 Pengujian Beban 60 Watt.....	35
4.2.3 Pengujian Beban 120 Watt.....	37
4.2.4 Pengujian Beban 180 Watt.....	39
4.2.5 Pengujian Beban 255 Watt.....	42
4.2.6 Pengujian Beban 315 Watt.....	44
4.2.7 Pengujian Beban 375 Watt.....	47
BAB 5 Penutup	51
5.1 Kesimpulan	51
5.2 Saran	52
DAFTAR PUSTAKA	53
LAMPIRAN	

DAFTAR TABEL

	Halaman
Tabel 3.1 Spesifikasi Arduino Uno.....	17
Tabel 4.1 Hasil pengujian sensor frekuensi	26
Tabel 4.2 Hasil pengujian sensor tegangan.....	28
Tabel 4.3 Hasil pengujian sensor arus.....	30
Tabel 4.4 Hasil pengujian <i>database</i> saat uji respon <i>interface</i>	34
Tabel 4.5 Data dari <i>database</i> saat pengujian beban 60 watt	36
Tabel 4.6 Data dari <i>database</i> saat pengujian beban 120 watt	38
Tabel 4.7 Data dari <i>database</i> saat pengujian beban 180 watt	41
Tabel 4.8 Data dari <i>database</i> saat pengujian beban 255 watt	43
Tabel 4.9 Data dari <i>database</i> saat pengujian beban 315 watt	46
Tabel 4.10 Data dari <i>database</i> saat pengujian beban 375 watt	49

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Konstruksi generator ac.....	6
Gambar 2.2 Kurva fungsi kerja mesin induksi terhadap slip.....	7
Gambar 2.3 Prinsip GGL induksi	8
Gambar 2.4 Proses timbulnya GGL induksi bolak – balik	9
Gambar 3.1 Monitoring dan pengontrolan beban pada generator induksi satu fasa berbasis raspberry pi	14
Gambar 3.2 Rangkaian <i>power supply</i>	15
Gambar 3.3 Rangkaian sensor tegangan	16
Gambar 3.4 Modul sensor arus ACS712-5A	16
Gambar 3.5 Arduino Uno.....	17
Gambar 3.6 Raspberry Pi model B	18
Gambar 3.7 Rangkaian beban lampu	19
Gambar 3.8 Desain tampilan HMI.....	20
Gambar 3.9 <i>Flowchart</i> sistem program pada <i>master station</i>	21
Gambar 3.10 Rangkaian sensor tegangan, sensor arus dan sensor frekuensi.....	20
Gambar 3.11 Gambar <i>Realtime</i> Grafik Pada <i>Web Plot.ly</i>	23
Gambar 3.12 <i>List</i> program grafik yang <i>diembedded</i> dari <i>plot.ly</i> ke <i>hmi</i>	23
Gambar 3.13 Rangkaian kontrol relay dengan masukan dari <i>gpio</i> raspberry pi	24
Gambar 4.1 Pengujian sensor frekuensi.....	25
Gambar 4.2 List program pembacaan sensor frekuensi pada Arduino	27
Gambar 4.3 Pengujian sensor Tegangan.....	27
Gambar 4.4 List program pembacaan sensor tegangan pada arduino.....	29
Gambar 4.5 Pengujian sensor Arus.....	30
Gambar 4.6 List program pembacaan sensor arus pada arduino	31
Gambar 4.7 Tampilan <i>realtime</i> grafik (a)	32

Gambar 4.7 Tampilan tombol kontrol beban (b)	33
Gambar 4.8 Respon grafik terhadap perubahan parameter	33
Gambar 4.9 Tampilan HMI saat beban 60 watt	35
Gambar 4.10 Grafik <i>database</i> pengujian beban 60 watt.....	37
Gambar 4.11 Tampilan HMI saat beban 120 watt	38
Gambar 4.12 Grafik <i>database</i> pengujian beban 120 watt.....	39
Gambar 4.13 Tampilan HMI saat beban 180 watt	40
Gambar 4.14 Grafik <i>database</i> pengujian beban 180 watt.....	42
Gambar 4.15 Tampilan HMI saat beban 225 watt	43
Gambar 4.16 Grafik <i>database</i> pengujian beban 225 watt.....	44
Gambar 4.17 Tampilan HMI saat beban 315 watt	45
Gambar 4.18 Grafik <i>database</i> pengujian beban 315 watt.....	47
Gambar 4.19 Tampilan HMI saat beban 375 watt	48
Gambar 4.20 Grafik <i>database</i> pengujian beban 375 watt.....	50

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Tenaga listrik sekarang ini sudah menjadi suatu kebutuhan *primer* bagi setiap penduduk, terutama didaerah perkotaan. Semakin berkembang dan meningkatnya kebutuhan akan tenaga listrik, maka kestabilan pada sistem tenaga listrik menjadi hal yang sangat penting dan harus diperhatikan, salah satu pembangkit listrik yang sering digunakan adalah generator AC (Ahmad, 2012). Pada prinsipnya generator membutuhkan energi mekanik yang berupa putaran yang dihasilkan oleh turbin (baik berupa turbin uap, turbin angin maupun turbin air). Kemudian putaran tersebut dikonversi menjadi energi listrik oleh generator sesuai dengan hukum Faraday, dimana apabila suatu penghantar digerak – gerakkan dalam medan magnet maka penghantar tersebut timbul GGL (Gaya Gerak Listrik) induksi dan dapat menghasilkan listrik. Generator induksi satu fasa merupakan salah satu jenis generator yang sangat menguntungkan untuk digunakan sebagai pembangkit tenaga listrik, khususnya pada penggunaan dengan daya listrik rendah. Dengan memanfaatkan sumber daya mikrohidro dan angin sebagai sumber energi penggerak yang merupakan energi ramah lingkungan. Kelebihan dari generator induksi satu fasa yaitu konstruksi sederhana, tidak membutuhkan sikat arang, handal, dan biaya perawatan rendah dan tidak memakai penguat dc (Djoekardi, 1996).

Dalam proses konversi energi pada generator induksi satu fasa memiliki keluaran yang berupa tegangan, arus, dan frekuensi. Saat generator tersebut dibebani maka nilai keluaran tersebut harus dijaga stabil sesuai dengan standar PLN yaitu nilai tegangan sama dengan 220 V dan nilai frekuensi sama dengan 50 Hz. Nilai beban yang berubah – ubah dapat menyebabkan nilai tegangan dan frekuensi tidak stabil, sehingga dibutuhkan suatu alat yang dapat mengontrol dan memonitor sistem pembebanan pada generator induksi satu fasa saat beroperasi secara *real time* dan dari jarak jauh.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut maka dapat dirumuskan masalah sebagai berikut :

1. Bagaimana merancang, membangun dan menganalisis sistem monitoring dan pengontrolan pada pembebanan generator induksi satu fasa berbasis Raspberry Pi.
2. Bagaimana sistem tersebut dapat memonitoring kinerja generator saat beroperasi dan mengontrol pembebanan pada generator secara *realtime* dan dilakukan dari jarak jauh.

1.3 Tujuan

Adapun tujuan pada penelitian ini adalah sebagai berikut :

1. Merancang, membangun dan menganalisis sistem sistem monitoring dan pengontrolan pada pembebanan generator induksi satu fasa berbasis Raspberry Pi.
2. Sistem tersebut dapat memonitoring kinerja generator saat beroperasi dan mengontrol pembebanan pada generator secara *realtime* dan dilakukan dari jarak jauh.

1.4 Manfaat

Manfaat yang diharapkan setelah penelitian ini tercapai adalah alat dapat digunakan untuk pembangkit listrik berdaya kecil yang menggunakan generator induksi satu fasa dengan beban yang fluktuatif sehingga saat generator beroperasi, semua keluaran baik tegangan, arus maupun frekuensi dapat dimonitoring dan beban yang terpasang pada generator dapat dikontrol secara *realtime* dan dari jarak jauh dengan menggunakan Raspberry Pi.

1.5 Batasan Penelitian

Agar dalam penulisan skripsi ini dapat mencapai sasaran dan tujuan yang diharapkan, maka dalam penelitian ini membatasi pembahasan menjadi beberapa permasalahan berikut :

1. Menggunakan Raspberry Pi sebagai *server* dan Arduino Uno sebagai pengolah data sensor..

2. Sistem HMI hanya dapat diakses dengan Ip tertentu pada jaringan lokal..
3. Sistem HMI dibuka dengan menggunakan *browser* pada komputer *client*.
4. Beban yang digunakan adalah lampu skala laboratorium.



BAB 2. TINJAUAN PUSTAKA

2.1 Generator

Generator listrik merupakan alat yang memproduksi energi listrik dari sumber energi mekanikal, biasanya dengan menggunakan induksi elektromagnetik. Proses ini dikenal sebagai pembangkit listrik. Sumber energi mekanikal bisa berupa resiprok atau turbin mesin uap, air yang jatuh melalui sebuah turbin maupun kincir air, mesin pembakaran dalam, turbin angin, engkol tangan, energi surya atau matahari, udara yang dimampatkan, atau apapun sumber energi mekanikal yang lain.

Prinsip dasar dari suatu generator sangat berhubungan dengan hukum Faraday yang menyatakan bahwa apabila sebuah konduktor digerakkan tegak lurus sejauh ds memotong suatu medan magnet dengan kerapatan fluks B , maka perubahan fluks pada konduktor dengan panjang efektif l adalah :

$$d = Bl ds \dots \dots \dots (2.1)$$

Apabila suatu generator diputar dengan kecepatan tertentu maka akan menghasilkan tegangan output yang dapat ditulis dengan persamaan sebagai berikut :

$$E = 4,44. f. N. \dots \dots \dots (2.2)$$

- Keterangan :
- E : tegangan output generator (V)
 - f : frekuensi stator (Hz)
 - N : jumlah lilitan
 - : fluks (Weber)

Tegangan output dari suatu generator dapat ditingkatkan dengan cara memperbanyak lilitan kumparan, menggunakan magnet permanen yang lebih kuat, mempercepat putaran dan menyisipkan inti besi lunak dalam kumparan.

2.1.1 Generator AC

Generator ini seperti yang dijelaskan di atas berfungsi sebagai pengubah energy mekanik menjadi energy listrik yang berupa tegangan bolak – balik (AC). Generator AC dibedakan menjadi generator sinkron dan asinkron. Generator sinkron bekerja pada kecepatan dan frekuensi yang konstan, dapat dikatakan besarnya frekuensi sebanding dengan kecepatan putar rotor. Sedangkan generator asinkron, pada rotornya diputar dengan kecepatan nominalnya dan kumparan medannya dicatu oleh jala-jala listrik PLN. Oleh karena itu terdapat selisih putaran medan dan putaran rotornya (slip).

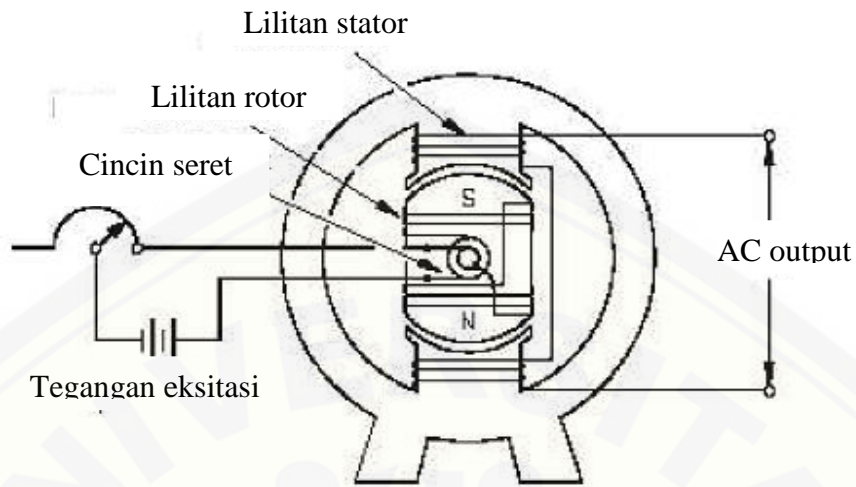
Pada generator AC 1 fasa merupakan suatu generator yang dimana dalam sistem melilitnya hanya terdiri dari satu kumpulan kumparan yang hanya dilukiskan dengan satu garis dan dalam hal ini tidak diperhatikan banyaknya lilitan. Ujung kumparan atau fasa yang satu dijelaskan dengan huruf besar X dan ujung yang satu lagi dengan huruf U.

Hampir semua generator mempunyai belitan jangkar sebagai tempat ggl arus bolak balik berupa stator yang diam dan struktur medan magnet berputar sebagai rotor.

2.1.2 Konstruksi Generator AC

Konstruksi dari suatu generator AC ini terdiri dari dua bagian utama yaitu stator dan rotor. Rotor merupakan bagian bergerak yang menghasilkan medan magnet yang menginduksikan ke stator. Bentuk rotor generator AC ada yang berbentuk kutub sepatu (salient pole) dan kutub silinder.

Stator merupakan bagian diam dari generator yang mengeluarkan tegangan bolak balik yang terdiri dari badan generator yang terbuat dari baja yang berfungsi melindungi bagian dalam generator, kotak terminal dan name plate pada generator. Inti Stator yang terbuat dari bahan ferromagnetik yang berlapis-lapis dan terdapat alur-alur tempat meletakkan lilitan stator. Lilitan stator yang merupakan tempat untuk menghasilkan tegangan yang akan dialirkan ke beban. Konstruksi dari generator AC dapat dilihat pada Gambar 2.1 di bawah ini :



Gambar 2.1 Konstruksi Generator AC

2.1.3 Prinsip Kerja Generator Induksi Satu fasa

Suatu generator mempunyai kecepatan medan putar stator lebih kecil daripada kecepatan putar rotor. Pada kondisi seperti ini nilai slip generator menjadi negatif. Slip adalah persentase perbedaan kecepatan medan putar stator dan rotor terhadap medan putar stator yang dinyatakan dengan :

$$S = \left| \frac{n_s - n_r}{n_s} \right| \times 100\% \dots\dots\dots(2.3)$$

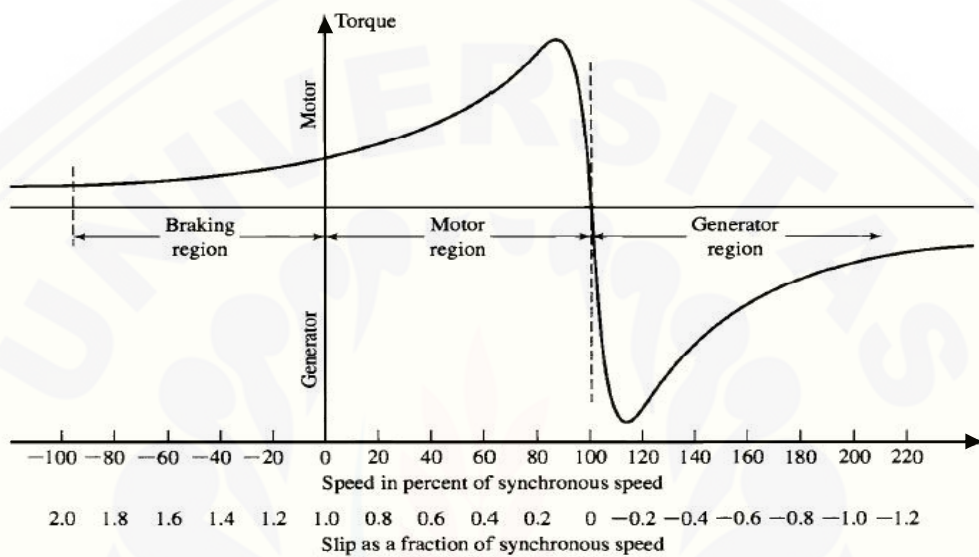
- Keterangan :
- S = slip
 - n_s = kecepatan medan putar stator (rpm)
 - n_r = kecepatan putar rotor (rpm)

Nilai n_r diperoleh dari putaran rotor yang dihasilkan oleh *prime mover* sedangkan nilai n_s dihasilkan oleh kumparan yang dialiri oleh arus dengan frekuensi tertentu. Besarnya n_s adalah :

$$n_s = \frac{120 \times f}{p} \dots\dots\dots(2.4)$$

- Keterangan :
- n_s = kecepatan medan putar stator (rpm)
 - f = frekuensi pada stator (Hz)
 - p = Jumlah kutub pada stator

Berubah-ubahnya kecepatan rotor mengakibatkan berubahnya harga slip dari 100% pada saat start mesin induksi ($n_r = 0$) menjadi 0% saat nilai $n_r = n_s$ atau saat kecepatan putar medan stator sama dengan kecepatan putar rotor (lihat Gambar 2.2). Harga slip juga dapat bernilai negatif ($S < 0$). Hal ini terjadi jika nilai putaran rotor lebih besar daripada nilai medan putar stator.



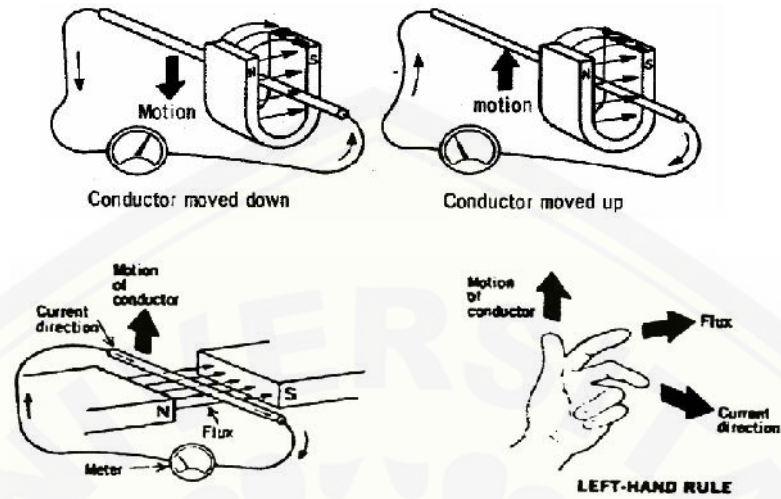
Gambar 2.2 Kurva fungsi Kerja Mesin Induksi Terhadap Slip

Sesuai hukum Faraday apabila suatu penghantar digerak-gerakkan dalam medan magnet maka penghantar tersebut timbul GGL (Gaya Gerak Listrik) induksi atau dapat menghasilkan listrik, yang besar GGL induksi tersebut adalah :

$$e = - N \frac{d\Phi}{dt} \dots\dots\dots(2.5)$$

- Keterangan :
- N = Jumlah penghantar
 - Φ = Fluxi / medan magnet
 - T = Persatuan waktu

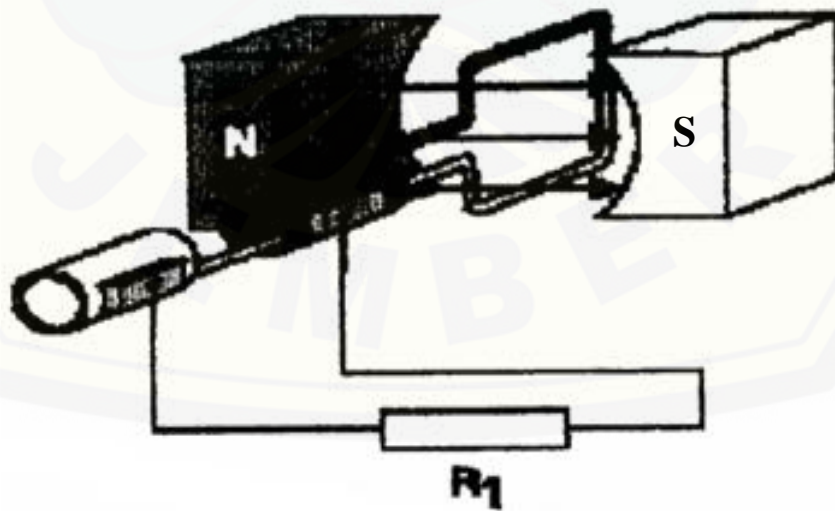
Adapun prinsip GGL induksi secara lengkap ada pada Gambar 2.3 di bawah ini :

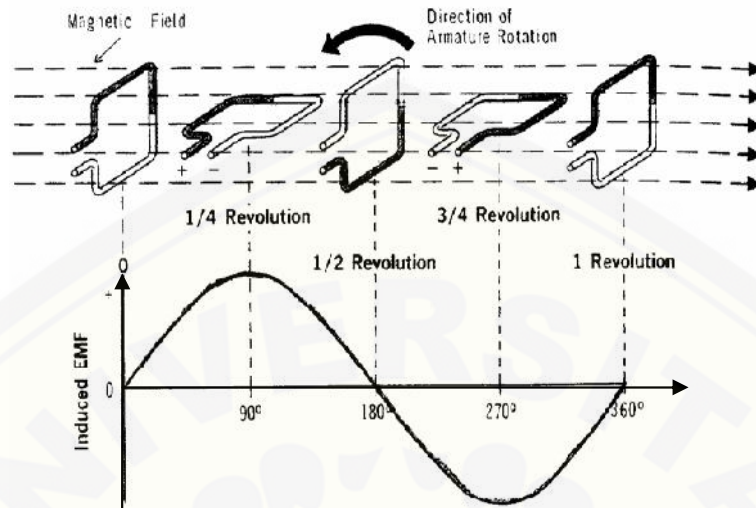


Gambar 2.3 Prinsip GGL Induksi

Berdasarkan prinsip tersebut yang ada pada gambar di atas digunakan sebagai dasar generator pembangkit listrik. Sehingga syarat terjadinya GGL induksi harus adanya medan magnet, penghantar dan kecepatan putaran.

Prinsip mesin listrik / Generator yang dapat membangkitkan listrik, seperti Gambar 2.4 berikut





Gambar 2.4 Proses Timbulnya GGL Induksi Bolak – balik

Pada Gambar 2.4 di atas menunjukkan sebuah gulungan penghantar diputar di dalam media medan magnet pada satu putaran (360°), menghasilkan GGL induksi arus bolak balik satu periode. Gelombang arus bolak-balik tersebut biasa disebut gelombang sinusoida sehingga apabila penghantar tersebut diputar oleh turbin dengan putaran 3000 rpm atau sama dengan putaran tiap detik 50 putaran, maka gelombang arus bolak-balik yang dihasilkan adalah juga sebanyak 0,02 periode (periode sama dengan waktu per banyak putaran) atau dikatakan dengan frekuensi 50 Hertz.

2.2 Mikrokontroler Arduino

Arduino adalah sebuah alat elektronik bebas yang mempunyai bentuk fleksibel, mudah digunakan baik *hardware* maupun *software*. Biasanya digunakan oleh para desainer elektronik, penggemar, dan siapapun yang membutuhkan interaktif obyek elektronik. Arduino dapat menerima semua input dari variasi sensor dan dapat menghasilkan kontrol yang digunakan untuk lampu, motor dan gerak mekanik yang lain. Pemrogramannya menggunakan bahasa pemrograman Arduino. Salah satu yang membuat arduino banyak diminati adalah karena

opensource baik *hardware* ataupun *software*, sehingga pengguna bebas untuk membeli, merangkai mengunduh programnya secara bebas dan gratis.

Alasan kenapa arduino dapat berkembang secara cepat adalah karena Murah dibandingkan mikrokontroller lain. *Software* Arduino dapat dijalankan semua sistem operasi baik Windows, Macintosh OSX dan Linux. Sangat mudah dipelajari dan digunakan, karena bahasa pemrograman arduino dialektanya mirip dengan C++ dan Java.

2.2.1 Arduino Uno

Arduino Uno menggunakan mikro control ATmega 328. Mempunyai 14 pin digital untuk *input/output* (dengan 6 buah pin dapat digunakan untuk keluaran PWM), 6 pin analog *input*, frekuensi 16 MHz, koneksi USB, *port supply*, ICSP *header*, tombol reset. Hal ini mendukung apapun yang dibutuhkan pada mikrokontrol, dapat *disupply* pada computer dengan kabel daya USB atau juga dari adaptor maupun baterai untuk memulai. ATmega 328 mempunyai 32KB (0,5KB digunakan untuk *bootloader*). Juga mempunyai kapasitas 2KB SRAM dan 1KB EEPROM.

Pin digital pada arduino uno terdapat 14 buah pin, pin tersebut dapat digunakan sebagai pin *input/output* dengan menggunakan aturan tulisan pada program yaitu *pinMode()*, *digital Write()*, dan *digital Read()*. Pin tersebut beroperasi pada tegangan 5V dan menahan arus maksimal 40mA. Pada kondisi tertentu beberapa pin mempunyai fungsi khusus:

1. Serial :0 (RX) dan 1 (TX). Digunakan untuk menerima dan mengirim data pada komunikasi serial. Pin ini terhubung pada *chip* serial.
2. *External Interrupts* : 2 dan 3. Pin ini dapat dikonfigurasi untuk *trigger* pada saat bernilai *LOW*, *RISSING* atau *FALLING*, atau bias merubah nilai. Dalam penulisan program gunakan fungsi *attach Interrupt()*.
3. PWM : 3, 5, 6, 9, 10 dan 11. Dapat digunakan dengan fungsi *analogWrite()* dengan kapasitas 8bit.
4. LED : 13. Pin 13 terhubung dengan LED sehingga apabila bernilai *HIGH* maka LED akan menyala, apabila bernilai *LOW* maka LED akan mati.

2.3 Raspberry Pi

Raspberry Pi adalah komputer papan tunggal (*Single Board Computer / SBC*) yang memiliki ukuran sebesar kartu kredit. Raspberry Pi bisa digunakan untuk berbagai keperluan, seperti *spreadsheet*, game dan bahkan bisa digunakan sebagai *media player* karena kemampuannya dalam memutar video *high definition*. Raspberry Pi dikembangkan oleh yayasan nirlaba Raspberry Pi *Foundation* yang diprakarsai sejumlah *developer* dan ahli komputer dari Universitas Cambridge, Inggris.

Ide dibalik komputer mungil ini diawali dari keinginan untuk mencetak generasi baru *programmer* pada tahun 2006 lalu. Seperti disebutkan dalam situs resmi Raspberry Pi *Foundation*, saat itu Eben Upton, Rob Mullins, Jack Lang dan Alan Mycroft dari Laboratorium Komputer Universitas Cambridge memiliki kekhawatiran melihat kian turunnya keahlian dan jumlah siswa yang berminat belajar ilmu komputer. Mereka kemudian mendirikan yayasan Raspberry Pi bersama dengan Pete Lomas dan David Braben pada tahun 2009. Tiga tahun kemudian, Raspberry Pi model B memasuki tahap produksi massal. Dalam peluncuran pertamanya pada akhir 2012 dalam beberapa jam saja sudah terjual 100.000 unit. Saat ini, raspberry pi telah terjual lebih dari 2,5 juta unit di seluruh dunia.

Raspberry pi memiliki dua model, yaitu model A dan model B. secara umum raspberry pi model B memiliki RAM sebesar 512 MB. Perbedaan pada model A dan model B terletak pada memori yang digunakan, model A menggunakan memori 256 MB dan model B dengan memori 512 MB. Selain itu, model B juga sudah dilengkapi dengan *Ethernet port* (kartu jaringan) yang tidak tersedia di model A. Desain raspberry pi didasarkan pada system Soc (*System on Chip*) Broadcom BCM2835 yang telah tertanam prosesor ARM1176JZF-S dengan *clock frequenct* sebesar 700 MHz, dan *VideoCore IV GPU*. Penyimpanan data didesain tidak untuk menggunakan *hard disk* atau *solid-state drive*, melainkan mengandalkan kartu memori SD (*SD memory card*) untuk *booting* dan media penyimpanan data jangka panjang. Raspberry pi merupakan komputer kecil yang

sangat murah dengan harga berkisar 25 dolar AS untuk model A dan 35 dolar AS untuk model B setiap unitnya.

Hardware raspberry pi tidak memiliki *real time clock*, sehingga OS harus memanfaatkan waktu pada jaringan server sebagai pengganti. Namun computer yang mudah dikembangkan ini dapat ditambahkan fungsi *real-time* (seperti DS1307) dan banyak lainnya melalui saluran GPIO (*General Purpose Input Output*) via antarmuka I²C (*Inter Integrated Circuit*). Raspberry pi bersifat *open source* (berbasis linux), raspberry pi bias dikembangkan sesuai kebutuhan penggunaanya. Sistem operasi utama raspberry pi menggunakan Debian GNU/Linux dan bahasa pemrograman Pyhon. Salah satu pengembang Os untuk raspberry pi telah meluncurkan sistem operasi yang diberi nama Raspbian, Raspbian diklaim mampu memaksimalkan perangkat raspberry pi. Sistem operasi tersebut dibuat berbasis Debian yang merupakan salah satu distribusi Linux Os. Berikut spesifikasi dari raspberry pi:

1. Chip : Broadcom BCM2835 (CPU, GPU, DSP dan SDRAM)
2. CPU : 700 MHz ARM1176JZF-S core (ARM6 family)
3. *Memory* (SDRAM) : 512 MB (*shared with GPU*)
4. USB 2.0 : 2 (via integrated USB hub)
5. *Video output* : Composite RCA (PAL and NTSC), HDMI (rev 1.3 and 1.4), raw LCD panels via DSI 14 HDMI resolutions from 640x350 to 1920x1200 plus various PAL and NTSC standards.
6. *Audio outputs* : 3.5 mm jack and HDMI
7. *Onboard storage* : SD / MMC / SDIO card slot.
8. *Onboard network* : 10 / 100 Ethernet (RJ45)
9. *Low-level peripherals* : 8xGPIO, UART, I²C bus, SPI bus with two chip selects, +3.3v, +5v and ground [58][63]
10. *Power ratings* : 700mA (3.5 W)
11. *Size* : 85.60x53.98 mm
12. *Weight* : 45 g
13. *Operating systems* : Debian GNU.Linux, Fedora, Arch Linux ARM and RISC OS

BAB 3. METODE PENELITIAN

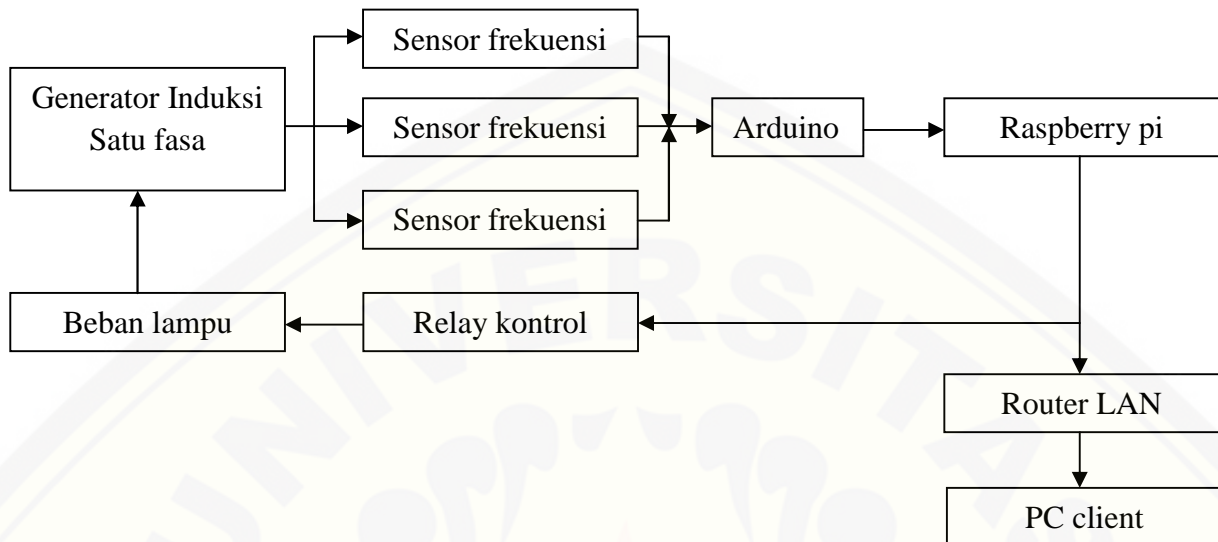
3.1 Prosedur Penelitian

Dalam pembuatan skripsi dan penelitian ini, dibuat langkah-langkah atau prosedur penelitian sebagai berikut :

1. Tahap persiapan, meliputi pembuatan laporan seminar proposal dan konsultasi kepada dosen pembimbing.
2. Studi literatur terhadap objek dan penelitian, meliputi pencarian materi tentang sistem monitoring, pengontrolan beban generator induksi dan Raspberry Pi baik berupa buku maupun literatur yang lain
3. Membangun rangkaian sensor tegangan, sensor arus, dan sensor frekuensi, yang nantinya nilai hasil pembacaan sensor akan diolah pada sistem yang dibangun
4. Membangun HMI (*Human machine Interface*), HMI berfungsi sebagai sarana *client* dalam memonitoring kinerja generator dan mengontrol pembebanan dari jarak jauh
5. Melakukan pengujian respon alat, setelah alat dan sistem selesai dibangun maka selanjutnya adalah melakukan pengujian terhadap alat dan system sehingga diperoleh hasil respon terhadap generator.
6. Pengambilan data, jika alat dan sistem sudah mampu berjalan dengan baik selanjutnya dilakukan pengambilan data terhadap pengujian yang dilakukan.
7. Analisis data, dari tahap pengambilan data dilakukan analisis data yang akan dijadikan acuan apakah alat dan system yang dibangun telah sesuai dengan keinginan.
8. Penyusunan Laporan, pada tahap penyusunan laporan semua hasil dari penelitian yang dilakukan baik berupa desain alat, pengujian alat, pengambilan data dan analisis data akan dilaporkan kepada dosen pembimbing dan kemudian akan diujikan sebagai syarat kelulusan dalam menempuh perkuliahan.

3.2 Perancangan Sistem

Perancangan sistem pada penelitian ini ditunjukkan pada gambar 3.1.

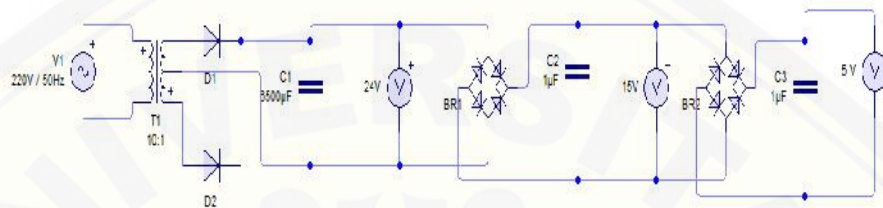


Gambar 3.1 Monitoring dan Pengontrolan Beban Pada Generator Induksi Satu Fasa Berbasis Raspberry Pi

Generator induksi mendapat putaran mekanik dari motor bakar (*prime mover*) yang dikople menggunakan *belt*. Nilai keluaran dari generator akan dibaca oleh sensor tegangan, sensor arus dan sensor frekuensi. Kemudian hasil pembacaan sensor – sensor tersebut diolah oleh arduino uno untuk mendapatkan nilai sensor berupa nilai digital. Nilai digital tersebut selanjutnya dikirim ke Raspberry pi melalui komunikasi serial data. Raspberry pi berfungsi sebagai *master station* yang mengontrol relay beban, menjadi pusat pengumpulan data yang akan dijadikan sebagai *database* dan tempat pusat pengoprasian HMI (*Human Machine Interface*). Sedangkan pada PC *client* berfungsi untuk operator dapat mengontrol, memonitor dan mengambil data pada generator induksi dari jarak jauh dengan cara memasukkan alamat *localhost* raspberry pi pada *browser* sehingga pada *browser* akan muncul tampilan dari HMI yang dapat diakses secara *real time*.

3.4.1 Rancangan Rangkaian *Power Supply*

Power supply yang digunakan dalam penelitian ini yaitu dengan mengubah tegangan AC menjadi tegangan DC dengan keluaran tegangan DC yaitu 5VDC untuk member sumber tegangan pada Raspberry Pi dan sistem pendingin *hardware* yang berupa kipas, seperti pada gambar 3.2.



Gambar 3.2 Rangkaian *Power Supply*

3.4.2 Rangkaian Sensor Tegangan

Rangkaian sensor tegangan berfungsi untuk membaca nilai tegangan keluaran dari generator. Terminal keluaran generator disambung dengan terminal masukan rangkaian sensor tegangan. Kemudian akan mengalir melalui sisi *input* dari optocoupler sehingga akan menghasilkan nilai resistansi pada sisi *output* dari optocoupler tersebut. Semakin besar nilai tegangan masukan pada rangkaian sensor tegangan, maka nilai resistansi pada sisi *output* optocoupler akan semakin besar. Selanjutnya, sisi *output* tersebut diberi masukan tegangan referensi sebesar 5 volt yang nantinya akan dijadikan nilai masukan pada arduino sehingga nilai tersebut dapat diolah dan dikalibrasi sesuai dengan tegangan keluaran generator. Berikut gambar rangkaian rangkaian sensor tegangan, seperti pada gambar 3.3.

3.4.4 Arduino Uno

Dalam penelitian ini, Arduino akan digunakan untuk mengolah nilai pembacaan sensor. Arduino yang digunakan dalam penelitian inia adalah Arduino Uno seperti yang ditunjukkan pada gambar 3.5 dan spesifikasid itunjukkan pada tabel 3.1 berikut:



Gambar 3.5 Arduino Uno

Tabel 3.1 Spesifikasi Arduino Uno

Mikrokontroler	Atmega328
Tegangan operasi	5V
V input(rekomendasi)	7-12V
V input(min/max)	6-20V
Pin I/O digital	14(6 untuk PWM output)
Pin analog	6
I DC per I/O pin	40mA
I DC untuk pin3.3V	50mA
Memori	32Kb, 8Kb untuk booting
SRAM	2Kb
EEPROM	1Kb
Clock speed	16MHz

Kecepatan aktual motor DC akan masuk ke *encoder* dan diumpkan balik kedalam Arduino. Dalam Arduino, selisih antara kecepatan referensi dan kecepatan aktual akan memicu PWM untuk mengatur driver motor DC menambah atau mengurangi kecepatan agar tercapai kecepatan yang diinginkan.

3.4.5 Raspberry Pi

Pada penilaian ini raspberry pi digunakan sebagai *master station*, dimana berfungsi sebagai pusat pengolah data dari kontroler arduino uno dan kontrol pembebanan pada sistem. Nantinya data tersebut akan diproses menjadi 2 bagian, yaitu sebagai acuan data pada realtime grafik yang ditampilkan pada HMI (*Human Machine Interface*) dan akan disimpan dalam database sebagai proses *logging data*. Untuk pengontrolan beban, pada raspberry pi terdapat GPIO (*General Pin Input Output*). Nantinya saat *switch button* pada HMI ditekan akan memberikan logika *high* atau *low* pada GPIO yang terhubung pada rangkaian kontrol beban. Gambar 3.6 Menunjukkan raspberry pi yang terhubung dengan arduino uno dan rangkaian kontrol beban



Gambar 3.6 Raspberry Pi Model B

3.4.6 Beban lampu

Pada penelitian ini beban yang digunakan adalah lampu pijar sebanyak 6 buah, dengan besar daya masing – masing lampu sebesar 60 Watt. Masing masing beban lampu dihubungkan dengan relay, sehingga untuk menghidupkan maupun

mematikan lampu sistem akan mengontrol relay pada tampilan HMI dari jarak jauh.



Gambar 3.7 Rangkaian Beban Lampu

3.4.7 Tampilan HMI (*Human Machine Interface*)

Pada penelitian ini tampilan HMI (*Human Machine Interface*) digunakan sebagai media *client* untuk mengakses data yang berupa tampilan dari keluaran generator yang berupa tegangan, arus dan frekuensi. Berikut desain tampilan HMI yang dibangun pada gambar 3.8.





Gambar 3.8 Desain Tampilan HMI

Tampilan *human machine interface* yang nantinya akan dibuka pada *pc client* berbasis HTML, dimana untuk menampilkan interface harus lewat aplikasi *browser* baik berupa mozilla maupun google chrome. Dari gambar desain tampilan HMI, *layout* dibagi menjadi 3 bagian, yaitu :

1. Pada angka 1 menunjukkan kotak alamat pencarian pada aplikasi *browser*. Untuk mengakses tampilan HMI, pada kolom ini masukkan alamat IP dari *master station*.
2. Pada angka 2 menunjukkan tampilan grafik dari ketiga pembacaan sensor yang meliputi tegangan, arus dan frekuensi.
3. Pada angka 3 menunjukkan bagian dari *switch button*, dimana bagian ini digunakan untuk mengontrol relay yang terhubung pada beban lampu.

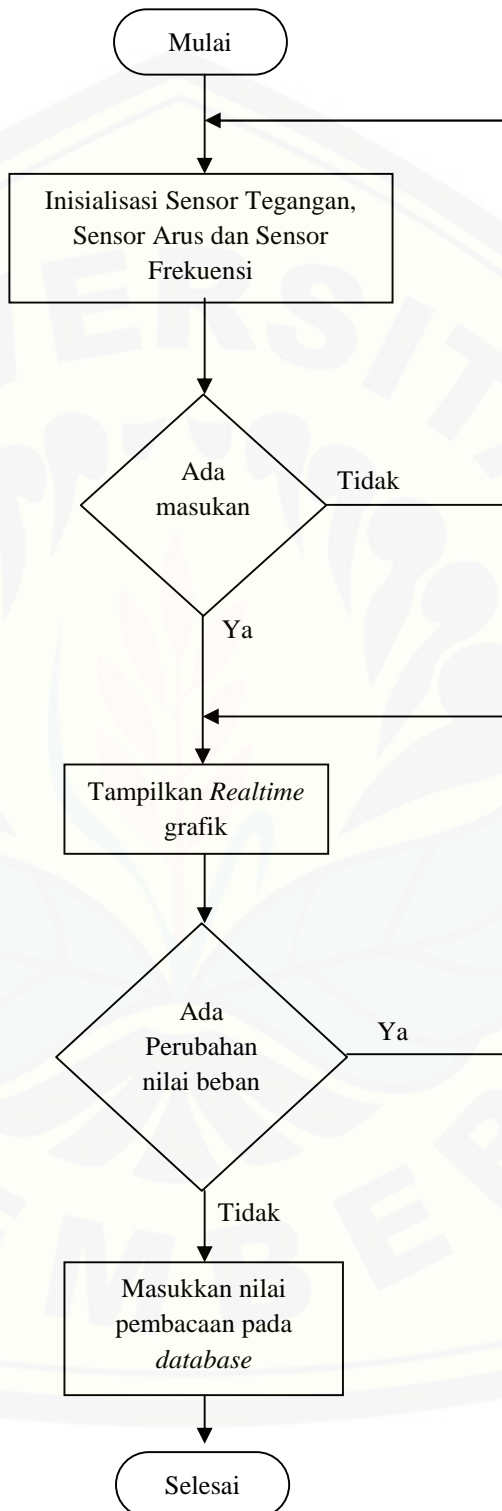
3.4.8 PC

Pada penelitian ini PC digunakan untuk memasukkan program kedalam Arduino dan sebagai *client* dari sistem SCADA ini. Pada sisi *client* digunakan untuk memonitoring dan mengontrol *plant* dari jarak jauh dengan memasukkan alamat IP dari *master station*.

3.3 Flowchart Sistem

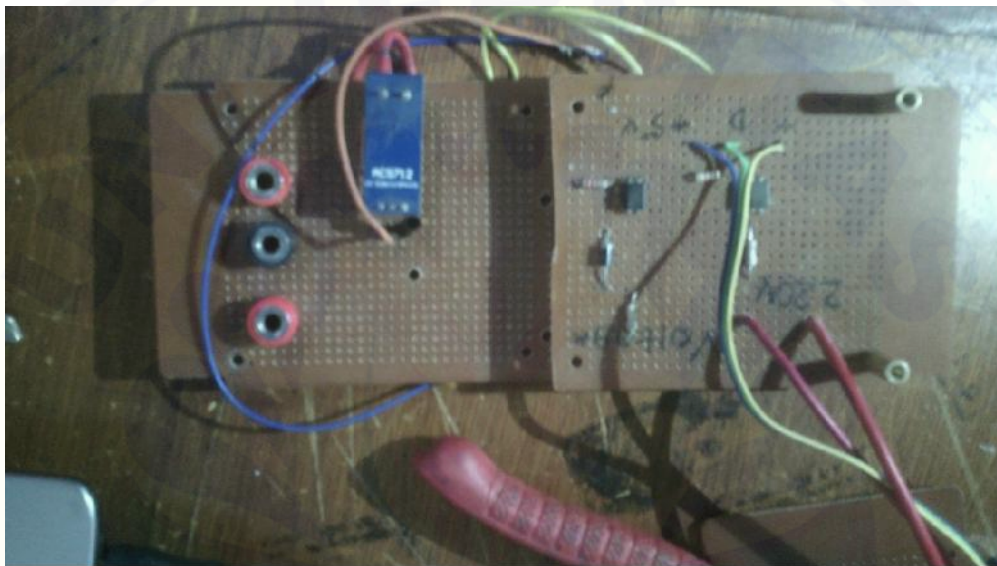
Cara kerja sistem ini dimulai dengan membaca parameter yang berupa nilai tegangan, arus dan frekuensi keluaran dari generator. Selanjutnya nilai yang

dibaca oleh sistem akan diproses menjadi 2 bagian, yaitu proses monitoring dan proses pengontrolan pembebanan.



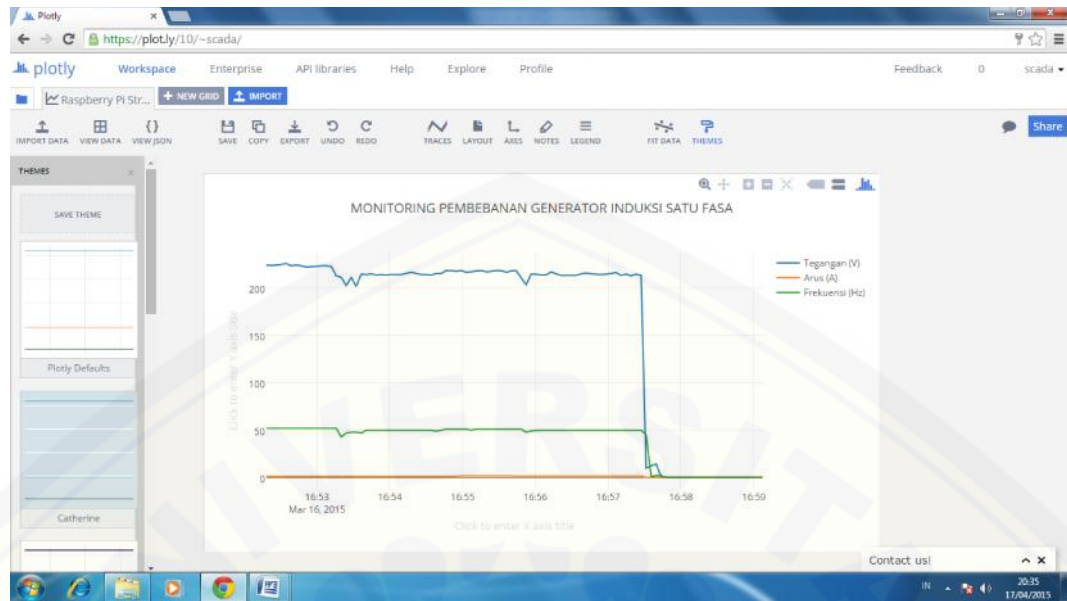
Gambar 3.9 Flowchart Sistem Program Pada Proses Monitoring

Pada proses monitoring digunakan untuk melihat dan menyimpan nilai parameter yang berupa tegangan, arus dan frekuensi. Gambar 3.9 menunjukkan tahapan alur dalam memonitor maupun menyimpan data parameter dari generator induksi satu fasa. Dimana pada tahap awal nilai keluaran dari generator dibaca oleh sensor – sensor yang selanjutnya akan diolah oleh kontroler Arduino. Gambar 3.10 menunjukkan sensor – sensor yang dipakai yang terdiri dari sensor arus, sensor tegangan dan sensor frekuensi.



Gambar 3.10 Rangkaian Sensor Tegangan, Sensor Arus Dan Sensor Frekuensi

Stelah data dari sensor diolah menjadi data digital oleh Arduino uno maka data tersebut selanjutnya dikirim ke raspberry pi menggunakan media USB serial. Data digital yang diterima kemudian diolah oleh program yang dibangun menggunakan bahasa python yang selanjutnya dikirim pada website Plot.ly. Gambar 3.11 menunjukkan data yang dikirim diubah menjadi realtime grafik. Kemudian realtime grafik tersebut diembedded pada tampilan HMI yang berbasis HTML



Gambar 3.11 Gambar *Realtime* Grafik Pada Web Plot.ly

```

width: 80px;
height: 70px;
font-size: 8pt;
font-weight: bold;
color: black;
}
<html>
<head></head></style><body>
<body background="elektro.jpg">
<div id="content" align="left">
  <a href="https://plot.ly/~scada/10/" target="_blank"
title="MONITORING PEMBEBANAN GENERATOR INDUKSI SATU FASA"
style="display: block; text-align: center;"></a>
  <script data-plotly="scada:10"
src="https://plot.ly/embed.js" async></script>

```

Gambar 3.12 *List* Program grafik Yang diembedded Dari Plot.ly Ke HMI

Sedangkan pada pengontrolan beban menggunakan relay – relay yang terhubung pada lampu yang dijadikan sebagai beban. Untuk mengontrol relay – relay tersebut maka dibuat program dengan menggunakan bahasa Python yang keluaran dari program tersebut berupa nilai digital (high atau low) melalui pin GPIO yang terdapat pada raspberry pi.



Gambar 3.13 Rangkaian Kontrol Relay Dengan masukan Dari GPIO Raspberry Pi

Untuk switch button pada tampilan HMI menggunakan bahasa HTML yang selanjutnya di link pada program pengontrolan sehingga saat tombol – tombol pada tampilan HMI ditekan akan memberi nilai high atau low pada pin GPIO pada raspberry pi. Kemudian pin GPIO tersebut dihubungkan pada pin masukan rangkaian relay – relay yang terhubung pada beban lampu. Sehingga beban lampu dapat dikontrol melalui switch button yang terdapat pada tampilan HMI.

BAB 4. ANALISIS DATA DAN PEMBAHASAN

Pada bab 4 ini akan membahas hasil penelitian Implementasi SCADA Pada Generator Induksi Satu Fasa Berbasis Raspberry Pi. Pengujian yang dilakukan menggunakan beban lampu yang berbeda - beda sebagai indikator perubahan pembebanan pada generator dan bagaimana sistem dapat memonitor *plant* dan mengontrol beban dari jarak jauh.

4.1 Pengujian Sub Sistem

Sebelum dilakukan pengujian alat secara keseluruhan, dilakukan pengujian-pengujian sub sistem, hal ini bertujuan agar tidak ada kendala saat dilakukan pengujian alat saat pengambilan data.

4.1.1 Pengujian Sensor Frekuensi (*Frequency Counter*)

Pengujian ini bertujuan untuk mengetahui respon fungsi sensor frekuensi saat generator dibebani. Gambar 4.1 menunjukkan pengujian sensor frekuensi. Pengujian ini mensimulasikan bahwa besarnya nilai tegangan keluaran generator mempengaruhi besarnya frekuensi sesuai dengan rumus dibawah ini :

$$E = K \cdot I_m \cdot f \dots \dots \dots (4.1)$$

- Dimana :
- E = Tegangan keluaran generator
 - K = Konstanta
 - I_m = Arus medan (A)
 - F = Frekuensi (Hz)



Gambar 4.1 Pengujian Sensor frekuensi

Setelah dilakukan pengujian sensor frekuensi didapatkan data hasil pengujian sensor frekuensi. Selanjutnya dibandingkan dengan hitungan rumus pembacaan. Tabel 4.1 menunjukkan hasil pembacaan dan perbandingannya.

Tabel 4.1 Hasil Pengujian Sensor Frekuensi

No	Nilai Tegangan	Pembacaan Sensor	Hasil Perhitungan	E %
1	100 V	25 Hz	23 Hz	8
2	150 V	36 Hz	34 Hz	5,56
3	175 V	43 Hz	40 Hz	6,97
4	220 V	52 Hz	50 Hz	3,84
5	240 V	58 Hz	54 Hz	6,89

Pengujian dilakukan sebanyak 5 kali dengan besarnya nilai tegangan yang berbeda – beda dengan menggunakan AVR (*Automatic Voltage Regulator*). Dapat dilihat pada table 4.1 saat tegangan diatur sebesar 100 V, nilai bacaan sensor 25 Hz dan hasil perhitungan 23 Hz sehingga timbul *error* % pembacaan yaitu 8 %. Untuk tegangan diatur sebesar 150 V, nilai bacaan sensor 36 Hz dan hasil perhitungan 34 Hz sehingga timbul *error* % pembacaan yaitu 5,56 %. Untuk tegangan diatur sebesar 175 V, nilai bacaan sensor 43 Hz dan hasil perhitungan 40 Hz sehingga timbul *error* % pembacaan yaitu 6,97 %. Untuk tegangan diatur sebesar 220 V, nilai bacaan sensor 52 Hz dan hasil perhitungan 50 Hz sehingga timbul *error* % pembacaan yaitu 3,84 %. tegangan diatur sebesar 240 V, nilai bacaan sensor 58 Hz dan hasil perhitungan 54 Hz sehingga timbul *error* % pembacaan yaitu 6,89 %. Pencarian nilai *error* % dapat dicari dengan persamaan 4.1

$$E \% = \left| \frac{\text{nilai sensor} - \text{nilai perhitungan}}{\text{nilai sensor}} \right| \times 100\% \dots \dots \dots (4.2)$$

$$E \% = \left| \frac{50 - 52}{50} \right| \times 100\%$$

$$E \% = 3,84 \%$$

Selisih hasil pengujian bacaan sensor dengan hasil perhitungan uang menjadi nilai *error* % tidak mencapai 30 % sehingga dapat disimpulkan bahwa nilai bacaan sensor tersebut valid. Gambar 4.2 menunjukkan *script* program dari sensor frekuensi.

```

//-----program frekuensi counter -----//
for;a=C;a<10;a++)
{
  for(i=0;i<1000;i++)
  {
    :
    data_adc_tegangan = analogRead(A1);
    data_tegangan=data_adc_tegangan;
    total_tegangan=total_tegangan+data_tegangan;
    :
    hasil_tegangan=total_tegangan/1000;
    hasil_jumlah=hasil_jumlah+hasil_tegangan;
    total_tegangan=0;
    hasil_tegangan=0;
  }
  hasil_rata2=abs(hasil_jumlah/10,-1023);
  frekuensi=hasil_rata2/10.2;// kalibrasi tegangan
  Serial.print(frekuensi);
  delay(500);
  total_tegangan=0;
  hasil_tegangan=0;
  hasil_rata2=0;
}

```

Gambar 4.2 List program Pembacaan Sensor Frekuensi Pada Arduino

4.1.2 Pengujian Sensor Tegangan

Pengujian ini bertujuan untuk mengukur jumlah tegangan keluaran dari generator yang masuk ke beban yang berupa lampu. Nilai pembacaan tersebut nantinya akan ditampilkan Alat yang digunakan yaitu AVR, rangkaian sensor tegangan, *power supply*, PC, dan *voltmeter*. Seperti pada gambar 4.3



Gambar 4.3 Pengujian sensor tegangan

Setelah dilakukan pengujian sensor tegangan didapatkan data hasil pengujian. Selanjutnya dibandingkan dengan pembacaan *voltmeter*. Tabel 4.2 menunjukkan hasil pembacaan dan perbandingannya.

Tabel 4.2 Hasil Pengujian Sensor Tegangan

No	V sensor	V Sumber	E%
1	105 V	100 V	5%
2	129 V	125 V	3,2%
3	155 V	150 V	3,3%
4	180 V	175 V	2,85%
5	205 V	200 V	2,5%
6	225 V	220 V	2,27%
7	249 V	250 V	0,47%

Pengujian dilakukan sebanyak 7 kali dengan besarnya nilai tegangan yang berbeda – beda dengan menggunakan AVR (*Automatic Voltage Regulator*). Dapat dilihat pada table 4.3 saat tegangan diatur sebesar 100 V, nilai bacaan sensor 105 V sehingga timbul *error* % pembacaan yaitu 5 %. Untuk tegangan diatur sebesar 125 V, nilai bacaan sensor 129 V sehingga timbul *error* % pembacaan yaitu 3,2 %. Untuk tegangan diatur sebesar 150 V, nilai bacaan sensor 155 V sehingga timbul *error* % pembacaan yaitu 3,3 %. Untuk tegangan diatur sebesar 175 V, nilai bacaan sensor 180 V sehingga timbul *error* % pembacaan yaitu 2,85 %. Untuk tegangan diatur sebesar 200 V, nilai bacaan sensor 205 V sehingga timbul *error* % pembacaan yaitu 2,5 %. Untuk tegangan diatur sebesar 220 V, nilai bacaan sensor 225 V sehingga timbul *error* % pembacaan yaitu 2,27 %. Untuk tegangan diatur sebesar 250 V, nilai bacaan sensor 249 V sehingga timbul *error* % pembacaan yaitu 0,47 % Pencarian nilai *error*% dapat dicari dengan persamaan 4.1

$$E \% = \left| \frac{\text{nilai sensor} - \text{nilai voltmeter}}{\text{nilai sensor}} \right| \times 100\% \dots \dots \dots (4.3)$$

$$E \% = \left| \frac{225 - 220}{225} \right| \times 100\%$$

$$E \% = 2,5 \%$$

Nilai bacaan pada sensor terdapat nilai koma karena pada program RPM didefinisikan sebagai nilai *float* yaitu nilai bilangan dengan dua angka dibelakang koma. Selisih hasil pengujian bacaan sensor dengan hasil perhitungan uang menjadi nilai *error* % tidak mencapai 30 % sehingga dapat disimpulkan bahwa nilai bacaan sensor tersebut valid. Gambar 4.4 menunjukkan *script* program sensor tegangan.

```

//-----program sensor tegangan-----//
for(a=0;a<10;a++)
{
  for(i=0;i<1000;i++)
  {
    data_adc_tegangan = analogRead(A1);
    data_tegangan=data_adc_tegangan;
    total_tegangan=total_tegangan+data_tegangan;
  }
  hasil_tegangan=total_tegangan/1000;
  hasil_jumlah=hasil_jumlah+hasil_tegangan;
  total_tegangan=0;
  hasil_tegangan=0;
}
hasil_rata2=abs((hasil_jumlah/10)-1023);
tegangan=hasil_rata2/2.40;// kalibrasi tegangan
Serial.println(tegangan);
delay(500);
total_tegangan=0;
hasil_tegangan=0;
hasil_rata2=0;
hasil_jumlah=0;
//-----//

```

Gambar 4.4 List Program Pembacaan Sensor Tegangan Pada Arduino

4.1.3 Pengujian Sensor Arus

Pengujian ini bertujuan untuk mengukur nilai arus keluaran dari generator yang masuk ke beban yang berupa lampu. Nilai pembacaan tersebut nantinya akan ditampilkan pada HMI (*Human Machine Interface*) dan dimasukkan kedalam system *database*. Alat yang digunakan yaitu *AC suply*, rangkaian sensor tegangan, *power supply*, PC, dan *amperemeter*. Seperti pada gambar 4.5.



Gambar 4.5 Pengujian Sensor Arus

Setelah dilakukan pengujian sensor arus didapatkan data hasil pengujian. Selanjutnya dibandingkan dengan *amperemeter*. Tabel 4.3 menunjukkan hasil pembacaan dan perbandingannya.

Tabel 4.3 Hasil Pengujian Sensor Arus

No	Nilai Beban	Sensor	Amperemeter	E %
1	60 W	0,27 A	0,27 A	0 %
2	120 W	0,54 A	0,54 A	0 %
3	180 W	0,81 A	0,81 A	0 %
4	240 W	0,99 A	1,09 A	0,1 %
5	300 W	1,30 A	1,36 A	0,046 %

Pengujian dilakukan sebanyak 5 kali sesuai dengan besarnya nilai beban yang berupa lampu bohlam dan tegangan yang diberikan sebesar 220 V. Dapat dilihat pada table 4.3 saat tidak ada beban atau beban 60 W, nilai bacaan sensor 0.27 A dan *amperemeter* 0,27 A, *error* % pembacaan yaitu 0 %. Untuk beban 120 W nilai sensor 0,54 A dan *amperemeter* 0,54 A, *error* % pembacaan yaitu 0%. Untuk beban 180 W nilai sensor 0,81 A dan *amperemeter* 0,81A, *error* %

pembacaan yaitu 0 %. Untuk beban 240 W nilai sensor 0,99 A dan *amperemeter* 1,09 A, *error* % pembacaan yaitu 0,01 %. Untuk beban 300 W nilai sensor 1,30 A dan *amperemeter* 1,36 A, *error* % pembacaan yaitu 0,046 %. Pencarian nilai *error*% dapat dicari dengan persamaan 4.1

$$E \% = \left| \frac{\text{nilai sensor} - \text{nilai amperemeter}}{\text{nilai sensor}} \right| \times 100\% \dots \dots \dots (4.4)$$

$$E \% = \left| \frac{0.99 - 1.09}{0.99} \right| \times 100\%$$

$$E \% = 0.1 \%$$

Nilai bacaan pada sensor terdapat nilai koma karena pada program RPM didefinisikan sebagai nilai *float* yaitu nilai bilangan dengan dua angka dibelakang koma. Selisih hasil pengujian bacaan sensor dengan *amperemeter* nilai *error* % tidak mencapai 30 % sehingga dapat disimpulkan bahwa nilai bacaan sensor tersebut valid. Gambar 4.6 menunjukkan *script* program sensor arus.

```

Serial.print(","); //split data pakai koma

//-----program sensor arus-----//

for(i=0;i<1000;i++)
{
  data_adc_arus = analogRead(A0);
  data_arus=abs(data_adc_arus-511);
  total_arus=total_arus+data_arus;
}
hasil_arus=total_arus/1000;
arus_A=hasil_arus/11.11; //kalibrasi arus
Serial.print(arus_A);
delay(500);
total_arus=0;
hasil_arus=0;
//-----//

```

Gambar 4.6 List program pembacaan sensor arus pada Arduino

4.2 Pengujian Alat

Pada tahap ini dilakukan pengujian alat secara keseluruhan, pada tahapan ini dilakukan pengujian dengan beban yang berbeda – beda yaitu 60 Watt, 120 Watt, 180 Watt, 255 Watt, 315 Watt dan 375 Watt. Pemilihan beban dilakukan

dengan pengontrolan melalui HMI (*Human Machine Interface*) dan kemudian melihat grafik dari hasil pengontrolan beban tersebut.

4.2.1 Tampilan Awal Sistem, Uji Respon Grafik dan Uji Database

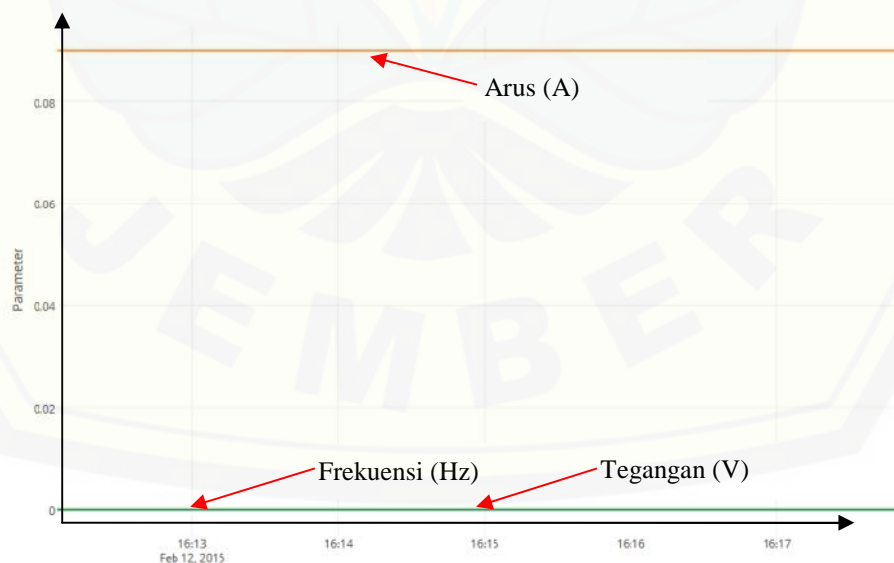
Saat sistem diaktifkan yang berupa *master station* (raspberry pi), *remote terminal unit* (arduino) dan sensor – sensor. Selanjutnya secara otomatis sistem telah membaca masukan – masukan yang berupa nilai tegangan, arus dan frekuensi, memasukkan pembacaan data masukan – masukan kedalam sistem penyimpanan *database* dan dapat menampilkan tampilan HMI yang dapat dilihat dari sisi *client*. Tampilan HMI terdiri dari 2 bagian yaitu :

a) *Realtime* grafik

Pada *realtime* grafik akan ditampilkan nilai dari ketiga indikator yang dimonitoring yaitu tegangan, arus dan frekuensi. Visualisasi dari *realtime* grafik sistem dapat dilihat pada gambar 4.1 (a).

b) *Button switch*

Button switch terdiri dari 12 tombol dimana 6 tombol memiliki fungsi untuk mengaktifkan relay dan 6 tombol yang lain berfungsi menonaktifkan relay. Gambar dari tampilan *button switch* dapat dilihat pada gambar 4.1 (b)



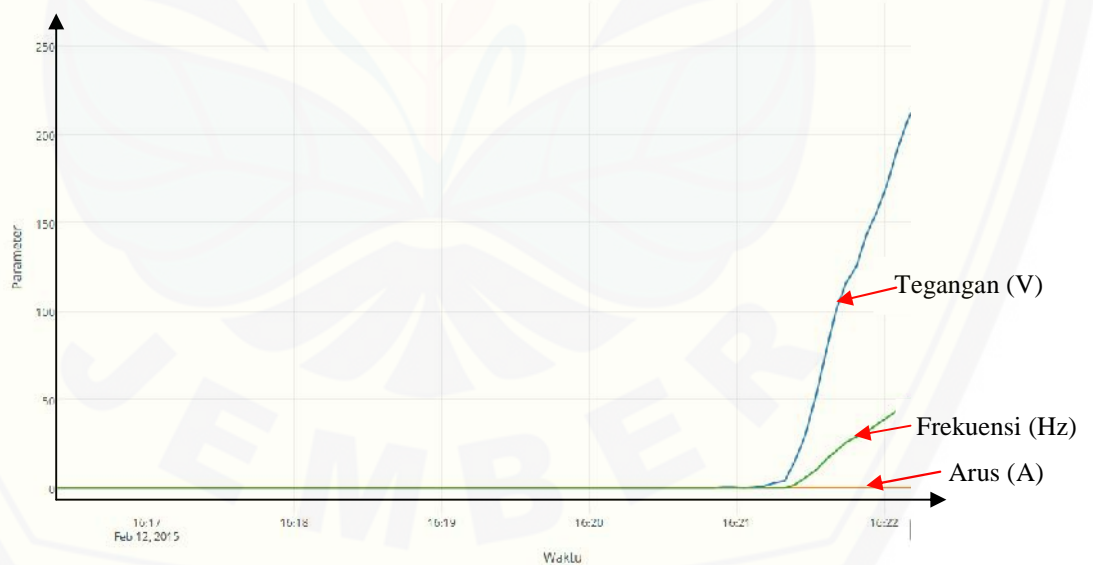
(a)



(b)

Gambar 4.7 (a) tampilan *realtime* grafik (b) tampilan tombol kontrol beban

Kemudian setelah tampilan HMI (*Human Machine Interface*) muncul dilakukan pengujian respon terhadap perubahan tegangan input. Dari gambar 4.8 Pada garis biru dapat dilihat bahwa interface dapat berjalan dengan baik karena dapat langsung mendeteksi perubahan tegangan masukan.



Gambar 4.8 Gambar Respon Grafik Terhadap Perubahan Parameter

Selanjutnya sistem akan menyimpan data dari hasil pembacaan sensor pada sistem *database*. Dari perubahan garis biru pada gambar 2. Nilai tegangan dan frekuensi yang mengalami perubahan. Perubahan – perubahan nilai tersebut juga tersimpan didalam *database*. Perubahan tersebut dapat dilihat pada tabel 4.4

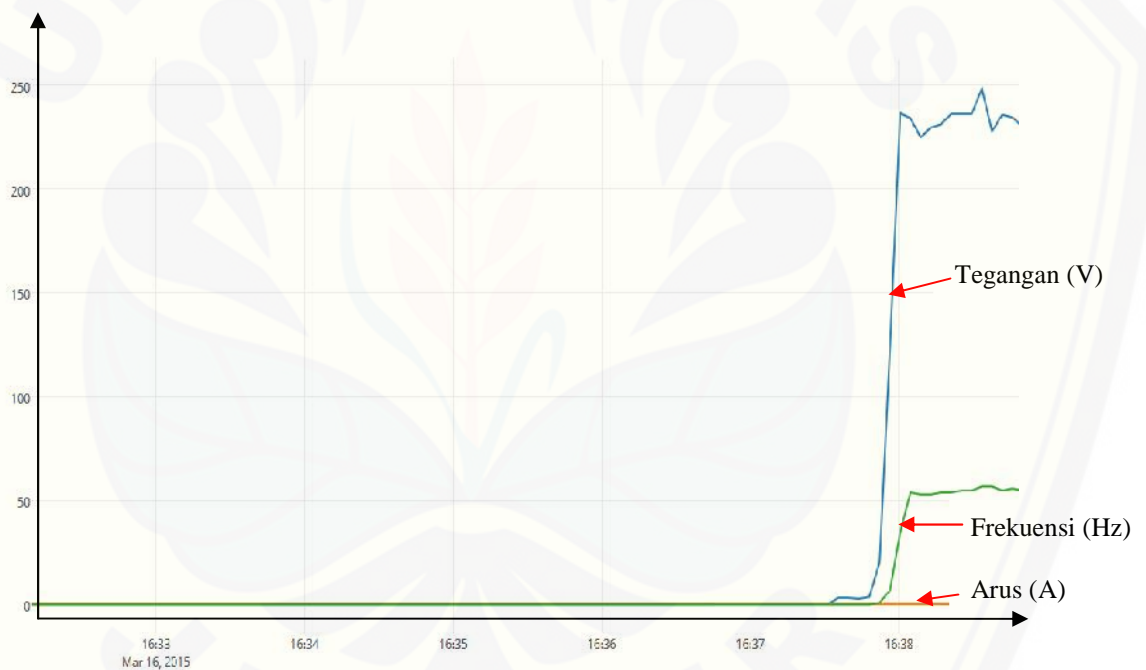
Tabel 4.4 Data *Database* Saat Uji Respon *Interface*

No	Waktu	Frekuensi	Arus	Tegangan	RPM
6	2015-02-12 16:21:23.830888	2 Hz	0.09 A	15.42 V	438
7	2015-02-12 16:21:28.004547	6 Hz	0.09 A	30.00 V	687
8	2015-02-12 16:21:32.174710	10 Hz	0.09 A	51.25 V	846
9	2015-02-12 16:21:36.343304	16 Hz	0.09 A	76.25 V	1068
10	2015-02-12 16:21:40.512607	21 Hz	0.09 A	100.42 V	1354
11	2015-02-12 16:21:44.680060	26 Hz	0.09 A	116.25 V	1534
12	2015-02-12 16:21:48.843823	29 Hz	0.09 A	125.42 V	1735
13	2015-02-12 16:21:53.013999	32 Hz	0.09 A	143.75 V	1926
14	2015-02-12 16:21:57.184519	36 Hz	0.09 A	156.25 V	2037
15	2015-02-12 16:22:01.357329	40 Hz	0.09 A	172.08 V	2264
16	2015-02-12 16:22:05.522391	44 Hz	0.09 A	191.67 V	2584
17	2015-02-12 16:22:09.655345	48 Hz	0.09 A	207.92 V	2648
18	2015-02-12 16:22:13.780006	51 Hz	0.09 A	220.00 V	2890
27	2015-02-12 16:22:50.803228	55 Hz	0.09 A	235.00 V	3097
28	2015-02-12 16:22:54.915207	55 Hz	0.09 A	244.58 V	3116

Dari tabel 4.4 Dapat dilihat pada kolom waktu merupakan waktu saat pengiriman data dari arduino, waktu yang dicatat oleh sistem sudah sangat lengkap yaitu berupa tanggal dan jam. Data akan dikirim dan ditampilkan setiap 4 detik, hal ini dapat dilihat selisih antar masing – masing data. Pada kolom frekuensi merupakan nilai pembacaan frekuensi, kolom selanjutnya merupakan hasil pembacaan arus dan kolom terakhir merupakan kolom tegangan. Dari kolom tegangan dapat dilihat bahwa perubahan – perubahan nilai tegangan akan selalu terbaca dan disimpan oleh sistem.

4.2.2 Pengujian Beban 60 Watt

Pada pengujian beban 60 Watt dilakukan selama 2 menit. Untuk mengaktifkan beban 60 Watt tekan tombol “lampu 1 on” sehingga lampu 1 akan menyala. Kemudian paantau kondisi baik tegangan, arus maupun frekuensi pada HMI (*Human Machine Interface*) selama durasi yang ditentukan. Dapat dilihat pada gambar 4.9 saat lampu 1 diaktifkan pada garis yang berwarna jingga menunjukkan bahwa arus yang mengalir saat beban lampu 60 Watt sebesar 0.27 A, garis hijau menunjukkan nilai frekuensi sebesar 54 Hz dan pada garis biru menunjukkan besar nilai tegangan sebesar 225,42 V. Untuk data selengkapnya dapat dilihat pada tabel 4.5, dimana data tabel tersebut berasal dari *database* sistem selama 2 menit.



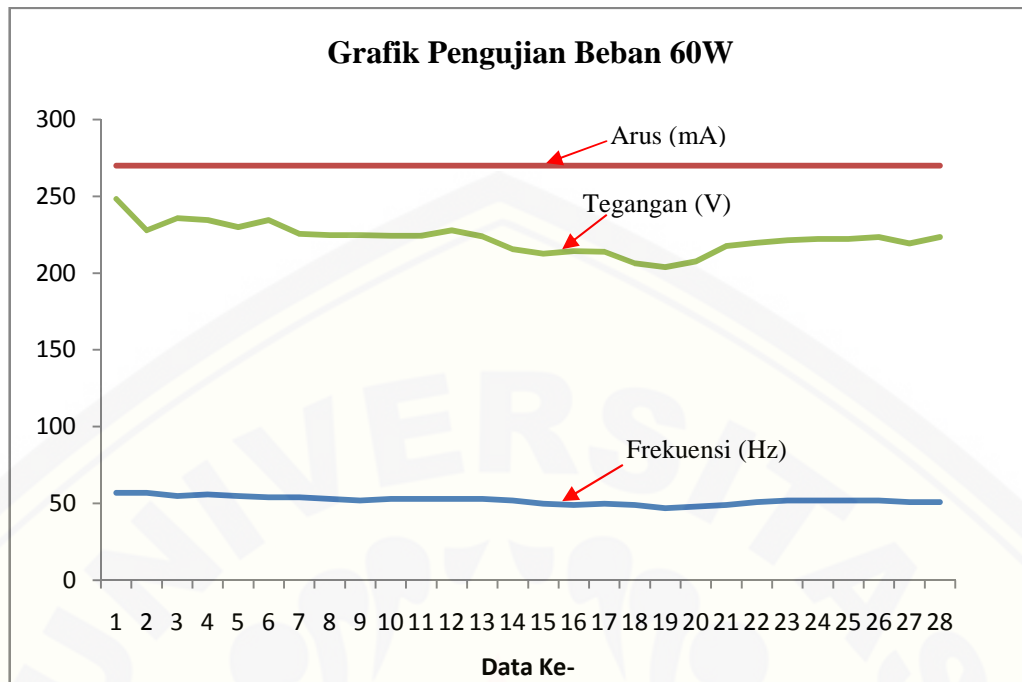
Gambar 4.9 Tampilan HMI Saat Beban 60 Watt

Dari tabel 4.5 Kemudian dijadikan bentuk grafik garis. Dilihat dari grafik garis yang dibuat, nilai pembacaan arus terlihat konstan (pada garis biru) yaitu stabil pada arus 270 mA. Namun pada nilai pembacaan frekuensi dan tegangan terlihat bahwa adanya perubahan pembacaan pada kedua nilai tersebut. Namun selisih nilai pembacaan antar data pada kedua parameter tersebut tidak terlalu jauh sehingga masih bisa dikatakan

bahwa sistem SCADA berjalan dengan baik pada pengujian beban 60 Watt. Gambar 4.10 menunjukkan grafik dari tabel *database*.

Tabel 4.5 Data dari *Database* Saat Pengujian Beban 60 Watt

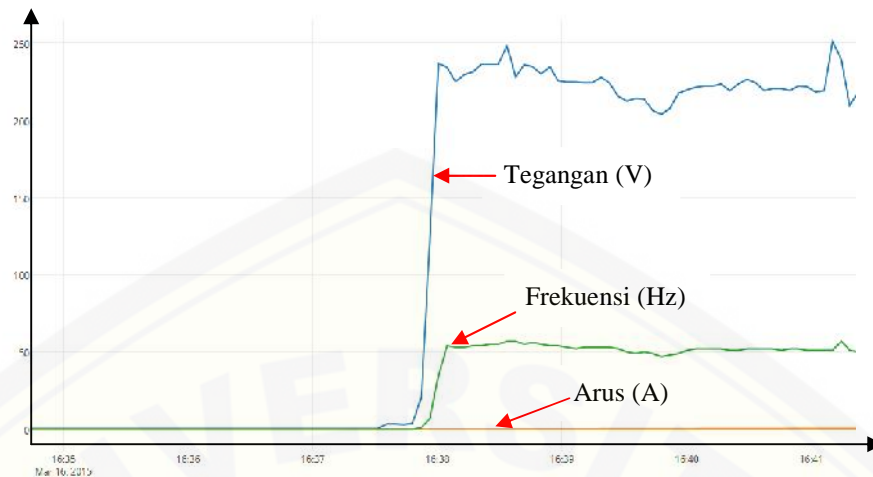
No	Waktu	Frekuensi	Arus	Tegangan
1	2015-03-16 16:38:33.511183	57 Hz	0.27 A	248.33 V
2	2015-03-16 16:38:37.628094	57 Hz	0.27 A	227.92 V
3	2015-03-16 16:38:41.745680	55 Hz	0.27 A	235.83 V
4	2015-03-16 16:38:45.863345	56 Hz	0.27 A	234.58 V
5	2015-03-16 16:38:49.985401	55 Hz	0.27 A	230.00 V
6	2015-03-16 16:38:54.103030	54 Hz	0.27 A	234.58 V
7	2015-03-16 16:38:58.225512	54 Hz	0.27 A	225.42 V
8	2015-03-16 16:39:02.349749	53 Hz	0.27 A	224.58 V
9	2015-03-16 16:39:06.475614	52 Hz	0.27 A	224.58 V
10	2015-03-16 16:39:10.598445	53 Hz	0.27 A	224.17 V
11	2015-03-16 16:39:14.722517	53 Hz	0.27 A	224.17 V
12	2015-03-16 16:39:18.845310	53 Hz	0.27 A	227.92 V
13	2015-03-16 16:39:22.970572	53 Hz	0.27 A	223.75 V
14	2015-03-16 16:39:27.100031	52 Hz	0.27 A	215.42 V
15	2015-03-16 16:39:31.229852	50 Hz	0.27 A	212.50 V
16	2015-03-16 16:39:35.359788	49 Hz	0.18 A	214.17 V
17	2015-03-16 16:39:39.489794	50 Hz	0.27 A	213.75 V
18	2015-03-16 16:39:43.623279	49 Hz	0.27 A	206.25 V
19	2015-03-16 16:39:47.761695	47 Hz	0.27 A	203.75 V
20	2015-03-16 16:39:51.901185	48 Hz	0.27 A	207.50 V
21	2015-03-16 16:39:56.029369	49 Hz	0.27 A	217.50 V
22	2015-03-16 16:40:00.154988	51 Hz	0.27 A	219.58 V
23	2015-03-16 16:40:04.281171	52 Hz	0.27 A	221.25 V
24	2015-03-16 16:40:08.406696	52 Hz	0.27 A	222.08 V
25	2015-03-16 16:40:12.532404	52 Hz	0.27 A	222.08 V
26	2015-03-16 16:40:16.657499	52 Hz	0.27 A	223.33 V
27	2015-03-16 16:40:20.784387	51 Hz	0.27 A	219.17 V
28	2015-03-16 16:40:24.909761	51 Hz	0.27 A	223.33 V



Gambar 4.10 Grafik *Database* Pengujian Beban 60 Watt

4.2.3 Pengujian beban 120 Watt

Pada pengujian beban 120 Watt dilakukan selama 2 menit. Untuk mengaktifkan beban 120 Watt tekan tombol “lampu 1 on” dan “lampu 2 on” sehingga lampu 1 dan lampu 2 akan menyala sehingga didapatkan nilai beban 120 Watt karena kedua lampu terhubung secara paralel. Kemudian paantau kondisi baik tegangan, arus maupun frekuensi pada HMI (*Human Machine Interface*) selama durasi yang ditentukan. Berikut tampilan HMI saat kedua lampu diaktifkan secara bersamaan. Dapat dilihat pada gambar 4.11 saat lampu 1 dan lampu 2 diaktifkan pada garis yang berwarna jingga menunjukkan bahwa arus yang mengalir saat beban lampu 120 Watt sebesar 0.54 A, garis hijau menunjukkan nilai frekuensi sebesar 52 Hz dan pada garis biru menunjukkan besar nilai tegangan sebesar 215,58 V. Untuk data selengkapnya dapat dilihat pada tabel 4.6, dimana data tabel tersebut berasal dari *database* sistem selama 2 menit.



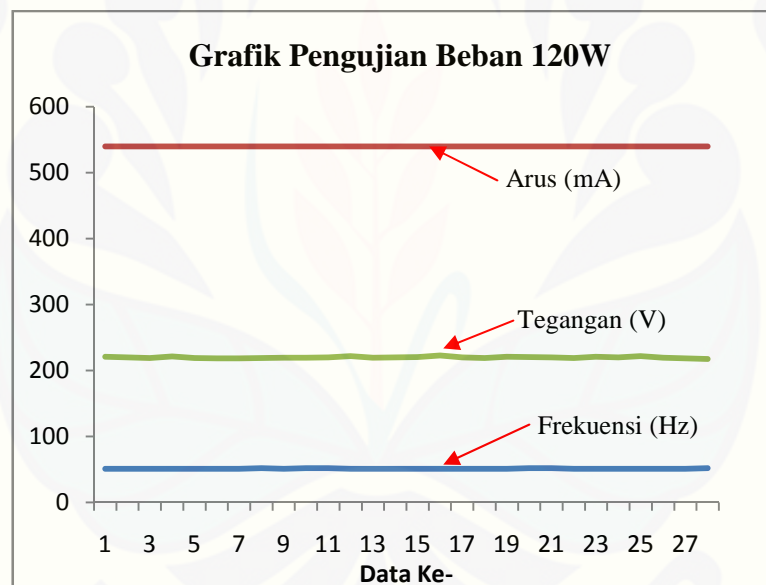
Gambar 4.11 Tampilan HMI Saat Beban 120 Watt

Tabel 4.6 Data dari *Database* Saat Pengujian Beban 120 Watt

No	Waktu	Frekuensi	Arus	Tegangan
1	2015-03-16 16:41:43.235620	51 Hz	0.54 A	220.83 V
2	2015-03-16 16:41:47.353685	51 Hz	0.54 A	220.00 V
3	2015-03-16 16:41:51.471154	51 Hz	0.54 A	219.17 V
4	2015-03-16 16:41:55.592532	51 Hz	0.54 A	221.67 V
5	2015-03-16 16:41:59.709636	51 Hz	0.54 A	219.17 V
6	2015-03-16 16:42:03.827558	51 Hz	0.54 A	218.33 V
7	2015-03-16 16:42:07.947322	51 Hz	0.54 A	218.33 V
8	2015-03-16 16:42:12.066842	52 Hz	0.54 A	219.17 V
9	2015-03-16 16:42:16.184429	51 Hz	0.54 A	219.58 V
10	2015-03-16 16:42:20.301900	52 Hz	0.54 A	219.58 V
11	2015-03-16 16:42:24.418874	52 Hz	0.54 A	220.00 V
12	2015-03-16 16:42:28.536438	51 Hz	0.54 A	222.08 V
13	2015-03-16 16:42:36.771200	51 Hz	0.54 A	219.58 V
14	2015-03-16 16:42:45.009744	51 Hz	0.54 A	220.00 V
15	2015-03-16 16:42:53.007253	51 Hz	0.54 A	220.42 V
16	2015-03-16 16:43:01.243209	51 Hz	0.54 A	222.92 V
17	2015-03-16 16:43:09.477134	51 Hz	0.54 A	220.00 V
18	2015-03-16 16:43:17.710029	51 Hz	0.54 A	218.75 V
19	2015-03-16 16:43:25.958780	51 Hz	0.54 A	220.83 V
20	2015-03-16 16:43:34.181063	52 Hz	0.54 A	220.42 V
21	2015-03-16 16:43:42.411976	52 Hz	0.54 A	220.00 V
22	2015-03-16 16:43:50.645631	51 Hz	0.54 A	219.17 V
23	2015-03-16 16:43:58.877829	51 Hz	0.54 A	220.83 V

24	2015-03-16 16:44:07.110558	51 Hz	0.54 A	220.00 V
25	2015-03-16 16:44:11.232229	51 Hz	0.54 A	222.08 V
26	2015-03-16 16:44:15.347422	51 Hz	0.54 A	219.58 V
27	2015-03-16 16:44:19.464448	51 Hz	0.54 A	218.33 V
28	2015-03-16 16:44:23.580291	52 Hz	0.54 A	217.50 V

Dari tabel 4.6 Kemudian dijadikan bentuk grafik garis. Dilihat dari grafik garis yang dibuat, nilai pembacaan arus terlihat konstan (pada garis biru) yaitu stabil pada arus 540 mA. Namun pada nilai pembacaan frekuensi dan tegangan terlihat bahwa adanya perubahan pembacaan pada kedua nilai tersebut. Namun selisih nilai pembacaan antar data pada kedua parameter tersebut tidak terlalu jauh sehingga masih bisa dikatakan bahwa sistem SCADA berjalan dengan baik pada pengujian beban 120 Watt. Gambar 4.12 menunjukkan grafik dari tabel *database* pengujian beban 120 Watt.

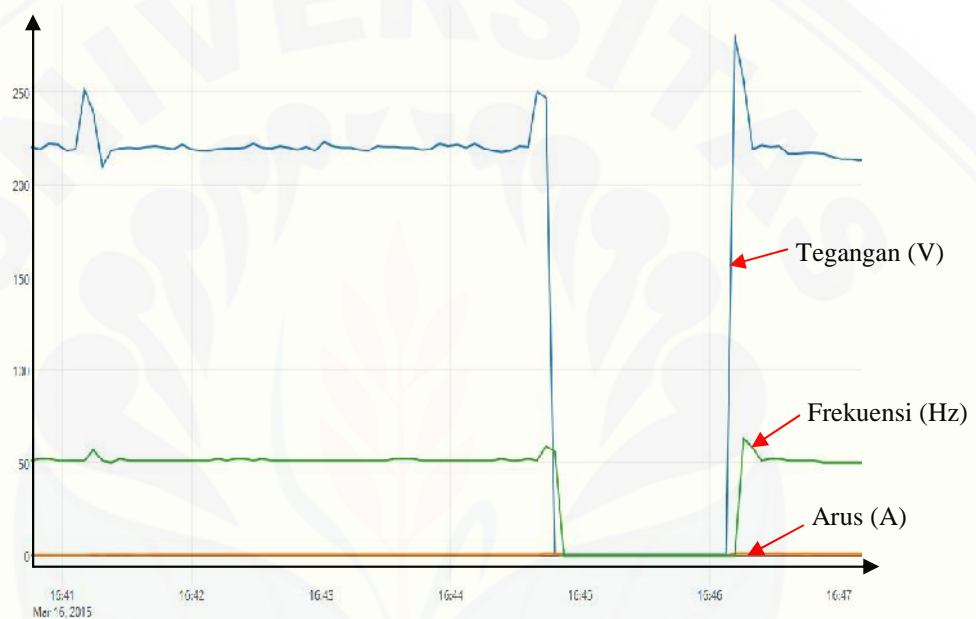


Gambar 4.12 Grafik *Database* Pengujian Beban 120 Watt

4.2.4 Pengujian beban 180 Watt

Pada pengujian beban 180 Watt dilakukan selama 2 menit. Untuk mengaktifkan beban 180 Watt tekan tombol “lampu 1 on”, “lampu 2 on” dan “lampu 3 on” sehingga lampu 1, lampu 2 dan lampu 3 akan menyala. Kemudian paantau kondisi baik tegangan, arus maupun frekuensi pada HMI (*Human Machine Interface*) selama durasi yang ditentukan. Berikut tampilan HMI saat

lampu ketiga lampu diaktifkan secara bersamaan. Dapat dilihat pada gambar 4.13 saat lampu 1, lampu 2 dan lampu 3 diaktifkan pada garis yang berwarna jingga menunjukkan bahwa arus yang mengalir saat beban lampu 180 Watt sebesar 0.81 A, garis hijau menunjukkan nilai frekuensi sebesar 51 Hz dan pada garis biru menunjukkan besar nilai tegangan sebesar 213.75 V. Untuk data selengkapnya dapat dilihat pada tabel 4.7, dimana data tabel tersebut berasal dari *database* sistem selama 2 menit.



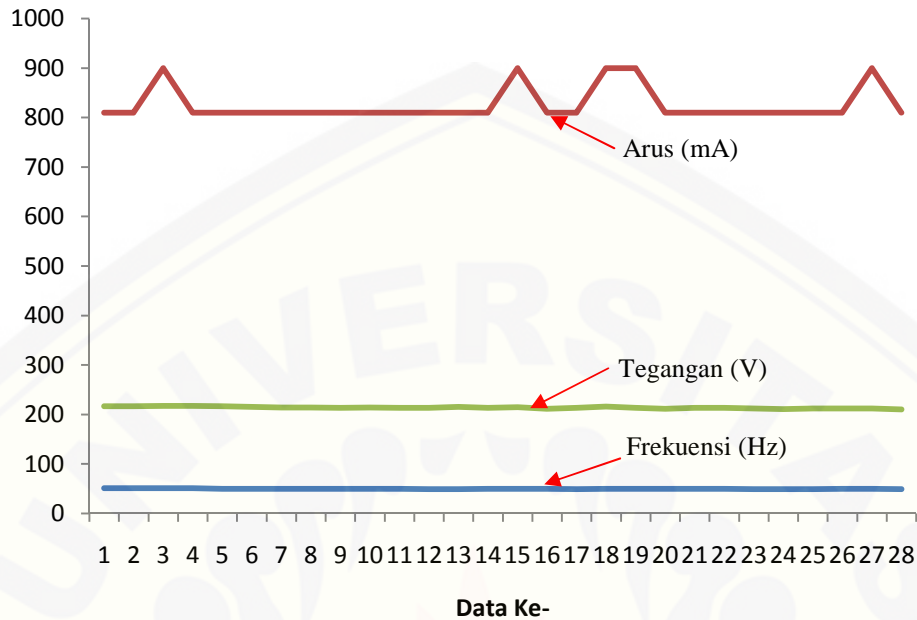
Gambar 4.13 Tampilan HMI Saat Beban 180 Watt

Dari tabel 4.7 Kemudian dijadikan bentuk grafik garis. Dilihat dari grafik garis yang dibuat, nilai pembacaan arus terlihat sedikit fluktuatif (pada garis biru) yaitu antara 810 Miliampere dan 900 Miliampere. Namun pada nilai pembacaan frekuensi dan tegangan terlihat bahwa adanya perubahan pembacaan pada kedua nilai tersebut. Namun selisih nilai pembacaan antar data pada kedua parameter tersebut tidak terlalu jauh sehingga masih bisa dikatakan bahwa sistem SCADA berjalan dengan baik pada pengujian beban 180 Watt. Gambar 4.14 menunjukkan grafik dari tabel *database* pengujian beban 180 Watt.

Tabel 4.7 Data dari *Database* Saat Pengujian Beban 180 Watt

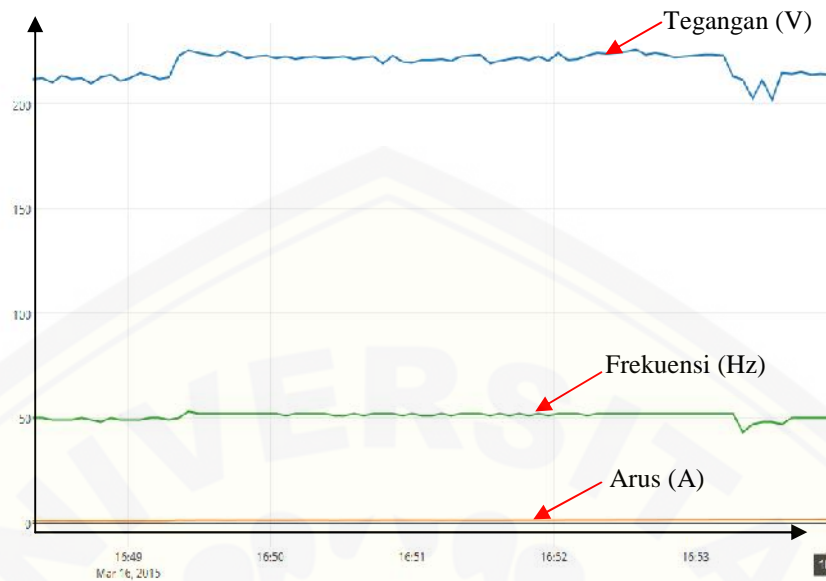
No	Waktu	Frekuensi	Arus	Tegangan
1	2015-03-16 16:46:36.407123	51 Hz	0.81 A	216.67 V
2	2015-03-16 16:46:40.528782	51 Hz	0.81 A	216.67 V
3	2015-03-16 16:46:44.649090	51 Hz	0.90 A	217.08 V
4	2015-03-16 16:46:48.769335	51 Hz	0.81 A	217.08 V
5	2015-03-16 16:46:52.890607	50 Hz	0.81 A	216.67 V
6	2015-03-16 16:46:57.010719	50 Hz	0.81 A	215.00 V
7	2015-03-16 16:47:01.135289	50 Hz	0.81 A	213.75 V
8	2015-03-16 16:47:05.256712	50 Hz	0.81 A	213.75 V
9	2015-03-16 16:47:09.380426	50 Hz	0.81 A	212.92 V
10	2015-03-16 16:47:13.502182	50 Hz	0.81 A	213.75 V
11	2015-03-16 16:47:17.625109	50 Hz	0.81 A	213.33 V
12	2015-03-16 16:47:21.750768	49 Hz	0.81 A	212.92 V
13	2015-03-16 16:47:25.871595	49 Hz	0.81 A	215.00 V
14	2015-03-16 16:47:30.000058	50 Hz	0.81 A	213.33 V
15	2015-03-16 16:47:34.116971	50 Hz	0.90 A	214.58 V
16	2015-03-16 16:47:38.240830	50 Hz	0.81 A	211.25 V
17	2015-03-16 16:47:42.365297	49 Hz	0.81 A	213.33 V
18	2015-03-16 16:47:46.486051	50 Hz	0.90 A	215.83 V
19	2015-03-16 16:47:50.609638	50 Hz	0.90 A	212.92 V
20	2015-03-16 16:47:54.731473	50 Hz	0.81 A	211.25 V
21	2015-03-16 16:47:58.854828	50 Hz	0.81 A	213.33 V
22	2015-03-16 16:48:02.976147	50 Hz	0.90 A	213.33 V
23	2015-03-16 16:48:07.100722	49 Hz	0.90 A	211.67 V
24	2015-03-16 16:48:11.224993	49 Hz	0.81 A	210.42 V
25	2015-03-16 16:48:15.350433	49 Hz	0.81 A	212.08 V
26	2015-03-16 16:48:19.474207	50 Hz	0.81 A	211.67 V
27	2015-03-16 16:48:23.594448	50 Hz	0.90 A	212.08 V
28	2015-03-16 16:48:27.719663	49 Hz	0.81 A	210.00 V

Grafik Pengujian Beban 180W

Gambar 4.14 Grafik *Database* Pengujian Beban 180 Watt

4.2.5 Pengujian beban 255 Watt

Pada pengujian beban 225 Watt dilakukan selama 2 menit. Untuk mengaktifkan beban 225 Watt tekan tombol “lampu 1 on”, “lampu 2 on”, “lampu 3 on” dan “lampu 4 on” sehingga lampu 1, lampu 2, lampu 3 dan lampu 4 akan menyala. Kemudian paantau kondisi baik tegangan, arus maupun frekuensi pada HMI (*Human Machine Interface*) selama durasi yang ditentukan. Berikut tampilan HMI saat lampu ketiga lampu diaktifkan secara bersamaan. Dapat dilihat pada gambar 4.15 saat lampu 1, lampu 2, lampu 3 dan lampu 4 diaktifkan pada garis yang berwarna jingga menunjukkan bahwa arus yang mengalir saat beban lampu 255 Watt sebesar 1.17 A, garis hijau menunjukkan nilai frekuensi sebesar 50 Hz dan pada garis biru menunjukkan besar nilai tegangan sebesar 213.75 V. Untuk data selengkapnya dapat dilihat pada tabel 4.8, dimana data tabel tersebut berasal dari *database* sistem selama 2 menit.



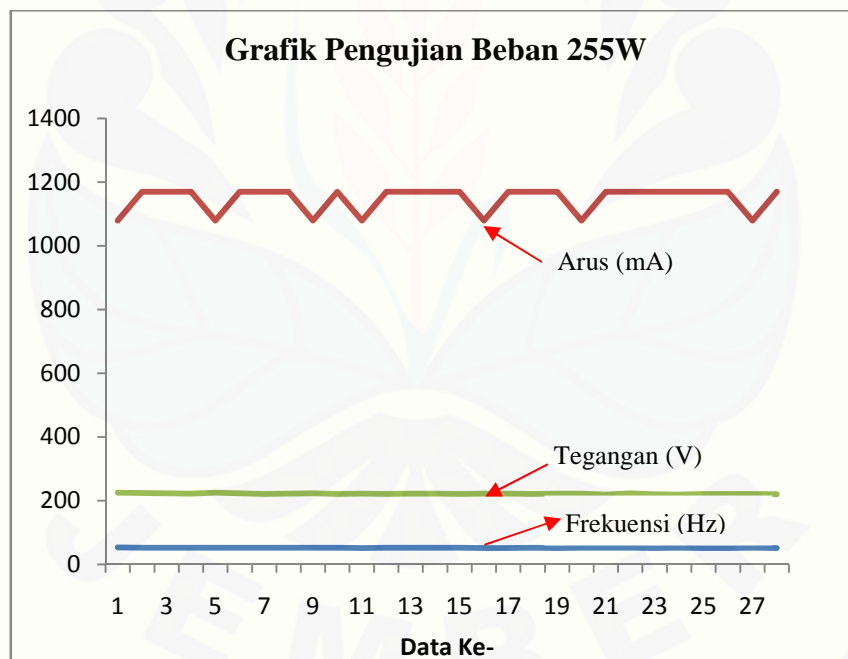
Gambar 4.15 Tampilan HMI Saat Beban 255 Watt

Tabel 4.8 Data dari *Database* Saat Pengujian Beban 255 Watt

No	Waktu	Frekuensi	Arus	Tegangan
1	2015-03-16 16:49:25.435522	53 Hz	1.08 A	225.42 V
2	2015-03-16 16:49:29.551143	52 Hz	1.17 A	224.17 V
3	2015-03-16 16:49:33.664239	52 Hz	1.17 A	223.33 V
4	2015-03-16 16:49:37.780219	52 Hz	1.17 A	222.50 V
5	2015-03-16 16:49:41.893001	52 Hz	1.08 A	225.00 V
6	2015-03-16 16:49:46.009667	52 Hz	1.17 A	223.75 V
7	2015-03-16 16:49:50.125832	52 Hz	1.17 A	221.67 V
8	2015-03-16 16:49:54.240170	52 Hz	1.17 A	222.50 V
9	2015-03-16 16:49:58.355034	52 Hz	1.08 A	222.92 V
10	2015-03-16 16:50:02.471016	52 Hz	1.17 A	221.67 V
11	2015-03-16 16:50:06.587507	51 Hz	1.08 A	222.50 V
12	2015-03-16 16:50:10.708704	52 Hz	1.17 A	221.25 V
13	2015-03-16 16:50:14.819936	52 Hz	1.17 A	222.08 V
14	2015-03-16 16:50:18.933364	52 Hz	1.17 A	222.50 V
15	2015-03-16 16:50:23.049466	52 Hz	1.17 A	221.67 V
16	2015-03-16 16:50:27.166175	51 Hz	1.08 A	222.08 V
17	2015-03-16 16:50:31.282892	51 Hz	1.17 A	222.50 V
18	2015-03-16 16:50:35.399258	52 Hz	1.17 A	221.25 V
19	2015-03-16 16:50:39.515744	51 Hz	1.17 A	222.08 V
20	2015-03-16 16:50:43.631548	52 Hz	1.08 A	222.50 V
21	2015-03-16 16:50:47.748138	52 Hz	1.17 A	219.17 V

22	2015-03-16 16:50:51.863836	52 Hz	1.17 A	222.92 V
23	2015-03-16 16:50:55.980308	51 Hz	1.17 A	220.00 V
24	2015-03-16 16:51:00.096827	52 Hz	1.17 A	219.58 V
25	2015-03-16 16:51:04.214368	51 Hz	1.17 A	220.83 V
26	2015-03-16 16:51:08.329072	51 Hz	1.17 A	220.83 V
27	2015-03-16 16:51:12.448896	52 Hz	1.08 A	221.25 V
28	2015-03-16 16:51:16.562736	51 Hz	1.17 A	220.42 V

Dari tabel 4.8 Kemudian dijadikan bentuk grafik garis. Dilihat dari grafik garis yang dibuat, nilai pembacaan arus terlihat sedikit fluktuatif (pada garis biru) yaitu antara 1080 Miliampere dan 1170 Miliampere. Namun pada nilai pembacaan frekuensi dan tegangan terlihat bahwa adanya perubahan pembacaan pada kedua nilai tersebut. Namun selisih nilai pembacaan antar data pada kedua parameter tersebut tidak terlalu jauh sehingga masih bisa dikatakan bahwa sistem SCADA berjalan dengan baik pada pengujian beban 225 Watt. Gambar 4.16 menunjukkan grafik dari tabel *database* pengujian beban 225 Watt.

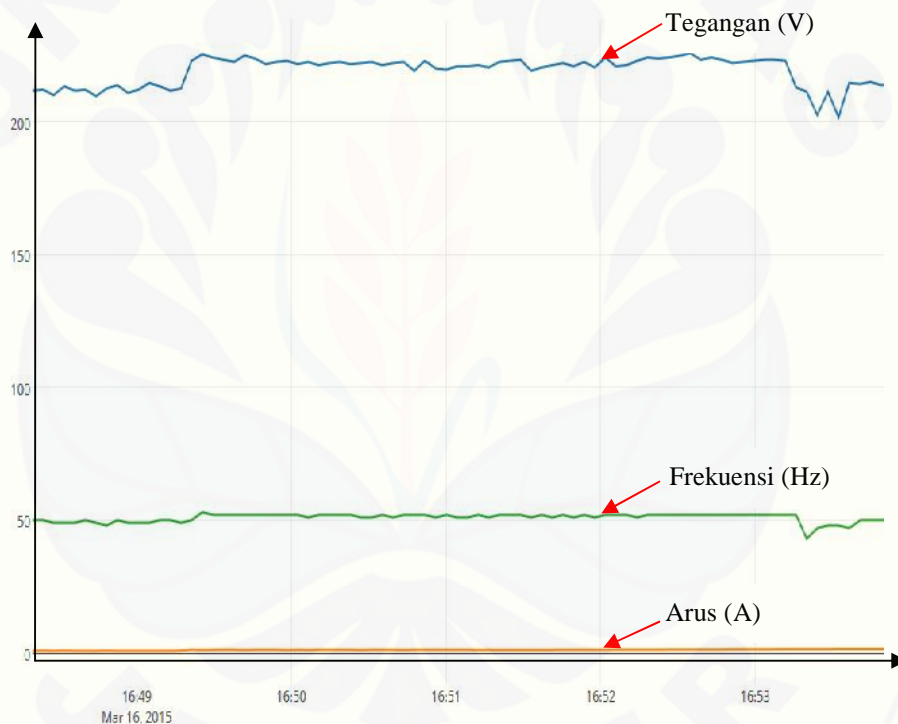


Gambar 4.16 Grafik *Database* Pengujian Beban 255 Watt

4.2.6 Pengujian beban 315 Watt

Pada pengujian beban 315 Watt dilakukan selama 2 menit. untuk mengaktifkan beban 315 Watt tekan tombol “lampu 1 on”, “lampu 2 on” dan

“lampu 3 on” sehingga lampu 1, lampu 2, lampu 3, lampu 4 dan lampu 5 akan menyala. Kemudian paantau kondisi baik tegangan, arus maupun frekuensi pada HMI (*Human Machine Interface*) selama durasi yang ditentukan. Berikut tampilan HMI saat lampu ketiga diaktifkan secara bersamaan. Dapat dilihat pada gambar 4.17 saat lampu 1, lampu 2, lampu 3, lampu 4 dan lampu 5 diaktifkan pada garis yang berwarna jingga menunjukkan bahwa arus yang mengalir saat beban lampu 315 Watt sebesar 1.44 A, garis hijau menunjukkan nilai frekuensi sebesar 51 Hz dan pada garis biru menunjukkan besar nilai tegangan sebesar 223.83 V. Untuk data selengkapnya dapat dilihat pada tabel 4.9, dimana data tabel tersebut berasal dari *database* sistem selama 2 menit.



Gambar 4.17 Tampilan HMI Saat Beban 315 Watt

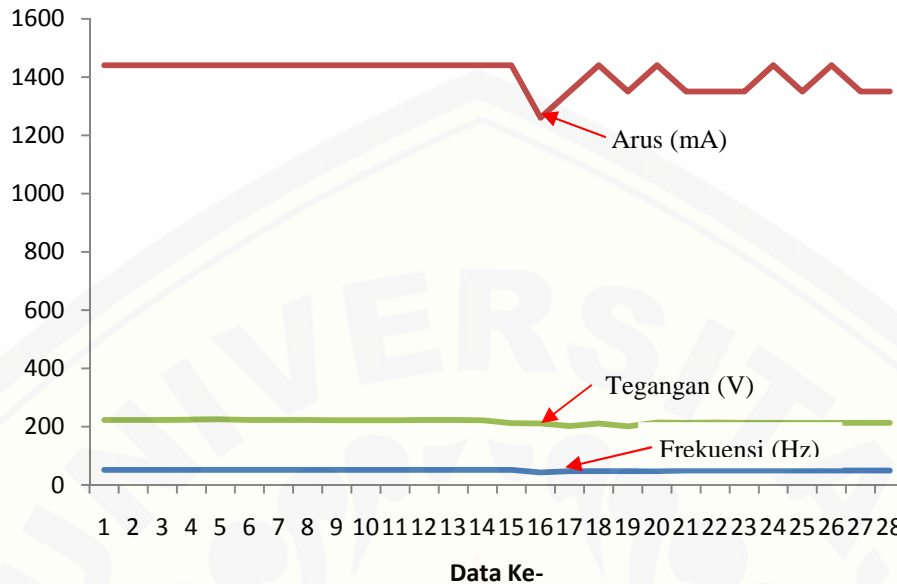
Dari tabel 4.9 kemudian dijadikan bentuk grafik garis. Dilihat dari grafik garis yang dibuat, nilai pembacaan arus terlihat sedikit fluktuatif (pada garis biru) yaitu antara 1350 Miliampere dan 1440 Miliampere. Namun pada nilai pembacaan frekuensi dan tegangan terlihat bahwa adanya perubahan pembacaan pada kedua nilai tersebut. Namun selisih nilai pembacaan antar data pada kedua parameter tersebut tidak terlalu jauh

sehingga masih bisa dikatakan bahwa sistem SCADA berjalan dengan baik pada pengujian beban 315 Watt. Gambar 4.18 menunjukkan grafik dari tabel *database* pengujian beban 315 Watt.

Tabel 4.9 Data dari *Database* Saat Pengujian Beban 315 Watt

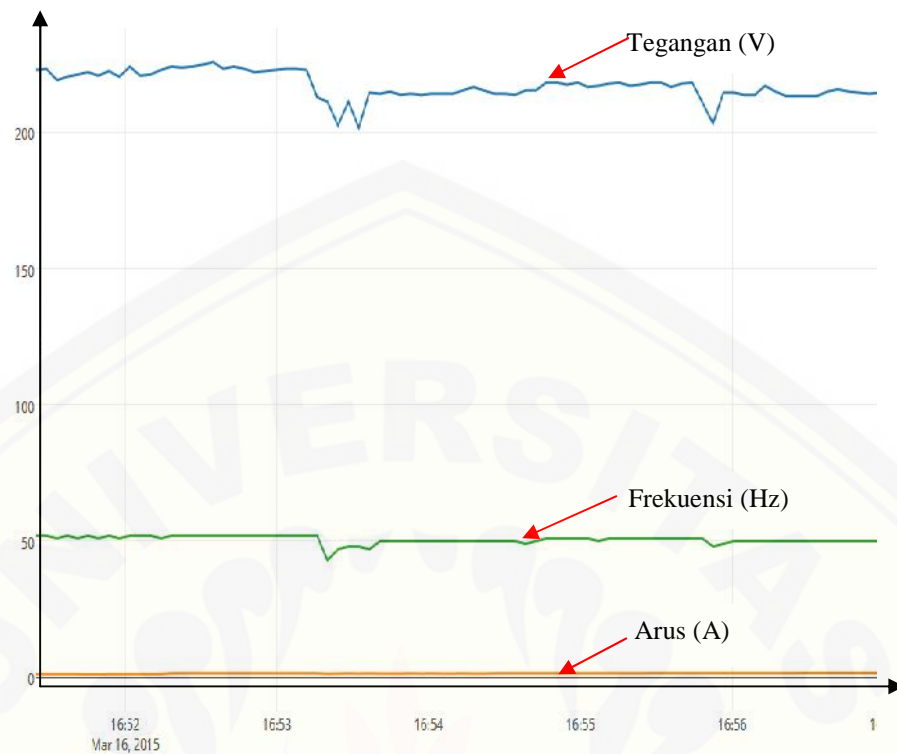
No	Waktu	Frekuensi	Arus	Tegangan
1	2015-03-16 16:52:18.296561	52 Hz	1.44 A	224.17 V
2	2015-03-16 16:52:22.409721	52 Hz	1.44 A	223.75 V
3	2015-03-16 16:52:26.525212	52 Hz	1.44 A	224.17 V
4	2015-03-16 16:52:31.524527	52 Hz	1.44 A	225.00 V
5	2015-03-16 16:52:34.751656	52 Hz	1.44 A	225.83 V
6	2015-03-16 16:52:38.866162	52 Hz	1.44 A	223.33 V
7	2015-03-16 16:52:42.979378	52 Hz	1.44 A	224.17 V
8	2015-03-16 16:52:47.091136	52 Hz	1.44 A	223.33 V
9	2015-03-16 16:52:51.207166	52 Hz	1.44 A	222.08 V
10	2015-03-16 16:52:55.324416	52 Hz	1.44 A	222.50 V
11	2015-03-16 16:52:59.436179	52 Hz	1.44 A	222.92 V
12	2015-03-16 16:53:03.549361	52 Hz	1.44 A	223.33 V
13	2015-03-16 16:53:07.664533	52 Hz	1.44 A	223.33 V
14	2015-03-16 16:53:11.778175	52 Hz	1.44 A	222.92 V
15	2015-03-16 16:53:15.897390	52 Hz	1.44 A	212.92 V
16	2015-03-16 16:53:20.023651	43 Hz	1.26 A	211.25 V
17	2015-03-16 16:53:24.150732	47 Hz	1.35 A	202.50 V
18	2015-03-16 16:53:28.272747	48 Hz	1.44 A	211.25 V
19	2015-03-16 16:53:32.397351	48 Hz	1.35 A	201.67 V
20	2015-03-16 16:53:36.521782	47 Hz	1.44 A	214.58 V
21	2015-03-16 16:53:40.873133	50 Hz	1.35 A	214.17 V
22	2015-03-16 16:53:44.758415	50 Hz	1.35 A	215.00 V
23	2015-03-16 16:53:48.878524	50 Hz	1.35 A	213.75 V
24	2015-03-16 16:53:52.995540	50 Hz	1.44 A	214.17 V
25	2015-03-16 16:53:57.115217	50 Hz	1.35 A	213.75 V
26	2015-03-16 16:54:01.233529	50 Hz	1.44 A	214.17 V
27	2015-03-16 16:54:05.352778	50 Hz	1.35 A	214.17 V
28	2015-03-16 16:54:09.473834	50 Hz	1.35 A	214.17 V

Grafik Pengujian Beban 315W

Gambar 4.18 Grafik *Database* Pengujian Beban 315 Watt

4.2.7 Pengujian beban 375 Watt

Pada pengujian beban 375 Watt dilakukan selama 2 menit. Untuk mengaktifkan beban 375 Watt tekan tombol “lampu 1 on”, “lampu 2 on”, “lampu 3 on”, “lampu 4 on”, “lampu 5 on dan “lampu 6 on” sehingga lampu 1, lampu 2, lampu 3, lampu 4, lampu 5 dan lampu 6 akan menyala. Kemudian pantau kondisi baik tegangan, arus maupun frekuensi pada HMI (*Human Machine Interface*) selama durasi yang ditentukan. Berikut tampilan HMI saat lampu ketiga lampu diaktifkan secara bersamaan. Dapat dilihat pada gambar 4.19 saat lampu 1, lampu 2, lampu 3, lampu 4, lampu 5 dan lampu 6 diaktifkan pada garis yang berwarna jingga menunjukkan bahwa arus yang mengalir saat beban lampu 375 Watt sebesar 1.71 A, garis hijau menunjukkan nilai frekuensi sebesar 50 Hz dan pada garis biru menunjukkan besar nilai tegangan sebesar 213.33 V. Untuk data selengkapnya dapat dilihat pada tabel 4.10, dimana data tabel tersebut berasal dari *database* sistem selama 2 menit.



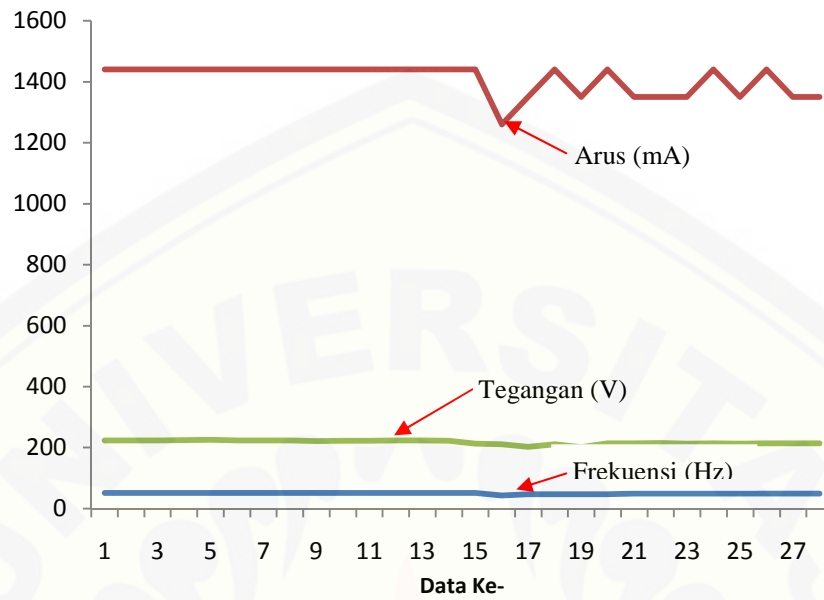
Gambar 4.19 Tampilan HMI Saat Beban 375 Watt

Dari tabel 4.10 Kemudian dijadikan bentuk grafik garis. Dilihat dari grafik garis yang dibuat, nilai pembacaan arus terlihat sedikit fluktuatif (pada garis biru) yaitu antara 1620 Miliampere dan 1710 Miliampere. Namun pada nilai pembacaan frekuensi dan tegangan terlihat bahwa adanya perubahan pembacaan pada kedua nilai tersebut. Namun selisih nilai pembacaan antar data pada kedua parameter tersebut tidak terlalu jauh sehingga masih bisa dikatakan bahwa sistem SCADA berjalan dengan baik pada pengujian beban 375 Watt. Gambar 4.20 menunjukkan grafik dari tabel *database* pengujian beban 375 Watt.

Tabel 4.10 Data dari *Database* Saat Pengujian Beban 375 Watt

No	Waktu	Frekuensi	Arus	Tegangan
1	2015-03-16 16:54:58.891017	51 Hz	1.71 A	218.33 V
2	2015-03-16 16:55:03.007585	51 Hz	1.71 A	216.67 V
3	2015-03-16 16:55:07.124526	50 Hz	1.71 A	217.08 V
4	2015-03-16 16:55:11.240525	51 Hz	1.71 A	217.92 V
5	2015-03-16 16:55:15.361515	51 Hz	1.71 A	218.33 V
6	2015-03-16 16:55:19.478007	51 Hz	1.71 A	217.08 V
7	2015-03-16 16:55:23.593969	51 Hz	1.71 A	217.50 V
8	2015-03-16 16:55:27.710612	51 Hz	1.71 A	218.33 V
9	2015-03-16 16:55:31.826657	51 Hz	1.71 A	218.33 V
10	2015-03-16 16:55:35.943525	51 Hz	1.71 A	216.67 V
11	2015-03-16 16:55:40.058942	51 Hz	1.71 A	217.92 V
12	2015-03-16 16:55:44.183254	51 Hz	1.71 A	218.33 V
13	2015-03-16 16:55:48.296714	51 Hz	1.71 A	210.83 V
14	2015-03-16 16:55:52.420905	48 Hz	1.62 A	203.33 V
15	2015-03-16 16:55:56.540536	49 Hz	1.62 A	214.58 V
16	2015-03-16 16:56:04.779210	50 Hz	1.62 A	213.75 V
17	2015-03-16 16:56:13.014208	50 Hz	1.62 A	217.08 V
18	2015-03-16 16:56:21.251055	50 Hz	1.62 A	213.33 V
19	2015-03-16 16:56:29.488848	50 Hz	1.62 A	213.33 V
20	2015-03-16 16:56:37.725973	50 Hz	1.71 A	215.00 V
21	2015-03-16 16:56:45.962157	50 Hz	1.71 A	215.00 V
22	2015-03-16 16:57:02.435489	50 Hz	1.62 A	215.42 V
23	2015-03-16 16:57:18.914358	50 Hz	1.71 A	212.92 V
24	2015-03-16 16:57:22.748358	50 Hz	1.71 A	213.68 V
25	2015-03-16 16:57:26.652483	50 Hz	1.71 A	217.98 V
26	2015-03-16 16:57:30.687454	50 Hz	1.71 A	216.54 V
27	2015-03-16 16:57:34.256981	49 Hz	1.71 A	217.35 V
28	2015-03-16 16:57:38.879557	51 Hz	1.71 A	218.34 V

Grafik Pengujian Beban 375W

Gambar 4.20 Grafik *Database* Pengujian Beban 375 Watt

BAB 5.PENUTUP

5.1 Kesimpulan

Dari hasil pembuatan, pengujian perangkat dan pembahasan dapat diambil kesimpulan sebagai berikut:

1. Respon *realtime* grafik dan *database* pada HMI (*Human Machine Interface*) sangat baik dalam menampilkan perubahan masukan pada sensor, baik pada sensor tegangan, sensor arus, maupun sensor frekuensi dengan nilai rata – rata *error %* sebesar 18 %.
2. Sistem pengontrolan beban yang dibangun dapat melakukan pengontrolan beban melalui rangkaian relay dari jarak jauh dengan menggunakan jaringan LAN dengan baik sesuai dengan yang diinginkan.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan tentang “*Monitoring Dan Pengontrolan Pembebanan Pada Generator Induksi Satu fasa Berbasis Raspberry Pi*” penulis memberikan saran berikut dengan harapan untuk penyempurnaan karya ilmiah ini dan lebih memberikan manfaat yang lebih baik dimasa mendatang:

1. Perlu dilakukan penelitian lebih teliti pada kalibrasi pembacaan sensor.
2. Perlu dilakukan pengembangan lebih lanjut pada tampilan HMI (*Human Machine Interface*) sehingga dapat lebih mudah dan nyaman dalam mengontrol dan melihat kinerja *plant*.
3. Perlu dilakukan pengembangan terhadap sistem *database* agar dalam pengambilan data yang disimpan lebih mudah.

DAFTAR PUSTAKA

- Axelson, Jan, 1999 , “*Designing RS-485 Circuit*”, CIRCUIT CELLAR.
- Axelson, Jan, 1995, “*Network for Monitoring and Control Using an RS-485 interface*”, MICROCOMPUTER JOURNAL.
- Groover, Mikell P., Zimmers, Emory W., 1984 “*CAD/CAM: Computer-Aided Design And Manufacturing*”, Prentice-Hall International, Inc.
- Hamzah A., 2011. “*Perancangan Kontrol Beban Elektronik Generator Satu Fasa Berbasis Smart Relay*”.
- Johnson, Curtis D., 1988, “*Process Control Instrumentation Technology*”, 3rd edition, John Wiley & Sons, Inc.
- Leigh, J.R., 1992, “*Applied Digital Control : Theory, Design and Implementation*”, Prentice-Hall International, Inc.
- Lunze, Jan, 1993, “*Large Scale System*”, Prentice-Hall International, Inc.
- Mahendra, Sandi, Surabaya 2001, “*Sistem Pengendalian Multiplant Dengan Menggunakan Minimum System*”, Tugas Akhir, Jurusan Teknik Elektro – FTI, ITS.
- Malik MI. dan Anistardi, 1997, “*Bereksperimen Dengan Mikrokontroler 8031*”, Elex Media Komputindo,.
- Permana I. S, 2012, “*Rancang Bangun Pengontrolan Beban Secara Elektronik Pada Pembangkit Listrik*”.
- Septian G. A, 2013, “*Rancang Bangun Kumparan Stator Generator AC Satu Fasa Dua Kutub dan Analisis Pengujian Dengan Beban*”.
- Uffenbeck, John, 1987 , “*The 8086/8088 Family : Design, Programming and Interfacing*”, Prentice-Hall International, Inc.
- Williams, R. I., 1992, “*Handbook of SCADA Systems for the Oil & Gas Industry*”
1st Edition, Elsevier Advanced Technology Limited, Oxford.

List Program Arduino

```
//-----Inisialisasi frekuensi counter-----//
```

```
int count=0;  
int frekuensi=0;
```

```
//-----Inisialisasi sensor arus-----//
```

```
int data_adc_arus=0;  
int data_arus=0;  
int rata2_arus=0;  
int hasil_arus=0;  
float arus_A=0.00;  
double total_arus=0;
```

```
//-----Inisialisasi sensor tegangan-----//
```

```
int data_adc_tegangan=0;  
int data_tegangan=0;  
int rata2_tegangan=0;  
int hasil_tegangan=0;  
int hasil_rata2=0;  
float tegangan=0.00;  
double total_tegangan=0;  
double hasil_jumlah=0;
```

```
//-----//
```

```
//-----inisialisasi iterasi-----//
```

```
int i=0;
```

```
int a=0;
//-----//

void setup(){

  Serial.begin(9600);// seting baut rate 9600
  while(!Serial){ }//Menunggu sambungan serial kalau belum ada sambungan serial
  program tidak bekerja

  pinMode(2, INPUT_PULLUP);// Deklarasi PIN DIGITAL 2 menjadi input dalam
  keadaan pull up
}

void loop(){

  //-----program frekuensi counter DIGITAL PIN 2-----//
  while((digitalRead(2)==1)&&(count<6000))
  {count++;}

  if(count<6000)
  {
    frekuensi=count/50;//kalibrasi frekuensi
    Serial.print(frekuensi);
    count=0;
  }
  else
  {
    frekuensi=0;
    Serial.print(frekuensi);
  }
}
```



```
while(digitalRead(2)==0){}
count=0;
//-----//

Serial.print(",");//split data pakai koma

//-----program sensor arus ANALOG PIN A0-----//

for(i=0;i<1000;i++)
{
  data_adc_arus = analogRead(A0);
  data_arus=abs(data_adc_arus-511);
  total_arus=total_arus+data_arus;
}
hasil_arus=total_arus/1000;
arus_A=hasil_arus/11.11;//kalibrasi arus
Serial.print(arus_A);
delay(500);
total_arus=0;
hasil_arus=0;
//-----//

Serial.print(",");//split data pake koma

//-----program sensor tegangan ANALOG PIN A1-----//
---//
for(a=0;a<10;a++)
{
  for(i=0;i<1000;i++)
  {
    data_adc_tegangan = analogRead(A1);
    data_tegangan=data_adc_tegangan;
    total_tegangan=total_tegangan+data_tegangan;
  }
}
```

```
hasil_tegangan=total_tegangan/1000;
hasil_jumlah=hasil_jumlah+hasil_tegangan;
total_tegangan=0;
hasil_tegangan=0;
}
hasil_rata2=abs((hasil_jumlah/10)-1023);
tegangan=hasil_rata2/2.20;// kalibrasi tegangan
Serial.println(tegangan);
delay(500);
total_tegangan=0;
hasil_tegangan=0;
hasil_rata2=0;
hasil_jumlah=0;
//-----//
}
```

List Program Raspberry Pi

```
#!/usr/bin/python
'''
by Ben Kenney - July 2012
This program reads data coming from the serial port and
saves that data to a text file. It expects data in the
format:
"photocell_reading,thermistor_reading"

It assumes that the Arduino shows up in /dev/ttyACM0 on the
Raspberry Pi which should happen if you're using Debian.
'''

import serial
from time import strftime
from datetime import datetime, time

ser = serial.Serial('/dev/ttyACM0',9600)

startTime = datetime.now()
try:
    while 1:
        line = ser.readline()
```

```
f, i, v = line.split(",")
now = datetime.now()
elapsedTime = now-startTime
    #print("%s,%s,%s,%s"%(now.strftime("%Y-%m-
%d %H:%M:%S"),str(elapsedSeconds), +str(f), +str(i),
+str(v)))
    print datetime.utcnow().strftime('%Y-%m-%d---
%H:%M:%S')+ ' f=' +str(f)+ ' i=' +str(i)+ ' v=' +
str(f)
    #print 'f=' +str(f)+ ' i=' +str(i)+ ' v='
+ str(f)
    h=open('tempLog.dat','a')
    #print >>h,("%s,%s,%s,%s"%(now.strftime("%Y-%m-%d
%H:%M:%S"),elapsedSeconds,f,i,v))
    print >> h, ' ' +
str(datetime.utcnow().strftime('%Y-%m-%d %H:%M:%S'))+ '
' +str(f)+ ' ' +str(i)+ ' ' + str(f)
    h.close()
except KeyboardInterrupt:
    print "\ndone"
import plotly.plotly as py
from plotly.graph_objs import Scatter, Layout, Figure
import datetime
import serial
import time

username = 'scada'
api_key = 'sjxp0rsl6u'
stream_token1 = 'zprz0mdwn2'
stream_token2 = 'pmoyi60nzm'
stream_token3 = '8n4wr0ql1v'
py.sign_in(username, api_key)

trace1 = Scatter(
    x=datetime.datetime.now(),
    y=[],
    stream=dict(
        token=stream_token1,
        maxpoints=100))
trace2 = Scatter(
    x=datetime.datetime.now(),
    y=[],
    stream=dict(
        token=stream_token2,
        maxpoints=100))
trace3 = Scatter(
    x=datetime.datetime.now(),
    y=[],
    stream=dict(
        token=stream_token3,
        maxpoints=100))
```

```
layout = Layout(
    title='SISTEM SCADA GENERATOR INDUKSI SATU FASA'
)

fig = Figure(data=[trace3,trace2,trace1], layout=layout)

print py.plot(fig, filename='Raspberry Pi Streaming Example
Values')

# temperature sensor connected channel 0 of mcp3008
#sensor_pin = 0
#readadc.initialize()
ser = serial.Serial('/dev/ttyACM0', 9600)
stream1 = py.Stream(stream_token1)
stream1.open()
stream2 = py.Stream(stream_token2)
stream2.open()
stream3 = py.Stream(stream_token3)
stream3.open()
#a = 0
#the main sensor reading loop
while True:
    #sensor_data = readadc.readadc(sensor_pin,
readadc.PINS.SPICK, readadc.PINS.SPIMOSI,
readadc.PINS.SPIMISO, readadc.PINS.SPICS)
    #stream.write({'x': i, 'y': sensor_data})
    #i += 1
    # delay between stream posts
    s = ser.readline()
    f, i, v = s.split(",")
    stream3.write({'x':datetime.datetime.now(), 'y': v })
    stream2.write({'x':datetime.datetime.now(), 'y': i })
    stream1.write({'x':datetime.datetime.now(), 'y': f })
    #a += 1
    print 'f=' +str(f)+ ' i=' +str(i)+ ' v=' + str(v)
    h=open('tempLog.dat','a')
    #print >>h,("%s,%s,%s,%s"%(now.strftime("%Y-%m-%d
%H:%M:%S"),elapsedSeconds,f,i,v))
    print >> h, ' ' + str(datetime.datetime.now())+ ' '
+str(f)+ ' ' +str(i)+ ' ' + str(v)
    h.close()
    #rint s
    time.sleep(0.25)
# Import modul webiopi
import webiopi

# Memanggil library GPIO
GPIO = webiopi.GPIO

# ----- #
```

```
# Mendefinisikan GPIO #
# ----- #

# GPIO untuk Kontrol Beban SCADA
L1=04 # Input relay 1
L2=23 # Input relay 2
L3=24 # input relay 3
L4=22 # Input relay 4
L5=17 # Input ralay 5
L6=18 # Input relay 6

# ----- #
# Membuat fungsi Saklar lampu #
# ----- #

def lampulon():
    GPIO.output(L1, GPIO.HIGH)
def lampuloff():
    GPIO.output(L1, GPIO.LOW)
def lampu2on():
    GPIO.output(L2, GPIO.HIGH)
def lampu2off():
    GPIO.output(L2, GPIO.LOW)
def lampu3on():
    GPIO.output(L3, GPIO.HIGH)
def lampu3off():
    GPIO.output(L3, GPIO.LOW)
def lampu4on():
    GPIO.output(L4, GPIO.HIGH)
def lampu4off():
    GPIO.output(L4, GPIO.LOW)
def lampu5on():
    GPIO.output(L5, GPIO.HIGH)
def lampu5off():
    GPIO.output(L5, GPIO.LOW)
def lampu6on():
    GPIO.output(L6, GPIO.HIGH)
def lampu6off():
    GPIO.output(L6, GPIO.LOW)

# ----- #
# Definisi Macro untuk JavaScript #
# ----- #

def lampu1_on():
    lampulon()
def lampu1_off():
    lampuloff()
def lampu2_on():
    lampu2on()
def lampu2_off():
    lampu2off()
```



```
def lampu3_on():
    lampu3on()
def lampu3_off():
    lampu3off()
def lampu4_on():
    lampu4on()
def lampu4_off():
    lampu4off()
def lampu5_on():
    lampu5on()
def lampu5_off():
    lampu5off()
def lampu6_on():
    lampu6on()
def lampu6_off():
    lampu6off()
# ----- #
# Inisialisasi #
# ----- #

# Setup GPIO

GPIO.setFunction(L1, GPIO.OUT)
GPIO.setFunction(L2, GPIO.OUT)
GPIO.setFunction(L3, GPIO.OUT)
GPIO.setFunction(L4, GPIO.OUT)
GPIO.setFunction(L5, GPIO.OUT)
GPIO.setFunction(L6, GPIO.OUT)

# ----- #
# Membuat Web Server #
# ----- #

# Menempatkan web server pada port 8090, dan membuat ID dan
password
server = webiopi.Server(port=9090, login="parto",
password="ganteng")

# Mendaftarkan Macro untuk dipanggil pada javascript di HTML

server.addMacro(lampu1_on)
server.addMacro(lampu1_off)
server.addMacro(lampu2_on)
server.addMacro(lampu2_off)
server.addMacro(lampu3_on)
server.addMacro(lampu3_off)
server.addMacro(lampu4_on)
server.addMacro(lampu4_off)
server.addMacro(lampu5_on)
server.addMacro(lampu5_off)
```

```
server.addMacro(lampu6_on)
server.addMacro(lampu6_off)

# ----- #
# Me-Loop Program Web Server #
# ----- #

# Menjalankan Loop sampai CTRL+C ditekan atau Raspberry
direstart
webiopi.runLoop()

# ----- #
# Mematikan Program Web Server #
# ----- #

# Stop Web server
server.stop()

# Mengatur Ulang fungsi GPIO
GPIO.setFunction(L1, GPIO.IN)
GPIO.setFunction(L2, GPIO.IN)
GPIO.setFunction(L3, GPIO.IN)
GPIO.setFunction(L4, GPIO.IN)
GPIO.setFunction(L5, GPIO.IN)
GPIO.setFunction(L6, GPIO.IN)

# recieving data sensor from arduino
import serial
from time import strftime
from datetime import datetime, time

ser = serial.Serial('/dev/ttyACM0',9600)

startTime = datetime.now()
try:
    while 1:
        line = ser.readline()
        f, i, v = line.split(",")
        now = datetime.now()
        elapsedTime = now-startTime
        elapsedSeconds =
        (elapsedTime.microseconds+(elapsedTime.days*24*3600+elapsedT
ime.seconds)*10**6)/10**6
        #print("%s,%s,%s,%s"%(now.strftime("%Y-%m-%d
%H:%M:%S"),str(elapsedSeconds), +str(f), +str(i), +str(v)))
        #print datetime.utcnow().strftime('%Y-%m-%d---
%H:%M:%S')+ ' f=' +str(f)+ ' i=' +str(i)+ ' v=' +
str(f)
        #print 'f=' +str(f)+ ' i=' +str(i)+ ' v='
+ str(v)
        h=open('tempLog.dat','a')
```

```
        #print >>h, ("%s,%s,%s,%s"%(now.strftime("%Y-%m-%d
%H:%M:%S"), elapsedSeconds, f, i, v))
        print >> h, ' ' +
str(datetime.utcnow().strftime('%Y-%m-%d %H:%M:%S'))+ '
    ' +str(f)+ '    ' +str(i)+ '    ' + str(v)
        h.close()
except KeyboardInterrupt:
    print "\ndone"
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd"> <html> <head> <h1
align="center"> LABORATORIUM SISTEM TENAGA </h1>
    <meta http-equiv="Content-Type"
content="text/html; charset=UTF-8">
    <meta name="viewport" content="height=device-height,
width=420, user-scalable=no" />
    <title>SCADA</title>
    <script type="text/javascript"
src="/webiopi.js"></script>
    <script type="text/javascript">
function init() {
    var button;

        button = webiopi().createButton("bt_1", "LAMPU 1
ON", lampu1_on, "stop");
        $("#up").append(button);
        button = webiopi().createButton("bt_2", "LAMPU 1
OFF", lampu1_off, "stop");
        $("#up").append(button);
        button = webiopi().createButton("bt_3", "LAMPU 2
ON", lampu2_on, "stop");
        $("#middle").append(button);
        button = webiopi().createButton("bt_4", "LAMPU 2
OFF", lampu2_off, "stop");
        $("#middle").append(button);
        button = webiopi().createButton("bt_5", "LAMPU 3
ON", lampu3_on, "stop");
        $("#down1").append(button);
        button = webiopi().createButton("bt_6", "LAMPU 3
OFF", lampu3_off, "stop");
        $("#down1").append(button);
        button = webiopi().createButton("bt_7", "LAMPU 4
ON", lampu4_on, "stop");
        $("#down2").append(button);
        button = webiopi().createButton("bt_8", "LAMPU 4
OFF", lampu4_off, "stop");
        $("#down2").append(button);
        button = webiopi().createButton("bt_9", "LAMPU 5
ON", lampu5_on, "stop");
        $("#down3").append(button);
```

```
        button = webiopi().createButton("bt_10", "LAMPU 5  
OFF", lampu5_off, "stop");  
        $("#down3").append(button);  
        button = webiopi().createButton("bt_11", "LAMPU 6  
ON", lampu6_on, "stop");  
        $("#down4").append(button);  
        button = webiopi().createButton("bt_12", "LAMPU 6  
OFF", lampu6_off, "stop");  
        $("#down4").append(button);  
    }  
  
    function lampu1_on() {  
        webiopi().callMacro("lampu1_on");  
    }  
    function lampu1_off() {  
        webiopi().callMacro("lampu1_off");  
    }  
    function lampu2_on() {  
        webiopi().callMacro("lampu2_on");  
    }  
    function lampu2_off() {  
        webiopi().callMacro("lampu2_off");  
    }  
    function lampu3_on() {  
        webiopi().callMacro("lampu3_on");  
    }  
    function lampu3_off() {  
        webiopi().callMacro("lampu3_off");  
    }  
    function lampu4_on() {  
        webiopi().callMacro("lampu4_on");  
    }  
    function lampu4_off() {  
        webiopi().callMacro("lampu4_off");  
    }  
    function lampu5_on() {  
        webiopi().callMacro("lampu5_on");  
    }  
    function lampu5_off() {  
        webiopi().callMacro("lampu5_off");  
    }  
    function lampu6_on() {  
        webiopi().callMacro("lampu6_on");  
    }  
    function lampu6_off() {  
        webiopi().callMacro("lampu6_off");  
    }  
    webiopi().ready(init);  
  
</script>  
<style type="text/css">
```

```
        button {
            margin: 5px 5px 5px 5px;
            width: 80px;
            height: 70px;
            font-size: 8pt;
            font-weight: bold;
            color: black;
        }
    <html>
    <head></head></style><body>
    <body background="elektro.jpg">
    <div id="content" align="left">
        <a href="https://plot.ly/~scada/10/"
target="_blank" title="SISTEM SCADA GENERATOR INDUKSI SATU
FASA" style="display: block; text-align: center;"></a>
        <script data-plotly="scada:10"
src="https://plot.ly/embed.js" async></script>
        <p> </p>
        <p> </p>
        <p> </p>
        <div id="up"></div>
        <div id="middle"></div>
        <div id="down1"></div>
        <div id="down2"></div>
        <div id="down3"></div>
        <div id="down4"></div>
    </div> </body> </html>
```