



TUGAS AKHIR - TE 145561

**SISTEM PELAPORAN PENGOPERASIAN GARDU
PORTAL KONVENSIONAL SATU FASA DALAM KONDISI
PEMELIHARAAN**

Lutfi Kurnia Hilmy
NRP 2214038005

Dosen Pembimbing
Ir. Sjamsul Anam, MT.
Danar Fahmi ST., MT.

PROGRAM STUDI TEKNIK LISTRIK
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - TE 145561

**REPORTING SYSTEM CONVENTIONAL PORTAL
SUBSTATIONS ONE PHASE IN MAINTENANCE
CONDITIONS**

Lutfi Kurnia Hilmy
NRP 2214038005

Advisor
Ir. Sjamsul Anam, MT.
Danar Fahmi ST., MT.

ELECTRICAL ENGINEERING STUDY PROGRAM
Electrical and Automation Engineering Department
Vocational Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2017

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas akhir saya dengan judul **“Sistem Pelaporan Pengoperasian Gardu Portal Konvensional Satu Fasa dalam Kondisi Pemeliharaan”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

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

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

Surabaya, 20 Juli 2017



Lutfi Kurnia Hilmy
NRP 2214038005

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

**SISTEM PELAPORAN PENGOPERASIAN GARDU PORTAL
KONVENSIONAL SATU FASA DALAM KONDISI
PEMELIHARAAN**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada**

**Program Studi Komputer Kontrol
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I,

Dosen Pembimbing II,

**Ir. Samsjul Anam, MT. Daniar Fahmi, ST., MT.
NIP. 196307251 99003 1 002 EKT NIP. 1989092520 1404 1 002**

**SURABAYA
JULI, 2017**

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

SISTEM PELAPORAN PENGOPERASIAN GARDU PORTAL KONVENSIIONAL SATU FASA DALAM KONDISI PEMELIHARAAN

Nama : Lutfi Kurnia Hilmy
Pembimbing I : Ir. Sjamsul Anam, MT.
Pembimbing II : Daniar Fahmi ST., MT.

ABSTRAK

Pada dasarnya dalam pengoperasian gardu portal konvensional masih memiliki kendala yaitu dari segi keamanannya. Hal ini dikarenakan tidak adanya pengaman pada gardu ketika dioperasikan secara manual serta tidak terdapat pendeteksian tegangan ketika petugas telah melakukan pemadaman gardu ketika dilakukan pemeliharaan. sehingga berpostensi dapat membahayakan operator. Berdasarkan kondisi tersebut maka dibuatlah "Sistem Pelaporan Pengoperasian Gardu Portal Konvensional Satu Fasa dalam Kondisi Pemeliharaan". Alat ini terdiri dari mikrokontroller Arduino yang dihubungkan dengan sensor peak detektor, modem GSM SIM 900, modul bluetooth HC-05, dan RTC. Data dari mikrokontroller yang terhubung dengan sensor peak detektor dan RTC akan dikirim ke android untuk dimonitoring dengan memanfaatkan media bluetooth HC-05, sehingga kondisi dari saklar relai dan tegangan *output* yang terdapat pada gardu dapat dipantau oleh operator PLN serta dilaporkan ke operator pusat menggunakan media SMS. Pelaporan tersebut akan mengirimkan SMS berupa kondisi dari gardu setelah dioperasikan oleh operator dan ketika gardu terjadi gangguan.

Dari tugas akhir yang dibuat, alat tersebut dapat melakukan monitoring kondisi dari saklar relai, tegangan *output* pada prototipe gardu, dan RTC. Sedangkan berdasarkan hasil pengujian dengan alat, kapasitas modul bluetooth HC-05 untuk mengirimkan data adalah maksimal 7 meter serta untuk pelaporan kondisi gardu alat ini dapat mengirimkan SMS dengan rata-rata waktu 14,50 detik. Hal ini tergantung dari baik buruknya sinyal *provider* yang digunakan.

Kata Kunci : Gardu Konvensional, Modem GSM SIM900, SOP (*Standart Operating procedure*)

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

REPORTING SYSTEM CONVENTIONAL PORTAL SUBSTATIONS ONE PHASE IN MAINTENANCE CONDITION

Name : Lutfi Kurnia Hilmy
Advisor I : Ir. Sjamsul Anam, MT.
Advisor II : Daniar Fahmi ST., MT.

ABSTRACT

Basically in the operation of conventional portal substations has a problem with its security. This is because there is no safety at the substation when operated manually and there is no voltage detector when the officer has extinguished the substation during maintenance so it can endanger the operator. Based on these conditions then it is made "Reporting System Conventional Portal Substations One Phase In Maintenance Conditions". This tool consists of an arduino microcontroller connected to a peak detector sensor, GSM SIM 900 modem, HC-05 bluetooth module, and RTC. Data from microcontroller that connected with peak detector and RTC sensors will be sent to android for monitoring with bluetooth HC-05. So, the conditions of relay switch and the output voltage in substation can be monitored by operator PLN and reported to central operator using SMS. That reporting will send SMS conditions from substation after operated by operator and when substation happened problem.

From the final project has been created, that tool can monitor condition of relay switch, output voltage on substation prototype, and RTC. While based on the test results with the tool, capacity of HC-05 bluetooth module to transmit data maximum is 7 meters and for reporting substation condition of this tool can send SMS with average time of 14.50 seconds. This depends on both the bad signal provider is used.

Keywords : *Conventional Substation, Modem GSM SIM900, SOP (Standart Operating procedure)*

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

KATA PENGANTAR

Puji syukur kehadirat Allah Yang Maha Esa atas segala rahmat, hidayah, dan inayah-Nya sehingga penulis dapat melaksanakan serta menyelesaikan tugas akhir ini dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Penulisan tugas akhir ini disusun sebagai syarat untuk menyelesaikan pendidikan diploma di Jurusan Teknik Elektro Otomasi, Institut Teknologi Sepuluh Nopember Surabaya. Adapun judul yang diangkat dalam tugas akhir ini adalah:

SISTEM PELAPORAN PENGOPERASIAN GARDU PORTAL KONVENSIONAL SATU FASA DALAM KONDISI PEMELIHARAAN

Dalam tugas akhir ini sistem pelaporan dengan menggunakan android dan SIM900 yang dikoordinasikan dengan *prototype* gardu konvensional.

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Ir. Sjamsul Anam, MT. dan Bapak Daniar Fahmi ST, MT. Atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya tugas akhir ini. Penulis juga mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Tugas akhir ini.

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

Surabaya, 20 Juli 2017



Penulis

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

DAFTAR ISI

HALAMAN

HALAMAN JUDUL.....	i
HALAMAN JUDUL.....	i
PERNYATAAN KEASLIAN TUGAS AKHIR.....	Error! Bookmark not defined.
HALAMAN PENGESAHAN	Error! Bookmark not defined.
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL.....	xxiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Batasan Masalah.....	3
1.4 Tujuan	3
1.5 Metodologi Penelitian	3
1.6 Sistematika Laporan.....	4
1.7 Relevansi.....	5
BAB II TEORI DASAR	7
2.1 Tinjauan Pustaka	7
2.2 Sistem Distribusi Tenaga Listrik.....	8
2.2.1 Struktur Jaringan Distribusi.....	8
2.3 Jenis- Jenis Gardu	9
2.3.1 Gardu Distribusi	9
2.3.2 Gardu Hubung	9
2.3.3 Gardu Cantol	9
2.3.4 Gardu Beton	10
2.3.5 Gardu Kios	10
2.3.6 Gardu Trafo.....	10
2.3.7 Gardu <i>Open Type</i> (Gardu Sel).....	11
2.3.8 Gardu <i>Close Type</i> (Kubikel).....	12
2.4 Gardu Portal Konvensional	12

2.5	Konstruksi Gardu Konvensional.....	13
2.6	Sistem Pengoperasian Gardu Konvensional 20 KV	16
2.6.1	Tujuan SOP.....	16
2.6.2	Pelaporan Pengoperasian Gardu Konvensional	17
2.6.3	Petunjuk / Langkah-Langkah Pengoperasian Gardu	17
2.7	Pemeliharaan Gardu Konvensional 20 KV	18
2.7.1	Macam-macam Pemeliharaan	19
2.8	Board Arduino Mega	20
2.8.1	Arduino IDE.....	21
2.9	RTC (Real Time Clock).....	22
2.10	Sensor Peak Detektor	23
2.11	LCD (<i>Liquid Crystal Display</i>)	24
2.12	Modem GSM SIM 900	25
2.13	Modul Bluetooth	26
2.14	App Inventor	27
2.15	Database.....	28
BAB III PERANCANGAN DAN PEMBUATAN ALAT		31
3.1	Perancangan <i>Hardware</i>	34
3.2	Pembuatan <i>Hardware</i>	34
3.2.1	LCD (<i>Liquid Crystal Display</i>)	35
3.2.2	Rangkaian Peak Detektor.....	36
3.2.3	Rangkaian <i>Power Supply</i>	37
3.2.4	Rangkaian RTC.....	38
3.2.5	Rangkaian Modem GSM SIM 900	38
3.2.6	Rangkaian Modul Bluetooth	39
3.3	Pembuatan <i>Software</i>	39
3.3.1	Pembuatan Sistem Arduino IDE	40
3.4	Pemrograman App Inventor.....	44
3.5	Perancangan Database	50
BAB IV PENGUJIAN DAN ANALISA DATA		53
4.1	Pengujian <i>Power Supply</i>	53
4.2	Pengujian Arduino Mega 2560	55
4.3	Pengujian LCD	57
4.4	Pengujian RTC.....	58
4.5	Pengujian dengan Sensor Peak Detektor	59
4.6	Pengujian Komunikasi dengan Modem GSM SIM 900.....	60
4.7	Pengujian Modul <i>Buck</i> Konverter.....	63
4.8	Pengujian dengan Modul Bluetooth.....	64

4.9 Pengujian Komunikasi Arduino dengan MIT Android	66
4.10 Pengujian Keseluruhan Alat	67
4.10.1 Pengujian dengan Kondisi Pertama	68
4.10.2 Pengujian dengan Kondisi Kedua	71
4.10.3 Pengujian dengan Kondisi Ketiga	87
4.10.4 Pengujian dengan Kondisi Keempat	90
4.11 Analisa relevansi	93
BAB V PENUTUP	95
5.1 Kesimpulan	95
5.2 Saran	95
DAFTAR PUSTAKA	97
LAMPIRAN A	A-1
A.1 Listing Program Arduino IDE pada Arduino Mega	A-1
A.1.1 Program Uji Pin Arduino Mega2560.....	A-1
A.1.2 Program Uji LCD	A-4
A.1.3 Program Uji RTC	A-4
A.1.4 Program Uji Komunikasi dengan SIM900	A-6
A.1.5 Program Database PHP MyAdmin.....	A-7
A.1.6 Program Uji Komunikasi dengan SIM900	A-8
A.1.7 Program Utama	A-9
A.2 Program Block pada App Inventor.....	A-22
LAMPIRAN B	B-1
B.1 Datasheet ATmega 2650.....	B-1
B.2 Datasheet RTC DS1307	B-6
B.3 Datasheet Modem GSM SIM900.....	B-9
LAMPIRAN C	C-1
C.1 Pototipe Gardu	C-1
C.2 Panel Box	C-1
C.3 Prototipe Simulasi Gangguan.....	C-2
LAMPIRAN D	D-1
D.1 Riwayat Hidup Penulis.....	D-1

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

DAFTAR GAMBAR

HALAMAN

Gambar 2.1	Gardu Cantol.....	10
Gambar 2.2	Gardu Beton.....	10
Gambar 2.3	Gardu Trafo.....	11
Gambar 2.4	Gardu <i>Open Type</i>	11
Gambar 2.5	Gardu <i>Close Type</i>	12
Gambar 2.6	Sel <i>Incoming</i> pada Gardu Konvensional.....	14
Gambar 2.7	Sel Metering.....	15
Gambar 2.8	Sel <i>Outgoing</i>	16
Gambar 2.9	Board Arduino Mega	21
Gambar 2.10	Jendela Arduino IDE.....	21
Gambar 2.11	Skematik dari RTC	23
Gambar 2.12	Rangkaian Penyearah Setengah Gelombang.....	24
Gambar 2.13	Gambar LCD 16x2.....	25
Gambar 2.14	Modem GSM SIM 900	26
Gambar 2.15	Modul Bluetooth.....	27
Gambar 2.16	Tampilan App Inventor.....	28
Gambar 3.1	Diagram Fungsional.....	31
Gambar 3.2	Prototipe Gardu.....	32
Gambar 3.3	Prototipe Panel.....	32
Gambar 3.4	Perancangan Alat	34
Gambar 3.5	Tata Letak Alat	35
Gambar 3.6	Rangkaian Skematik LCD	36
Gambar 3.7	Rangkaian Peak Detektor.....	37
Gambar 3.8	Skema Modul <i>Power Supply</i> dengan Arduino Mega	37
Gambar 3.9	Skema Rangkaian modul RTC dengan Arduino Mega.....	38
Gambar 3.10	Perancangan SIM 900 dengan Arduino Mega dan RTC.....	39
Gambar 3.11	Skema Pembuatan Modul Bluetooth dengan Arduino	39
Gambar 3.12	Flowchart 1 Program Arduino	42
Gambar 3.13	Flowchart 2 Program Arduino	43
Gambar 3.14	Flowchart 3 Program Arduino	44
Gambar 3.15	Tampilan Screen 1 pada Android.....	45
Gambar 3.16	Flowchart Pemrograman <i>Screen 1</i>	45
Gambar 3.17	Tampilan <i>Screen 2</i> pada Aplikasi	46

Gambar 3.18	Flowchart Pemrograman <i>Screen 2</i>	46
Gambar 3.19	Tampilan Database pada Aplikasi	47
Gambar 3.20	Tampilan <i>Screen</i> Jadwal Pemeliharaan	48
Gambar 3.21	Tampilan Monitoring pada Aplikasi.....	49
Gambar 3.22	Pemrograman Flowchart <i>Screen</i> Monitoring	49
Gambar 3.23	Tampilan Php MyAdmin pada Android	50
Gambar 3.24	Tampilan Tabel Pada PHP MyAdmin	51
Gambar 4.1	<i>Flowchart</i> Pemrograman Pengujian Tegangan pada PinArduino Mega 2560	54
Gambar 4.2	Proses Pengujian Arus <i>Power Supply</i> dengan Arduino ..	54
Gambar 4.3	Proses Pengujian Tegangan Power Supply dengan Arduino.....	56
Gambar 4.4	Pengujian pada LCD.....	57
Gambar 4.5	Flowchart Pemrograman Arduino untuk LCD	57
Gambar 4.6	Flowchart Pemrograman Arduino Mega untuk RTC.....	58
Gambar 4.7	Pengujian RTC	58
Gambar 4.8	Pengujian Sensor Peak Detektor.....	60
Gambar 4.9	Flowchart Pemrograman Pengiriman SMS	61
Gambar 4.10	Tampilan Serial Monitor Program Pengiriman SMS SIM 900	62
Gambar 4.11	Hasil Pengujian SIM 900.....	62
Gambar 4.12	Hasil Pengujian Waktu Pengiriman SMS	63
Gambar 4.13	Pengujian Modul Buck Konverter	64
Gambar 4.14	Android Mendeteksi Keberadaan Modul Bluetooth	64
Gambar 4.15	Proses Pairing Modul Bluetooth dengan Android	65
Gambar 4.16	Modul Bluetooth Sudah Terpasang dengan Android.....	65
Gambar 4.17	Pengujian Koneksi Antara Arduino Mega dengan Aplikasi Android	66
Gambar 4.18	Flowchart Pemrograman <i>Interface</i> dengan Android.....	67
Gambar 4.19	Kondisi Normal Gardu pada Layar Android	69
Gambar 4.20	Kondisi Gardu Normal pada Layar LCD.....	70
Gambar 4.21	Pembukaan Saklar CB	72
Gambar 4.22	Status LCD Industri Padam	73
Gambar 4.23	Pembukaan Saklar LBS <i>Incoming</i>	74
Gambar 4.24	Status LCD Gardu Padam.....	75
Gambar 4.25	Pembukaan Saklar LBS <i>Outgoing</i>	76
Gambar 4.26	Status LCD <i>Maintenance Ready</i>	77
Gambar 4.27	Gardu Bebas Tegangan.....	78
Gambar 4.28	Pengiriman SMS Setelah Gardu Bebas Tegangan.....	78

Gambar 4.29	Penutupan Saklar LBS Outgoing	80
Gambar 4.30	Status LCD Masukkan LBS <i>Incoming</i>	81
Gambar 4.31	Penutupan Saklar pada LBS <i>Incoming</i>	82
Gambar 4.32	Status LCD Gardu Normal	83
Gambar 4.33	Penutupan Saklar pada CB	84
Gambar 4.34	Pengiriman SMS Setelah Gardu Telah Dilakukan Pemeliharaan	85
Gambar 4.35	Database Pengoperasian Gardu Konvensional	87
Gambar 4.36	Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Beroperasi Secara Normal pada Tanggal 13 Januari 2014	88
Gambar 4.37	Kondisi Lampu Beban pada Tanggal 13 Januari 2014	88
Gambar 4.38	Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Beroperasi Secara Normal pada Tanggal 14 Juni 2014	89
Gambar 4.39	Kondisi Lampu Beban pada Tanggal 14 Juni 2014	89
Gambar 4.40	Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Padam pada Tanggal 15 Maret 2016	90
Gambar 4.41	Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Padam pada Tanggal 22 April 2017	91
Gambar 4.42	Kondisi Gardu dengan Lampu Beban dalam Kondisi Padam	91
Gambar 4.43	Tampilan LCD Untuk Kondisi Gardu Terjadi Gangguan	92
Gambar 4.44	Pengiriman SMS Indikasi Terjadi Gangguan pada Gardu	92
Gambar 4.45	Pengiriman SMS Gardu Telah Selesai Diperbaiki	93

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

DAFTAR TABEL

	HALAMAN
Tabel 2.1	Spesifikasi Modem GSM SIM 90025
Tabel 2.2	Konfigurasi Pin Modul Bluetooth.....26
Tabel 2.3	Menu pada App Inventor27
Tabel 2.4	Menu pada Designer28
Tabel 2.5	Pemrograman PHP MyAdmin29
Tabel 4.1	Pengujian <i>Power Supply</i>53
Tabel 4.2	Pengujian Keluaran Tegangan Pin pada Arduino Mega 256055
Tabel 4.3	Pengujian Data Waktu pada RTC59
Tabel 4.4	Pengujian Tegangan Output Sensor Peak Detektor59
Tabel 4.5	Pengujian Tegangan Output Sensor Peak Detektor untuk eban R dan RL59
Tabel 4.6	Tegangan <i>Output Buck</i> Konverter.....63
Tabel 4.7	Pengujian Jarak Koneksi Modul Bluetooth.....65
Tabel 4.8	Jadwal Pemeliharaan Gardu Konvensional68
Tabel 4.9	Waktu Pengiriman SIM90086

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

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam sistem penyaluran daya pada pelanggan industri tegangan menengah selalu dilengkapi dengan adanya panel yang digunakan sebagai pembagi, pemutus, pengontrol, dan sebagai sistem proteksi. Sehingga dalam kurun waktu tertentu perlu diadakan pemeliharaan yang dilakukan oleh PLN pada setiap panel yang terdapat di masing-masing industri menengah dengan tujuan untuk menjaga peralatan listrik yang ada didalamnya agar keandalannya tetap terjaga dan dapat beroperasi dalam jangka waktu yang lama.

Umumnya panel yang terdapat pada pelanggan industri dibedakan menjadi 2 tipe yaitu tipe terbuka (konvensional) dan tipe tertutup (kubikel). Untuk tipe tertutup tergolong aman ketika dioperasikan karena pada panel tipe tertutup sudah memiliki sistem pengamanannya sendiri (interlock) dan memiliki lampu indikator yang menampilkan status dari gardu. Sedangkan untuk tipe terbuka, cenderung lebih beresiko ketika dilakukan pemeliharaan daripada panel dengan tipe tertutup, hal ini dikarenakan tidak tersedianya pengaman pada gardu dalam artian pada beberapa gardu konvensional yang terdapat pada pelanggan industri memiliki sistem grounding yang terpisah dengan *body* gardu sehingga ketika dilakukan pemadaman oleh petugas operator, petugas harus melakukan pengecekan secara berulang untuk mengetahui apakah tegangan sisa yang terdapat di masing-masing sambungan sudah hilang atau tidak, serta melakukan pengecekan saat proses pemeliharaan yakni apakah sudah dilakukan penyambungan ke *ground* untuk sisi *incoming*, *metering*, maupun *outgoing* atau belum. Sehingga dengan demikian ketika operator tidak melaksanakan pengoperasian berdasarkan SOP yang ada atau dengan kata lain terdapat kesalahan saat melakukan proses pengoperasian maka hal ini tentu membahayakan operator dan akan mengganggu kinerja dari sistem.

Sehingga tugas kami sebagai mahasiswa Teknik Elektro Otomasi adalah dengan membuat “Sistem Pelaporan Pengoperasian Gardu Portal Konvensional Satu Fasa dalam Kondisi Pemeliharaan” yang fungsinya adalah untuk melaporkan setiap tahapan pengoperasian yang dilakukan oleh petugas ke dalam aplikasi android serta melaporkan status

pemeliharaan yang dilakukan oleh operator dan ada tidaknya gangguan yang terjadi pada gardu konvensional melalui media SMS. Hal ini dilakukan dengan tujuan agar keamanan gardu dan keselamatan operator ketika melakukan pengoperasian gardu tetap terjaga.

Dalam sistem ini digunakan prototipe panel yang dilengkapi dengan adanya sensor peak detektor sebagai pendeteksi buka tutupnya saklar LBS, SIM 900 yang digunakan sebagai indikator pengiriman SMS kepada operator pusat ketika pengoperasian akan dimulai, selesai dilakukan, dan ketika gardu konvensional sedang mengalami gangguan dengan kondisi diluar jadwal pemeliharaan, RTC yang digunakan sebagai penampil waktu jadwal pemeliharaan gardu, LCD sebagai penampil status dari setiap tahapan pengoperasian gardu yang dilakukan oleh operator, serta lampu yang digunakan sebagai indikator akan dimulai dan diakhirinya suatu proses pengoperasian. Prototipe panel ini selanjutnya akan berkoordinasi dengan prototipe gardu sebagai prototipe pengoperasian, dimana dalam prototipe gardu ini dilengkapi dengan adanya lampu indikator beban pelanggan. Sensor peak detektor ini dipasang pada CB, LBS *incoming*, dan LBS *outgoing* untuk mendeteksi buka tutupnya saklar pada masing-masing kontak hubung tersebut sehingga setiap perubahan status dari CB, LBS *incoming*, dan LBS *outgoing* akan diproses oleh arduino dan dikimkan ke android untuk dimonitoring. Sementara untuk memantau ada tidaknya gangguan pada gardu konvensional digunakan SIM 900 yang akan mengirimkan sms kepada operator ketika terjadi gangguan yang ditandai dengan padamnya lampu beban. Selanjutnya hasil data dari monitoring tersebut akan di simpan ke dalam database petugas operator.

1.2 Permasalahan

Untuk pembuatan tugas akhir ini, yang menjadi permasalahan adalah:

1. Selama ini sistem pelaporan kondisi gardu portal yang ada di PLN tergolong lambat karena harus menunggu pemberitahuan dari pelanggan terlebih dahulu.
2. Belum terdapat sistem monitoring kondisi relai yang terdapat pada gardu portal.
3. Dalam pembuatan membuat aplikasi android sebagai media interface yang ditunjang dengan adanya database untuk menyimpan riwayat pengoperasian gardu oleh operator.

1.3 Batasan Masalah

Dalam pembuatan tugas akhir ini, diberikan batasan permasalahan agar pembahasan tidak meluas dan menyimpang dari tujuan. Adapun batasan masalah dalam tugas akhir ini adalah :

1. Penggunaan prototipe sebagai ujicoba hanya sebatas pada tegangan 220V.
2. Pembuatan protortipe dengan menggunakan jaringan satu fasa
3. Alat ini menggunakan Arduino Mega sebagai kontroller dan pengolah data serta modem GSM pada SIM900 sebagai media komunikasi.
4. Media monitoring sensor tegangan dan tahapan pengoperasian gardu berupa aplikasi MIT pada android sebagai media antarmuka (interface).

1.4 Tujuan

Tujuan dari penyusunan dan penulisan tugas akhir ini adalah sebagai berikut:

1. Memantau pengoperasian saklar pemutus ketika proses pemeliharaan gardu portal 20 kV.
2. Menerapkan teknologi berbasis android untuk mempercepat pemberian informasi kepada operator agar operator dapat mengetahui ada tidaknya gangguan pada gardu dan untuk mengetahui apakah SOP pemeliharaan yang dilakukan sudah berjalan dengan benar atau tidak.

1.5 Metodologi Penelitian

Penelitian ini dilakukan melalui beberapa tahapan metodologi yaitu, tahap persiapan, tahap perancangan *hardware* dan *software*, tahap pengujian dan analisis, dan yang terakhir adalah tahap penyempurnaan alat.

Pada tahap persiapan akan dilakukan studi literatur yang mempelajari konsep dasar mengenai gardu konvensional 20 kV beserta SOP pemeliharaannya, sistem pemrograman MIT pada android, sistem komunikasi antara *hardware* (Arduino) dengan handphone sebagai media display, dan sistem komunikasi dengan memanfaatkan modul GSM SIM 900. Pada tahap perancangan *hardware* dan *software* dilakukan perancangan sesuai dengan data yang telah didapatkan melalui studi literatur. Pada tahap pengujian dan analisis dilakukan pengujian alat yang telah dibuat, tahap pengujian ini meliputi pengujian

hardware maupun *software* sedangkan untuk tahap analisa akan dilakukan analisa mengenai faktor apa saja yang menyebabkan alat tidak bekerja sesuai dengan keinginan sehingga masalah tersebut dapat segera diantisipasi. Untuk tahap akhir yaitu dilakukan penyempurnaan pada alat dan membenahi alat ketika terjadi error yang didasarkan pada data yang telah didapat pada analisa.

1.6 Sistematika Laporan

Pembahasan tugas akhir ini akan dibagi menjadi lima bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

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

Bab II Teori Dasar

Bab ini menjelaskan tentang tinjauan pustaka, penjelasan mengenai tipe gardu distribusi, konsep dasar mengenai gardu konvensional 20 kV, metode pengoperasian gardu konvensional dan standar operasi gardu konvensional, pemeliharaan gardu konvensional, sistem pemrograman MIT pada android, sistem komunikasi dengan modul GSM SIM 900, dan sistem database berbasis web .

Bab III Perancangan Sistem

Bab ini membahas desain dan perancangan *software* maupun *hardware* dari sistem pelaporan pengoperasian gardu portal konvensional satu fasa.

Bab IV Simulasi, Implementasi dan Analisis Sistem

Bab ini memuat hasil simulasi dan implementasi serta analisa dari hasil tersebut.

Bab V Penutup

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.7 Relevansi

Dengan adanya tugas akhir ini diharapkan dapat dijadikan sebagai metode untuk mempermudah petugas PLN dalam melaksanakan pengoperasian gardu konvensional 20 KV, sehingga hal-hal yang dapat menyebabkan kecelakaan saat proses pengoperasian gardu dan kegagalan sistem jaringan dapat terminimalisir.

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

BAB II

TEORI DASAR

Dalam bab ini dijelaskan mengenai teori-teori yang mendasari perancangan dan pembuatan Tugas Akhir, dimana teori yang mendukung penyelesaian Tugas Akhir mengenai “Sistem Pelaporan Pengoperasian Gardu Portal Konvensional Satu Fasa dalam Kondisi Pemeliharaan” antara lain adalah sistem pengoperasian gardu konven, sensor tegangan, mikrokontroller, RTC, LCD, modem GSM SIM 900, modul bluetooth dan app inventor.

2.1 Tinjauan Pustaka

Masalah yang terdapat dalam gardu dengan tipe konvensional adalah sistem pengoperasiannya masih dilakukan secara manual serta tidak adanya sistem pengaman yang terdapat pada *body* gardu dalam artian gardu tipe konvensional ini memiliki sistem pentanahan yang terpisah dengan *body* sehingga petugas atau operator pemelihara harus melakukan pembuangan sisa muatan di sepanjang saluran yang masuk pada gardu secara berulang-ulang. Dengan demikian, ketika terjadi kesalahan dalam pemeliharaan gardu akan sangat membayakan operator. Oleh karena itu, untuk mencegah hal tersebut maka setiap tahapan dalam pemeliharaan ini harus dioperasikan berdasarkan SOP pemeliharaan sesuai dengan *standart* yang ditetapkan oleh PLN [1]. Setiap tahapan yang dilakukan oleh petugas operator ini kemudian akan dilaporkan dan dimonitoring dengan menggunakan android, sehingga apabila operator tidak melakukan pengoperasian gardu berdasarkan SOP yang ada maka kesalahan tersebut dapat langsung terdeteksi dalam aplikasi android sehingga hasil yang tertampil dalam android tidak akan menunjukkan data yang valid.

Sementara itu pada gardu konvensional [2] untuk memonitoring buka tutupnya saklar di masing-masing sel yang terdapat pada gardu konvensional, digunakan sensor *peak detector* sebagai pendeteksi pergerakan saklar apabila telah dilakukan proses pembebasan tegangan oleh operator ketika sedang melakukan pemeliharaan. Dengan adanya sensor *peak detector* ini petugas dapat mengetahui kondisi dari setiap saklar yang ada.

Pada tugas akhir ini akan dilakukan perancangan sistem pelaporan pengoperasian gardu portal konvensional. Sehingga setiap kondisi gardu dapat terpantau serta setiap tahapan pengoperasian yang dilakukan oleh

operator yang ditandai dengan hasil pembacaan dari sensor peak detektor dapat termonitoring dalam aplikasi android dan juga pada layar LCD dengan tujuan agar kesalahan akibat pengoperasian oleh petugas operator dapat diminimalisir serta kendalan sistem jaringan distribusi tetap terjaga.

2.2 Sistem Distribusi Tenaga Listrik

Sistem distribusi tenaga listrik adalah penyaluran energi listrik dari pembangkit tenaga listrik hingga sampai ke konsumen sesuai dengan tingkat tegangan yang diperlukan. Sistem tenaga listrik ini terdiri dari unit pembangkit, unit transmisi, dan unit distribusi.

2.2.1 Struktur Jaringan Distribusi

Sistem distribusi tenaga listrik pada dasarnya terdiri dari beberapa bagian yaitu:

1. Jaringan Subtransmisi
Merupakan jaringan yang menyalurkan daya yang berasal dari pembangkit ke beberapa gardu induk yang tersebar di beberapa daerah.
2. Gardu Induk atau Pusat Pembangkit Tenaga Listrik
Gardu induk ini digunakan untuk menerima daya listrik melalui jaringan subtransmisi yang kemudian disalurkan ke jaringan distribusi primer.
3. Jaringan Distribusi Primer
Merupakan bagian dari sistem distribusi yang menghubungkan antara gardu induk dengan gardu trafo distribusi. Jaringan primer terdiri dari jaringan hantaran udara dan jaringan bawah tanah.
4. Gardu Pembagi atau Gardu Distribusi
Merupakan suatu gardu yang menurunkan tegangan primer menjadi tegangan sekunder.
5. Jaringan Distribusi Sekunder
Merupakan jaringan yang menghubungkan gardu distribusi dengan konsumen atau pemakai.

2.3 Jenis- Jenis Gardu

Pada dasarnya dalam penyaluran daya pada pelanggan industri tegangan menengah dapat diketahui beberapa jenis gardu dengan konstruksi dan fungsi yang berbeda-beda. Adapun penjelasan dari beberapa gardu tersebut adalah sebagai berikut.

2.3.1 Gardu Distribusi

Gardu distribusi adalah suatu bangunan atau tempat yang terdiri dari beberapa peralatan tenaga listrik yaitu antara lain trafo distribusi, perlengkapan hubung bagi tegangan menengah (PHB-TM), dan perlengkapan hubung bagi tegangan rendah (PHB-TR). Peralatan-peralatan tersebut digunakan untuk menunjang pendistribusian kebutuhan daya listrik bagi pelanggan menengah maupun pelanggan rendah. Adapun fungsi dari gardu distribusi adalah sebagai berikut:

1. Menyalurkan atau meneruskan tenaga listrik tegangan menengah ke gardu distribusi lainnya dan ke gardu hubung.
2. Menyalurkan atau meneruskan tenaga listrik tegangan menengah ke konsumen tegangan rendah.
3. Menurunkan tegangan menengah menjadi tegangan rendah yang selanjutnya disalurkan ke konsumen tegangan rendah.

2.3.2 Gardu Hubung

Gardu hubung merupakan suatu gardu yang berfungsi sebagai penerima daya listrik dari gardu induk yang telah diturunkan menjadi tegangan menengah dan menyalurkan atau membagi daya listrik tersebut tanpa merubah tegangannya melalui jaringan distribusi primer (JTM) menuju ke gardu atau trafo distribusi. Pada dasarnya gardu hubung ini digunakan sebagai sarana *manuver* pengendali beban listrik jika terjadi gangguan aliran listrik, program pelaksanaan pemeliharaan atau untuk mempertahankan kontinuitas pelayanan.

2.3.3 Gardu Cantol

Merupakan gardu distribusi untuk pasangan terbuka (*outdoor*) dengan memakai konstruksi satu tiang. Pada gardu cantol seluruh trafo distribusi dan papah hubung bagi (PHB-TR) dicantolkan pada tiang yang sama. Sedangkan untuk bentuk dari gardu cantol tersebut dapat dilihat pada Gambar 2.1.



Gambar 2.1 Gardu Cantol

2.3.4 Gardu Beton

Merupakan suatu gardu yang seluruh komponen utama instalasinya seperti trafo dan peralatan proteksi terangkai di dalam sebuah bangunan sipil yang dibangun dan difungsikan dengan konstruksi pasangan batu dan beton. Untuk bentuk dari gardu beton dapat dilihat pada Gambar 2.2.



Gambar 2.2 Gardu Beton

2.3.5 Gardu Kios

Merupakan suatu gardu yang bangunan keseluruhannya terbuat dari plat besi dengan konstruksi seperti kios.

2.3.6 Gardu Trafo

Merupakan suatu gardu yang berfungsi sebagai trafo daya penurunan tegangan dari tegangan menengah ke tegangan rendah, dimana gardu ini digunakan untuk membagikan energi listrik kepada konsumen yang memerlukan tegangan rendah. Sedangkan untuk bentuk dari gardu trafo dapat dilihat pada Gambar 2.3.



Gambar 2.3 Gardu Trafo

2.3.7 Gardu *Open Type* (Gardu Sel)

Pada gardu distribusi tipe sel ini memiliki peralatan hubung terbuka, dimana konstruksi rel busbar terlihat dan biasanya terpasang dibagian atas. Untuk pembagian antara *incoming*, metering, dan *outgoing* biasanya digunakan sekat tembok untuk membatasi antar sel tersebut. Sementara itu ketika dioperasikan pisau-pisau dalam peralatan hubungnya dapat dengan mudah diamati dengan mata biasa baik dalam keadaan masuk (menutup) atau dalam keadaan keluar (membuka). Sementara itu bentuk dari gardu sel dapat dilihat pada Gambar 2.4.



Gambar 2.4 Gardu *Open Type*

2.3.8 Gardu *Close Type* (Kubikel)

Gardu *closed type* adalah gardu distribusi baik gardu trafo atau gardu hubung yang memiliki peralatan hubung tertutup. Dimana peralatan hubung baik untuk *incoming*, *outgoing*, pengamatan trafo, dan sebagainya ditempatkan dalam suatu lemari khusus yang tertutup sehingga bekerjanya pisau-pisau peralatan hubung tidak dapat dilihat langsung oleh mata. Gardu *close type* ini biasa disebut sebagai gardu kubikel. Untuk bentuk dari kubikel ini dapat dilihat pada Gambar 2.5.



Gambar 2.5 Gardu *Close Type*

2.4 Gardu Portal Konvensional

Dalam gardu portal konvensional, sistem pengoperasian untuk gardu ini masih dilakukan secara manual, dimana pada gardu ini tidak terdapat sistem pengaman seperti hanya yang ada pada kubikel sehingga saat dioperasikan oleh operator PLN kurang aman. Selain itu karena dalam gardu ini tidak terdapat indikator pendeteksi tegangan yang ada pada gardu sehingga operator harus mengecek secara berulang-ulang muatan sisa yang terdapat pada gardu apakah sudah hilang atau tidak. dari segi pelaporannya, untuk indikasi gardu ketika terjadi gangguan maka sistem pelaporannya masih tergolong lambat karena untuk mengetahui kondisi tersebut harus menunggu notifikasi dari pelanggan terlebih dahulu. Sehingga berdasarkan kondisi tersebut, maka dibuat suatu prototipe gardu portal konvensional yang dilengkapi dengan adanya sistem interlock sebagai pengaman, modem GSM SIM 900 untuk notifikasi status gardu konvensional terhadap operator pusat, dan sensor peak detektor sebagai pendeteksi kondisi relai apakah sudah terbuka atau tidak sehingga dari adanya sensor ini dapat diketahui *output* tegangan pada gardu konvensional.

2.5 Konstruksi Gardu Konvensional [3]

Gardu distribusi tipe konvensional termasuk ke dalam jenis gardu *open type* karena memiliki peralatan hubung yang terbuka, selain itu gardu jenis ini memiliki konstruksi rel yang terletak di bagian atas dan menghubungkan masing-masing sel *incoming*, metering, dan *outgoing*. Untuk pembagian antara *incoming*, metering, dan *outgoing* digunakan sekat tembok untuk membatasi antar sel tersebut. Adapun penjelasan antar sel pada gardu konvensional adalah sebagai berikut:

1. Sel *Incoming*

Sel ini digunakan untuk menghubungkan sisi sekunder trafo daya ke busbar 20 KV. Sedangkan dari busbar tenaga listrik tersebut akan dibagi ke tempat-tempat yang diperlukan. Busbar itu sendiri merupakan suatu rel penghubung antara sel yang satu dengan sel yang lain dimana posisi dari rel dari gardu sel ini terletak pada bagian atas, untuk menyalurkan tagangan dengan kapasitas besar busbar ini harus terbuat dari bahan aluminium atau tembaga, bahan tembaga ini dipilih karena memiliki tingkat korosi yang sangat kecil. Biasanya untuk busbar yang digunakan di panel-panel digunakan busbar telanjang (tanpa isolasi). Selain busbar tipe telanjang tanpa isolasi, terdapat juga busbar berisolasi yang disebut dengan istilah *fully insulated*. Isolasi tersebut biasanya terbuat dari bahan karet atau bahan EPDM yang digunakan untuk memenuhi tingkat ketahanan isolasinya, dimana untuk isolasi jenis ini tidak mudah terbakar dan apabila terbakar api akan cepat mati dengan sendirinya.

Sementara itu di dalam sel *incoming* juga terdapat LBS (*load break switch*) yang umumnya digunakan untuk memutus dan menyambungkan arus dan tegangan dalam keadaan berbeban, berbeda dengan CB yang akan langsung trip jika saluran yang diproteksi mengalami gangguan, untuk LBS ini dia hanya bisa dioperasikan secara manual (*open/close*) seperti *switch* pada umumnya. Untuk gambar sel *incoming* beserta LBSnya dapat dilihat pada Gambar 2.6.



Gambar 2.6 Sel *Incoming* pada Gardu Konvensional

2. Sel *Metering*

Fungsi dari sel *metering* adalah untuk pengukuran sehingga di dalam sel *metering* terdapat adanya PT . Selain itu terdapat juga fuse yang berfungsi untuk melindungi PT . PT (*Potential Transformer*) merupakan peralatan yang digunakan untuk membantu mengukur besarnya tegangan pada KWHmeter, selain itu alat ini juga digunakan untuk membantu dalam sistem proteksi pada *relay* UFR (*under frekuensi relay*) yang mendeteksi frekuensi dari tegangan tersebut. Fungsi dari PT ini adalah untuk mentransformasikan nilai tegangan yang besar ke tegangan yang kecil yang selanjutnya digunakan untuk pengukuran dan proteksi. Selain itu juga digunakan sebagai isolasi antara sisi tegangan yang diukur dengan alat ukur maupun alat proteksinya. Antara sel *metering* dan *incoming* dihubungkan dengan busbar tiga fasa dimana daya yang masuk ke *metering* akan melewati fuse kemudian menuju ke trafo tegangan $20 \text{ KV}\sqrt{3} / 100\sqrt{3}$ selanjutnya dari fuse daya akan menuju Voltmeter dan Frekuensimeter. Sementara itu untuk CT (*Current Transformer*) merupakan suatu alat yang digunakan untuk pengukuran arus dan sebagai proteksi terhadap arus lebih. Dimana dengan adanya CT ini banyaknya tenaga listrik yang telah terpakai dapat diketahui dengan menentukan besaran

faktor kalinya, dengan adanya faktor kali ini pemakaian tenaga listrik sebenarnya dari KWH meter dapat diketahui. Pada CT umumnya memiliki sisi sekunder yang terhubung dengan relai. Untuk bentuk dari sel metering dapat dilihat pada Gambar 2.7



Gambar 2.7 Sel Metering

3. Sel *Outgoing*

Sel ini digunakan sebagai penyalur daya dari busbar menuju ke beban. Dalam sel *outgoing* ini terdapat CB dan LBS. Sementara itu, dalam sel *outgoing* juga terdapat adanya CB yang berfungsi sebagai pemutus dengan media gas SF₆ dan juga sebagai proteksi terhadap arus lebih. CB ini dapat dioperasikan dalam keadaan berbeban maupun dalam keadaan tidak berbeban dimana dalam pengoperasiannya dapat dilakukan secara manual dan otomatis. Secara otomatis yaitu dengan mengkoordinasikan dengan *relay* sehingga apabila arus yang mengalir melebihi *setting* dari CB maka CB akan trip secara otomatis. Sementara itu, untuk bentuk dari sel *outgoing* dapat dilihat pada Gambar 2.8.



Gambar 2.8 Sel *Outgoing*

2.6 Sistem Pengoperasian Gardu Konvensional 20 KV

Pengertian pengoperasian gardu konvensional adalah merubah posisi keluar atau masuknya kontak hubung (LBS, PMT) yang ada pada gardu induk, gardu distribusi, dan gardu hubung untuk keperluan :

1. Pengaturan beban, pengoperasian jaringan baru, dan pekerjaan pemeliharaan.
2. Pengusutan gangguan pada jaringan 20 KV.
3. Persiapan sumber cadangan untuk acara khusus.
4. Pengaturan jaringan dalam rangka pengamanan bencana alam atau huru hara.

Sementara itu, untuk mengoperasikan gardu konvensional itu sendiri harus didasarkan pada SOP pengoperasian. SOP (*Standart Operation Prosedure*) merupakan suatu bentuk ketentuan tertulis yang berisi prosedur atau langkah-langkah kerja yang dipergunakan untuk melaksanakan suatu kegiatan. Sedangkan SOP pengoperasian gardu konvensional 20 KV berarti ketentuan tentang prosedur / langkah – langkah kerja untuk mengoperasikan gardu konvensional 20 KV pada pengoperasian instalasi atau jaringan distribusi 20 KV.

2.6.1 Tujuan SOP

Tujuan diadakannya SOP dalam pengoperasian gardu konvensional adalah dengan membuat peralatan yang ada di gardu bekerja atau tidak bekerja, diberikan aliran listrik maupun dipadamkan dari aliran arus listrik. Hal ini dilakukan agar dalam pengoperasian gardu memiliki dampak agar energi listrik yang dialirkan kepada

peralatan yang ada dalam gardu dapat dapat digunakan sesuai keperluannya. Akan tetapi apabila pengoperasian gardu konvensional tidak dilakukan dengan benar maka dapat menimbulkan bahaya baik pada peralatan, lingkungan, maupun pada personil yang mengoperasikannya. Adapun bahaya yang ditimbulkan antara lain adalah apabila pengoperasian tidak berdasarkan SOP yang ada maka akan menyebabkan bahaya pada sisi operator. Hal ini dikarenakan apabila muatan statis yang terdapat pada gardu mengenai manusia maka akan sangat berbahaya bahkan dapat menyebabkan kematian. Selain itu apabila pengoperasian gardu konvensional pada jaringan atau pada beban di sisi hulu dan di sisi hilir tidak berkoordinasi dengan baik maka akan menyebabkan kecelakaan pada pemakai listrik maupun pihak pemeliharaan.

2.6.2 Pelaporan Pengoperasian Gardu Konvensional

Setiap pengoperasian yang dilakukan di gardu konvensional maka setiap perubahannya harus dilaporkan ke pusat pengatur distribusi. Pelaporan ini dilakukan untuk memantau setiap pengoperasian yang dilakukan oleh petugas. Adapun hal-hal yang perlu dilaporkan adalah:

1. Perubahan posisi keluar masuk LBS / PMT / PMS harus di laporkan ke pusat pengatur distribusi.
2. Jam *start* pengeluaran dan pemasukan / LBS / PMT / PMS pada gardu konvensional harus dilaporkan menggunakan alat komunikasi radio yang disediakan dipusat pengatur distribusi / posko.

2.6.3 Petunjuk / Langkah-Langkah Pengoperasian Gardu

Dalam pengoperasian gardu konvensional ada beberapa tahapan yang harus dilakukan antara lain adalah:

- a. Menyiapkan peralatan yang dibutuhkan dalam pengoperasian gardu konven seperti kunci gardu, sepatu 20 KV, sarung tangan 20 KV, helm pengaman, lembar isolasi 20 KV, perlengkapan P3K, *multi tester* / AVO meter, megger 5.000 atau 10.000 Volt, megger 1.000 Volt, *phase squence indikacator*, *tool set*, radio komunikasi, lampu penerangan, *single line diagram*, kendaraan operasional.
- b. Membuka pintu gardu konvensional
- c. Membuka PMT

Tahapan ini dilakukan untuk memutus aliran beban yang terhubung dengan pelanggan.

d. Melepas DS pada SUTM

Tahapan ini dilakukan untuk membuat gardu bebas tegangan (tegangan yang ada di gardu sama dengan nol).

e. Membuka LBS pada setiap sel gardu

Tahapan ini dilakukan untuk memeriksa kesiapan gardu konvensional untuk dioperasikan pada suatu sistem jaringan.

Kegiatan ini dilakukan dengan tujuan untuk:

1. Mengukur tahanan isolasi alat hubung di masing-masing sel
2. Mengukur tahanan pembumian badan gardu konven
3. Mengukur tahanan kontak alat hubung
4. Mengukur keserempakan alat kontak
5. Mengukur tahanan isolasi (disebut megger) kabel yang akan diberi tegangan
6. Memeriksa kebenaran urutan phase kabel antara satu gardu ke gardu lain (disebut cek *phase*)

f. Menutup pintu gardu konvensional: tahap ini menandakan pekerjaan pemeriksaan telah dilakukan dan dengan hasil baik, berarti gardu konvensional siap dioperasikan.

g. Memasukkan kontak hubung (LBS,PMT), tahap ini berarti memasukkan tegangan dari saluran / penyulang ke busbar untuk sel *incoming* ke saluran busbar untuk kubikel *outgoing* dan dari busbar menuju ke pusat beban.

h. Mengeluarkan kontak hubung, tahap ini merupakan kebalikan dari tahap memasukkan kontak hubung.

2.7 Pemeliharaan Gardu Konvensional 20 kV [4]

Pemeliharaan adalah suatu kegiatan yang dilakukan untuk menjaga keandalan peralatan listrik dimana dalam melakukan suatu pemeliharaan diperlukan tahapan-tahapan yang meliputi perencanaan, pelaksanaan, pengendalian hingga evaluasi pekerjaan pemeliharaan pada suatu sistem. Sedangkan tujuan dilakukannya pemeliharaan antara lain adalah:

1. Aman (*safe*) bagi manusia maupun lingkungan
2. Andal (*reliable*)
3. Memiliki kesiapan (*availability*) tinggi
4. Memiliki performance yang baik
5. Memiliki umur (*life time*) sesuai desain
6. Memiliki biaya pemeliharaan (*cost*) yang efisien

2.7.1 Macam-macam Pemeliharaan

Pada dasarnya pemeliharaan gardu konvensional dibedakan menjadi 4 macam yaitu pemeliharaan rutin, pemeliharaan korektif, pemeliharaan prediktif, dan pemeliharaan darurat (*emergency*).

1. Pemeliharaan Rutin

Pemeliharaan rutin dimaksudkan untuk mempertahankan agar peralatan tetap sesuai dengan kemampuannya. Pemeliharaan rutin dibedakan dengan kurun waktu pelaksanaan pemeliharaan yaitu:

a. Harian (inspeksi)

Hal ini dilakukan dengan cara melakukan pemeriksaan visual keadaan kubikel dalam kondisi bertegangan, tujuannya adalah untuk memantau kondisi fisik peralatan sewaktu operasi apakah terdapat kelainan atau tidak, sehingga kesiapan operasi peralatan selalu terpantau. Pemeriksaan tersebut antara lain:

1. Pemeriksaan indikator pegas mekanik pada PMT sistem pegas.
2. Monitor tekanan Gas SF 6
3. Pemeriksaan visual terhadap benda asing, bunyi-bunyian dan bau-bauan.

b. Bulanan

Kegiatan yang dilakukan adalah pencatatan counter kerja PMT sekaligus pencatatan KWH

c. 6 bulanan dan Tahunan

Pemeliharaan ini dilakukan pada saat kondisi gardu padam. Adapun peralatan yang dipelihara antara lain adalah:

1. Pemeriksaan PMS
2. Pemeriksaan PMT
3. Pemeriksaan trafo CT dan PT
4. Pemeriksaan rel/busbar 20 kV
5. Pemeriksaan kabel 20 kV
6. Pemeriksaan peralatan pengukuran (Amperemeter, KV meter, KWH meter)

Kurun waktu ini disesuaikan dengan pengalaman operasi dari peralatan tersebut.

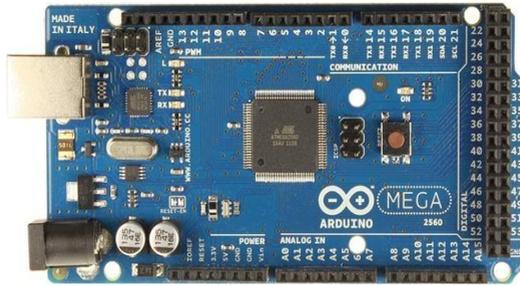
2. Pemeliharaan Korektif
Merupakan pemeliharaan yang bertujuan untuk meningkatkan keandalan peralatan.
3. Pemeliharaan Prediktif
Merupakan pemeliharaan yang dilakukan untuk meningkatkan frekuensi pemantauan terhadap peralatan instalasi. Hasil dari pemantauan ini merupakan input untuk memprediksi kelainan kinerja peralatan dan rencana perbaikannya.
4. Pemeliharaan Darurat
Merupakan suatu pekerjaan untuk memperbaiki jaringan yang rusak akibat adanya bencana alam, kebakaran, huru-hara dan sebagainya.

2.8 Board Arduino Mega [5]

Arduino mega 2560 adalah papan mikrokontroler Atmega2560 berdasarkan (*datasheet*) yang memiliki 54 digital pin *input / output* (dimana 15 dapat digunakan sebagai *output* PWM), 16 *analog input*, 4 UART (*hardware port serial*), *osilator* kristal 16 MHz, koneksi USB, *jack* listrik, *header* ICSP, dan tombol reset. Spesifikasi *board* Arduino berdasarkan *datasheet* adalah sebagai berikut :

1. Tegangan aktif 5V
2. Tegangan masukan 7-12V
3. Tegangan masukan (batas) 6-20V
4. Pin *input/output* digital 54 (6 dapat digunakan sebagai PWM)
5. Pin *Input Analog* 16
6. Arus DC pin I/O 40 mA
7. Arus DC pada 3,3V 50 mA
8. *Flash Memory* 128 KB (Atmega328), 0,5 KB digunakan *bootloader*
9. SRAM 8 KB (Atmega328)
10. EEPROM 4 KB (Atmega328)
11. *Clock Speed* 16 MHz

Bentuk dari *board* Arduino Mega ditunjukkan seperti pada Gambar 2.9.

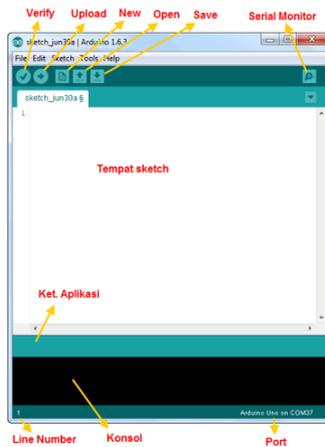


Gambar 2.9 Board Arduino Mega

Untuk menggunakan arduino ini dengan menghubungkannya dengan kabel USB atau power dengan adaptor AC-DC atau baterai.

2.8.1 Arduino IDE

Untuk memprogram *board* Arduino, dibutuhkan aplikasi IDE (*Integrated Development Environment*) yang merupakan bawaan dari Arduino itu sendiri. Aplikasi ini digunakan untuk memprogram *source code* Arduino atau biasa disebut sebagai *sketch*. *Sketch* merupakan source code yang berisi logika dan algoritma yang selanjutnya akan di *upload* ke dalam IC mikrokontroler (Arduino). Adapun tampilan bagian-bagian dari arduino IDE dapat dilihat pada Gambar 2.10.



Gambar 2.10 Jendela Arduino IDE

Berdasarkan Gambar 2.10. Di dapat penjelasan sebagai berikut:

1. *Verify* pada versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi di *upload* ke *board* Arduino, biasakan untuk memverifikasi terlebih dahulu *sketch* yang dibuat. Jika ada kesalahan pada *sketch*, nanti akan muncul *error*. Proses *Verify* / *Compile* adalah suatu proses mengubah *sketch* ke *binary code* untuk di *upload* ke mikrokontroler.
2. *Upload* tombol ini berfungsi untuk mengupload *sketch* ke *board* Arduino. Walaupun kita tidak mengklik tombol *verify*, maka *sketch* akan di *compile*, kemudian langsung di *upload* ke *board*. Berbeda dengan tombol *verify* yang hanya berfungsi untuk memverifikasi *source code* saja.
3. *New Sketch* berfungsi untuk membuka *window* dan membuat *sketch* baru
4. *Open Sketch* berfungsi untuk membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file *.ino*
5. *Save Sketch* untuk menyimpan *sketch*, tapi tidak disertai dengan mengcompile.
6. *Serial Monitor* untuk membuka *interface* untuk komunikasi serial
7. Keterangan Aplikasi merupakan pesan-pesan yang dilakukan aplikasi akan muncul di sini, seperti “*Compiling*” dan “*Done Uploading*” ketika kita mengcompile dan mengupload *sketch* ke *board* Arduino.
8. Konsol adalah pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada *sketch* yang kita buat, maka informasi *error* dan baris akan diinformasikan di bagian ini.
9. Baris *Sketch* adalah bagian yang akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.
10. Informasi *Port* adalah bagian yang menginformasikan *port* yang dipakai oleh board Arduino.

2.9 RTC (*Real Time Clock*) [6]

RTC (*Real Time Clock*) merupakan sebuah IC yang memiliki fungsi untuk menghitung waktu, mulai dari detik, menit, jam, tanggal, bulan, serta tahun. RTC ini mempunyai *clock* sumber sendiri dan *internal*

battery untuk menyimpan data waktu dan tanggal. Sehingga jika *system* komputer / *microcontroller* mati waktu dan tanggal didalam memori RTC tetap *uptodate*. RTC yang mudah dijumpai di pasaran adalah RTC DS1307. Adapun fitur-fitur yang dimiliki oleh DS 1307 adalah sebagai berikut:

1. *Real-time clock* (RTC) menghitung detik, menit, jam, tanggal, bulan, hari, dan tahun *valid* sampai tahun 2100.
2. Ram 56-byte, *nonvolatile* untuk menyimpan data.
3. 2 jalur *serial interface* (I2C).
4. *Output* gelombang kotak yg terprogram.
5. *Automatic power-fail detect and switch*.
6. Konsumsi arus hanya 500nA pada *battery internal*.
7. *Mode* dengan *oscillator running*.
8. *Temperature range* -40°C sampai +85°C

Sedangkan bentuk fisik dari RTC dapat dilihat pada Gambar 2.11 .



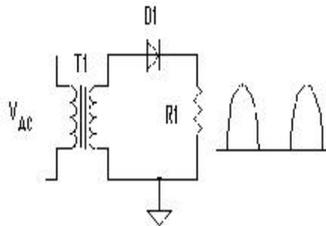
Gambar 2.11 Skematik dari RTC

2.10 Sensor Peak Detektor

Sensor peak detektor adalah sensor yang digunakan untuk mendeteksi besar tegangan yang ada pada peralatan listrik, sensor tegangan ini menggunakan trafo *step-down* dengan rangkaian penyearah. Transformator *step-down* ini memiliki lilitan sekunder lebih sedikit daripada lilitan primer, sehingga berfungsi sebagai penurun tegangan pada sisi primer terhadap sisi sekunder. Sehingga pada sisi sekunder digunakan untuk peralatan pengukuran. Sementara itu transformator itu sendiri bekerja berdasarkan prinsip induksi elektromagnetik. Tegangan masukan bolak-balik yang membentangi primer menimbulkan *fluks* magnet yang idealnya semua bersambung dengan lilitan sekunder. *Fluks* bolak-balik ini menginduksikan GGL

dalam lilitan sekunder. Jika efisiensi sempurna, semua daya pada lilitan primer akan dilimpahkan ke lilitan sekunder.

Sedangkan untuk rangkaian penyearah pada dasarnya dibedakan menjadi dua yaitu rangkaian penyearah setengah gelombang dan rangkaian penyearah gelombang penuh. Penyearah setengah gelombang menggunakan 1 dioda sedangkan penyearah gelombang penuh ada yang menggunakan 4 dioda dan 2 dioda tergantung dengan jenis trafo yang digunakan. Pada tugas akhir ini menggunakan trafo CT dan ditambahkan dengan menggunakan 1 dioda untuk penambahan penyearah gelombang. Rangkaian penyearah gelombang ini dapat dilihat pada Gambar 2.12.



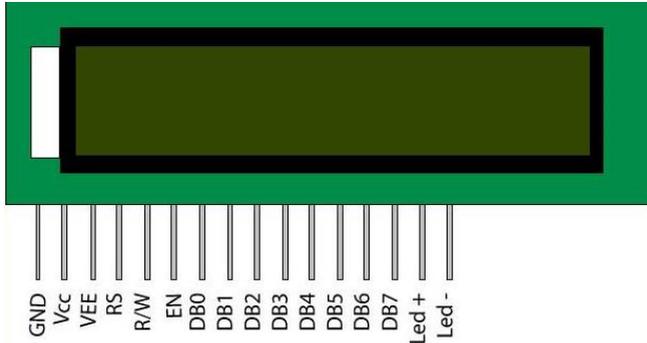
Gambar 2.12 Rangkaian Penyearah Setengah Gelombang

2.11 LCD (*Liquid Crystal Display*)

LCD (*Liquid Crystal Display*) merupakan alat penampil karakter yang dapat berupa huruf, angka, dan gambar. Hal ini dikarenakan pada LCD terdapat banyak sekali titik cahaya yang terdiri dari satu buah kristal cair sebagai sebuah titik cahaya. Adapun untuk modul LCD memiliki karakteristik sebagai berikut:

1. Terdapat 192 macam karakter
2. Bekerja pada suhu 0°C sampai 55°C
3. Terdapat 80×8 bit display RAM (maksimal 80 karakter)
4. Terdapat 16×2 karakter huruf yang bisa ditampilkan
5. Memiliki kemampuan penulisan dengan 8 bit maupun dengan 4 bit.
6. Dibangun dengan *osilator local*.
7. Satu sumber tegangan 5 Volt.
8. Otomatis *reset* saat tegangan dihidupkan.

Untuk bentuk fisik dari LCD dapat dilihat pada Gambar 2.13.



Gambar 2.13 Gambar LCD 16x2

2.12 Modem GSM SIM 900

SIM900 GSM/GPRS *shield* merupakan modul GSM untuk Arduino yang berperan untuk melakukan komunikasi antara arduino dengan android, bentuk komunikasi ini dapat berupa pengiriman dan penerimaan SMS, MMS, melakukan atau menerima panggilan. Modul SIM900 menggunakan *AT Command* untuk menjalankan fungsinya dan menggunakan komunikasi UART/Serial untuk berkomunikasi dengan Arduino. SIM900 memiliki fitur *Quad Band*, yang artinya mendukung jaringan GSM ber-frekuensi 850MHz, 900MHz, 1800MHz, dan 1900MHz sementara di Indonesia sendiri jaringan yang digunakan adalah 900/1800MHz sehingga sangat mendukung penggunaan SIM900. Modul ini menggunakan *protokol* komunikasi UART dalam berkomunikasi data dengan Arduino. Modul mempunyai 8 pin yang dapat digunakan untuk di gabungkan dengan arduino (pin 0 sampai pin 7) akan dipakai 2 pin sebagai pin RX dan TX yang akan digunakan pada komunikasi UART dengan Arduino. Sedangkan spesifikasi dari SIM900 terdapat pada Tabel di 2.1.

Tabel 2.1 Spesifikasi Modem GSM SIM 900

Spesifikasi	
<i>Dimension (with Antenna)</i>	110x58x19 mm
<i>Current</i>	Max 450 mA
<i>Voltage</i>	Max 5 Volt
Protokol Komunikasi	UART

Bentuk fisik dari SIM 900 dapat dilihat pada Gambar 2.14.



Gambar 2.14 Modem GSM SIM 900

2.13 Modul Bluetooth

Adalah sebuah modul yang digunakan untuk komunikasi serial *wireless (nirkabel)* yang mengkonversi port serial ke bluetooth. Modul ini dapat digunakan sebagai *slave* maupun *master*. Pada modul bluetooth memiliki 2 mode konfigurasi, yaitu *AT mode* dan *communication mode*. *AT mode* berfungsi untuk melakukan pengaturan konfigurasi dari HC-05. Sedangkan *communication mode* berfungsi untuk melakukan komunikasi bluetooth dengan piranti lain. Dalam penggunaannya, modul bluetooth dapat beroperasi tanpa menggunakan *driver* khusus. sementara untuk melakukan komunikasi antar bluetooth maka harus memenuhi dua kondisi sebagai berikut :

1. Komunikasi harus antara *master* dan *slave*.
2. *Password* harus benar (saat melakukan *pairing*).

Jarak sinyal dari modul bluetooth rata-rata adalah 30 meter, hal ini dapat berubah sesuai dengan tipe modul yang digunakan. Konfigurasi pin yang digunakan dapat dilihat pada tabel 2.1:

Tabel 2.2 Konfigurasi Pin Modul Bluetooth

No.	Nomor pin	Nama pin	Fungsi
1.	Pin 1	Key	-
2.	Pin 2	Vcc	Sumber tegangan 5V
3.	Pin 3	Gnd	<i>Ground</i> tegangan
4.	Pin 4	Txd	Mengirim data
5.	Pin 5	Rxd	Menerima data
6.	Pin 6	State	-

Sedangkan gambar fisik dari modul bluetooth dapat dilihat pada Gambar 2.15.



Gambar 2.15 Modul Bluetooth

2.14 App Inventor [7]

App Inventor adalah sebuah *tool* untuk membuat aplikasi android, *tool* ini dikembangkan *Massachusetts Institute of Technology* (MIT) dimana pada App Inventor digunakan teknik *visual programming*, berbentuk seperti susunan blok yang memiliki fungsi logika tertentu. App Inventor menggunakan antarmuka grafis, serupa dengan antarmuka pengguna pada *scratch* dan *starLogo* TNG, yang memungkinkan pengguna untuk melakukan proses *drag* dan *drop* objek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. Adapun bagian-bagian menu yang terdapat dalam App Inventor dapat ditunjukkan dalam tabel 2.2.

Tabel 2.3 Menu pada App Inventor

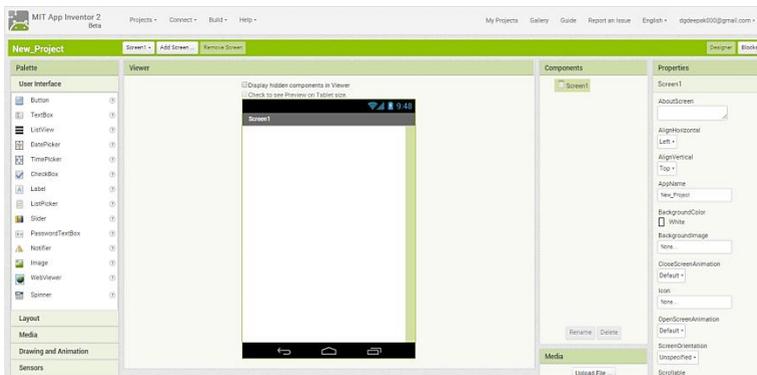
Menu pada App Inventor	Keterangan
<i>Project</i>	Adalah menu awal yang terdiri dari <i>start new project</i> , <i>save project</i> , <i>delete project</i> .
<i>Connect</i>	Merupakan menu untuk menghubungkan project yang telah dibuat dengan menggunakan media seperti <i>companion</i> , <i>emulator</i> , dan USB.
<i>Build</i>	Menu yang digunakan untuk mengunduh <i>project</i> yang telah dibuat agar dapat di <i>install</i> di Smartphone.
<i>Add Screen</i>	Menu untuk menambah dan menghapus <i>screen</i> .
<i>Menu Designer</i>	Menu yang digunakan untuk mendesain tampilan aplikasi sesuai dengan keinginan pengguna.
<i>Block</i>	Menu yang berisi beberapa kode program yang digunakan untuk pemrograman logika

Sedangkan untuk menu designer dibagi menjadi beberapa bagian seperti yang ditunjukkan dalam Tabel 2.3

Tabel 2.3 Menu pada Designer

Menu pada Designer	Keterangan
<i>Viewer</i>	Untuk menempatkan komponen dan mengaturnya sesuai tampilan yang diinginkan.
<i>Pallette</i>	Berisi seluruh komponen yang dapat dipakai untuk membuat aplikasi.
<i>Component List</i>	Merupakan tempat list komponen dari proyek yang dipakai.
<i>Media</i>	Berisi tempat untuk mengambil media audio dan gambar yang digunakan untuk proyek
<i>Properties</i>	Berfungsi untuk mengatur properties komponen yang digunakan, seperti <i>width</i> , <i>height</i> , <i>name</i> , dan sebagainya.

Sedangkan tampilan dari App Inventor dapat dilihat pada Gambar 2.16.



Gambar 2.16 Tampilan App Inventor

2.15 Database [8]

Database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Dalam pembuatan database digunakan aplikasi yang bernama

PHP MyAdmin. PHP MyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL melalui Jaringan (*World Wide Web*). Aplikasi yang dibangun menggunakan PHP umumnya akan memberikan hasil pada *web browser* tetapi prosesnya secara keseluruhan dijalankan pada *server*. Adapun penggunaan bahasa pemrograman yang digunakan oleh PHP dapat dilihat pada Tabel 2.5.

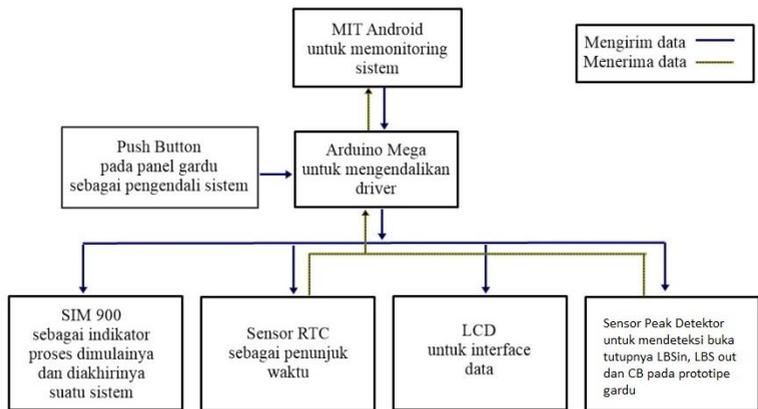
Tabel 2.5 Pemrograman PHP MyAdmin

Pemrograman pada php	Keterangan
<i>Sintaks</i>	Untuk menuliskan sintaks PHP harus diawali dengan tag <code><?></code> dan diakhiri dengan tag <code>?></code> . Sedangkan sintaks untuk menampilkan dalam <i>web browser</i> dapat menggunakan perintah <i>print</i> atau <i>echo</i> .
<i>Variabel</i>	<i>Variabel</i> dalam PHP digunakan untuk menyimpan suatu nilai atau data sementara seperti text, angka, atau <i>array</i> . Ketika sebuah variabel dibuat, variabel tersebut dapat dipakai berulang-ulang. Namun data yang disimpan dalam variabel akan hilang setelah program selesai dieksekusi. Pada PHP semua <i>variabel</i> harus dimulai dengan karakter '\$'. Variabel PHP tidak perlu dideklarasikan dan ditetapkan jenis datanya sebelum variabel tersebut digunakan. Panjang variabel tidak terbatas setelah diawali '\$' oleh huruf atau <i>under_score</i> (<code>_</code>), karakter berikutnya bisa terdiri dari huruf, angka, dan karakter tertentu yang diperbolehkan (karakter ASCII dari 127-255)
Konstanta	Konstanta merupakan variabel konstan yang nilainya tidak berubah-ubah. Untuk mendefinisikan konstanta dalam PHP menggunakan fungsi <i>define()</i> karena konstanta merupakan variabel yang nilainya tetap. Konstanta hanya diberi nilai pada awal program dan nilainya tidak pernah berubah selama program berjalan.

Pemrograman pada php	Keterangan
Operator dalam php	Operator merupakan simbol yang digunakan untuk memanipulasi data seperti penambahan dan pengurangan. Selain itu, operator juga digunakan untuk mengoperasikan operand baik tunggal atau lebih dari satu. Operator dibagi menjadi empat yaitu operator aritmatika, operator logika, operator perbandingan, operator penugasan.
Perulangan dalam php	<i>Loop</i> merupakan proses eksekusi operasi program secara berulang-ulang sampai ditemui kondisi atau batasan untuk mengakhiri eksekusi tersebut.

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Pada bab ini akan dibahas mengenai tahapan yang dilakukan dalam perancangan dan pembuatan tugas akhir terutama dalam pembuatan dan perancangan alat yang digunakan sebagai media komunikasi untuk menjalankan proses pengoperasian gardu portal konvensional. Bagian awal dari bab ini akan membahas mengenai blok dari tugas akhir yang dapat dilihat pada gambar berikut:



Gambar 3.1 Diagram Fungsional

Penjelasan Diagram Fungsional:

1. MIT App Inventor 2 android : Digunakan sebagai media untuk memonitoring sistem penoperasian gardu portal konvensional.
2. Arduino Mega : Digunakan untuk mengendalikan *driver* dan sensor.
3. Modem GSM SIM 900 : Digunakan sebagai media komunikasi yang digunakan sebagai penanda proses dimulainya dan diakhirinya suatu sistem pengoperasian gardu konvensional serta sebagai pemberi informasi ada tidaknya gangguan yang ada pada gardu.
4. Sensor RTC : Digunakan sebagai penunjuk waktu yang menandakan waktu sekarang saat gardu konvensional mulai dioperasikan dan gardu konvensional selesai dioperasikan.

5. LCD : Digunakan sebagai tampilan *interface* dari sensor RTC dan status gardu konvensional.
6. Sensor *Peak Detektor* : Digunakan sebagai pendeteksi buka tutupnya saklar pada *LBS incoming*, *LBS outgoing*, dan CB pada prototipe gardu.

Sementara itu, untuk bentuk dari prototipe gardu yang digunakan sebagai prototipe pengoperasian dan prototipe panel sebagai alat untuk memonitoring status dari gardu dapat dilihat pada Gambar 3.2 dan Gambar 3.3.



Gambar 3.2 Prototipe Gardu

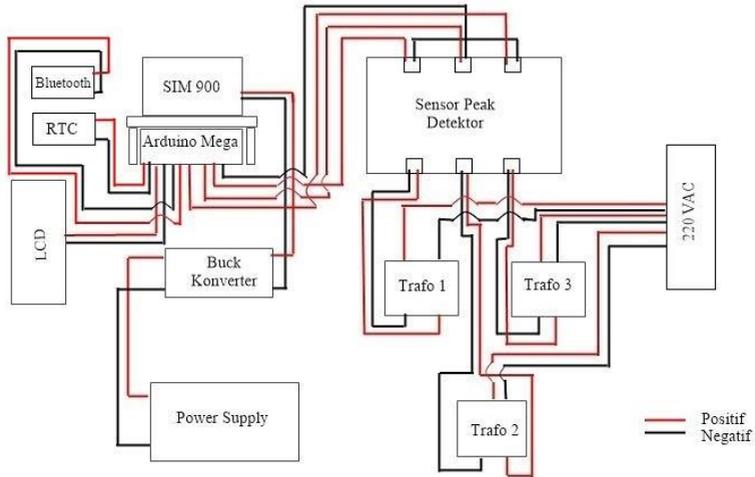


Gambar 3.3 Prototipe Panel

Secara umum sistem yang terdapat dalam Gambar 3.1 ini merupakan sistem kerja komunikasi berbasis android pada gardu portal konvensional. Dalam prototipe gardu yang terdapat pada Gambar 3.2, disediakan 6 tombol yakni dua tombol untuk *Circuit Breaker* yang meliputi tombol 1 untuk pembukaan CB dan tombol 4 untuk penutupan CB, tombol 2 untuk pembukaan LBS *incoming*, dan tombol 5 untuk penutupan LBS *incoming*, serta tombol 3 untuk pembukaan LBS *outgoing* dan tombol 6 untuk penutupan LBS *outgoing*. Selanjutnya ketika gardu konven mulai dilakukan pemeliharaan yang ditandai dengan proses pembukaan *Circuit Breaker* pada gardu maka ketika tombol 1 untuk CB ditekan oleh operator, maka sensor peak detektor akan mendeteksi pergerakan saklar pada CB sehingga dapat membaca tegangan yang selanjutnya akan diproses oleh arduino dan mengirimkan data tersebut pada layar android melalui modul bluetooth, sehingga pada layar akan tampil status dari CB apakah dalam kondisi terbuka atau tertutup, begitu juga ketika tombol 2, 3, 4, 5 dan 6 ditekan maka akan mendeteksi pergerakan saklar dari LBS *incoming* dan LBS *outgoing* dan arduino akan memproses data tegangan dan mengirimnya ke aplikasi android. Selanjutnya status dari masing-masing sambungan yang ada dalam gardu tersebut akan diproses oleh arduino dan ditampilkan ke layar LCD yang terletak pada prototipe panel seperti pada Gambar 3.2 dan layar android melalui modul bluetooth. Untuk sensor RTC pada dasarnya digunakan untuk menampilkan waktu saat ini yang selanjutnya juga akan ditampilkan pada MIT android serta pada LCD yang dipasang pada *prototype* panel, jadi ketika proses pemeliharaan gardu konven sedang dilaksanakan petugas dapat mengetahui waktu dan berapa lama pelaksanaan proses pengoperasian pada gardu portal konvensional. Untuk Modem GSM SIM 900 digunakan sebagai indikator dimulai dan diakhirinya suatu proses pengoperasian gardu konven, dalam artian ketika gardu konven mulai dioperasikan maka arduino akan memproses data sehingga modem GSM SIM 900 akan mengirimkan pesan berupa teks ke nomor petugas operator dan pengawas PLN begitu juga ketika gardu konven selesai dioperasikan modem GSM juga akan mengirim pesan berupa teks ke hp operator dan pengawas PLN. Begitu juga ketika kondisi gardu padam secara tiba-tiba ketika sedang tidak dilakukan pemeliharaan oleh petugas atau dalam artian gardu sedang mengalami gangguan maka SIM 900 akan mengirimkan notifikasi berupa teks kepada hp petugas.

3.1 Perancangan *Hardware*

Untuk perancangan *hardware* pada tugas akhir ini terdiri dari beberapa komponen yakni diantaranya rangkaian sensor peak detektor, *power supply*, Arduino Mega 2650, modul bluetooth HC-05, RTC, *buck konverter*, SIM 900, Trafo 1, Trafo 2, dan Trafo 3. Sementara itu untuk gambar perancangannya dapat dilihat pada Gambar 3.4.

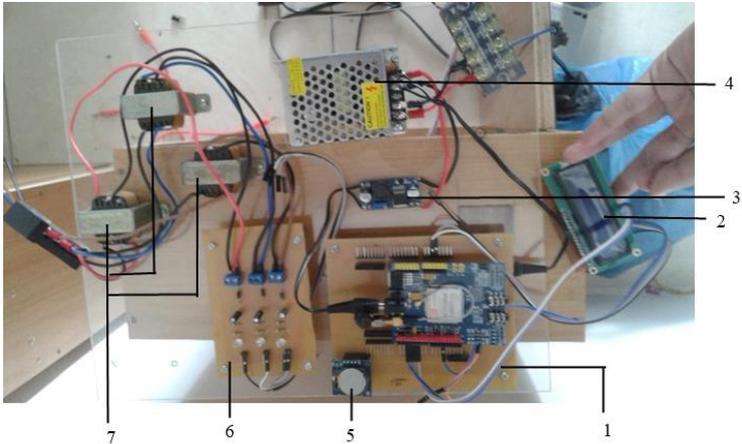


Gambar 3.4 Perancangan Alat

Dalam Gambar 3.4 susunan untuk arduino mega dan modul GSM SIM 900 dibuat bertingkat dengan tujuan untuk meminimalisir tempat yang ada sehingga bisa digunakan untuk meletakkan komponen lainnya. Dimana arduino mega diletakkan pada bagian bawah sedangkan modul GSM SIM 900 diletakkan pada bagian atas. Tata letak ini dibuat seperti demikian untuk mempermudah dalam pengoperasiannya selain itu agar menjadi rapi dan efisien.

3.2 Pembuatan *Hardware*

Pada pembuatan *hardware* untuk tugas akhir ini membahas mengenai rangkaian LCD, rangkaian sensor peak detektor, rangkaian *power supply*, modem GSM SIM 900, dan rangkaian RTC. Tampilan tata letak *hardware* dapat dilihat pada Gambar 3.5.



Gambar 3.5 Tata Letak Alat

Keterangan:

1. Rangkaian Arduino mega, diatasnya terdapat *board* yaitu Modem GSM SIM900
2. LCD 16x2
3. Modul *Buck* Konverter
4. Modul *Power Supply* 12 V
5. RTC
6. Rangkaian Peak Detektor
7. CT 1, CT 2, dan CT 3

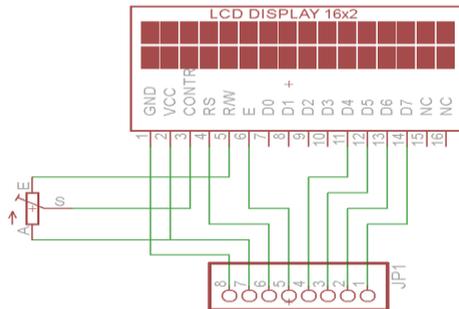
3.2.1 LCD (*Liquid Crystal Display*)

LCD yang digunakan pada tugas akhir ini adalah LCD dengan ukuran 16x2. 16x2 mewakili 16 kolom dan 2 baris, berarti LCD ini dapat menampilkan karakter pada sepanjang 32 buah dengan masing-masing kolom maksimal berisi 16 buah karakter. LCD ini membutuhkan *power supply* sebesar 5 Volt DC sebagai sumber dan pin yang digunakan pada LCD pada alat ini yaitu pin CONTR, RS, RW, E, D4, D5, D6, D7. Adapun perancangan rangkaian LCD dengan Arduino Mega adalah sebagai berikut:

1. Pin CONTR pada LCD dihubungkan dengan potensio 10k Ohm
2. Pin RS pada LCD dihubungkan dengan pin 12 pada arduino
3. Pin E pada LCD dihubungkan dengan pin 11 pada arduino

4. Pin D4 pada LCD dihubungkan dengan pin 5 pada arduino
5. Pin D5 pada LCD dihubungkan dengan pin 4 pada arduino
6. Pin D6 pada LCD dihubungkan dengan pin 3 pada arduino
7. Pin D7 pada LCD dihubungkan dengan pin 2 pada arduino
8. Pin RW pada LCD dihubungkan dengan pin GND pada arduino

Rangkaian LCD ini dapat dilihat pada Gambar 3.6.



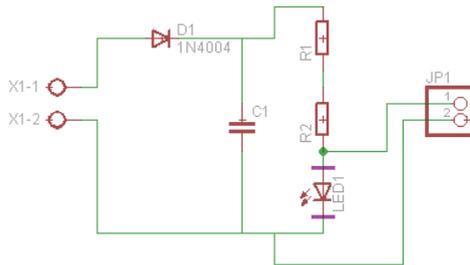
Gambar 3.6 Rangkaian Skematik LCD

Data yang akan ditampilkan pada LCD adalah data dari sensor RTC dan status kondisi gardu pada saat dilakukan proses pemeliharaan gardu konvensional.

3.2.2 Rangkaian Peak Detektor

Sensor Peak Detektor adalah alat yang digunakan untuk mendeteksi ada tidaknya tegangan yang melalui suatu peralatan listrik. Dimana untuk menurunkan nilai tegangan AC dari 220 V menjadi tegangan AC 6 V digunakan trafo *stepdown* yaitu trafo CT 350mA. Selanjutnya agar tegangan AC 6 V ini dapat dibaca oleh arduino maka dibutuhkan rangkaian pengkondisi sinyal yang berupa rangkaian penyearah untuk menyearahkan sinyal dari keluaran trafo yang masih berupa sinyal AC 6 V disearahkan menjadi sinyal DC 0-5 V agar dapat dibaca oleh arduino. Komponen penyearah gelombang penuh ini menggunakan satu buah dioda tipe 1N4002 yang diseri dengan kapasitor tipe 470nF yang menghasilkan *output* berupa tegangan DC yang besarnya sama dengan amplitudo puncak (V_p) dari input tagangan AC, serta adanya 2 resistor 470 Ohm yang dirangkai seri yang di gunakan untuk kalibrasi sensor tegangan dengan prinsip pembagi tegangan agar

tegangan maksimum dari input AC sebesar 6 V dapat menghasilkan output tegangan DC sebesar 5 V. Pemilihan jenis kapasitor tipe ini dimaksudkan untuk menstabilkan tegangan keluaran sensor dengan mengurangi *ripple* tegangan yang dihasilkan oleh keluaran dari sensor tegangan. Selain itu terdapat juga lampu LED yang digunakan sebagai indikator ada tidaknya suatu tegangan. Rangkaian sensor peak detektor dapat dilihat pada Gambar 3.7.



Gambar 3.7 Rangkaian Peak Detektor

3.2.3 Rangkaian *Power Supply*

Power Supply digunakan untuk mensuplai tegangan listrik ke beberapa komponen seperti Arduino Mega, sensor peak detektor, RTC, LCD, dan Modem GSM SIM 900. *Power Supply* yang digunakan dalam tugas akhir ini adalah *power supply* dengan tegangan 12 V. Sedangkan untuk rangkaian arduino dengan power supply ini dapat dilihat pada Gambar 3.8.



Gambar 3.8 Skema Modul *Power Supply* dengan Arduino Mega

Power supply ini digunakan untuk mensuplai tegangan DC ke modul GSM SIM900.

3.2.4 Rangkaian RTC

Rangkaian modul RTC ini dihubungkan pada Arduino Mega, dimana modul ini menggunakan 4 buah pin yang di sambungkan ke arduino yakni pin VCC, pin Ground, pin SDA, serta pin SCL. Untuk pin SDA dan pin SCL akan disambungkan ke pin 20 dan pin 21 pada arduino Mega, sedangkan untuk pin VCC dan pin *Ground* akan disambungkan ke pin +5V dan pin GND pada Arduino. Tipe RTC yang digunakan dalam tugas akhir ini adalah RTC DS1307 dimana wiring rangkaiannya dapat dilihat pada Gambar 3.9.

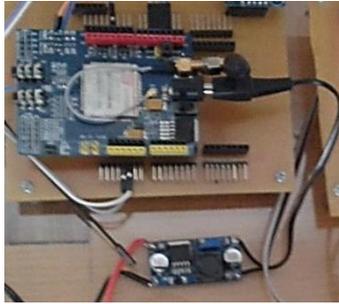


Gambar 3.9 Skema Rangkaian modul RTC dengan Arduino Mega

Rangkaian RTC ini selanjutnya akan digunakan untuk menandakan waktu yang akan disesuaikan dengan proses penjadwalan untuk pemeliharaan gardu.

3.2.5 Rangkaian Modem GSM SIM 900

Pada rangkaian ini, SIM 900 disambung dengan arduino mega dan modul buck konverter untuk menurunkan tegangan *input* yang berasal dari *power supply*. Modem SIM900 yang digunakan pada tugas akhir ini yaitu “SIM900 GSM / GPRS Shield with Antenna For Arduino Mega” dimana modul ini berbentuk shield yang dapat terintegrasi langsung dengan Arduino Mega 2560. SIM900 mengambil tegangan sumber sebesar 5 Volt DC dari Arduino Mega 2560. Sementara itu untuk perancangannya, pin 19 (RX1) dari Arduino Mega disambungkan dengan TX7 dari modul SIM900. Dan pin 18(TX1) dari Arduino Mega tersambung dengan pin RX7 yang terdapat pada modul SIM900. Gambar dari SIM900 yang dihubungkan dengan Arduino Mega 2560 dan modul *buck* konverter terdapat pada Gambar 3.10.

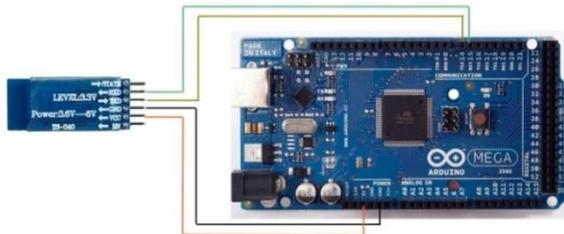


Gambar 3.10 Rangkaian SIM 900 dengan Arduino Mega dan RTC

Rangkaian SIM 900 dalam tugas akhir ini selanjutnya digunakan untuk mengirim pemberitahuan kepada operator pusat mengenai kondisi gardu konvensional.

3.2.6 Rangkaian Modul Bluetooth

Modul Bluetooth ini digunakan sebagai media komunikasi *software* dan *hardware* antara Arduino Mega dengan android. Untuk perancangan modul bluetooth ini pin Vcc dihubungkan dengan pin +5 pada arduino, pin GND dihubungkan dengan GND, pin RX dihubungkan dengan RX3 15, dan pin TX dihubungkan dengan TX3 14. Skema perancangan modul bluetooth dapat dilihat pada Gambar 3.11.



Gambar 3.11 Skema Perancangan Modul Bluetooth dengan Arduino

3.3 Pembuatan Software

Untuk pembuatan *software* pada tugas akhir ini menggunakan *software* arduino IDE untuk proses pengolahan datanya, sedangkan untuk pembuatan aplikasi di android perancangannya menggunakan App

Inventor. Untuk pembuatan database yang digunakan untuk menyimpan riwayat data selama pengoperasian maka digunakan PHP Runner sedangkan untuk pembuatan tampilan program database yang ditampilkan dalam halaman website digunakan *software anWriter free*.

3.3.1 Pembuatan Sistem Arduino IDE

Proses pembuatan sistem untuk arduino IDE meliputi proses pengambilan data dari sensor peak detector yang ada pada gardu konvensional, pengambilan data RTC, serta monitoring status dari CB, LBS *incoming*, maupun LBS *outgoing* yang selanjutnya akan dtampilkan dalam aplikasi android dan juga pada LCD. *Flowchart* tersebut dapat dilihat pada Gambar 3.12. Selanjutnya berdasarkan *flowchart* pada Gambar 3.13 dan Gambar 3.14 didapat penjelasan:

1. Arduino Mega melakukan inialisasi variabel data
2. Pembacaan operasi boolean, yakni operasi=true ketika dilakukan proses pengoperasian gardu konvensional dan operasi=false ketika dilakukan proses pemadaman gardu. Sedangkan kondisi == HIGH menandakan status gardu berada pada kondisi pertama yakni kondisi bertegangan. Selain itu, untuk gangguan, saat gangguan=true mengindikasikan gardu konvensional telah terjadi gangguan sedangkan saat gangguan=false gardu konvensional sudah selesai diperbaiki.
3. Jika pembacaan operasi adalah benar maka arduino akan melakukan pembacaan nilai untuk peak detektor tegangan, RTC, dan bluetooth HC-05. Sehingga jika tegangan CB yang dideteksi lebih dari 1,7 V maka data akan ditampilkan ke ke layar LCD dan juga pada android melalui bluetooth HC-05 yang akan menampilkan status CB terbuka dan tegangan *relay* yang ada pada CB bernilai 0, sedangkan jika tegangan yang terbaca kurang dari 1,7 maka pada layar akan menampilkan status CB dalam keadaan tertutup dan tegangan *relay* yang ada pada CB bernilai 20000V. Jika tegangan DS *incoming* yang di deteksi lebih dari 1,7 V maka data yang tertampil pada layar menunjukkan status DS *incoming* dalam keadaan terbuka dan tegangan yang dihasilkan bernilai 0, sedangkan jika tegangan DS *incoming* yang dideteksi kurang dari 1,7 android akan menampilkan status kondisi DS *incoming* dalam keadaan tertutup dan tegangan yang dideteksi sebesar 20000V. Untuk DS *outgoing*, ketika tegangan yang dideteksi lebih dari 1,7

maka data yang tertampil pada layar menunjukkan status DS *outgoing* dalam keadaan terbuka dan tegangan yang dihasilkan bernilai 0 serta menunjukkan pembacaan sensor RTC pada layar, kondisi ini akan berulang sampai pembacaan CB, Dsin, dan DS *outgoing* dalam keadaan tertutup, sedangkan jika tegangan DS *outgoing* yang dideteksi kurang dari 1,7 android akan menampilkan status kondisi DS *outgoing* dalam keadaan tertutup dan tegangan yang dideteksi sebesar 20000V dan menampilkan sensor RTC pada android. selanjutnya setelah proses pengoperasian selesai maka SIM 900 akan mengirimkan notifikasi kepada operator PLN yang ada di kantor bahwa pengoperasian gardu konvensional telah selesai dilaksanakan.

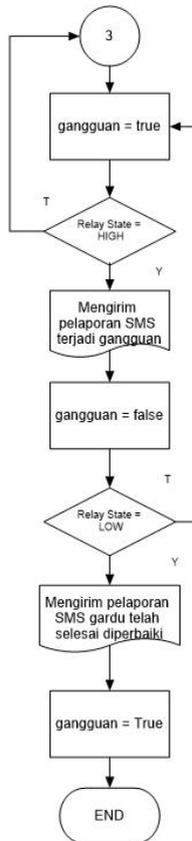
4. Jika pembacaan operasi adalah salah maka arduino akan melakukan pembacaan nilai untuk peak detektor tegangan, RTC, dan bluetooth HC-05. Sehingga jika tegangan CB yang dideteksi lebih dari 1,7 V maka data akan ditampilkan ke layar LCD dan juga pada layar android melalui bluetooth HC-05 yang akan menampilkan status CB terbuka dan tegangan *relay* yang ada pada CB bernilai 0, sedangkan jika tegangan yang terbaca kurang dari 1,7 maka pada layar akan menampilkan status CB dalam keadaan tertutup dan tegangan *relay* yang ada pada CB bernilai 20000V. Jika tegangan DS *incoming* yang dideteksi lebih dari 1,7 V maka data yang tertampil pada layar menunjukkan status DS *incoming* dalam keadaan terbuka dan tegangan yang dihasilkan bernilai 0, sedangkan jika tegangan DS *incoming* yang dideteksi kurang dari 1,7 maka layar akan menampilkan status kondisi DS *incoming* dalam keadaan tertutup dan tegangan yang dideteksi sebesar 20000V. Untuk DS *outgoing*, ketika tegangan yang dideteksi lebih dari 1,7 maka data yang tertampil pada layar menunjukkan status DS *outgoing* dalam keadaan terbuka dan tegangan yang dihasilkan bernilai 0 serta menunjukkan pembacaan sensor RTC pada layar selanjutnya setelah proses pengoperasian selesai maka SIM 900 akan mengirimkan notifikasi kepada operator PLN yang ada di kantor bahwa gardu siap dioperasikan. Sedangkan jika tegangan DS *outgoing* yang dideteksi kurang dari 1,7 maka layar akan menampilkan status kondisi DS *outgoing* dalam keadaan tertutup dan tegangan yang dideteksi sebesar 20000V serta menampilkan sensor RTC pada layar. Kondisi ini terus

berlanjut sampai pembacaan CB, DS *incoming*, dan DS *outgoing* dalam keadaan tertutup.

5. Jika pembacaan gangguan= true maka jika tombol relaystate yang ada pada beban ditekan maka SIM 900 akan mengirimkan SMS berupa pemberitahuan telah terjadi gangguan di gardu, jika gangguan= false maka tombol *relaystate* yang ada pada beban dikembalikan ke posisi semula SIM 900 akan mengirim pelaporan SMS gardu telah selesai diperbaiki.



Gambar 3.12 *Flowchart 1* Program Arduino



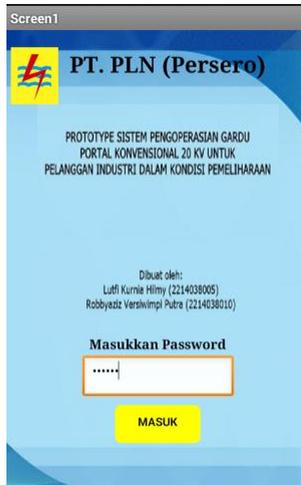
Gambar 3.14 Flowchart 3 Program Arduino

3.4 Pemrograman App Inventor

Pada pemrograman App Inventor dibagi menjadi 3 *screen*, yakni antara lain:

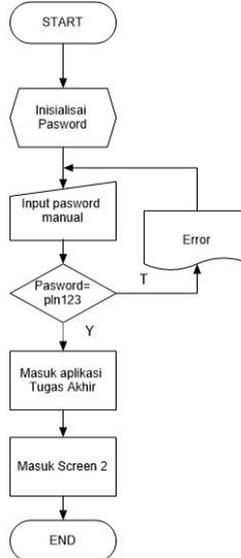
1. Tampilan *Screen 1* pada aplikasi

Tampilan *screen 1* ini merupakan tampilan halaman utama ketika aplikasi pertama kali dibuka. Pada *screen 1* ini terdapat tombol masuk dan kolom berisi *password* yang harus diinputkan agar dapat masuk ke halaman selanjutnya. Adapun tampilan pada *screen 1* ini dapat dilihat pada Gambar 3.15.



Gambar 3.15 Tampilan Screen 1 pada Android

Sedangkan *flowchart* pemrograman dapat dilihat pada Gambar 3.16.



Gambar 3.16 *Flowchart* Pemrograman Screen 1

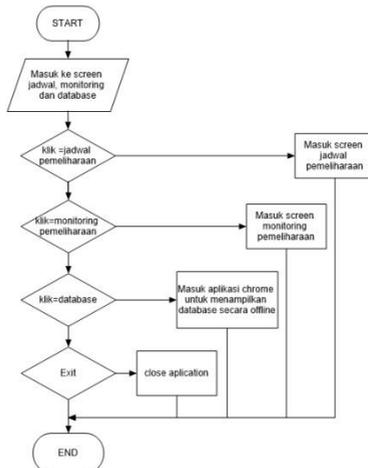
2. Tampilan *Screen 2* pada aplikasi

Screen 2 ini berisi tombol yang menghubungkan ke *screen* yang dituju. Adapun tampilan pada *screen 2* ini dapat dilihat pada Gambar 3.17.



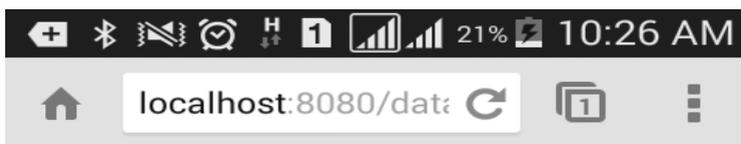
Gambar 3.17 Tampilan *Screen 2* pada Aplikasi

Dalam *screen 2* ini juga disertakan adanya tombol *exit* yang digunakan untuk menutup aplikasi. Untuk *flowchart* pemrogramannya dapat dilihat pada Gambar 3.18.



Gambar 3.18 *Flowchart* Pemrograman *Screen 2*

Sedangkan tampilan database dapat dilihat pada Gambar 3.19.



Database Industri

Nama perusahaan	Status CB	Sensor Tegangan 1	Status LBS Incoming	Sensor Tegangan 2	Status LBS Outgoing	Kondisi Gardu	Tanggal dan Waktu
PT. Argo Jaya	CB Terbuka	0 V	LBS Incoming Te	0 V	LBS Outgoing Te	Gardu Aman Dipe	30.10.2014 / 09
PT. Argo Jaya	CB Terbuka	0 V	LBS Incoming Te	0 V	LBS Outgoing Te	Gardu Dipelihar	30.10.2014 / 09
PT. Argo Jaya	CB Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	30.10.2014 / 09:30:39
PT. Argo Jaya	CB Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipelihara	30.10.2014 / 09:34:48
PT. Argo Jaya	CB Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	30.10.2014 / 09:39:24
PT. Argo Jaya	CB Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipelihara	15.09.2015 / 08:31:50
PT. Argo Jaya	CB Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	15.09.2015 / 08:34:06
PT. Argo Jaya	CB Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipelihara	21.12.2016 / 11:22:58
PT. Argo Jaya	CB Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	21.12.2016 / 11:25:19
PT. Argo Jaya	CB Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipelihara	08.06.2017 / 10:31:59
PT. Argo Jaya	CB Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	08.06.2017 / 10:33:25

Gambar 3.19 Tampilan Database pada Aplikasi

Dalam gambar 3.17 jika jadwal pemeliharaan di klik maka akan langsung masuk ke *screen* jadwal pemeliharaan, jika monitoring pemeloharaan diklik maka akan masuk ke halaman monitoring, sedangkan jika database pemeliharaan yang diklik maka akan masuk ke halaman database seperti yang ditunjukkan pada Gambar 3.19.

3. Tampilan *Screen* Jadwal Pemeliharaan

Tampilan ini berisi jadwal pemeliharaan pada gardu konvensional dimana dalam *screen* ini terdapat tombol kembali untuk kembali ke *screen* 2. Untuk tampilan jadwal pemeliharaan dapat dilihat pada Gambar 3.20.

Jadwal Pemeliharaan				
No.	Nama Industri	Tipe Gardu	Tanggal Pemeliharaan	Waktu
1.	PT. Argo Jaya	Konvensional	30 Oktober 2014	09.15 – 12.15
2.	PT. Argo Jaya	Konvensional	15 September 2015	08.15 – 11.15
3.	PT. Argo Jaya	Konvensional	21 Desember 2016	11.15 – 14.15
4.	PT. Argo Jaya	Konvensional	08 Juni 2017	10.20 – 13.20

Kembali

Gambar 3.20 Tampilan *Screen* Jadwal Pemeliharaan

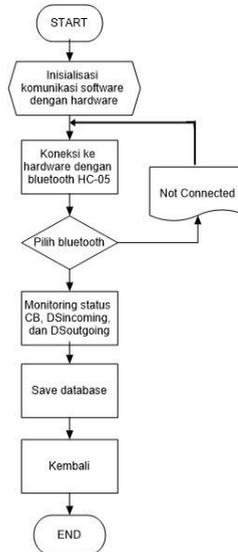
4. Tampilan *Screen* Monitoring pada Aplikasi

Tampilan monitoring berisi monitoring tahapan pengoperasian gardu konvensional. Dalam *screen* monitoring ini terdapat tombol pilih bluetooth yang berfungsi sebagai *interface* dari arduino agar dapat terkoneksi ke android, serta terdapat tombol *not connected* yang akan berubah menjadi *connected* ketika bluetooth sudah terhubung dengan android. Selanjutnya terdapat tombol nama industri yang berisi *list* dari industri yang akan dilakukan pemeliharaan gardu konvensional, dibawahnya terdapat kolom monitoring tahapan yang dilakukan ketika proses pengoperasian gardu yang akan tampil ketika dikoneksikan dengan sensor peak detektor. Di bawahnya terdapat tombol *save* yang akan menyimpan data pengoperasian gardu konvensional ke dalam database. Selain itu juga terdapat tombol kembali Adapun tampilan dari *screen* monitoring ini dapat dilihat pada Gambar 3.21.



Gambar 3.21 Tampilan Monitoring pada Aplikasi

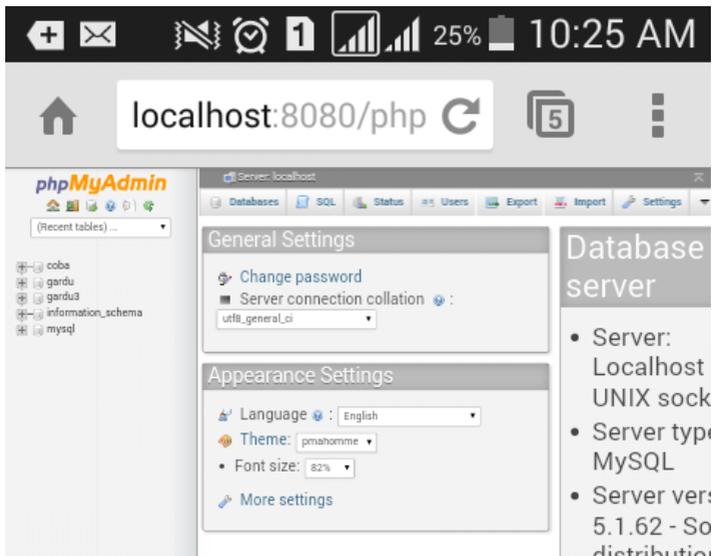
Sedangkan *flowchart* pemrogramannya dapat dilihat pada Gambar 3.22.



Gambar 3.22 Pemrograman *Flowchart Screen Monitoring*

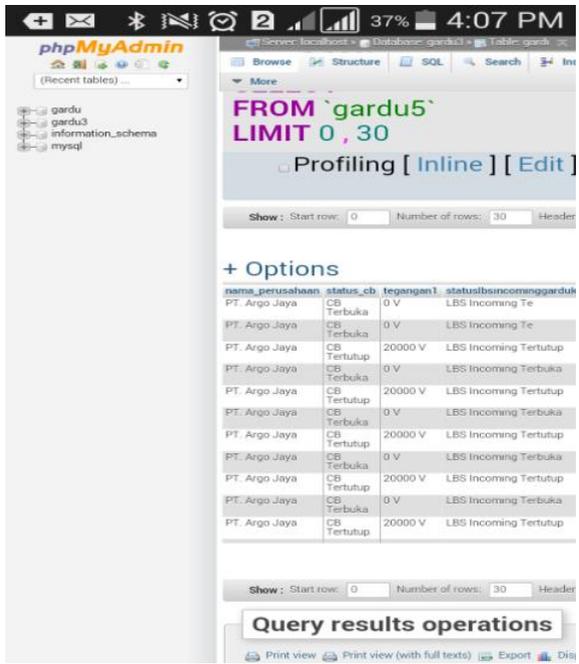
3.5 Pembuatan Database

Untuk pembuatan database yang digunakan untuk penyimpanan data pengoperasian, digunakan aplikasi *PHPrunner*, *PHPrunner* ini merupakan *website* berbasis database MySQL yang pengoperasiannya sama seperti *PHPMyAdmin* yang terdapat pada PC. Adapun tampilan utama dari database pada halaman *PHPMyAdmin* dapat dilihat pada Gambar 3.23.



Gambar 3.23 Tampilan *PHP MyAdmin* pada Android

Pada Gambar 3.23 terdapat database *gardu3* yang di dalamnya terdiri dari tabel yang digunakan untuk menyimpan database dari industri Argo Jaya. Sementara itu untuk melihat isi tabel dari database yang telah dibuat dapat dilakukan dengan mengklik menu *databases* yang kemudian terdapat tabel *Gardu 5* yang digunakan untuk menyimpan riwayat pengoperasian gardu konvensional. Tampilan dari database ini di rancang dalam versi android yang mana untuk mengakses database ini dapat dilakukan dalam bentuk *offline* dengan menggunakan aplikasi *browser* internet. Adapun tampilan dari tabel *gardu 5* yang terdapat pada database *gardu 3* ini dapat dilihat pada gambar 3.24.



Gambar 3.24 Tampilan Tabel Pada PHP *MyAdmin*

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

BAB IV PENGUJIAN DAN ANALISA DATA

Pada bab ini akan dibahas mengenai hasil pengujian dari peralatan dan analisa dari Tugas Akhir yang berjudul “Sistem Pelaporan Gardu Portal Konvensional Satu Fasa dalam Kondisi Pemeliharaan”. Pengujian dilakukan untuk mengetahui kinerja dari peralatan dan pembuatan sistem yang telah dirancang dan direncanakan agar sistem dalam pelaporan gardu konvensional ini dapat berjalan sesuai dengan yang diharapkan, sedangkan untuk analisa dilakukan untuk mengamati dan mengetahui penyebab terjadinya kesalahan atau kekurangan yang terdapat dalam sistem. Adapun bagian-bagian yang akan diuji pada alat ini adalah:

1. Pengujian Arduino Mega 2560
2. Pengujian *Power Supply*
3. Pengujian LCD
4. Pengujian RTC
5. Pengujian Sensor Peak Detektor
6. Pengujian Komunikasi dengan SIM 900
7. Pengujian *Buck Konverter*
8. Pengujian dengan Modul Bluetooth
9. Pengujian Komunikasi Arduino dengan MIT Android
10. Pengujian Keseluruhan

4.1 Pengujian *Power Supply*

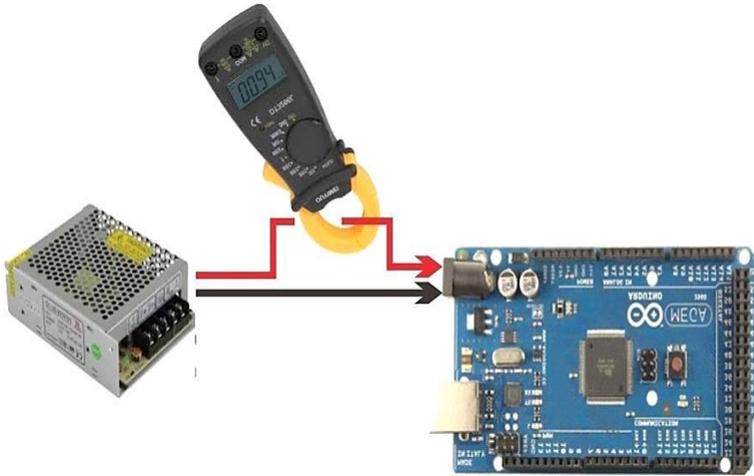
Dalam pengujian *power supply* ini digunakan multimeter “SANWA” yang digunakan sebagai alat ukur. *Probe* merah dihubungkan pada terminal tegangan *output* sedangkan *probe* hitam dihubungkan dengan terminal *ground*. Hasil pengujian tegangan *output* keluaran dapat dilihat pada Tabel 4.1.

Tabel 4.1 Pengujian *Power Supply*

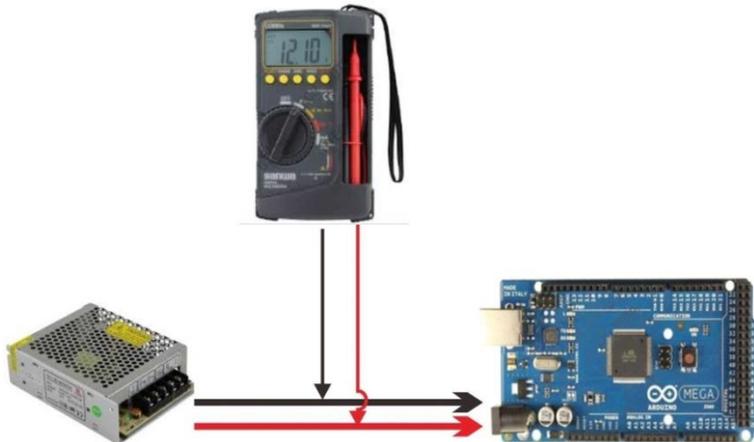
Vout	Iout
12,10 Volt	0,94 Ampere

Dari data diatas, *power supply* memberikan tegangan *output* sebesar 12,09 Volt. Nilai ini masih termasuk dari *range* tegangan *input* dari arduino mega yaitu antara 7 V-12 V. Sehingga Arduino Mega

masih dapat bekerja ketika diberi tegangan ini. Sementara itu, untuk pengujian arus *output* dan tegangan *output* pada *power supply* dapat dilihat pada Gambar 4.1 dan Gambar 4.2.



Gambar 4.1 Proses Pengujian Arus *Power Supply* dengan Arduino



Gambar 4.2 Proses Pengujian Tegangan *Power Supply* dengan Arduino

4.2 Pengujian Arduino Mega 2560

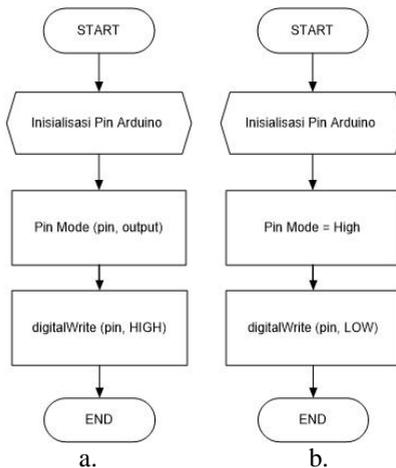
Pengujian ini dilakukan dengan menguji pin pada *board* arduino mega 2560 yang digunakan dalam sistem. Adapun pengujian ini dilakukan sengan mengatur keluarannya pada logika 1 dan logika 0. Program logika 1 merupakan program untuk memberikan tegangan terhadap arduino mega sebesar 5 Volt. Sedangkan program logika 0 merupakan program untuk memberikan tegangan pada arduino mega sebesar 0 Volt. Setelah diberikan program logika 0 dan logika 1, keluaran pada masing-masing pin pada arduino akan diukur menggunakan Multimeter, untuk tegangan yang digunakan sebagai *supply* pada arduino mega 2560 digunakan *power supply* sebesar 12 V. Untuk skema pengujian pin pada arduino dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian Keluaran Tegangan Pin pada Arduino Mega 2560.

Pin	Tegangan (logika 1)	Tegangan (logika 0)
A0	4.96 V	0.7 mV
A1	4.96 V	0.7 mV
A2	4.96 V	0.7 mV
A3	4.96 V	0.5 mV
A4	4.96 V	0.7 mV
A5	4.96 V	0.21 mV
A6	4.96 V	0.52 mV
A7	4.96 V	0.21 mV
D0	4.96 V	0.46 mV
D1	4.96 V	0.3 mV
D2	4.96 V	0.3 mV
D3	4.96 V	0.23 mV
D4	4.96 V	0.45 mV
D5	4.96 V	0.23 mV
D6	4.96 V	0.23 mV
D7	4.96 V	0.21 mV

Pin	Tegangan (logika 1)	Tegangan (logika 0)
D8	4.96 V	0.4 mV
D9	4.96 V	0.3 mV
D10	4.96 V	0.52 mV
D11	4.96 V	0.4 mv
D12	4.96 V	0.3 mV
D13	4.96 V	0.4 mV
D14	4.96 V	0.2 mV
D15	4.96 V	0.33 mV
D16	4.96 V	0.3 mV
D17	4.96 V	0.3 mV
D18	4.96 V	0.3 mV
D19	4.96 V	0.2 mV
D20	4.96 V	0.2 mV
D21	4.96 V	0.3 mV

Sedangkan *flowchart* pemrograman arduino yang digunakan untuk pengujian dapat dilihat pada Gambar 4.3.



Gambar 4.3 *Flowchart* Pemrograman Pengujian Tegangan pada Pin Arduino Mega 2560 a adalah logika 1 dan b adalah logika 0.

Untuk *flowchart* 4.3 a merupakan pemrograman pemberian tegangan pada pin arduino berlogika 1 sedangkan untuk *flowchart* 4.3 b merupakan pemrograman pemberian tegangan pada pin arduino berlogika 0.

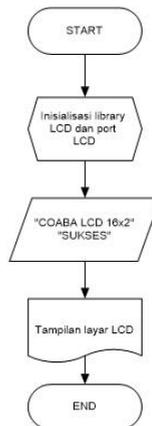
4.3 Pengujian LCD

Pengujian LCD dilakukan dengan cara memberikan program ke arduino mega agar diproses sehingga dapat ditampilkan pada layar LCD 16x2. Dalam pengujian LCD ini, diberikan program “ COBA LCD 16x2” pada baris pertama kemudian diikuti dengan text “ “SUKSES” pada baris berikutnya. Untuk pengujian LCD ini dapat dilihat pada Gambar 4.4.



Gambar 4.4 Pengujian pada LCD

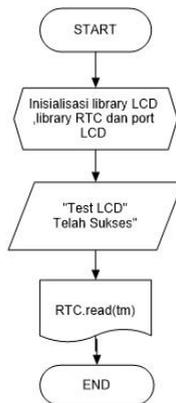
Sedangkan untuk *flowchart* pemrograman arduino dapat dilihat pada Gambar 4.5.



Gambar 4.5 *Flowchart* Pemrograman Arduino untuk LCD

4.4 Pengujian RTC

Pengujian RTC dilakukan dengan memberikan program pada arduino mega agar dapat memproses sensor RTC dan selanjutnya menampilkan hasil tersebut secara langsung waktu yang terbaca pada LCD 16x2 dan membandingkannya dengan data waktu yang tertampil di layar komputer. Adapun *flowchat* pemrograman RTC dengan arduino mega dapat dilihat pada Gambar 4.6.



Gambar 4.6 Flowchart Pemrograman Arduino Mega untuk RTC

Sedangkan untuk pengujian RTC sendiri dapat dilihat pada Gambar 4.7



Gambar 4.7 Pengujian RTC

Berdasarkan pengujian tersebut didapatkan data seperti pada Tabel 4.3.

Tabel 4.3 Pengujian Data Waktu pada RTC

No	Waktu pada RTC	Waktu pada Layar Komputer	Selisih Waktu
1	15:56:44	15:56:51	7 detik
2	16:05:06	16:05:13	7 detik
3	16:12:58	16:12:05	7 detik

Dalam Tabel 4.3, dilakukan pengujian pada RTC yang dipadukan dengan jam yang terdapat pada komputer. Pada tabel tersebut diketahui bahwa selisih antara jam RTC dan jam sesungguhnya (pada komputer) 7 detik. Dan hal ini selalu tetap (tidak berubah).

4.5 Pengujian dengan Sensor Peak Detektor

Untuk melakukan pengujian dengan sensor peak detektor digunakan multimeter SANWA yang mana hasil pengujian dapat dilihat pada Tabel 4.4 dan Tabel 4.5.

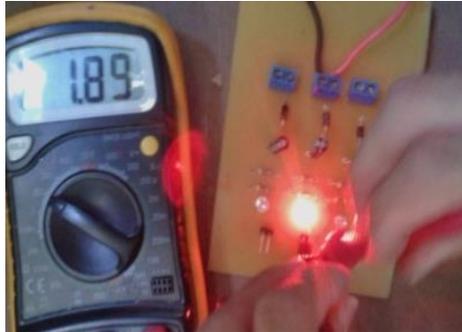
Tabel 4.4 Pengujian Tegangan Output Sensor Peak Detektor

	Sensor 1	Sensor 2	Sensor 3
Vin	227 V	228 V	227 V
Vout AC	6,4 V	5,6 V	5,6 V
Vout LED (DC)	1,88 V	1,92V	1,8 V

Tabel 4.5 Pengujian Tegangan Output Sensor Peak Detektor untuk beban R dan RL

	Sensor 1	Sensor 2	Sensor 3
R	5,39 V	5,18 V	6,32 V
RL	7,18 V	7,14 V	8,15 V

Sedangkan proses pengujian alat dapat dilihat pada Gambar 4.8

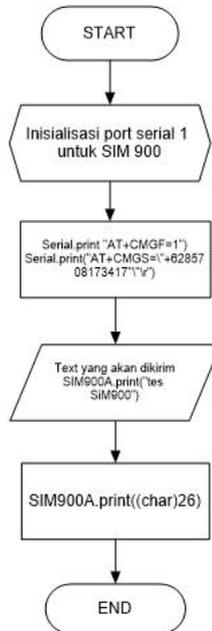


Gambar 4.8 Pengujian Sensor *Peak Detektor*

Dalam Tabel 4.4 tegangan V_{in} 227 Volt pada sensor 1 adalah tegangan *input* yang berasal dari tegangan AC (jala-jala). Sedangkan tegangan 6,4 V merupakan tegangan AC keluaran dari trafo *step down* yang telah diturunkan, sementara itu untuk tegangan 1,88 V merupakan tegangan keluaran dari sensor peak detektor yang telah dirubah menjadi DC. Begitu juga untuk tegangan pada sensor 2 dan sensor 3. Hasil tegangan *output* 1,88 V, 1,92 V, dan 1,8 V ini merupakan tegangan yang nantinya akan masuk ke Arduino yang kemudian akan diproses untuk mendeteksi status dari saklar relai yang terdapat pada prototipe gardu. Untuk tegangan pada beban R dan RL yang terdapat pada Tabel 4.5 pada sensor 1, sensor 2 dan pada sensor 3 merupakan tegangan *output* DC yang telah dirubah dari AC menjadi DC oleh sensor peak detektor dimana tegangan ini digunakan sebagai pembanding tegangan output yang ada pada sensor peak detektor sebelum ditambahkan dengan beban LED.

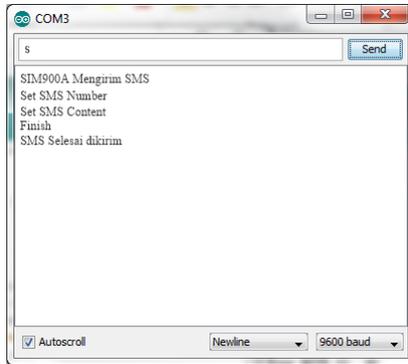
4.6 Pengujian Komunikasi dengan Modem GSM SIM 900

Untuk melakukan pengujian dengan SIM 900 maka diperlukan program *AT+command*. Untuk mengirimkan sms berupa *text* ke android maka digunakan perintah *AT+CMGF=1*, *AT CMGF* itu sendiri merupakan suatu perintah untuk memprogram sms dalam bentuk *text*. Sedangkan untuk mengirimnya ke nomor tujuan maka digunakan perintah *AT+CMGS*. Untuk *flowchart* pemrogramannya dapat dilihat pada Gambar 4.9.



Gambar 4.9 Flowchart Pemrograman Pengiriman SMS

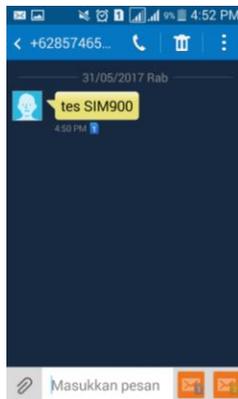
Untuk menjalankan modul GSM SIM 900 dibutuhkan tegangan sebesar 5.15 Volt sesuai dengan datasheet. Sedangkan untuk pengujian modem GSM SIM 900 menggunakan aplikasi arduino IDE pada komputer yang dihubungkan dengan arduino mega dan modem GSM menggunakan kabel serial. Untuk program pengiriman SMS menggunakan SIM 900 dapat dilihat pada halaman lampiran. Sedangkan, untuk hasil pengujian dengan menggunakan AT+*command* pada komputer. Data-data yang ditampilkan pada serial monitor pada komputer untuk pengiriman SMS menggunakan SIM 900 dapat dilihat pada Gambar 4.10.



Gambar 4.10 Tampilan Serial Monitor Program Pengiriman SMS SIM 900

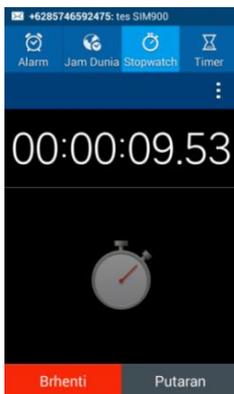
Pada serial monitor menunjukkan komunikasi antara mikrokontroller dengan SIM 900 melalui perantara komputer, setelah huruf “s” diketik dan dikirim melalui komputer maka SIM 900 akan memproses perintah tersebut dan akan memberikan balasan “SMS selesai” dikirim ketika program berhasil dieksekusi. Sehingga dalam Gambar 4.10 menunjukkan bahwa komunikasi antara mikrokontroller dengan SIM 900 telah berhasil dilakukan.

Sementara itu, untuk keberhasilan pengiriman SMS dari Modem GSM SIM 900 ini dapat dilihat pada Gambar 4.11.



Gambar 4.11 Hasil Pengujian SIM 900

Pada tugas akhir ini SMS yang dikirim tersebut akan digunakan sebagai pemberitahuan ke operator pusat. SMS tersebut akan mengirim kepada petugas saat gardu konvensional pada suatu industri terjadi tiga kondisi yaitu kondisi akan dilakukan pemeliharaan., kondisi selesai dilakukan pemeliharaan, dan kondisi ketika gardu konvensional sedang mengalami gangguan. Sedangkan hasil pengujian waktu pengiriman SMS tersebut dapat dilihat pada Gambar 4.12.



Gambar 4.12 Hasil Pengujian Waktu Pengiriman SMS

Berdasarkan Gambar 4.12 lamanya waktu pengiriman SMS dari SIM900 ke android tersebut adalah sekitar 00:09:53 detik, hal ini dipengaruhi oleh lokasi dan sinyal dari *provider* yang digunakan.

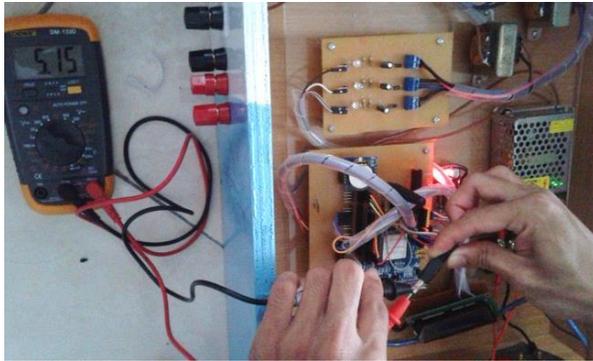
4.7 Pengujian Modul *Buck* Konverter

Untuk membatasi tegangan yang masuk ke SIM 900 agar tidak terlalu besar, maka digunakan agar tidak merusak SIM900. Pengujian ini dengan mengukur tegangan output pada modul buck konverter yang ditunjukkan pada Tabel 4.6.

Tabel 4.6 Tegangan *Output* Buck konverter

Vin	12 Volt
Vout	5.15 Volt

Sedangkan proses pengujian *buck* konverter dapat dilihat pada Gambar 4.13



Gambar 4.13 Pengujian Modul *Buck* Konverter

Berdasarkan Gambar 4.13 pengukuran tegangan *output* dari modul *buck* konverter menggunakan multimeter SANWA dengan hasil keluaran yang didapat adalah sebesar 5.15 Volt.

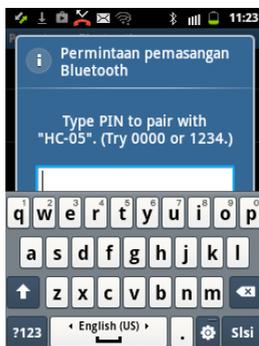
4.8 Pengujian dengan Modul Bluetooth

Pengujian ini dilakukan dengan tujuan untuk mengetahui apakah modul bluetooth HC-05 ini dapat terkoneksi dengan dengan Android atau tidak. Pengujian ini dilakukan dengan memberikan *power* 5V pada modul bluetooth dan selanjutnya keberadaannya akan dideteksi oleh Android dengan men-*scan* bluetooth. Untuk pengujian modul bluetooth dengan android dapat dilihat pada Gambar 4.14, Gambar 4.15, dan Gambar 4.16.



Gambar 4.14 Android Mendeteksi Keberadaan Modul Bluetooth

Kemudian muncul kotak dialog Permintaan pemasangan Bluetooth yang meminta PIN (default :1234) untuk *pairing* dengan Android. Proses *pairing* dengan android dapat dilihat pada Gambar 4.15. sedangkan gambar yang menunjukkan bluetooth sudah terpasang dengan android dapat dilihat pada Gambar 4.16.



Gambar 4.15 Proses *Pairing* Modul Bluetooth dengan Android



Gambar 4.16 Modul Bluetooth Sudah Terpasang dengan Android

Selanjutnya untuk menguji jarak koneksi dari modul bluetooth dapat dilihat pada Tabel 4.7.

Tabel 4.7 Pengujian Jarak Koneksi Modul Bluetooth

Jarak	Koneksi
2 meter	Terhubung
4 meter	Terhubung
6 meter	Terhubung
7 meter	Terhubung
8 meter	Tidak Terhubung

Dari Tabel 4.7 diketahui bahwa jarak jangkauan modul bluetooth untuk terhubung dengan perangkat adalah maksimal 7 meter, ketika jarak pengkoneksian bluetooth diatas 7 meter maka bluetooth sudah tidak tersambung.

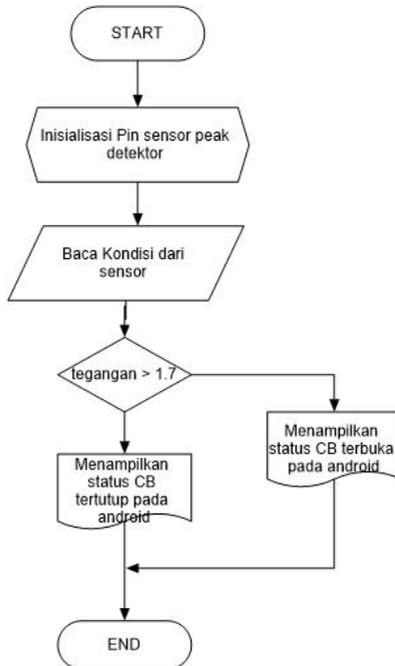
4.9 Pengujian Komunikasi Arduino dengan MIT Android

App inventor ini digunakan untuk memonitoring tiap status pekerjaan yang dilakukan oleh operator terhadap pengoperasian gardu konvensional. Setiap kondisi yang diberikan akan tampil di layar android dan dapat digunakan oleh petugas untuk memonitoring status gardu konvensional. Adapun untuk menghubungkannya dengan arduino mega maka dibutuhkan bantuan modul bluetooth HC-05. Berikut adalah hasil pengujian koneksi aplikasi App Inventor dengan arduino mega dapat dilihat pada Gambar 4.17.



Gambar 4.17 Pengujian Koneksi Antara Arduino Mega dengan Aplikasi Android.

Dalam pengujian ini untuk menampilkan teks yang pada layar android maka dibutuhkan adanya penambahan sensor peak detektor yang digunakan sebagai kondisi penampilan tulisan pada layar. Sensor ini akan mendeksi ada tidaknya tegangan yang ada pada gardu dimana ketika tidak terdapat tegangan yang dideteksi maka arduino akan memproses program dari keluaran sensor yang diberikan dan oleh android program tersebut akan ditampilkan ke layar dengan menggunakan modul bluetooth HC-05. Adapun *flowchart* pemrograman pada arduino ditunjukkan pada Gambar 4.18.



Gambar 4.18 Flowchart Pemrograman *Interface* dengan Android

4.10 Pengujian Keseluruhan Alat

Pengujian ini dilakukan dengan mengkoordinasikan komponen-komponen seperti sensor peak detektor, sensor RTC, modem GSM SIM 900, *prototype* gardu konvensional beserta sistem *interlock* yang digunakan pada LBS yang ada di dalamnya, dan tampilan pada android yang digunakan sebagai monitoring.

Pengujian ini dilakukan dengan memonitoring status CB, LBS *outgoing*, dan LBS *incoming* dengan memperhatikan dua kondisi dari masing-masing saklarnya. Dua kondisi tersebut adalah kondisi pembukaan dan penutupan saklar dimana ketika melakukan pengujian ini maka pengujian dilakukan berdasarkan kondisi real pemeliharaan yang dilakukan operator. Oleh karena itu pengujian yang akan dilakukan akan didasarkan pada jadwal pemeliharaan yang telah dibuat. Jadwal pemeliharaan ini dapat dilihat pada Tabel 4.8.

Tabel 4.8 Jadwal Pemeliharaan Gardu Konvensional

No.	Nama Industri	Tipe Gardu	Tanggal Pemeliharaan	Waktu
1.	PT. Argo Jaya	Konvensional	30 Oktober 2014	09.15-12.15
2.	PT. Argo Jaya	Konvensional	15 September 2015	18.15-11.15
3.	PT. Argo Jaya	Konvensional	21 Desember 2016	11.15-14.15
4.	PT. Argo Jaya	Konvensional	08 Juni 2017	10.20-13.20

Dalam Tabel 4.8, pembuatan jadwal pemeliharaan didasarkan pada jenis pemeliharaan tahunan yang dilakukan oleh PLN, sehingga dalam hal ini saat petugas akan melakukan pemeliharaan maka gardu harus dipadamkan terlebih dahulu. Sementara itu pemeliharaan gardu dilakukan dengan memperhatikan 4 kondisi yaitu:

1. Ada jadwal, gardu berkerja secara normal.
2. Ada jadwal, gardu padam
3. Tidak ada jadwal, gardu bekerja secara normal
4. Tidak ada jadwal, gardu padam (terjadi gangguan)

Adapun pengujian dari masing-masing kondisi adalah sebagai berikut:

4.10.1 Pengujian dengan Kondisi Pertama

Dalam kondisi ini, pengujian dilakukan dengan mengacu pada jadwal yang terdapat pada Tabel 4.8. Pada kondisi pertama ini gardu bekerja secara normal, dalam artian tidak terdapat gangguan pada seluruh peralatan yang ada pada gardu selama dioperasikan. Sedangkan, untuk pembuatan jadwalnya pemeliharaan tersebut disesuaikan dengan adanya pemeliharaan yang dilakukan oleh PLN yang mana pemeliharaan ini dilakukan setiap tahunnya. Oleh karena itu dalam pengujian ini untuk sensor RTC, tanggal dan waktu akan diatur menyesuaikan dengan jadwal yang telah dibuat. Pengujian dalam kondisi pertama ini dapat dipantau melalui aplikasi android dan juga pada layar LCD yang terdapat pada gardu panel.

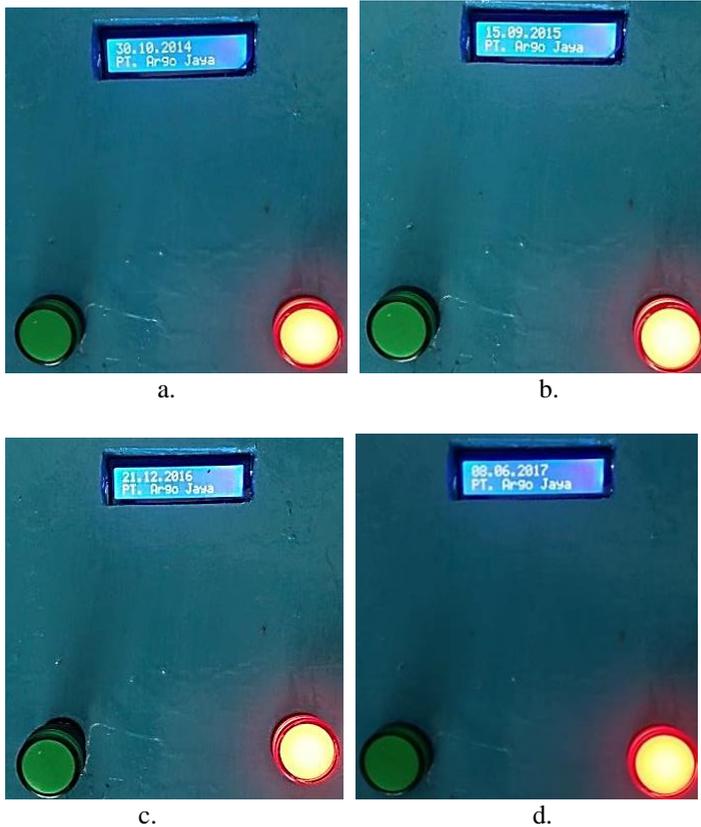
Pada kondisi gardu dalam keadaan normal hal ini berarti masih tertutupnya semua saklar yang ada pada gardu tersebut. Sehingga yang tertampil dalam layar android adalah status dari masing-masing saklar

masih dalam keadaan tertutup. Tampilan pada android ketika kondisi gardu dalam kondisi normal dapat dilihat pada Gambar 4.19.



Gambar 4.19 Kondisi Normal Gardu pada Layar Android

Sedangkan tampilan gardu dalam kondisi normal pada layar LCD dapat dilihat pada Gambar 4.20.



Gambar 4.20 Kondisi Gardu Normal pada Layar LCD

Dalam Gambar 4.20, gambar a menunjukkan kondisi gardu beroperasi secara normal pada tanggal 30 Oktober 2014, gambar b menunjukkan kondisi gardu secara normal pada tanggal 15 September 2015, gambar c menunjukkan kondisi gardu normal pada tanggal 21 Desember 2016, serta gambar d menunjukkan kondisi gardu normal pada tanggal 8 Juni 2017. Kondisi gardu dalam keadaan normal ditandai dengan adanya lampu yang berwarna merah dalam artian gardu tersebut

dalam kondisi bertegangan serta semua saklar yang ada di dalamnya dalam keadaan tertutup.

4.10.2 Pengujian dengan Kondisi Kedua

Pengujian ini dilakukan dengan memperhatikan kondisi gardu yaitu ketika gardu dalam kondisi padam sesuai dengan jadwal yang telah ditentukan. Kondisi padamnya gardu ini dilakukan ketika operator akan melakukan pemeliharaan. Padamnya gardu konvensional tersebut menyebabkan tidak adanya tegangan yang ada pada gardu karena saklar yang ada pada CB, LBS *incoming*, dan LBS *outgoing* dalam keadaan terbuka. Sementara itu, untuk memonitoring pembukaan dan penutupan saklar di masing-masing sambungan maka petugas dapat menggunakan aplikasi yang ada pada android yang mana untuk pembukaan saklar yang terdapat pada *prototype* gardu akan dilakukan oleh petugas dengan menekan tombol yang ada pada gardu secara berurutan. Adapun untuk proses monitoring yang dilakukan oleh petugas didasarkan pada 7 kondisi yaitu antara lain:

1. Kondisi pembukaan saklar pada CB

Untuk membuka saklar CB maka petugas harus menekan tombol 1 yang terdapat pada *prototype* gardu. Selanjutnya ketika CB telah terbuka maka status CB pada layar android yang semula dalam keadaan tertutup dan bertegangan 20 KV akan berubah menjadi terbuka sedangkan tegangan yang terdapat di dalamnya akan berubah menjadi 0. Sedangkan untuk kondisi gardu ketika saklar dibuka maka kondisi tersebut akan berubah dari yang semula menampilkan status gardu bertegangan menjadi gardu masih bertegangan. Adanya status gardu masih bertegangan menandakan bahwa setelah dilakukan pembukaan saklar pada CB maka gardu portal konvensional masih belum benar-benar aman yang ditandai dengan masih adanya tegangan pada gardu portal. Sehingga operator PLN harus melakukan langkah selanjutnya agar gardu portal dapat dipadamkan sehingga petugas dapat melakukan pemeliharaan peralatan pada gardu portal konvensional. Adapun untuk hasil pengujian ini dapat dilihat pada Gambar 4.21.



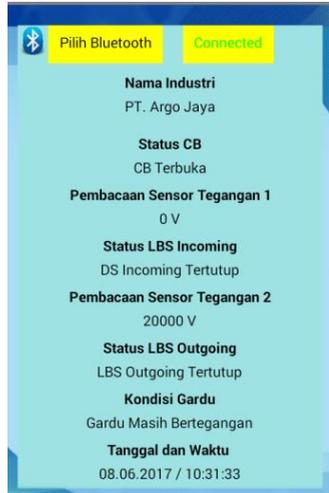
a.



b.



c.



d.

Gambar 4.21 Pembukaan Saklar CB

Dalam gambar tersebut gambar a merupakan status CB yang dimonitor pada tanggal 30 Oktober 2014, gambar b

merupakan status CB yang dimonitor pada tanggal 15 September 2015, gambar c merupakan status CB yang dimonitor pada tanggal 21 Desember 2016, sedangkan gambar d merupakan status CB yang dimonitor pada tanggal 08 Juni 2017. Sedangkan pada LCD ketika CB dalam kondisi terbuka maka LCD akan menampilkan teks “Industri Padam” hal ini dapat dilihat pada Gambar 4.22.



Gambar 4.22 Status LCD Industri Padam

Pada Gambar 4.22 tersebut pengujian LCD untuk kondisi pembukaan saklar CB dilakukan pada tanggal 8 Juni 2017. Pembukaan CB ini dimaksudkan untuk memutus jaringan antara gardu konvensional dengan beban yang ada pada pelanggan industri.

2. Kondisi Pembukaan Saklar pada LBS *Incoming*

Pembukaan saklar LBS *incoming* dilakukan dengan menekan tombol 2 yang ada pada *prototype* gardu. Selanjutnya ketika DS *incoming* telah terbuka maka status LBS *incoming* pada layar android yang semula dalam keadaan tertutup dan bertegangan 20 KV akan berubah menjadi terbuka sedangkan tegangan yang terdapat di dalamnya akan berubah menjadi 0. Untuk kondisi gardu ketika saklar dibuka maka kondisi tersebut akan berubah dari yang semula menampilkan status gardu masih bertegangan menjadi gardu bebas bertegangan. Hasil pengujian ini dapat dilihat pada Gambar 4.23.



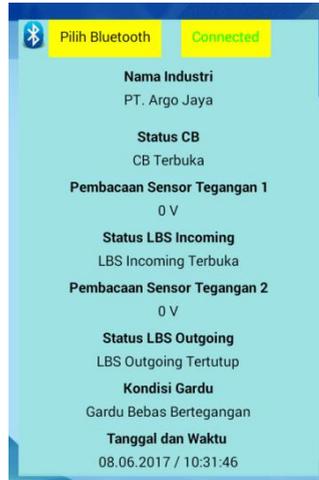
a.



b.



c.



d.

Gambar 4.23 Pembukaan Saklar DS *Incoming*

Dalam Gambar 4.23, gambar a merupakan status LBS *incoming* yang dimonitor pada tanggal 30 Oktober 2014, gambar b merupakan status LBS *incoming* yang dimonitor pada

tanggal 15 September 2015, gambar c merupakan status LBS *incoming* yang dimonitor pada tanggal 21 Desember 2016, sedangkan gambar d merupakan status LBS *incoming* yang dimonitor pada tanggal 08 Juni 2017. Begitu juga untuk LCD yang terdapat pada panel ketika LBS *outgoing* terbuka maka LCD akan menunjukkan teks yang menampilkan “Gardu Padam”. Untuk pengujian status dari LCD dalam kondisi LBS *incoming* terbuka dapat dilihat pada Gambar 4.24.



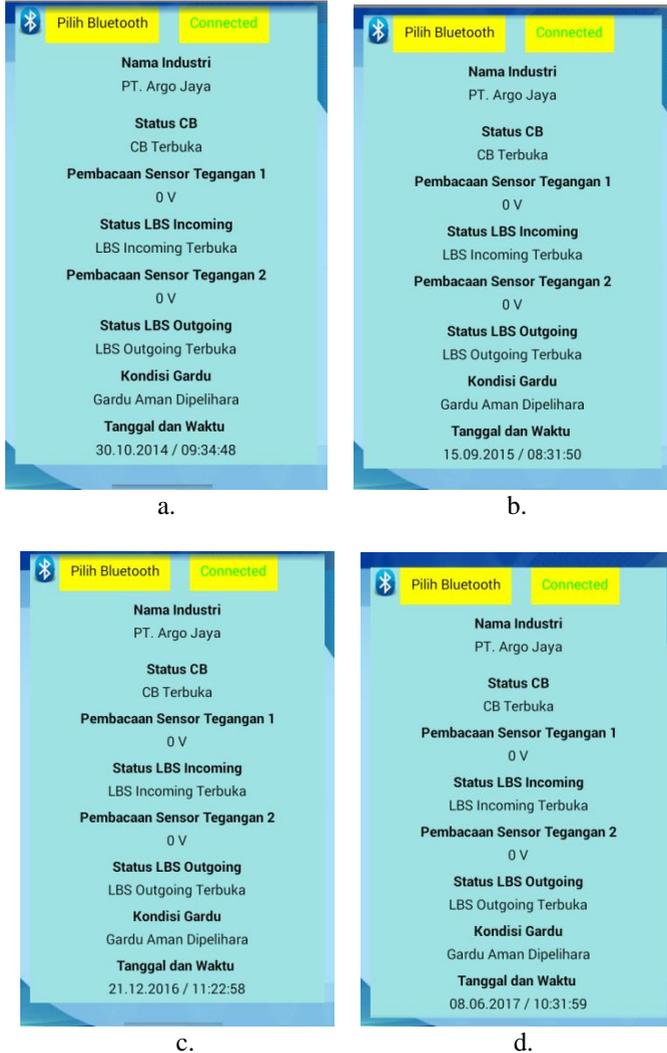
Gambar 4.24 Status LCD Gardu Padam

Pada Gambar 4.24 tersebut pengujian LCD untuk kondisi pembukaan saklar LBS *incoming* dilakukan pada tanggal 8 Juni 2017. Pembukaan LBS *incoming* ini dimaksudkan untuk memutus jaringan antara gardu konvensional dengan SUTM pada jaringan PLN.

3. Kondisi Pembukaan Saklar pada LBS *Outgoing*

Pembukaan saklar LBS *outgoing* dilakukan dengan menekan tombol 3 yang ada pada *prototype* gardu. Selanjutnya ketika LBS *outgoing* telah terbuka maka status LBS *outgoing* pada layar android yang semula dalam keadaan tertutup dan bertegangan 20 KV akan berubah menjadi terbuka sedangkan tegangan yang terdapat di dalamnya akan berubah menjadi 0. Untuk kondisi gardu ketika saklar dibuka maka kondisi tersebut akan berubah dari yang semula menampilkan status

gardu bebas bertegangan menjadi gardu aman dipelihara. Hasil pengujian ini dapat dilihat pada Gambar 4.25.



Gambar 4.25 Pembukaan Saklar LBS *Outgoing*

Dalam gambar tersebut gambar a merupakan status LBS *outgoing* yang dimonitor pada tanggal 30 Oktober 2014, gambar b merupakan status LBS *outgoing* yang dimonitor pada tanggal 15 September 2015, gambar c merupakan status LBS *outgoing* yang di monitor pada tanggal 21 Desember 2016, sedangkan gambar d merupakan status LBS *outgoing* yang dimonitor pada tanggal 08 Juni 2017. Sedangkan pada layar LCD ketika LBS *outgoing* terbuka maka layar LCD akan menampilkan *text* “Maintenance READY”. Hal ini dapat dilihat pada Gambar 4.26.



Gambar 4.26 Status LCD *Maintenance Ready*

Pembukaan LBS *incoming* ini dimaksudkan untuk memutus jaringan antara sel *incoming* yang ada pada gardu dengan sel *outgoing* dengan tujuan untuk menghilangkan tegangan yang ada pada gardu. Sedangkan, pada Gambar 4.26 lampu yang terdapat pada panel gardu yang semula berwarna merah berubah menjadi hijau. Hal ini menandakan kondisi dari gardu konvensional sudah bebas dari tegangan sehingga operator dapat melanjutkan ke tahapan pemeliharaan gardu.

4. Kondisi Bebas Tegangan

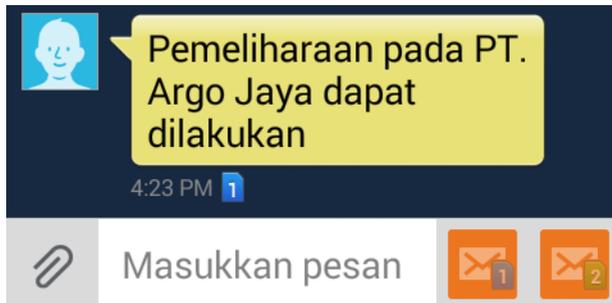
Kondisi awal bebas tegangan pada gardu konvensional adalah masih terbukanya semua saklar yang ada pada gardu tersebut. Sehingga yang tertampil dalam layar android adalah

status dari masing-masing saklar masih dalam keadaan terbuka. Sementara itu lampu yang tertampil pada *prototype* panel akan menyala hijau yang menandakan gardu dalam keadaan bebas tegangan. Kondisi bebas tegangan ini dapat dilihat pada Gambar 4.27.



Gambar 4.27 Gardu Bebas Tegangan

Selanjutnya ketika gardu konvensional telah bebas bertegangan maka modem SIM900 akan mengirimkan sms pemberitahuan yang ditunjukkan kepada operator pusat. Untuk sms pemberitahuan tersebut dapat dilihat pada Gambar 4.28.



Gambar 4.28 Pengiriman SMS Setelah Gardu Bebas Tegangan

Pada Gambar 4.28, dalam pengiriman SMS menggunakan modem GSM SIM 900. Setelah sensor peak detektor yang dipasang pada coil relai mendeteksi tegangan $> 1,7 \text{ V}$ maka relai yang terdapat pada CB, LBS *incoming*, dan LBS *outgoing* akan berubah dari yang semula *normally close* (tertutup) menjadi *normally open* (terbuka) dalam keadaan terbuka sehingga arduino akan mengolah data tersebut dan mengirimkan perintah AT+CMGF=1 ke SIM 900 sehingga sms dapat terkirim ke operator pusat yang berisi “Pemeliharaan pada PT Argo Jaya dapat dilakukan”.

5. Kondisi Penutupan Saklar pada LBS *Outgoing*

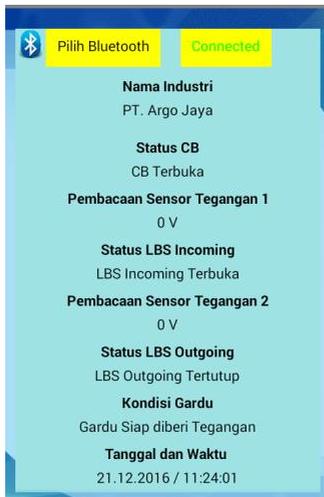
Setelah gardu konvensional bebas bertegangan petugas dapat melakukan pemeliharaan gardu konvensional. Dan setelah dilakukan pemeliharaan untuk mengoperasikan gardu kembali seperti semula maka dapat dilakukan dengan penutupan saklar LBS *outgoing* yaitu dengan menekan tombol 4 yang terdapat pada *prototype* gardu. Selanjutnya ketika LBS *outgoing* telah tertutup maka status LBS *outgoing* pada layar android yang semula dalam keadaan terbuka dan bertegangan 0 KV akan berubah menjadi tertutup sedangkan tegangan yang terdapat di dalamnya akan berubah menjadi 20 KV. Sedangkan untuk kondisi gardu ketika saklar ditutup maka kondisi tersebut akan berubah dari yang semula menampilkan status gardu aman dipelihara menjadi gardu siap diberi tegangan. Hasil pengujian ini dapat dilihat pada Gambar 4.29.



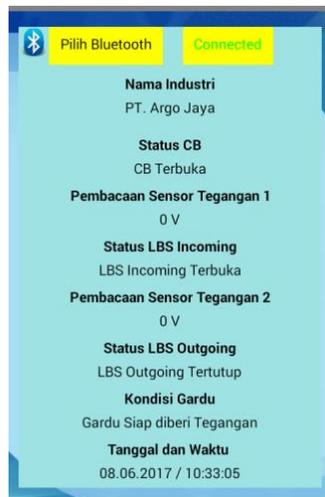
a.



b.



c.



d.

Gambar 4.29 Penutupan Saklar LBS *Outgoing*

Dalam gambar tersebut gambar a merupakan status LBS *outgoing* yang dimonitor pada tanggal 30 Oktober 2014,

gambar b merupakan status LBS *outgoing* yang dimonitor pada tanggal 15 September 2015, gambar c merupakan status LBS *outgoing* yang dimonitor pada tanggal 21 Desember 2016, sedangkan gambar d merupakan status LBS *outgoing* yang dimonitor pada tanggal 08 Juni 2017. Sedangkan pada layar LCD ketika LBS *outgoing* tertutup maka layar LCD akan menampilkan *text* agar petugas melaksanakan langkah selanjutnya dalam pengoperasian gardu. Adapun *text* yang tertampil dalam LCD adalah “Masukkan LBS *Incoming*”. Hal ini dapat dilihat pada Gambar 4.30.



Gambar 4.30 Status LCD Masukkan LBS *Incoming*

Pada Gambar 4.30 tersebut pengujian LCD untuk kondisi penutupan saklar LBS *outgoing* dilakukan pada tanggal 8 Juni 2017. Pembukaan LBS *outgoing* ini dimaksudkan untuk mengembalikan kondisi gardu agar kembali dalam kondisi bertegangan, dalam hal ini jaringan antara LBS *outgoing* dan LBS *incoming* kembali disambungkan.

6. Kondisi Penutupan Saklar pada LBS *Incoming*

Untuk penutupan saklar LBS *incoming* yaitu dengan menekan tombol 5 yang terdapat pada *prototype* panel. Selanjutnya ketika LBS *incoming* telah tertutup maka status LBS *incoming* pada layar android yang semula dalam keadaan terbuka dan bertegangan 0 KV akan berubah menjadi tertutup sedangkan tegangan yang terdapat di dalamnya akan berubah menjadi 20 KV. Sedangkan untuk kondisi gardu ketika saklar ditutup maka kondisi tersebut akan berubah dari yang semula menampilkan status gardu siap diberi tegangan menjadi gardu

bertegangan. Hasil pengujian ini dapat dilihat pada Gambar 4.31.



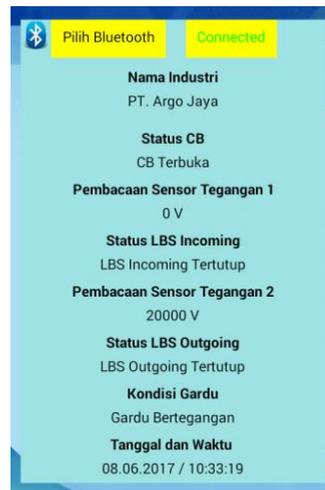
a.



b.



c.



d.

Gambar 4.31 Penutupan Saklar pada LBS *Incoming*

Dalam gambar tersebut gambar a merupakan status LBS *incoming* yang dimonitor pada tanggal 30 Oktober 2014, gambar b merupakan status LBS *incoming* yang dimonitor pada tanggal 15 September 2015, gambar c merupakan status LBS *incoming* yang dimonitor pada tanggal 21 Desember 2016, sedangkan gambar d merupakan status LBS *incoming* yang dimonitor pada tanggal 08 Juni 2017. Untuk layar LCD ketika LBS *incoming* ditutup maka layar LCD akan menampilkan *text* “Gardu Normal”. Hal ini dapat dilihat pada Gambar 4.32.



Gambar 4.32 Status LCD Gardu Normal

Pada Gambar 4.32 status LCD untuk gardu normal menandakan gardu konvensional sudah kembali bertegangan. Dalam hal ini jaringan antara LBS *incoming* dan SUTM pada jaringan PLN sudah tersambung kembali. Pada Gambar 4.32 tersebut tampilan LCD yang menampilkan kondisi penutupan saklar LBS *outgoing* dilakukan pada tanggal 8 Juni 2017.

7. Kondisi penutupan saklar pada CB

Untuk penutupan saklar CB yaitu dengan menekan tombol 6. Selanjutnya ketika CB telah tertutup maka status CB pada layar android yang semula dalam keadaan terbuka dan bertegangan 0 KV akan berubah menjadi tertutup sedangkan tegangan yang terdapat di dalamnya akan berubah menjadi 20 KV. Sedangkan untuk kondisi gardu ketika saklar ditutup maka kondisi tersebut akan berubah dari yang semula menampilkan status gardu siap diberi tegangan menjadi gardu

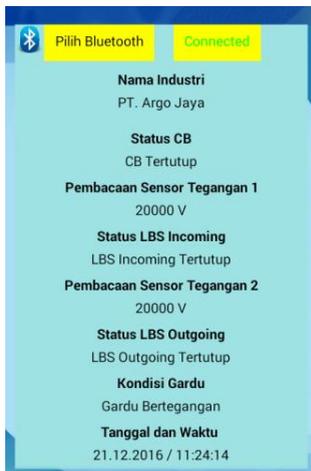
bertegangan. Hasil pengujian ini dapat dilihat pada Gambar 4.33.



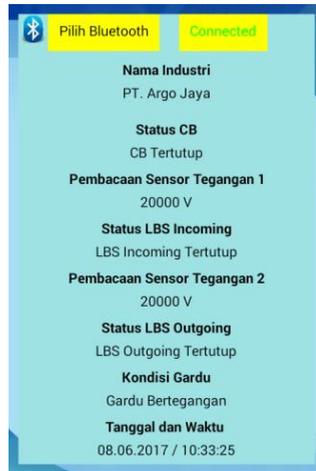
a.



b.



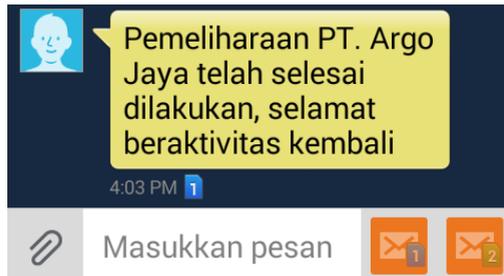
c.



d.

Gambar 4.33 Penutupan Saklar pada CB

Dalam gambar tersebut gambar a merupakan status CB yang dimonitor pada tanggal 30 Oktober 2014, gambar b merupakan status CB yang dimonitor pada tanggal 15 September 2015, gambar c merupakan status CB yang dimonitor pada tanggal 21 Desember 2016, sedangkan gambar d merupakan status LBS *incoming* yang dimonitor pada tanggal 08 Juni 2017. Selanjutnya ketika gardu konvensional telah bertegangan maka modem SIM900 akan mengirimkan sms pemberitahuan seperti Gambar 4.34.



Gambar 4.34 Pengiriman SMS Setelah Gardu Telah Dilakukan Pemeliharaan

Pada Gambar 4.34, dalam pengiriman SMS menggunakan modem GSM SIM 900. Setelah sensor peak detektor yang dipasang pada coil relai mendeteksi tegangan $< 1,7$ V maka relai yang terdapat pada CB, LBS *incoming*, dan LBS *outgoing* akan berubah dari yang semula *normally open* (terbuka) menjadi *normally close* (tertutup) dalam keadaan terbuka sehingga arduino akan mengolah data tersebut dan mengirimkan perintah AT+CMGF=1 ke SIM 900 sehingga sms dapat terkirim ke operator pusat yang berisi “Pemeliharaan pada PT Argo Jaya telah selesai dilakukan, selamat beraktivitas kembali”.

Pada sistem pengoperasian gardu konvensional, setelah petugas melakukan pemutusan tegangan pada gardu dan setelah petugas melakukan penyambungan kembali tegangan pada gardu maka SIM 900 akan mengirim pemberitahuan kepada operator pusat. Sedangkan untuk pengujian pengiriman sms tersebut dapat dilihat pada Tabel 4.9.

Tabel 4.9 Waktu Pengiriman SIM900

No	Waktu setelah gardu bebas tegangan (padam)	Waktu setelah dioperasikan kembali (normal)	Tanggal Pemeliharaan
1.	12 detik	18 detik	30 Oktober 2014
2.	19 detik	10 detik	15 September 2015
3.	14 detik	08 detik	21 Desember 2016
4.	18 detik	17 detik	08 Juni 2017

Pada pengujian komunikasi SIM 900 lamanya waktu pengiriman ketika gardu bebas bertegangan dan ketika gardu telah dioperasikan kembali ketika dirata-rata adalah sebesar 14,05 detik. Hasil ini didapat berdasarkan rumus.

$$\begin{aligned} \text{waktu gardu padam} &= \frac{\text{jumlah waktu gardu padam}}{4} \\ &= \frac{12+19+14+18}{4} = 15,75 \text{ detik} \end{aligned}$$

$$\begin{aligned} \text{waktu gardu normal} &= \frac{\text{jumlah waktu gardu normal}}{4} \\ &= \frac{18+10+8+17}{4} = 13,25 \text{ detik} \end{aligned}$$

Sehingga, untuk waktu pengiriman SMS didapat dari

$$\begin{aligned} \text{waktu pengiriman} &= \frac{\text{waktu gardu padam} + \text{waktu gardu normal}}{2} \\ &= \frac{15,75+13,25}{2} = 14,05 \text{ detik} \end{aligned}$$

Waktu pengiriman ini tergantung pada kekuatan sinyal pada provider yang digunakan serta lamanya operator mengoperasikan gardu. Sedangkan dalam pengiriman SMS menggunakan modem GSM SIM 900.

Selanjutnya setelah operator telah selesai melakukan pembebasan tegangan pada gardu maupun telah selesai mengembalikan gardu ke keadaan semula maka operator dapat menyimpan pengoperasian tersebut

ke dalam database. Tampilan dari database tersebut dapat dilihat pada Gambar 4.35.

Nama perusahaan	Status CB	Sensor Tegangan 1	Status LBS Incoming	Sensor Tegangan 2	Status LBS Outgoing	Kondisi Gardu	Tanggal dan Waktu
PT. Argo Jaya	Terbuka	0 V	LBS Incoming Te	0 V	LBS Outgoing Te	Gardu Aman Dipe	30.10.2014 09
PT. Argo Jaya	Terbuka	0 V	LBS Incoming Te	0 V	LBS Outgoing Te	Gardu Dipeihaar	30.10.2014 09
PT. Argo Jaya	Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	30.10.2014 09:30:39
PT. Argo Jaya	Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipeihaar	30.10.2014 09:34:48
PT. Argo Jaya	Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	30.10.2014 09:39:24
PT. Argo Jaya	Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipeihaar	15.09.2015 08:31:50
PT. Argo Jaya	Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	15.09.2015 08:34:06
PT. Argo Jaya	Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipeihaar	21.12.2016 11:22:38
PT. Argo Jaya	Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	21.12.2016 11:25:19
PT. Argo Jaya	Terbuka	0 V	LBS Incoming Terbuka	0 V	LBS Outgoing Terbuka	Gardu Aman Dipeihaar	08.06.2017 10:31:59
PT. Argo Jaya	Tertutup	20000 V	LBS Incoming Tertutup	20000 V	LBS Outgoing Tertutup	Gardu Bertegangan	08.06.2017 10:33:25

Gambar 4.35 Database Pengoperasian Gardu Konvensional

Dalam gambar 4.35 riwayat pengoperasian gardu portal konvensional dapat diketahui sehingga memudahkan operator untuk membandingkan lamanya pengoperasian gardu portal yang dilakukan dengan waktu sebelumnya.

4.10.3 Pengujian dengan Kondisi Ketiga

Pengujian dengan kondisi ketiga dilakukan diluar jadwal pemeliharaan dan dalam kondisi ini gardu bekerja secara normal. Untuk kondisi ketiga ini waktu dibuat dengan memperhatikan riwayat pemeliharaan gardu yang dilakukan mulai tahun 2014 sampai tahun 2017. Maka untuk pengujian ini waktu ditentukan pada tanggal 13 Januari 2014 dan 14 Juni 2016, dua waktu tersebut tidak terdapat di jadwal pemeliharaan dan khusus dilakukan untuk memantau kinerja dari gardu konvensional. Untuk hasil pengujian tersebut dapat dilihat pada Gambar 4.36, Gambar 4.37, Gambar 4.38, dan Gambar 4.39.



Gambar 4.36 Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Beroperasi Secara Normal pada Tanggal 13 Januari 2014

Pada Gambar 4.36 tampilan LCD yang menunjukkan tanggal 13 Januari 2014 menandakan bahwa gardu portal konvensional telah beroperasi secara normal walaupun sedang tidak terdapat adanya jadwal pemeliharaan.



Gambar 4.37 Kondisi Lampu Beban pada Tanggal 13 Januari 2014

Untuk Gambar 4.37 kondisi normal gardu ditandai dengan adanya lampu pada panel yang berwarna merah yang menyatakan gardu konvensional tersebut dalam kondisi bertegangan. Selain itu kondisi ini

juga ditandai dengan adanya lampu beban yang menyala seperti pada Gambar 4.37. Sedangkan, pada Gambar 4.38 tampilan LCD yang menunjukkan tanggal 14 Juni 2016 menandakan bahwa gardu portal konvensional telah beroperasi secara normal walaupun sedang tidak terdapat adanya jadwal pemeliharaan.



Gambar 4.38 Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Beroperasi Secara Normal pada Tanggal 14 Juni 2016

Untuk Gambar 4.39 kondisi normal gardu dengan lampu panel yang berwarna merah yang menyatakan gardu konvensional dalam kondisi bertegangan. Selain itu kondisi ini juga ditandai dengan adanya lampu beban yang menyala dan ditampilkan pada Gambar 4.39.



Gambar 4.39 Kondisi Lampu Beban pada Tanggal 14 Juni 2014

4.10.4 Pengujian dengan Kondisi Keempat

Pada pengujian dengan kondisi keempat ini dilakukan ketika tidak adanya jadwal pemeliharaan akan tetapi gardu padam. Kondisi keempat ini menyatakan bahwa gardu konvensional telah terjadi gangguan. Adapun gangguan yang disebabkan antara lain adalah adanya kegagalan operasi dari komponen yang terdapat di dalam gardu, CB yang tidak dapat trip, maupun adanya gangguan atau kerusakan pada kabel tanah SUTM yang mensuplai gardu. Dalam kondisi keempat ini pengujian dilakukan dengan memperhatikan lampu beban yang telah disediakan. Apabila lampu beban dimatikan maka lampu dari LBS *outgoing* juga dalam keadaan padam, sementara itu pada panel LCD akan menunjukkan tampilan terjadi gangguan. Pengujian dengan kondisi keempat ini dilakukan diluar jadwal pemeliharaan yang telah ditentukan. Dimana dalam pengujian keempat ini gangguan ditentukan terjadi pada tanggal 15 Maret 2014 dan 22 April 2017. Adapun untuk pengujian dengan kondisi keempat ini dapat dilihat pada Gambar 4.40 dan Gambar 4.41.



Gambar 4.40 Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Padam pada Tanggal 15 Maret 2014.



Gambar 4.41 Tampilan LCD Kondisi Gardu Ketika Tidak Ada Jadwal dan Padam pada Tanggal 22 April 2017.

Untuk kondisi gardu dengan lampu beban dalam kondisi mati dapat dilihat pada Gambar 4.42. Sementara itu, untuk tampilan LCD yang menyatakan gardu sedang mengalami gangguan dapat dilihat pada Gambar 4.43.

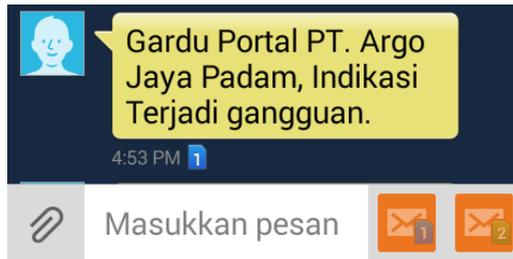


Gambar 4.42 Kondisi Gardu dengan Lampu Beban dalam Kondisi Padam



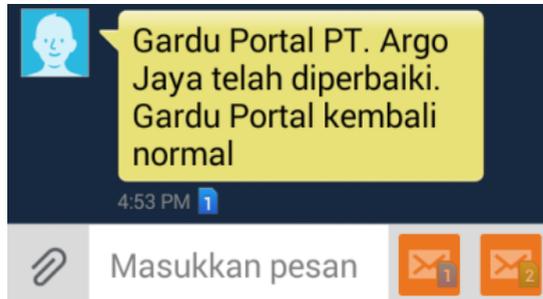
Gambar 4.43 Tampilan LCD Untuk Kondisi Gardu Terjadi Gangguan

Sementara itu ketika lampu beban dalam kondisi padam, maka SIM900 akan mengirimkan notifikasi berupa SMS ke operator pusat sehingga PLN dapat mengirimkan petugas ke lokasi tersebut tanpa menunggu adanya pemberitahuan dari pihak pelanggan. Dari SMS tersebut dapat diketahui tanggal dan waktu terjadinya gangguan. Adapun pemberitahuan terjadinya gangguan dapat dilihat pada Gambar 4.44



Gambar 4.44 Pengiriman SMS Indikasi Terjadi Gangguan pada Gardu

Setelah gardu selesai diperbaiki maka SIM900 juga akan mengirimkan notifikasi kembali berupa SMS ke operator pusat. Adapun pemberitahuan tersebut dapat dilihat pada Gambar 4.45.



Gambar 4.45 Pengiriman SMS Gardu Telah Selesai Diperbaiki

Pada Gambar 4.46, dalam pengiriman SMS menggunakan modem GSM SIM 900. Setelah saklar pada lampu beban ditekan maka relai akan dalam kondisi HIGH sehingga arduino mega akan mengolah data tersebut dan mengirimkan perintah AT+CMGF=1 ke SIM 900 sehingga sms dapat terkirim ke operator pusat yang berisi “Gardu Portal PT. Argo Jaya padam, Indikasi terjadi gangguan”. Sedangkan untuk Gambar 4.45, dalam pengiriman SMS menggunakan modem GSM SIM 900. Setelah saklar pada lampu beban ditekan kembali (dikembalikan ke posisi semula) maka relai akan dalam kondisi LOW sehingga arduino mega akan mengolah data tersebut dan mengirimkan perintah AT+CMGF=1 ke SIM 900 sehingga sms dapat terkirim ke operator pusat yang berisi “Gardu Portal Argo Jaya telah diperbaiki, Gardu Portal kembali normal”.

4.11 Analisa Relevansi

Untuk analisa relevansi dari alat ini, untuk sensor peak detektor yang digunakan harus disesuaikan dengan standar rating yang dibutuhkan Potensial Transformer yang terdapat pada gardu portal 20KV. Sementara itu untuk modul bluetooth yang digunakan sebagai media pengiriman data, harus menggunakan modul bluetooth yang memiliki jangkauan jarak koneksi yang luas. Untuk penggunaan SIM900 sebagai media pengiriman SMS maka provider yang digunakan harus memiliki sinyal yang kuat sesuai dengan kondisi lokasi gardu tersebut ditempatkan. sementara itu, jika alat ini digunakan pada gardu portal konvensional di suatu industri maka dapat digunakan sebagai pemantauan operasi dan sebagai pelaporan kondisi gardu sehingga dapat meningkatkan keselamatan dalam pengoperasian.

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

BAB V

PENUTUP

Dari pembuatan tugas akhir mengenai “Sistem Pelaporan Pengoperasian Gardu Konvensional Satu Fasa dalam Kondisi Pemeliharaan” berdasarkan pengujian perangkat keras (*hardware*) hingga perangkat lunak (*software*) dapat ditarik kesimpulan dan saran sebagai berikut.

5.1 Kesimpulan

Dalam pengoperasian gardu portal konvensional lebih mudah karena ditunjang dengan adanya aplikasi android yang digunakan untuk memonitor tahapan pengoperasian gardu sehingga apabila buka tutupnya saklar dari CB, LBS *incoming*, dan LBS *outgoing* dapat dimonitoring dengan android. Untuk mengindikasikan saklar itu terbuka atau tertutup dapat diketahui dengan sensor peak detektor, ketika sensor mendeteksi tegangan $>1,7$ V maka saklar bekerja membuka begitupula sebaliknya. Aplikasi android ini ditunjang dengan adanya penyimpanan database berbasis web yang dapat diakses secara *offline* sehingga mempermudah operator untuk menyimpan riwayat pengoperasian gardu. Selain itu, untuk pelaporan dengan menggunakan SIM900 sebagai media SMS memiliki rata-rata pengiriman sebesar 14,50 detik, hasil ini didapat dari rata-rata pengiriman SMS saat dilakukan pemadaman gardu dan penormalan gardu kembali.

Waktu pengiriman ini dipengaruhi dengan lamanya operator melakukan pengoperasian dan baik buruknya sinyal dari *provider* yang digunakan.

5.2 Saran

Saran untuk mengembangkan tugas akhir ini adalah pemilihan provider SIM card harus diperhatikan dengan baik agar tidak menghambat proses pengiriman SMS. Dalam pengembangan lebih lanjut disarankan untuk menambahkan fitur jenis gangguan lain yang dapat mengirimkan notifikasi selain padamnya lampu beban.

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

DAFTAR PUSTAKA

- 1] _____, “*SOP (Standing Operating Procedure)*”, Pusat Pelatihan dan Pendidikan PT. PLN (Persero), Jakarta, 2010.
- 2] Suswanto Daman, “*Sistem Distribusi Tenaga Listrik*”, Teknik Elektro Fakultas Teknik Universitas Negeri Padang, Padang, 2009.
- 3] _____, “*Gardu Induk Sisi 20 KV*”, Pusat Pelatihan dan Pendidikan PT. PLN (Persero), Jakarta, 2015.
- 4] Andarbeni Mintuno Dison, “*Pemeliharaan Kubikel 20 KV Gardu Induk PT. PLN (Persero) Area Ciledug*”, Laporan Kerja Praktek, Teknik Elektro Universitas Mercu Buana, Jakarta, 2014.
- 5] Santoso Hari, “*Panduan Praktis Arduino untuk Pemula*”, Elangsakti, Trenggalek, 2015
- 6] Barizi Alfian dan Yogi Moch, “*Prototype Sistem Pelaporan Gangguan Beserta Posisi Gangguan pada Jaringan Distribusi*”, Tugas Akhir, Program D3 Teknik Elektro FTI-ITS, Surabaya, 2015.
- 7] Andi, “*Pemrograman Android dengan APP Inventor No Experience Required*”, Wahana Komputer, Yogyakarta, 2013.
- 8] Raharjo Budi, “*Belajar Otodidak MySQL*”, Informatika, Bandung, 2015.
- 9] Saraswati Hana Nafita dan Nurafian Rifki, “*Smart Electricity Meter*”, Tugas Akhir, Program D3 Teknik Elektro FTI-ITS, Surabaya, 2015.
- 10] _____, “*DS1307 64x8 Serial real Time Clock*”, Datasheet, Dallas Semiconductor, 2006.

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

LAMPIRAN A

A.1. Listing Program Arduino IDE pada Arduino Mega

A.1.1 Program Uji Pin Arduino Mega2560

```
//Program Logika 1
void setup() {
  pinMode(A0,OUTPUT);
  pinMode(A1,OUTPUT);
  pinMode(A2,OUTPUT);
  pinMode(A3,OUTPUT);
  pinMode(A4,OUTPUT);
  pinMode(A5,OUTPUT);
  pinMode(A6,OUTPUT);
  pinMode(A7,OUTPUT);
  pinMode(0,OUTPUT);
  pinMode(1,OUTPUT);
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);
  pinMode(7,OUTPUT);
  pinMode(8,OUTPUT);
  pinMode(9,OUTPUT);
  pinMode(10,OUTPUT);
  pinMode(11,OUTPUT);
  pinMode(12,OUTPUT);
  pinMode(13,OUTPUT);
  pinMode(14,OUTPUT);
  pinMode(15,OUTPUT);
  pinMode(16,OUTPUT);
  pinMode(17,OUTPUT);
  pinMode(18,OUTPUT);
  pinMode(19,OUTPUT);
  pinMode(20,OUTPUT);
  pinMode(21,OUTPUT);
}
void loop() {
  digitalWrite(A0,HIGH);
```

```
digitalWrite(A1,HIGH);
digitalWrite(A2,HIGH);
digitalWrite(A3,HIGH);
digitalWrite(A4,HIGH);
digitalWrite(A5,HIGH);
digitalWrite(A6,HIGH);
digitalWrite(A7,HIGH);
digitalWrite(0,HIGH);
digitalWrite(1,HIGH);
digitalWrite(2,HIGH);
digitalWrite(3,HIGH);
digitalWrite(4,HIGH);
digitalWrite(5,HIGH);
digitalWrite(6,HIGH);
digitalWrite(7,HIGH);
digitalWrite(8,HIGH);
digitalWrite(9,HIGH);
digitalWrite(10,HIGH);
digitalWrite(11,HIGH);
digitalWrite(12,HIGH);
digitalWrite(13,HIGH);
digitalWrite(14,HIGH);
digitalWrite(15,HIGH);
digitalWrite(16,HIGH);
digitalWrite(17,HIGH);
digitalWrite(18,HIGH);
digitalWrite(19,HIGH);
digitalWrite(20,HIGH);
digitalWrite(21,HIGH);
}
//Program Logika 0

void setup() {
pinMode(A0,OUTPUT);
pinMode(A1,OUTPUT);
pinMode(A2,OUTPUT);
pinMode(A3,OUTPUT);
pinMode(A4,OUTPUT);
pinMode(A5,OUTPUT);
```

```
pinMode(A6,OUTPUT);
pinMode(A7,OUTPUT);
pinMode(0,OUTPUT);
pinMode(1,OUTPUT);
pinMode(2,OUTPUT);
pinMode(3,OUTPUT);
pinMode(4,OUTPUT);
pinMode(5,OUTPUT);
pinMode(6,OUTPUT);
pinMode(7,OUTPUT);
pinMode(8,OUTPUT);
pinMode(9,OUTPUT);
pinMode(10,OUTPUT);
pinMode(11,OUTPUT);
pinMode(12,OUTPUT);
pinMode(13,OUTPUT);
pinMode(14,OUTPUT);
pinMode(15,OUTPUT);
pinMode(16,OUTPUT);
pinMode(17,OUTPUT);
pinMode(18,OUTPUT);
pinMode(19,OUTPUT);
pinMode(20,OUTPUT);
pinMode(21,OUTPUT);
}
void loop() {
digitalWrite(A0,LOW);
digitalWrite(A1,LOW);
digitalWrite(A2, LOW);
digitalWrite(A3, LOW);
digitalWrite(A4, LOW);
digitalWrite(A5, LOW);
digitalWrite(A6, LOW);
digitalWrite(A7, LOW);
digitalWrite(0, LOW);
digitalWrite(1, LOW);
digitalWrite(2, LOW);
digitalWrite(3, LOW);
digitalWrite(4, LOW);
```

```
digitalWrite(5, LOW);
digitalWrite(6, LOW);
digitalWrite(7, LOW);
digitalWrite(8, LOW);
digitalWrite(9, LOW);
digitalWrite(10, LOW);
digitalWrite(11, LOW);
digitalWrite(12, LOW);
digitalWrite(13, LOW);
digitalWrite(14, LOW);
digitalWrite(15, LOW);
digitalWrite(16, LOW);
digitalWrite(17, LOW);
digitalWrite(18, LOW);
digitalWrite(19, LOW);
digitalWrite(20, LOW);
digitalWrite(21, LOW);
}
```

A.1.2 Program Uji LCD

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
  lcd.begin(16,2);
  lcd.setCursor(0,0);
  lcd.print("COBA LCD 16x2");
  lcd.setCursor(0,1);
  lcd.print("SUKSES");
}
void loop() {
}
```

A.1.3 Program Uji RTC

```
#include <Wire.h>
#include <Time.h>
#include <DS1307RTC.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11,5,4,3,2);
```

```

void setup() {
  lcd.begin(16,2);
  while (!Serial);
  delay(200);
}
void loop() {
  tmElements_t tm;
  if (RTC.read(tm)) {
    lcd.clear();
    lcd.print("Date: ");
    printkurangdarinol(tm.Day);
    lcd.print("-");
    printkurangdarinol(tm.Month);
    lcd.print("-");
    lcd.print(tm.YearToCalendar(tm.Year));

    lcd.setCursor(0,1);
    lcd.print("Time: ");
    printkurangdarinol(tm.Hour);
    lcd.print(":");
    printkurangdarinol(tm.Minute);
    lcd.print(":");
    printkurangdarinol(tm.Second);
  } else {
    if (RTC.chipPresent()) {
      lcd.print("DS1307 Terhenti!");
    } else {
      lcd.print(" DS1307 Error!");
    }
  }
  delay(9000);
}
delay(1000);
}
void printkurangdarinol(int nomor) {
  if (nomor >= 0 && nomor < 10) {
    lcd.write('0');
  }
  lcd.print(nomor);
}

```

A.1.4 Program Uji Komunikasi dengan SIM900

```
#include <SoftwareSerial.h>
SoftwareSerial SIM900A(7,8);

void setup()
{
  SIM900A.begin(9600); // Setting the baud rate of GSM Module
  Serial.begin(9600); // Setting the baud rate of Serial Monitor
  (Arduino)
  Serial.println ("SIM900A Ready");
  delay(100);
  Serial.println (" Ketik 's' untuk Mengirim SMS dan 'r' untuk
  Membaca SMS masuk");
}

void loop()
{
  if (Serial.available()>0)
  switch(Serial.read())
  {
  case 's':
  SendMessage();
  break;
  //case 'r':
  //RecieveMessage();
  //break;
  }

  if (SIM900A.available()>0)
  Serial.write(SIM900A.read());
}

void SendMessage()
{
  Serial.println ("SIM900A Mengirim SMS");
  SIM900A.println("AT+CMGF=1"); //Sets the GSM
  Module in Text Mode
  delay(1000); // Delay of 1000 milli seconds or 1 second
  Serial.println ("Set SMS Number");
}
```

```

        SIM900A.println("AT+CMGS=\"+6285708173417\"r"); //
Replace with your mobile number
delay(1000);
Serial.println ("Set SMS Content");
        SIM900A.println("Hello, I am SIM900A GSM Module");//
The SMS text you want to send
delay(100);
Serial.println ("Finish");
SIM900A.println((char)26);// ASCII code of CTRL+Z
delay(1000);
Serial.println (" ->SMS Selesai dikirim");
}

```

A.1.5 Program Database PHP MyAdmin

```

<?php
$koneksi = mysqli_connect ("localhost", "root", "", "gardu3");

$nama_perusahaan = $_POST ["nama_perusahaan"];
$statuscb = $_POST ["status_cb"];
$sensor_tegangan1 = $_POST ["tegangan1"];
$statuslbsin = $_POST ["statuslbsincominggardukonvensional"];
$sensor_tegangan2 = $_POST ["tegangan2"];
$statuslbsout = $_POST ["statuslbsoutgoinggardukonvensional"];
$ kondisi_gardu = $_POST ["kondisisekaranggardukonvensional"];
$tanggal_danwaktu = $_POST
["tanggal_danwaktugardukonvensional"];

mysqli_query($koneksi, "INSERT INTO
gardu5(nama_perusahaan,status_cb,tegangan1,statuslbsincomingga
rdukonvensional,tegangan2,statuslbsoutgoinggardukonvensional,k
ondisisekaranggardukonvensional,tanggal_danwaktugardukonvensi
onal) VALUES
('$nama_perusahaan','$statuscb','$sensor_tegangan1','$statuslbsin','
$sensor_tegangan2','$statuslbsout','$kondisi_gardu','$tanggal_danw
aktu')");
?>

```

A.1.6 Program Tampil Database di Web

```
<?php
    $koneksi = mysqli_connect ("localhost", "root","", "gardu3");
    ?>
<html>
<head>
<title>Database Industri</title>
<head>
<body>
<h3 align="center">Database Industri</h3>
<table border="1" width= "600px" align="center">
<tr>
<td>Nama perusahaan</td>
<td>Status CB</td>
<td>Sensor Tegangan 1</td>
<td>Status LBS Incoming</td>
<td>Sensor Tegangan 2</td>
<td>Status LBS Outgoing</td>
<td>Kondisi Gardu</td>
<td>Tanggal dan Waktu</td>
</tr>
<?php
$query = mysqli_query ($koneksi, "SELECT * FROM gardu5");
while ($data= mysqli_fetch_array($query)) {
    ?>
<tr>
<td><?php echo $data ['nama_perusahaan']; ?></td>
<td><?php echo $data ['status_cb']; ?></td>
<td><?php echo $data ['tegangan1']; ?></td>
<td><?php echo $data ['statuslbsincominggardukonvensional'];
?></td>
<td><?php echo $data ['tegangan2']; ?></td>
<td><?php echo $data ['statuslbsoutgoinggardukonvensional'];
?></td>
<td><?php echo $data ['kondisisekaranggardukonvensional'];
?></td>
<td><?php echo $data ['tanggaldanwaktugardukonvensional'];
?></td>
</tr>
```

```

<?php
}
?>
</table>
</body>
</html>

```

A.1.7 Program Utama

```

#include <SoftwareSerial.h>
#include <DS3231.h>
#include <Time.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11,5,4,3,2);
SoftwareSerial SIM900A(7,8);
DS3231 rtc(SDA, SCL);
Time waktu;

SoftwareSerial bluetooth(14, 15);
int data;
int data2;
int data3;
int teganganCB;
int teganganDSin;
int teganganDSout;
int tegangan0;
int tegangan00;
int tegangan20;
int tegangan200;
String a = "Pemeliharaan pada PT. Argo Jaya dapat dilakukan";
String b = "Pemeliharaan PT. Argo Jaya telah selesai dilakukan,
selamat beraktivitas kembali";
String c = "Gardu Portal PT. Argo Jaya Padam, Indikasi Terjadi
gangguan.";
String d = "Gardu Portal PT. Argo Jaya telah diperbaiki. Gardu
Portal kembali normal";
int PEBE = 16; // Ini pin PEBE
int led1 = 17; // Ini led 1
int kondisi = 1; // Ini variable untuk membaca status PEBE
const int relay = 18; // the number of the pushbutton pin

```

```

const int led_relay_Pin = 13;    // the number of the LED pin
boolean operasi=true;
int relayState = 0;           // variable for reading the pushbutton status
boolean gangguan=true;

void setup() {
// put your setup code here, to run once:
Serial.begin(9600);          // komunikasi Arduino ke Komputer
lcd.begin(16,2);
SIM900A.begin(9600);        // Setting the baud rate of GSM Module
bluetooth.begin(9600);      //Baudrate Modul Bluetooth
rtc.begin();
pinMode(led1, OUTPUT);      // Mendefinisikan led1 sebagai output
pinMode(PEBE, INPUT);       // Mendefinisikan PEBE sebagai input

//karakter simbol
uint8_t Baik[8]={0x00,0x00,0x01,0x02,0x14,0x08,0x00,0x00};
uint8_t Buruk[8]={0x00,0x11,0x0A,0x04,0x0A,0x11,0x00,0x00};

lcd.createChar(1,Baik);
lcd.createChar(2,Buruk);
lcd.setCursor(0,0);
lcd.print(" Bismillah");
lcd.setCursor(0,1);
lcd.print(" TUGAS AKHIR ");
delay(5000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" Gardu Portal");
lcd.setCursor(0,1);
lcd.print(" Konvensional");
delay(5000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" OLEH :");
lcd.setCursor(0,1);
lcd.print(" ROBY & LUTFI ");
delay(5000);

```

```

lcd.clear();
}

void SIM900power()
{
  digitalWrite(9,HIGH);
  delay(500);
  digitalWrite(9,LOW);
  return;
}

void Kirim_sms_pra()
{
  Serial.println ("SIM900A Mengirim SMS");
  SIM900A.println("AT+CMGF=1"); //Sets the GSM Module in
Text Mode
  delay(1000); // Delay of 1000 milli seconds or 1 second
  Serial.println ("Set SMS Number");
  SIM900A.println("AT+CMGS=\"+6285755267595\"\\r"); //
Replace with your mobile number
  delay(1000);
  Serial.println ("Set SMS Content");
  SIM900A.println(a);// The SMS text you want to send
  delay(100);
  Serial.println ("Finish");
  SIM900A.println((char)26);// ASCII code of CTRL+Z
  delay(1000);
  Serial.println (" ->SMS Selesai dikirim");
  operasi = false;
}

void Kirim_sms_pasca()
{
  Serial.println ("SIM900A Mengirim SMS");
  SIM900A.println("AT+CMGF=1"); //Sets the GSM Module in
Text Mode
  delay(1000); // Delay of 1000 milli seconds or 1 second
  Serial.println ("Set SMS Number");

```

```

SIM900A.println("AT+CMGS=\"+6285755267595\\r"); //
Replace with your mobile number
delay(1000);
Serial.println ("Set SMS Content");
SIM900A.println(b);// The SMS text you want to send
delay(100);
Serial.println ("Finish");
SIM900A.println((char)26);// ASCII code of CTRL+Z
delay(1000);
Serial.println (" ->SMS Selesai dikirim");
operasi = true;
}

```

```

void Kirim_sms_padam()
{
Serial.println ("SIM900A Mengirim SMS");
SIM900A.println("AT+CMGF=1"); //Sets the GSM Module in
Text Mode
delay(1000); // Delay of 1000 milli seconds or 1 second
Serial.println ("Set SMS Number");
SIM900A.println("AT+CMGS=\"+6285755267595\\r"); //
Replace with your mobile number
delay(1000);
Serial.println ("Set SMS Content");
SIM900A.println(c);// The SMS text you want to send
delay(100);
Serial.println ("Finish");
SIM900A.println((char)26);// ASCII code of CTRL+Z
delay(1000);
Serial.println (" ->SMS Selesai dikirim");
gangguan=false;
}

```

```

void Kirim_sms_selesai_perbaikan()
{
Serial.println ("SIM900A Mengirim SMS");
SIM900A.println("AT+CMGF=1"); //Sets the GSM Module in
Text Mode
delay(1000); // Delay of 1000 milli seconds or 1 second

```

```

Serial.println ("Set SMS Number");
SIM900A.println("AT+CMGS=\"+6285755267595\"\\r"); //
Replace with your mobile number
delay(1000);
Serial.println ("Set SMS Content");
SIM900A.println(d);// The SMS text you want to send
delay(100);
Serial.println ("Finish");
SIM900A.println((char)26);// ASCII code of CTRL+Z
delay(1000);
Serial.println (" ->SMS Selesai dikirim");
gangguan=true;
}

```

```

void Memadamkan_Gardu()
{
  lcd.clear();
  int data= analogRead (A0); //KonversiNilaiSensor(dari 0-
1023)menjadi tegangan (0-5V):
  float teganganCB = data/1023.0* 5.0 ;
  int tegangan0 = teganganCB*0 ;
  int tegangan20 = teganganCB/tegangancB*20000 ;
  int data2= analogRead (A4); //KonversiNilaiSensor(dari 0-
1023)menjadi tegangan (0-5V);
  float teganganDSin = data2/1023.0* 5.00 ;
  int tegangan00 = teganganDSin*0 ;
  int tegangan200 = teganganDSin/tegangancDSin*20000 ;
  int data3= analogRead (A2); //KonversiNilaiSensor(dari 0-
1023)menjadi tegangan (0-5V):
  float teganganDSout = data3/1023.0* 5.0 ;
  if (tegangancB > 1.7){
    lcd.clear();
    bluetooth.print("CB");
    bluetooth.print("|");
    bluetooth.print("CB Terbuka");
    bluetooth.print("|");
    bluetooth.print(teganganc0); bluetooth.print(" V");
    bluetooth.print("|");
    if (tegangancDSin > 1.8){

```

```

lcd.clear();
bluetooth.print("LBS Incoming Terbuka");
bluetooth.print("|");
bluetooth.print(tegangan00); bluetooth.print(" V");
bluetooth.print("|");
if (teganganDSout > 1.7){
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Pintu Dapat");
  lcd.setCursor(0,1);
  lcd.print("dibuka");
  bluetooth.print("LBS Outgoing Terbuka");
  bluetooth.print("|");
  bluetooth.print("Gardu Aman Dipelihara");
  bluetooth.print("|");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Maintenance");
  lcd.setCursor(0,1);
  lcd.print("READY!!!");
  delay(1000);
  while ( operasi == true){
    bluetooth.print(rtc.getDateStr());   bluetooth.print(" / ");
    bluetooth.print(rtc.getTimeStr());
    bluetooth.print("|");
    Kirim_sms_pra(); }
  lcd.clear();
  lcd.setCursor(0,1);
  lcd.print("CB:");lcd.write(2);
  lcd.setCursor(5,1);
  lcd.print("DSi:");
  lcd.write(2);
  lcd.setCursor(11,1);
  lcd.print("DSo:");
  lcd.write(2);
  delay(2000);
  lcd.clear();
  lcd.setCursor(0,0);

```

```

lcd.print("Pemeliharaan");
lcd.setCursor(0,1);
lcd.print("Sedang Dilakukan");
delay(5000);
}
else{
lcd.clear();
lcd.setCursor(0,0);
lcd.print(rtc.getDateStr());
lcd.setCursor(0,1);
lcd.print("Gardu Padam");
bluetooth.print("LBS Outgoing Tertutup");
bluetooth.print("|");
bluetooth.print("Gardu Bebas Bertegangan");
bluetooth.print("|");
bluetooth.print(rtc.getDateStr());    bluetooth.print(" / ");
bluetooth.print(rtc.getTimeStr());
bluetooth.print("|");
}
delay(500);
}
else{
lcd.clear();
lcd.setCursor(0,0);
lcd.print(rtc.getDateStr());
lcd.setCursor(0,1);
lcd.print("Industri Padam");
bluetooth.print("DS Incoming Tertutup");
bluetooth.print("|");
bluetooth.print(tegangan200); bluetooth.print(" V");
bluetooth.print("|");
bluetooth.print("LBS Outgoing Tertutup");
bluetooth.print("|");
bluetooth.print("Gardu Masih Bertegangan");
bluetooth.print("|");
bluetooth.print(rtc.getDateStr());    bluetooth.print(" / ");
bluetooth.print(rtc.getTimeStr());
bluetooth.print("|");
}
}

```

```

    delay(500);
  }
  else{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(rtc.getDateStr());
  lcd.setCursor(0,1);
  lcd.print("PT. Argo Jaya");
  bluetooth.print("CB");
  bluetooth.print("|");
  bluetooth.print("CB Tertutup");
  bluetooth.print("|");
  bluetooth.print(tegangan20); bluetooth.print(" V");
  bluetooth.print("|");
  bluetooth.print("LBS Incoming Tertutup");
  bluetooth.print("|");
  bluetooth.print(tegangan200); bluetooth.print(" V");
  bluetooth.print("|");
  bluetooth.print("LBS Outgoing Tertutup");
  bluetooth.print("|");
  bluetooth.print("Gardu Bertegangan");
  bluetooth.print("|");
  bluetooth.print(rtc.getDateStr());   bluetooth.print(" / ");
  bluetooth.print(rtc.getTimeStr());
  bluetooth.print("|");
  }
  delay(100);
  }

```

```

void Menyalakan_Gardu()
{
  lcd.clear();
  int data= analogRead (A0); //KonversiNilaiSensor(dari 0-
1023)menjadi tegangan (0-5V):
  float teganganCB = data/1023.0* 5.0 ;
  int tegangan0 = teganganCB*0 ;
  int tegangan20 = teganganCB/teganganCB*20000 ;
  int data2= analogRead (A4); //KonversiNilaiSensor(dari 0-
1023)menjadi tegangan (0-5V);

```

```

float teganganDSin = data2/1023.0* 5.00 ;
int tegangan00 = teganganDSin*0 ;
int tegangan200 = teganganDSin/tegangandSin*20000 ;
int data3= analogRead (A2); //KonversiNilaiSensor(dari 0-
1023)menjadi tegangan (0-5V):
float teganganDSout = data3/1023.0* 5.0 ;
if (tegangandCB > 1.7){
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(rtc.getDateStr());
    lcd.setCursor(0,1);
    lcd.print("Gardu Normal");
    bluetooth.print("CB");
    bluetooth.print("|");
    bluetooth.print("CB Terbuka");
    bluetooth.print("|");
    bluetooth.print(tegangand0); bluetooth.print(" V");
    bluetooth.print("|");
    if (tegangandSin > 1.7){
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Masukkan");
        lcd.setCursor(0,1);
        lcd.print("LBS Incoming");
        bluetooth.print("LBS Incoming Terbuka");
        bluetooth.print("|");
        bluetooth.print(tegangand00); bluetooth.print(" V");
        bluetooth.print("|");
        if (tegangandSout > 1.7){
            bluetooth.print("LBS Outgoing Terbuka");
            bluetooth.print("|");
            bluetooth.print("Gardu Dipelihara");
            bluetooth.print("|");
            bluetooth.print(rtc.getDateStr());    bluetooth.print(" / ");
            bluetooth.print(rtc.getTimeStr());
            bluetooth.print("|");
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Pengoperasian");

```

```

    lcd.setCursor(0,1);
    lcd.print("Siap Dilakukan");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("CB:");lcd.write(2);
    lcd.setCursor(5,1);
    lcd.print("DSi:");lcd.write(2);
    lcd.setCursor(11,1);
    lcd.print("DSo:");lcd.write(2);
    delay(1000);
}
else{
    bluetooth.print("LBS Outgoing Tertutup");
    bluetooth.print("|");
    bluetooth.print("Gardu Siap diberi Tegangan");
    bluetooth.print("|");
    bluetooth.print(rtc.getDateStr());    bluetooth.print(" / ");
    bluetooth.print(rtc.getTimeStr());
    bluetooth.print("|");
}
    delay(500);
}
else{
    bluetooth.print("LBS Incoming Tertutup");
    bluetooth.print("|");
    bluetooth.print(tegangan200); bluetooth.print(" V");
    bluetooth.print("|");
    bluetooth.print("LBS Outgoing Tertutup");
    bluetooth.print("|");
    bluetooth.print("Gardu Bertegangan");
    bluetooth.print("|");
    bluetooth.print(rtc.getDateStr());    bluetooth.print(" / ");
    bluetooth.print(rtc.getTimeStr());
    bluetooth.print("|");
}
    delay(500);
}
else{

```

```

lcd.clear();
lcd.setCursor(0,0);
lcd.print(rtc.getDateStr());
lcd.setCursor(0,1);
lcd.print("PT. Argo Jaya");
bluetooth.print("CB");
bluetooth.print("|");
bluetooth.print("CB Tertutup");
bluetooth.print("|");
bluetooth.print(tegangan20); bluetooth.print(" V");
bluetooth.print("|");
bluetooth.print("LBS Incoming Tertutup");
bluetooth.print("|");
bluetooth.print(tegangan200); bluetooth.print(" V");
bluetooth.print("|");
bluetooth.print("LBS Outgoing Tertutup");
bluetooth.print("|");
bluetooth.print("Gardu Normal");
bluetooth.print("|");
lcd.clear();
lcd.setCursor(0,1);
lcd.print("CB:");lcd.write(1);
lcd.setCursor(5,1);
lcd.print("DSi:");lcd.write(1);
lcd.setCursor(11,1);
lcd.print("DSo:");lcd.write(1);
delay(1000);
while (operasi == false){
  bluetooth.print(rtc.getDateStr());   bluetooth.print(" / ");
  bluetooth.print(rtc.getTimeStr());
  bluetooth.print("|");
  Kirim_sms_pasca();}
delay(2000);
}
delay(100);
}

void loop() {
  SIM900power();

```

```

kondisi = digitalRead(PEBE);
// Jika PB ditekan == HIGH, maka aktifkan pola Running LED
if (kondisi == HIGH) {
    if (operasi==true){
        operasi= false;
    } else {
        operasi= true;
    }
}

if (operasi==true){
    digitalWrite(led1, HIGH);
    relayState = digitalRead(relay);
    Memadamkan_Gardu();
    if (relayState == HIGH){
        digitalWrite(led_relay_Pin, HIGH);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Gardu Portal");
        lcd.setCursor(0,1);
        lcd.print("Terjadi Gangguan");
        while (gangguan==true){
            Kirim_sms_padam();
        }
    }
    if (relayState == LOW){
        digitalWrite(led_relay_Pin, LOW);
        while (gangguan==false){
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Gardu Portal");
            lcd.setCursor(0,1);
            lcd.print("Telah Diperbaiki");
            Kirim_sms_selesai_perbaikan();
        }
    }
}
else {
    digitalWrite(led1, LOW);

```

```
Menyalakan_Gardu();  
Serial.print("Mati");  
delay(100);  
}  
delay(5000);  
}
```

A.2. Program Block pada App Inventor

1. Program Blok untuk *Screen 1*

```
when Screen1 .BackPressed
do close application

initialize global Password to "pln123"

when tombol_masuk .Click
do if PasswordTextBox1 .Text = get global Password
then open another screen screenName "Screen2"
else call Notifier1 .ShowMessageDialog
      message "Password salah coba lagi"
      title "Error!!"
      buttonText "Retry"
```

2. Program Blok untuk *Screen 2*

```
when jadwal .Click
do open another screen screenName "Screen10"

when monitoring .Click
do open another screen screenName "Screen4"

when database .Click
do set ActivityStarter1 .DataUri to "http://localhost:8080/data96.php"
   call ActivityStarter1 .StartActivity

when Button1 .Click
do close application
```

3. Program Blok untuk Monitoring

```
when tombollanjut Click
do open another screen screenName : "Screen2"

initialize global (name) to create empty list

when pilihbluetooth BeforePicking
do set pilihbluetooth . Elements to BluetoothClient1 . AddressesAndNames

when pilihbluetooth AfterPicking
do if call BluetoothClient1 . Connect
    address pilihbluetooth . Selection
then set pilihbluetooth . Elements to BluetoothClient1 . AddressesAndNames
    if BluetoothClient1 . IsConnected
    then set hasilbluetooth . Text to Connected
        set hasilbluetooth . TextColor to green
    else set hasilbluetooth . Text to Not Connected
        set hasilbluetooth . TextColor to red

when Clock1 Timer
do if BluetoothClient1 . IsConnected and call BluetoothClient1 . BytesAvailableToReceive > 0
then set global name to split text call BluetoothClient1 . ReceiveText
    numberOfBytes call BluetoothClient1 . BytesAvailableToReceive
    at :
    if length of list list get global name >= 2
    then set status_cb . Text to select list item list get global name
        index 2
    if length of list list get global name >= 3
    then set tegangan1 . Text to select list item list get global name
        index 3
    if length of list list get global name >= 4
    then set statusbsincominggardukonvensional . Text to select list item list get global name
        index 4
    if length of list list get global name >= 5
    then set tegangan2 . Text to select list item list get global name
        index 5
    if length of list list get global name >= 6
    then set statusbsoutgoinggardukonvensional . Text to select list item list get global name
        index 6
    if length of list list get global name >= 7
    then set kondisisekaranggardukonvensional . Text to select list item list get global name
        index 7
    if length of list list get global name >= 8
    then set tanggaidanwaktugardukonvensional . Text to select list item list get global name
        index 8
```

```

when save Click
do
  if is empty nama_perusahaan . Text = true
  then call Notifier1 .ShowAlert
        notice
  else if is empty status_cb . Text = true
  then call Notifier1 .ShowAlert
        notice
  else if is empty tegangan1 . Text = true
  then call Notifier1 .ShowAlert
        notice
  else if is empty statusbsincominggardukonvensional . Text = true
  then call Notifier1 .ShowAlert
        notice
  else if is empty tegangan2 . Text = true
  then call Notifier1 .ShowAlert
        notice
  else if is empty statusbsoutgoinggardukonvensional . Text = true
  then call Notifier1 .ShowAlert
        notice
  else if is empty tanggalanwaktugardukonvensional . Text = true
  then call Notifier1 .ShowAlert
        notice
  else
    set Web1 .Url to http://localhost:8080/gardukonven1417.php
    call Web1 .PostText
      text
      join
      nama_perusahaan=
      nama_perusahaan . Text
      &status_cb=
      status_cb . Text
      &tegangan1=
      tegangan1 . Text
      &statusbsincominggardukonvensional=
      statusbsincominggardukonvensional . Text
      &tegangan2=
      tegangan2 . Text
      &statusbsoutgoinggardukonvensional=
      statusbsoutgoinggardukonvensional . Text
      &kondisisekaranggardukonvensional=
      kondisisekaranggardukonvensional . Text
      &tanggalanwaktugardukonvensional=
      tanggalanwaktugardukonvensional . Text

    set nama_perusahaan . Text to
    set status_cb . Text to
    set tegangan1 . Text to
    set statusbsincominggardukonvensional . Text to
    set tegangan2 . Text to
    set statusbsoutgoinggardukonvensional . Text to
    set kondisisekaranggardukonvensional . Text to
    set tanggalanwaktugardukonvensional . Text to

    call Notifier1 .ShowAlert
      notice Data Berhasil disimpan
  
```

LAMPIRAN B

B.1 Datasheet ATmega 2650

Features

- High Performance, Low Power AVR[®] 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-Chip 2-cycle Multiplier
- Non-volatile Program and Data Memories
 - 64K/128K/256K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 4K Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 8K Bytes Internal SRAM
 - Up to 64K Bytes Optional External Memory Space
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - Four 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM Channels
 - Six/Twelve PWM Channels with Programmable Resolution from 2 to 16 Bits (ATmega1281/2561, ATmega640/1280/2560)
 - Output Compare Modulator
 - 8/16-channel, 10-bit ADC
 - Two/Four Programmable Serial USART (ATmega1281/2561, ATmega640/1280/2560)
 - Master/Slave SPI Serial Interface
 - Byte Oriented 2-wire Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 51/86 Programmable I/O Lines (ATmega1281/2561, ATmega640/1280/2560)
 - 64-lead (ATmega1281/2561)
 - 100-lead (ATmega640/1280/2560)
 - 100-lead TQFP (64-lead TQFP Option)
- Temperature Range:
 - -40°C to 85°C Industrial
- Speed Grade:
 - ATmega1281/2561V/ATmega640/1280/2560V:
 - 0 - 4 MHz @ 1.8 - 5.5V, 0 - 8 MHz @ 2.7 - 5.5V
 - ATmega640/1280/1281/2560Q/2561:
 - 0 - 8 MHz @ 2.7 - 5.5V, 0 - 16 MHz @ 4.5 - 5.5V



8-bit AVR[®]
Microcontroller
with 256K Bytes
In-System
Programmable
Flash

ATmega1281/25
61/V
ATmega640/128
0/2560/V

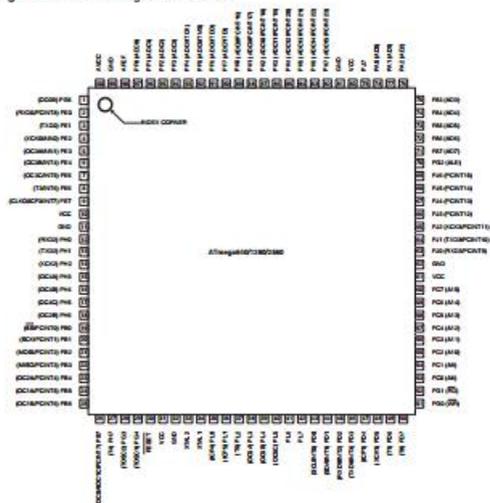
Advance
Information

25491-AVR-0305



Pin Configurations

Figure 1. Pinout ATmega640/1280/2560





Comparison Between ATmega1281/2561 and ATmega640/1280/2560

Each device in the ATmega640/1280/1281/2560/2561 family differs only in memory size and number of pins. Table 1 summarizes the different configurations for the six devices.

Table 1. Configuration Summary

Device	Flash	EEPROM	RAM	General Purpose I/O pins	16 bits resolution PWM channels	Serial USARTs	ADC Channels
ATmega640	64KB	4KB	8KB	86	12	4	16
ATmega1280	128KB	4KB	8KB	86	12	4	16
ATmega1281	128KB	4KB	8KB	54	6	2	8
ATmega2560	256KB	4KB	8KB	86	12	4	16
ATmega2561	256KB	4KB	8KB	54	6	2	8

Pin Descriptions

VCC	Digital supply voltage.
GND	Ground.
Port A (PA7..PA0)	<p>Port A is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port A pins that are externally pulled low will source current if the pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port A also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 88.</p>
Port B (PB7..PB0)	<p>Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port B has better driving capabilities than the other ports.</p> <p>Port B also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 89.</p>
Port C (PC7..PC0)	<p>Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port C also serves the functions of special features of the ATmega640/1280/1281/2560/2561 as listed on page 92.</p>
Port D (PD7..PD0)	<p>Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source</p>

ATmega640/1280/1281/2560/2561

	<p>current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port D also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 94.</p>
Port E (PE7..PE0)	<p>Port E is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port E output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port E pins that are externally pulled low will source current if the pull-up resistors are activated. The Port E pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port E also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 96.</p>
Port F (PF7..PF0)	<p>Port F serves as analog inputs to the A/D Converter.</p> <p>Port F also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port F output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port F pins that are externally pulled low will source current if the pull-up resistors are activated. The Port F pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PF7(TDI), PF5(TMS), and PF4(TCK) will be activated even if a reset occurs.</p> <p>Port F also serves the functions of the JTAG interface.</p>
Port G (PG5..PG0)	<p>Port G is a 6-bit I/O port with internal pull-up resistors (selected for each bit). The Port G output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port G pins that are externally pulled low will source current if the pull-up resistors are activated. The Port G pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port G also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 102.</p>
Port H (PH7..PH0)	<p>Port H is a 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port H output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port H pins that are externally pulled low will source current if the pull-up resistors are activated. The Port H pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port H also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 104.</p>
Port J (PJ7..PJ0)	<p>Port J is a 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port J output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port J pins that are externally pulled low will source current if the pull-up resistors are activated. The Port J pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port J also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 106.</p>
Port K (PK7..PK0)	<p>Port K serves as analog inputs to the A/D Converter.</p>





	<p>Port K is a 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port K output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port K pins that are externally pulled low will source current if the pull-up resistors are activated. The Port K pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port K also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 108.</p>
Port L (PL7..PL0)	<p>Port L is a 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port L output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port L pins that are externally pulled low will source current if the pull-up resistors are activated. The Port L pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port L also serves the functions of various special features of the ATmega640/1280/1281/2560/2561 as listed on page 110.</p>
RESET	<p>Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 23 on page 58. Shorter pulses are not guaranteed to generate a reset.</p>
XTAL1	<p>Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.</p>
XTAL2	<p>Output from the inverting Oscillator amplifier.</p>
AVCC	<p>AVCC is the supply voltage pin for Port F and the A/D Converter. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.</p>
AREF	<p>This is the analog reference pin for the A/D Converter.</p>

About Code Examples

This documentation contains simple code examples that briefly show how to use various parts of the device. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.

These code examples assume that the part specific header file is included before compilation. For I/O registers located in extended I/O map, "IN", "OUT", "SBIS", "SBIC", "CBI", and "SBI" instructions must be replaced with instructions that allow access to extended I/O. Typically "LDS" and "STS" combined with "SBR", "SBR", "SBR", and "CBR".

B.2 Datasheet RTC DS1307



DS1307 64 X 8 Serial Real Time Clock

www.dalsemi.com

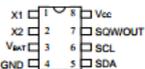
FEATURES

- Real time clock counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap year compensation valid up to 2100
- 56 byte nonvolatile RAM for data storage
- 2-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500 nA in battery backup mode with oscillator running
- Optional industrial temperature range -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Recognized by Underwriters Laboratory

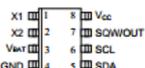
ORDERING INFORMATION

DS1307	8-Pin DIP
DS1307Z	8-Pin SOIC (150 mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

PIN ASSIGNMENT



DS1307 8-Pin DIP (300 mil)



DS1307Z 8-Pin SOIC (150 mil)

PIN DESCRIPTION

V _{CC}	- Primary Power Supply
X1, X2	- 32.768 kHz Crystal Connection
V _{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square wave/Output Driver

DESCRIPTION

The DS1307 Serial Real Time Clock is a low power, full BCD clock/calendar plus 56 bytes of nonvolatile SRAM. Address and data are transferred serially via a 2-wire bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with less than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit which detects power failures and automatically switches to the battery supply.

ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	-0.5V to +7.0V
Operating Temperature	0°C to 70°C (-40°C to 85°C for industrial)
Storage Temperature	-55°C to +125°C
Soldering Temperature	260°C for 10 seconds DIP See JPC/JEDEC Standard J-STD-020A for Surface Mount Devices

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

RECOMMENDED DC OPERATING CONDITIONS

(0°C to 70°C or -40°C to +85°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V _{CC}	4.5	5.0	5.5	V	1
Logic 1	V _{BH}	2.2		V _{CC} +0.3	V	1
Logic 0	V _{BL}	-0.3		+0.8	V	1
V _{BAT} Battery Voltage	V _{BAT}	2.0		3.5	V	1

DC ELECTRICAL CHARACTERISTICS(0°C to 70°C or -40°C to +85°C; V_{CC} = 4.5V to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage	I _{LI}			1	μA	10
I/O Leakage	I _{LO}			1	μA	11
Logic 0 Output	V _{OL}			0.4	V	2
Active Supply Current	I _{CCA}			1.5	mA	9
Standby Current	I _{CCS}			200	μA	3
Battery Current (OSC ON); SQW/OUT OFF	I _{BAT1}		300	500	nA	4
Battery Current (OSC ON); SQW/OUT ON (32 kHz)	I _{BAT2}		480	800	nA	4

AC ELECTRICAL CHARACTERISTICS(0°C to 70°C or -40°C to +85°C; $V_{CC}=4.5V$ to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	kHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			μs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			μs	5
LOW Period of SCL Clock	t_{LOW}	4.7			μs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			μs	
Set-up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			μs	
Data Hold Time	$t_{HD:DAT}$	0			μs	6, 7
Data Set-up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_r			1000	ns	
Fall Time of Both SDA and SCL Signals	t_f			300	ns	
Set-up Time for STOP Condition	$t_{SU:STO}$	4.7			μs	
Capacitive Load for each Bus Line	C_B			400	pF	8
I/O Capacitance	C_{IO}		10		pF	
Crystal Specified Load Capacitance			12.5		pF	

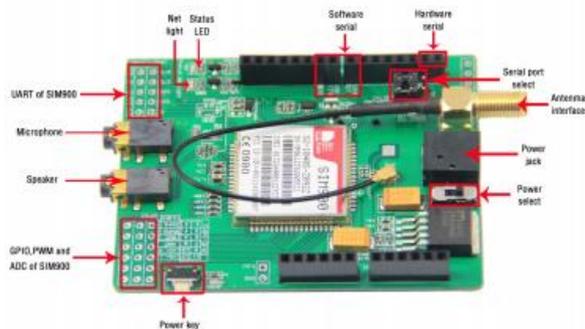
NOTES:

- All voltages are referenced to ground.
- Logic zero voltages are specified at a sink current of 5 mA at $V_{CC}=4.5V$, $V_{OL}=GND$ for capacitive loads.
- I_{CCS} specified with $V_{CC}=5.0V$ and $SDA, SCL=5.0V$.
- $V_{CL}=0V$, $V_{BAT}=3V$.
- After this period, the first clock pulse is generated.
- A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{HMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
- The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
- C_B - total capacitance of one bus line in pF.
- I_{CCA} - SCL clocking at max frequency = 100 kHz.
- SCL only.
- SDA and SQW/OUT

B.3 Datasheet Modem GSM SIM900

Electrical Specification

Item	Min	Typical	Max	Unit
Voltage	4.8	5.0	5.2	VDC
Current	/	50	450	mA
Dimension(with antenna)	110x58x19			mm
Net Weight	76±2			g



Berikut Fitur SIM900 GSM/GPS Shield :

- Quad-Band 850 / 900 / 1800 / 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
- Class 4 (2 W @ 850 / 900 MHz)
- Class 1 (1 W @ 1800 / 1900MHz)
- Control via AT commands - Standard Commands: GSM 07.07 & 07.05 | Enhanced Commands: SIMCOM AT Commands.
- Short Message Service - so that you can send small amounts of data over the network (ASCII or raw hexadecimal).
- Embedded TCP/UDP stack - allows you to upload data to a web server.
- RTC supported.
- Selectable serial port.
- Speaker and Headphone jacks
- Low power consumption - 1.5mA(sleep mode)
- Industrial Temperature Range - -40°C to +85 °C

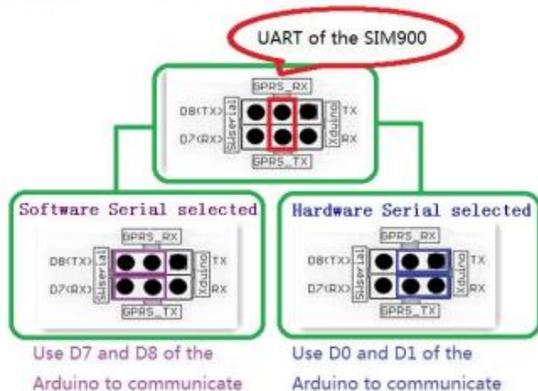
Application

- M2M (Machine 2 Machine) Applications.
- Remote control of appliances.
- Remote Weather station or a Wireless Sensor Network.
- Vehicle Tracking System with a GPS module.

LEDs(color)	Status	Description
PWR(Green)	ON	Power of the GPRS Shield is on
	OFF	Power of the GPRS Shield is off
Staus(Blue)	ON	SIM900 is on
	OFF	SIM900 is off
Netlight(Red)	64ms On/800ms Off	SIM900 has not registered to a network
	64ms On/3000ms Off	SIM900 has registered to a network
	64ms On/300ms Off	GPRS communication
	OFF	SIM900 is not running

3. Komunikasi Serial

Untuk berkomunikasi dengan Arduino, shield ini menggunakan komunikasi serial UART. Pin yang digunakan bisa port hardware Arduino biasa, hardware serial RX/TX di (D0/D1) atau menggunakan software serial di RX/TX (D8/D7). Agar tidak mengganggu komunikasi Arduino dengan PC, yang kita gunakan adalah software serial D7/D8. Untuk menggunakan software serial, kita perlu men-set jumper ada di posisi SWSERIAL seperti terlihat pada gambar.



LAMPIRAN C

C.1 Prototipe Gardu



C.2 Panel Box



C.3 Prototipe Simulasi Gangguan



RIWAYAT HIDUP PENULIS



Nama : Lutfi Kurnia Hilmy
TTL : Tuban, 1 Januari 1996
Jenis Kelamin : Perempuan
Agama : Islam
Alamat Rumah : Jalan Punggul Rejo RT
3 RW 3 Kecamatan
Rengel Kabupaten
Tuban
Telp/No. Hp : 085708173417
E-mail : hilmylutfi@gmail.com

RIWAYAT PENDIDIKAN

2000 – 2002 : TK Bhayangkari

2002 – 2008 : SDN 1 Rengel

2008 – 2011 : SMPN 1 Rengel

2011 – 2014 : SMAN 1 Tuban

2014 – 2017 : Bidang Studi Teknik Listrik, Program Teknik Otomasi,
ITS-PLN

PENGALAMAN KERJA

On The Job Training PT. PLN (Persero) Rayon Rungkut