



**IMPLEMENTASI SISTEM SCADA PADA GENERATOR DC
PENGUATAN TERPISAH MENGGUNAKAN ARDUINO MEGA 2560**

SKRIPSI

Oleh:
Ibrahim Saiful Millah
NIM 131910201103

**JURUSAN TEKNIK ELEKTRO STRATA 1
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2017**



**IMPLEMENTASI SISTEM SCADA PADA GENERATOR DC
PENGUATAN TERPISAH MENGGUNAKAN ARDUINO MEGA 2560**

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Teknik Elektro (S1)
dan mencapai gelar Sarjana Teknik

Oleh:
Ibrahim Saiful Millah
NIM 131910201103

**JURUSAN TEKNIK ELEKTRO STRATA 1
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2017**

PERSEMBAHAN

Dengan ini saya persembahkan skripsi kepada:

1. Allah SWT atas kemudahan, rahmat dan berkah yang tiada habisnya;
2. Rasulullah SAW, yang menjadi panutan di dunia maupun akhirat;
3. Ibunda Siti Khotijah dan Ayahanda Ahmad Baihaki yang senantiasa memberikan doa, dukungan, bimbingan, dan kasih sayang tiada henti;
4. Kakakku dr. Siti Nur Rohmah, dan Shofiatul Fajariyah, S.farm. Apt yang dengan tulus memberikan doa dan dukungan dalam setiap langkah Saudaranya;
5. RB. Moch. Gozali, ST., MT dan Dr. Triwahju Hardianto, ST., MT yang telah meluangkan waktunya untuk membimbing dalam menyelesaikan skripsi ini;
6. Almamater Fakultas Teknik Universitas Jember.

MOTTO

Allah akan meninggikan orang-orang yang beriman diantara kamu dan orang-orang yang berilmu beberapa derajat. Dan Allah maha mengetahui apa yang kamu kerjakan.

(QS. Al-Mujadalah: 11)

“Barangsiapa yang mempermudah urusan orang yang mengalami kesulitan, maka Allah akan mempermudah urusannya di dunia dan di akhirat”

(HR. Ibnu Majah).

“To be a good professional engineer, always start to study late for exams. Because it teaches you how to manage time and tackle emergencies.”

(Bill Gates)

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama: Ibrahim Saiful Millah

NIM : 131910201103

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul ” Implementasi Sistem SCADA Pada Generator DC Penguatan Terpisah Menggunakan Arduino Mega 2560” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab penuh 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 pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 05 Oktober 2017

Yang menyatakan,

Ibrahim Saiful Millah

NIM 131910201103

SKRIPSI

**IMPLEMENTASI SISTEM SCADA PADA GENERATOR DC
PENGUATAN TERPISAH MENGGUNAKAN ARDUINO MEGA 2560**

Ibrahim Saiful Millah
NIM 131910201103

Pembimbing

Dosen Pembimbing Utama : RB. Moch. Gozali, ST., M.T.

Dosen Pembimbing Anggota : Dr. Triwahju Hardianto, S.T., MT.

PENGESAHAN

Skripsi berjudul "Implementasi Sistem SCADA Pada Generator DC Penguatan Terpisah Menggunakan Arduino Mega 2560" Telah diuji dan disahkan pada :

Hari, Tanggal : Kamis, 05 Oktober 2017

Tempat : Fakultas Teknik Universitas Jember

Tim Penguji

Ketua,

RB. Moch. Gozali S.T., M.T.
NIP 196906081999031002

Anggota II,

Samsul Bachri M, S.T., M.MT.
NIP 19640317 199802 1 001

Anggota I,

Dr. Triwahju Hardianto S.T., M.T.
NIP 19700826 199702 1 002

Anggota III,

Dr. Azmi Saleh, S.T., M.T.
NIP 19710614 199702 1 001

Mengesahkan

Dekan Fakultas Teknik
Universitas Jember,

Dr.Ir. Entin Hidayah M.U.M
NIP 19661215 199503 2 001

RINGKASAN

Implementasi Scada Pada Generator DC Penguatan Terpisah Menggunakan Arduino Mega 2560; Ibrahim Saiful Millah, 131910201103; 2017; 116 halaman; Fakultas Teknik Universitas Jember

Permasalahan yang sering terjadi pada kestabilan tegangan generator DC adalah saat dibebani. Maka *output* generator DC akan menimbulkan efek *drop* tegangan. *Drop* tegangan ini diatasi dengan SCADA (*supervisory control and data acquisition*). SCADA merupakan suatu sistem yang dapat memonitor dan mengontrol suatu peralatan atau sistem secara *real time*. SCADA digunakan untuk mengontrol output generator dengan kontrol *fuzzy* untuk menstabilkan tegangan *output*. Sehingga tujuan penelitian ini untuk menerapkan sistem SCADA pada Arduino dengan generator DC penguatan terpisah sehingga diperoleh kestabilan *output* tegangan generator DC dan juga menerapkan kontrol *fuzzy logic* sehingga diperoleh kestabilan *output* tegangan generator DC terhadap variasi beban yang diberikan. Manfaat dari penelitian ini ialah memudahkan dalam pengendalian dan pengecekan kondisi pembangkit listrik dengan generator DC penguatan terpisah dan mengatasi *drop* tegangan dari generator DC penguatan terpisah ketika diberikan beban. SCADA ini sendiri dibangun dengan merancang *buck converter* yang digunakan untuk menstabilkan tegangan dari *feedback* tegangan *output* generator menggunakan kontrol *fuzzy*. Tahapan selanjutnya merancang dan memonitoring tegangan *premotor*, tegangan input *exciter*, tegangan *output* generator dan arus *output* generator akan disimpan pada database *master terminal unit*. Hasil pengujian generator tanpa kontrol di dapatkan drop tegangan ketika tanpa beban tegangannya sebesar 100 V, ketika dibebani 6,006 W sebesar 95.7 V. Sementara itu ketika beban semakin naik seperti 8,7375 W maka tegangan *output* menjadi 94.8 V, pada pembebanan 15,778 W tegangan *output* sebesar 90.5 V, pada pembebanan 17,59167 W tegangan *output* sebesar 90.2 V dan pada pembebanan 25,04118 W terjadi *drop* tegangan yang sangat besar yakni tegangan *output* menjadi sebesar 86.1 V. ketika diberi *feedback* kontrol *fuzzy* dengan *setpoint* 100 di dapatkan tegangan *output* tanpa beban 100 V dan berbeban mulai 6 W-25,04 W sebesar 99 V. artinya sistem SCADA ini mampu mengatasi *drop* tegangan yang terjadi.

PRAKATA

Segala puji dan syukur kehadiran Allah SWT atas segala anugrah dan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul **“Implementasi Sistem Scada Pada Generator DC Penguatan Terpisah Arduino Mega2560”**.

Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Fakultas Teknik Universitas Jember. Penyusunan skripsi ini tidak lepas dari bantuan dan bimbingan berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terimakasih kepada:

1. Allah SWT atas berkat rahmat-Nya saya dapat menyelesaikan skripsi ini;
2. RB. Moch. Gozali, ST., MT dan Dr. Triwahju Hardianto, ST., MT selaku Dosen Pembimbing yang telah meluangkan waktu di tengah kesibukannya untuk memberikan bimbingan, saran, motivasi, sehingga skripsi ini dapat terselesaikan;
3. Samsul Bachri M., ST.,M.MT. dan Dr. Azmi Saleh, S.T., M.T. selaku Dosen Penguji yang telah memberikan masukan dan saran yang membangun demi kesempurnaan skripsi ini;
4. Instruktur Schneider Elektrik PLC-SCADA Cahyo Tri Wibowo, S.T. yang telah memberikan inspirasi, saran, dan motivasi dalam penyusunan skripsi;
5. Kedua orang tuaku tercinta, ibunda Siti Khotijah dan ayahanda Ahmad Baihaki yang tak kenal lelah mendoakan, memberi dukungan, perhatian, serta kasih sayang yang teramat tulus selama ini;
6. Kakakku tersayang yang selalu saya rindukan di kota perantauan ini; dr. Siti Nur Rohmah, dan Shofiatul Fajariyah, S.farm.;
7. Teman-teman Alumni SMAN 1 Pamekasan angkatan 2010 (SSS 13) yang tidak bisa disebutkan satu persatu, yang selalu menghibur, memberi saran dan motivasi dalam penulisan skripsi ini;
8. Keluarga besar Laboratorium Konversi Energi, terutama Sugianto, A.Md, Andi Setiawan, S.T.,M.T. dan Ir. Widyono Hadi, M.T. yang selalu memberi saran dan solusi disetiap permasalahan yang ada;

9. Nur Sita Dewi, S.Kg yang selalu direpotkan, selalu memberi perhatian, motivasi, saran dan solusi dalam penyusunan skripsi;
10. “Nak-Kanak KKN 99” yang mewarnai perjalanan penyusunan skripsi;
11. Sahabat seperjuangan INTEL 2013 yang selalu menemani, menghibur dan memberikan motivasi;
12. Semua pihak yang turut terlibat baik secara langsung maupun tidak langsung yang tidak dapat saya sebutkan satu persatu, terimakasih untuk bantuan dan dukungan dalam penyelesaian skripsi ini.

Penulis berharap semoga skripsi ini dapat bermanfaat bagi semua pihak yang terkait dengan hasil penelitian dari penelitian skripsi ini.

Jember, 05 Oktober 2017

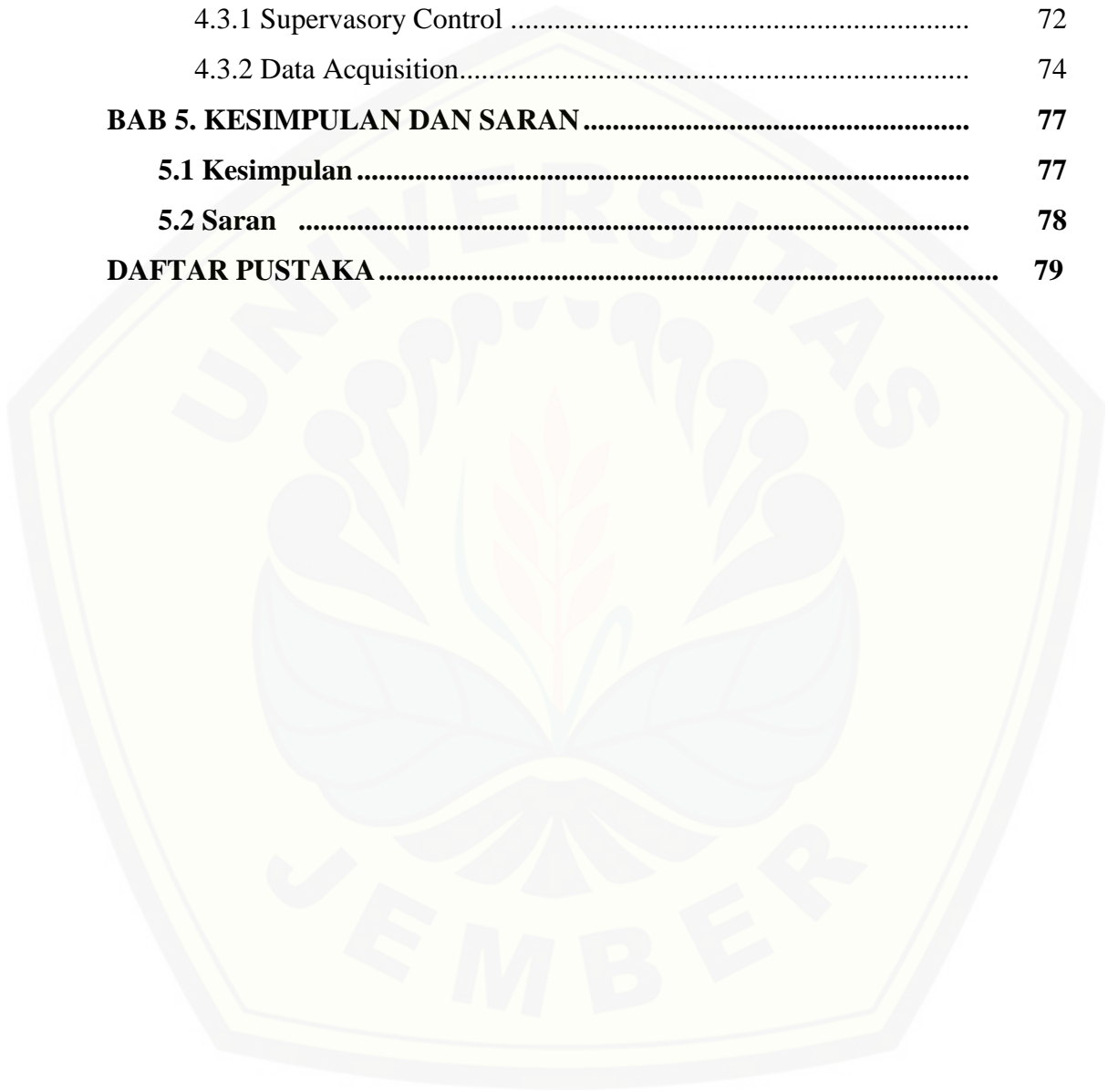
Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	ii
PERSEMBAHAN.....	iii
MOTTO	iv
LEMBAR PERNYATAAN	v
HALAMAN PEMBIMBING	vi
HALAMAN PEGESAHAN.....	vii
RINGKASAN	viii
PRAKATA	ix
DAFTAR ISI.....	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN	xvii
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan	4
1.5 Manfaat	4
BAB 2. TINJAUAN PUSTAKA.....	5
2.1 SCADA (<i>Supervisory Control And Data Acquisition</i>)	5
2.1.1 MTU	5
2.2.1 RTU.....	6
2.2 Arduino Mega 2560	7
2.3 Aduino Uno	8
2.4 Sensor	9
2.4.1 Sensor Tegangan	10
2.4.2 Sensor Arus	10
2.5 Akuator	11

2.5.1 Buck Converter.....	12
2.5.2 Fuzzy Logic	14
2.5.3 Relay.....	17
2.6 Media Komunikasi	17
2.6.1 Module Bluetooth.....	18
2.6.2 I2C.....	19
BAB 3. METODELOGI PENELITIAN	21
3.1 Alat dan Bahan.....	21
3.2 Tahap Penelitian	22
3.3 Blok Diagram dan Flowchat Sistem SCADA	24
3.4 Perancangan Komunikasi SCADA.....	27
3.5 Perancangan RTU	32
3.5.1 Perancangan dan Pengujian Tegangan Output Generator.....	33
3.5.2 Perancangan dan Pengujian Buck Converter	35
3.5.3 Perancangan dan Pengujian Kontrol Fuzzy	36
3.5.4 Perancangan dan Pengujian Sensor Tegangan Premover dan Exciter	40
3.5.5 Perancangan dan Pengujian Sensor Arus.....	41
3.5.6 Perancangan dan Pengujian Relay	41
3.6 Perancangan MTU	43
3.6.1 Perancangan HMI.....	44
3.6.2 Perancangan dan Pengujian Database	46
3.7 Wiring Diagram SCADA.....	46
BAB 4. HASIL DAN PEBAHASAN.....	48
4.1 Komunikasi Sistem	48
4.2 Hasil Pengujian RTU	56
4.2.1 Hasil Pengujian Buck Converter.....	58
4.2.2 Hasil Pengujian Sensor Tegangan Output Generator	62
4.2.3 Hasil Pengujian Sensor Tegangan Premover dan Exciter.....	63
4.2.4 Hasil Pengujian Relay	64
4.2.5 Hasil Pengujian Sensor Arus.....	65

4.3 Hasil Pengujian MTU	67
4.3.1 Hasil Pengujian Database.....	67
4.3.2 Hasil Perancangan HMI.....	71
4.4 Analisis Proses SCADA Secara Keseluruhan.....	72
4.3.1 Supervisory Control	72
4.3.2 Data Acquisition.....	74
BAB 5. KESIMPULAN DAN SARAN	77
5.1 Kesimpulan	77
5.2 Saran	78
DAFTAR PUSTAKA	79



DAFTAR TABEL

	Halaman
2.1 Tabel Spesifikasi Arduino Uno.....	9
2.2 Tabel Konfigurasi Pin Modul Bluetooth HC-05.....	19
3.1 Tabel Daftar Perlatan yang Digunakan.....	21
3.2 Tabel Data Yang Melalui Komunikasi Serial.....	31
4.1 Tabel Data Yang Melalui Komunikasi Serial.....	55
4.2 Tabel Pengujian Buck Converter.....	66
4.3 Tabel Hasil Pengujian Buck Converter.....	67
4.4 Tabel Hasil Kalibrasi Sensor Tegangan RTU.....	70
4.5 Tabel Hasil Kalibrasi Tegangan Premover dan Exciter.....	72
4.6 Tabel Hasil Pengujian Relay.....	73
4.7 Tabel Hasil Pengujian Sensor Arus.....	75
4.8 Tabel Hasil Sampling Data Dari Database.....	79
4.9 Tabel Hasil Pengukuran Tegangan Generator DC terhadap Beban tanpa Kontrol.....	82
4.10 Tabel Hasil PengukuranTegangan Generator DC terhadap Beban dengan Kontrol Fuzzy.....	83

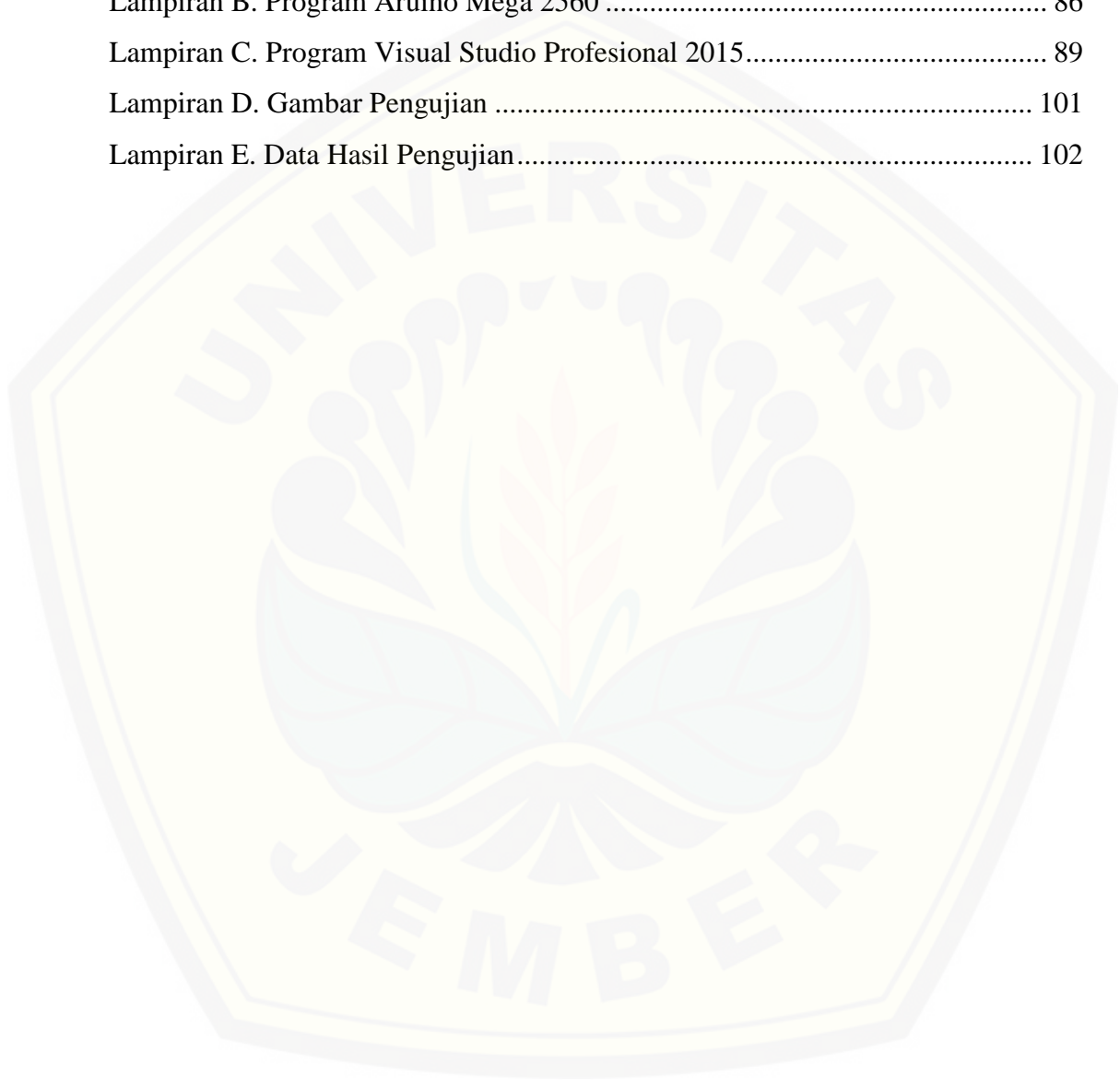
DAFTAR GAMBAR

	Halaman
2.1 Gambar Board Arduino Mega 2560.....	7
2.2 Gambar Model Arduino Uno	8
2.3 Gambar Sensor Tegangan DCT-Elektronik.....	10
2.4 Gambar Sensor Arus ACS712	11
2.5 Gambar Output Voltage And Current Waveform.....	12
2.6 Gambar Voltage Step Down Chopper With Feedback Cntroller.....	14
2.7 Gambar Blok Diagram Sistem Fuzzy Logic	15
2.8 Gambar Skematik Relay	17
2.9 Gambar Konfigurasi Pin Bluetooth HC-05	18
3.1 Gambar Diagram Alur Penelitian.....	23
3.2 Gambar Block Diagram SCADA	25
3.3 Gambar Flowchart Sistem SCADA	26
3.4 Gambar Rangkaian I2C	28
3.5 Gambar Model Pengalamatan I2C.....	29
3.6 Gambar Rangkaian dan Pengalamatan Komunikasi Serial.....	30
3.7 Gambar Pemodelan RTU pada PCB	33
3.8 Gambar Skematik Sensor Tegangan	38
3.9 Gambar Skematik Buck Converter	35
3.10 Gambar Membership Function Error	36
3.11 Gambar Membership Fuction Delta Error	37
3.12 Gambar Rule Base.....	38
3.13 Gambar Himpunan Defuzzyfikasi.....	38
3.14 Gambar Perhitungan Titik Berat Luasan.....	39
3.15 Gambar Skematik Sensor Tegangan Dengan Optocupler.....	40
3.16 Gambar Konfigurasi Pin Sensor Arus.....	41
3.17 Gambar Konfigurasi Relay	42
3.18 Gambar Blok Diagram Master Terminal Unit	43
3.19 Gambar Flowchart Program Master Terminal Unit.....	44

3.20	Gambar Human Machine Interface Sistem Utama	45
3.21	Gambar Human Machine Interface Sistem Login	45
3.22	Gambar Wiring Diagram Pengkabelan SCADA.....	47
4.1	Gambar Tampilan Passcode Bluetooth HC-05	48
4.2	Gambar Interface Komunikasi Panel	42
4.3	Gambar Program Pembacaan Port Otomatis	49
4.4	Gambar Program Pengiriman Data dari RTU	51
4.5	Gambar Program Menerima Dan Mengolah Data Dari MTU	52
4.6	Gambar Program Komunikasi Master I2C	53
4.7	Gambar Program Komunikasi Slave I2C.....	54
4.8	Gambar Program Penerimaan Data Komunikasi Serial MTU.....	55
4.9	Gambar Program Pengiriman Data Komunikasi Serial MTU	56
4.10	Gambar Hasil Pemodelan RTU.....	57
4.11	Gambar Program Pengujian Duty Cycle Buck Converter	59
4.12	Gambar Karakteristik Duty Cycle Terhadap Output Tegangan.....	60
4.13	Gambar Respon Kontrol Pada Buck Converter	61
4.14	Gambar Program Sensor Tegangan Output Generator	62
4.15	Gambar Program Sensor Prime Mover Dan Exciter.....	63
4.16	Gambar Program Sensor Arus	65
4.17	Gambar Program Database	67
4.18	Gambar Hasil Pengujian Database Pada MTU	68
4.19	Gambar Hasil Database Sistem.....	69
4.20	Gambar Hasil Monitoring Dan Kontrol Sistem SCADA.....	71
4.21	Gambar Instrumen Kontrol Sistem	72
4.22	Gambar Grafik Perbandingan Tegangan <i>Output</i> Generator DC Tanpa Kontrol dan Dengan Kontrol.....	74

DAFTAR LAMPIRAN

Lampiran A. Program Arduino Uno	82
Lampiran B. Program Aruino Mega 2560	86
Lampiran C. Program Visual Studio Profesional 2015.....	89
Lampiran D. Gambar Pengujian	101
Lampiran E. Data Hasil Pengujian.....	102



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Sektor Industri dan energi menduduki posisi teratas dalam penyumbang devisa negara (BPS, 2010). Para pekerja yang bekerja pada sektor ini dituntut untuk memiliki sifat pekerja keras, inovatif, dan konsentrasi yang tinggi terutama di bidang industri listrik. Tetapi pada kenyataan saat ini, pekerja di sektor ini bekerja dengan menggunakan metode konvensional yaitu dengan cara mengecek, mengontrol serta mengukur kondisi sistem yang sedang berjalan pada umumnya menggunakan cara yang manual. Proses ini memakan waktu yang lama dan melelahkan. Secara psikologis ini akan mempengaruhi kondisi dari pekerja tersebut. Sehingga data dan pekerjaan yang dilakukan kurang baik. Hal ini bisa berdampak buruk terhadap mesin dan komponen yang berada di lapangan. Komponen dan peralatan dalam industri seperti PLC, sensor-sensor dan *aktuator* sangatlah mahal sehingga jika terjadi kesalahan maka pembangkit tersebut harus di matikan sementara untuk perbaikan. Hal ini akan mengakibatkan kerugian besar terhadap perusahaan. Pekerja dalam bidang industri listrik juga diuntut untuk menjaga kesetabilan tegangan dari generator DC.

Permasalahan pada kesetabilan tegangan dari generator DC saat dibebani dari *output* generator DC akan menimbulkan efek *drop* tegangan. Pada umumnya untuk tegangan *output* akan cenderung turun bila arus beban semakin besar. Mengatasi hal tersebut dapat dilakukan dengan cara meningkatkan tegangan induksi (E_a) pada jangkar generator DC. Tegangan induksi jangkar dapat ditingkatkan dengan cara menaikkan kecepatan putaran atau *flux* pada medan generator DC. Kecepatan putaran generator DC sulit dikendalikan karena dibutuhkan pengontrol kecepatan yang terdiri dari kontrol mekanis atau elektris yang rumit, terlebih pada kecepatan yang tinggi. Oleh karena itu, untuk meningkatkan tegangan induksi pada jangkar generator DC akan lebih mudah bila diatur arus eksitasi pada medan generator DC sehingga nilai *flux* pada medan akan meningkat (Chapman, 2005).

Pesatnya perkembangan ilmu dan teknologi membuat kehidupan manusia jauh lebih efektif dan efisien dalam mengerjakan suatu pekerjaan. Keefektifan dan keefisienan ini dapat meningkatkan produktivitas kerja seorang pekerja terutama pada sektor industri dan energi. Upaya dalam meningkatkan produktivitas pekerja dengan membuat suatu sistem berbasis SCADA. SCADA adalah sistem yang mengumpulkan informasi atau data-data dari lapangan dan kemudian mengirimkannya ke sebuah komputer pusat yang akan mengatur dan mengolah data-data tersebut. Protokol komunikasi SCADA juga digunakan pada sistem otomasi kontrol di industri sebagai sistem komunikasi data untuk memantau dan mengontrol peralatan industri. SCADA pada industri saat ini masih menggunakan sistem kontrol PLC (*programmable logic controller*) (Kusriyanto dan Syarifudin).

SCADA (*Supervisory Control And Data Acquisition*) sangat dibutuhkan untuk lebih mempermudah pekerjaan. SCADA memberikan keleluasaan mengatur maupun mengkonfigurasi sistem. Kita bisa menempatkan sensor dan kendali di setiap titik kritis didalam proses. Seiring dengan teknologi sistem SCADA yang semakin baik, kita bisa menempatkan lebih banyak sensor di banyak tempat sehingga semakin banyak hal yang bisa dipantau, semakin detail operasi yang bisa dilihat, dan semuanya bekerja secara *real-time*, yaitu kondisi pengoperasian dari suatu sistem perangkat keras dan perangkat lunak yang dibatasi oleh rentang waktu dan memiliki tenggang waktu yang jelas, relatif terhadap waktu suatu peristiwa atau operasi terjadi. Tidak peduli serumit apapun prosesnya, kita bisa melihat operasi proses dalam skala besar maupun kecil, dan setidaknya bisa melakukan penelusuran jika terjadi kesalahan dan sekaligus meningkatkan efisiensi. Namun ditinjau dari segi ekonomis, biaya pengadaan dan pembuatan sistem SCADA sangatlah mahal di dalam sistem pembangkit listrik .

Terinspirasi dari sistem yang sangat kompleks, tingkat *safety* yang tinggi, pengolahan data serta harga yang mahal membuat topik ini hangat untuk diangkat. Peneliti ingin mengangkat topik permasalahan tentang implementasi SCADA pada generator DC penguatan terpisah. Sistem ini menggunakan controller arduino mega 2560 dan aplikasi yang dibangun dari microsoft visual studio yang diharapkan lebih ekonomis. Menggunakan *buck converter* yang di kontrol secara *close loop* oleh

Arduino mega 2560 melalui *Master terminal unit* (MTU). Untuk mengatur arus eksitasi pada medan generator DC sehingga nilai *flux* pada medan akan meningkat yang di harapkan akan membuat *output* generator DC stabil. Dari penelitian ini diharapkan sistem SCADA jadi lebih ekonomis dan efektif.

1.2 Rumusan Masalah

Dalam penelitian ini ada beberapa hal yang menjadi rumusan masalah untuk menjadi rujukan apa yang akan dilakukan dan diteliti, diantaranya :

1. Bagaimana penerapan SCADA pada generator penguatan terpisah dengan Arduino?
2. Bagaimana menghubungkan sistem SCADA pada Arduino dengan PC?
3. Bagaimana kestabilan tegangan *output* generator dengan sistem SCADA?

1.3 Batasan Masalah

Berdasarkan uraian rumusan masalah tersebut, maka pembahasan pada penelitian ini dibatasi pada pembahasan berikut:

1. Kontrol *fuzzy logic* pada tegangan *output* hanya dengan mengatur arus eksitasi medan di stator generator DC penguatan terpisah menggunakan *buck converter* sedangkan kontrol *boolean logic* di gunakan untuk pensaklaran pada sistem.
2. *Premover* menggunakan mesin DC *compound*.
3. Pengujian pada kestabilan tegangan *output* menggunakan variasi beban lampu pijar 15 W, 25 W, 40 W , 50 W dan 65 W dengan cara pemberian beban.
4. Media komunikasi menggunakan menggunakan modul Bluetooth secara *wireless* untuk *Remote terminal unit* ke *Master terminal unit*
5. Pengujian dilakukan pada tegangan *output* sebesar 100 Volt dengan variasi beban yang berbeda.
6. HMI dan *Master terminal unit* dibangun dari Microsoft Visual Studio 2015 dengan Bahasa pemograman basic sedangkan *Remote terminal unit* menggunakan Arduino dengan Bahasa pemograman C#.

1.4 Tujuan

Penelitian yang dilakukan ini memiliki beberapa tujuan untuk dicapai sebagai pencapaian akhir, diantaranya :

1. Membuat sistem SCADA dengan Arduino sebagai pengganti dari PLC-SCADA yang mahal.
2. Menerapkan sistem SCADA pada Arduino dengan generator DC penguatan terpisah.
3. Menerapkan kontrol *fuzzy logic* sehingga diperoleh kestabilan *output* tegangan generator DC terhadap variasi beban yang diberikan.

1.5 Manfaat

Penelitian ini memiliki beberapa manfaat diantaranya :

1. Memudahkan dalam pengendalian dan pengecekan kondisi pembangkit listrik dengan generator DC penguatan terpisah.
2. Membuat sistem SCADA jauh lebih murah
3. Mengatasi *drop* tegangan dari generator DC penguatan terpisah ketika diberikan beban.
4. Tegangan *output* generator DC penguatan terpisah akan diatur secara otomatis dengan sistem *close loop*.

BAB 2. TINJAUAN PUSTAKA

2.1 SCADA

Sistem pengaturan yang berbasis komputer disebut sebagai SCADA (*supervisory control and data acquisition*). SCADA adalah suatu sistem yang dapat memonitor dan mengontrol suatu peralatan atau sistem secara *real time*. SCADA terdiri dari perlengkapan *hardware* dan *software*. Dengan adanya peralatan SCADA penyampaian dan pemrosesan data dari sistem tenaga listrik akan lebih cepat diketahui oleh operator (*dispatcher*) (Winarno, 2010). Penyaluran sistem peralatan memungkinkan penyampaian data secara *remote*. Data yang baru dapat juga dihitung dan disimpan dalam data base melalui pengumpulan nilai secara otomatis.

SCADA memiliki 3 fungsi dari Sistem, yaitu adalah *telecontrol*, *telesignaling*, dan *telemetry*. *Telecontrol* berfungsi melakukan perintah *remote control* (*open/close*) terhadap peralatan yang berada dilapangan. *Telesignaling* berfungsi mengumpulkan data status dan alarm (*open, close, power supply fault*, indikasi relay atau parameter lainnya) yang dianggap perlu yang dapat membantu *dispatcher* dalam memonitor peralatan yang berada dilapangan. *Telemetry* berfungsi mengukur beban yang terpasang pada alat ukur tenaga listrik (arus, tegangan, daya aktif, frekuensi dan lain-lain) dan semua peralatan yang berada di lapangan. Adanya peralatan SCADA ini penyampaian dan pemrosesan data dari sistem tenaga listrik akan lebih cepat diketahui oleh *dispatcher* (Novel, 2009).

2.1.1 MTU

MTU (*master terminal unit*) merupakan sebuah sistem komputer yang berfungsi memberikan data kepada HMI dari RTU. Di sisi lain MTU ini juga mengambil data dari RTU untuk diterjemahkan dan diberikan ke HMI. Sistem pengambilan data dari RTU disebut "*Polling*". Terkadang MTU dan HMI dapat dijadikan 1 bagian, ketika MTU menggunakan komputer yang sama dengan HMI.

HMI adalah subsistem dari MTU yang berfungsi menampilkan data dari hasil pengukuran di RTU ataupun menampilkan proses yang sedang terjadi pada

keseluruhan sistem. HMI merupakan sebuah *software* pada komputer berbasis grafis yang berfungsi untuk mempermudah pengawasan (*supervisory*) kepada operator. HMI mengubah data-data dan angka kedalam animasi, grafik/trend, dan bentuk yang mudah diterjemahkan oleh operator (Anita *et al.*, 2016).

Data yang diolah oleh HMI kemudian disimpan pada *database*. Data adalah fakta yang dapat direkam dan memiliki arti secara implisit. Sedangkan kumpulan data yang memiliki hubungan secara implisit itu disebut *database* (Stouffer *et al.*, 2006). *Database* sebagai kumpulan data, umumnya mendeskripsikan aktivitas satu organisasi atau lebih yang berhubungan. Dalam *database* juga dikenal istilah DBMS (*database management systems*) yaitu sekumpulan program yang memungkinkan pengguna untuk membuat dan memelihara suatu *database* (System, 2004). Bisa juga dikatakan bahwa DBMS merupakan perangkat *general purpose software system* yang berfungsi untuk mewartakan proses-proses dalam *database* seperti pendefinisian, pembuatan, *sharing*, maupun manipulasi *database*. Dalam *database*, dikenal istilah ER (*entity relationship*). Entitas adalah sesuatu atau objek dalam dunia nyata yang dapat dibedakan dari objek lain. Misalnya mahasiswa, dan matakuliah. Entitas digambarkan dalam basis data dengan kumpulan atribut. Misalnya nim, nama, alamat, dan kota. Relasi adalah hubungan antara beberapa entitas. Misalnya relasi menghubungkan mahasiswa dengan mata kuliah yang diambilnya. Bisa juga dikatakan bahwa ER didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut entitas dan hubungan antar objek.

2.1.2 RTU

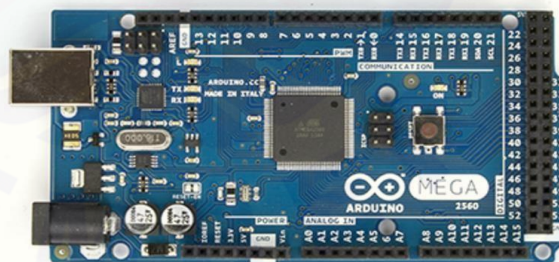
RTU (*Remote Terminal Unit*) adalah subsistem SCADA yang berfungsi sebagai terminal-terminal dari hasil pengukuran, pengendalian, pemantauan status dan lain-lain. RTU juga berfungsi menerjemahkan, mengkonversi, menghitung sinyal dari *transducer* seperti pengukuran arus listrik, *Flow*, *Static Pressure*, *Differensial Pressure*, temperatur, dan lain-lain. Dari hasil pengukuran tersebut hal yang dilakukan RTU adalah melakukan kendali kemudian mengirim data ke MTU

(Novel, 2009). RTU juga dapat berfungsi sebagai pengatur *setpoint* yang dikirimkan dari HMI ke RTU tersebut.

RTU berbeda dari *programmable logic controller* (PLC). RTU lebih cocok untuk telemetri geografis yang luas, yang sering menggunakan komunikasi nirkabel, sementara PLC lebih cocok untuk media fisik untuk kontrol. Dalam sistem *basic SCADA* umumnya digunakan PLC sebagai pengganti RTU (winarno, 2010).

2.2 Arduino Mega 2560

Arduino adalah sebuah mikrokontroler *single board* yang bersifat *open source hardware* mikrokontroler Arduino diprogram dengan menggunakan bahasa pemrograman *wiringbased* yang berbasiskan *sysntax* dan *library*. Untuk memudahkan dalam pengembangan aplikasi, Arduino dilengkapi *software IDE (integrated development environment)* berbasis *processing* (Hadiyanto, 2015). Perangkat ini sangat populer di kalangan mahasiswa dan *developer* dikarenakan kemudahan dalam penggunaannya. Selain itu, penggunaan Arduino dapat meningkatkan kreatifitas serta inovasi dalam pengembangan ilmu pengetahuan. Bentuk dari Aduino Mega 2560 seperti Gambar 2.1.



Gambar 2.1. Board Arduino Mega 2560
(Sumber: <https://www.Arduino.cc>)

Arduino Mega 2560 berbeda dari papan sebelumnya, karena versi terbaru sudah tidak menggunakan chip driver FTDI USB-to-serial. Tetapi, menggunakan chip ATmega16U2 (ATmega8U2 pada papan Revisi 1 dan Revisi 2) yang diprogram sebagai konverter USB-to-serial. Arduino Mega2560 Revisi 2 memiliki resistor penarik jalur HWB 8U2 ke Ground, sehingga lebih mudah untuk

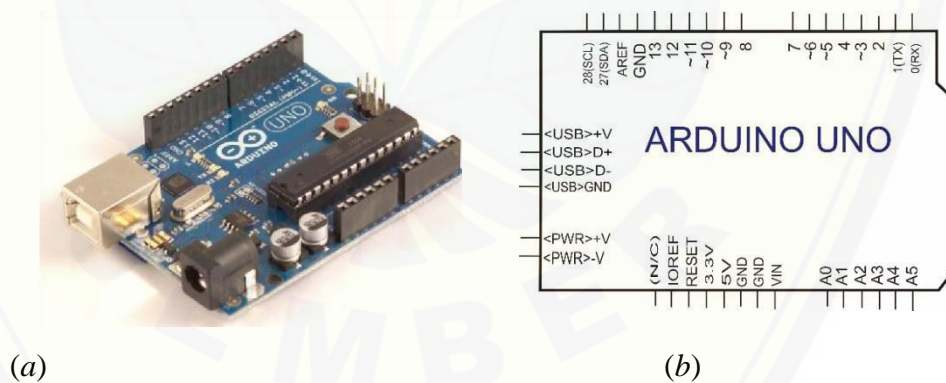
dimasukkan ke dalam mode DFU. Arduino Mega 2560 Revisi 3 memiliki fitur-fitur baru berikut:

- a. *Pin out*;
- b. Sirkuit RESET;
- c. Chip ATmega16U2 menggantikan chip ATmega8U2.

2.3 Arduino Uno

Arduino Uno adalah modul mikrokontroler berbasis ATmega 328. Modul ini memiliki empat belas pin *input-output* (14 I/O) di mana enam diantaranya dapat digunakan sebagai *output* PWM (*pulse width modulation*), enam *input* analog, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, tombol *reset* dan koneksi ke catu daya. Kontroler ini semua yang dibutuhkan untuk mendukung mikrokontroler.

“Uno” berarti satu dalam Bahasa Italia dan hal ini untuk menandai versi pertama (Arduion 1.0) yang akan menjadi versi acuan bagi Arduion berikutnya. Uno adalah seri terakhir dari seri papan atau modul USB Arduino (Fajar, 2016). Uno tidak lagi menggunakan *chip* FTDI USB-to-serial tetapi menggunakan Atmega 8U2 sebagai konverter USB ke serial. Skematik dan bentuk dari Arduino Uno R3 seperti pada Gambar 2.2.



(a) Mikrokontroler Arduino Uno R3; (b) Skematik Arduino UNO

Gambar 2.2 Model Arduino Uno
(Sumber: Fajar, 2016)

Gambar 2.2(a) adalah modul Arduino Uno-R3 sedangkan Gambar 2.2(b) menunjukkan skema dari modul Arduino Uno-R3 yang merupakan penyederhanaan dari gambar modul aslinya (Arduino Uno 2015). Masing-masing pin I/O dapat digunakan sebagai *input* maupun *output*. Masing-masing pin ini

beroperasi pada tegangan 5 V dan dapat menerima arus 40 mA serta sudah memiliki *pull-up* resistor 20-50 k Ω yang secara *default* tidak tersambung (*disconnected*). Berikut ini adalah tabel spesifikasi dari Arduino Uno R3 dapat dilihat pada Tabel 2.1.

Tabel 2.1 Spesifikasi Arduino Uno

Spesifikasi	Keterangan
Microcontroller	ATmega328
Tegangan Operasi	5 V
Tegangan <i>Input</i>	7-12 V
Tegangan <i>Input</i> (Batas)	60-20 V
I/O Digital	14
Input Analog	6 buah
Arus DC per I/O pin	40 Ma
Arus DC untuk pin 3.3	50 Ma
Flash Memory	Bootloader
SRAM	2 Kb
EEPROM	1 kB
Kecepatan Klok	16 MHz

Diterjemahkan dari: (Arduino UNO 2015)

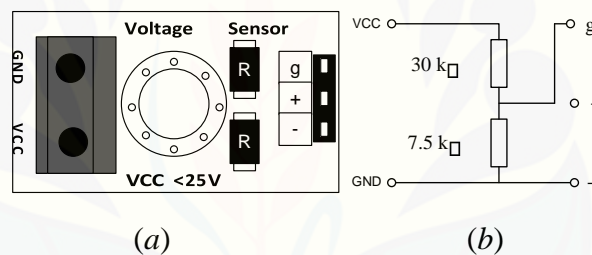
Atmega328 memiliki 32KB *on-Chip in system reprogrammable flash memory* untuk menyimpan program. Memori *flash* dibagi kedalam dua bagian, yaitu bagian program *bootloader* dan aplikasi. *Bootloader* adalah program kecil yang bekerja pada saat sistem dimulai yang dapat memasukkan seluruh program aplikasi kedalam memori prosesor. Memori data Atmega 328 terbagi menjadi 4 bagian yaitu 32 lokasi untuk register I/O tambahan dan sisanya 2048 lokasi untuk data SRAM internal. Register umum menempati alamat data terbawah yaitu 0x0000 sampai 0x001F. Register I/O menempati 64 alamat berikutnya mulai dari 0x0020 hingga 0x005F. Register I/O tambahan menempati 160 alamat berikutnya mulai dari 0x0060 hingga 0x00FF. Sisa alamat berikutnya mulai dari 0x0100 hingga 0x08FF digunakan untuk SRAM internal (Setiawan, 2016).

2.4 Sensor

Untuk menunjang keakuratan data hasil dari pengontrolan, maka digunakan juga beberapa sensor sesuai kebutuhan. Sensor disini bekerja memantau langsung kondisi objek yang hendak diatur, informasi tersebut kemudian dikirim ke *controller* untuk diolah. Agar data yang diperoleh dapat akurat sesuai keinginan.

2.4.1 Sensor Tegangan

Sensor tegangan adalah modul pembagi tegangan yang digunakan untuk mengubah tegangan yang akan diukur untuk dapat diumpankan ke pin kontroler yang sudah memiliki konverter analog ke digital (ADC). Sensor tegangan ini seperti pada Gambar 2.3.



(a) Modul Sensor Pembagi Tegangan; (b) Skematik Sensor Tegangan

Gambar 2.3 Sensor tegangan DCT-Elektronik
(Sumber: Setiyawan, 2016)

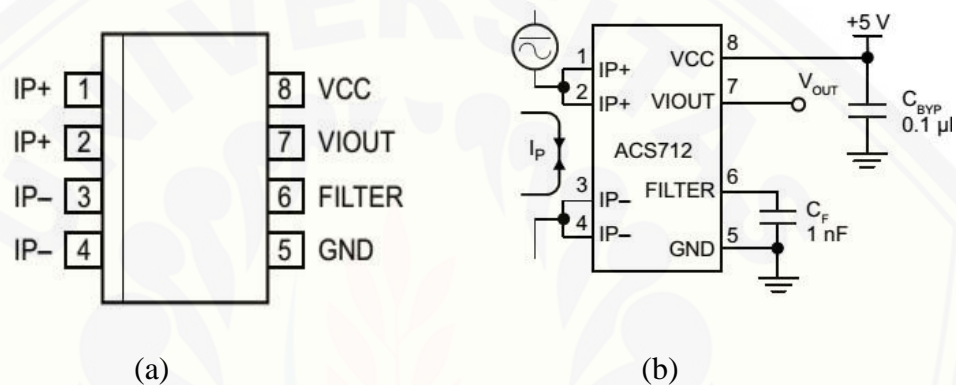
Gambar 2.2(a) adalah modul sensor pembagi tegangan, sedangkan Gambar 2.2(b) adalah diagram skematik dari sensor tersebut. Kaki VCC dan GND adalah kaki-kaki untuk menempatkan terminal tegangan yang akan diukur sedangkan pin g, + dan – adalah pin untuk dihubungkan dengan kaki ADC kontroler (Setiyawan, 2016).

2.4.2 Sensor Arus

ACS712 adalah sensor arus yang dapat digunakan untuk mengindera arus DC atau AC dengan standar industri, biasanya digunakan untuk keperluan kendali motor, manajemen beban, proteksi arus lebih dan juga untuk keperluan pembuatan

switchmode power supply (SMPS). Piranti ini tidak didesain untuk keperluan otomotif (Arfianto, 2015).

Output dari piranti ini (V_{IOUT}) memiliki kemiringan positif jika sebuah arus dengan nilai yang meningkat mengalir melalui jalur konduktor tembaga primer (dari pin 1 dan pin 2 ke pin 3 dan pin 4), yang merupakan jalur dari arus yang akan diukur. Resistansi internal dari jalur konduksi ini adalah $1.2\text{ m}\Omega$ (tipikal) sehingga tidak menyebabkan tegangan jatuh yang besar. Modul dan skematik sensor seperti pada Gambar 2.4.



(a) Modul Sensor Pembagi Tegangan; (b) Skematik Sensor Tegangan

Gambar 2.4 Sensor Arus ACS712
(Sumber: Arfianto, 2015)

Gambar 2.4(a) menunjukkan kemasan IC sensor arus ACS712. Gambar 2.4(b) adalah diagram skematik yang menggambarkan hubungan kaki-kaki *input* dan *output* dari ACS712.

2.5 Aktuator

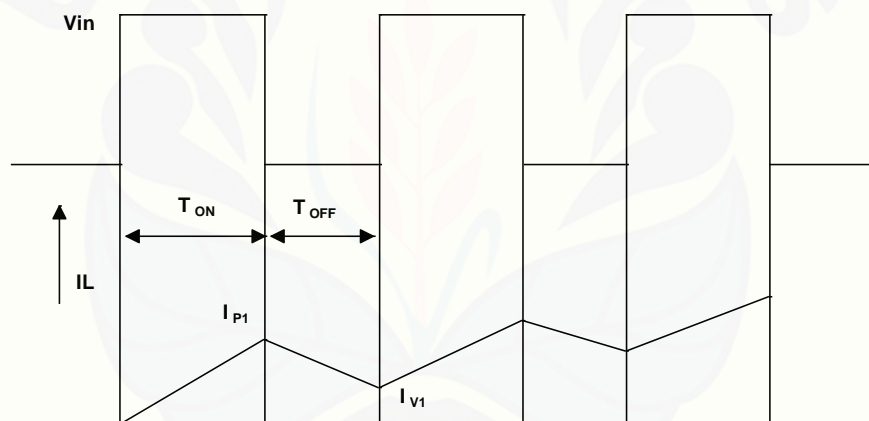
Aktuator adalah sebuah peralatan mekanis untuk menggerakkan atau mengontrol sebuah mekanisme atau sistem. Aktuator diaktifkan dengan menggunakan lengan mekanis yang biasanya digerakkan oleh motor listrik, yang dikendalikan oleh media pengontrol otomatis yang terprogram di antaranya mikrokontroler. Aktuator adalah elemen yang mengkonversikan besaran listrik analog menjadi besaran lainnya misalnya kecepatan putaran dan merupakan perangkat elektromagnetik yang menghasilkan daya gerakan sehingga dapat menghasilkan gerakan pada robot. Untuk meningkatkan tenaga mekanik aktuator

ini dapat dipasang sistem gearbox. Aktuator dapat melakukan hal tertentu setelah mendapat perintah dari controller (Sutarna *et al.*, 2011).

2.5.1 Buck Converter

Buck converter adalah salah satu DC-DC *converter* yang dapat menurunkan tegangan DC berdasarkan *duty-cycle* MOSFET pada rangkaian *buck converter*. *Output* tegangan pada *buck converter* akan turun bila *duty-cycle* pada buck turun (Khadiri & Qjidaa, 2014).

Menurut Fajar (2016), Prinsip kerja dari *buck converter* yaitu dimana MOSFET yang mengatur proses *switching* yang tinggi pada saat pemberian sinyal PWM yang dikontrol. Selain MOSFET yang digunakan untuk proses *switching* bisa juga dengan menggunakan IGBT. Gelombang yang dihasilkan dari tegangan *output* seperti pada Gambar 2.5.



Gambar 2.5 *Output voltage and current waveform*
(Sumber: Fajar, 2016)

Ketika MOSFET pada keadaan ON maka tegangan V_{in} diterapkan secara *reverse* pada dioda. Karena itu dioda harus tetap pada kondisi OFF selama MOSFET tetap pada keadaan ON. Keadaan ON dari MOSFET selalu menunjukkan keadaan OFF dari dioda. Ketika MOSFET pada keadaan ON, maka arus I_L meningkat. Kenaikan arus I_L terjadi secara eksponensial dikarenakan induktansi (L). T_{ON} untuk selang waktu MOSFET pada kondisi ON dan T_{OFF} untuk keadaan MOSFET pada kondisi OFF. Pada saat MOSFET pada kondisi OFF, I_L memiliki nilai yang terbatas (I_{P1}) yang merupakan nilai puncak arus *output* selama selang waktu siklus pertama. Nilai puncak terjadi pada saat MOSFET berubah

kondisi OFF dimana terdapat induktansi yang bertujuan untuk mencegah penurunan atau *drop* I_L ke nol secara tiba-tiba.

Untuk tegangan *output* yang dihasilkan dari *buck converter* akan memiliki hubungan dengan tegangan *input* dan juga *duty cycle* dimana D merupakan *duty cycle* hal ini dapat ditunjukkan dengan persamaan rumus dibawah ini:

$$V_{out} = V_{in} \cdot D \quad (2.1)$$

$$D_{in} = \frac{T}{T_{on} + T_{off}} \quad (2.2)$$

Sinyal PWM merupakan salah satu sinyal kontrol yang paling sering diterapkan pada proses *switch* dalam *switching DC converter*. Sinyal PWM ini sering digunakan pada proses *switching* dikarenakan sinyal PWM merupakan sinyal dengan frekuensi tetap. Frekuensi sangat berhubungan dengan *output ripple*. Untuk mendapatkan tegangan *output ripple* yang jauh lebih kecil dari nilai rata-rata, maka sebagian besar arus *ripple* induktor harus melalui kapasitor. Tegangan *output ripple* dapat ditentukan oleh persamaan berikut:

$$\Delta V_{out} = \frac{(1-D)T_{sw}}{L \cdot C} \quad (2.3)$$

Dimana

T_{sw} merupakan periode *switching*

$$f_{sw} = 1 / T_{sw}$$

Tegangan *output ripple* dari DC-DC *converter* dapat diminimalkan dengan penentuan pada besarnya nilai induktor dan juga pada kapasitor yang diberikan, hal ini dapat ditentukan seperti pada perhitungan dibawah ini $D(V^{IN} - V^{OUT})$

$$L = \frac{D(V_{in} - V_{out})}{I_{ripple} \times F_s} \quad (2.4)$$

$$C = \frac{I_{ripple}}{8 \cdot F_s \times \Delta V} \quad (2.5)$$

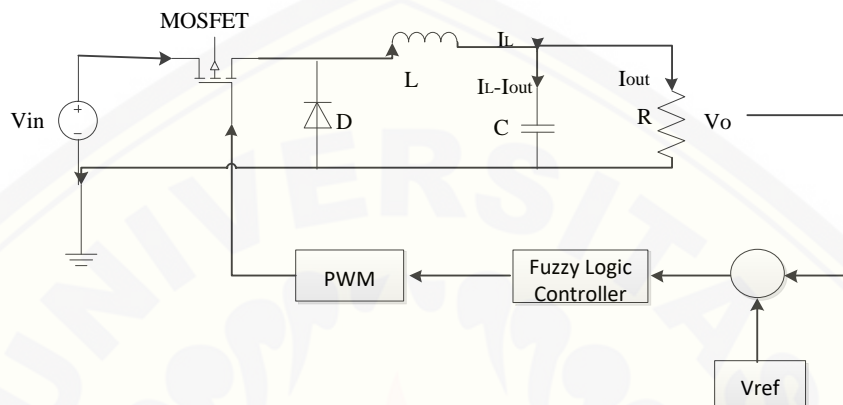
Dimana V_{IN} = *Input* tegangan pada *buck converter*

V_{OUT} = *Output* tegangan

F_s = *Frekuensi switching*

I_{ripple} merupakan arus *ripple* induktor, biasanya 30% dari arus keluaran maksimum. ΔV adalah tegangan *output ripple*, biasanya 1% dari tegangan *output*.

Kemudian untuk ESR (*effective series resistance*) merupakan resistansi seri efektif kapasitor dimana pada umumnya 0.03Ω . Sebuah kapasitor dengan ESR rendah akan membantu untuk meminimalkan *ripple* pada *output* tegangan (Khadiri & Qjidaa, 2014). Rangkaian penurun tegangan dengan kontrol seperti pada Gambar 2.6.



Gambar 2.6 Voltage step down chopper with feed back controller
(Sumber: Fajar, 2016)

Menurut Khadiri dan Qjidaa (2014), Dalam DC-DC *converter*, kerugian dapat diklasifikasikan menjadi dua jenis yaitu kerugian konduksi dan juga pada frekuensi kerugian *switching*.

Dari dua jenis kerugian tersebut masing-masing dibagi lagi menjadi:

a. Rugi konduksi

Selama mode konduksi kontinyu (berarti induktor saat ini tidak akan mencapai ke nol) di mana arus beban relatif besar, kontribusi utama dari rugi daya adalah hilangnya konduksi pada resistensi dari sisi tinggi (R_{on_PFET}) dan sisi rendah (R_{on_NFET}) switch dan resistansi seri dari induktor dan kapasitor (R_L, R_{ESR})

- 1) Rugi konduksi saat *switching*;
- 2) Rugi konduksi pada induktor dan kapasitor.

b. Rugi *switching*

Kerugian-kerugian pada *switching* tergantung pada frekuensi, hal ini dapat dibedakan menjadi dua jenis yaitu:

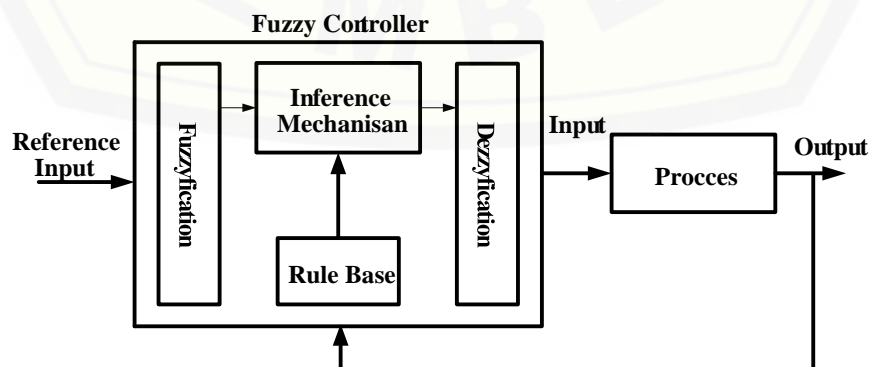
- 1) *Hard Loss (overlap loss)*;
- 2) *Soft Loss (gate drive loss)*.

2.5.2 Fuzzy Logic

Logika fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang *input* ke dalam ruang *output* (Hadiyanto, 2015). Untuk sistem yang sangat rumit, penggunaan logika fuzzy (*fuzzy logic*) adalah salah satu pemecahannya. Sistem tradisional dirancang untuk mengontrol keluaran tunggal yang berasal dari beberapa masukan yang tidak saling berhubungan. Karena ketidaktergantungan ini, penambahan masukan yang baru akan memperumit proses kontrol dan membutuhkan proses perhitungan kembali dari semua fungsi. Kebalikannya, penambahan masukan baru pada sistem fuzzy, yaitu sistem yang bekerja berdasarkan prinsip-prinsip logika fuzzy, hanya membutuhkan penambahan fungsi keanggotaan yang baru dan aturan-aturan yang berhubungan dengannya (Kusumadewi & Hartati, 2006).

Secara umum, sistem fuzzy sangat cocok untuk penalaran pendekatan terutama untuk sistem yang menangani masalah-masalah yang sulit didefinisikan dengan menggunakan model matematis. Misalkan, nilai masukan dan parameter sebuah sistem bersifat kurang akurat atau kurang jelas, sehingga sulit mendefinisikan model matematikanya.

Logika fuzzy memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk *linguistik*, konsep tidak pasti seperti "sedikit", "lumayan", dan "sangat". Dia berhubungan dengan *set fuzzy* dan teori kemungkinan. Dia diperkenalkan oleh Dr. Lotfi Zadeh dari Universitas California, Berkeley pada 1965 (Kusumadewi & Hartati, 2006). Perbedaan *Fuzzy Logic* dengan *Boolean Logic* terlihat pada Gambar 2.7.



Gambar 2.7 Blok diagram sistem *Fuzzy Logic*.
(Sumber: Hadiyanto, 2015)

Menurut Kusumadewi & Hartati (2006), Fuzzifikasi atau *fuzzyfication* adalah proses yang dilakukan untuk mengubah variabel nyata menjadi variabel fuzzy, ini ditujukan agar masukan kontroler fuzzy bisa dipetakan menuju jenis yang sesuai dengan himpunan fuzzy. Pemetaan dilakukan dengan bantuan model dari fungsi keanggotaan agar dapat diketahui besar masukan tersebut (derajat keanggotaan). Terdapat beberapa jenis penggambaran fungsi keanggotaan, antara lain:

- a. Gaussian
- b. Segitiga
- c. Trapesium
- d. Bahu

Komponen lainnya yang memiliki peranan penting adalah Label. Label didefinisikan dari fungsi keanggotaan, fungsi keanggotaan apabila dikumpulkan akan menghasilkan fuzzy set. Dalam logika fuzzy terdapat *basis* pengetahuan, dimana terdiri dari fakta (*data base*), dan kaidah atur (*rulebase*) (Setiyawan, 2016).

Prosedur fuzzifikasi merupakan proses untuk mengubah variabel non fuzzy (variabel *numerik*) menjadi variabel fuzzy (variabel *linguistik*). Nilai *error* dan *delta error* yang dikuantisasi sebelumnya diolah oleh kontroler logika fuzzy, kemudian diubah terlebih dahulu ke dalam variabel fuzzy. Melalui *membership function* (fungsi keanggotaan) yang telah disusun, maka dari nilai *error* dan *delta error* kuantisasi akan didapatkan derajat keanggotaan bagi masing-masing nilai *error* dan *delta error* (Kusumadewi & Hartati, 2006).

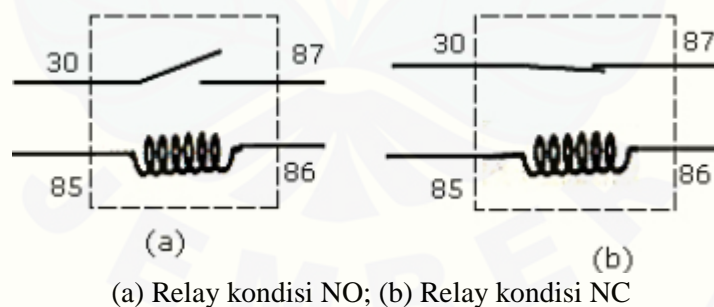
Pada umumnya aturan-aturan fuzzy dinyatakan dalam bentuk “*IF antecedent THEN consequent*” yang merupakan inti dari relasi fuzzy. Terdapat dua model aturan fuzzy yang digunakan secara luas dalam berbagai aplikasi, yaitu model mamdani dan model sugeno (Hadianto, 2015).

Defuzzifikasi dapat didefinisikan sebagai proses perubahan besaran fuzzy yang disajikan dalam bentuk himpunan-himpunan fuzzy keluaran dengan fungsi keanggotaannya untuk mendapatkan kembali bentuk tegasnya (*crisp*). Hal ini diperlukan sebab dalam aplikasi nyata yang dibutuhkan adalah nilai tegas (*crisp*). Prosesnya adalah ketika suatu nilai fuzzy *output* yang berasal dari *rule evaluation*

diambil kemudian dimasukkan ke dalam suatu *membership function output*. Bentuk bangun yang digunakan dalam *membership function output* adalah bentuk *singleton* yaitu garis lurus vertikal ke atas, seperti yang ditunjukkan pada gambar di bawah. Besar nilai fuzzy *output* dinyatakan sebagai *degree of membership function output*. Nilai-nilai tersebut dimasukkan ke dalam suatu rumus yang dinamakan COG (*center of gravity*) untuk mendapatkan hasil akhir yang disebut *crisp output*. *Crisp output* adalah suatu nilai analog yang akan kita butuhkan untuk mengolah data pada sistem yang telah dirancang (Hardianto, 2015).

2.5.3 Relay

Menurut Novel (2009), Relay adalah saklar elektrik yang digunakan untuk memutus dan menghubungkan arus secara elektrik. Cara kerja dari relay sendiri adalah bila dialiri arus listrik kumparan akan menjadi magnet, sehingga kontak point akan tertarik dan terhubung. Ada dua kondisi pada relay. Relay kondisi terbuka (NO), jika dialiri listrik kontak point akan terhubung. Relay kondisi tertutup (NC), jika dialiri arus listrik akan terputus. Gambaran relay dapat dilihat pada Gambar 2.8.



(a) Relay kondisi NO; (b) Relay kondisi NC

Gambar 2.8. Skematik Relay
(Sumber: Novel, 2009)

Saklar atau kontaktor relay dikendalikan menggunakan tegangan listrik yang diberikan ke induktor pembangkit magnet untuk menarik armatur tuas saklar atau kontaktor relay. Relay yang ada dipasaran terdapat berbagai bentuk dan ukuran dengan tegangan kerja dan jumlah saklar yang bervariasi, Contoh relay elektro mekanik relay dibutuhkan dalam rangkaian elektronika sebagai eksekutor sekaligus *interface* antara beban dan sistem kendali elektronik yang berbeda sistem *power*

supply-nya. Secara fisik antara saklar atau kontaktor dengan elektromagnet relay terpisah sehingga antara beban dan sistem kontrol terpisah (Novel, 2009).

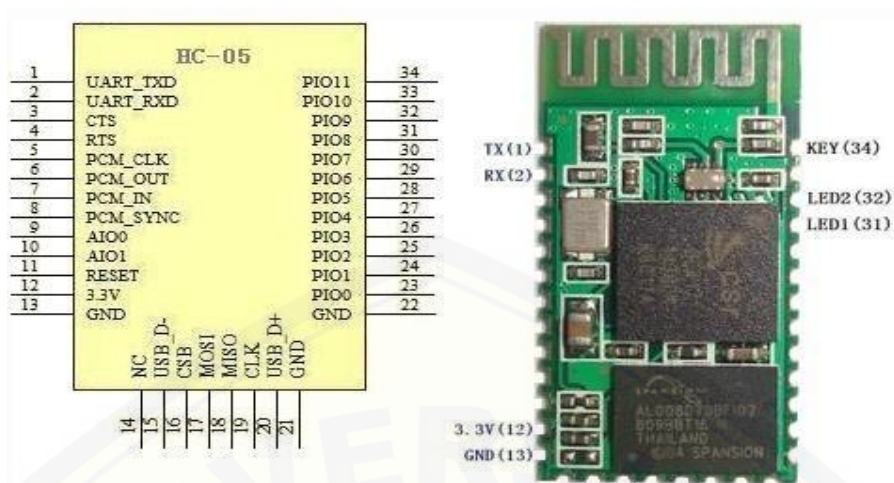
2.6 Media Komunikasi

SCADA tidak pernah lepas dengan komunikasi. Komunikasi yang terdapat pada SCADA yaitu menghubungkan perangkat RTU dan MTU. Menurut Winarno (2010), Ada beberapa media komunikasi yang digunakan dalam proses pengoperasian peralatan sistem SCADA, diantaranya:

- a. Kabel (Pilot Cable, RS 232,RS 485 & PLC);
- b. *Fiber Optic*;
- c. *Radio/Wireless*;
- d. *Microwave*.

2.6.1 Module Bluetooth

Bluetooth adalah protokol komunikasi *wireless* yang bekerja pada frekuensi radio 2.4 GHz untuk pertukaran data pada perangkat bergerak seperti PDA, laptop, HP, dan lain-lain. Salah satu hasil contoh modul Bluetooth yang paling banyak digunakan adalah tipe HC-05. Bluetooth HC05 merupakan salah satu modul Bluetooth yang dapat ditemukan dipasaran dengan harga yang relatif murah. Modul Bluetooth HC-05 terdiri dari 6 pin konektor, yang setiap pin konektor memiliki fungsi yang berbeda – beda. Berikut ini adalah konfigurasi pin Bluetooth HC-05 ditunjukkan pada Gambar 2.9.



(a) Skematik IC HC-05; (b) Bluetooth HC-05

Gambar 2.9. Konfigurasi pin Bluetooth HC-05
(Sumber: Eritha, 2014)

Bluetooth HC-05 dengan *supply* tegangan sebesar 3,3 V ke pin 12 Bluetooth sebagai VCC. Pin 1 pada modul Bluetooth sebagai transmitter, kemudian pin 2 pada Bluetooth sebagai *receiver*. Konfigurasi pin modul Bluetooth HC-05 dapat dilihat pada Tabel 2.2.

Tabel 2.2 Konfigurasi pin modul Bluetooth *HC-05*

No.	Nomor Pin	Nama	Fungsi
1.	Pin 1	Key	
2.	Pin 2	VCC	Sumber Tegangan 3,3 V
3.	Pin 3	GND	Ground Tegangan
4.	Pin 4	TXD	Mengirim Data
5.	Pin 5	RXD	Menerima Data
6.	Pin 6	STATE	

Modul Bluetooth HC-05 merupakan modul Bluetooth yang bisa menjadi *slave* ataupun *master* hal ini dibuktikan dengan bisa memberikan notifikasi untuk melakukan pairing ke perangkat lain, maupun perangkat lain tersebut yang melakukan pairing ke modul Bluetooth HC-05. Untuk mengatur perangkat Bluetooth dibutuhkan perintah-perintah *AT Command* yang mana perintah *AT Command* tersebut akan di respon oleh perangkat Bluetooth jika modul Bluetooth tidak dalam keadaan terkoneksi dengan perangkat lain (Eritha, 2014).

2.6.2 I2C

Sistem bus I2C ini hanya memerlukan dua jalur yaitu SCL dan SDA. Menurut Sulistiyono (2014), Karakter I2C adalah data dikirim serial secara per-bit, Menggunakan dua penghantar koneksi dengan *ground* bersama. Dua penghantar tersebut adalah SCL (*serial clock line*) untuk menghantarkan sinyal clock dan SDA (*serial data*) untuk mentransaksikan data, Jumlah *slave* maksimal 127. *Slave* dialamatkan melalui 7-bit-alamat, Setiap transaksi data terjadi antara pengirim (*transmitter*) dan penerima (*receiver*) (Sulistiyono, 2014).

Menurut Sulistiyono (2014), Komunikasi I2C memiliki 3 aturan agar komunikasi dapat berjalan dengan baik. Aturan pertama komunikasi I2C adalah *device* atau komponen yang mengirim data disebut transmitter, sedangkan *device* yang menerimanya disebut receiver. Aturan kedua *device* yang mengendalikan operasi transfer data disebut master, sedangkan *device* lainnya yang dikendalikan oleh master disebut *slave*. Aturan ketiga master harus menghasilkan serial clock melalui pin SCL, mengendalikan akses ke bus *serial*, dan menghasilkan sinyal kendali *start* dan *stop*. Komunikasi ini diatur secara *synchronous* menggunakan *clock* sinyal SCL kemudian disimpan pada RAM dengan type *byte by byte* komunikasi ini menggunakan *parting data* yang diolah oleh *integrated circuit*. Proses penyimpanan data ini dapat mengakibatkan *leak memory* jika tidak tepat untuk melakukan penyimpanan pada RAM Karena untuk file yang tidak dapat di hilangkan atau masih tetap melekat pada memori akan mengakibatkan program terhenti atau *hang* (Lake, 2010).

BAB 3. METODOLOGI PENELITIAN

Penelitian dengan judul “Implementasi Sistem SCADA pada Generator Penguatan Terpisah Menggunakan Arduino Mega 2560” ini dilaksanakan di Laboratorium Konversi Energi Listrik Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember, Jl. Slamet Riyadi No. 62 Patrang, Jember 68111. Untuk memperoleh data dan hasil dari penelitian yang sesuai dengan tujuan, maka perlu dilakukan beberapa tahapan-tahapan penelitian yang dijelaskan lebih rinci pada subbab-subbab dalam bab ini.

3.1 Alat dan Bahan

Penelitian ini membutuhkan dukungan beberapa alat dan bahan agar mendapatkan hasil yang sesuai dengan yang diharapkan. Alat dan bahan yang digunakan dalam penelitian ini diuraikan dalam Tabel 3.1.

Tabel 3.1 Daftar peralatan yang digunakan

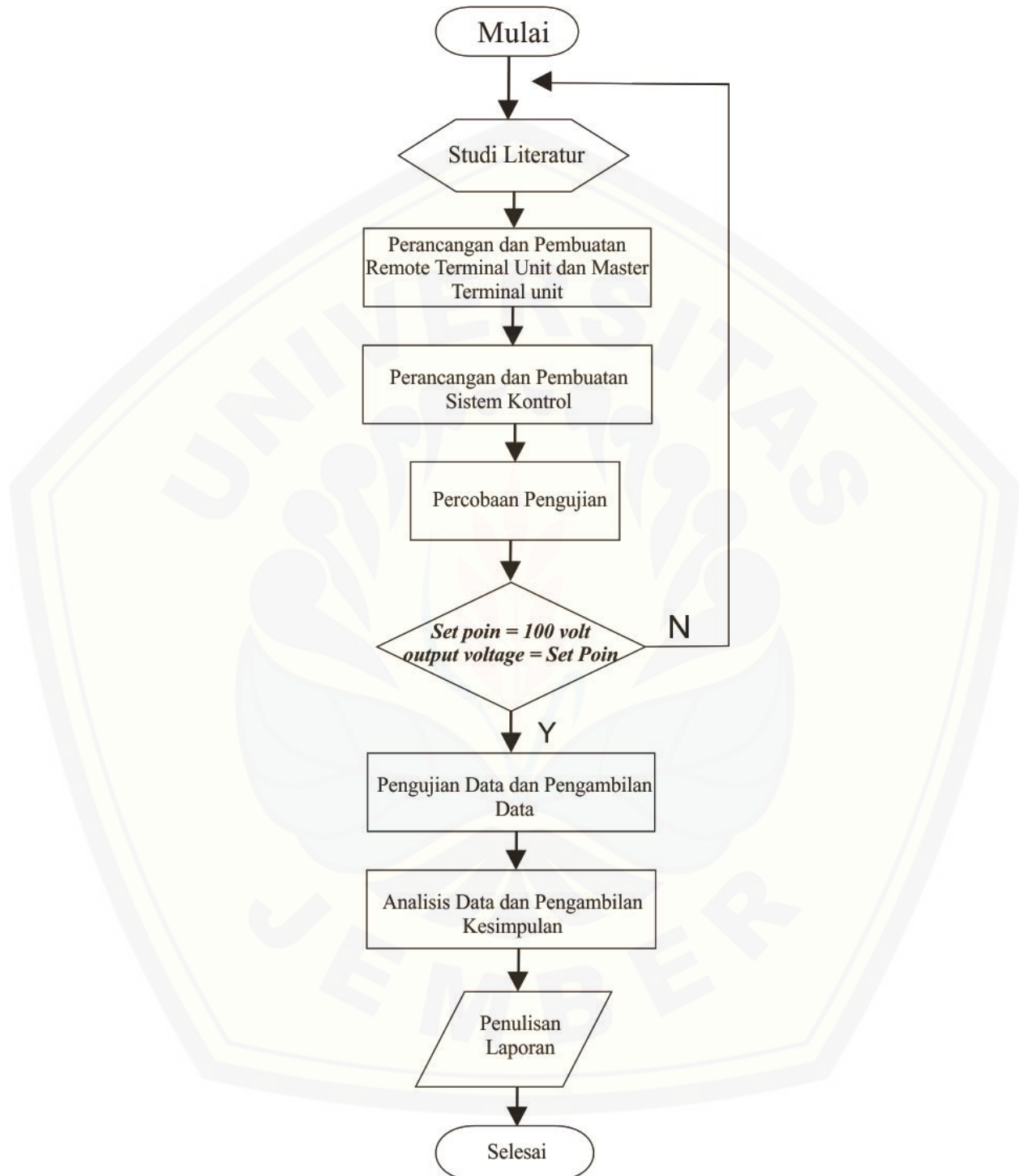
No	Nama Alat	Spesifikasi Teknis dan Penjelasan
1.	Digital Multimeter	Sanwa CD751 3200 Count/Bar Graph DC-1000 V AC-750 V MAX (CAT.II) DC-AC 600 V MAX(CAT.III)
2.	DC Power Supply	Protek DF1730B5A Single <i>Output</i> Power Supply 0 ~ 30V, 0 ~ 5A, <i>Output</i> terisolasi untuk pengoperasian seri atau parallel, Tampilan baca LED 3 digit
3.	Arduino UNO	ARDUINO UNO R3. ATmega328-AU microcontroller, Input voltage - 7-12V, 5V Electric current: 500MA, 3.3V Electric current: 50MA, 14 Digital I/O Pins (6 PWM <i>outputs</i>), 8 Analog Inputs, 32 KB Flash Memory, 16Mhz Clock Speed, 2 KB SRAM
4	Arduino Mega 2560	Arduino Mega 2560, ATmega16u2 mikrokontroller, input voltage Input voltage - 7-12V, 5V Electric current: 500MA, 3.3V Electric current: 50MA, 14 Digital I/O Pins (15 PWM <i>outputs</i>), 39 Analog Inputs, 32k Flash Memory, 16Mhz Clock Speed, 8 KB SRAM
5.	<i>Relay</i>	<i>Modem relay</i> Songle 8 Channel

lanjutan Tabel 3.1

6.	Kontaktor	Mitsubishi model S-N12 20 A
7.	<i>Prime Over</i>	Motor DC kompon panjang
8.	Generator DC	Generator DC kompon Penguatan Terpisah
9.	<i>Belt</i>	Bandi Fan Belt B42340 9,5x865La FM-34
10.	Tachometer	Lutron DT 2234C Tachometer. 0.1 – 5 – 999.9 rpm. 1-1000-99,999 rpm.
11.	AVR	Model: TDGC-2/0,5; INPUT 110/220 Volt; <i>OUTPUT</i> 0-250 Volt; 20-60 Hz ; I _{max} 8A ; CAP. 2KVA
12.	Sensor Tegangan	Terdiri dari 4 buah resistor yang dirangkai secara seri 3 buah resistor 100 k Ω dan 1 buah resistor 4k7 Ω
13.	Sensor Arus	Modul Sensor IC ACS 712
14.	<i>Buck converter</i>	Menggunakan IGBT SSH40N50, 2 buah kapasitor 470 μ F, Inductor 22.0 mH, Resistor sebesar 914 Ω
15.	<i>Rectifier</i>	Mnggunakan 4 buah diode Masig-masing dengan tipe V06 6A10
16.	Lampu Pijar	Phillip 1 buah lampu pijar 15 Watt, 2 buah lampu pijar 25 Watt
17.	Personal	
18.	Komputer/Laptop <i>Modem Bluetooth</i>	Lenovo G405s windows 10 <i>modem Bluetooth HC-05</i>
19.	Peralatan tambahan	<i>Jumper, bus bar, Obeng, Kunci dan lain-lain</i>

3.2 Tahapan Penelitian

Langkah kerja dalam penelitian ini terbagi menjadi beberapa tahap. Tahapan-tahapan tersebut digambarkan dalam diagram alir seperti yang terlihat dalam Gambar 3.1.



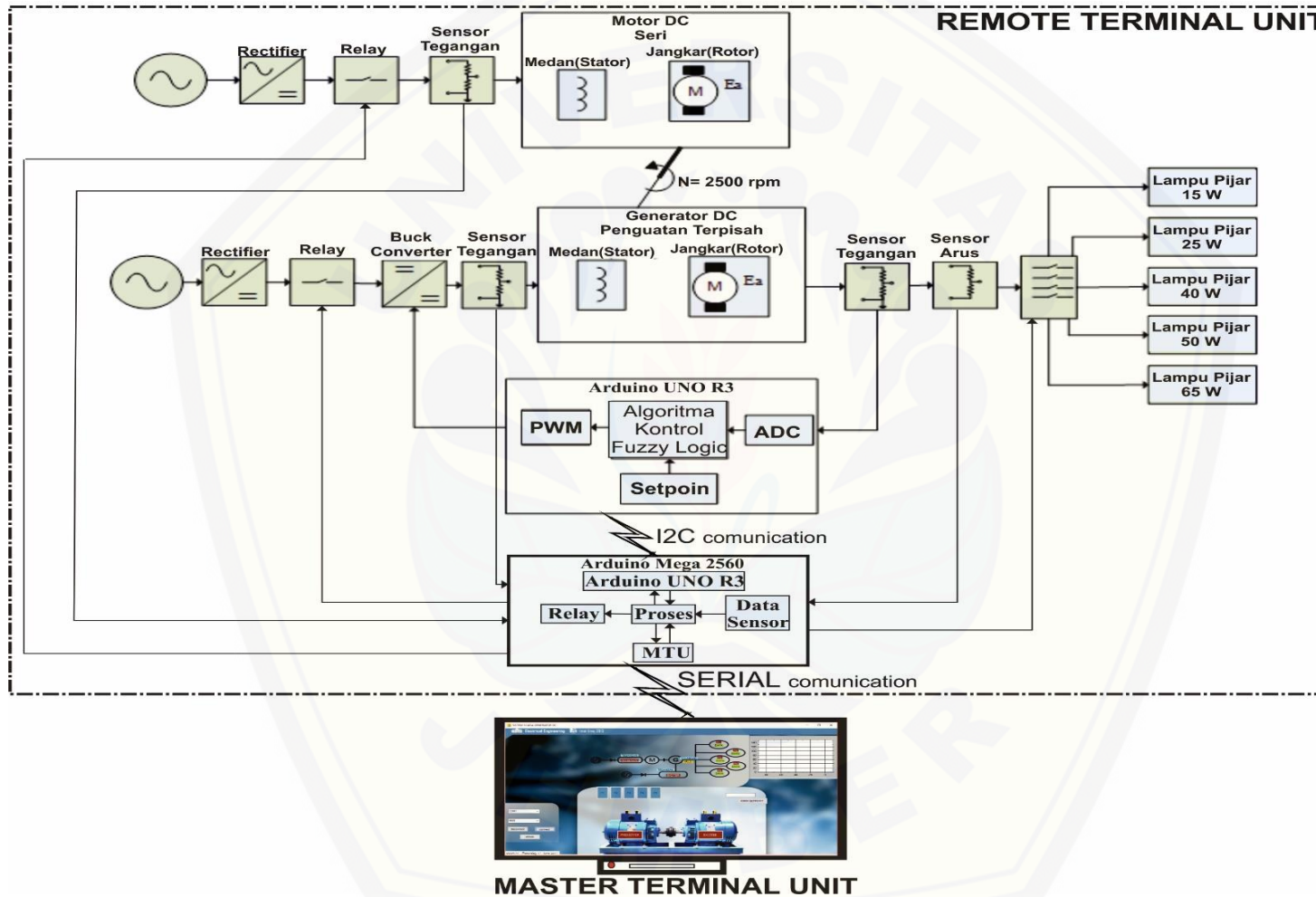
Gambar 3.1 Diagram alur penelitian

Penjelasan dari diagram alir dalam Gambar 3.1 adalah sebagai berikut:

- a. Studi literatur adalah kegiatan untuk mendapatkan bahan bacaan yang diperlukan untuk mendukung kegiatan penelitian ini, termasuk didalamnya literature spesifik yang berhubungan dengan komponen dan peralatan yang digunakan.
- b. Perancangan RTU (*remote terminal unit*) dan MTU (*master terminal unit*) dilakukan untuk merancang RTU yang digunakan untuk mengontrol *output* generator, memonitor dan mengontrol pembebanan. Sedangkan MTU digunakan sebagai pusat kendali dan *database*.
- c. Kegiatan perancangan dan pembuatan sistem kontrol adalah untuk merancang sistem *fuzzy logic* yang berfungsi untuk mengatur *duty cycle* dari *buck converter*.
- d. Percobaan pengujian adalah untuk mendapatkan sebuah sistem yang dapat beroperasi dengan *setpoint* tegangan sebesar 100 volt. Dalam kegiatan ini dilakukan pengujian sensor tegangan dan arus, pengujian *buck converter*, *relay*, dan pengujian pembebanan generator.
- e. Pengujian dan pengambilan data adalah menguji kerja sistem secara keseluruhan dengan mengambil data hasil pengujian sensor, *buck converter*, *relay*, dan hasil pengujian pembebanan generator pada kondisi tanpa beban dan berbeban dengan variasi beban lampu pijar.
- f. Kegiatan analisis data dan pengambilan kesimpulan dilakukan untuk menganalisis hasil pengujian sistem dan mengambil kesimpulan apakah sistem yang dirancang telah bekerja sesuai rancangan yang diharapkan.
- g. Penulisan laporan dilakukan setelah semua kerja sistem diuji dan dianalisis.

3.3 Blok Diagram dan *Flowchart* Sistem SCADA

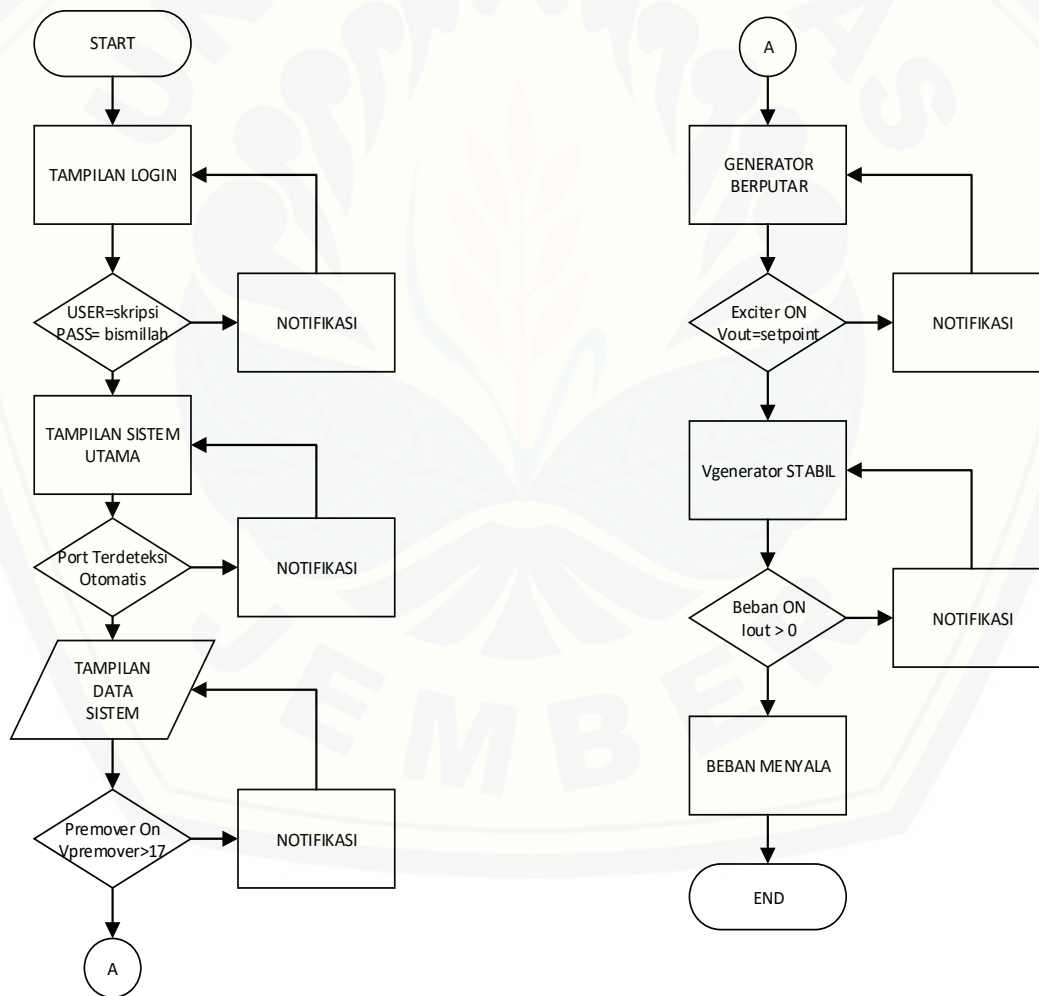
Blok diagram SCADA yang dirancang dalam penelitian ini sebagaimana ditunjukkan dalam Gambar 3.2. Sedangkan *flowchart* yang menjelaskan Blok diagram dalam Gambar 3.2 adalah sebagaimana terlihat dalam Gambar 3.3.



Gambar 3.2 Blok diagram SCADA

MTU berfungsi untuk mengontrol dan menyimpan data dari RTU. MTU terdiri dari HMI (*human machine interface*), sistem komunikasi dan *database*. HMI berfungsi sebagai antarmuka antara pengguna dengan sistem. Sedangkan *database* berisi data-data yang dikirim dari RTU. RTU berfungsi sebagai pengontrol pembebanan, pengontrol generator dan *exciter*.

RTU terdiri dari dua mikrokontroler yang dibangun dari Arduino UNO dan Arduino Mega 2560. Arduino UNO digunakan untuk mengontrol *output* generator dengan kontrol *fuzzy logic*, sedangkan Arduino Mega 2560 sebagai pusat kontrol dan komunikasi RTU.



Gambar 3.3 Flowchart sistem SCADA

HMI pada MTU dilengkapi dengan sistem *login* untuk membatasi akses pengguna. Komunikasi dari MTU ke RTU dilakukan secara nirkabel melalui Bluetooth.

Untuk menyalakan generator MTU memberi perintah untuk menyalakan *prime mover* yang diangun dari Motor DC seri. Sumber penggerak *prime mover* dalam penelitian ini sumber listrik AC yang disearahkan oleh *rectifier*. Untuk dapat dinyalakan dan dimatikan dari MTU, *prime mover* ini dilengkapi dengan kontaktor sehingga proses kendali dapat dilakukan dengan menyalakan dan mematikan *relay* pada *coil* kontaktor. Informasi status operasi dari *prime mover* didapatkan dari sensor tegangan yang di tempatkan pada *input prime mover*. *prime mover* dianggap bekerja jika sensor tegangan membaca tegangan lebih dari 17 V.

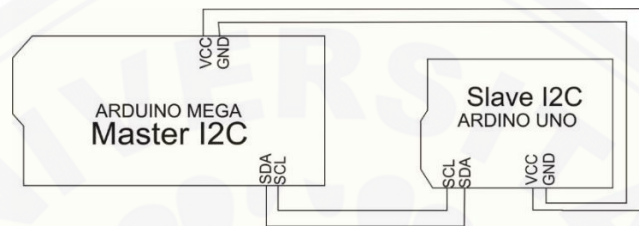
Setelah *prime mover* bekerja maka sistem akan memerintahkan agar RTU melakukan pengaturan *exciter*. Untuk mengatur *exciter* MTU memberikan *setpoint* yang dimasukkan melalui HMI sehingga *exciter* bekerja untuk menghasilkan tegangan *output* generator yang sesuai dengan *setpoint*. Pengaturan tegangan *output* ini dilakukan dengan *fuzzy logic*.

3.4 Perancangan Komunikasi SCADA

Komunikasi SCADA pada sistem ini menggunakan dua macam sistem komunikasi, yaitu I2C dan komunikasi serial dengan media Bluetooth. Sistem komunikasi I2C adalah komunikasi yang menghubungkan antara Arduino Mega 2560 dengan Arduino UNO. Pemakain dua arduino karena kebutuhan memori yang tidak dapat dipenuhi oleh satu Arduino saja. Arduino Mega 2560 memiliki SRAM sebesar 8 KB, sedangkan pada Arduino UNO memiliki memori SRAM sebesar 2 KB. Penggunaan Arduino UNO hanya untuk membantu kerja dari Arduino Mega 2560 sebagai pusat kontrol dan komunikasi pada RTU. Komunikasi ini dilakukan untuk mengirim nilai tegangan *output* generator dari Arduino UNO ke Arduino Mega 2560

dan mengirim data *setpoint* untuk kontrol *fuzzy logic* dari Arduino Mega 2560 ke Arduino UNO dengan menggunakan SDA (serial data), SCL (serial clock) dan *ground*.

Pada Arduino UNO, SDA terletak pada pin analog 4 dan SCL pada pin analog 5, sedangkan pada Arduino Mega 2560 SDA terletak pada pin 20 dan SCL pada pin 21. Rangkaian dari komunikasi I2C dapat dilihat pada Gambar 3.4.



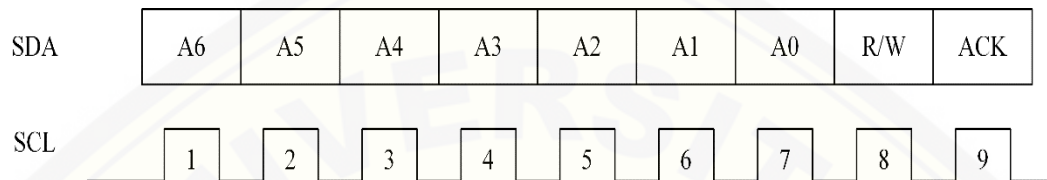
Gambar 3.4 Rangkaian I2C

Pengiriman data dengan sistem I2C memerlukan *library wire.h* untuk mendapatkan *function* yang mempermudah ketika melakukan komunikasi. Beberapa *function* yang didapatkan adalah:

- `begin()` = menginisialisasi protokol I2C pada Arduino
- `requestFrom()` = meminta data byte dari arduino yang mengambil data
- `beginTransaction()` = memulai transmisi data ke Arduino UNO
- `endTransmission()` = mengakhiri transmisi data
- `write()` = mengirim data serial
- `read()` = membaca data serial I2C
- `available()` = mengecek kesiapan *device* I2C
- `onRieceived()` = memanggil *event handler* tertentu saat data masuk
- `onRequest()` = memanggil *event handler* tertentu saat ada permintaan data

Untuk dapat melakukan pembacaan dan pengiriman data tegangan *output* generator dari Arduino UNO ke Arduino Mega 2560 *output* data sensor tegangan ditempatkan pada pin A0 pada Arduino UNO. Pin ini membaca nilai ADC dan

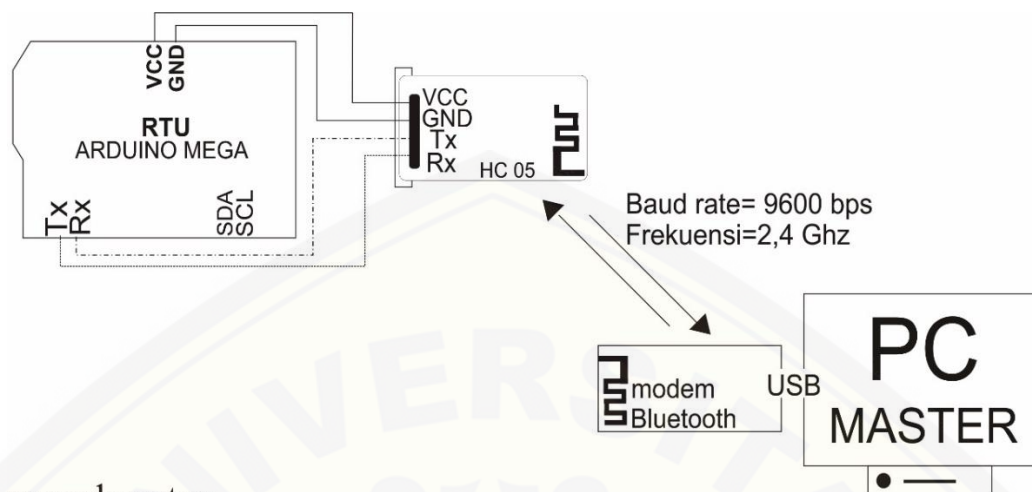
mengkonversi nilai analog dari sensor menjadi nilai digital. Nilai digital kemudian diubah menjadi nilai desimal oleh Arduino UNO. Untuk dikirim ke Arduino Mega 2560 data desimal ini dirubah menjadi data bit dengan register seperti terlihat dalam Gambar 3.5.



Gamabr 3.5 Model pengalamatan I2C

Data dikirim secara terpisah dan setiap data menggunakan mode pengalamatan terdiri dari 7 bit + 1 bit R/W + 1 bit ACK. Pengiriman data diawali dengan MSB (*Most Significant Bit*) yang berisi data nilai tegangan dalam bentuk bit dan diakhiri dengan LSB (*Least Significant Bit*). Setiap data ini juga dilengkapi dengan 1 bit ACK (*Acknowledge*) yang berguna untuk mengetahui bahwa data sudah diterima. Data LSB yang terdiri dari 1 bit R/W akan bernilai 1 untuk membaca data dan 0 untuk menulis data.

Data pembacaan tegangan *output* yang diterima oleh Arduino Mega 2560 kemudian dikelompokkan dengan data dari sensor arus, sensor tegangan *prime mover*, dan sensor tegangan *exciter*. Semua data ini kemudian dikirim secara bersamaan ke MTU melalui komunikasi serial. Sistem pengalamatan data yang sudah dikelompokkan ini seperti terlihat dalam Gambar 3.6.



pengalaman

start bit = 1 bit
 stop bit = 0 bit
 separator = |

to PC: 1|SENSOR Iout| SENSOR Vpremove| SENSOR Vexciter| SENSOR Vout|0
 to Arduino: 1|NUMBER|0

Gambar 3.6 Rangkaian dan pengalaman komunikasi serial

Data yang dikirim dari RTU ke MTU kemudian dikonversi menjadi data biner sesuai dengan sistem pengalaman komunikasi serial. Data ini kemudian diubah menjadi nilai desimal dan ditampilkan ke HMI. Sebaliknya data yang dikirim dari MTU ke RTU dilakukan dengan cara yang sama untuk nantinya diolah dalam bentuk desimal, misalnya ketika diperlukan proses *fuzzy logic* di RTU. Perintah dan data-data yang dikomunikasikan secara serial dari RTU ke MTU dan sebaliknya sebagaimana terlihat dalam Tabel 3.2.

Tabel 3.2 Data yang melalui komunikasi serial

Data Dikirim RTU	Data Dikirim MTU
Sensor Arus <i>Output</i> Generator	Menyalakan/Mematikan Generator
Sensor Tegangan <i>prime mover</i>	Menyalakan/Mematikan <i>Exciter</i>
Sensor Tegangan <i>Exciter</i>	Menyalakan/Mematikan P1
Sensor Tegangan <i>Output</i> Generator	Menyalakan/ Mematikan P2
Separator	Menyalakan/ Mematikan P3
	Menyalakan/Mematikan P4
	Menyalakan/ Mematikan P5
	Setpoint
	Separator

Komunikasi serial menggunakan kecepatan akses data 9600 bps (*bit per second*). Sebenarnya kecepatan maksimum dari komunikasi Bluetooth ini sebesar 1 Mbps, namun komunikasi ini dirancang dengan kecepatan 9600 bps untuk mengurangi data *error* dan memudahkan *monitoring*. Komunikasi ini menggunakan tipe *synchronous* dengan frekuensi kerja 2,4 GHz. Untuk mengoptimisasi pengiriman digunakan modulasi FSK (*frekuensi shift keying*) dengan *filter* spektrum gelombang metode Gaussian.

Pengalamatan komunikasi serial ini dimulai dengan *start bit* dengan nilai 1 dan diakhiri dengan *stop bit* dengan nilai 0. *Stop bit* menandai penutup data yang dikirimkan.

Dalam sistem ini HMI dan database dibangun dengan Visual Basic sedangkan proses perhitungan di RTU, sebagai contoh untuk membangun *fuzzy logic* digunakan Bahasa C. untuk menghindari kerancuan proses maka dalam program digunakan separator menggunakan (|).

RTU menggunakan modem Bluetooth HC-05 dengan frekuensi 2,4 GHz. Konfigurasi Tx dan Rx dibalik agar Arduino dapat mengolah, mengirim dan menerima

data dari komputer MTU. Modem ini memiliki fitur keamanan standar *authentication* dan *encryption*. Artinya modul ini memiliki keamanan yang baik dengan pin standar 12345. Modem ini bekerja dengan catu daya 3,6 V - 6 V, dengan arus saat *unpaired* sekitar 30 mA, dan saat *paired* (terhubung) sebesar 10 mA. Jarak efektif modem ini sejauh 10 Meter. Beberapa *function* yang digunakan pada modem Bluetooth HC-05 sebagai berikut:

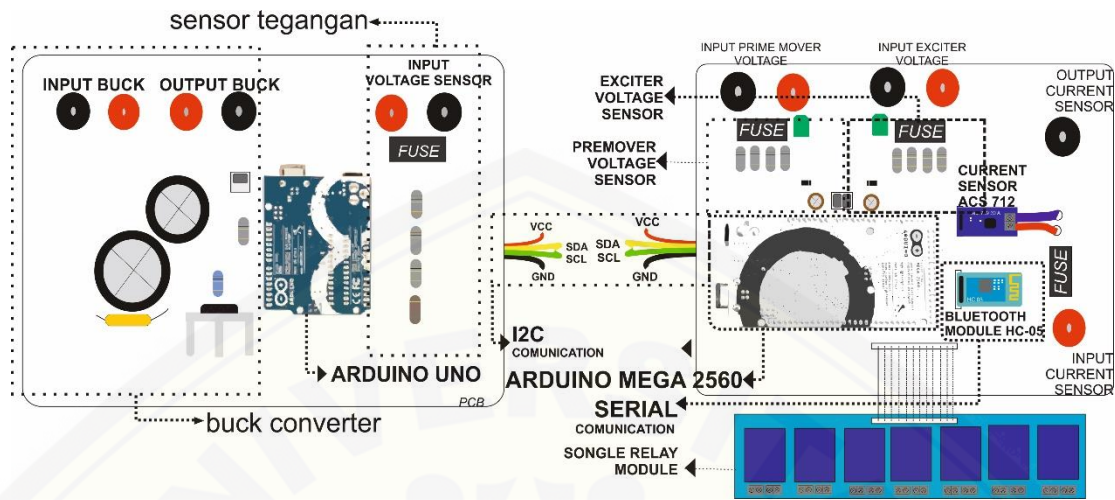
- `Serial.print` = Mengirim data serial ke MTU
- `Char` = Proses pemisahan data

MTU menggunakan modem Bluetooth 4.0 USB adapter. Modem Bluetooth ini bekerja pada frekuensi kerja 2,40 - 2,48 GHz dengan jangkauan 20 Meter – 50 Meter dengan catu daya 5 V. Beberapa *function* yang digunakan pada *modem* Bluetooth 4.0 adalah:

- `Buffer` = Proses pembacaan data
- `Split` = Proses pemisahan separator
- `Writeline` = Perintah mengirim data

3.5 Perancangan RTU

RTU (*remote terminal unit*) dibangun dengan dua buah Arduino yaitu sebuah Arduino UNO dan sebuah Arduino Mega 2560. Masing-masing kontroler ini terhubung dengan peralatan luar dan berkomunikasi antar keduanya dengan skema seperti terlihat dalam Gambar 3.7

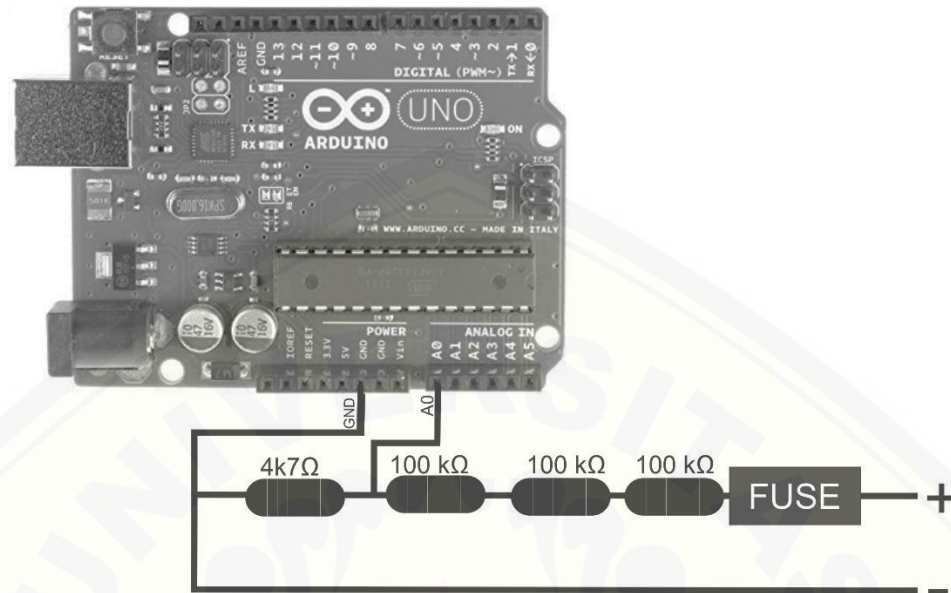


Gambar 3.7 Pemodelan RTU pada PCB

Arduino UNO terhubung dengan sensor tegangan *output generator* dan *input switching buck converter*. Arduino Mega 2560 terhubung dengan sensor tegangan *input prime mover*, tegangan *input exciter*, dan sensor arus. Dua buah controller ini berkomunikasi dengan komunikasi I2C.

3.5.1 Perancangan dan Pengujian Sensor Tegangan *Output Generator*

Sensor tegangan dirancang dari rangkaian pembagi tegangan. Perancangan sensor tegangan dilakukan dengan menyusun resistor secara seri dari dua buah resistor $300\text{ k}\Omega$ ($3 \times 100\text{ k}\Omega$) dan resistor $4\text{ k}7\ \Omega$ untuk dapat membaca tegangan $0 - 324.15\text{ V}$. Rangkaian sensor tegangan seperti terlihat dalam Gambar 3.8, sedangkan perhitungan rangkaian ini sebagaimana terlihat dalam Persamaan (3.1).



Gambar 3.8 Skematik sensor tegangan

Perhitungan dari perancangan sensor tegangan ini dapat dilihat seperti pada persamaan 3.1

$$V_{out} = \frac{V_{in} \times R_2}{(R_1 + R_2)} \quad (3.1)$$

dimana, $V_{out} = 5$; $V_{in} = 324.15$; $R_1 = 300 \text{ k}\Omega$

$$5 = \frac{324.15 \times R_2}{(300k + R_2)}$$

$$5 \times (300k + R_2) = 324.15 \times R_2$$

$$1500k + 5R_2 = 324.15R_2$$

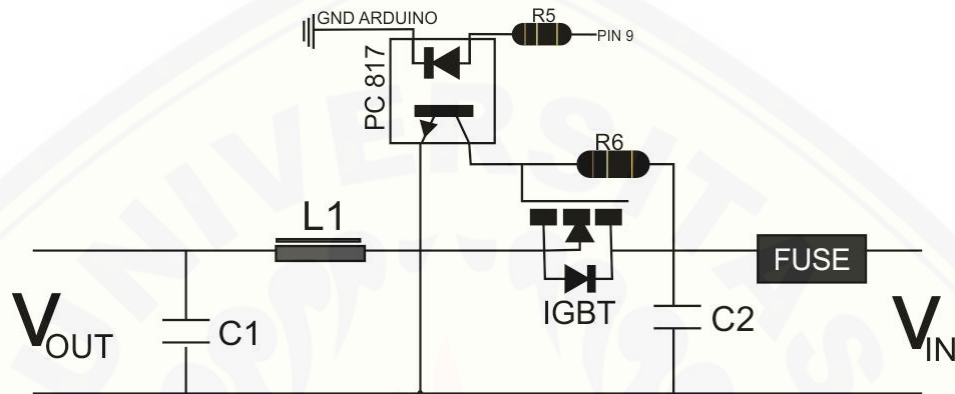
$$R_2 = \frac{1500k}{319.15}$$

$$R_2 = 4k7 \Omega$$

Untuk memvalidasi hasil perancangan ini, pada pengujian hasil pembacaan sensor ini dibandingkan dengan pembacaan multimeter Sanwa CD751. Kalibrasi untuk mendapatkan hasil yang sesuai dengan pembacaan multimeter ini dilakukan dengan memberikan nilai error dalam program.

3.5.2 Perancangan dan Pengujian *Buck Converter*

Buck converter berfungsi untuk mengatur besarnya tegangan eksitasi yang diberikan pada generator DC. Rangkaian *buck converter* dapat dilihat pada Gambar 3.9.



Gambar 3.9 Rangkaian *buck converter*

Buck converter menggunakan IGBT SSH40N50 dengan V_{CE} 100 volt. Untuk menghubungkan *gate* dengan *output* Arduino yang berfungsi sebagai tegangan pemuncian digunakan sebuah *switch Optocoupler*. Nilai kapasitor C1 dan C2 adalah 470 μ F dengan tegangan kerja 480 V. Induktor L1 sebesar 22.0 mH dan resistansi kumparan *exciter* terukur sebesar 914 Ω . Agar *buck converter* bekerja dengan sistem CCM (*countinuous current mode*) maka dilakukan perhitungan frekuensi pemuncian seperti dalam Persamaan (3.2)

$$f > \frac{(1-D)R}{2L} \quad (3.2)$$

dimana, $R = 914 \Omega$, $L = 22 \text{ mH}$ dan $D = \text{duty cycle}$

$$f > \frac{(1-D)914}{2(22 \times 10^{-3})}$$

$$f > 20772.72 (1-D)$$

Pengujian *buck converter* ini dilakukan dengan melihat hubungan antara *duty cycle* dan *output* tegangan yang dihasilkan sekaligus mengukur efisiensi pada setiap nilai *duty cycle* yang digunakan. Pengujian ini dilakukan dengan menempatkan resistor

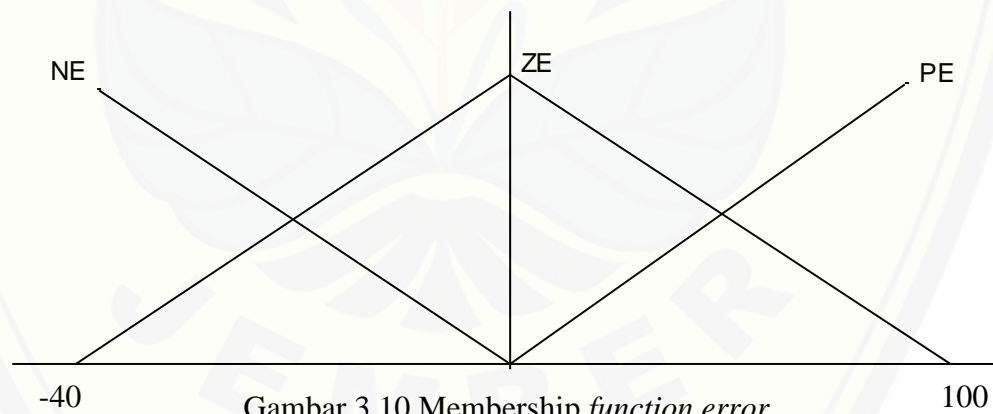
sebesar 1 k Ω sebagai beban. Nilai *duty cycle* yang diterapkan bervariasi dengan kelipatan 10 %.

3.5.3 Perancangan dan Pengujian Kontrol *Fuzzy*

Kontrol *fuzzy logic* untuk mengatur sistem eksitasi pada generator DC penguatan terpisah diprogramkan pada Arduino UNO sehingga menjadi sebuah *embedded system*. Pemrograman *fuzzy logic* terdiri dari tiga tahap yaitu fuzzyfikasi, inferensi, dan defuzzyfikasi.

a. Fuzzyfikasi

Pada tahap ini dilakukan pembuatan fungsi keanggotaan *error* dan *delta error*. Kemudian dilakukan penentuan derajat keanggotaan *error* (e) dan *delta error* (de) tersebut. Nilai *setpoint* tegangan *output* digunakan sebagai nilai maksimum dari fungsi keanggotaan *error*. Nilai minimum *error* dipilih (-40). Nilai ini didapatkan dari percobaan. Fungsi keanggotaan *error* seperti pada Gambar 3.10.



Nilai *error* diperoleh dari nilai *setpoint* dikurangi dengan tegangan *output* dari generator yang berasal dari pembacaan sensor tegangan. Nilai maksimum yang mungkin dari *output* generator adalah 140 V sehingga nilai *error* minimum dalam *membership function error* dipilih sebesar -40. Persamaan *membership function error* adalah sebagai berikut:

$$\text{Jika } -40 < E < 0$$

$$NE_{-} = \frac{-E}{40}$$

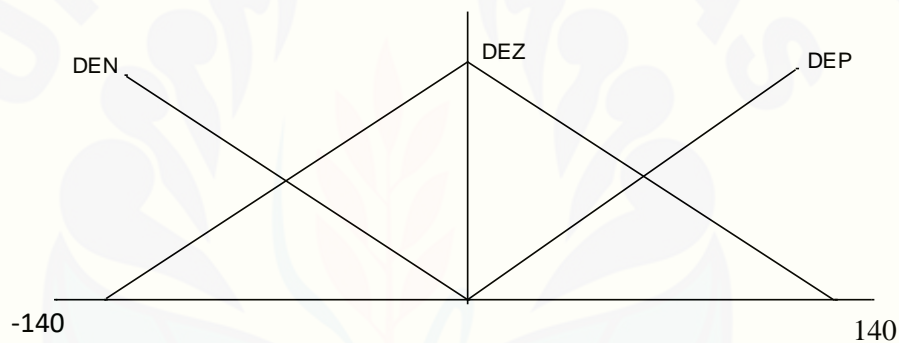
$$ZE = \frac{E+40}{40}$$

Jika $0 < E < 100$

$$ZE_{-} = \frac{100-E}{100}$$

$$PE = \frac{-E}{100}$$

Membership function delta error dibuat berdasarkan kemungkinan *delta error* minimum dan *delta error* maksimum. Fungsi keanggotaan *delta error* seperti terlihat dalam Gambar 3.11.



Gambar 3.11 *Membership function delta error*

Persamaan *delta error* sebagai berikut

Jika $-140 < E < 0$

$$DEN_{-} = \frac{-DE}{140}$$

$$DEZ = \frac{DE+140}{140}$$

Jika $0 < E < 140$

$$DEZ_{-} = \frac{140-DE}{140}$$

$$DEP = \frac{DE}{140}$$

b. Inferensi

Inferensi dimulai dengan memetakan nilai derajat keanggotaan dari masing-masing *error* dan *delta error* ke sebuah tabel yang berisi basis aturan yang dipilih (*rule base*) dan menggunakan metode Mamdani untuk pemilihan nilai dari derajat keanggotaan. Dengan demikian terlebih dahulu harus membuat matrik atau tabel *rule base* seperti pada Gambar 3.12.

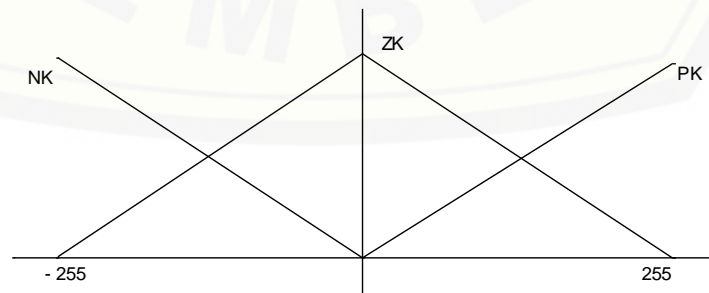
	E			
DE		PE	ZE	NE
DEN	U1 NK	U2 NK	U3 PK	
DEZ	U4 NK	U5 ZK	U6 PK	
DEP	U7 NK	U7 PK	U9 PK	

Gambar 3.12 *Rule base*

Setelah langkah ini maka proses inferensi selesai dan untuk selanjutnya adalah langkah defuzzyfikasi untuk menentukan besarnya nilai *duty cycle* yang dikirim ke *buck converter*.

c. Defuzzyfikasi

Defuzzyfikasi adalah proses memetakan kembali nilai-nilai *fuzzy* yang didapat pada proses fuzzyfikasi dan inferensi ke dalam himpunan fungsi defuzzyfikasi seperti terlihat dalam Gambar 3.13. Berbeda dengan fuzzyfikasi, proses defuzzyfikasi akan menghasilkan luasan-luasan trapesium.

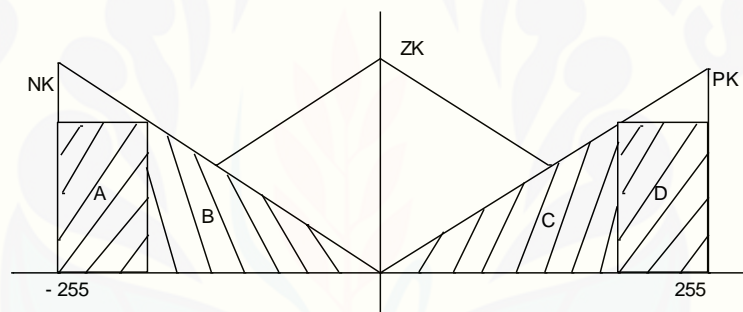


Gambar 3.13 Himpunan defuzzyfikasi

Penentuan nilai kontrol (c) ditentukan dengan perhitungan titik berat seluruh luasan hasil pemetaan, atau diistilahkan dengan *weighted average method*. Secara matematik, titik berat ini dihitung sebagai berikut:

$$Z = \frac{\sum_{i=1}^9 \alpha_i * \alpha w_i}{\sum_{i=1}^9 \alpha_i} \quad (3.3)$$

$$Z = \frac{WANK1 \times U1 + WANK2 \times U2 + WAPK3 \times U3 + WANK4 \times U4 + WAZK \times U5 + WAPK6 \times U6 + WANK7 \times U7 + WAPK8 \times U8 + WAPK9 \times U9}{U1 + U2 + U3 + U4 + U5 + U6 + U7 + U8 + U9}$$



Gambar 3.14 Perhitungan titik berat luasan

NK / Negatif Kontrol terletak pada bangun A dan B. Untuk perhitungannya seperti dibawah ini

- 1) $LA = ((1 - U) \times 255) \times U$
- 2) $XA = ((1 - U) \times (-255)) / 2 + (-255 \times U)$
- 3) $LB = 0,5 \times U \times (255 \times U)$
- 4) $XB = ((U \times -255) \times 2) / 3$

$$\text{Sehingga } WANK = \frac{LA \times XA + LB \times XB}{LA + LB}$$

PK / Positif Kontrol terletak pada bangun C dan D. Untuk perhitungannya seperti dibawah ini

- 1) $LC = 0,5 \times U \times (255 \times U)$
- 2) $XC = ((U \times 255) \times 2) / 3$

$$3) LD = ((1 - U) \times 255) \times U$$

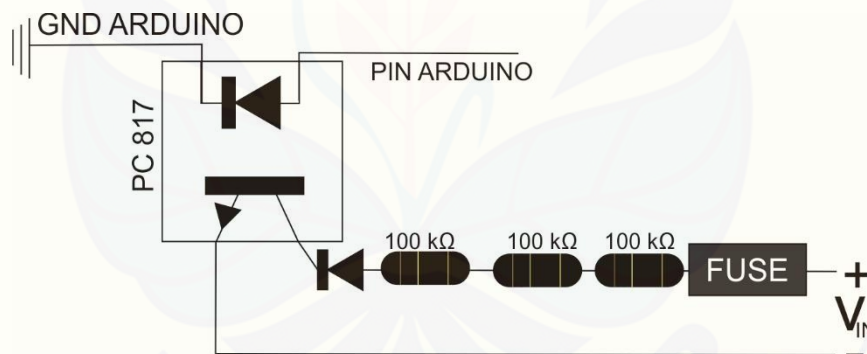
$$4) XD = ((1 - U) \times (-255))/2 + (-255 \times U)$$

$$\text{Sehingga } WAPK = \frac{LC \times XC + LD \times XD}{LC + LD}$$

Setelah semua tahap pembuatan *fuzzy* selesai maka dilakukan pengujian untuk pengambilan data. Pengujian dilakukan dengan memberikan nilai *setpoint* melalui HMI dan melihat tegangan *output*. Jika tegangan *output* yang dihasilkan mendekati atau sama dengan *setpoint* maka kontrol ini dianggap berhasil.

3.5.4 Perancangan dan Pengujian Sensor Tegangan *prime mover* dan *Exciter*

Dua sensor ini masing-masing menggunakan *ground* tersendiri. Untuk menghubungkan ke kaki Arduino digunakan *optocoupler* seperti terlihat dalam Gambar 3.15. *Output* sensor *prime mover* menggunakan pin A0 sedangkan sensor *exciter* menggunakan pin A1.



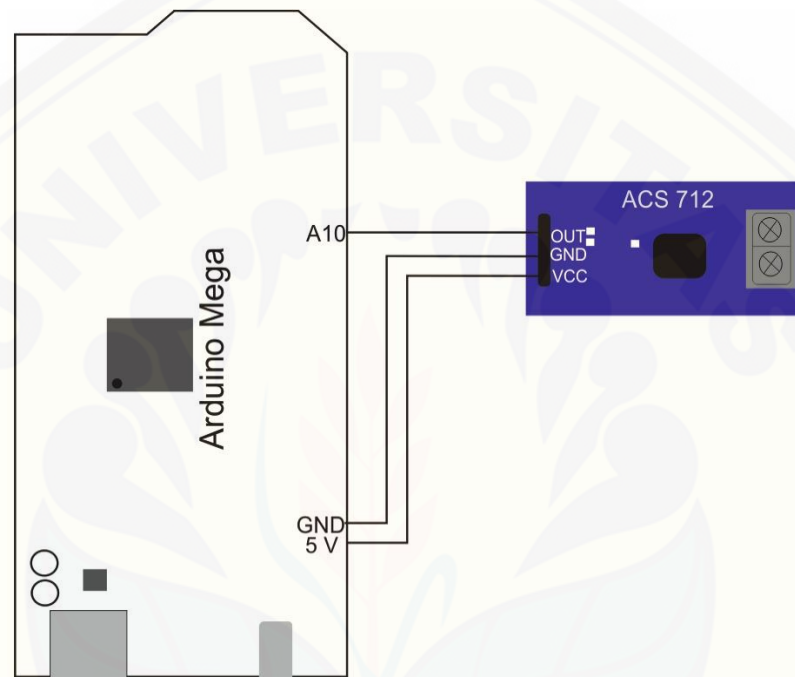
Gambar 3.15 Skematik sensor tegangan dengan optocoupler

Untuk mendapatkan pembacaan nilai yang akurat. Hasil pembacaan sensor ini dibandingkan dengan multimeter digital Sanwa CD751. Kalibrasi dilakukan dengan memberikan angka-angka pengali dalam program. Untuk mendapatkan angka pengali yang sesuai terlebih dahulu dilakukan pengambilan beberapa data dan menghitung nilai *error* seperti persamaan 3.4.

$$\text{Error \%} = \left| \frac{\text{pembacaan avometer} - \text{pembacaan sensor}}{\text{pembacaan avometer}} \right| \times 100\% \quad (3.4)$$

3.5.5 Perancangan dan Pengujian Sensor Arus

Untuk sensor arus digunakan modul ACS 712 yang memiliki kemampuan pembacaan arus sampai 30 A dengan tegangan kerja 1,2 KV seperti dalam Gambar 3.16.

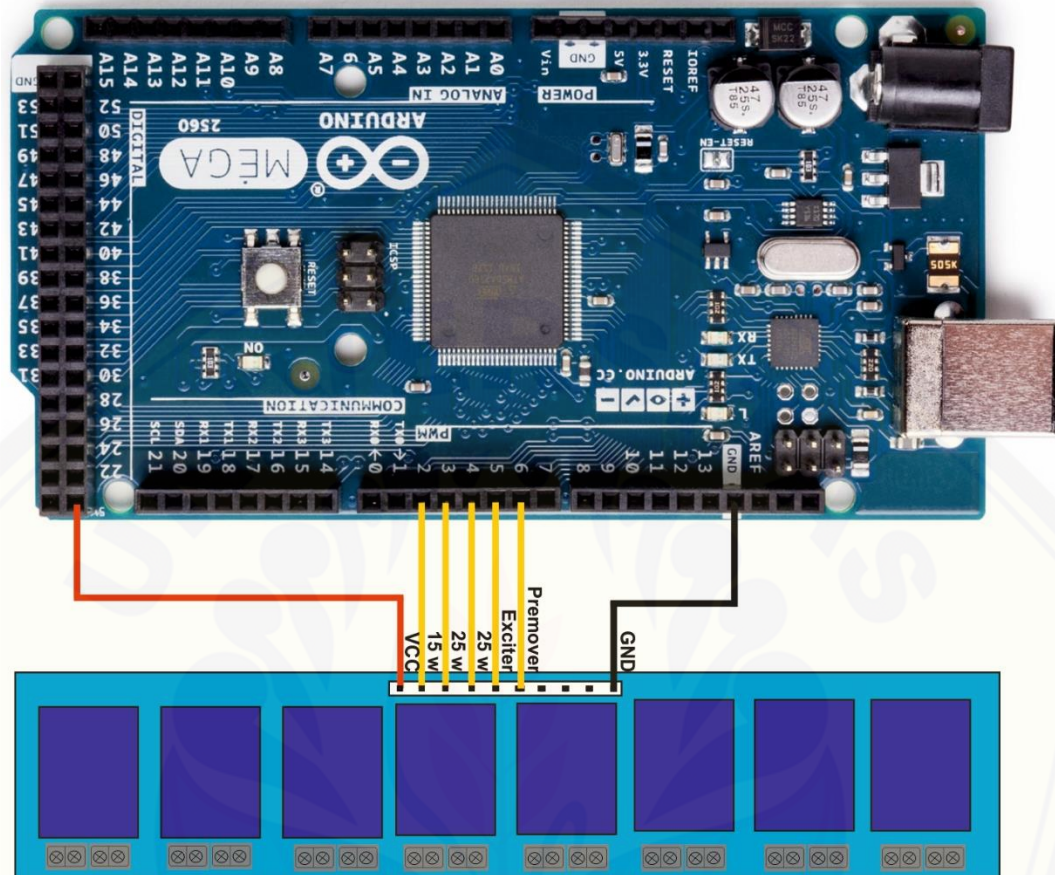


Gambar 3.16 Konfigurasi pin sensor arus

Output sensor ini adalah nilai analog yang dihubungkan paada pin A10 yang akan mengubah nilai analog menjadi digital. Hasil pembacaan sensor arus dibandingkan dengan multimeter Sanwa CD751 dan dikalibrasi didalam program sampai mendapatkan nilai *error* paling kecil.

3.5.6 Perancangan dan Pengujian Relay

Relay yang digunakan adalah modul *relay* Songle dengan tegangan *coil* 5 V. Relay ini memiliki proteksi Optocupler dan dioda pada outputnya. *Relay* terdiri dari 8 chanel *relay* dengan pin yang digunakan sebanyak 5 seperti pada Gambar 3.17. Semua pin ini aktif rendah.



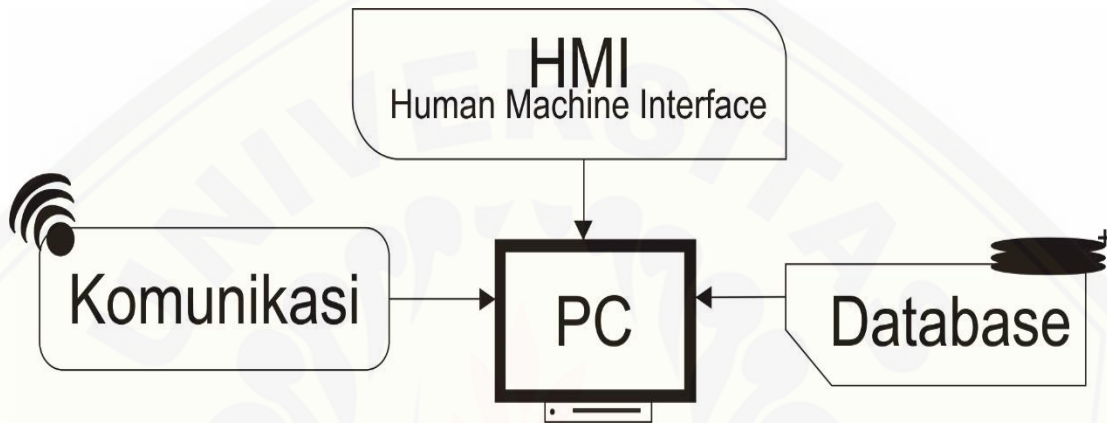
Gambar 3.17 Konfigurasi *relay*

Data serial yang dikirimkan untuk menyalakan relay adalah 100, 200, 300, 400, dan 500, sedangkan untuk mematikan digunakan nilai 101, 201, 301, 401, dan 501. Nilai 100 dan 101 untuk mengontrol *exciter*, nilai 200 dan 201 untuk mengontrol *prime mover*.

Untuk pembebanan, nilai 300 dan 301 pembebanan P1 sebesar 15 W, nilai 400 dan 401 untuk mengontrol beban P2 sebesar 25 W. Untuk mendapatkan pembebanan P3 sebesar 40 W digunakan dua relay dengan data serial 300, 400, 301 dan 401. Untuk pembebanan P4 sebesar 50 W digunakan data serial 400, 500, 401 dan 501. Untuk pembebanan P5 sebesar 65 W digunakan tiga relay dengan data serial 300, 400, 500, 301, 401, dan 50.

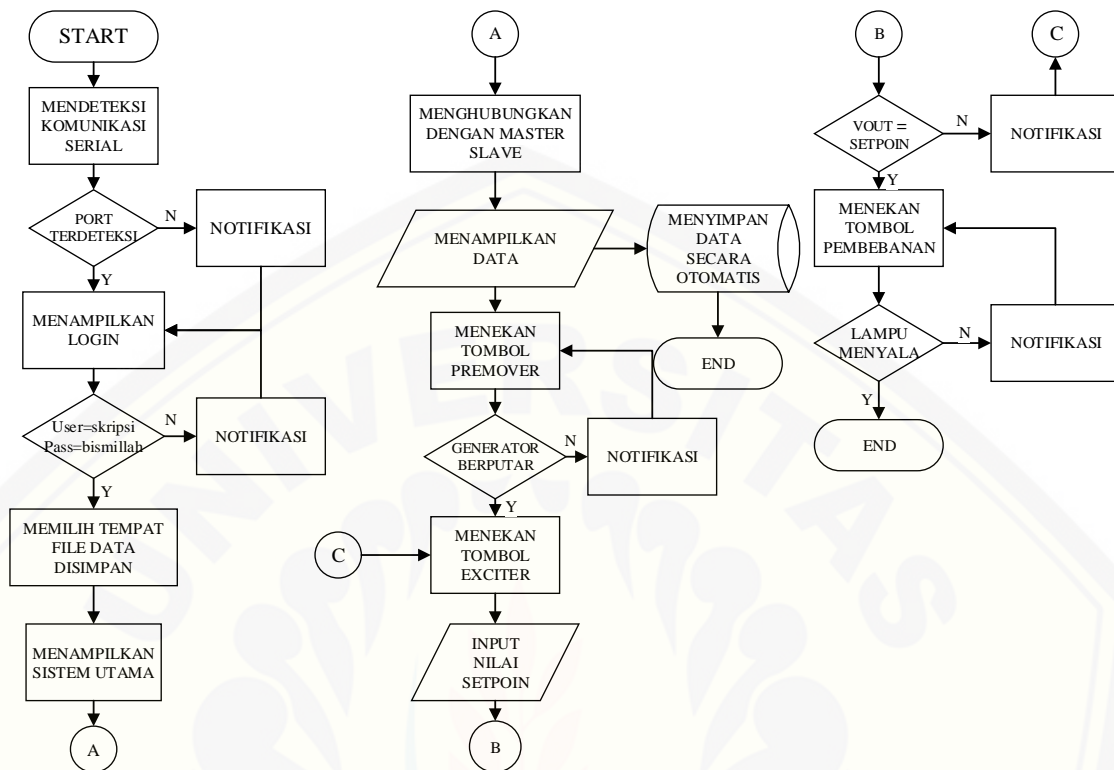
3.6 Perancangan MTU

MTU merupakan pengendali sistem yang meliputi pengaturan saklar dengan kontrol *boolean logic*, pemberian *setpoint* untuk RTU sebagai pengaturan tegangan *output* generator, serta *monitoring sistem* dan *database*. Blok diagram perancangan RTU seperti Gambar 3.18.



Gambar 3.18 Blok diagram MTU

Dalam penelitian ini MTU dirancang dengan menggunakan Bahasa pemrograman Visual Basic. Data yang ditampilkan di MTU tidak hanya nilai-nilai tetapi juga status atau indikator kondisi dari sistem yang dimonitor dan dikontrol pada RTU. Flowchart program MTU yang merupakan gabungan dari HMI dan database sebagaimana terlihat dalam Gambar 3.19.



Gambar 3.19 Flowchart program MTU

Sistem ini dirancang untuk dapat mendeteksi port secara otomatis. HMI menampilkan data *real time* dari sistem yang dikontrol dan dimonitor. Tombol-tombol HMI berfungsi untuk mengirim data perintah pada sistem yang terhubung dengan RTU.

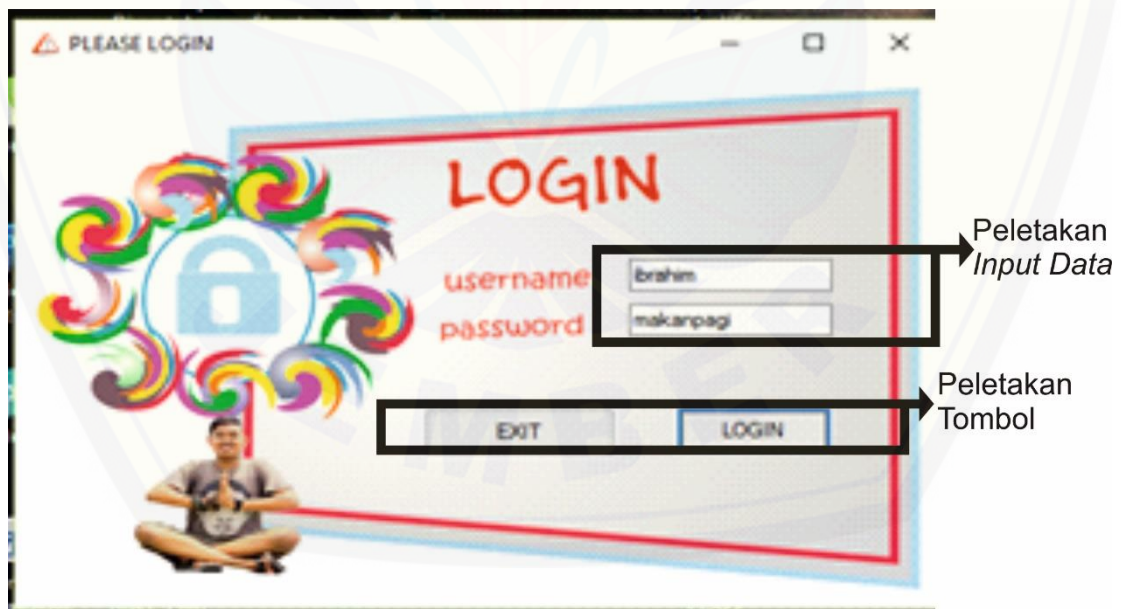
3.6.1 Perancangan HMI

HMI adalah jendela antarmuka yang menghubungkan operator (manusia) dengan sistem yang sedang dikontrol dan dimonitor secara *real time*. Jendela HMI terdiri dari dua bagian yaitu jendela sistem utama seperti yang terlihat pada Gambar 3.20 dan jendela sistem login seperti terlihat pada Gambar 3.21.



Gambar 3.20 *Human machine interface* sistem utama

Perancangan antarmuka ini dibuat sedemikian rupa untuk memudahkan operator dan memberikan tampilan yang mewakili sistem yang sedang dimonitor. Sebagai contoh ketika generator berputar maka pada jendela sistem utama akan muncul gambar generator dan *prime mover* yang status operasinya di tandai dengan perubahan warna.



Gambar 3.21 *Human machine interface* sistem login

Jendela login meminta operator memasukkan username dan password. Jendela login ini membatasi akses pengguna untuk keamanan ke sistem utama.

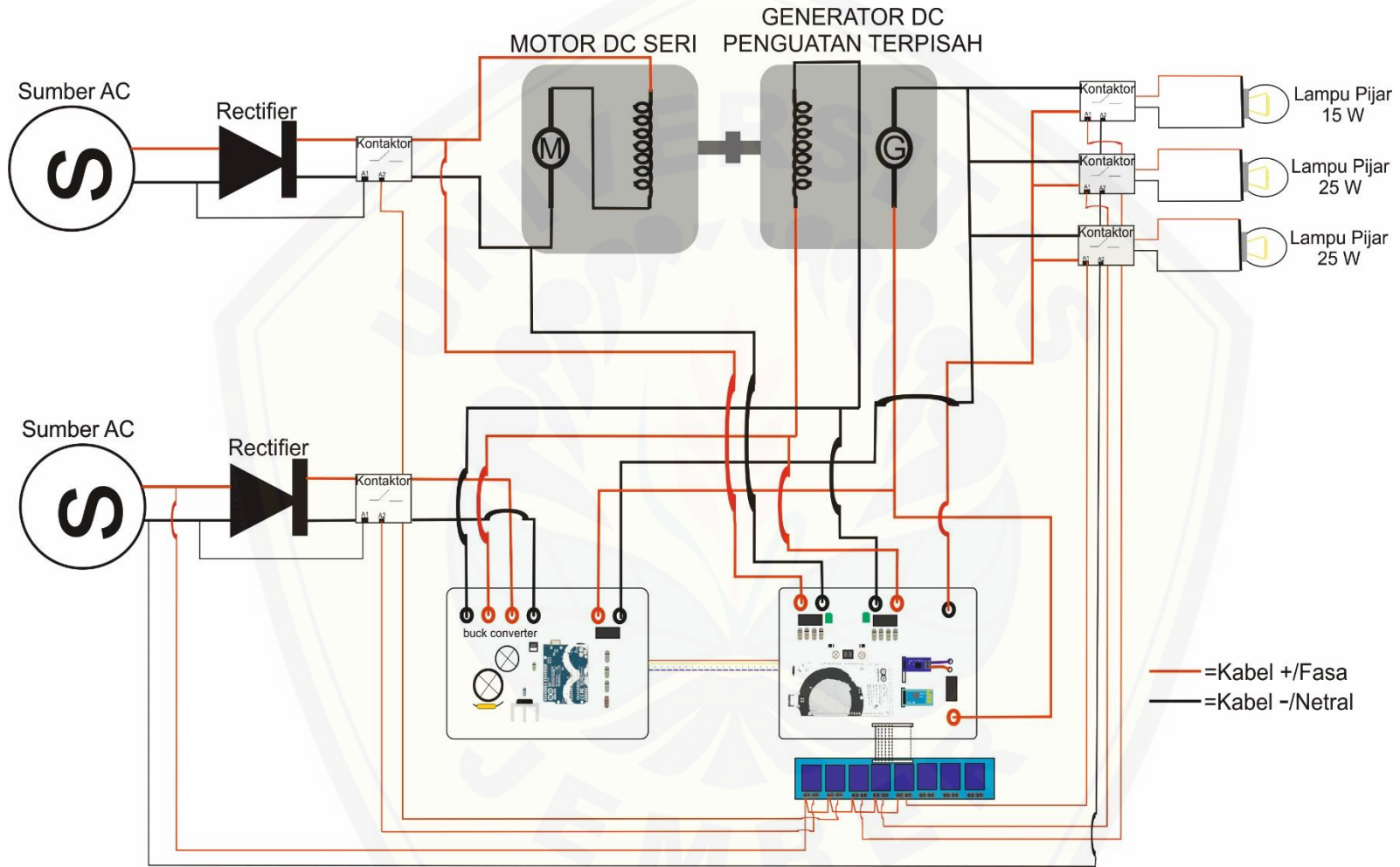
3.6.2 Perancangan dan Pengujian *Database*

Data yang disimpan dalam *database* adalah data yang diterima oleh MTU setelah komunikasi terhubung dengan RTU. Data disimpan dalam format *text* dengan *extention* (.txt). format ini dipilih agar *database* tetap ringan dan mudah dikonversi ke format lain. Keunggulan lain dari penggunaan format ini adalah data ini dapat dibaca dengan menggunakan aplikasi *open source* atau aplikasi yang tidak berbayar.

Dalam perancangan ini *database* didesain untuk dapat langsung disimpan dan di-*backup* ke penyimpanan *online* OneDrive yang merupakan fasilitas penyimpanan data *online* yang tidak berbayar. Lokasi penyimpanan di OneDrive diatur pada jendela sistem utama.

3.7 Wiring Diagram SCADA

Wiring Diagram SCADA digunakan untuk menggambarkan *sistem* SCADA secara menyeluruh mulai dari menyalakan *sistem* pembangkit sampai ke pembebanan. Untuk mempermudah pengambilan data dan mengurangi kesalahan maka perlu adanya *wiring diagram* secara mendetail dan mudah dipahami. *Wiring diagram* SCADA yang dirancang seperti Gambar 3.22. Dalam gambar tersebut digambarkan sistem pengkabelan dari relay, sensor, serta generator dan *prime mover*.



Gambar 3.22 Wiring diagram pengkabelan SCADA

BAB 5. KESIMPULAN DAN SARAN

Pada bab sebelumnya telah dibahas kegiatan perancangan, pengujian dan analisis hasil pengujian untuk “Implementasi Sistem SCADA pada Generator Penguatan Terpisah Menggunakan Arduino Mega 2560”. Pada bab ini akan disampaikan kesimpulan dari keseluruhan hasil kegiatan penelitian ini. Serta saran-saran yang dianggap perlu untuk melanjutkan kegiatan penelitian ini.

5.1 Kesimpulan

Dari hasil perancangan, pengujian dan pembahasan pada bab sebelumnya maka dapat disimpulkan hal-hal yang berhubungan dengan tujuan kegiatan penelitian ini sebagai berikut:

1. Proses perancangan dan implementasi dari sistem SCADA yang dibangun berbasis pada rangkaian controller Arduino yang didukung dengan *software developer* Microsoft Visual Studio 2015 Profesional telah berhasil dengan baik. Hal ini digambarkan dengan hasil pengujian sistem baik pada proses *data acquisition* dan *supervisory control*. Proses pengiriman perintah dari MTU ke RTU dan proses pengiriman data dari RTU ke MTU sudah berjalan sesuai dengan yang diharapkan, dengan kata lain tidak banyak terjadi *error* pada proses ini secara umum. Waktu yang dibutuhkan untuk proses komunikasi data ini tidak lebih dari 4 detik. Hal ini dianggap sebagai delay waktu yang tidak signifikan.
2. Penerapan sistem SCADA untuk melakukan kendali dan supervisi pada *plant* berupa generator DC dengan penguatan terpisah sudah berjalan dengan baik. Semua perintah yang diterapkan pada aktuator menghasilkan respon sistem yang sudah sesuai. Demikian juga halnya semua parameter *plant* mulai dari arus *output* generator, tegangan *output* generator, tegangan *exciter* dan tegangan *prime mover* dapat dimonitor secara *real time* dan dicatat pada *database* yang dapat disimpan secara lokal maupun *online*

3. *Kontroller fuzzy* yang merupakan fitur kendali otomatis untuk menghasilkan tegangan *output* generator sesuai dengan *setpoint* sudah bekerja dengan baik dan menghasilkan pengaturan dengan *error* rata-rata tidak lebih dari 1 %.

5.2 Saran

Untuk tujuan kegiatan penelitian yang dimaksudkan untuk mengembangkan penelitian ini penulis menyarankan beberapa hal berikut ini:

1. *Plant* dapat dikembangkan dengan menggunakan generator AC. Hal ini tentunya akan memerlukan kegiatan penelitian yang berkaitan dengan penggunaan sensor-sensor dan pemrosesan data yang lebih kompleks.
2. Sistem komunikasi jarak jauh yang dalam penelitian ini masih menggunakan sistem Bluetooth dengan jarak jangkauan yang terbatas dapat dikembangkan dengan menggunakan sistem komunikasi dengan jarak jangkauan yang lebih jauh.
3. *Variable* atau parameter *plant* yang akan dimonitor dan dikendalikan juga dapat ditambahkan, misalnya dengan memasukkan *variable* kecepatan, suhu, dan lain-lain.

DAFTAR PUSTAKA

- A. danieels, & w.salter. (1999). WHAT IS SCADA? *International Conference on Accelerator and Large Experimental Physics Control Systems* (pp. 339-343). trieste: CERN.
- Anita, S., Shintal, G., & Joshi, S. (2016). SCADA Implementation For Power Generator Performance and Condition Monitoring System. *International Journal of Advanced Research In Computer and Communication Engineering VOL 5*, 557-558.
- Arfianto, M. (2015). *monitoring dan pengontrolan pembebanan pada pembebanan generator induksi satu fasa berbasis raspberry pi*. jember: Universitas Jember.
- Bayusari, I., Caroline, septiadi, R., & Suprpto, b. (2013). Perancangan Sistem Pemantauan Pengendali Suhu pada Stirred Tank Heater menggunakan Supervisory Control and Data Acquisition. *Jurnal Rekayasa ElektriKa Vol. 10, No. 3*, 153- 159.
- Bondici, C., & Dulau, M. (2014). application of modern SCADA in power operation. *journal of computer science and control system*, 25-28.
- Chapman, S. (2005). *Electric Machinery Fundamentals Fifth Edition*. Melbourne: The McGraw-Hill Companies.
- Densiatama, B. (2012). Rancang Bangun Penyimpanan Mobil Dengan Konsep Master-Slave Berbasis Nirkabel dan Mikrokontroller ATMega8535. 1-15.
- Endaryono, P. J., Harianto, & Wibowo, m. C. (2014). Rancang Bangun Sistem Pembayaran Mandiri Pada Wahana Permainan. *Journal of Control and Network Systems*, 70-77.
- Eritha, F., Nurussa'adah, & Zainuri, a. (2014). Implementasi Bluetooth HC-05 Untuk Mengurangi Tingkat Kecelakaan Pada Pengendara Sepeda Motor.
- Fajar, A. (2016). *Peningkatan Kinerja Generator DC dengan Sistem Eksitasi Berbasis Arduino Uno R3 Menggunakan Buck Converter Dan Kontrol PID*. Jember: Universitas Jember.
- Gajipara, N., & Ahire, P. (2014). Design of SCADA for Real Time System with LabVIEW and Microcontroller. *International Journal of Innovative Research in Advenced Engineering*, 85-90.
- Hadiyanto, R., Widjonarko, & Suprihadi Prasetyono. (2015). Prototype Sistem Compressed Air Energy Storage dengan Pengaturan Valve Menggunakan Fuzzy Logic Controller. *BERKALA SAINSTEK*.

- Indonesia, B. –S. (2010). *RINGKASAN METADATA KEGIATAN STATISTIK*. Jakarta: BPS – Statistics Indonesia.
- Khadiri, K. E., & Qjidaa, H. (2014). Inverting Buck-Boost DCDC Converter Design Challenges. *International Journal of Embedded Systems and Applications (IJESA) Vol.4, No.1*, 1-12.
- Keeli, A., Teja, E., Devi, N., & Venkateswarlu. (2015). Wireless SCADA for Industrial Automation. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2024-2030.
- Kusnadi, N. S., & Azhar, E. (2011). Pengaturan Kecepatan Motor dengan Sistem SCADA Berbasis Mikrokontroler. *JURNAL ILMIAH ELITE ELEKTRO, VOL. 2*, 95-98.
- Kusriyanto, M., & Syariffudin, M. (n.d.). Mini SCADA Berbasis Mikrokontroler ATMEGA 32 Dengan Komunikasi MODBUS RS 485 dan Sistem Monitoring Menggunakan Visual Basic. *Seminar Nasional ke-9: Rekayasa Teknologi Industri dan Informasi*, (pp. 243-248).
- Kusumadewi, S., & Hartati, S. (2006). *NEURO FUZZY: Integrasi Sistem Fuzzy & Jaringan Saraf*. Yogyakarta: Graha Ilmu.
- Lake, J. M. (2010). System And Method For Reducing Memory Leaks In Virtual Machine Programs. *United States Patent US &,734,667B2*.
- Novel, R. (2009). *analisa perancangan sistem SCADA pada sistem kelistrikan Universitas Indonesia*. Depok: universitas indonesia.
- Setiyawan, H. R. (2016). *Sistem Kontrol Fuzzy Logic Pada Generator DC Penguatan Terpisah Berbasis Arduino Uno R3*. Jember: Universitas Jember.
- Sohail, S., Nasim, S., & Ali, F. (2015). Switching of Generator using PLC with SCADA and Implementation of ATS with GSM Alert. *Quest Journals*, 30-34.
- Stouffer, k., falco, j., & kent, k. (2006). *Guide to supervisory control and data Acquisition (SCADA) and industrial control system security*. Gaithersburg: National institute of standart and technology.
- Sulistiyono, T. Y. (2014). *Komparasi Sistem Komunikasi Serial Multipoint Pada Robot Management Sampah Menggunakan I2C dan SPI*. Malang: Universitas Brawijaya.
- Sutarna, N., Kusnadi, & Napiri, A. E. (2011). Pengaturan Kecepatan Motor dengan Sistem SCADA berbasis Mikrokontroler. *Jurnal Ilmiah Elektro VOL 2 NO2*, 95-98.

System, N. C. (2004 , oktober). Supervisory Control and Data. *TECHNICAL INFORMATION BULLETIN 04-1* , pp. 1-43.

Winarno, b. s. (2010). *perancangan sistem SCADA(supervisory control and data acquisition) pada miniatur warehouse berbasis PLC(programmable logic controller)*. depok: universitas indonesia.



A. LAMPIRAN PROGRAM ARDUINO UNO**Inisialisasi variable dan tipe data yang digunakan**

```

#include <Wire.h>
uint8_t Buffer[1] = {1}; //12V, 23°C
String textFromSerial = "";
boolean parsing = false;
String inData = "";
String data[10];
float data_masuk = 0;
int nilaiadcra=0;
int nilaiadc =0;
double nilaiotaladc =0;
float V = 0.00;
float error = 0.00;
float setpoint = 0.00;
float delta_error = 0.00;
float error_sekarang;
float error_lalu;
float U1, U2, U3, U4, U5, U6, U7, U8, U9;
float Z = 0.00;
float NE_, ZE, ZE_, PE;
float DEN_, DEZ, DEZ_, DEP;
float LA, XA, LB, XB;
int WANK1, WANK2, WANK4, WANK7;
float LC, XC, LD, XD;
int WAPK3, WAPK6, WAPK8, WAPK9, WAZK;
float sens = 0.00;
int tegangan = 0, pwm, con, out_kontrol;

```

Inisialisasi variable yang hanya dibaca 1 kali

```

void setup() {
  Serial.begin(9600);
  pinMode(9, OUTPUT);
  analogWrite(9, 134);
  Wire.begin(4); // join i2c bus with address #4
  Wire.onReceive(receiveEvent); // register event
  Wire.onRequest(requestEvent);
}

```

Inisialisasi variable yang dibaca berulang

```

void loop() {
  akses_tegangan();
  kontrol();
}

```

```

setpoint;
tegangan;
}

```

Sensor tegangan output generator

```

void akses_tegangan() {
  nilaitotaladc=0;
  for (int i= 0; i<1000;i++){
    delayMicroseconds(100);
    nilaiadc=analogRead(A0);
    nilaitotaladc=nilaitotaladc+nilaiadc;}
  nilaiadcrata=(nilaitotaladc)/1000;
  tegangan=((float)nilaiadcrata*325.69/1024);
}

```

Mengirim dan menerima data Arduino Mega 2560

```

void requestEvent() {
  akses_tegangan();
  Buffer[0] = tegangan;
  Serial.println(Buffer[0]);
  Wire.write(Buffer, 3);
}
void receiveEvent(int howMany){
  byte index = 0;
  int arrays[2];
  for (int i = 0; i < 3; i++)
  { int c = Wire.read(); // receive a byte as character
    arrays[i] = c; }
  setpoint = arrays[0];
  Serial.print("ar: ");
  Serial.println(arrays[0]);
}

```

Kontrol fuzzy

```

akses_tegangan();
U1 = 0, U2 = 0, U3 = 0, U4 = 0, U5 = 0, U6 = 0, U7 = 0, U8 = 0,U9 = 0;
V = tegangan;
error = setpoint - V;
if(error < -40){
  error = -40;
}
if(error > 220){
  error = 220;
}

```

```
error_sekarang = error;
delta_error = error_sekarang - error_lalu;
  if(delta_error < -260){
    delta_error = -260;
  } if(delta_error > 260){
    delta_error = 260;
  } error_lalu = error_sekarang;
if(error < 0 && error >= -40){
  NE_ = (-error)/40;
  ZE = (error + 40)/40;
if(delta_error < 0 && delta_error >= -260){
  DEN_ = (-delta_error)/260;
  DEZ = (delta_error + 260)/260;
  U1 = min(NE_,DEN_);
  U4 = min(NE_,DEZ);
  U2 = min(ZE,DEN_);
  U5 = min(ZE,DEZ);
} if(delta_error >= 0 && delta_error <= 260){
  DEZ_ = (260 - delta_error)/260;
  DEP = delta_error/260;
  U4 = min(NE_,DEZ_);
  U7 = min(NE_,DEP);
  U5 = min(ZE,DEZ_);
  U8 = min(ZE,DEP);
} if(error >= 0 && error <= 220){
  ZE_ = (220 - error)/220;
  PE = error/220;
if(delta_error < 0 && delta_error >= -260){
  DEN_ = (-delta_error)/260;
  DEZ = (delta_error + 260)/260;
  U2 = min(ZE_,DEN_);
  U5 = min(ZE_,DEZ);
  U3 = min(PE,DEN_);
  U6 = min(PE,DEZ);
} if(delta_error >= 0 && delta_error <= 260){
  DEZ_ = (260 - delta_error)/260;
  DEP = delta_error/260;
  U5 = min(ZE_,DEZ_);
  U8 = min(ZE_,DEP);
  U6 = min(PE,DEZ_);
  U9 = min(PE,DEP);
}
```


$$\begin{aligned} LA &= ((1 - U1) * 255) * U1; \\ XA &= ((1 - U1) * -255)/2 + (-255 * U1); \\ LB &= 0.5 * U1 * (255 * U1); \\ XB &= ((U1 * -255) * 2)/3; \\ WANK1 &= (LA * XA + LB * XB)/(LA + LB); \\ LA &= ((1 - U2) * 255) * U2; \\ XA &= ((1 - U2) * -255)/2 + (-255 * U2); \\ LB &= 0.5 * U2 * (255 * U2); \\ XB &= ((U2 * -255) * 2)/3; \\ WANK2 &= (LA * XA + LB * XB)/(LA + LB); \\ LA &= ((1 - U4) * 255) * U4; \\ XA &= ((1 - U4) * -255)/2 + (-255 * U4); \\ LB &= 0.5 * U4 * (255 * U4); \\ XB &= ((U4 * -255) * 2)/3; \\ WANK4 &= (LA * XA + LB * XB)/(LA + LB); \\ LA &= ((1 - U7) * 255) * U7; \\ XA &= ((1 - U7) * -255)/2 + (-255 * U7); \\ LB &= 0.5 * U7 * (255 * U7); \\ XB &= ((U7 * -255) * 2)/3; \\ WANK7 &= (LA * XA + LB * XB)/(LA + LB); \\ LC &= 0.5 * U3 * (255 * U3); \\ XC &= ((255 * U3) * 2)/3; \\ LD &= ((1 - U3) * 255) * U3; \\ XD &= ((1 - U3) * 255)/2 + (255 * U3); \\ WAPK3 &= (LC * XC + LD * XD)/(LC + LD); \\ LC &= 0.5 * U6 * (255 * U6); \\ XC &= ((255 * U6) * 2)/3; \\ LD &= ((1 - U6) * 255) * U6; \\ XD &= ((1 - U6) * 255)/2 + (255 * U6); \\ WAPK6 &= (LC * XC + LD * XD)/(LC + LD); \\ LC &= 0.5 * U8 * (255 * U8); \\ XC &= ((255 * U8) * 2)/3; \\ LD &= ((1 - U8) * 255) * U8; \\ XD &= ((1 - U8) * 255)/2 + (255 * U8); \\ WAPK8 &= (LC * XC + LD * XD)/(LC + LD); \\ LC &= 0.5 * U9 * (255 * U9); \\ XC &= ((255 * U9) * 2)/3; \\ LD &= ((1 - U9) * 255) * U9; \\ XD &= ((1 - U9) * 255)/2 + (255 * U9); \\ WAPK9 &= (LC * XC + LD * XD)/(LC + LD); \\ WAZK &= 0; \end{aligned}$$

```

    Z = ((WANK1*U1) + (WANK2*U2) + (WAPK3*U3) + (WANK4*U4) +
(WAZK*U5) + (WAPK6*U6) + (WANK7*U7) + (WAPK8*U8) +
(WAPK9*U9))/(U1 + U2 + U3 + U4 + U5 + U6 + U7 + U8 + U9);
    out_kontrol = Z;
}

```

Konversi kontrol fuzzy ke PWM

```

void kontrol() {
    pwm = pwm + out_kontrol;
    if (pwm >= 255)
    { pwm = 255;
    } if (pwm <= 0)
    { pwm = 0;
    } con =255-pwm;
    delay(360);
    analogWrite(9, con);
}

```

B. LAMPIRAN PROGRAM ARDUINO MEGA 2560

Inisialisasi variable dan tipe data yang digunakan

```

#include <Wire.h>
int setpoint = 0;
float average = 0;
float volt0,volt1;
uint8_t Buffer[3] = {1, 400}; //12V, 23*C
Adafruit_INA219 ina219;
byte index = 0;
float arrays[2];
int number = 0;

```

Inisialisasi variable yang hanya dibaca 1 kali

```

void setup()
{
    Wire.begin();           // join i2c bus with address #2
    Serial.begin(9600);
    uint32_t currentFrequency;
    ina219.begin();
    pinMode(6, OUTPUT);//premotor
    pinMode(5, OUTPUT);//exciter
    pinMode(4, OUTPUT);//lampu1
    pinMode(3, OUTPUT);//lampu2
    pinMode(2, OUTPUT);//lampu3
}

```

```
pinMode(A0,INPUT_PULLUP);
pinMode(A1,INPUT_PULLUP);
pinMode(A10, INPUT);
}
```

Menerima data dari MTU

```
void SerialData() {
  if ( Serial.available() > 0 )
  {
    static char input[16];
    static uint8_t i;
    char c = Serial.read();
    if ( c != '\r' && i < 15 ) // assuming "Carriage Return" is chosen in the Serial
    monitor as the line ending character
      input[i++] = c;
    else
    {
      input[i] = '\0';
      i = 0;
      number = atoi(input);
    }
  }
}
```

Pembacaan sensor tegangan premer dan exciter

```
void sensing(){
  #include "sensor.h"
  int adcSensor0=1024-analogRead(A0);
  int adcSensor1=1024-analogRead(A1);
```

Mengirim data ke MTU

```
Serial.print(average);
Serial.print("|");
Serial.print(volt0);
Serial.print("|");
Serial.print(volt1);
Serial.print("|");
Serial.print(arrays[0]);
Serial.print("|");
Serial.println(number);
delay(1000);
}
```

Pembacaan sensor arus ACS 712

```
void arus() {
  unsigned int x=0;
  float ascv=0.0,sams=0.0,y=0.0,sam=0.0,avg=0.0,acsvf=0.0;
  for(int x=0;x<150;x++){
    ascv= analogRead(A10);
    sam=sam+ascv;
    delay(3);
  }
  avg=sam/150.00000;
  acsvf=abs(2.379638321-(avg*(4.79/1024.0)))/0.066;
  if (avg<510.00){average = acsvf;}
  if (avg>510.00){average=(acsvf/2)+0.03;}
}
```

Perintah relay

```
void relaay() {
  SerialData();
  if(number >= 20000 && number <= 20110){
    number = number - 20000;
    setpoint = number;
  } if (number == 200) {
    digitalWrite(3, HIGH); //premove
  } if (number == 201) {
    digitalWrite(3, LOW); //premove
  } if (number == 101) {
    digitalWrite(4, LOW); //exciter
  }if (number == 100 ) {
    digitalWrite(4, HIGH); //exciter
  } if (number == 300) {
    digitalWrite(2, HIGH);
  } //lampu1
  if (number == 301) {
    digitalWrite(2, LOW);
  } //lampu1
  if (number == 400) {
    digitalWrite(5, HIGH);
  } //lampu2
  if (number == 401) {
    digitalWrite(5, LOW);
  } //lampu2
  if (number == 500) {
    digitalWrite(6, HIGH);
  }
```

```

} //lampu3
if (number == 501) {
  digitalWrite(6, LOW);} //lampu3
}

```

Inisialisasi variable yang dibaca berulang

```

void loop()
{
  relaay();
  arus();
  sensing();
}

```

Mengirim dan menerima data Arduino Uno

```

Wire.beginTransmission(4); // transmit to device #4
  Buffer[0] = setpoint;
Wire.write(Buffer, 3);}
Wire.endTransmission(); // stop transmitting
Wire.requestFrom(4, 6);
for (int i = 0; i < 1; i++)
{
  int c = Wire.read(); // receive a byte as character
  arrays[i] = c;
}

```

C. LAMPIRAN PROGRAM VISUAL STUDIO PROFESIONAL 2015

Inisialisasi variable dan tipe data yang digunakan

```

Imports System
Imports System.ComponentModel
Imports System.Threading
Imports System.IO.Ports
Public Class Form2
  Dim comOpen As Boolean 'Keeps track of the port s tatus. True = Open; False
= Closed
  Dim readbuffer As String 'Buffer of whats read from the serial port
  Dim tombol As Boolean
  Dim aryTextFile() As String
  Dim abc As Integer = 0
  Dim detik As Integer = 0
  Dim simpan As Boolean = False

```

Real time clock

```

Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles
Timer1.Tick

```

```

    tanggal.Text = Format(Today, "dddd dd MMMM yyyy")
    jam.Text = Format(TimeOfDay, "HH:mm:ss")
End Sub

```

Pemilihan tipe database

```

Private Sub Form2_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
    SaveFileDialog1.Filter = "TXT File (*.txt)|*.txt"
    If SaveFileDialog1.ShowDialog() = DialogResult.OK Then
        Form3.TextBox1.Text = SaveFileDialog1.FileName
        'simpan = True
    End If
    Timer1.Interval = 500
    Timer1.Enabled = True
    Button2.Enabled = False
    Button3.Enabled = False
    Button4.Enabled = False
    Button5.Enabled = False
    Button6.Enabled = False
    Dim comPorts As String() = System.IO.Ports.SerialPort.GetPortNames
    If comPorts.Count < 1 Then
        'If there are not ports connected, show an error
        MsgBox("tidak ada port yang tersedia coba cek koneksi jaringan anda")
    Else
        Cmbport.Items.AddRange(comPorts)
        Cmbport.Text = comPorts(0)
    End If
End Sub

```

Deklarasi dan inisialisasi komunikasi serial

```

Private Sub SerialPort1_DataReceived(ByVal sender As System.Object, ByVal e
As System.IO.Ports.SerialDataReceivedEventArgs) Handles
SerialPort1.DataReceived
    If comOpen Then
        Try
            'Send data to a new thread to update the ph display
            readbuffer = SerialPort1.ReadLine()
            Me.Invoke(New EventHandler(AddressOf updateData))
        Catch ex As Exception
        End Try
    End If
End Sub

```

Syntax delay

```
Private Sub Delay(ByVal DelayInSeconds As Integer)
    Dim ts As TimeSpan
    Dim targetTime As DateTime = DateTime.Now.AddSeconds(DelayInSeconds)
    Do
        ts = targetTime.Subtract(DateTime.Now)
        Application.DoEvents() ' keep app responsive
        System.Threading.Thread.Sleep(50) ' reduce CPU usage
    Loop While ts.TotalSeconds > 0
End Sub
```

Menerima data RTU

```
Public Sub updateData(ByVal sender As Object, ByVal e As System.EventArgs)
    Dim read As String
    read = readbuffer.Replace(vbCr, "").Replace(vbLf, "")
    aryTextFile = read.Split("|")
    Label9.Text = aryTextFile(0)
    Label10.Text = aryTextFile(1)
    Label11.Text = aryTextFile(2)
    Label12.Text = aryTextFile(3)
    If aryTextFile(1) >= 17 Then
        Label2.BackColor = Color.Blue
    End If
    If aryTextFile(1) < 17 Then
        Label2.BackColor = Color.Maroon
    End If
    If aryTextFile(2) > 1 Then
        Label8.BackColor = Color.Blue
    End If
    If aryTextFile(2) = 0 Then
        Label8.BackColor = Color.Maroon
    End If
    If aryTextFile(3) > 80 Then
        Button2.Enabled = True
        Button3.Enabled = True
        Button4.Enabled = True
        Button5.Enabled = True
        Button6.Enabled = True
    End If
    If aryTextFile(3) < 80 Then
        Button2.Enabled = False
        Button3.Enabled = False
    End If
End Sub
```

```

        Button4.Enabled = False
        Button5.Enabled = False
        Button6.Enabled = False
    End If
    If aryTextFile(0) = 0 Then
        Label3.BackColor = Color.Maroon
        Label4.BackColor = Color.Maroon
        Label5.BackColor = Color.Maroon
        Label6.BackColor = Color.Maroon
        Label7.BackColor = Color.Maroon
    End If
    If aryTextFile(0) > 2 Then
        SerialPort1.WriteLine("301")
        SerialPort1.WriteLine("401")
        SerialPort1.WriteLine("501")
        Button6.BackColor = Color.SteelBlue
        Label7.BackColor = Color.Maroon
        Button4.BackColor = Color.Maroon
        Label5.BackColor = Color.Maroon
        Label3.BackColor = Color.Maroon
        Button3.BackColor = Color.SteelBlue
        Label4.BackColor = Color.Maroon
        Button5.BackColor = Color.SteelBlue
        Label6.BackColor = Color.Maroon
        MsgBox("PERINGATAN:Sistem mematikan otomatis pembebanan generator
        arus melebihi 2 A")
    End If

```

Penyimpanan database

```

    Dim file As System.IO.StreamWriter
    My.Computer.FileSystem.OpenTextFileWriter(Environment.GetFolderPath(Environment.SpecialFolder.Desktop) & "\datalog.txt", True)
    file = My.Computer.FileSystem.OpenTextFileWriter(Form3.TextBox1.Text, True)
    Form3.RichTextBox1.Text = jam.Text + "|" + aryTextFile(0) + "|" +
    aryTextFile(1) + "|" + aryTextFile(2) + "|" + aryTextFile(3) + "|" + TextBox1.Text +
    "|" + daya15.Text + "|" + daya25.Text + "|" + daya40.Text + "|" + daya50.Text + "|" +
    daya65.Text
    file.WriteLine(Form3.RichTextBox1.Text + "|")
    file.Close()
    detik = detik + 1
    AxMathworks_Strip1.AddXY(0, detik, Convert.ToDouble(aryTextFile(3)))
    Form3.RichTextBox1.Text, True)

```



```
End Sub
```

Tombol connect

```
Private Sub ConnectButton_Click(sender As Object, e As EventArgs) Handles  
ConnectButton.Click  
    DoConnect()  
End Sub
```

Tombol Disconnect

```
Private Sub DisconnectButton_Click(sender As Object, e As EventArgs) Handles  
DisconnectButton.Click  
    DoDisconnect()  
End Sub  
Public Sub DoConnect()  
    With SerialPort1()  
        .PortName = cmbPort.Text           'Selected Port  
        .BaudRate = CInt(cmbBaud.Text)     'Baud Rate. 9600 is default.  
        .Parity = IO.Ports.Parity.None  
        .DataBits = 8  
        .StopBits = IO.Ports.StopBits.One  
        .Handshake = IO.Ports.Handshake.None  
        .RtsEnable = False  
        .ReceivedBytesThreshold = 1  
        .NewLine = vbCr  
        .ReadTimeout = 10000  
    End With  
    Try  
        SerialPort1.Open()  
        comOpen = SerialPort1.IsOpen  
    Catch ex As Exception  
        comOpen = False  
        MsgBox("Error Open: " & ex.Message)  
    End Try  
    DisconnectButton.Enabled = True  
    ConnectButton.Enabled = False  
    cmbBaud.Enabled = False  
    cmbPort.Enabled = False  
End Sub  
Public Sub DoDisconnect()  
    If comOpen Then  
        SerialPort1.DiscardInBuffer()  
        SerialPort1.Close()
```

```
comOpen = False
DisconnectButton.Enabled = False
ConnectButton.Enabled = True
cmbBaud.Enabled = True
cmbPort.Enabled = True
End If
End Sub
```

Mendeteksi port otomatis

```
Private Sub RefreshButton_Click(sender As Object, e As EventArgs) Handles
RefreshButton.Click
    Dim comPorts As String() = System.IO.Ports.SerialPort.GetPortNames
    If comPorts.Count < 1 Then
        MsgBox("There are no com ports available! Closing program.")
    Else
        cmbPort.Items.Clear()
        cmbPort.Items.AddRange(comPorts)
        cmbPort.Text = comPorts(0)
    End If
End Sub
```

Tombol preover

```
Private Sub Button7_Click(sender As Object, e As EventArgs) Handles
Button7.Click
    abc = abc + 1
    If abc = 1 Then
```

Pengiriman perintah menyalakan preover pada RTU

```
SerialPort1.WriteLine("200")
```

Data acquisition

```
Button7.BackColor = Color.Maroon
End If
If abc = 2 And aryTextFile(1) >= 17 Then
```

Pengiriman perintah mematikan preover pada RTU

```
SerialPort1.WriteLine("201")
```

Data acquisition

```
Button7.BackColor = Color.SteelBlue
abc = 0
End If
If abc = 2 And aryTextFile(1) < 17 Then
    abc = 0
End If
Delay(15)
If aryTextFile(1) < 17 Then
```

```
Label2.BackColor = Color.Maroon
MsgBox("Generator belum berputar")
End If
End Sub
```

Tombol exciter

```
Private Sub Button8_Click(sender As Object, e As EventArgs) Handles
Button8.Click
abc = abc + 1
If abc = 1 Then
```

Pengiriman perintah menyalakan exciter pada RTU

```
SerialPort1.WriteLine("101")
```

Data acquisition

```
Button8.BackColor = Color.SteelBlue
End If
If abc = 2 Then
```

Pengiriman perintah Mematikan exciter pada RTU

```
SerialPort1.WriteLine("99")
```

Data acquisition

```
Button8.BackColor = Color.Maroon
abc = 0
End If
If abc = 2 Then
abc = 0
End If
End Sub
```

Tombol P1

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
abc = abc + 1
If abc = 1 Then
```

Pengiriman perintah menyalakan P1 pada RTU

```
SerialPort1.WriteLine("300")
```

Data acquisition

```
Button2.BackColor = Color.Maroon
Delay(10)
daya15.Text = Label12.Text * Label9.Text
daya25.Text = 0
daya40.Text = 0
daya50.Text = 0
daya65.Text = 0
If aryTextFile(0) = 0 Then
```

```

        Label3.BackColor = Color.Maroon
        MsgBox("ada permasalahan pada pembebanan generator P1")
    Else
        Label3.BackColor = Color.SteelBlue
    End If
End If
If abc = 2 Then

```

Pengiriman perintah mematikan P1 pada RTU

```

    SerialPort1.WriteLine("301")

```

Data acquisition

```

    Button2.BackColor = Color.SteelBlue
    Label3.BackColor = Color.Maroon
    daya15.Text = 0
    abc = 0
End If
If abc = 2 Then
    abc = 0
End If
End Sub

```

Tombol P2

```

Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
    abc = abc + 1
    If abc = 1 Then

```

Pengiriman perintah menyalakan P2 pada RTU

```

    SerialPort1.WriteLine("400")

```

Data acquisition

```

    Button3.BackColor = Color.Maroon
    Delay(10)
    daya15.Text = 0
    daya25.Text = Label12.Text * Label9.Text
    daya40.Text = 0
    daya50.Text = 0
    daya65.Text = 0
    If aryTextFile(0) = 0 Then
        Label4.BackColor = Color.Maroon
        MsgBox("ada permasalahan pada pembebanan generator P2")
    Else
        Label4.BackColor = Color.SteelBlue
    End If
    Label4.BackColor = Color.SteelBlue
End If
If abc = 2 Then

```

Pengiriman perintah mematikan P2 pada RTU

```
SerialPort1.WriteLine("401")
```

Data acquisition

```
Button3.BackColor = Color.SteelBlue
```

```
Label4.BackColor = Color.Maroon
```

```
daya25.Text = 0
```

```
abc = 0
```

```
End If
```

```
If abc = 2 Then
```

```
abc = 0
```

```
End If
```

```
End Sub
```

Tombol P3

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles
```

```
Button4.Click
```

```
abc = abc + 1
```

```
If abc = 1 Then
```

Pengiriman perintah menyalakan P3 pada RTU

```
SerialPort1.WriteLine("400")
```

```
SerialPort1.WriteLine("300")
```

Data acquisition

```
Button4.BackColor = Color.Maroon
```

```
Delay(10)
```

```
daya15.Text = 0
```

```
daya25.Text = 0
```

```
daya40.Text = Label12.Text * Label9.Text
```

```
daya50.Text = 0
```

```
daya65.Text = 0
```

```
If aryTextFile(0) = 0 Then
```

```
Label5.BackColor = Color.Maroon
```

```
MsgBox("ada permasalahan pada pembebanan generator P3")
```

```
Else
```

```
Label5.BackColor = Color.SteelBlue
```

```
End If
```

```
End If
```

```
If abc = 2 Then
```

Pengiriman perintah mematikan P3 pada RTU

```
SerialPort1.WriteLine("401")
```

```
SerialPort1.WriteLine("301")
```

Data acquisition

```
Button4.BackColor = Color.SteelBlue
```

```
Label5.BackColor = Color.Maroon
```

```

        daya40.Text = 0
        abc = 0
    End If
    If abc = 2 Then
        abc = 0
    End If
End Sub

```

Tombol P4

Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click

```

    abc = abc + 1
    If abc = 1 Then

```

Pengiriman perintah menyalakan P4 pada RTU

```

        SerialPort1.WriteLine("400")
        SerialPort1.WriteLine("500")

```

Data acquisition

```

        Button5.BackColor = Color.Maroon
        Delay(10)
        daya15.Text = 0
        daya25.Text = 0
        daya40.Text = 0
        daya50.Text = Label12.Text * Label9.Text
        daya65.Text = 0
        If aryTextFile(0) = 0 Then
            Label6.BackColor = Color.Maroon
            MsgBox("ada permasalahan pada pembebanan generator P4")
        Else
            Label6.BackColor = Color.SteelBlue
        End If
    End If
    If abc = 2 Then

```

Pengiriman perintah mematikan P4 pada RTU

```

        SerialPort1.WriteLine("401")
        SerialPort1.WriteLine("501")

```

Data acquisition

```

        Button5.BackColor = Color.SteelBlue
        Label6.BackColor = Color.Maroon
        daya50.Text = 0
        abc = 0
    End If
    If abc = 2 Then
        abc = 0

```

```
End If  
End Sub
```

Tombol P5

```
Private Sub Button6_Click(sender As Object, e As EventArgs) Handles  
Button6.Click  
    abc = abc + 1  
    If abc = 1 Then
```

Pengiriman perintah menyalakan P5 pada RTU

```
    SerialPort1.WriteLine("300")  
    SerialPort1.WriteLine("400")  
    SerialPort1.WriteLine("500")
```

Data acquisition

```
    Button6.BackColor = Color.Maroon  
    Delay(10)  
    daya15.Text = 0  
    daya25.Text = 0  
    daya40.Text = 0  
    daya50.Text = 0  
    daya65.Text = Label12.Text * Label9.Text  
    If aryTextFile(0) = 0 Then  
        Label7.BackColor = Color.Maroon  
        MsgBox("ada permasalahan pada pembebanan generator P5")  
    Else  
        Label7.BackColor = Color.SteelBlue  
    End If  
End If  
If abc = 2 Then
```

Pengiriman perintah mematikan P5 pada RTU

```
    SerialPort1.WriteLine("301")  
    SerialPort1.WriteLine("401")  
    SerialPort1.WriteLine("501")
```

Data acquisition

```
    Button6.BackColor = Color.SteelBlue  
    Label7.BackColor = Color.Maroon  
    daya65.Text = 0  
    abc = 0  
End If  
If abc = 2 Then  
    abc = 0  
End If  
End Sub
```

```
Private Sub TextBox1_TextChanged(sender As Object, e As EventArgs) Handles  
    TextBox1.TextChanged  
End Sub
```

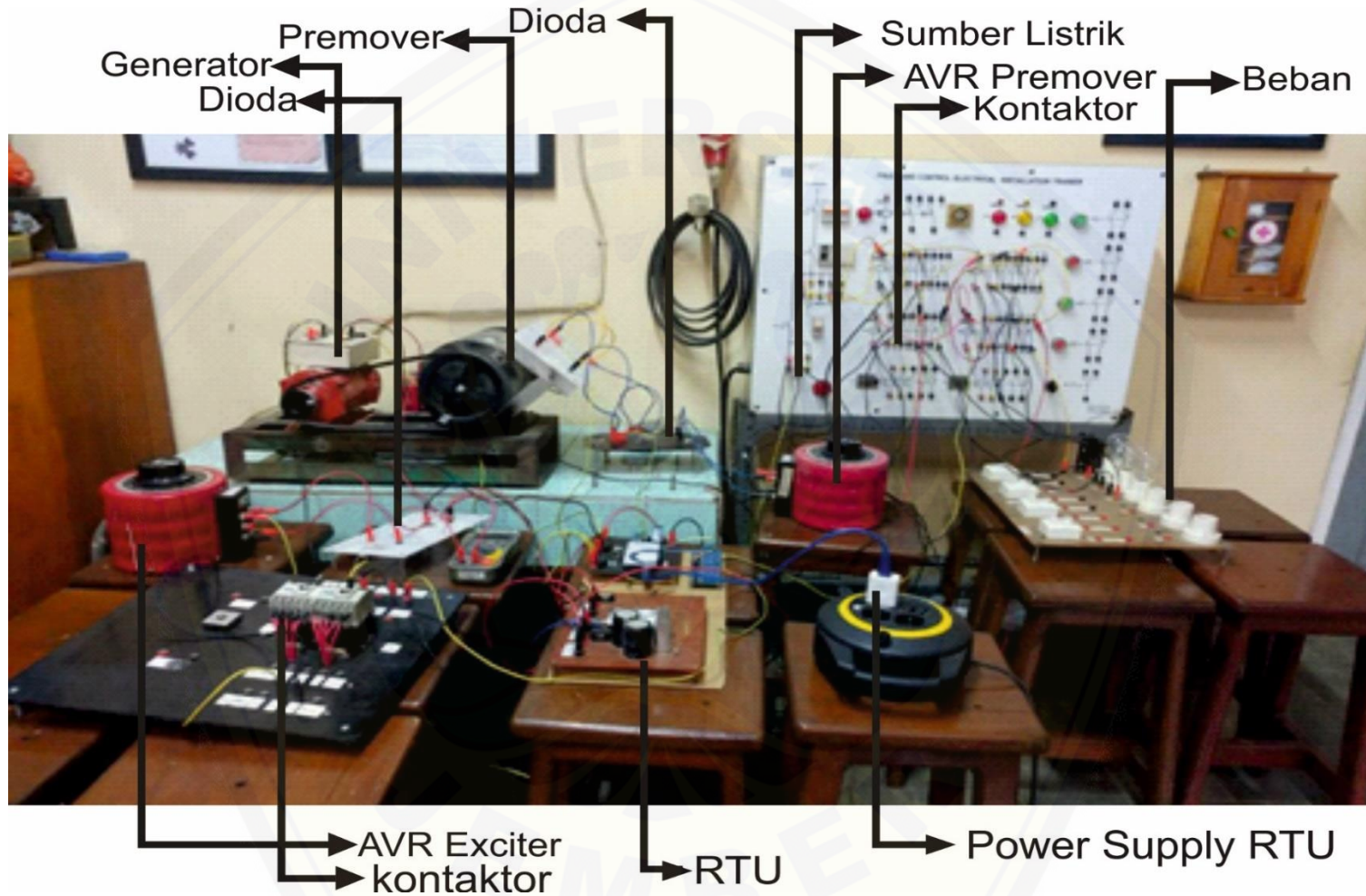
Tombol kirim setpoint

```
Private Sub Button9_Click(sender As Object, e As EventArgs) Handles  
    Button9.Click
```

Pengiriman perintah setpoint pada RTU

```
        SerialPort1.WriteLine(TextBox1.Text + 20000)  
    End Sub  
Private Sub Form2_Closing(sender As Object, e As CancelEventArgs) Handles  
    Me.Closing  
        Form1.Close()  
        Form3.Close()  
    End Sub  
End Class
```


D. LAMPIRAN GAMBAR PENGUJIAN



E. Lampiran Data Hasil Penelitian

Tabel A.1 Data hasil penelitian yang tersimpan di *database*

Waktu	I out Generator	V premove	V Exciter	V out Generator	Set Point	Daya P1	Daya P1	Daya P1	Daya P1	Daya P1
10:22:00	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:00	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:00	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:01	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:01	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:01	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:01	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:01	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:03	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:03	0.00 A	0.00 V	1.55 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:03	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:03	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:05	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:05	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:07	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:09	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:11	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:13	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:15	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:18	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:20	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:22	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:22:24	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:26	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:28	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:30	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:32	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:34	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:36	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:37	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:39	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:41	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:43	0.00 A	4.89 V	4.89 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:45	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:47	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:49	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:51	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:53	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:55	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:57	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:22:58	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:00	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:02	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:04	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:06	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:08	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:10	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:12	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:23:14	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:16	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:18	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:19	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:21	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:23	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:25	0.00 A	0.00 V	0.00 V	-1.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:27	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:29	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:31	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:33	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:35	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:37	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:39	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:40	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:42	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:44	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:46	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:48	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:50	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:52	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:54	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:56	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:58	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:23:59	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:01	0.00 A	0.00 V	0.00 V	0.00 V	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:24:03	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:05	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:07	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:09	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:11	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:13	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:15	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:17	0.00 A	0.00 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:19	0.00 A	19.90 V	6.55 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:20	0.00 A	11.69 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:22	0.00 A	12.48 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:24	0.00 A	13.92 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:26	0.00 A	23.63 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:28	0.00 A	31.38 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:30	0.00 A	30.34 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:32	0.00 A	31.10 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:34	0.00 A	33.34 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:36	0.00 A	58.85 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:38	0.00 A	75.41 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:40	0.00 A	79.02 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:41	0.00 A	89.97 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:43	0.00 A	92.98 V	0.00 V	-1.00		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:45	0.00 A	86.97 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:47	0.00 A	89.97 V	0.00 V	0.00 V		0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:49	0.00 A	95.98 V	0.00 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:51	0.00 A	92.98 V	0.00 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:24:52	0.00 A	92.98 V	0.00 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:54	0.00 A	89.97 V	0.00 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:56	0.00 A	86.97 V	0.00 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:24:58	0.00 A	89.97 V	0.00 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:00	0.00 A	89.97 V	43.69 V	107.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:02	0.00 A	89.97 V	45.99 V	79.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:04	0.00 A	92.98 V	46.29 V	90.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:06	0.00 A	86.97 V	49.06 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:08	0.00 A	92.98 V	50.28 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:10	0.00 A	89.97 V	50.78 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:12	0.00 A	89.97 V	51.12 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:14	0.00 A	95.98 V	51.34 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:15	0.00 A	83.97 V	51.64 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:17	0.00 A	89.97 V	51.82 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:19	0.00 A	89.97 V	51.96 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:21	0.00 A	86.97 V	52.00 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:23	0.00 A	95.98 V	51.89 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:25	0.00 A	86.97 V	52.00 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:27	0.00 A	89.97 V	51.96 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:29	0.00 A	92.98 V	54.43 V	101.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:31	0.00 A	89.97 V	54.49 V	101.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:33	0.00 A	92.98 V	51.74 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:35	0.00 A	89.97 V	51.21 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:37	0.00 A	86.97 V	51.56 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:39	0.00 A	92.98 V	51.93 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:40	0.00 A	89.97 V	51.89 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:25:42	0.00 A	89.97 V	52.00 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:25:44	0.06 A	89.97 V	54.36 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	9.00 W
10:25:46	0.09 A	89.97 V	54.36 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	8.73 W
10:25:48	0.17 A	96.97 V	54.92 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	16.49 W
10:25:50	0.17 A	92.98 V	55.54 V	93.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	15.81 W
10:25:52	0.25 A	95.98 V	56.92 V	95.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	23.75 W
10:25:54	0.23 A	89.97 V	58.04 V	91.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	20.93 W
10:25:56	0.26 A	95.98 V	59.47 V	92.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	23.92 W
10:25:58	0.25 A	89.97 V	60.71 V	93.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	23.25 W
10:26:00	0.26 A	95.98 V	61.52 V	94.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.44 W
10:26:02	0.25 A	89.97 V	62.83 V	95.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	23.75 W
10:26:04	0.24 A	89.97 V	67.31 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	23.76 W
10:26:06	0.27 A	92.98 V	68.68 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	26.73 W
10:26:07	0.24 A	86.97 V	69.18 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	23.76 W
10:26:09	0.26 A	92.98 V	69.31 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	25.74 W
10:26:11	0.25 A	92.98 V	69.31 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:13	0.26 A	89.97 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	25.74 W
10:26:15	0.27 A	92.98 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	26.73 W
10:26:17	0.25 A	89.97 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:19	0.26 A	86.97 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	25.74 W
10:26:21	0.25 A	89.97 V	69.18 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:23	0.25 A	92.98 V	69.18 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:25	0.25 A	89.97 V	69.18 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:27	0.25 A	92.98 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:29	0.25 A	95.98 V	69.31 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:31	0.25 A	95.98 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W

lanjutan Tabel A.1

10:26:32	0.25 A	92.98 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:34	0.25 A	92.98 V	69.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	24.75 W
10:26:36	0.18 A	89.97 V	69.31 V	105.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	18.90 W
10:26:38	0.06 A	92.98 V	63.89 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	5.88 W
10:26:10	0.00 A	89.97 V	62.08 V	105.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:26:42	0.00 A	89.97 V	59.97 V	103.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:26:44	0.00 A	86.97 V	57.54 V	109.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:26:46	0.00 A	92.98 V	49.39 V	91.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:26:48	0.09 A	86.97 V	49.88 V	100.00 V	100	0.00 W	0.00 W	0.00 W	9.00 W	0.00 W
10:26:50	0.09 A	89.97 V	50.47 V	103.00 V	100	0.00 W	0.00 W	0.00 W	9.27 W	0.00 W
10:26:52	0.09 A	89.97 V	48.46 V	98.00 V	100	0.00 W	0.00 W	0.00 W	8.82 W	0.00 W
10:26:54	0.10 A	89.97 V	47.66 V	97.00 V	100	0.00 W	0.00 W	0.00 W	9.70 W	0.00 W
10:26:55	0.18 A	89.97 V	47.93 V	92.00 V	100	0.00 W	0.00 W	0.00 W	16.56 W	0.00 W
10:26:57	0.18 A	92.98 V	48.59 V	92.00 V	100	0.00 W	0.00 W	0.00 W	16.56 W	0.00 W
10:26:59	0.18 A	89.97 V	49.84 V	90.00 V	100	0.00 W	0.00 W	0.00 W	16.20 W	0.00 W
10:27:01	0.18 A	95.98 V	51.49 V	93.00 V	100	0.00 W	0.00 W	0.00 W	16.74 W	0.00 W
10:27:03	0.18 A	86.97 V	55.36 V	94.00 V	100	0.00 W	0.00 W	0.00 W	16.92 W	0.00 W
10:27:05	0.18 A	89.97 V	56.85 V	95.00 V	100	0.00 W	0.00 W	0.00 W	17.10 W	0.00 W
10:27:07	0.18 A	89.97 V	57.79 V	96.00 V	100	0.00 W	0.00 W	0.00 W	17.28 W	0.00 W
10:27:09	0.18 A	89.97 V	58.35 V	97.00 V	100	0.00 W	0.00 W	0.00 W	17.46 W	0.00 W
10:27:11	0.18 A	92.98 V	58.85 V	97.00 V	100	0.00 W	0.00 W	0.00 W	17.46 W	0.00 W
10:27:13	0.18 A	89.97 V	59.10 V	97.00 V	100	0.00 W	0.00 W	0.00 W	17.46 W	0.00 W
10:27:15	0.16 A	86.97 V	59.47 V	98.00 V	100	0.00 W	0.00 W	0.00 W	15.68 W	0.00 W
10:27:16	0.16 A	89.97 V	59.72 V	98.00 V	100	0.00 W	0.00 W	0.00 W	15.68 W	0.00 W
10:27:18	0.17 A	89.97 V	59.91 V	98.00 V	100	0.00 W	0.00 W	0.00 W	16.66 W	0.00 W
10:27:20	0.18 A	89.97 V	60.09 V	98.00 V	100	0.00 W	0.00 W	0.00 W	17.64 W	0.00 W

lanjutan Tabel A.1

10:27:22	0.17 A	92.98 V	60.34 V	98.00 V	100	0.00 W	0.00 W	0.00 W	16.66 W	0.00 W
10:27:24	0.16 A	89.97 V	60.65 V	99.00 V	100	0.00 W	0.00 W	0.00 W	15.84 W	0.00 W
10:27:26	0.18 A	89.97 V	60.84 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:28	0.18 A	86.97 V	60.96 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:30	0.18 A	89.97 V	60.96 V	98.00 V	100	0.00 W	0.00 W	0.00 W	17.64 W	0.00 W
10:27:34	0.18 A	86.97 V	61.34 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:36	0.18 A	92.98 V	61.40 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:38	0.18 A	89.97 V	61.40 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:40	0.18 A	92.98 V	61.34 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:41	0.18 A	89.97 V	61.21 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:43	0.18 A	95.98 V	61.15 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:45	0.16 A	89.97 V	61.15 V	99.00 V	100	0.00 W	0.00 W	0.00 W	15.84 W	0.00 W
10:27:47	0.17 A	92.98 V	61.09 V	98.00 V	100	0.00 W	0.00 W	0.00 W	16.66 W	0.00 W
10:27:49	0.18 A	89.97 V	61.21 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:51	0.18 A	95.98 V	61.52 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:53	0.18 A	86.97 V	61.71 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:55	0.18 A	89.97 V	61.71 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:57	0.18 A	89.97 V	61.71 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:27:59	0.18 A	92.98 V	61.59 V	98.00 V	100	0.00 W	0.00 W	0.00 W	17.64 W	0.00 W
10:28:01	0.18 A	92.98 V	61.71 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:28:03	0.18 A	89.97 V	61.77 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:28:06	0.18 A	86.97 V	61.84 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:28:08	0.18 A	92.98 V	61.71 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:28:10	0.18 A	89.97 V	61.65 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:28:12	0.18 A	89.97 V	61.90 V	99.00 V	100	0.00 W	0.00 W	0.00 W	17.82 W	0.00 W
10:28:14	0.10 A	86.97 V	61.77 V	99.00 V	100	0.00 W	0.00 W	0.00 W	9.90 W	0.00 W

lanjutan Tabel A.1

10:28:16	0.09 A	89.97 V	61.77 V	105.00 V	100	0.00 W	0.00 W	0.00 W	9.45 W	0.00 W
10:28:18	0.09 A	89.97 V	60.28 V	103.00 V	100	0.00 W	0.00 W	0.00 W	9.27 W	0.00 W
10:28:20	0.09 A	86.97 V	57.60 V	107.00 V	100	0.00 W	0.00 W	0.00 W	9.63 W	0.00 W
10:28:22	0.09 A	89.97 V	49.60 V	96.00 V	100	0.00 W	0.00 W	0.00 W	8.64 W	0.00 W
10:28:24	0.09 A	89.97 V	46.19 V	92.00 V	100	0.00 W	0.00 W	0.00 W	8.28 W	0.00 W
10:28:26	0.09 A	89.97 V	47.30 V	97.00 V	100	0.00 W	0.00 W	0.00 W	8.73 W	0.00 W
10:28:28	0.00 A	83.97 V	48.55 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:31	0.00 A	89.97 V	48.94 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:33	0.00 A	89.97 V	49.06 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:35	0.00 A	89.97 V	49.02 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:37	0.00 A	86.97 V	49.15 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:39	0.00 A	92.98 V	49.10 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:41	0.00 A	86.97 V	49.10 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:43	0.00 A	95.98 V	49.02 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:28:45	0.06 A	86.97 V	49.10 V	99.00 V	100	0.00 W	0.00 W	5.94 W	0.00 W	0.00 W
10:28:47	0.07 A	89.97 V	49.15 V	99.00 V	100	0.00 W	0.00 W	6.93 W	0.00 W	0.00 W
10:28:49	0.06 A	89.97 V	49.10 V	99.00 V	100	0.00 W	0.00 W	5.94 W	0.00 W	0.00 W
10:28:51	0.18 A	89.97 V	50.28 V	96.00 V	100	0.00 W	0.00 W	17.28 W	0.00 W	0.00 W
10:28:53	0.16 A	89.97 V	51.56 V	93.00 V	100	0.00 W	0.00 W	14.88 W	0.00 W	0.00 W
10:28:55	0.16 A	89.97 V	55.05 V	95.00 V	100	0.00 W	0.00 W	15.20 W	0.00 W	0.00 W
10:28:56	0.16 A	86.97 V	56.42 V	96.00 V	100	0.00 W	0.00 W	15.36 W	0.00 W	0.00 W
10:28:58	0.16 A	89.97 V	57.29 V	96.00 V	100	0.00 W	0.00 W	15.36 W	0.00 W	0.00 W
10:29:00	0.16 A	89.97 V	57.91 V	97.00 V	100	0.00 W	0.00 W	15.52 W	0.00 W	0.00 W
10:29:02	0.16 A	83.97 V	58.29 V	97.00 V	100	0.00 W	0.00 W	15.52 W	0.00 W	0.00 W
10:29:04	0.16 A	92.98 V	58.60 V	97.00 V	100	0.00 W	0.00 W	15.52 W	0.00 W	0.00 W
10:29:06	0.16 A	86.97 V	58.85 V	98.00 V	100	0.00 W	0.00 W	15.68 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:29:08	0.16 A	89.97 V	59.16 V	98.00 V	100	0.00 W	0.00 W	15.68 W	0.00 W	0.00 W
10:29:10	0.16 A	95.98 V	59.41 V	98.00 V	100	0.00 W	0.00 W	15.68 W	0.00 W	0.00 W
10:29:12	0.16 A	89.97 V	59.66 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:14	0.16 A	89.97 V	59.78 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:16	0.16 A	92.98 V	59.84 V	98.00 V	100	0.00 W	0.00 W	15.68 W	0.00 W	0.00 W
10:29:18	0.16 A	92.98 V	60.15 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:20	0.16 A	92.98 V	60.40 V	99.00 V	1 00	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:22	0.17 A	89.97 V	60.59 V	99.00 V	100	0.00 W	0.00 W	16.83 W	0.00 W	0.00 W
10:29:23	0.15 A	92.98 V	60.59 V	99.00 V	100	0.00 W	0.00 W	14.85 W	0.00 W	0.00 W
10:29:25	0.16 A	89.97 V	60.53 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:27	0.16 A	86.97 V	60.40 V	98.00 V	100	0.00 W	0.00 W	15.68 W	0.00 W	0.00 W
10:29:29	0.16 A	92.98 V	60.34 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:31	0.16 A	89.97 V	60.34 V	98.00 V	100	0.00 W	0.00 W	15.68 W	0.00 W	0.00 W
10:29:33	0.16 A	86.97 V	60.47 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:35	0.16 A	86.97 V	60.59 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:37	0.18 A	89.97 V	60.53 V	99.00 V	100	0.00 W	0.00 W	17.82 W	0.00 W	0.00 W
10:29:39	0.16 A	89.97 V	60.65 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:41	0.16 A	89.97 V	60.65 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:43	0.15 A	89.97 V	60.65 V	99.00 V	100	0.00 W	0.00 W	14.85 W	0.00 W	0.00 W
10:29:45	0.16 A	89.97 V	60.78 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:47	0.15 A	92.98 V	60.78 V	99.00 V	100	0.00 W	0.00 W	14.85 W	0.00 W	0.00 W
10:29:48	0.16 A	92.98 V	60.84 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:00	0.16 A	89.97 V	60.96 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:52	0.16 A	86.97 V	61.03 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:29:54	0.15 A	86.97 V	60.96 V	99.00 V	100	0.00 W	0.00 W	14.85 W	0.00 W	0.00 W
10:29:56	0.16 A	89.97 V	60.96 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:29:58	0.16 A	92.98 V	60.84 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:30:00	0.16 A	86.97 V	60.78 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:30:02	0.16 A	89.97 V	60.59 V	99.00 V	100	0.00 W	0.00 W	15.84 W	0.00 W	0.00 W
10:30:04	0.00 A	86.97 V	60.71 V	104.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:06	0.00 A	86.97 V	58.54 V	102.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:08	0.00 A	89.97 V	54.86 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:09	0.00 A	89.97 V	54.86 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:11	0.00 A	92.98 V	55.11 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:13	0.00 A	89.97 V	55.30 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:15	0.00 A	95.98 V	57.79 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:17	0.00 A	92.98 V	57.17 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:19	0.00 A	86.97 V	59.47 V	102.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:21	0.00 A	95.98 V	56.29 V	102.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:23	0.00 A	92.98 V	52.07 V	101.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:25	0.00 A	89.97 V	51.53 V	108.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:26	0.00 A	98.98 V	48.55 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:28	0.00 A	89.97 V	47.34 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:30	0.00 A	89.97 V	47.62 V	101.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:32	0.00 A	89.97 V	47.16 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:34	0.00 A	92.98 V	46.49 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:36	0.00 A	92.98 V	46.44 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:38	0.00 A	89.97 V	46.68 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:40	0.00 A	92.98 V	46.73 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:42	0.00 A	98.98 V	46.44 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:30:44	0.00 A	89.97 V	46.58 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:30:46	0.09 A	89.97 V	46.53 V	93.00 V	100	0.00 W	8.37 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:30:48	0.09 A	92.98 V	47.75 V	95.00 V	100	0.00 W	8.55 W	0.00 W	0.00 W	0.00 W
10:30:50	0.09 A	92.98 V	48.72 V	97.00 V	100	0.00 W	8.73 W	0.00 W	0.00 W	0.00 W
10:30:52	0.09 A	86.97 V	49.27 V	97.00 V	100	0.00 W	8.73 W	0.00 W	0.00 W	0.00 W
10:30:54	0.09 A	89.97 V	49.60 V	97.00 V	100	0.00 W	8.73 W	0.00 W	0.00 W	0.00 W
10:30:56	0.09 A	95.98 V	49.80 V	98.00 V	100	0.00 W	8.82 W	0.00 W	0.00 W	0.00 W
10:30:57	0.09 A	92.98 V	49.80 V	98.00 V	100	0.00 W	8.82 W	0.00 W	0.00 W	0.00 W
10:30:59	0.09 A	89.97 V	50.12 V	98.00 V	100	0.00 W	8.82 W	0.00 W	0.00 W	0.00 W
10:31:01	0.09 A	92.98 V	50.32 V	98.00 V	100	0.00 W	8.82 W	0.00 W	0.00 W	0.00 W
10:31:03	0.09 A	89.97 V	50.47 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:05	0.09 A	92.98 V	50.43 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:07	0.09 A	92.98 V	50.55 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:09	0.09 A	89.97 V	50.59 V	98.00 V	100	0.00 W	8.82 W	0.00 W	0.00 W	0.00 W
10:31:11	0.08 A	89.97 V	50.66 V	99.00 V	100	0.00 W	7.92 W	0.00 W	0.00 W	0.00 W
10:31:13	0.09 A	92.98 V	50.74 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:15	0.09 A	89.97 V	50.85 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:16	0.09 A	89.97 V	50.74 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:18	0.09 A	92.98 V	50.93 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:20	0.09 A	92.98 V	50.85 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:22	0.09 A	89.97 V	50.74 V	99.00 V	100	0.00 W	8.91 W	0.00 W	0.00 W	0.00 W
10:31:25	0.08 A	92.98 V	50.78 V	99.00 V	100	0.00 W	7.92 W	0.00 W	0.00 W	0.00 W
10:31:26	0.00 A	92.98 V	50.89 V	106.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:28	0.00 A	95.98 V	48.07 V	101.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:30	0.00 A	86.97 V	45.69 V	95.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:32	0.00 A	89.97 V	45.84 V	97.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:34	0.00 A	92.98 V	46.24 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:36	0.00 A	92.98 V	46.58 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:31:38	0.00 A	92.98 V	46.58 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:40	0.00 A	89.97 V	46.87 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:41	0.00 A	92.98 V	46.92 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:43	0.00 A	92.98 V	46.92 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:45	0.06 A	86.97 V	46.92 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:47	0.06 A	89.97 V	46.82 V	93.00 V	100	5.58 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:49	0.06 A	89.97 V	47.93 V	96.00 V	100	5.76 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:51	0.06 A	92.98 V	48.81 V	97.00 V	100	5.82 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:53	0.06 A	92.98 V	49.31 V	97.00 V	100	5.82 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:55	0.06 A	92.98 V	49.60 V	98.00 V	100	5.88 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:57	0.06 A	92.98 V	49.72 V	98.00 V	100	5.88 W	0.00 W	0.00 W	0.00 W	0.00 W
10:31:59	0.06 A	86.97 V	49.92 V	98.00 V	100	5.88 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:01	0.06 A	92.98 V	50.04 V	98.00 V	100	5.88 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:03	0.06 A	92.98 V	50.20 V	98.00 V	100	5.88 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:05	0.06 A	89.97 V	50.32 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:07	0.06 A	89.97 V	50.51 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:09	0.06 A	92.98 V	50.66 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:11	0.06 A	92.98 V	50.70 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:12	0.06 A	92.98 V	50.63 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:14	0.06 A	89.97 V	50.66 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:16	0.06 A	92.98 V	50.59 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:18	0.06 A	89.97 V	50.66 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:20	0.06 A	92.98 V	50.74 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:22	0.07 A	89.97 V	50.74 V	99.00 V	100	6.93 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:24	0.06 A	89.97 V	50.59 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:26	0.06 A	92.98 V	50.63 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:32:28	0.06 A	92.98 V		99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:30	0.06 A	92.98 V	50.74 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:32	0.06 A	92.98 V	50.66 V	99.00 V	100	5.94 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:34	0.00 A	89.97 V	50.59 V	99.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:35	0.00 A	89.97 V	50.78 V	106.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:37	0.00 A	89.97 V	48.07 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:39	0.00 A	92.98 V	45.01 V	94.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:41	0.00 A	89.97 V	45.74 V	98.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:43	0.00 A	92.98 V	46.82 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:45	0.00 A	89.97 V	47.20 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:47	0.00 A	92.98 V	47.16 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:49	0.00 A	89.97 V	47.20 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:51	0.00 A	92.98 V	47.25 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:53	0.00 A	92.98 V	47.39 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:55	0.00 A	92.98 V	47.53 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:57	0.00 A	89.97 V	47.48 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:32:59	0.00 A	92.98 V	47.48 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:01	0.00 A	92.98 V	47.53 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:03	0.00 A	92.98 V	47.53 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:04	0.00 A	89.97 V	47.48 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:06	0.00 A	95.98 V	47.53 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:08	0.00 A	92.98 V	47.48 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:10	0.00 A	89.97 V	47.48 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:12	0.00 A	92.98 V	47.57 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:14	0.00 A	98.98 V	47.39 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:16	0.00 A	89.97 V	47.25 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W

lanjutan Tabel A.1

10:33:18	0.00 A	92.98 V	47.43 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:20	0.00 A	92.98 V	47.43 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:21	0.00 A	92.98 V	47.43 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:23	0.00 A	86.97 V	47.57 V	100.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:26	0.00 A	58.41 V	47.57 V	81.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:28	0.00 A	34.24 V	49.56 V	37.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:30	0.00 A	20.61 V	57.23 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:31	0.00 A	13.40 V	60.40 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:33	0.00 A	23.24 V	50.93 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:35	0.00 A	4.89 V	30.91 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W
10:33:37	0.00 A	0.00 V	17.86 V	0.00 V	100	0.00 W	0.00 W	0.00 W	0.00 W	0.00 W