

**DUAL SUPPLY RTU (REMOTE TELEMETRY UNIT)  
PADA SISTEM PEMANTAU BANJIR TERPUSAT  
MENGUNAKAN SMS SEBAGAI MEDIA TRANSMISI DATA**

**SKRIPSI  
JURUSAN TEKNIK ELEKTRO**

Diajukan untuk memenuhi persyaratan  
memperoleh gelar Sarjana Teknik



Disusun oleh :

**JUDIS PRIASTONO UTAMA**

**NIM. 0710630059-63**

**KEMENTERIAN PENDIDIKAN NASIONAL  
UNIVERSITAS BRAWIJAYA  
FAKULTAS TEKNIK**

**MALANG**

**2011**



**LEMBAR PERSETUJUAN**

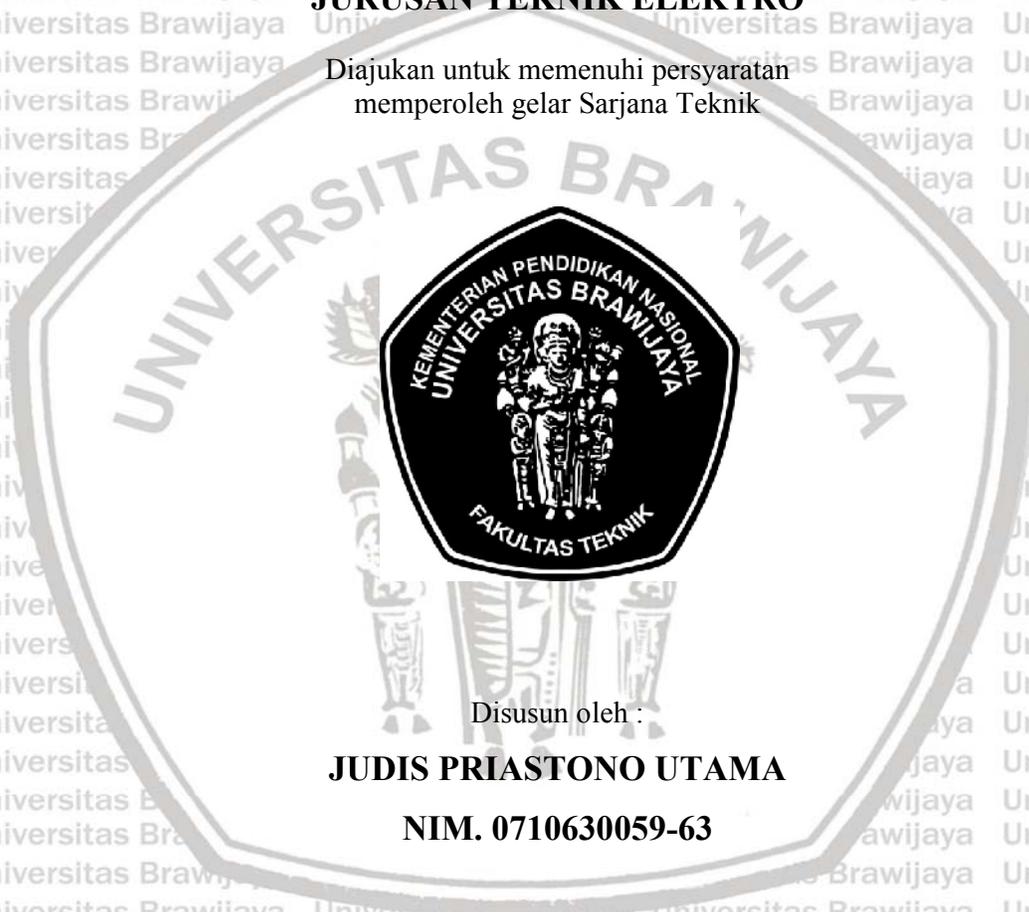
***DUAL SUPPLY RTU (REMOTE TELEMETRY UNIT)***

**PADA SISTEM PEMANTAU BANJIR TERPUSAT  
MENGUNAKAN SMS SEBAGAI MEDIA TRANSMISI DATA**

**SKRIPSI**

**JURUSAN TEKNIK ELEKTRO**

Diajukan untuk memenuhi persyaratan  
memperoleh gelar Sarjana Teknik



Disusun oleh :

**JUDIS PRIASTONO UTAMA**

**NIM. 0710630059-63**

Telah diperiksa dan disetujui oleh :

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Nurussa'adah, MT.

Panca Mudjirahardjo, ST., MT.

NIP. 19680706 199203 2 001

NIP. 19700329 200012 1 001



**LEMBAR PENGESAHAN**

***DUAL SUPPLY RTU (REMOTE TELEMETRY UNIT)***  
**PADA SISTEM PEMANTAU BANJIR TERPUSAT**  
**MENGGUNAKAN SMS SEBAGAI MEDIA TRANSMISI DATA**

**SKRIPSI**  
**JURUSAN TEKNIK ELEKTRO**

Diajukan untuk memenuhi persyaratan  
memperoleh gelar Sarjana Teknik

Disusun oleh :

**JUDIS PRIASTONO UTAMA**

**NIM. 0710630059-63**

Skripsi ini telah diuji dan dinyatakan lulus pada  
tanggal 10 Agustus 2011

DOSEN PENGUJI

**M. Rif'an, ST., MT.**

**NIP. 19710301 200012 1 001**

**Ir. Ponco Siwindarto, MS.**

**NIP. 19590304 198903 1 001**

**Ir. M. Julius St., MS.**

**NIP. 19540720 198203 1 002**

Mengetahui

Ketua Jurusan Teknik Elektro

**Rudy Yuwono, ST., M.Sc.**

**NIP. 19710615 199802 1 003**

## PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT, karena dengan rahmat, taufik dan hidayah-Nya lah skripsi ini dapat diselesaikan.

Skripsi berjudul “*Dual Supply RTU (Remote Telemetry Unit) Pada Sistem Pemantau Banjir Terpusat Menggunakan SMS Sebagai Media Transmisi Data*” ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik di Jurusan Teknik Elektro Universitas Brawijaya.

Penulis menyadari bahwa penyusunan skripsi ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, dengan ketulusan dan kerendahan hati penulis menyampaikan terima kasih kepada :

- Ibu tercinta Rini Yulianti Widyastuti, Adikku yang tersayang Rian Respati Andarini dan Aga Dia Priasmoro serta seluruh keluarga besar, atas segala nasehat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta telah banyak mendoakan kelancaran penulis hingga terselesaikannya skripsi ini,
- Bapak Rudy Yuwono, ST., M.Sc selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya dan Bapak M. Aziz Muslim, ST., MT., Ph.D selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya,
- Bapak Ir. M. Julius St, MS. selaku Ketua Kelompok Dosen Keahlian Elektronika Jurusan Teknik Elektro Universitas Brawijaya,
- Bapak Mochammad Rif'an, ST.,MT. selaku Ketua Program Studi Jurusan Teknik Elektro Universitas Brawijaya,
- Ibu Ir. Nurussa'adah, MT. dan Bapak Panca Mudjirahardjo, ST., MT. selaku Dosen Pembimbing I dan Dosen Pembimbing II, atas segala bimbingan, nasehat, pengarahan, motivasi, saran dan masukan yang telah diberikan,
- Sahabat karibku, Agung .N. Pramuditha, M. Syaiful. H, I Gusti Agung .D, dan Nourma Devita .P, atas dukungan serta semangat yang diberikan pada penulis selama pengerjaan skripsi ini,
- Teman-teman Tim KORINDO Teknik Elektro Universitas Brawijaya periode 2010 – 2011 (AL-FATIH) dan 2011 – 2012 (E-FLY63), atas kerja sama dan kekompakannya serta teman-teman tim robot yang turut memberikan banyak ilmu pengetahuan dalam bidang elektronika.



- Rekan dalam pengerjaan skripsi, Adelia Revani .S, Riky Triyunardi, Farasofi Nuriesa .R, Maulina Wulandari, Mas Pratama Duta, Mas Raden Paku, Mas Agung Setiabudi, dan Mas Eko Setiawan atas segala bantuan serta saran dan masukannya,
- Teman-teman Elkamania, CORE, teman-teman di kelembagaan, senior serta semua pihak yang tidak mungkin bagi penulis untuk mencantumkan namanya satu-persatu, terima kasih banyak atas bantuan dan dukungannya.

Pada akhirnya, penulis menyadari bahwa skripsi ini masih belum sempurna.

Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun.

Penulis berharap semoga skripsi ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi.

Malang, Agustus 2011

Penulis



## ABSTRAK

**Judis Priastono Utama**, Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya, Agustus 2011, *Dual Supply RTU (Remote Telemetry Unit) Pada Sistem Pemantau Banjir Terpusat Menggunakan SMS Sebagai Media Transmisi Data*, Dosen Pembimbing : Ir. Nurussa'adah, MT. dan Panca Mudjirahardjo, ST., MT.

Banjir merupakan bencana musiman yang paling sering melanda sebagian besar wilayah Indonesia. Indonesia merupakan negara berkembang dimana kurang lebih 75% wilayahnya terdiri atas perairan, sehingga bencana banjir dapat dengan mudah melanda beberapa wilayah di Indonesia. Kurangnya kesadaran masyarakat Indonesia untuk memelihara kelestarian sumber daya air yang melimpah, serta kurang efisiennya sistem pemantau banjir yang hingga saat ini masih digunakan menjadikan banjir sebagai bencana yang sewaktu-waktu dapat membahayakan kelangsungan hidup, harta benda, serta mata pencaharian dari masyarakat Indonesia pada umumnya.

Maka dalam skripsi ini dirancang sebuah *Dual Supply RTU (Remote Telemetry Unit)* pada Sistem Pemantau Banjir Terpusat Menggunakan SMS Sebagai Media Transmisi Data. RTU merupakan suatu peralatan yang berada di tempat yang sama dengan besaran yang diamati, dan bertugas mengirimkan data hasil pengamatan parameter banjir ke *Server*. Sedangkan *Server* adalah unit yang terpisah jauh dari RTU dan bertugas mengolah data yang diterima dari RTU, serta bertindak sebagai penyedia layanan *public access* atas data-data parameter banjir yang aktual. RTU terdiri atas dua bagian utama yaitu sistem catu daya dan sistem mikrokontroler.

RTU dapat mengakuisisi data-data parameter banjir yang meliputi ketinggian permukaan air dengan resolusi 1 cm, suhu dan kelembaban udara dengan resolusi berturut-turut 1°C dan 1%RH, data-data ini kemudian ditransmisikan ke *Server* melalui SMS (*Short Message Service*) dengan menggunakan *handphone* SIEMENS C45, dimana besarnya durasi pengiriman data ditentukan oleh ketinggian permukaan air saat itu. Semakin tinggi permukaan air, maka transmisi data akan semakin sering dilakukan tiap waktunya. RTU bekerja dengan dua catu daya, yaitu catu daya AC dari PLN dan baterai yang bekerja bergantian secara *switching* sehingga dapat mengurangi ketergantungan terhadap arus AC PLN, disamping itu RTU juga dilengkapi dengan *charger* baterai yang dapat bekerja secara otomatis.

**Kata Kunci** : Banjir, RTU (*Remote Telemetry Unit*), SMS (*Short Message Service*).

**DAFTAR ISI**

PENGANTAR .....	i
ABSTRAK .....	iii
DAFTAR GAMBAR .....	vii
DAFTAR TABEL .....	x
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Ruang Lingkup .....	3
1.4 Tujuan .....	3
1.5 Sistematika Pembahasan .....	3
<b>BAB II TINJAUAN PUSTAKA</b> .....	<b>5</b>
2.1 Banjir .....	5
2.1.1 Suhu Udara .....	6
2.1.2 Kelembaban Relatif .....	7
2.2 Konverter AC ke DC .....	7
2.3 <i>Charger</i> Baterai <i>Automatic</i> .....	8
2.4 Rangkaian <i>Switching</i> .....	9
2.5 Sensor Ultrasonik SRF02 .....	11
2.6 Sensor Suhu-Kelembaban SHT11 .....	12
2.7 Mikrokontroler ATmega8 .....	13
2.7.1 Sistem <i>Interrupt</i> .....	14
2.7.2 Komunikasi Serial .....	14
2.7.3 ADC ( <i>Analog to Digital Converter</i> ) .....	16
2.8 I2C ( <i>Inter Integrated Circuit</i> ) .....	16
2.9 <i>AT-Command</i> .....	17
2.10 PDU ( <i>Protocol Data Unit</i> ) .....	18
2.11 LCD ( <i>Liquid Crystal Display</i> ) .....	19



BAB III	METODOLOGI PENELITIAN	21
3.1	Penentuan Spesifikasi Alat	21
3.2	Perancangan dan Pembuatan Alat	22
3.2.1	Perancangan dan Pembuatan Perangkat Keras	22
3.2.2	Perancangan dan Penyusunan Perangkat Lunak	22
3.3	Pengujian Alat	22
3.3.1	Pengujian Tiap Blok	23
3.3.2	Pengujian Keseluruhan Sistem	23
BAB IV	PERANCANGAN DAN PEMBUATAN ALAT	24
4.1	Perancangan Sistem	24
4.2	Perancangan Perangkat Keras	26
4.2.1	Perancangan Konverter AC ke DC	26
4.2.2	Perancangan <i>Charger</i> Baterai <i>Automatic</i>	29
4.2.3	Perancangan Rangkaian <i>Switching</i>	33
4.2.4	Perancangan Antarmuka Sensor ke Mikrokontroler ATMega8	38
4.2.5	Perancangan Minimum Sistem Mikrokontroler ATMega8	39
4.2.6	Perancangan Rangkaian LCD ( <i>Liquid Crystal Display</i> ) 4X16 Karakter	41
4.2.7	Perancangan Antarmuka <i>Handphone</i> SIEMENS C45 ke Mikrokontroler ATMega8	42
4.2.8	Perancangan Sistem Mekanik	45
4.3	Perancangan Perangkat Lunak	46
4.3.1	Perancangan Format Data Transmisi	46
4.3.2	Perancangan Algoritma Program Utama	46
4.3.3	Perancangan Subrutin RESPONSE	50
4.3.4	Perancangan Subrutin TIMER1_OVF_ISR	50
4.3.5	Perancangan Subrutin SENSOR_READ	51
4.3.6	Perancangan Subrutin CHARGE_CONTROL	53
4.3.7	Perancangan Subrutin LCD_DISPLAY	54
4.3.8	Perancangan Subrutin TRANS_CONTROL	55
4.3.9	Perancangan Subrutin REPORT_CHECK	58

BAB V PENGUJIAN DAN ANALISIS ..... 61

5.1 Pengujian Konverter AC ke DC ..... 61

5.2 Pengujian *Charger Baterai Automatic* ..... 63

5.3 Pengujian Rangkaian *Switching* ..... 65

5.4 Pengujian Sensor Ultrasonik SRF02 ..... 68

5.5 Pengujian Sensor Suhu-Kelembaban SHT11 ..... 70

5.6 Pengujian LCD 4X16 Karakter ..... 73

5.7 Pengujian Transmisi Data Melalui SMS ..... 74

5.7.1 Pengujian Rangkaian Antarmuka SIEMENS C45 ..... 74

5.7.2 Pengujian Komunikasi Serial Dua Arah ..... 75

5.7.3 Pengujian Pengiriman SMS Melalui SIEMENS C45 ..... 78

5.8 Pengujian Sistem Secara Keseluruhan ..... 80

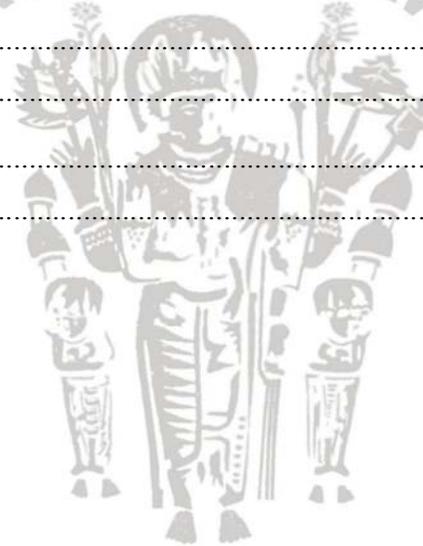
BAB VI KESIMPULAN DAN SARAN ..... 84

6.1 Kesimpulan ..... 84

6.2 Saran ..... 85

DAFTAR PUSTAKA ..... 86

LAMPIRAN ..... 88



## DAFTAR GAMBAR

Gambar 2.1 Bagan Peristiwa Bencana Banjir.....	6
Gambar 2.2 Rangkaian Konverter AC ke DC .....	7
Gambar 2.3 Rangkaian <i>Charger</i> Baterai <i>Automatic</i> .....	9
Gambar 2.4 Rangkaian <i>Switching</i> .....	10
Gambar 2.5 <i>Relay</i> DPDT MY2N .....	11
Gambar 2.6 Modul Sensor Ultrasonik SRF02.....	11
Gambar 2.7 Antarmuka SRF02 dengan Mikrokontroler.....	12
Gambar 2.8 Modul Sensor Suhu-Kelembaban SHT11 .....	12
Gambar 2.9 Antarmuka SHT11 dengan Mikrokontroler .....	13
Gambar 2.10 Konfigurasi PIN Mikrokontroler ATmega8.....	13
Gambar 2.11 Format Data Pengiriman.....	15
Gambar 2.12 Susunan Rangkaian Dengan Jalur Komunikasi I2C .....	16
Gambar 2.13 Transfer Data Pada Bus I2C .....	17
Gambar 2.14 Rangkaian <i>Interface</i> ke LCD Karakter 4X16 .....	20
Gambar 4.1 Diagram Blok Sistem <i>Dual Supply</i> RTU ( <i>Remote Telemetry Unit</i> ) .....	24
Gambar 4.2 Rangkaian Konverter AC ke DC .....	26
Gambar 4.3 Rangkaian <i>Charger</i> baterai <i>Automatic</i> .....	29
Gambar 4.4 Rangkaian <i>Charger</i> LM317.....	30
Gambar 4.5 Rangkaian Pembagi Tegangan Baterai.....	31
Gambar 4.6 Rangkaian <i>Driver Relay</i> .....	32
Gambar 4.7 Rangkaian <i>Switching</i> .....	34
Gambar 4.8 Rangkaian <i>Driver Relay Self Bias</i> .....	34
Gambar 4.9 Penyederhanaan Rangkaian <i>Driver Relay Self Bias</i> .....	35
Gambar 4.10 Rangkaian Catu Sistem Mikrokontroler-Sensor.....	37
Gambar 4.11 Rangkaian Antarmuka SRF02 ke ATmega8.....	38
Gambar 4.12 Rangkaian Antarmuka SHT11 ke ATmega8.....	39
Gambar 4.13 Konfigurasi PIN pada Minimum Sistem ATmega8 .....	40
Gambar 4.14 Rangkaian Koneksi LCD 4X16 Karakter ke Mikrokontroler ATmega8.....	42
Gambar 4.15 Konfigurasi PINOUT <i>Handphone</i> SIEMENS C45.....	43
Gambar 4.16 Rangkaian Antarmuka SIEMENS C45 ke ATmega8.....	43



Gambar 4.17 Sistem Mekanik Bagian <i>Monitoring</i> .....	45
Gambar 4.18 Sistem Mekanik Bagian <i>Processing</i> .....	45
Gambar 4.19 Algoritma Program Utama RTU .....	47
Gambar 4.20 Algoritma Subrutin RESPONSE .....	50
Gambar 4.21 Algoritma Subrutin TIMER1_OVF_ISR .....	51
Gambar 4.22 Algoritma Subrutin SENSOR_READ .....	52
Gambar 4.23 Algoritma Subrutin CHARGE_CONTROL .....	53
Gambar 4.24 Algoritma Subrutin LCD_DISPLAY .....	55
Gambar 4.25 Algoritma Subrutin TRANS_CONTROL .....	56
Gambar 4.26 Algoritma Subrutin REPORT_CHECK .....	59
Gambar 5.1 Diagram Blok Pengujian Rangkaian Konverter AC ke DC .....	61
Gambar 5.2 Tegangan Masukan dan Keluaran pada Penyearah Gelombang Penuh .....	62
Gambar 5.3 Tegangan Keluaran Pada <i>Fixed Output Regulator</i> 7812 .....	63
Gambar 5.4 Diagram Blok Pengujian <i>Charger</i> Baterai <i>Automatic</i> .....	63
Gambar 5.5 Hasil Pengujian <i>Charger</i> Baterai <i>Automatic</i> .....	64
Gambar 5.6 Grafik Tegangan Baterai Terhadap Waktu .....	65
Gambar 5.7 Grafik Arus <i>Charging</i> Terhadap Waktu .....	65
Gambar 5.8 Diagram Blok Pengujian Rangkaian <i>Switching</i> .....	65
Gambar 5.9 Hasil Pengujian Tegangan Suplai Beban dan Arus Kumparan <i>Relay</i> .....	66
Mode AC .....	66
Gambar 5.10 Hasil Pengujian Tegangan Suplai Beban dan Arus Kumparan <i>Relay</i> .....	67
Mode Baterai .....	67
Gambar 5.11 Diagram Blok Pengujian Sensor Ultrasonik SRF02 .....	68
Gambar 5.12 Hasil Pengujian Sensor Ultrasonik SRF02 .....	68
Gambar 5.13 Grafik <i>Error</i> Pengukuran Ketinggian Permukaan Air Terhadap Waktu .....	70
Gambar 5.14 Diagram Blok Pengujian Sensor Suhu-Kelembaban SHT11 .....	70
Gambar 5.15 Hasil Pengujian Sensor Suhu-Kelembaban SHT11 .....	70
Gambar 5.16 Grafik <i>Error</i> Pengukuran Suhu Udara Terhadap Waktu .....	71
Gambar 5.17 Grafik <i>Error</i> Pengukuran Kelembaban Udara Terhadap Waktu .....	72
Gambar 5.18 Diagram Blok Pengujian LCD 4X16 Karakter .....	73
Gambar 5.19 Komposisi <i>String</i> Terdefinisi pada Baris LCD 4X16 Karakter .....	73
Gambar 5.20 Hasil Pengujian LCD 4X16 Karakter .....	73
Gambar 5.21 Diagram Blok Pengujian Rangkaian Antarmuka SIEMENS C45 .....	74



Gambar 5.22 Hasil Pengujian Rangkaian Antarmuka SIEMENS C45..... 75

Gambar 5.23 Diagram Blok Pengujian Komunikasi Serial Dua Arah..... 76

Gambar 5.24 Algoritma Program Pengujian Komunikasi Serial Dua Arah..... 76

Gambar 5.25 Algoritma Subrutin USART\_RX\_ISR..... 77

Gambar 5.26 Hasil Pengujian Komunikasi Serial Dua Arah ..... 77

Gambar 5.27 Diagram Blok Pengujian Pengiriman SMS Melalui SIEMENS C45 ..... 78

Gambar 5.28 Hasil Pengujian Pengiriman SMS Melalui SIEMENS C45..... 79

Gambar 5.29 Diagram Blok Pengujian Sistem Secara Keseluruhan ..... 80

Gambar 5.30 Hasil Pengujian Sistem Secara Keseluruhan ..... 81

Gambar 5.31 Data Transmisi yang Diterima Oleh *Server*..... 82

Gambar 5.32 *Record* Data pada *Server* Selama Pengujian Keseluruhan Sistem..... 82



DAFTAR TABEL

Tabel 2.1 Alamat Vektor *Interrupt* dari ATmega8..... 14

Tabel 2.2 Rumus Menghitung *Baudrate* ..... 15

Tabel 2.3 *AT-Command* yang Berhubungan Dengan SMS..... 18

Tabel 2.4 Respon Komunikasi yang Diberikan *Handphone* ..... 18

Tabel 2.5 Cara Mengubah Format Data PDU dari Septet ke Oktet..... 19

Tabel 2.6 Cara Mengubah Format Data PDU dari Oktet ke Septet..... 19

Tabel 2.7 Tabel I/O LCD..... 20

Tabel 4.1 Fungsi Masing-Masing PIN pada *Handphone* SIEMENS C45 ..... 43

Tabel 4.2 *Range* Besaran Hasil Konversi ADC..... 54

Tabel 4.3 *Mapping* Karakter pada Data Kirim RTU..... 57

Tabel 5.1 Hasil Pengujian Rangkaian Konverter AC ke DC..... 63

Tabel 5.2 *Track* Hasil Pengujian *Charger* Baterai *Automatic* ..... 64

Tabel 5.3 Hasil Pengujian Rangkaian *Switching* ..... 67

Tabel 5.4 Hasil Pengujian Pengukuran Ketinggian Permukaan Air Sensor SRF02..... 69

Tabel 5.5 Hasil Pengujian Pengukuran Suhu Sensor SHT11 ..... 71

Tabel 5.6 Hasil Pengujian Pengukuran Kelembaban Sensor SHT11 ..... 72

Tabel 5.7 Hasil Pengujian Komunikasi Serial Dua Arah..... 78

Tabel 5.8 Hasil Pengujian Pengiriman SMS Melalui SIEMENS C45..... 80



## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Indonesia merupakan negara berkembang dimana hampir 75% wilayahnya terdiri atas perairan. Berlimpahnya kuantitas sumber daya air yang dimiliki oleh Indonesia perlu dimanfaatkan dan dijaga kelestariannya agar justru tidak membahayakan kelangsungan hidup dari masyarakat Indonesia sendiri.

Dalam 5 tahun terakhir prosentase kejadian banjir di kota-kota besar di Indonesia mengalami kenaikan, hal ini menandakan bahwa sistem pemantau banjir yang digunakan hingga saat ini belum efektif. Sistem tersebut memiliki akurasi dan efisiensi yang rendah dalam memberikan informasi parameter banjir. Sistem yang bekerja saat ini masih belum dapat diakses langsung oleh masyarakat, padahal masyarakat memiliki andil yang besar dalam pemanfaatan dan eksploitasi sumber daya air.

Untuk itu dibutuhkan suatu sistem terpusat yang memiliki kepekaan tinggi dalam mengukur parameter banjir dan dapat diakses langsung oleh masyarakat luas secara murah, sehingga dapat diperoleh data valid mengenai parameter banjir untuk dijadikan sebagai standar acuan dalam tindakan pencegahan banjir.

Sistem yang dikembangkan oleh Glorya Rakhmawaty, dkk (2009) bersifat tersebar, sehingga untuk mengetahui ketinggian air pada beberapa titik pantau diperlukan beberapa nomor *handphone*, disamping itu sistem tidak dilengkapi dengan sensor pendukung, serta modul penampil yang kurang efisien. Kemudian sistem hanya mengandalkan satu daya tunggal yang berasal dari *adaptor*, akibatnya saat arus listrik PLN putus maka secara otomatis sistem akan berhenti beroperasi.

Sistem yang dikembangkan oleh Shinta Romadhona, dkk (2008) juga bersifat tersebar dengan menggunakan satu daya tunggal yang berasal dari *adaptor*. Kemudian sistem hanya dilengkapi tiga digit *seven segment* berukuran 0,5 inci sebagai penampil data ketinggian permukaan air, akibatnya tampilan data menjadi kurang efisien.

Berdasarkan permasalahan tersebut maka dalam penelitian ini akan dirancang *Dual Supply RTU (Remote Telemetry Unit)* pada Sistem Pemantau Banjir Terpusat Menggunakan SMS Sebagai Media Transmisi Data. Perbedaan dengan skripsi sebelumnya yaitu sistem ini menggunakan sensor ultrasonik SRF02 dengan jangkauan tinggi, disamping itu sistem dilengkapi sensor suhu-kelembaban SHT11, dan LCD.

Perbedaan lainnya yaitu sistem menggunakan dua jenis catu daya saat beroperasi, saat arus listrik dari jala-jala PLN putus maka peran catu daya utama akan diambil alih oleh baterai, sebaliknya saat catu daya kembali terhubung ke jala-jala PLN maka baterai akan secara otomatis di *charge* hingga penuh.

Sensor akan merubah besaran fisik menjadi besaran elektrik. Hasil pembacaan data dari tiap sensor akan diproses oleh mikrokontroler untuk ditampilkan pada LCD dan ditransmisikan via SMS ke *Server*. Keuntungan menggunakan SMS (*Short Message Service*) sebagai media transmisi data adalah jangkauannya yang sangat luas, sehingga antara RTU dan *Server* bisa terpisah jarak yang sangat jauh.

Variasi akan parameter pemantauan banjir, penggunaan catu daya ganda, serta SMS sebagai media transmisi data diharapkan dapat meningkatkan efisiensi dan akurasi proses pemantauan banjir secara terpusat. Penggunaan SMS sebagai media transmisi data cenderung lebih murah sehingga dapat menekan biaya operasional.

Penelitian ini membahas tentang perancangan dan pembuatan *Dual Supply* RTU (*Remote Telemetry Unit*) pada Sistem Pemantau Banjir Terpusat Menggunakan SMS Sebagai Media Transmisi Data. RTU terdiri atas sistem catu daya dan sistem mikrokontroler. Sistem catu daya terdiri atas konverter AC ke DC, rangkaian *switching*, serta *charger* baterai *automatic*. Sedangkan sistem mikrokontroler terdiri atas sensor ultrasonik SRF02, sensor suhu-kelembaban SHT11, LCD (*Liquid Crystal Display*) 4X16 karakter, dan *handphone* SIEMENS C45.

## 1.2 Rumusan Masalah

Dari masalah yang telah disebutkan dalam latar belakang skripsi ini, maka rumusan masalah ditekankan pada hal-hal sebagai berikut :

- 1). Bagaimana merancang dan membuat sistem elektronika yang mampu melakukan proses akuisisi data ketinggian permukaan air, data suhu udara, serta data kelembaban udara.
- 2). Bagaimana merancang dan membuat rangkaian *switching* antara catu daya AC dari jala-jala PLN dan baterai terhadap beban, serta merancang dan membuat rangkaian *charger* baterai yang mampu bekerja secara otomatis.
- 3). Bagaimana merancang dan membuat rangkaian antarmuka serta perangkat lunak dari sistem mikrokontroler sebagai media pengolah data sekaligus media kendali utama transmisi data melalui SMS (*Short Message Service*) menggunakan *handphone* SIEMENS C45.

### 1.3 Ruang Lingkup

Perancangan sistem dalam skripsi ini terdapat permasalahan yang dibatasi oleh hal-hal sebagai berikut :

- 1). Parameter banjir yang diukur adalah ketinggian permukaan air, suhu udara, dan kelembaban udara.
- 2). Mikrokontroller yang digunakan adalah ATmega8.
- 3). *Handphone* yang digunakan adalah SIEMENS C45.
- 4). Baterai yang digunakan adalah jenis *sealed lead acid* 12 V dengan kapasitas penyimpanan 1200 mAh.
- 5). Sistem menggunakan catu daya AC 220 V dan DC 12 V (baterai).
- 6). Penelitian bersifat *open loop*, hanya sebatas memantau parameter banjir yang ada pada suatu lokasi.
- 7). Tidak membahas secara detail tentang telepon seluler yang digunakan dan mekanisme penggunaannya.
- 8). Tidak membahas tentang operator telepon seluler, jaringan telepon yang digunakan dan mekanisme pembayaran jasa operator telepon seluler.

### 1.4 Tujuan

Tujuan penyusunan skripsi ini adalah merealisasikan sebuah *Dual Supply* RTU (*Remote Telemetry Unit*) yang dapat mengamati parameter banjir antara lain ketinggian permukaan air, suhu, dan kelembaban udara serta mengirimkan data-data tersebut ke *Server* melalui layanan SMS (*Short Message Service*). Dengan demikian dapat mempermudah masyarakat untuk mengetahui status banjir dan kondisi ketinggian air, serta parameter banjir lainnya di daerah tertentu.

### 1.5 Sistematika Pembahasan

Skripsi ini terdiri dari 6 bab dengan sistematika pembahasan yang dijabarkan sebagai berikut :

#### BAB I Pendahuluan

Membahas latar belakang, rumusan masalah, batasan masalah, serta tujuan dan sistematika pembahasan.

#### BAB II Tinjauan Pustaka

Membahas teori-teori yang mendukung dalam perencanaan dan pembuatan RTU (*Remote Telemetry Unit*), yang meliputi :

banjir dan parameter-parameternya, konverter AC ke DC, rangkaian *switching*, *charger* baterai *automatic*, sensor ultrasonik SRF02, sensor suhu-kelembaban SHT11, mikrokontroler ATmega8, komunikasi I2C (*Inter Integrated Circuit*), komunikasi serial, *AT-Command*, PDU (*Protocol Data Unit*), serta LCD (*Liquid Crystal Display*).

### **BAB III Metodologi Penulisan**

Membahas metode penelitian dan perencanaan dan pengujian RTU (*Remote Telemetry Unit*).

### **BAB IV Perancangan dan Pembuatan Alat**

Membahas tentang rangkaian elektronika sistem, algoritma perangkat lunak, dan sistem mekanik pada RTU (*Remote Telemetry Unit*).

### **BAB V Pengujian dan Analisis**

Membahas hasil pengujian sistem untuk masing-masing blok penyusun RTU (*Remote Telemetry Unit*) dan pengujian sistem secara keseluruhan dari RTU yang telah direalisasikan.

### **Bab VI Kesimpulan dan Saran**

Membahas kesimpulan dari perancangan ini dan saran-saran yang diperlukan untuk melakukan pengembangan aplikasi selanjutnya.

## BAB II TINJAUAN PUSTAKA

Sistem ini merupakan pengembangan dari sistem yang dirancang oleh Glorya Rakhmawaty, dkk (2009) dan Shinta Romadhona, dkk (2008). Sistem terdahulu memiliki kekurangan yaitu sifat operasinya tersebar, tidak dilengkapi sensor pendukung, serta penampil kurang efisien, dan hanya mengandalkan catu daya AC.

Perbedaan dengan sistem sebelumnya yaitu, sistem ini menggunakan sensor ultrasonik SRF02, sistem dilengkapi sensor suhu-kelembaban SHT11, dan LCD, sistem beroperasi menggunakan dua jenis catu daya. Beberapa teori pendukung yang perlu dibahas dalam pembuatan sistem RTU (*Remote Telemetry Unit*) ini meliputi :

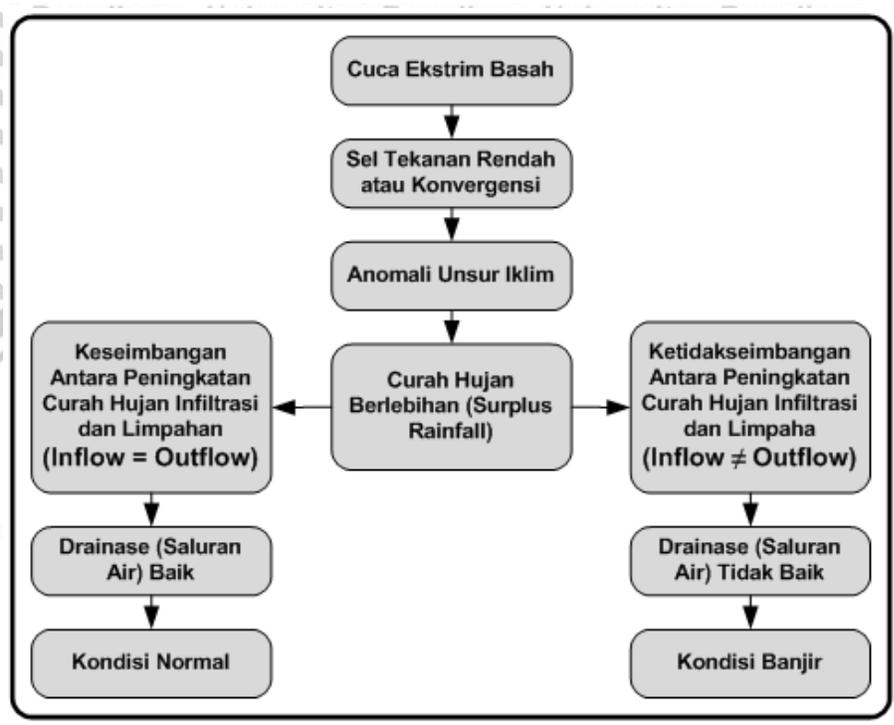
- 1). Banjir.
- 2). Konverter AC ke DC.
- 3). *Charger Baterai Automatic*.
- 4). Rangkaian *Switching*.
- 5). Sensor Ultrasonik SRF02.
- 6). Sensor Suhu-Kelembaban SHT11.
- 7). Mikrokontroler ATMega8.
- 8). *I2C (Inter Integrated Circuit)*.
- 9). *AT-Command*.
- 10). PDU (*Protocol Data Unit*).
- 11). LCD (*Liquid Crystal Display*) 4X16 Karakter.

### 2.1 Banjir

Banjir merupakan aliran air sungai yang tingginya melebihi muka air normal sehingga air meluber keluar palung sungai dan menyebabkan adanya genangan di sisi sungai. Aliran air yang meluber tersebut meninggi dan melimpahi muka tanah yang biasanya tidak dilewati aliran air (Bayong, 2007:67).

Banjir pada umumnya disebabkan oleh curah hujan yang tinggi di atas normal sehingga sistem pengaliran air yang terdiri atas sungai dan anak sungai alamiah serta sistem *drainase* dan kanal penampung banjir buatan yang ada tak mampu menampung akumulasi air hujan sehingga ketinggian muka air meningkat dan air dengan cepat meluap serta menutup daratan hingga ketinggian tertentu (Khairullah, 2009:1).

Faktor meteorologis utama yang menyebabkan bencana banjir adalah hujan torensial (*torrential rain*) yaitu hujan deras yang sangat lebat dan cenderung menyebabkan banjir. Banjir memiliki hubungan yang erat dengan hujan. Hujan sendiri dapat dideteksi dengan mengamati parameter perubahan suhu ( $^{\circ}\text{C}$ ) dan kelembaban relatif (%RH) dalam jangka waktu tertentu (Khairullah, 2009:1). Bagan peristiwa bencana banjir ditunjukkan dalam Gambar 2.1.



Gambar 2.1 Bagan Peristiwa Bencana Banjir  
 Sumber: Bayong, 2003: 70

Banjir tergolong dalam kategori bencana dikarenakan efeknya yang merugikan masyarakat. Banjir dapat menimbulkan kerusakan lingkungan hidup berupa rusaknya area pertanian, pemukiman, sarana dan prasarana penduduk, sulitnya mendapatkan air bersih, serta timbulnya penyakit-penyakit.

**2.1.1 Suhu Udara**

Suhu adalah derajat panas sebuah bahan yang berhubungan dengan energi kinetik rata-rata atom-atom atau molekul bahan. Terdapat empat skala untuk menyatakan nilai suhu yaitu skala *reamur*, *celcius*, *fahrenheit*, dan *calvin*.

Untuk menyatakan skala atau nilai suhu secara umum dalam bidang meteorologi digunakan skala suhu *celsius*, meskipun skala *fahrenheit* masih sering digunakan di kawasan Amerika dan Eropa (Smith, 2006: 214).



Rentang perubahan suhu udara pada saat musim penghujan lebih lebar dibandingkan saat musim kemarau. Untuk daerah tropis seperti Indonesia, rentang perubahan suhu udara sekitar 20°C – 30°C, sedangkan rentang perubahan suhu udara saat musim kemarau sekitar 17°C – 35°C (BMKG, 2009:1).

**2.1.2 Kelembaban Relatif**

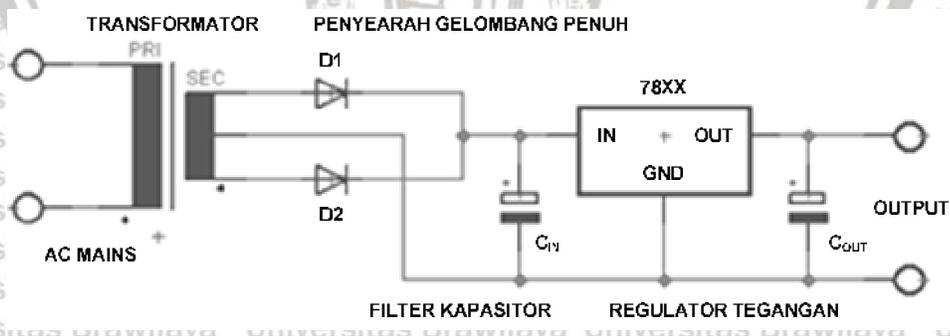
Kelembaban adalah banyak sedikitnya uap air di udara. Kelembaban dapat dinyatakan dalam beberapa cara yaitu kelembaban absolut (*absolute humidity*), dan kelembaban relatif (*relative humidity*) (Waryono, 1987: 58).

Nilai kelembaban udara saat hujan akan berubah. Untuk wilayah tropis seperti Indonesia, saat musim penghujan rentang perubahan kelembaban relatifnya akan berkisar antara 56%RH – 96%RH, sedangkan saat musim kemarau nilai kelembaban relatifnya berkisar antara 40%RH – 55%RH (BMKG, 2009:1).

**2.2 Konverter AC ke DC**

Perangkat elektronika mestinya dicatu oleh suplai arus searah DC (*Direct Current*) yang stabil. Sumber daya listrik yang besar dan paling mudah ditemui dalam kehidupan sehari-hari adalah sumber AC (*Alternating Current*) dari PLN.

Maka diperlukan suatu konverter yang dapat melakukan konversi arus AC dari jala-jala PLN menjadi arus DC yang dapat langsung digunakan oleh perangkat elektronika. Rangkaian konverter AC ke DC ditunjukkan dalam Gambar 2.2.



Gambar 2.2 Rangkaian Konverter AC ke DC  
Sumber: <http://www.sound.westhost.com/>

Konverter AC ke DC terdiri atas sebuah transformator, penyearah gelombang penuh, filter kapasitor, dan sebuah IC regulator. Transformator *stepdown* digunakan untuk menurunkan tegangan AC 220 V ke level tertentu. Rangkaian penyearah gelombang penuh berfungsi untuk merubah arus AC menjadi arus DC. Kapasitor digunakan untuk mengurangi besarnya tegangan *ripple* pada keluaran DC.



Regulator 78XX menstabilisasi tegangan keluaran sesuai dengan tipe yang digunakan. Tegangan *ripple* muncul akibat tegangan DC keluaran penyearah gelombang penuh masih memiliki unsur sinusoida pada siklus positifnya dengan frekuensi dua kali frekuensi jala-jala PLN yaitu 100 Hz. Tegangan *ripple* ( $V_r$ ) yang dapat terjadi dinyatakan dalam Persamaan (2-1).

$$V_r = \frac{I}{f \cdot C_{IN}} \quad (2-1)$$

Arus maksimum yang mengalir ke beban diinginkan sebesar  $I$ , sedangkan tegangan *ripple* maksimum yang diinginkan keluar dari penyearah sebesar  $V_r$ . Maka nilai kapasitansi yang dibutuhkan dinyatakan dalam Persamaan (2-2).

$$C_{IN} = \frac{I}{V_r \cdot f} \quad (2-2)$$

Kapasitansi sebesar  $C$  mampu meredam tegangan *ripple* sebesar  $V_r$  saat beban penuh.

### 2.3 *Charger Baterai Automatic*

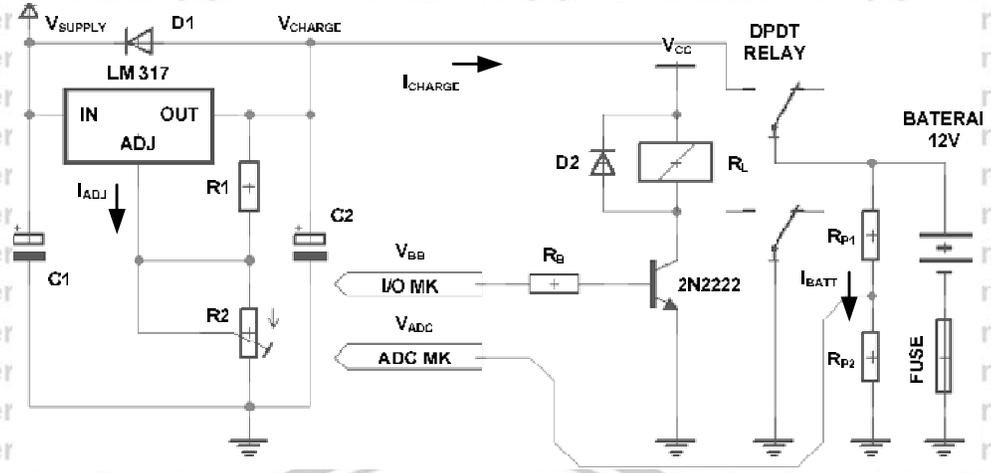
*Charger* baterai berfungsi untuk mengisi kembali energi pada baterai yang sebelumnya telah berkurang saat digunakan. Sebuah *charger* baterai yang baik harus mempunyai spesifikasi antara lain :

- 1). *Auto Shut-Off*, merupakan kemampuan *charger* untuk menghentikan arus pengisian ke sebuah baterai jika kapasitas baterai sudah terisi penuh.
- 2). Tegangan *output* konstan, dan arus *output* cukup untuk mengisi beberapa baterai sekaligus secara paralel.
- 3). *Short Circuit Protection*, jika terjadi hubungan singkat yang diakibatkan oleh baterai atau rangkaian *charger* tidak akan merusak bagian yang lain.

Berikut adalah parameter-parameter yang perlu diperhatikan dalam merancang sebuah rangkaian *charger* :

- 1). Tegangan pengisian harus dijaga 2,3 V – 2,4 V per 2 V *cell*. Untuk baterai *sealed lead acid* 12 V (2V terdiri 6 *cell*) harus di *charge* dengan tegangan sebesar  $6 \times 2,3 \text{ V} = 13,8 \text{ V}$ .
- 2). Memiliki pengamanan terhadap kondisi *over current*.

*Auto Shut-Off charger* dapat diwujudkan dengan menggunakan efek kontak dari *relay*. Jika koil *relay* terkontak dengan baterai, maka proses *charging* berlangsung. Sebaliknya saat koil *relay* tidak terkontak dengan baterai, maka baterai dalam kondisi *floating*. Rangkaian *automatic charger* baterai 12 V ditunjukkan dalam Gambar 2.3.



Gambar 2.3 Rangkaian Charger Baterai Automatic.  
 Sumber: <http://www.batteryuniversity.com/>

Nilai-nilai resistor  $R_{P1}$  dan  $R_{P2}$  yang terhubung ke PIN ADC pada mikrokontroler dapat dianalisis berdasarkan Persamaan (2-3) dan Persamaan (2-4).

$$(R_{P1} + R_{P2}) = \frac{V_{Batt}}{I_{Batt}} \tag{2-3}$$

$$V_{ADC} = \frac{R_{P2} \cdot V_{Batt}}{R_{P1} + R_{P2}} \tag{2-4}$$

Nilai resistor  $R_B$  yang terhubung pada PIN I/O mikrokontroler dapat dianalisis berdasarkan nilai resistansi kumparan *relay* ( $R_L$ ), penguatan arus transistor, dan tegangan basis transistor yang dinyatakan dalam Persamaan (2-5).

$$R_B = \frac{hfe \cdot R_L \cdot (V_{BB} - 0,7)}{V_{CC} - V_{CE}} \tag{2-5}$$

Tegangan keluaran *charger* dapat dianalisis berdasarkan nilai resistor  $R1$  dan  $R2$  serta arus *adjustable* dari LM317 ( $I_{ADJ} = 50 \mu A$ ) dalam Persamaan (2-6).

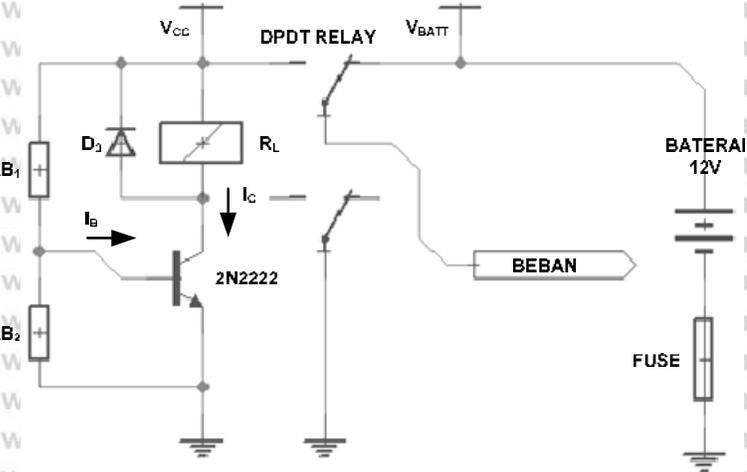
$$V_{Charge} = 1,25 \cdot \left( 1 + \left( \frac{R2}{R1} \right) \right) + (I_{ADJ} \cdot R2) \tag{2-6}$$

### 2.4 Rangkaian Switching

Rangkaian *switching* berfungsi untuk mengalihkan peran catu daya dari jala-jala PLN ke catu daya baterai ketika arus listrik dari PLN putus atau sebaliknya ketika arus AC PLN kembali mengalir. Rangkaian *switching* tersusun atas *driver relay* yang terdiri atas beberapa komponen, diantaranya sebuah transistor dengan karakteristik *switching*, dioda *flyback*, dan sebuah *relay* yang dikondisikan dalam keadaan *normally ON*.

Rangkaian *switching* ditunjukkan dalam Gambar 2.4.





Gambar 2.4 Rangkaian *Switching*  
 Sumber: <http://www.electro-tech-online.com/>

Rangkaian dalam Gambar 2.4 dianalisis dengan menggunakan sumber tegangan pengganti ( $V_{TH}$ ) yang seri dengan tahanan penggantinya ( $R_{TH}$ ) dan terhubung pada kaki basis transistor,  $V_{CC}$  merupakan sumber tegangan utama pada rangkaian. Analisa rangkaian dalam Gambar 2.4 dinyatakan dalam Persamaan (2-7) dan Persamaan (2-8).

$$R_{TH} = R_{B1} // R_{B2} = \frac{R_{B1} \cdot R_{B2}}{R_{B1} + R_{B2}} \quad (2-7)$$

$$V_{TH} = \frac{R_{B2} \cdot V_{CC}}{R_{B1} + R_{B2}} \quad (2-8)$$

Besarnya resistor pada kaki basis transistor  $R_{B1}$  dan  $R_{B2}$  dapat ditentukan berdasarkan besarnya nilai resistansi kumparan *relay* ( $R_L$ ) dan arus kolektor transistor seperti dinyatakan dalam Persamaan (2-9), Persamaan (2-10), dan Persamaan (2-11).

$$I_C = \frac{V_{CC} - V_{CE}}{R_L} \quad (2-9)$$

$$I_B = \frac{I_C}{\beta} = \frac{V_{CC} - V_{CE}}{R_L \cdot \beta} \quad (2-10)$$

$$V_{TH} = I_B \cdot R_{TH} + V_{BE}$$

$$\frac{R_{B2} \cdot V_{CC}}{R_{B1} + R_{B2}} = \frac{(V_{CC} - V_{CE}) \cdot R_{B1} \cdot R_{B2}}{R_L \cdot \beta \cdot (R_{B1} + R_{B2})} + V_{BE} \quad (2-11)$$

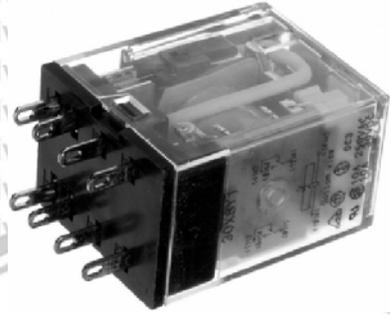
Besarnya nilai  $R_{B1}$  dan  $R_{B2}$  ditentukan dengan melakukan permisalan terhadap nilai resistansi  $R_{B1}$  atau  $R_{B2}$  melalui Persamaan (2-11) berdasarkan parameter-parameter nilai  $V_{CC}$ , nilai  $V_{CE}$ , nilai  $V_{BE}$ , dan  $\beta$  dari transistor yang digunakan.



Kondisi *default* dari koneksi koil *relay* terhadap baterai adalah *normally closed*.

Ketika terhubung arus AC PLN, koil *relay* dalam kondisi *open* terhadap baterai dan *closed* terhadap rangkaian catu daya AC. Bentuk fisik *relay* DPDT (*Double Pole*

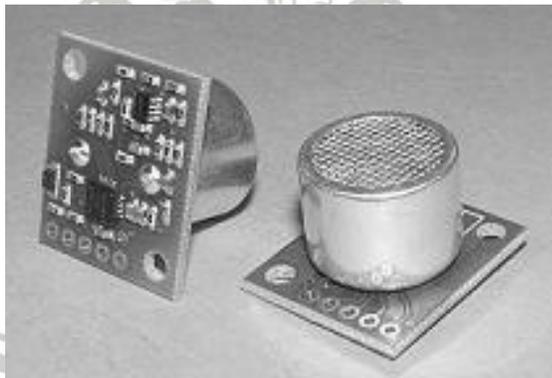
*Double Throw*) MY2N ditunjukkan dalam Gambar 2.5.



Gambar 2.5 *Relay* DPDT MY2N  
Sumber: Omron, 2006:1

**2.5 Sensor Ultrasonik SRF02**

SRF02 adalah modul sensor jarak yang memiliki dua mode komunikasi yaitu mode serial dan mode I2C (*Inter Integrated Circuit*). SRF02 hanya mempunyai satu transduser ultrasonik yang berfungsi sebagai *transmitter* sekaligus *receiver* dengan output digital. Bentuk fisik modul sensor SRF02 ditunjukkan dalam Gambar 2.6.

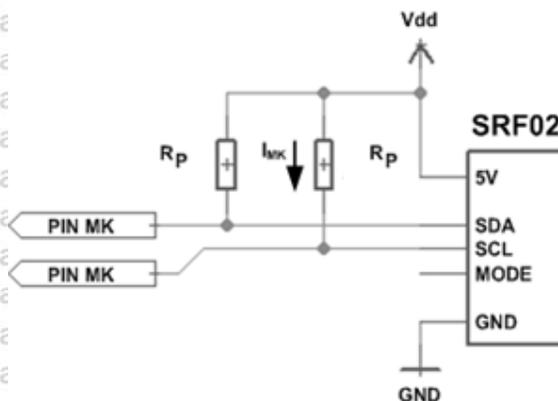


Gambar 2.6 Modul Sensor Ultrasonik SRF02  
Sumber: Devantech, 2008: 1

SRF02 merupakan modul sensor ultrasonik yang memiliki *range* pengukuran jarak 15 cm – 600 cm. *Output* dari sensor ini dapat dipilih dalam  $\mu$ s, mm, atau inci.

Sensor SRF02 bekerja pada catu daya DC 5 V dengan konsumsi arus 4 mA. Apabila SRF02 dikomunikasikan secara I2C maka PIN *Mode* diambangkan, jalur SDA dan SCL harus dihubungkan ke resistor *pull-up* sebelum akhirnya terhubung ke masing-masing PIN pada mikrokontroler, seperti ditunjukkan dalam Gambar 2.7.





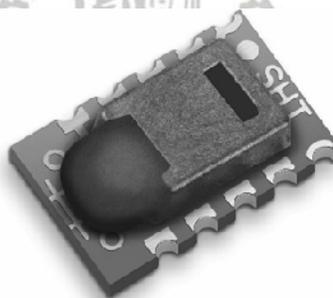
Gambar 2.7 Antarmuka SRF02 dengan Mikrokontroler  
 Sumber: Devantech, 2008: 3

Nilai resistor *pull-up* ditentukan berdasarkan tegangan output minimum ( $V_{OH}$  minimum) serta arus maksimum yang masuk ke mikrokontroler. Nilai resistor *pull-up* dapat ditentukan berdasarkan Persamaan (2-12).

$$R_p = \frac{V_{dd}}{I_{MK}} \quad (2-12)$$

## 2.6 Sensor Suhu-Kelembaban SHT11

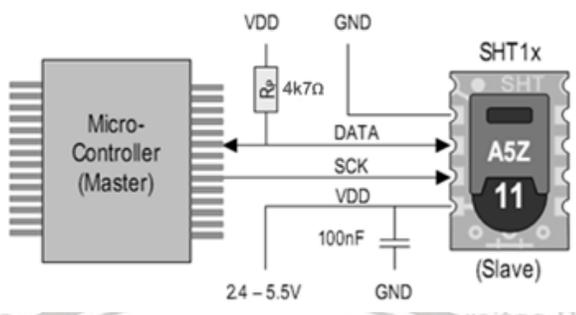
SHT11 ialah modul sensor digital untuk temperatur sekaligus kelembaban pertama di dunia yang diklaim oleh pabrik pembuatnya, SENSIRION CORP. Sensor ini telah dikalibrasikan secara standar oleh pabrik pembuatnya dan telah mempunyai keluaran berupa data suhu dan kelembaban secara digital. Bentuk fisik modul sensor suhu-kelembaban SHT11 ditunjukkan dalam Gambar 2.8.



Gambar 2.8 Modul Sensor Suhu-Kelembaban SHT11  
 Sumber: Sensirion, 2008: 1

Sensor suhu-kelembaban SHT11 mempunyai kisaran pengukuran suhu  $-40^{\circ}\text{C}$  hingga  $120^{\circ}\text{C}$  dan kisaran pengukuran kelembaban mulai  $0\%\text{RH}$  –  $100\%\text{RH}$ . SHT11 memiliki akurasi pengukuran suhu  $\pm 0.4^{\circ}\text{C}$ , sedangkan akurasi pengukuran RH relatif  $\pm 3\%\text{RH}$ . Konsumsi daya dari sensor ini sangatlah kecil.

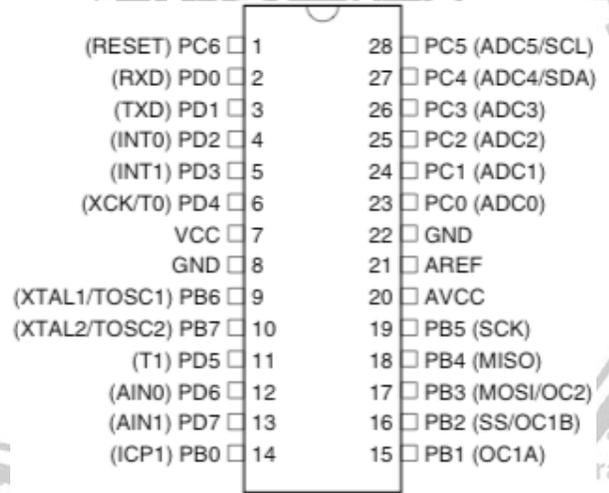
Sensor SHT11 bekerja melalui *2-wire interface*, maksudnya adalah hanya membutuhkan dua jalur sebagai perantara komunikasi datanya dengan mikrokontroller. Antarmuka SHT11 dengan mikrokontroller ditunjukkan dalam Gambar 2.9.



Gambar 2.9 Antarmuka SHT11 dengan Mikrokontroler  
Sumber: Sensirion, 2008: 5

### 2.7. Mikrokontroler ATmega8

Mikrokontroler ATmega8 merupakan IC yang berorientasi pada kontrol 8 bit yang dapat diprogram ulang dengan daya rendah. Konfigurasi PIN mikrokontroler ATmega8 ditunjukkan dalam Gambar 2.10.



Gambar 2.10 Konfigurasi PIN Mikrokontroler ATmega8  
Sumber: Atmel, 2006:2

Mikrokontroler ATmega8 mempunyai karakteristik utama sebagai berikut :

- 1). Mikrokontroler 8 bit dengan performansi tinggi dan daya rendah.
- 2). Sistem *self-programable flash* 8 kByte.
- 3). EEPROM sebesar 512 Byte, dan 1 kByte SRAM internal.
- 4). Memiliki 23 saluran I/O dan 32 *general purpose register*.
- 5). Dua *timer/counter* 8 bit dengan *prescaler* terpisah, satu mode pembanding.



- 6). Satu *timer/counter* 16 bit dengan *prescaler* terpisah, mode pembandingan (*compare mode*) dan perekam (*capture*).
- 7). Internal dan eksternal *interrupt*.
- 8). Enam PIN ADC dengan resolusi konversi sebesar 10 bit.
- 9). USART (*Universal Synchronous and Asynchronous Receiver Transmitter*).

**2.7.1 Sistem Interrupt**

Mikrokontroler ATmega8 memiliki 19 alamat vektor *interrupt*. Nomor urut dari vektor *interrupt* menyatakan prioritas dari *interrupt* tersebut. Alamat vektor *interrupt* dari mikrokontroler ATmega8 ditunjukkan dalam Tabel 2.1.

Tabel 2.1 Alamat Vektor *Interrupt* dari ATmega8

Vector No.	Program Address	Source	Interrupt Definition
1	0x0000	RESET	External PIN, Power-on Reset, Brown-out Reset, and Watchdog Reset
2	0x0001	INT0	External Interrupt Request 0
3	0x0002	INT1	External Interrupt Request 1
4	0x0003	TIMER2 COMP	Timer/Counter2 Compare Match
5	0x0004	TIMER2 OVF	Timer/Counter2 Overflow
6	0x0005	TIMER1 CAPT	Timer/Counter1 Capture Event
7	0x0006	TIMER1 COMPA	Timer/Counter1 Compare Match A
8	0x0007	TIMER1 COMPB	Timer/Counter1 Compare Match B
9	0x0008	TIMER1 OVF	Timer/Counter1 Overflow
10	0x0009	TIMER0 OVF	Timer/Counter0 Overflow
11	0x000A	SPI, STC	Serial Transfer Complete
12	0x000B	USART, RXC	USART, Rx Complete
13	0x000C	USART, UDRE	USART Data Register Empty
14	0x000D	USART, TXC	USART, Tx Complete
15	0x000E	ADC	ADC Conversion Complete
16	0x000F	EE_RDY	EEPROM Ready
17	0x0010	ANA_COMP	Analog Comparator
18	0x0011	TWI	Two-wire Serial Interface
19	0x0012	SPM_RDY	Store Program Memory Ready

Sumber: Atmel, 2006:46

**2.7.2 Komunikasi Serial**

Mikrokontroler ATmega8 dilengkapi dengan fasilitas komunikasi serial USART dengan fitur sebagai berikut :

- 1). Dapat dioperasikan pada mode komunikasi sinkron dan asinkron.
- 2). Komunikasi *full-duplex* dengan register serial untuk penerima dan pengirim data yang disertai dengan *interrupt system*.
- 3). Mempunyai resolusi tinggi untuk pembangkit *baudrate*.



- 4). Pengiriman data terdiri atas 5, 6, 7, 8, dan 9 bit dan 1 atau 2 bit *stop*.
- 5). Paritas genap atau ganjil dan didukung dengan pengecekan bit paritas oleh perangkat keras (*hardware*).
- 6). Pendeteksi pengiriman kelebihan data.
- 7). Pendeteksi kesalahan pada format data yang dikirim.
- 8). Memiliki 3 layanan *interrupt* yaitu TX *Complete*, TX *Data Empty*, dan RX *Complete*.

Untuk menghitung *baudrate* dari komunikasi serial digunakan rumus seperti yang ditunjukkan dalam Tabel 2.2.

Tabel 2.2 Rumus Menghitung *Baudrate*

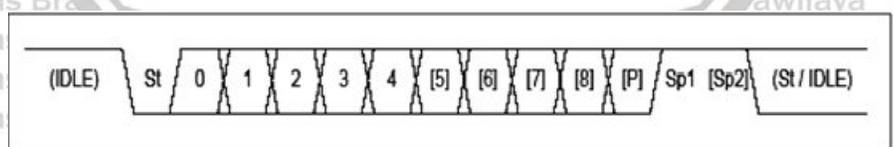
Operating Mode	Equation for Calculating Baud Rate	Equation for Calculating UBRR Value
Asynchronous Normal mode (U2X = 0)	$BAUD = \frac{f_{osc}}{16(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = 1)	$BAUD = \frac{f_{osc}}{8(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{osc}}{2(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{2BAUD} - 1$

Sumber : Atmel, 2006:136

Dengan :

- $f_{osc}$  = Frekuensi *clock* sistem osilator.
- UBRR = Register *baudrate* yang terdiri atas UBRRH dan UBRRL.
- BAUD = *Baudrate* dalam *bit per second* (bps).

Pada pengiriman data secara serial menggunakan mikrokontroler ATmega8 menggunakan format seperti yang ditunjukkan dalam Gambar 2.11.



Gambar 2.11 Format Data Pengiriman  
Sumber: Atmel, 2006:137

Dengan :

- St = Bit *start* selalu berlogika rendah.
- (n) = Banyaknya data yang dikirim (0-8).
- P = Bit paritas (ganjil atau genap).



Sp = Bit stop berlogika tinggi (bit stop bisa berjumlah 1 atau 2).

IDLE = Tanpa data yang ditransfer pada RX dan TX, IDLE selalu berlogika tinggi.

### 2.7.3 ADC (Analog to Digital Converter)

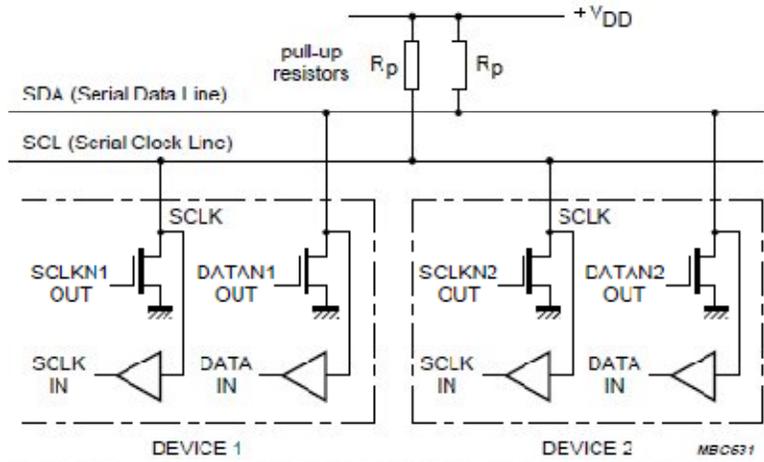
Mikrokontroler ATmega8 memiliki fasilitas ADC (Analog to Digital Converter) yang terintegrasi dengan pilihan resolusi 8 bit atau 10 bit. ADC terhubung pada analog multiplexer 8 channel yang untuk memproses enam buah sinyal analog. Ke enam channel tersebut terhubung pada PIN.C0 hingga PIN.C5 pada mikrokontroler ATmega8. ADC pada mikrokontroler ATmega8 memiliki beberapa fitur :

- 1). Resolusi 8 bit dan 10 bit.
- 2). Batas tegangan input ADC 0 V hingga Vcc.
- 3). Memiliki tegangan referensi internal 2,56 V dan 1,1 V.
- 4). Memiliki mode *free running* atau *single conversion*.
- 5). *Interrupt* ketika konversi selesai.

### 2.8 I2C (Inter Integrated Circuit)

Secara fisik, jalur komunikasi I2C terhadap mikrokontroler terdiri atas dua buah jalur. Kedua buah jalur tersebut umumnya dikenal sebagai jalur SDA (Serial Data Line) dan jalur SCL (Serial Clock Line).

SDA (Serial Data Line) adalah jalur data baik masuk maupun keluar, sedangkan SCL (Serial Clock Line) adalah jalur clock. I2C dapat dihubungkan pada lebih dari satu slave dimana tiap slave harus memiliki nilai alamat yang berbeda-beda. Bentuk umum susunan perangkat yang berkomunikasi dengan I2C ditunjukkan dalam Gambar 2.12.

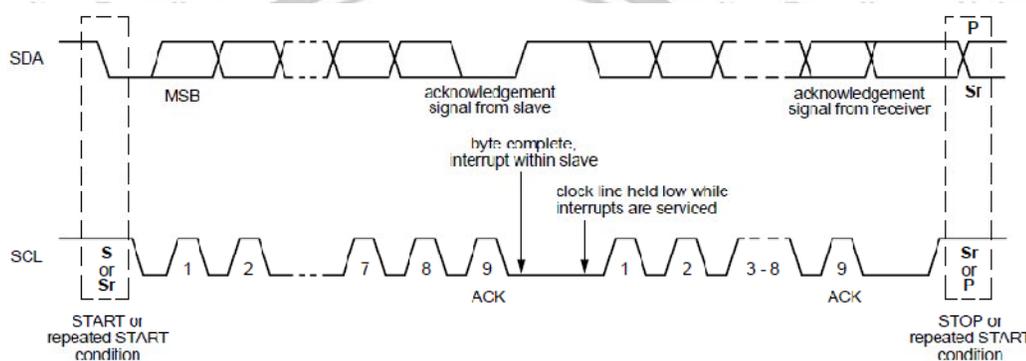


Gambar 2.12 Susunan Rangkaian Dengan Jalur Komunikasi I2C  
Sumber: Philips Semiconductor, 2000: 8



Urutan komunikasi dalam protokol I2C adalah sebagai berikut. Pertama, mikrokontroler memberi kondisi *start*. Kemudian mikrokontroler mengirimkan sinyal berupa nilai alamat dari perangkat yang akan diakses ke jalur I2C. Tiap perangkat *slave* akan membandingkan alamat yang dikirim dengan alamat yang dimiliki.

Jika perbandingan tidak sesuai, maka *slave* menunggu hingga *master* mengirim sinyal *stop*. Namun jika sesuai, maka *slave* mengirim sinyal *acknowledge*. Saat mikrokontroler menerima *acknowledge*, maka mikrokontroler dapat mengirim data serta melakukan operasi baca tulis. Setelah selesai, mikrokontroler mengirim sinyal *stop*. Representasi transfer data pada bus I2C ditunjukkan dalam Gambar 2.13.



Gambar 2.13 Transfer Data Pada Bus I2C  
Sumber: Philips Semiconductor, 2000: 10

## 2.9 AT-Command

*Handphone* merupakan piranti elektronik yang terdiri atas *hardware* dan *software*. Untuk melakukan komunikasi dengan devais lain setiap *handphone* dilengkapi dengan PIN RX dan PIN TX. Bahasa pemrograman yang digunakan untuk memprogram setiap *handphone* dikenal dengan nama *AT-Command*.

*AT-Command* untuk tiap devais berbeda namun pada intinya sama. Fungsi *AT-Command* adalah memberikan perintah pada *handphone* untuk melakukan tindakan berdasarkan pada perintah dari *AT-Command* yang diberikan misalnya mengirimkan SMS, menampilkan SMS, menghapus SMS, melakukan *dial*, dll. Daftar perintah *AT-Command* yang berhubungan dengan SMS (*Short Message Service*) pada *handphone* SIEMENS C45 ditunjukkan dalam Tabel 2.3.

Perintah *AT-Command* pada *handphone* SIEMENS C45 didahului dengan sintak ‘AT’ diikuti dengan ‘+’ setelah itu perintah diberikan. Selain menerima perintah melalui *AT-Command*, *handphone* akan memberikan *acknowledgement* pada setiap komunikasi data. Daftar respon yang diberikan *handphone* ditunjukkan dalam Tabel 2.4.

Tabel 2.3 AT-Command yang Berhubungan Dengan SMS

07.05 Commands	Function	Type of Command
AT+CMGC	Send an SMS command	Message sending and writing
AT+CMGD	Delete an SMS in the SMS memory	Message sending and writing
AT+CMGF	SMS format	General configuration
AT+CMGL	List SMS	Message receiving and reading
AT+CMGR	Read ing an SMS	Message receiving and reading
AT+CMGS	Send an SMS	Message sending and writing
AT+CMGW	Write an SMS to the SMS memory	Message sending and writing
AT+CMSS	Send an SMS from the SMS memory	Message sending and writing
AT+CNMA	Acknowledgement of a short message directly <i>output</i>	Message receiving and reading
AT+CNMI	Display new incoming SMS	Message receiving and reading
AT+CPMS	Prefferd SMS message storage	General configuration
AT+CSCA	Address of the SMS service <i>center</i>	Message configuration
AT+CSCB	Select cell broadcast message	Message configuration
AT+CSMS	Selection of message service	General configuration

Sumber: SIEMENS, 2001: 9

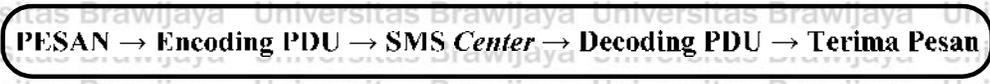
Tabel 2.4 Respon Komunikasi yang Diberikan Handphone

Response	Numeric	Meaning
OK	0	Command execute, no error
RING	2	Ring detected
NO CARRIER	3	Link not established or disconnected
ERROR	4	Invalid command or command line too long
NO DIALTONE	6	No dial tone, dialing impossible, wrong mode
BUSY	7	Remote station busy

Sumber: SIEMENS, 2001: 6

**2.10 PDU (Protocol Data Unit)**

Saat suatu SMS (*Short Message Service*) dikirim atau pun diterima, data-data SMS tersebut diubah ke dalam format heksadesimal yang dinamakan sebagai PDU (*Protocol Data Unit*) dan sudah distandarisasi langsung oleh sebuah lembaga khusus bernama ETSI. Gambaran sebuah proses pengiriman SMS dapat dilihat dibawah ini :



Pesan yang kita ketik melalui *handphone* akan dikonversi ke format PDU kemudian terkirim ke SMS Center. Dari SMS Center inilah akan dicek bilangan pulsa pengirim, nomor yang dituju dan lain-lain. Setelah itu pesan akan dikirim ke nomor yang dituju dalam bentuk PDU, kemudian dari sisi *handphone* penerima akan melakukan proses *decoding* PDU agar SMS yang dimaksud akan bisa terbaca.



Data di PDU mempunyai format data oktet yaitu format data 8 bit. Adapun karakter yang dipakai mempunyai format data 7 bit (septet). Untuk menterjemahkan karakter menjadi data PDU supaya bisa dimengerti, maka dilakukan proses konversi yaitu dari data 7 bit (format septet) ke data 8 bit (format oktet). Cara mengubah data dari format sektet ke format oktet ditunjukkan dalam Tabel 2.5. Cara mengubah data dari format oktet ke format sektet ditunjukkan dalam Tabel 2.6.

Tabel 2.5 Cara Mengubah Format Data PDU dari Septet ke Oktet

Value	h	e	l	l	o
Decimal	104	101	108	108	111
Hex	0X68	0X65	0X6C	0X6C	0X6F
Septet	1101000	1100101	1101100	1101100	1101111
8-bit	11101000	00110010	10011011	11111101	00000110
Octet	E8	32	9B	FD	06

Sumber: Atmel, 2006 : 7

Tabel 2.6 Cara Mengubah Format Data PDU dari Oktet ke Septet

Octet	E8	32	9B	FD	6
8-bit	11101000	00110010	10011011	11111101	00000110
Septet	1101000	1100101	1101100	1101100	1101111
Decimal	104	101	108	108	111
Value	h	e	l	l	o

Sumber:Atmel, 2006: 7

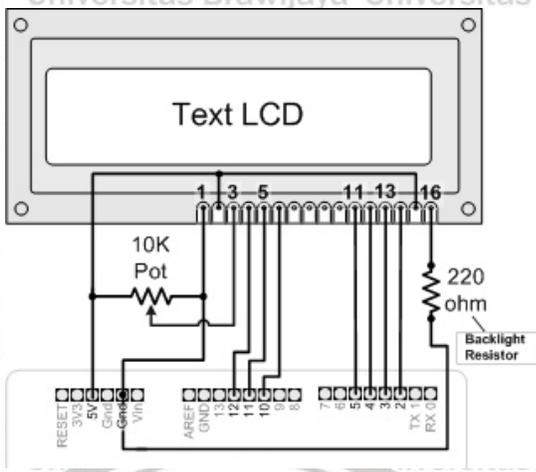
Format PDU pada saat mengirim SMS ke suatu nomor melalui SMS *Center* berbeda dengan format PDU pada saat menerima SMS dari SMS *Center*. Format PDU ini terdiri atas beberapa *header*, setiap *header* nya memiliki keterangan-keterangan yang berkaitan dengan isi pesan yang dikirim.

**2.11 LCD (Liquid Crystal Display)**

LCD (*Liquid Crystal Display*) merupakan komponen elektronika yang digunakan untuk menampilkan karakter angka, huruf, atau simbol-simbol lainnya sehingga dapat dilihat secara visual. Agar LCD karakter dapat bekerja dengan baik maka diperlukan suplai dengan sumber tegangan sebesar 5 V.

Pada perancangan, sistem ini menggunakan LCD modul M1632. LCD modul M1632 dilengkapi panel LCD dengan tingkat kontras yang cukup tinggi serta pengendali LCD CMOS yang telah terpasang dalam modul tersebut. Rangkaian *interface* ke LCD 4X16 karakter ditunjukkan dalam Gambar 2.14. Konfigurasi PIN pada LCD 4X16 karakter ditunjukkan dalam Tabel 2.7.





Gambar 2.14 Rangkaian Interface ke LCD 4X16 Karakter  
 Sumber: Manual book LCD 4X16 Character

Tabel 2.7 Tabel I/O LCD

No	Symbol	Level	Fungsi	
1	Vss		Power Supply	GND
2	Vcc			5 V
3	Vee		LCD Drive	
4	RS	H/L	H : Data Input L : Ins Input	
5	R/W	H/L	H : Read L : Write	
6	E	H	Enable Signal	
7	DB0	H/L	<b>Data Bus</b>	
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L		
12	DB5	H/L		
13	DB6	H/L		
14	DB7	H/L		
15	V+BL		Power Supply	4 V – 4,2 V
16	V-BL			GND

Sumber: Manual book LCD 4X16 Character



## BAB III METODOLOGI PENELITIAN

Metode penyelesaian masalah diperlukan untuk menyelesaikan rumusan masalah dan merealisasikan tujuan penelitian yang terdapat di bab pendahuluan.

### 3.1 Penentuan Spesifikasi Alat

Sebelum melakukan perencanaan dan perealisasiian alat, maka ditentukan spesifikasi dan parameter dari alat yang akan dibuat. Adapun spesifikasi dan parameter alat yang akan direalisasikan sebagai berikut :

- 1). Mikrokontroler ATmega8 berkecepatan *clock* 8 MHz berfungsi sebagai pengendali kinerja seluruh sensor, pengendali jalannya proses *charging* baterai, dan pengendali jalannya transmisi data melalui SMS (*Short Message Service*) dengan menggunakan *handphone* SIEMENS C45.
- 2). Jalur transmisi data yang digunakan adalah *Short Message Service* (SMS) dengan menggunakan *handphone* SIEMENS C45.
- 3). Data dikirim ke *Server* secara deterministik dengan acuan durasi pengiriman terlama setiap 30 menit dalam kondisi normal, dan durasi pengiriman tercepat setiap 60 detik, frekuensi pengiriman data meningkat seiring meningkatnya ketinggian permukaan air.
- 4). Alat dilengkapi LCD (*Liquid Crystal Display*) 4X16 karakter yang berfungsi untuk menampilkan lima jenis informasi yaitu ketinggian permukaan air, kecepatan perubahan ketinggian permukaan air, suhu dan kelembaban udara, serta level tegangan baterai.
- 5). Ketinggian air yang dapat diukur adalah 30 cm hingga 600 cm dengan resolusi pengukuran sebesar 1 cm.
- 6). Suhu udara yang dapat diukur adalah 0°C hingga 100°C dengan resolusi pengukuran sebesar 1°C.
- 7). Kelembaban udara yang dapat diukur adalah kelembaban udara relatif pada 0%RH hingga 100%RH dengan resolusi pengukuran sebesar 1%RH.
- 8). Alat memiliki kemampuan mendeteksi masuknya SMS *report*, sehingga saat SMS *report* tidak terdeteksi lebih dari 30 detik, maka data-data saat itu akan ditransmisi ulang melalui SMS (*resend*).

9). Data yang dikirim melalui SMS (*Short Message Service*) berisikan informasi kode RTU, ketinggian permukaan air, suhu dan kelembaban udara yang dinyatakan dalam format teks :

“Data\*Kode RTU\*Ketinggian Permukaan Air\*Suhu\*Kelembaban\*”

10). Alat menggunakan dua jenis catu daya yaitu AC 220 V dan DC 12 V yang bekerja secara *switching* dengan prioritas catu daya utama AC 220 V, *switching* dilakukan dengan menggunakan *relay*.

11). Alat dilengkapi *charger* baterai *automatic* untuk secara otomatis mengisi kembali energi pada baterai yang sebelumnya telah digunakan.

### 3.2 Perancangan dan Pembuatan Alat

Perancangan dan pembuatan alat meliputi perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*).

#### 3.2.1 Perancangan dan Pembuatan Perangkat Keras

Secara garis besar perancangan perangkat keras (*hardware*) dibagi dalam beberapa tahap sebagai berikut :

- 1). Penentuan spesifikasi alat.
- 2). Pembuatan blok diagram keseluruhan sistem.
- 3). Penentuan dan perhitungan komponen yang akan digunakan.
- 4). Desain *layout* papan rangkaian tercetak (*Printed Circuit Board/PCB*) menggunakan *software Eagle 5.6 Layout Editor*.
- 5). Merakit perangkat keras masing-masing blok.
- 6). Pembuatan *casing* sebagai sistem mekanik dan pelindung alat.

#### 3.2.2 Perancangan dan Penyusunan Perangkat Lunak

Penyusunan perangkat lunak (*software*) digunakan untuk mengendalikan dan mengatur kinerja dari alat ini. Desain dan parameter yang telah dirancang kemudian diterapkan kedalam mikrokontroler ATmega8 dengan menggunakan bahasa pemrograman C dan *compiler CodeVision AVR Version 1.24.8d Professional*.

### 3.3 Pengujian Alat

Pengujian alat dilakukan untuk memastikan bahwa sistem ini berjalan sesuai yang direncanakan. Pengujian alat mencakup pengujian perangkat keras dan perangkat lunak yang dilakukan baik tiap-tiap blok maupun secara keseluruhan sistem.

### 3.3.1 Pengujian Tiap Blok

Pengujian tiap-tiap blok penyusun sistem dilakukan dengan tujuan untuk menyesuaikan nilai masukan dan nilai keluaran tiap-tiap blok sesuai dengan perancangan yang telah dilakukan sebelumnya.

### 3.3.2 Pengujian Keseluruhan Sistem

Pengujian sistem secara keseluruhan dilakukan dengan tujuan untuk mengetahui unjuk kerja alat setelah perangkat keras dan perangkat lunak diintegrasikan bersamaan.



# BAB IV PERANCANGAN DAN PEMBUATAN ALAT

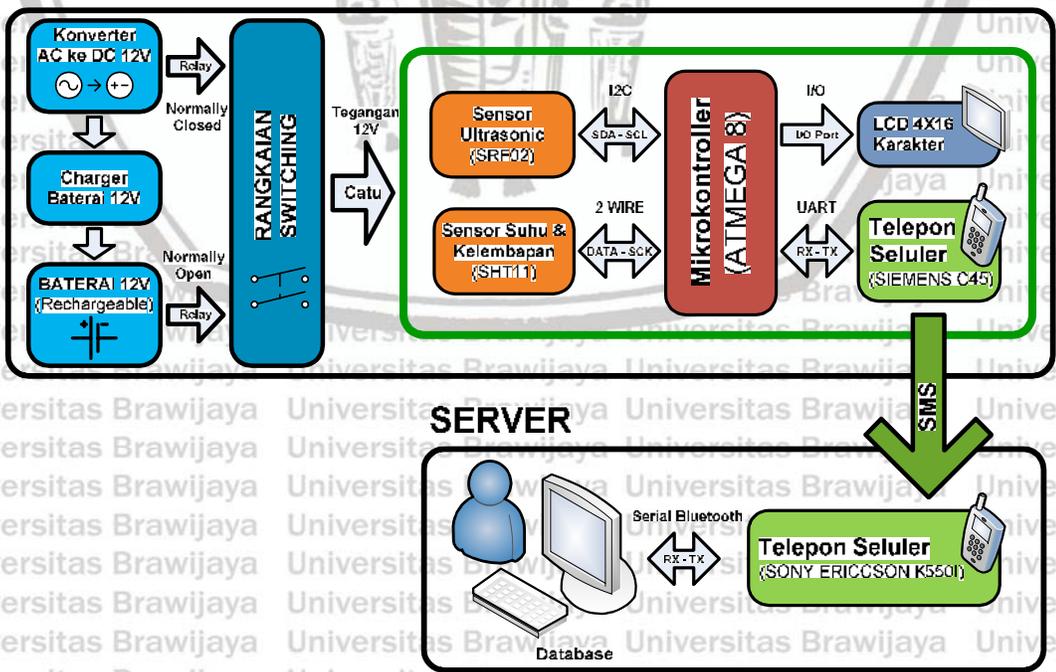
Perancangan *Dual Supply RTU (Remote Telemetry Unit)* ini dilakukan secara bertahap sehingga akan memudahkan analisis pada setiap bloknnya maupun secara keseluruhan. Perancangan ini terdiri dari beberapa hal sebagai berikut :

- 1). Perancangan sistem RTU (*Remote Telemetry Unit*).
- 2). Perancangan perangkat keras (*hardware*) terdiri dari rangkaian pada sistem catu daya, rangkaian pada sistem mikrokontroler, serta sistem mekanik.
- 3). Perancangan perangkat lunak (*software*) terdiri dari metode pengumpulan, dan penampilan data sensor, kendali proses *charging* baterai, serta kendali transmisi data melalui SMS (*Short Message Service*).

## 4.1 Perancangan Sistem

Sistem *Dual Supply RTU (Remote Telemetry Unit)* secara umum terbagi menjadi dua bagian utama yaitu sistem catu daya dan sistem mikrokontroler. Diagram blok sistem *dual supply RTU (Remote Telemetry Unit)* ditunjukkan dalam Gambar 4.1.

### RTU (Remote Telemetry Unit)



Gambar 4.1 Diagram Blok Sistem *Dual Supply RTU (Remote Telemetry Unit)*

Fungsi masing-masing blok dalam diagram blok diatas adalah sebagai berikut :

1). Blok Sistem Catu Daya

Blok sistem catu daya berfungsi memberikan suplai daya untuk operasi sistem secara keseluruhan serta menangani *switching* daya ke baterai ketika arus dari PLN putus. Blok sistem catu daya terdiri atas konverter AC ke DC, *charger* baterai *automatic*, baterai 12 V, dan rangkaian *switching*. Berikut adalah fungsi dari masing-masing bagian blok sistem catu daya :

a). Konverter AC ke DC

Berfungsi menyearahkan arus AC (*Alternating Current*) menjadi DC (*Direct Current*) sehingga dapat digunakan sebagai sumber daya utama untuk operasi sistem secara keseluruhan.

b). *Charger* Baterai *Automatic*

Berfungsi untuk mengisi ulang energi pada baterai 12 V ketika terdeteksi nilai tegangan baterai berada pada level kurang dari 80% tegangan pada kondisi idealnya (12 V).

c). Baterai 12 V

Berfungsi sebagai sumber daya cadangan yang bekerja saat arus AC dari PLN terputus sehingga sistem tidak berhenti beroperasi.

d). Rangkaian *Switching*

Berfungsi mengatur pergantian sumber daya utama sistem dari arus AC PLN ke baterai, ketika terdeteksi terputusnya arus AC dari PLN.

2). Blok Sistem Mikrokontroller

Blok sistem mikrokontroller berfungsi melakukan akusisi data sensor dan menampilkannya ke LCD, mengatur proses *charging* baterai, dan mengontrol aliran data yang ditransmisikan ke *Server* melalui SMS. Sistem mikrokontroller terdiri atas sensor ultrasonik SRF02, sensor suhu-kelembaban SHT11, mikrokontroller ATmega8, LCD 4X16 karakter, dan *handphone* SIEMENS C45. Berikut adalah fungsi dari masing-masing bagian blok sistem mikrokontroller :

a). Sensor Ultrasonik SRF02

Berfungsi mengukur dan mengkonversi besaran jarak (ketinggian permukaan air) menjadi representasi tegangan digital dengan lebar data 16bit yang diletakkan terpisah dalam dua buah register 8 bit.

b). Sensor Suhu-Kelembaban SHT11

Berfungsi mengukur dan mengkonversi besaran suhu serta kelembaban udara menjadi representasi tegangan digital dengan lebar data 16 bit.

c). Mikrokontroler ATmega8

Berfungsi mengambil dan mengakusisi data sensor, menampilkan data ke LCD, mengatur proses *charging* baterai, dan mengontrol transmisi data ke *Server* melalui SMS (*Short Message Service*).

d). LCD (*Liquid Crystal Display*) 4X16 Karakter

Berfungsi menampilkan data-data ketinggian permukaan air, kecepatan perubahan ketinggian permukaan air, suhu dan kelembaban udara, serta level tegangan baterai.

e). *Handphone* SIEMENS C45

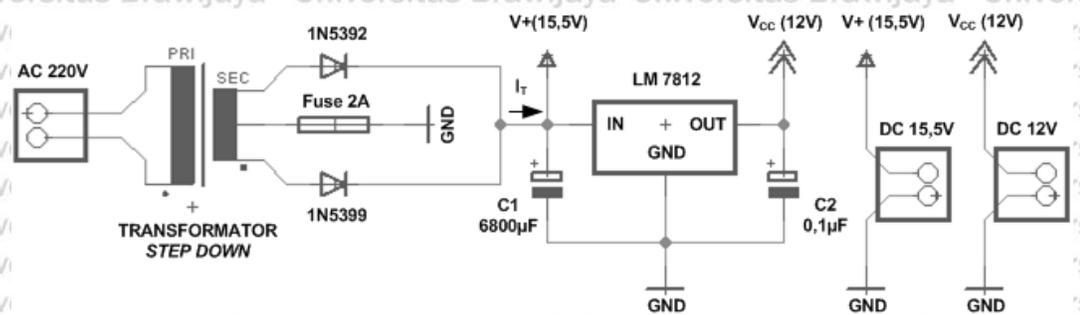
Berfungsi sebagai sarana transmisi data parameter banjir ke *Server* melalui layanan SMS (*Short Message Service*).

4.2 Perancangan Perangkat Keras

4.2.1 Perancangan Konverter AC ke DC

RTU (*Remote Telemetry Unit*) membutuhkan catu tegangan DC (*Direct Current*) yang dapat berasal dari baterai atau tegangan AC (*Alternating Current*) yang disearahkan. RTU sendiri dapat beroperasi dengan tegangan catu sebesar 12 V, sementara itu *charger* baterai membutuhkan catu tegangan DC dengan nilai tegangan diatas 13,8 V untuk dapat bekerja.

Nilai catu tegangan DC yang dapat digunakan untuk mengoperasikan RTU sekaligus *charger* baterainya dalam kondisi terhubung jala-jala PLN adalah 16 V. Tegangan DC sebesar 16 V didapat dari penyearahan tegangan AC pada kumparan sekunder transformator CT (*Center Tap*) *step down* untuk terminal 12 V. Skema rangkaian konverter AC ke DC ditunjukkan dalam Gambar 4.2.



Gambar 4.2 Rangkaian Konverter AC ke DC



Konverter AC ke DC akan menerima tegangan AC 220 V dari jala-jala PLN, kemudian level tegangan AC 220 V diturunkan menjadi AC 12 V menggunakan transformator CT *step down* dengan kapasitas arus  $I_T = 2$  A. *Fuse* 2 A dipasang pada *ground* transformator sebagai pengaman kondisi *short circuit*.

Tegangan AC 12 V disearahkan melalui penyearah gelombang penuh yang terdiri atas dua buah dioda 2 A (1N5392) yang disusun *cascade*, akibatnya frekuensi tegangan DC keluaran penyearah gelombang penuh menjadi  $2 \times$  frekuensi jala-jala PLN yaitu 100 Hz. Kapasitor C1 digunakan untuk mengurangi *ripple* pada tegangan keluaran penyearah gelombang penuh dengan perhitungan sebagai berikut :

$$V_C = \frac{Q(t)}{C} \quad \text{dan} \quad I_C = \frac{dQ}{dt}$$

$$V_C = R \cdot I_C$$

$$\frac{Q(t)}{C} = -R \cdot \frac{dQ}{dt}$$

$$\int_{Q_{\max}}^{Q(t)} \frac{1}{Q(t)} dQ = -\frac{1}{R \cdot C} \int_0^T dt$$

$$\ln Q(t) - \ln Q_{\max} = -\frac{1}{R \cdot C} \cdot (T - 0)$$

$$\ln \left( \frac{Q(t)}{Q_{\max}} \right) = -\frac{T}{R \cdot C}$$

$$\frac{Q(t)}{Q_{\max}} = e^{-\frac{T}{R \cdot C}}$$

$$Q(t) = Q_{\max} \cdot e^{-\frac{T}{R \cdot C}}$$

Muatan listrik pada kapasitor C1 akan berangsur-angsur turun secara eksponensial dari nilai muatan listrik awalnya sebesar  $Q_{\max}$ . Tegangan *discharge* pada kapasitor C1 akan berangsur-angsur turun hingga  $V_{\text{low}}$  seiring dengan berkurangnya jumlah muatan listrik pada kapasitor C1.

$$V_{C1} = \frac{Q(t)}{C_1}$$

$$V_{\text{low}} = \frac{Q_{\max}}{C_1} \cdot e^{-\frac{T}{R \cdot C_1}}$$

$$V_{\text{low}} = V_{\max} \cdot e^{-\frac{T}{R \cdot C_1}}$$

Tegangan *ripple* ( $V_{\text{ripple}}$ ) yang dinyatakan dalam puncak ke puncak (*peak to peak*) merupakan selisih antara tegangan maksimum kapasitor C1 saat *charge* ( $V_{\max}$ ) dengan tegangan minimum kapasitor C1 saat *discharge* ( $V_{\text{low}}$ ).

$$\begin{aligned}
 V_{ripple} &= V_{max} - V_{low} \\
 &= V_{max} - V_{max} \cdot e^{-\left(\frac{T}{R \cdot C_1}\right)} \\
 &= V_{max} \left(1 - e^{-\left(\frac{T}{R \cdot C_1}\right)}\right)
 \end{aligned}$$

Nilai  $V_{ripple}$  yang dinyatakan dalam bentuk eksponensial  $e^{-\left(\frac{T}{R \cdot C_1}\right)}$  dapat dituliskan secara *limit*. Nilai  $V_{ripple}$  dapat dihitung berdasarkan Persamaan (4-1).

$$e^{-\left(\frac{T}{R \cdot C_1}\right)} = \lim_{T \rightarrow 0} \left(1 - \frac{T}{R \cdot C_1}\right)$$

$$e^{-\left(\frac{T}{R \cdot C_1}\right)} \cong 1 - \frac{T}{R \cdot C_1}$$

$$V_{ripple} = V_{max} \left[1 - \left(1 - \frac{T}{R \cdot C_1}\right)\right]$$

$$V_{ripple} = \frac{V_{max} \cdot T}{R \cdot C_1} \tag{4-1}$$

$(V_{max}/ R)$  tidak lain adalah arus beban ( $I_T$ ) sehingga terlihat hubungan antara  $I_T$ , kapasitansi  $C_1$ , dan tegangan  $V_{ripple}$  seperti ditunjukkan dalam Persamaan (4-2).

$$V_{ripple} = \frac{I_T}{f \cdot C_1}$$

$$C_1 = \frac{I_T}{f \cdot V_{ripple}} \tag{4-2}$$

Pada perancangan diinginkan untuk keadaan arus beban maksimum ( $I_T = 2 \text{ A}$ ), tegangan *ripple* yang terjadi tidak lebih dari 3 Vpp. Berdasarkan Persamaan (4-2), nilai  $C_1$  yang dibutuhkan adalah :

$$\begin{aligned}
 C_1 &= \frac{2 \text{ A}}{100 \text{ Hz} \cdot 3 \text{ Vpp}} \\
 &= 6666 \text{ } \mu\text{F}
 \end{aligned}$$

Kapasitansi 6666  $\mu\text{F}$  merupakan nilai minimum, karena keterbatasan nilai kapasitor yang ada dipasaran, maka nilai  $C_1$  yang digunakan sebesar 6800  $\mu\text{F}$ .

Untuk menciptakan tegangan DC 12 V guna menjalankan RTU, maka tegangan keluaran penyearah gelombang penuh (V+) diumpankan ke *Fixed Output Regulator* LM7812. Berdasarkan *datasheet* regulator LM7812, tegangan masukan minimum regulator LM7812 agar dapat bekerja dengan baik sebesar 14,6 V.



Berdasarkan *datasheet* dioda 1N5392 dapat diketahui bahwa nilai *forward voltage* dioda 1N5392 sebesar 1,4 V. Tegangan keluaran penyearah gelombang penuh (V+) dapat dihitung berdasarkan Persamaan (4-3).

$$V_+ = V_{trafo} - V_{dioda} \tag{4-3}$$

$$V_+ = (12V \cdot \sqrt{2}) - 1,4V$$

$$= 15,5V \approx 16V$$

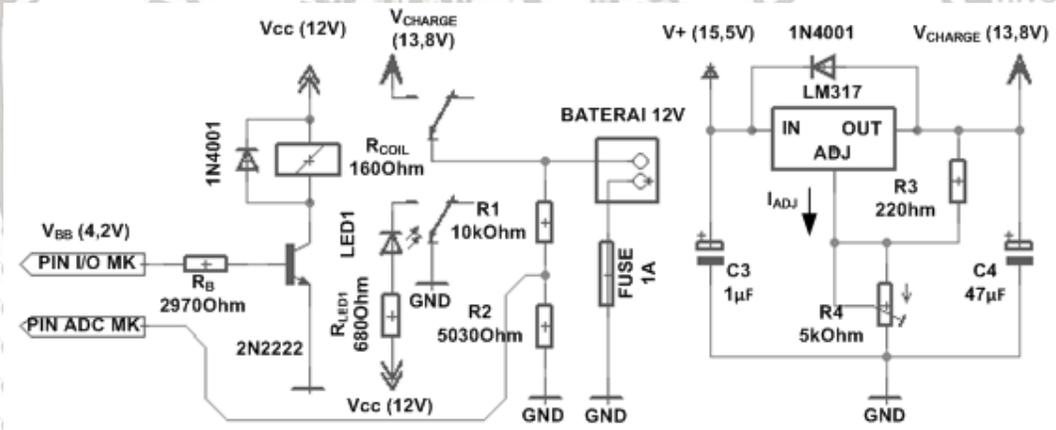
Tegangan masukan LM7812 masih dalam batas aman sebesar 15,5 V.

### 4.2.2 Perancangan Charger Baterai Automatic

*Charger* baterai berfungsi untuk mengisi kembali energi pada baterai jika level tegangannya terdeteksi dibawah 80% kondisi idealnya yaitu 9,6 V (80% dari 12 V).

Spesifikasi baterai yang digunakan adalah baterai Pb (timbal) *lead acid* 12 V, 6cells, dengan kapasitas 1200 mA<sub>H</sub> (habis dalam waktu 1 jam untuk konsumsi arus 1200 mA).

Skema rangkaian *charger* baterai *automatic* ditunjukkan dalam Gambar 4.3.

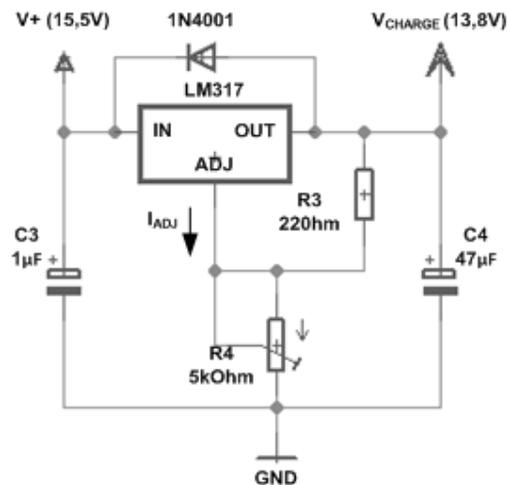


Gambar 4.3 Rangkaian *Charger* baterai *Automatic*

Rangkaian *charger* baterai *automatic* dalam Gambar 4.3 terdiri atas tiga bagian yaitu rangkaian *charger* LM317, *driver relay*, dan pembagi tegangan baterai. Rangkaian *charger* LM317 berfungsi untuk memberikan suplai tegangan ke baterai saat terjadi *charging*. *Driver relay* berfungsi untuk mengaktifkan *relay* berdasarkan sinyal yang diberikan oleh PORT mikrokontroler. Pembagi tegangan baterai berfungsi untuk membawa level tegangan baterai (0 – 15 V) ke level tegangan TTL (0 – 5 V).

Tegangan pengisian *charger* dijaga 2,3 V – 2,4 V per 2 V *cell*, maka *charging* baterai 12 V (2 V terdiri 6 *cell*) dilakukan dengan tegangan  $6 \times 2,3 V = 13,8 V$ . Tegangan sebesar 13,8 V diperoleh melalui *adjustable positive output regulator* LM317. Skema rangkaian *charger* LM317 ditunjukkan dalam Gambar 4.4.





Gambar 4.4 Rangkaian Charger LM317

Tegangan masukan LM317 diambil dari keluaran penyearah gelombang penuh ( $V^+$ ) yang besarnya 15,5 V. Berdasarkan *datasheet* regulator LM317, nilai  $I_{ADJ}$  LM317 adalah 50  $\mu A$ , nilai R3 ditentukan sebesar 220  $\Omega$ . Nilai R4 dapat dicari berdasarkan hubungan tegangan keluaran LM317 dengan R3 dan R4 dalam Persamaan (4-4).

$$V_{CHARGE} = 1,25 V \cdot \left(1 + \frac{R4}{R3}\right) + I_{ADJ} \cdot R4 \quad (4-4)$$

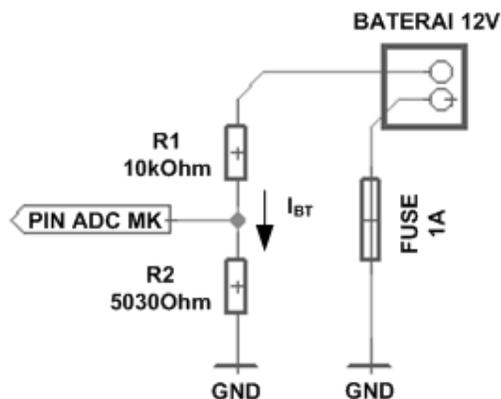
$$13,8V = 1,25 V \cdot \left(1 + \frac{R4}{220\Omega}\right) + 50 \mu A \cdot R4$$

$$R4 = \frac{13,8V - 1,25V}{\left(\frac{1,25V}{220\Omega} + 50 \mu A\right)} = 2189 \Omega$$

Karena R4 yang akan dipasang merupakan variabel resistor, maka nilai R4 harus lebih besar dari nilai idealnya 2189  $\Omega$ , untuk itu nilai variabel resistor R4 yang dipasang sebesar 5 k $\Omega$  (dikondisikan pada nilai resistansi 2189  $\Omega$ ).

Proses *charging* dikendalikan oleh mikrokontroler berdasarkan tegangan analog yang diterima oleh PIN ADC nya. *Charging* dilakukan jika tegangan baterai dibawah 80% dari kondisi idealnya, yaitu sebesar 9,6 V.

Tegangan yang dapat diterima oleh PIN ADC mikrokontroler adalah 0-5V, sedangkan tegangan baterai idealnya 0 – 12 V. Maka diperlukan rangkaian pembagi tegangan untuk membawa tegangan baterai ke level tegangan 0 – 5 V. Rangkaian pembagi tegangan dirancang untuk dapat bekerja pada level tegangan baterai 0 – 15 V. Skema rangkaian pembagi tegangan baterai ditunjukkan dalam Gambar 4.5.



Gambar 4.5 Rangkaian Pembagi Tegangan Baterai

Tegangan yang masuk ke PIN ADC mikrokontroler dapat dihitung berdasarkan hubungan bagi tegangan dalam Persamaan (4-5).

$$V_{ADC} = \frac{R2}{R1 + R2} \cdot V_{BATT} \quad (4-5)$$

$$5V = \frac{R2}{R1 + R2} \cdot 15V$$

Nilai arus maksimal yang diinginkan melewati rangkaian pembagi tegangan baterai sebesar 1 mA, nilai arus tersebut dapat dihitung berdasarkan Persamaan (4-6).

$$I_{BT} = \frac{V_{BATT}}{R1 + R2} \quad (4-6)$$

$$1mA = \frac{15V}{R1 + R2}$$

$$R1 + R2 = 15k\Omega$$

Substitusi nilai (R1+R2) ke Persamaan (4-5) akan didapatkan nilai R1 dan R2 sebesar :

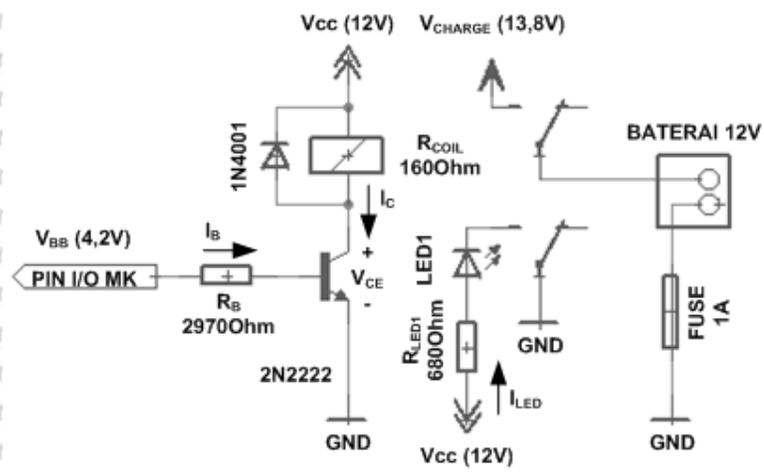
$$5V = \frac{R2}{15k\Omega} \cdot 15V$$

$$R2 = 5k\Omega \approx 5030\Omega \text{ sedangkan } R1 = 10k\Omega$$

Nilai R1 yang dipasang sebesar 10 kΩ, sedangkan nilai R2 terpasang sebesar 5030 Ω.

Mikrokontroler akan mengeluarkan tegangan sekitar 5 V pada PORT I/O yang dituju saat proses *charging* berlangsung, tegangan tersebut akan memicu *driver relay* untuk memindahkan koil *relay* sehingga menghubungkan  $V_{CHARGE}$  dengan baterai.

*Relay* yang digunakan dalam perancangan merupakan jenis *relay DPDT (Double Pole Double Throw)* keluaran OMRON dengan tipe MY2N yang memiliki dua buah koil paralel terhadap *common*. *Relay* ini memiliki resistansi koil sebesar 160 Ω untuk tegangan kerja 12 V. Skema rangkaian *driver relay* ditunjukkan dalam Gambar 4.6.



Gambar 4.6 Rangkaian Driver Relay

Tegangan dari PORT mikrokontroler sebesar ±5 V tidak cukup untuk mengaktifkan *relay*, maka digunakan transistor 2N2222 dengan karakteristik *fast switching* sebagai penjematan antara tegangan keluaran mikrokontroler dengan *relay*.

Berdasarkan *datasheet*, transistor 2N2222 memiliki spesifikasi sebagai berikut :

$I_C \text{ max}$	= 800 mA
VCE Saturasi	= 400 mV
VBE Saturasi	= 1,3 V
Hfe	= 75
(Untuk test condition $I_C = 10 \text{ mA}$ )	

Nilai arus kolektor ( $I_C$ ) yang melalui transistor 2N2222 saat kondisi saturasi dapat dihitung berdasarkan Persamaan (4-7).

$$V_{CC} = I_C \cdot R_{COIL} + V_{CE} \tag{4-7}$$

$$12V = I_C \cdot 160 \Omega + 0,4V$$

$$I_C = \frac{12V - 0,4V}{160 \Omega}$$

$$= 72,5 \text{ mA}$$

Arus kolektor ( $I_C$ ) yang mengalir pada kaki kolektor transistor 2N2222 sebesar 72,5 mA. Nilai arus basis ( $I_B$ ) yang mengalir pada kaki basis transistor 2N2222 dapat dihitung berdasarkan Persamaan (4-8).

$$I_B = \frac{I_C}{h_{fe}} \tag{4-8}$$

$$I_B = \frac{72,5 \text{ mA}}{75}$$

$$= 0,96 \text{ mA}$$



Nilai arus basis ( $I_B$ ) yang mengalir pada transistor 2N2222 sebesar 0,96 mA.

Tegangan *high level* ( $V_{OH}$ ) minimum yang mampu dikeluarkan oleh mikrokontroler sebesar 4,2 V. Nilai RB dapat dihitung berdasarkan hubungan antara tegangan bias maju basis transistor ( $V_{BB}$ ), arus basis ( $I_B$ ), dan  $V_{BE}$  dalam Persamaan (4-9).

$$V_{BB} = I_B \cdot RB + V_{BE} \quad (4-9)$$

$$4,2 V = 0,96 mA \cdot RB + 1,3 V$$

$$RB = \frac{4,2 V - 1,3 V}{0,96 mA}$$

$$= 3020 \Omega \approx 2970 \Omega$$

Nilai RB sebesar 2970  $\Omega$  merupakan seri antara resistor 2,7 k $\Omega$  dan 270  $\Omega$ .

Untuk mengindikasikan terjadinya proses *charging* digunakan LED (*Light Emitting Diode*), *forward voltage* sebuah LED sebesar 1,8 V sedangkan arus maksimum yang melewati LED ( $I_{LED}$ ) sebesar 20 mA. Nilai  $R_{LED1}$  yang harus dipasang seri terhadap LED1 dapat dihitung berdasarkan Persamaan (4-10).

$$V_{CC} = I_{LED} \cdot R_{LED} + V_{LED} \quad (4-10)$$

$$12 V = 20 mA \cdot R_{LED1} + 1,8 V$$

$$R_{LED1} = \frac{12 V - 1,8 V}{20 mA}$$

$$= 510 \Omega$$

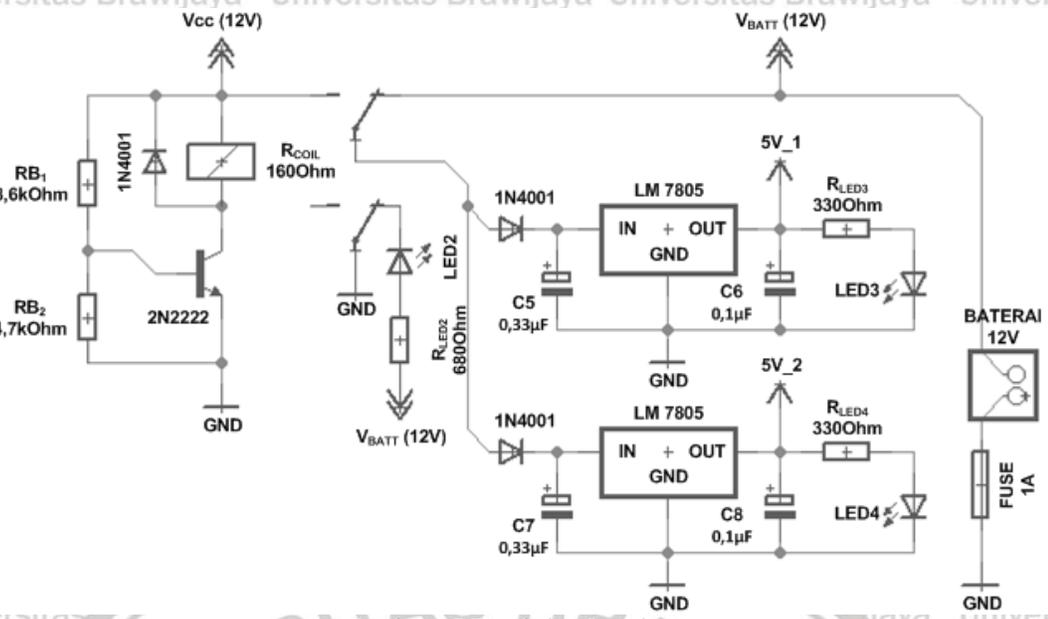
$R_{LED1}$  sebesar 510  $\Omega$  merupakan nilai minimum agar LED1 tidak putus, maka dari itu digunakan nilai  $R_{LED1}$  sebesar 680  $\Omega$ .

#### 4.2.3 Perancangan Rangkaian *Switching*

Rangkaian *switching* dimaksudkan untuk memindahkan peran sumber daya utama dari jala-jala PLN ke baterai saat arus AC PLN putus, dan sebaliknya ketika arus AC PLN kembali mengalir maka jala-jala PLN akan kembali berperan sebagai sumber daya utama disamping baterai sebagai sumber daya cadangan. Skema rangkaian *switching* ditunjukkan dalam Gambar 4.7.

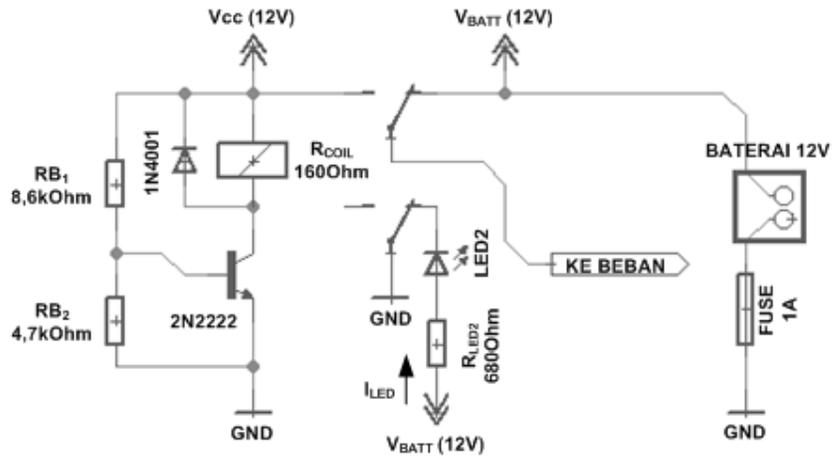
Rangkaian *switching* dalam Gambar 4.7 terdiri atas dua bagian yaitu rangkaian *driver relay self bias* serta rangkaian catu sistem mikrokontroler-sensor. Rangkaian *driver relay self bias* berfungsi untuk memindahkan koil *relay* guna merubah peran sumber daya utama dari jala-jala PLN ke baterai ataupun sebaliknya.

Rangkaian catu sistem mikrokontroler-sensor berfungsi untuk meregulasi tegangan yang diberikan oleh sumber daya utama (AC PLN / Baterai) sebesar 12 V ke level tegangan yang dapat diterima oleh sistem mikrokontroler dan sensor sebesar 5 V.



Gambar 4.7 Rangkaian Switching

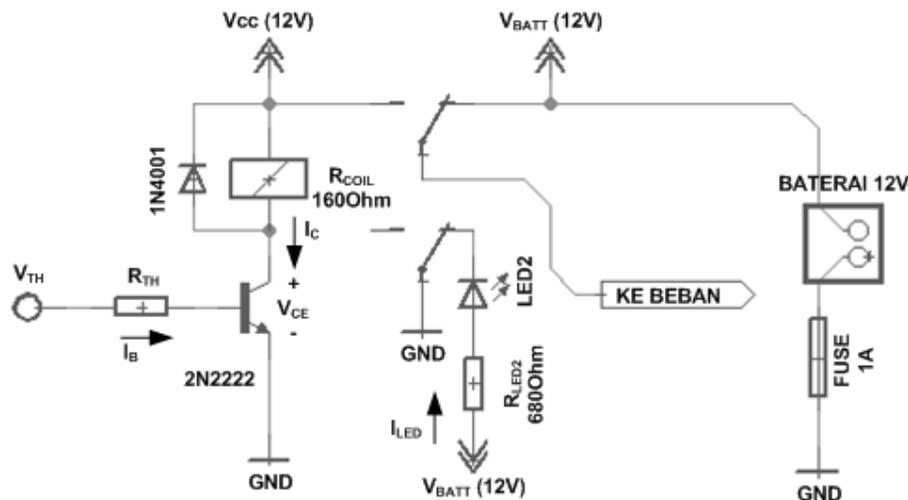
Rangkaian *driver relay self bias* memindahkan koil *relay* (*normally ON*) saat sistem terhubung arus AC dari jala-jala PLN, akibatnya beban akan menerima tegangan sumber sebesar 12 V dari keluaran rangkaian konverter AC ke DC. Saat arus AC PLN putus maka koil *relay* kembali pada posisi awalnya (*OFF*), akibatnya beban akan menerima tegangan sumber dari baterai dengan nilai yang sama yaitu 12 V . Skema rangkaian *driver relay self bias* ditunjukkan dalam Gambar 4.8.



Gambar 4.8 Rangkaian Driver Relay Self Bias

Guna mempermudah analisa, rangkaian *driver relay self bias* dalam Gambar 4.8 dapat disederhanakan dengan menggunakan resistansi pengganti ( $R_{TH}$ ) serta tegangan pengganti ( $V_{TH}$ ) seperti yang ditunjukkan dalam Gambar 4.9.





Gambar 4.9 Penyederhanaan Rangkaian Driver Relay Self Bias

$V_{TH}$  merupakan tegangan yang memicu basis transistor sehingga *relay* dapat aktif, nilai  $V_{TH}$  dikondisikan seolah berasal dari PORT mikrokontroler yaitu 4,2 V. Hubungan antara  $V_{TH}$  dengan  $RB_1$ ,  $RB_2$ , dan  $V_{CC}$  dinyatakan dalam Persamaan (4-11).

$$V_{TH} = \frac{RB_2}{RB_1 + RB_2} \cdot V_{CC} \quad (4-11)$$

$$4,2 V = \frac{RB_2}{RB_1 + RB_2} \cdot 12 V$$

$$RB_1 = \frac{7,8 V \cdot RB_2}{4,2 V}$$

$$RB_1 = 1,85 \cdot RB_2$$

Untuk memperoleh  $V_{TH}$  dengan nilai pendekatan sebesar 4,2 V, maka nilai  $RB_1$  harus dikondisikan 1,85 kali nilai  $RB_2$ . Sedangkan hubungan antara  $R_{TH}$ ,  $RB_1$  serta  $RB_2$  dinyatakan dalam Persamaan (4-12).

$$R_{TH} = \frac{RB_1 \cdot RB_2}{RB_1 + RB_2} \quad (4-12)$$

$$R_{TH} = \frac{1,85 \cdot RB_2 \cdot RB_2}{(1,85 \cdot RB_2) + RB_2}$$

$$= \frac{1,85 \cdot RB_2^2}{2,85} = 0,65 \cdot RB_2$$

$R_{TH}$  dinyatakan dalam  $RB_2$  harus memenuhi hubungan  $R_{TH} = 0,65 \cdot RB_2$ . Mengacu pada *datasheet* transistor 2N2222, Nilai arus kolektor ( $I_C$ ) yang melalui transistor 2N2222 saat kondisi saturasi dapat dihitung berdasarkan Persamaan (4-7).

$$V_{CC} = I_C \cdot R_{COIL} + V_{CE} \quad (4-7)$$

$$12V = I_C \cdot 160\Omega + 0,4V$$

$$I_C = \frac{12V - 0,4V}{160\Omega}$$

$$= \frac{11,6V}{160\Omega}$$

$$= 72,5mA$$

Arus kolektor ( $I_C$ ) yang mengalir pada kaki kolektor transistor 2N2222 sebesar 72,5 mA. Nilai arus basis ( $I_B$ ) yang mengalir pada kaki basis transistor 2N2222 dapat dihitung berdasarkan Persamaan (4-8).

$$I_B = \frac{I_C}{h_{fe}} \quad (4-8)$$

$$I_B = \frac{72,5mA}{75}$$

$$= 0,96mA$$

Nilai arus basis ( $I_B$ ) yang mengalir pada transistor 2N2222 sebesar 0,96 mA.

Nilai  $RB_1$  dan  $RB_2$  dapat dihitung dengan mensubstitusikan nilai  $V_{TH}$ ,  $R_{TH}$ , dan  $I_B$  ke dalam Persamaan (4-9), dimana  $V_{BB}$  digantikan oleh  $V_{TH}$  serta  $RB$  digantikan oleh  $R_{TH}$ .

$$V_{BB} = I_B \cdot RB + V_{BE} \quad (4-9)$$

$$V_{TH} = I_B \cdot R_{TH} + V_{BE}$$

$$4,2V = 0,96mA \cdot (0,65 \cdot RB_2) + 1,3V$$

$$RB_2 = \frac{4,2V - 1,3V}{0,96mA \cdot 0,65}$$

$$= 4647\Omega \approx 4700\Omega$$

$$RB_1 = 1,85 \cdot RB_2$$

$$= 1,85 \cdot 4647\Omega$$

$$= 8597\Omega \approx 8600\Omega$$

Nilai  $RB_2$  sebesar 4700  $\Omega$ , sedangkan nilai  $RB_1$  sebesar 8600  $\Omega$  merupakan seri antara resistor 6,8 k $\Omega$  dengan resistor 1,8 k $\Omega$ .

Untuk mengindikasikan baterai sebagai sumber daya cadangan digunakan LED (*Light Emiting Diode*). Saat sistem terhubung baterai, LED akan menyala. *Forward voltage* sebuah LED sebesar 1,8 V sedangkan arus maksimum yang melewati LED ( $I_{LED}$ ) sebesar 20 mA. Nilai resistor  $R_{LED2}$  yang harus dipasang seri terhadap LED2 dapat dihitung berdasarkan Persamaan (4-10).

$$V_{CC} = I_{LED} \cdot R_{LED} + V_{LED} \quad (4-10)$$

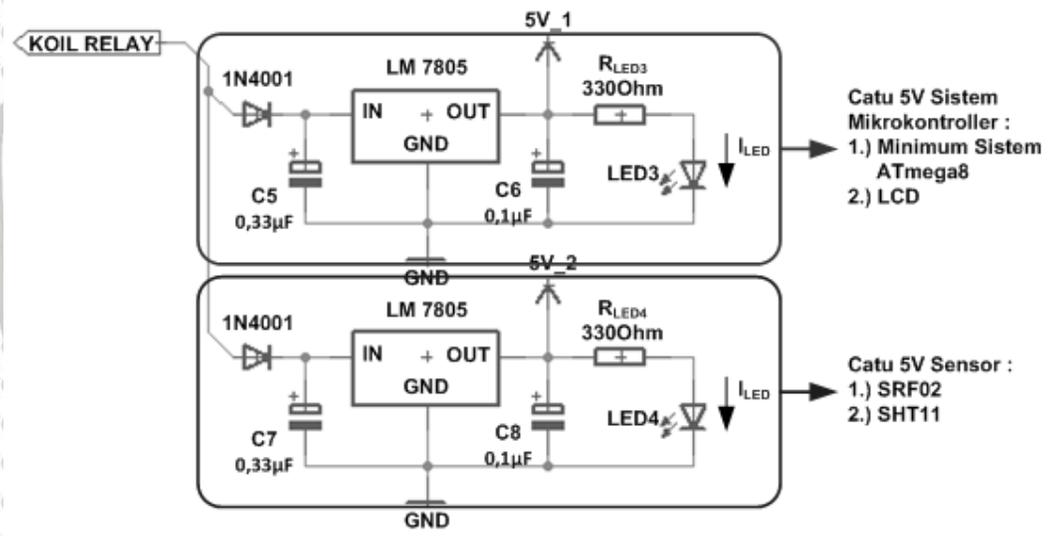
$$12V = 20mA \cdot R_{LED2} + 1,8V$$

$$R_{LED2} = \frac{12V - 1,8V}{20mA}$$

$$= 510\Omega$$

Nilai  $R_{LED2}$  sebesar  $510\Omega$  merupakan nilai minimum agar LED2 tidak putus, maka dari itu digunakan nilai  $R_{LED2}$  sebesar  $680\Omega$ .

Tegangan yang disuplai oleh sumber daya utama yang berasal dari jala-jala PLN ataupun baterai memiliki nilai 12 V, sedangkan beban yang terdiri dari minimum sistem mikrokontroler, sensor-sensor, dan LCD membutuhkan suplai tegangan sebesar 5 V. Maka dibutuhkan rangkaian catu sistem mikrokontroler-sensor yang berfungsi meregulasi tegangan sumber daya utama sebesar 12 V menjadi 5 V. Skema rangkaian catu sistem mikrokontroler-sensor ditunjukkan dalam Gambar 4.10.



Gambar 4.10 Rangkaian Catu Sistem Mikrokontroller-Sensor

Rangkaian catu sistem mikrokontroler-sensor terdiri atas dua buah rangkaian *Fixed Output Regulator* LM7805 yang terpisah, hal ini dikarenakan sensor diletakkan pada lokasi yang terpisah jarak  $\pm 5$  m dari sistem mikrokontroler. Tiap rangkaian *Fixed Output Regulator* LM7805 dilengkapi oleh LED indikator. Nilai resistansi seri LED dapat dihitung berdasarkan Persamaan (4-10).

$$V_{CC} = I_{LED} \cdot R_{LED} + V_{LED} \tag{4-10}$$

$$5V = 20mA \cdot R_{LED3} + 1,8V$$

$$R_{LED3} = \frac{5V - 1,8V}{20mA}$$

$$= 160\Omega$$

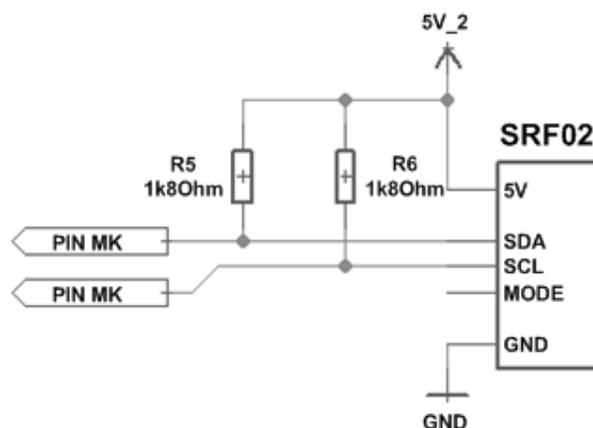


Nilai  $R_{LED3}$  sebesar  $160 \Omega$  merupakan nilai minimum agar LED3 tidak putus, untuk itu digunakan nilai  $R_{LED3}$  sebesar  $330 \Omega$ . Karena kedua rangkaian memiliki tegangan kerja yang sama, maka nilai  $R_{LED4}$  sama dengan nilai  $R_{LED3}$  yaitu sebesar  $330 \Omega$ .

#### 4.2.4 Perancangan Antarmuka Sensor ke Mikrokontroller ATmega8

Sensor SRF02 merupakan unit input yang memberikan masukan berupa data digital ketinggian air ke mikrokontroller. Sensor SRF02 diakses dengan metode komunikasi serial atau komunikasi I2C (*Inter Integrated Circuit*), untuk mengakses SRF02 secara I2C maka PIN *mode* pada SRF02 dikondisikan ambang (*unconnected*) atau terkoneksi dengan suplai tegangan 5 V.

Komunikasi I2C membutuhkan dua jalur, yaitu SDA (*Serial Data Line*) dan SCL (*Serial Clock Line*). Agar komunikasi I2C dapat berfungsi dengan baik maka sangat penting untuk menambahkan sepasang resistor *pull-up* pada masing-masing jalur SDA dan SCL. Resistor *pull-up* memiliki peran penting dalam meminimalisir kondisi *idle*. Skema rangkaian antarmuka SRF02 ke ATmega8 ditunjukkan dalam Gambar 4.11.



Gambar 4.11 Rangkaian Antarmuka SRF02 ke ATmega8

Nilai resistor *pull-up* dapat dihitung berdasarkan nilai *sink current* dalam keadaan *low level output voltage*. Berdasarkan PHILIPS I2C *reference* diketahui bahwa untuk *sink current* sebesar 3 mA, nilai *low level output voltage* nya adalah 0,4 V. Nilai resistor *pull-up* dapat dihitung berdasarkan Persamaan (4-13).

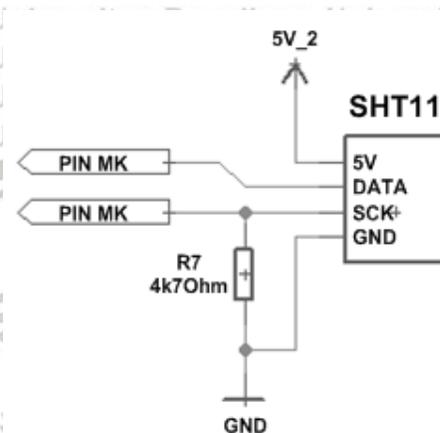
$$R_p = \frac{V_{dd} - V_{ol}}{I_{\text{sink}}} \quad (4-13)$$

$$R_p = \frac{5V - 0,4V}{3mA} = 1533 \Omega \approx 1,8 k\Omega$$

Nilai resistor *pull-up* yang digunakan sebesar  $1,8 k\Omega$ .

Sensor SHT11 merupakan unit input yang memberikan masukan berupa data digital atas nilai suhu dan kelembaban. SHT 11 diakses dengan metode *Two Wire Interface* yang artinya SHT11 membutuhkan dua jalur untuk berkomunikasi.

Kedua jalur komunikasi *2-Wire Interface* dari sensor suhu-kelembaban SHT11 tersebut adalah *DATA line* dan *SCK line*. Skema rangkaian antarmuka SHT11 ke ATmega8 ditunjukkan dalam Gambar 4.12.



Gambar 4.12 Rangkaian Antarmuka SHT11 ke ATmega8

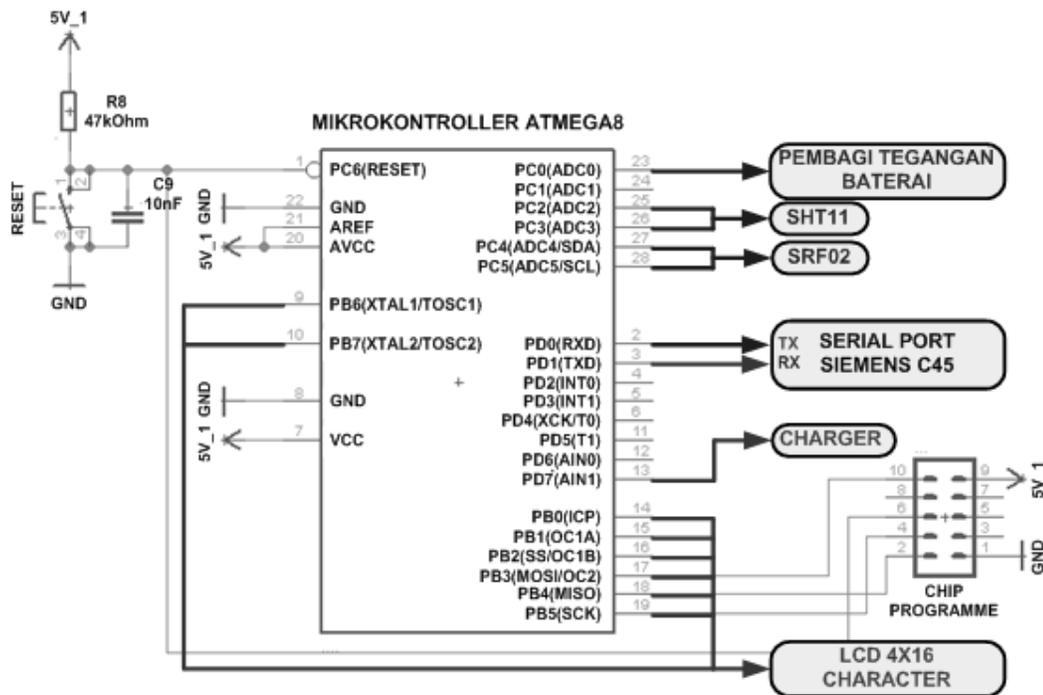
Berdasarkan *datasheet* SENSIRION SHT *Module*, nilai R7 sebesar 4,7 k $\Omega$  diperlukan jika SHT11 digunakan pada aplikasi dengan tingkat keamanan bit data yang tinggi.

#### 4.2.5 Perancangan Minimum Sistem Mikrokontroler ATmega8

Mikrokontroler sebagai unit pemroses data berfungsi untuk melakukan akuisisi data dari sensor SRF02 dan SHT11, mengatur proses *charging*, serta melakukan kontrol penuh terhadap transmisi data yang dilakukan via SMS. Jenis mikrokontroler yang digunakan dalam perancangan adalah ATmega8.

Minimum sistem pada mikrokontroler ATmega8 terdiri dari sebuah mikrokontroler ATmega8, 10 buah PIN eksternal untuk memasukkan program ke *flash memory*, dan sebuah modul *RESET* yang terdiri dari sebuah kapasitor, resistor *pull-up*, serta sebuah *pushbutton*. Minimum sistem mikrokontroler ATmega8 tidak dilengkapi dengan *external crystal oscillator* sehingga kecepatan *clock* dari mikrokontroler ATmega8 akan mengikuti *default shipped clock speed* sebesar 8 MHz.

PIN-PIN pada mikrokontroler ATmega8 akan terkoneksi dengan modul sensor SRF02, modul sensor SHT11, rangkaian pembagi tegangan baterai, *charger*, *handphone* SIEMENS C45 serta LCD 4X16 karakter. Konfigurasi PIN pada minimum sistem ATmega8 ditunjukkan dalam Gambar 4.13.



Gambar 4.13 Konfigurasi PIN pada Minimum Sistem ATmega8

Mikrokontroler ATmega8 terdiri atas tiga PORT I/O yang dapat diprogram yaitu PORTB dengan jumlah data 8 bit, PORTD dengan jumlah data 8 bit dan PORTC dengan jumlah data 7 bit. PIN-PIN yang digunakan dalam perancangan minimum sistem mikrokontroler ATmega8 adalah sebagai berikut :

- 1). PINB.0 : dihubungkan dengan PIN RS pada LCD 4X16.
- 2). PINB.1 : dihubungkan dengan PIN R/W pada LCD 4X16.
- 3). PINB.2 : dihubungkan dengan PIN ENABLE pada LCD 4X16.
- 4). PINB.4 : dihubungkan dengan PIN D4 pada LCD 4X16.
- 5). PINB.5 : dihubungkan dengan PIN D5 pada LCD 4X16.
- 6). PINB.6 : dihubungkan dengan PIN D6 pada LCD 4X16.
- 7). PINB.7 : dihubungkan dengan PIN D7 pada LCD 4X16.
- 8). PINC.0 : dihubungkan dengan output pembagi tegangan baterai.
- 9). PINC.2 : dihubungkan dengan PIN SCK dari modul SHT11.
- 10). PINC.3 : dihubungkan dengan PIN DATA dari modul SHT11.
- 11). PINC.4 : dihubungkan dengan PIN SDA dari modul SRF02.
- 12). PINC.5 : dihubungkan dengan PIN SCL dari modul SRF02.
- 13). PINC.6 : dihubungkan dengan modul *RESET*.
- 14). PIND.0 : dihubungkan dengan PIN TX dari SIEMENS C45.

15). PIND.1 : dihubungkan dengan PIN RX dari SIEMENS C45.

16). PIND.7 : dihubungkan dengan *driver relay* pada *charger*.

Modul *RESET* pada minimum sistem ATmega8 disusun dengan menggunakan resistor dan kapasitor yang terhubung dengan PIN *RESET* mikrokontroller. Hubungan antara waktu *reset*, R8, C9 ditunjukkan dalam Persamaan (4-14).

$$t = -R8.C9.\ln\left(1 - \frac{V_{RST}}{V_{CC}}\right) \quad (4-14)$$

Berdasarkan *datasheet* ATmega8 dapat diketahui bahwa waktu minimal untuk *mereset* mikrokontroller sebesar 1,5  $\mu$ s, sedangkan tegangan logika rendah pada PIN *RESET* ( $V_{RST}$ ) sebesar  $0,1 \times V_{CC}$  yaitu 0,5 V, dan  $V_{CC}$  pada mikrokontroller ATmega8 sebesar 5 V. Nilai R8 harus bernilai antara 30 k $\Omega$  – 80 k $\Omega$ , dengan menentukan nilai R8 sebesar 47 k $\Omega$  dan C9 sebesar 10 nF akan didapat waktu *reset* selama :

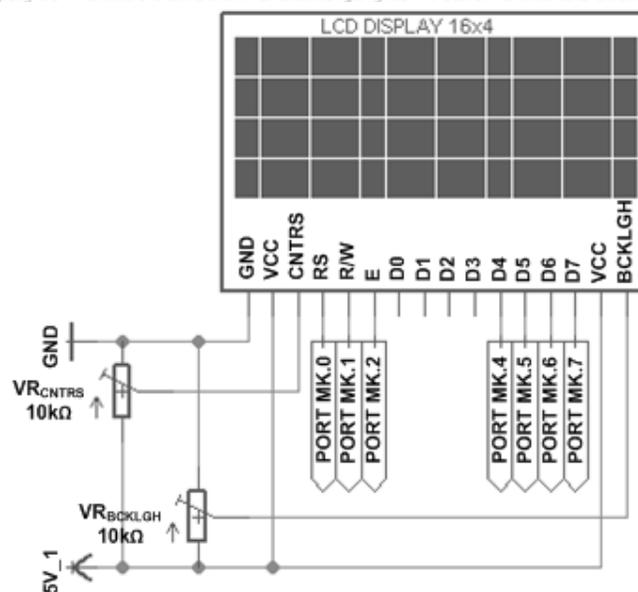
$$\begin{aligned} t &= -47 \text{ k}\Omega \cdot 10 \text{ nF} \cdot \ln\left(1 - \frac{0,5 \text{ V}}{5 \text{ V}}\right) \\ &= 47 \text{ k}\Omega \cdot 10 \text{ nF} \cdot 0,105 \\ &= 49,5 \mu\text{s} \end{aligned}$$

Nilai t sebesar 49,5  $\mu$ s sudah memenuhi batas waktu yang dibutuhkan untuk melakukan *reset* ke mikrokontroller ATmega8.

#### 4.2.6 Perancangan Rangkaian LCD (*Liquid Crystal Display*) 4X16 Karakter

Pada perancangan sistem ini digunakan LCD modul 1604A yang merupakan sebuah modul LCD *dot matrix* yang membutuhkan daya kecil. LCD modul 1604 dilengkapi panel LCD dengan tingkat kontras yang cukup tinggi serta pengendali LCD CMOS yang telah terpasang dalam modul tersebut. Rangkaian koneksi LCD 4X16 karakter ke mikrokontroller ditunjukkan dalam Gambar 4.14. LCD modul 1604A mempunyai spesifikasi sebagai berikut :

- 1). Memiliki 16 karakter dan 4 baris tampilan yang terdiri atas 5X7 dot matrik ditambah dengan kursor.
- 2). Memerlukan catu daya DC 5 V.
- 3). *Automatic reset* saat catu daya dinyalakan.
- 4). 80 X 8 *display* data RAM (max 80 karakter).
- 5). Menggunakan 4 bit data dan 3 bit kontrol.
- 6). *Adjustable contrast*.
- 7). *Adjustable backlight*.



Gambar 4.14 Rangkaian Koneksi LCD 4X16 Karakter ke Mikrokontroler ATmega8

Dalam perancangan diinginkan agar kontras dan *backlight* dari LCD dapat diatur, maka dipasangkan sebuah variabel resistor dengan nilai 10 kΩ pada masing-masing PIN *contrast* dan *backlight* dari LCD 4X16 karakter untuk mengatur nilai tegangan yang masuk ke PIN *contrast* dan *backlight* nya.

#### 4.2.7 Perancangan Antarmuka *Handphone* SIEMENS C45 ke Mikrokontroler ATmega8

SIEMENS C45 merupakan jenis *handphone* yang telah dilengkapi dengan PIN RX dan PIN TX untuk berkomunikasi secara serial dengan perangkat lain seperti komputer ataupun mikrokontroler. Hal ini menjadikan *handphone* SIEMENS C45 sebagai pilihan yang tepat untuk mendukung aplikasi telekomunikasi berbasis mikrokontroler sebagai pengendali transmisi data.

Mikrokontroler ATmega8 berkomunikasi dengan *handphone* SIEMENS C45 melalui metode komunikasi serial dengan parameter komunikasi serial 8data, 1stop, no *parity* dengan *baudrate* 19200 bps. *Baudrate* sebesar 19200 bps digunakan untuk mengikuti *baudrate* dari standar komunikasi *handphone* SIEMENS C45.

Komunikasi serial membutuhkan tiga buah jalur yaitu RX, TX, dan *ground*. Agar mikrokontroler ATmega8 dapat berkomunikasi dengan *handphone* SIEMENS C45 maka PIN RX dari mikrokontroler harus terhubung PIN TX dari *handphone* SIEMENS C45, sedangkan PIN TX dari mikrokontroler harus terhubung PIN RX dari *handphone* SIEMENS C45, dan *ground* keduanya saling terhubung.



*Handphone* SIEMENS C45 memiliki 12 PIN ekspansi eksternal yang mencakup PIN RX dan TX yang dibutuhkan untuk melakukan komunikasi serial. Konfigurasi PINOUT *handphone* SIEMENS C45 ditunjukkan dalam Gambar 4.15. Fungsi masing-masing PIN pada *handphone* SIEMENS C45 ditunjukkan dalam Tabel 4.1.

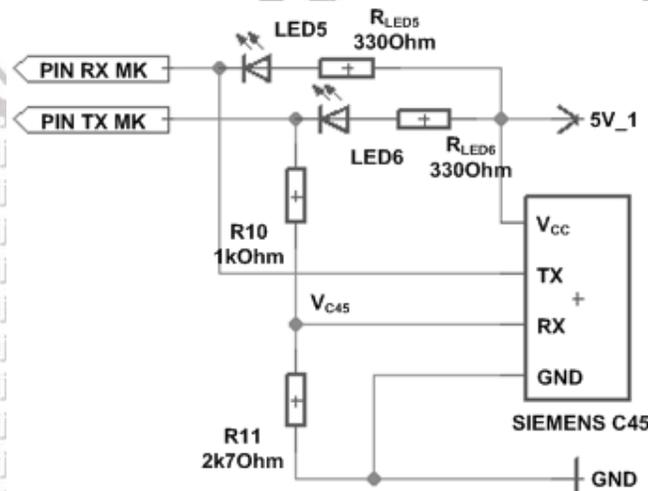


Gambar 4.15 Konfigurasi PINOUT *Handphone* SIEMENS C45

Tabel 4.1 Fungsi Masing-Masing PIN pada *Handphone* SIEMENS C45

PIN	Nama	State	Keterangan
1	Gnd	-	Ground
2	Self Service	In/Out	Recognition/control battery charger
3	Load	In	Charging voltage
4	Battery	Out	Battery (S25 only)
5	Data Out (TX)	Out	Data sent
6	Data In (RX)	In	Data received
7	Z_Clock	-	Clock line for accessory bus. Use as DCD In data operation
8	Z_Data	-	Data line for accessory bus. Use as CTS in data operation
9	MICG	-	Ground for microphone
10	MIC	In	Microphone input
11	AUD	Out	Loudspeaker
12	AUDG	-	Ground for external speaker

Untuk mengindikasikan berjalannya komunikasi serial antara mikrokontroler ATmega8 dengan *handphone* SIEMENS C45, maka ditambahkan sepasang LED dengan konfigurasi *pull-up* pada masing-masing PIN TX dan RX. Masing-masing LED akan menyala jika terdeteksi logika 0 pada PIN TX atau RX. Skema rangkaian antarmuka SIEMENS C45 ke ATmega8 ditunjukkan dalam Gambar 4.16.



Gambar 4.16 Rangkaian Antarmuka SIEMENS C45 ke ATmega8

Level tegangan pada PIN mikrokontroler adalah 0 – 5 V, sedangkan level tegangan pada PIN SIEMENS C45 adalah 0 – 3,6 V, maka dari itu dibutuhkan rangkaian pembagi tegangan untuk mengkonversi level tegangan TTL dari mikrokontroler ke level 0 – 3,6 V. Tegangan yang masuk ke PIN SIEMENS C45 dapat dihitung berdasarkan hubungan bagi tegangan dalam Persamaan (4-15).

$$V_{C45} = \frac{R11}{R11 + R10} \cdot V_{CC}$$

$$3,6V = \frac{R11}{R11 + R10} \cdot 5V \quad (4-15)$$

Dengan mengambil nilai R10 sebesar 1 k $\Omega$ , maka nilai R11 dapat ditentukan berdasarkan Persamaan (4-15).

$$3,6V = \frac{R11}{R11 + 1k\Omega} \cdot 5V$$

$$R11 = \frac{3,6V \cdot 1k\Omega}{5V - 3,6V}$$

$$= 2571\Omega \approx 2,7k\Omega$$

Nilai R11 sebesar 2571  $\Omega$  merupakan nilai minimum, digunakan R11 sebesar 2,7 k $\Omega$ .

PIN TX dari *handphone* SIEMENS C45 terhubung langsung dengan PIN RX dari mikrokontroler. Untuk mengantisipasi habisnya energi pada baterai SIEMENS C45, maka tegangan 5 V dihubungkan pada PIN *charge handphone* SIEMENS C45.

Struktur data komunikasi serial secara umum adalah munculnya tegangan 0 V pada PIN TX atau RX yang merepresentasikan logika 1 secara biner. PORT serial akan mengeluarkan *low level voltage* ketika terdeteksi ada data yang masuk atau ada data yang keluar, sedangkan LED diinginkan menyala ketika terdeteksi ada data masuk atau ada data keluar. Maka dari itu LED dikonfigurasi secara *pull-up* sehingga ketika terdeteksi data berlogika *low level voltage*, LED akan menyala. Nilai resistor R<sub>LED5</sub> dihitung berdasarkan Persamaan (4-10).

$$V_{CC} = I_{LED} \cdot R_{LED} + V_{LED} \quad (4-10)$$

$$5V = 20mA \cdot R_{LED5} + 1,8V$$

$$R_{LED5} = \frac{5V - 1,8V}{20mA}$$

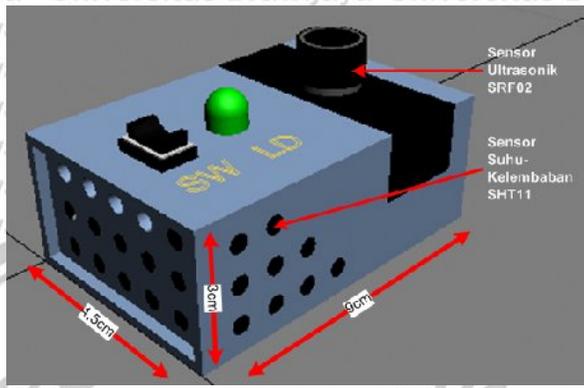
$$= 160\Omega$$

Nilai R<sub>LED5</sub> sebesar 160  $\Omega$  merupakan nilai minimum agar LED5 tidak putus, untuk itu digunakan nilai R<sub>LED5</sub> sebesar 330  $\Omega$ . Karena memiliki tegangan kerja yang sama, maka nilai R<sub>LED5</sub> sama dengan nilai R<sub>LED6</sub> yaitu sebesar 330  $\Omega$ .

### 4.2.8 Perancangan Sistem Mekanik

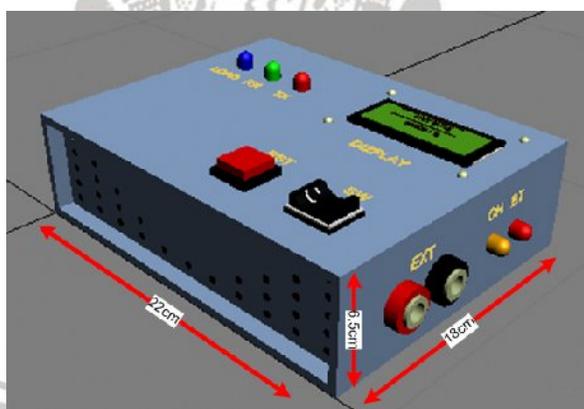
Sistem mekanik membagi RTU (*Remote Telemetry Unit*) menjadi dua bagian terpisah yaitu bagian *monitoring* dan bagian *processing*. Sistem mekanik bagian *monitoring* berisikan sensor ultrasonik SRF02 dan sensor suhu-kelembaban SHT11.

Sistem mekanik bagian *monitoring* ditunjukkan dalam Gambar 4.17.



Gambar 4.17 Sistem Mekanik Bagian *Monitoring*

Sistem mekanik bagian *processing* berisikan seluruh komponen penyusun RTU diluar sensor ultrasonik SRF02 dan sensor suhu-kelembaban SHT11. Sistem mekanik bagian *processing* ditunjukkan dalam Gambar 4.18.



Gambar 4.18 Sistem Mekanik Bagian *Processing*

Agar RTU bekerja dengan baik, maka sistem mekanik bagian *monitoring* dan *processing* harus dihubungkan. Untuk menghubungkan sistem mekanik bagian *monitoring* dan *processing* dibutuhkan setidaknya enam jalur (12 V, GND, SDA, SDL, DATA, SCK). Maka digunakan konektor RJ45 sebagai penghubung sistem mekanik bagian *monitoring* dan *processing*. Konektor RJ45 memiliki delapan jalur yang dapat digunakan, disamping itu terdapat kait pengunci untuk lebih memperkuat koneksi antara sistem mekanik bagian *monitoring* dan *processing*.



Sistem mekanik bagian *monitoring* dipasang pada kondisi menghadap permukaan air sungai pada ketinggian normal dengan jarak 600 cm dari muka air, sedangkan sistem mekanik bagian *processing* diletakkan pada suatu *shelter* dengan jarak maksimum 500 cm dari sistem mekanik bagian *monitoring*.

### 4.3 Perancangan Perangkat Lunak

#### 4.3.1 Perancangan Format Data Transmisi

Data parameter banjir yang akan dikirim ke *Server* meliputi data kode RTU (*Remote Telemetry Unit*), ketinggian permukaan air, data suhu serta kelembaban udara. Data-data tersebut harus dapat dikenali oleh *Server* serta memiliki pemisah/*separator* untuk masing-masing besaran yang dikirim.

Sebagai penanda data transmisi yang berasal dari RTU, maka ditambahkan *string* "Data", karakter "\*" digunakan sebagai separator. Data yang akan ditransmisikan oleh RTU ke *Server* melalui SMS akan mengikuti format data sebagai berikut :

**Data\*Kode RTU\*Ketinggian Permukaan Air\*Suhu\*Kelembaban\***

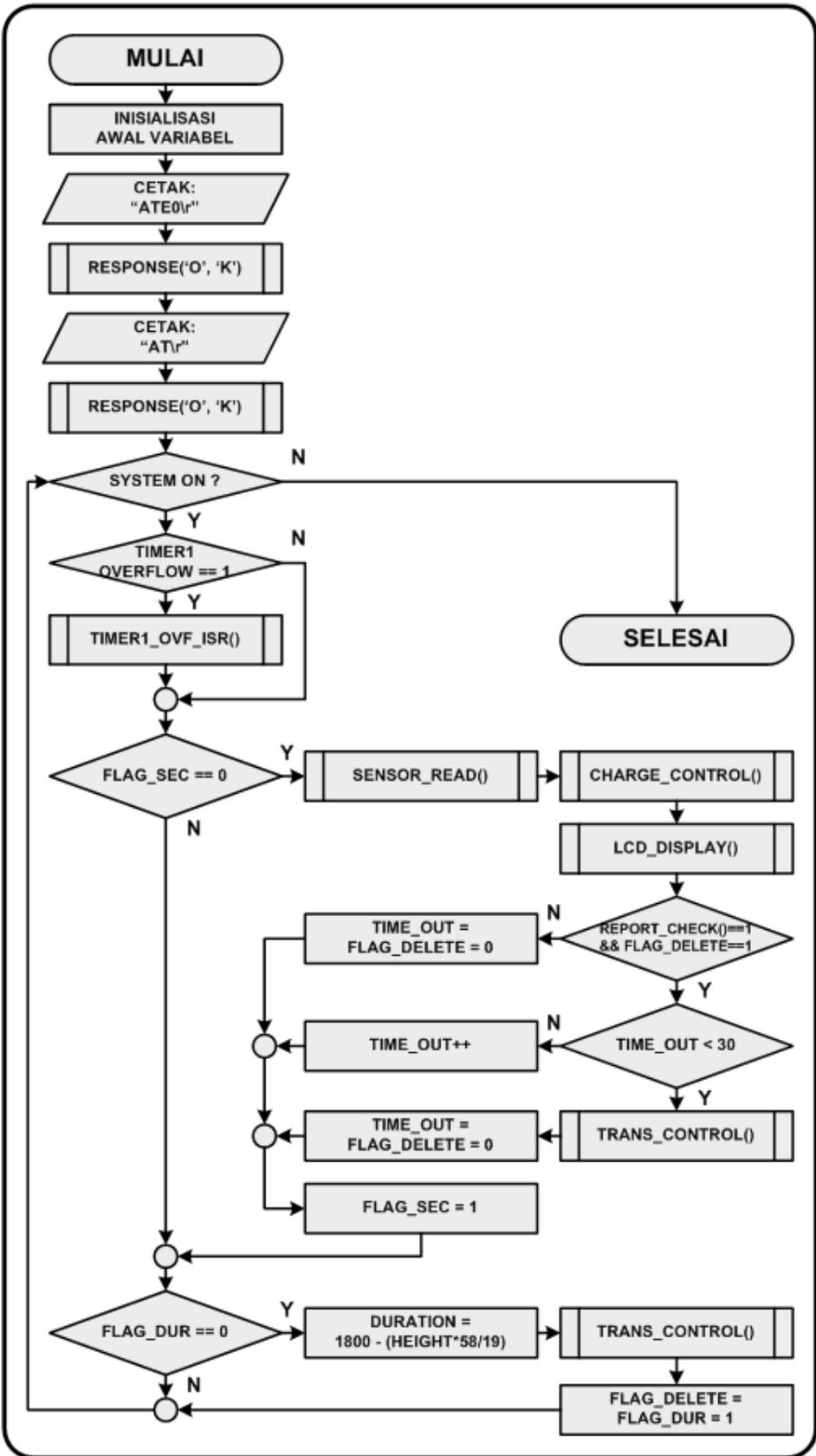
Format data diatas berisikan lima jenis informasi yang siap dikirim ke *Server*, berikut adalah keterangan detail dari format data transmisi ke *Server* :

<b>Data</b>	: Identifier transmisi ke Server
<b>Kode RTU</b>	: 1 Byte kode RTU
<b>Tinggi Air</b>	: 3 Byte data ketinggian permukaan air.
<b>Suhu</b>	: 2 Byte data suhu
<b>Kelembaban</b>	: 2 Byte data kelembaban

Total data yang ditransmisikan ke *Server* sebesar 17 byte dengan komposisi 17 karakter. Misalkan data dikirim dari RTU dengan kode 1 dimana ketinggian air sebesar 320 cm, sedangkan suhu sebesar 28°C dan kelembaban sebesar 77%RH, maka data yang akan ditransmisikan oleh RTU1 ke *Server* adalah "Data\*1\*320\*28\*77\*".

#### 4.3.2 Perancangan Algoritma Program Utama

Tujuan dari program utama adalah mengatur urutan kerja sistem yang diwujudkan melalui suatu algoritma, sehingga sistem mampu menjalankan fungsinya dengan baik. Secara umum tugas yang harus dikerjakan oleh program utama meliputi pengumpulan data ketinggian air, suhu, dan kelembaban udara, menampilkan data sensor ke LCD 4X16 karakter, mengontrol proses *charging*, serta mengontrol transmisi data ke *Server*. Algoritma program utama RTU ditunjukkan dalam Gambar 4.19.



Gambar 4.19 Algoritma Program Utama RTU



Program utama RTU (*Remote Telemetry Unit*) tersusun atas beberapa subrutin yang meliputi subrutin RESPONSE, subrutin TIMER1\_OVF\_ISR, subrutin SENSOR\_READ, subrutin CHARGE\_CONTROL, subrutin LCD\_DISPLAY, subrutin TRANS\_CONTROL, dan subrutin REPORT\_CHECK. Beberapa subrutin tersebut akan dieksekusi secara kontinyu sesuai dengan algoritma program utama RTU (*Remote Telemetry Unit*) selama sistem dalam keadaan ON.

Saat sistem ON, hal yang pertama kali dilakukan adalah memberi inisialisasi awal pada nilai variabel-variabel yang ada dalam program, dan dilanjutkan dengan pengecekan komunikasi serial ke *handphone* SIEMENS C45 dengan mengirimkan *string* "ATE0\r" untuk menghilangkan *echo*, dan pengiriman *string* "AT\r" untuk mengetahui respon *handphone* terhadap instruksi *AT-Command*. Jika koneksi berhasil, maka akan dikembalikan *string* "OK". Pengecekan *string* "OK" dilakukan dengan mengirim karakter 'O' dan 'K' ke subrutin RESPONSE.

Sistem akan melakukan sampling tiap 1 detik serta melakukan transmisi data dengan durasi tertentu, maka digunakan TIMER1 (16 bit) pada mikrokontroller ATmega8 yang *interrupt* tiap 1 detik. Jika TIMER1 *interrupt*, maka subrutin TIMER1\_OVF\_ISR akan dieksekusi. Pada subrutin TIMER1\_OVF\_ISR terdapat variabel-variabel yang digunakan sebagai pembatas sekaligus penanda waktu transmisi data ke *Server* (FLAG\_DUR), serta penanda operasi sampling (FLAG\_SEC).

Jika variabel FLAG\_SEC bernilai 0, maka subrutin SENSOR\_READ, subrutin CHARGE\_CONTROL, dan subrutin LCD\_DISPLAY dieksekusi secara berurutan. Subrutin SENSOR\_READ berisi algoritma pembacaan dan akuisisi data dari sensor SRF02 dan SHT11. Subrutin CHARGE\_CONTROL berisi algoritma pengontrolan proses *charging* ke baterai. Sedangkan subrutin LCD\_DISPLAY berisi algoritma penampil data sensor ke LCD 4X16 karakter.

Subrutin REPORT\_CHECK akan dieksekusi untuk memeriksa terdeteksi atau tidaknya *report* atas transmisi data terakhir yang dikirim melalui SMS. Jika SMS *report* terdeteksi maka nilai yang akan dikembalikan oleh subrutin REPORT\_CHECK adalah 0, sebaliknya jika SMS *report* belum terdeteksi maka nilai yang akan dikembalikan oleh subrutin REPORT\_CHECK adalah 1. Jika nilai yang dikembalikan oleh subrutin REPORT\_CHECK dan variabel FLAG\_DELETE sama-sama bernilai 1 maka nilai variabel TIME\_OUT akan diperiksa. Jika TIME\_OUT bernilai kurang dari 30 maka TIME\_OUT akan di *increment*, jika tidak maka subrutin TRANS\_CONTROL akan dieksekusi sekali lagi dan variabel TIME\_OUT serta FLAG\_DELETE diisi 0.

Eksekusi ulang terhadap subrutin TRANS\_CONTROL ditujukan untuk melakukan transmisi data ulang dengan *update* data terbaru ketika SMS *report* tidak terdeteksi selama 30 detik lamanya. Pengisian nilai 0 terhadap variabel TIME\_OUT serta FLAG\_DELETE dilakukan untuk menandai bahwa transmisi data ulang telah dilakukan, setelah selesai FLAG\_SEC akan diisi dengan nilai 1 untuk mencegah eksekusi berulang yang dapat terjadi selama 1 detik.

Jika FLAG\_DUR bernilai 0, maka program dilanjutkan dengan penentuan nilai *integer* yang akan diisikan ke variabel DURATION. Kondisi variabel DURATION yang diinginkan dapat dijelaskan sebagai berikut :



Fungsi alih untuk menentukan nilai variabel DURATION berdasarkan kondisi diatas dapat dicari dengan memanfaatkan sifat kelinearannya melalui persamaan garis seperti yang ditunjukkan dalam Persamaan (4-16).

$$\frac{DURATION - DURATION_1}{DURATION_2 - DURATION_1} = \frac{HEIGHT - HEIGHT_1}{HEIGHT_2 - HEIGHT_1} \quad (4-16)$$

Substitusi terhadap nilai-nilai HEIGHT<sub>1</sub>, HEIGHT<sub>2</sub>, DURATION<sub>1</sub>, serta DURATION<sub>2</sub> dalam Persamaan (4-16) akan menghasilkan fungsi alih dari variabel HEIGHT ke variabel DURATION.

$$\frac{DURATION - 1800}{60 - 1800} = \frac{HEIGHT - 0}{570 - 0}$$

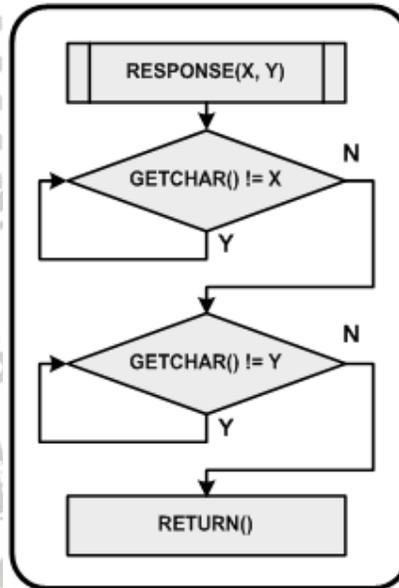
$$DURATION = 1800 - \left( \frac{58}{19} \cdot HEIGHT \right)$$

Fungsi alih dari ketinggian permukaan air (HEIGHT) ke DURATION adalah DURATION = 1800 - (58\*HEIGHT/19).

Setelah didapatkan nilai yang harus diisikan ke variabel DURATION, maka proses dilanjutkan dengan eksekusi terhadap subrutin TRANS\_CONTROL untuk mengirimkan data-data ketinggian permukaan air, suhu, serta kelembaban udara ke *Server* melalui *handphone* SIEMENS C45. Setelah subrutin TRANS\_CONTROL tereksekusi, maka nilai variabel FLAG\_DUR diisi dengan 1 untuk mencegah terjadinya eksekusi berulang selama menunggu nilai DURATION yang baru. Selanjutnya nilai FLAG\_DELETE diisi dengan 1 sebagai penanda belum masuknya SMS *report*.

### 4.3.3 Perancangan Subrutin RESPONSE

Subrutin RESPONSE berfungsi sebagai pemeriksa kecocokan sepasang karakter yang masuk secara serial sebagai respon atas perintah *AT-Command* dengan nilai sepasang karakter ASCII yang dikirim oleh fungsi pemanggilnya. Algoritma subrutin RESPONSE ditunjukkan dalam Gambar 4.20.



Gambar 4.20 Algoritma Subrutin RESPONSE

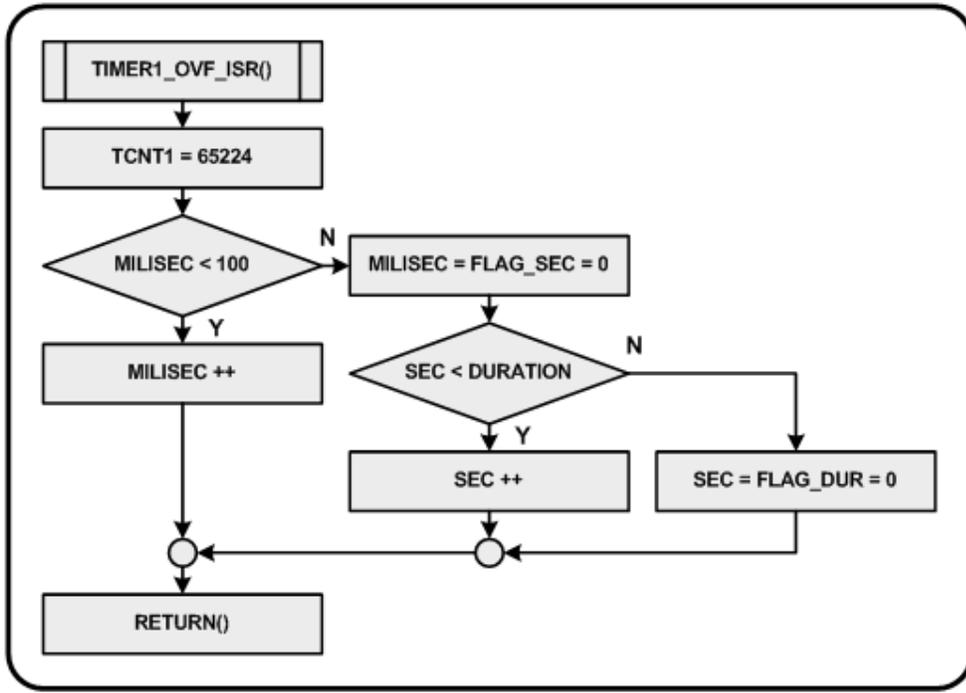
Sepasang karakter yang dikirim oleh fungsi pemanggil akan disalin ke variabel X dan Y pada subrutin RESPONSE. Selanjutnya perintah *getchar()* akan membaca data serial yang masuk. Jika karakter pertama tidak sama dengan nilai X, maka program akan berulang terus menerus hingga ditemukan karakter yang bernilai sama dengan X. Jika karakter pertama telah sama dengan X, maka karakter kedua akan dibaca.

Jika karakter kedua tidak sama dengan nilai Y, maka program akan berulang terus menerus hingga ditemukan karakter yang bernilai sama dengan Y. Jika karakter pertama dan kedua telah sama dengan variabel X dan Y, maka program akan dikembalikan ke fungsi pemanggilnya.

### 4.3.4 Perancangan Subrutin TIMER1\_OVF\_ISR

Subrutin TIMER1\_OVF\_ISR berfungsi untuk mengatur periode sampling pada operasi sistem serta mengatur durasi pengiriman data dari sistem ke *Server*. Periode sampling operasi sistem adalah 1 detik, artinya sistem akan membaca data sensor dan melakukan operasi lainnya setiap 1 detik. Sedangkan durasi pengiriman data dari sistem ke *Server* ditentukan oleh ketinggian permukaan air (variabel HEIGHT).

TIMER1 pada mikrokontroler ATmega8 dirancang untuk bekerja dengan frekuensi *clock* 31,25 kHz (*prescaler* 256) dengan nilai *start* pada register TCNT1 sebesar 0XFEC8 (dinyatakan dalam desimal 65224), sehingga *interrupt* tiap 10 ms. Artinya TIMER1 akan mulai mengitung dari 0XFEC8 hingga 0XFFFF dengan periode 31,98  $\mu$ s, sehingga total waktu yang dibutuhkan sebesar 10 ms. Algoritma subrutin TIMER1\_OVF\_ISR ditunjukkan dalam Gambar 4.21.

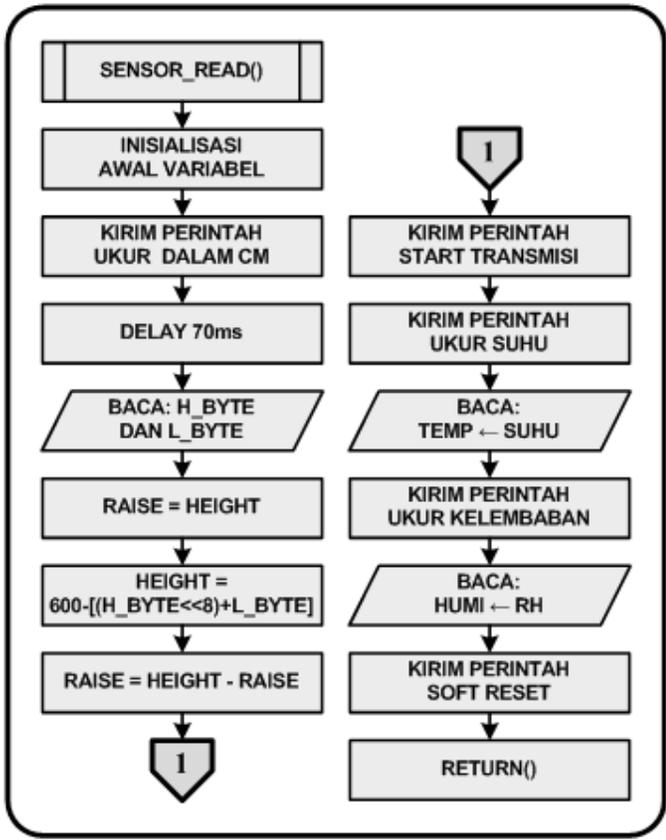


Gambar 4.21 Algoritma Subrutin TIMER1\_OVF\_ISR

Tiap *interrupt* terjadi, register TCNT1 diisi dengan nilai 0XFEC8. Jika variabel MILISEC kurang dari 100 maka variabel MILISEC akan di *increment*, jika tidak maka variabel MILISEC dan FLAG\_SEC akan diisi dengan nilai 0 sebagai penanda waktu sampling. Jika variabel SEC kurang dari nilai DURATION, maka variabel SEC akan di *increment*, jika tidak maka variabel SEC dan FLAG\_DUR akan diisi dengan nilai 0 yang menandakan waktu untuk melakukan pengiriman data ke *Server*. Setelah selesai maka program akan dikembalikan ke posisi terakhir saat *interrupt* terjadi.

### 4.3.5 Perancangan Subrutin SENSOR\_READ

Subrutin SENSOR\_READ berfungsi untuk mengambil serta mengakusisi data dari sensor. Pembacaan data sensor SRF02 dilakukan dengan metode komunikasi I2C, sedangkan pembacaan data sensor SHT11 dilakukan dengan metode komunikasi *2-Wire Interface*. Algoritma subrutin SENSOR\_READ ditunjukkan dalam Gambar 4.22.



Gambar 4.22 Algoritma Subrutin SENSOR\_READ

Algoritma subrutin SENSOR\_READ diawali dengan inisialisasi awal variabel kemudian dilanjutkan pembacaan data dari sensor ultrasonik SRF02 lalu sensor suhu-kelembaban SHT11. Pembacaan data sensor ultrasonik SRF02 diawali dengan pengiriman perintah pengukuran ketinggian dalam cm (heksa 0X51) ke register0. Setelah perintah diterima maka SRF02 akan memancarkan gelombang ultrasonik dan menangkap kembali *echo* nya, proses ini memakan waktu sekitar 70 ms.

Hasil pengukuran ketinggian oleh SRF02 memiliki lebar 16 bit dan disimpan ke register2 dan register3 dengan lebar 8 bit. Data H\_BYTE pada register2 dibaca terlebih dahulu setelah itu data L\_BYTE pada register3 dibaca. Setelah selesai nilai variabel HEIGHT akan disalin ke variabel RAISE.

Data HEIGHT merupakan selisih nilai 600 terhadap hasil penjumlahan dari pergeseran variabel H\_BYTE dari register2 ke kiri sejauh 8 bit dengan nilai variabel L\_BYTE dari register3 pada sensor ultrasonik SRF02. Variabel RAISE merupakan selisih dari variabel HEIGHT dengan nilai lama dari variabel RAISE, dengan periode sampling 1 detik, maka variabel RAISE merupakan hasil perhitungan kecepatan perubahan ketinggian permukaan air yang dinyatakan dalam cm/s.



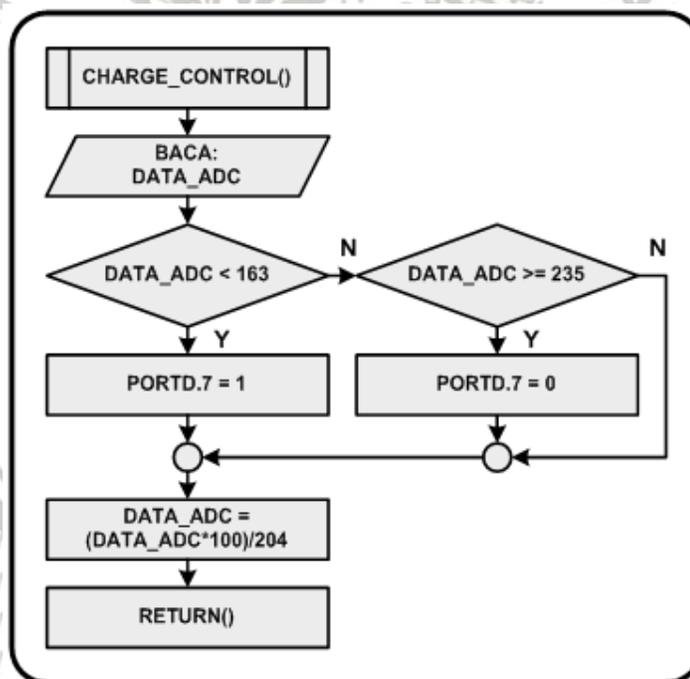
Pembacaan data SHT11 diawali dengan pengiriman perintah memulai transmisi.

Untuk mendapatkan data suhu, terlebih dahulu dikirimkan perintah untuk mengukur suhu, dilanjutkan dengan pembacaan data suhu yang kemudian disimpan ke variabel TEMP. Untuk mendapatkan data kelembaban, terlebih dahulu dikirimkan perintah untuk mengukur kelembaban, dilanjutkan dengan pembacaan data kelembaban yang kemudian disimpan ke variabel HUMI.

Setelah kedua data suhu dan kelembaban tersimpan di variabel TEMP dan HUMI, maka perintah *soft reset* diberlakukan dengan mengirimkan sinyal *high* pada DATA line, dan *toggle* pada SCK line sejumlah 9 kali atau lebih.

#### 4.3.6 Perancangan Subrutin CHARGE\_CONTROL

Kontrol terhadap *charging* baterai dilakukan berdasarkan input tegangan analog (0 – 5 V merepresentasikan tegangan baterai 0 – 15 V) dari keluaran rangkaian pembagi tegangan baterai yang ditangkap oleh PIN ADC mikrokontroler ATmega8. Algoritma subrutin CHARGE\_CONTROL ditunjukkan dalam Gambar 4.23.



Gambar 4.23 Algoritma Subrutin CHARGE\_CONTROL

Kontrol terhadap proses *charging* diawali dengan pembacaan level tegangan dari keluaran rangkaian pembagi tegangan baterai yang terhubung ke PIN ADC pada mikrokontroler ATmega8. Resolusi ADC yang digunakan adalah 8 bit, sehingga DATA\_ADC berada dalam *range* 0 hingga 255.

Jika nilai variabel DATA\_ADC kurang dari 163 (representasi 9,6 V) maka mikrokontroler akan mengaktifkan *charger* untuk mengisi kembali energi baterai dengan memberikan nilai 1 pada PORTD.7. Proses *charging* akan dihentikan jika DATA\_ADC mencapai 235 (representasi 13,8 V) atau lebih dengan memberikan nilai 0 pada PORTD.7. *Range* besaran hasil konversi ADC ditunjukkan dalam Tabel 4.2.

Tabel 4.2 *Range* Besaran Hasil Konversi ADC

$V_{BATT}$ (Volt)	Keluaran Pembagi Tegangan (Volt)	DATA_ADC (Int)	DATA_ADC (%)
<i>Range</i> : 0 – 15 V	$V_{OUT} = \frac{5030 \Omega}{15030 \Omega} \cdot V_{BATT}$	$Data_{ADC} = \frac{V_{OUT}}{5V} \cdot 255$	$Data_{ADC} = \frac{Data_{ADC}}{204} \cdot 100\%$
	<i>Range</i> : 0 – 5 V	<i>Range</i> : 0 – 255	<i>Range</i> : 0 – 125%

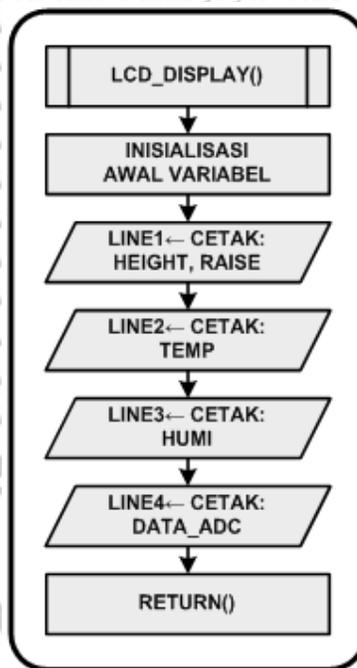
Setelah data ADC 8 bit masuk ke variabel DATA\_ADC dalam SRAM mikrokontroler, maka proses dilanjutkan dengan mengkonversi DATA\_ADC ke dalam persen (%) dimana DATA\_ADC dikalikan dengan 100% lalu dibagi 204 (representasi 12 V). DATA\_ADC kini merepresentasikan level tegangan baterai 0 hingga 12 V yang dinyatakan dalam persentase bilangan 0 hingga 100%.

#### 4.3.7 Perancangan Subrutin LCD\_DISPLAY

Subrutin LCD\_DISPLAY berfungsi mengatur tampilan data sensor pada tiap baris LCD 4X16 karakter. Data yang ditampilkan pada LCD 4X16 karakter adalah variabel HEIGHT dan RAISE, variabel TEMP, variabel HUMI, serta variabel DATA\_ADC. Variabel HEIGHT dan RAISE ditampilkan pada baris 1 LCD, variabel TEMP ditampilkan pada baris 2 LCD, dan variabel HUMI ditampilkan pada baris 3 LCD, sedangkan variabel DATA\_ADC ditampilkan pada baris 4 LCD. Algoritma subrutin LCD\_DISPLAY ditunjukkan dalam Gambar 4.24.

Algoritma subrutin LCD\_DISPLAY diawali dengan inisialisasi awal variabel-variabel yang ada pada program. Sebelum data ditampilkan ke LCD, karakter yang ada pada layar LCD perlu dibersihkan terlebih dahulu melalui perintah *lcd\_clear()* dan *pointer* pada LCD perlu diarahkan terlebih dahulu ke koordinat yang dituju melalui perintah *lcd\_gotoxy(x, y)*, dimana *x* adalah kolom LCD dan *y* adalah baris LCD.

Data variabel dari hasil pembacaan sensor ditampilkan ke LCD 4X16 karakter dimana baris 1 diisi oleh variabel HEIGHT (cm) dan RAISE (cm/s), baris 2 diisi oleh variabel TEMP (°C), baris 3 diisi oleh variabel HUMI (%RH), sedangkan baris 4 diisi oleh variabel DATA\_ADC (%).



Gambar 4.24 Algoritma Subrutin LCD\_DISPLAY

#### 4.3.8 Perancangan Subrutin TRANS\_CONTROL

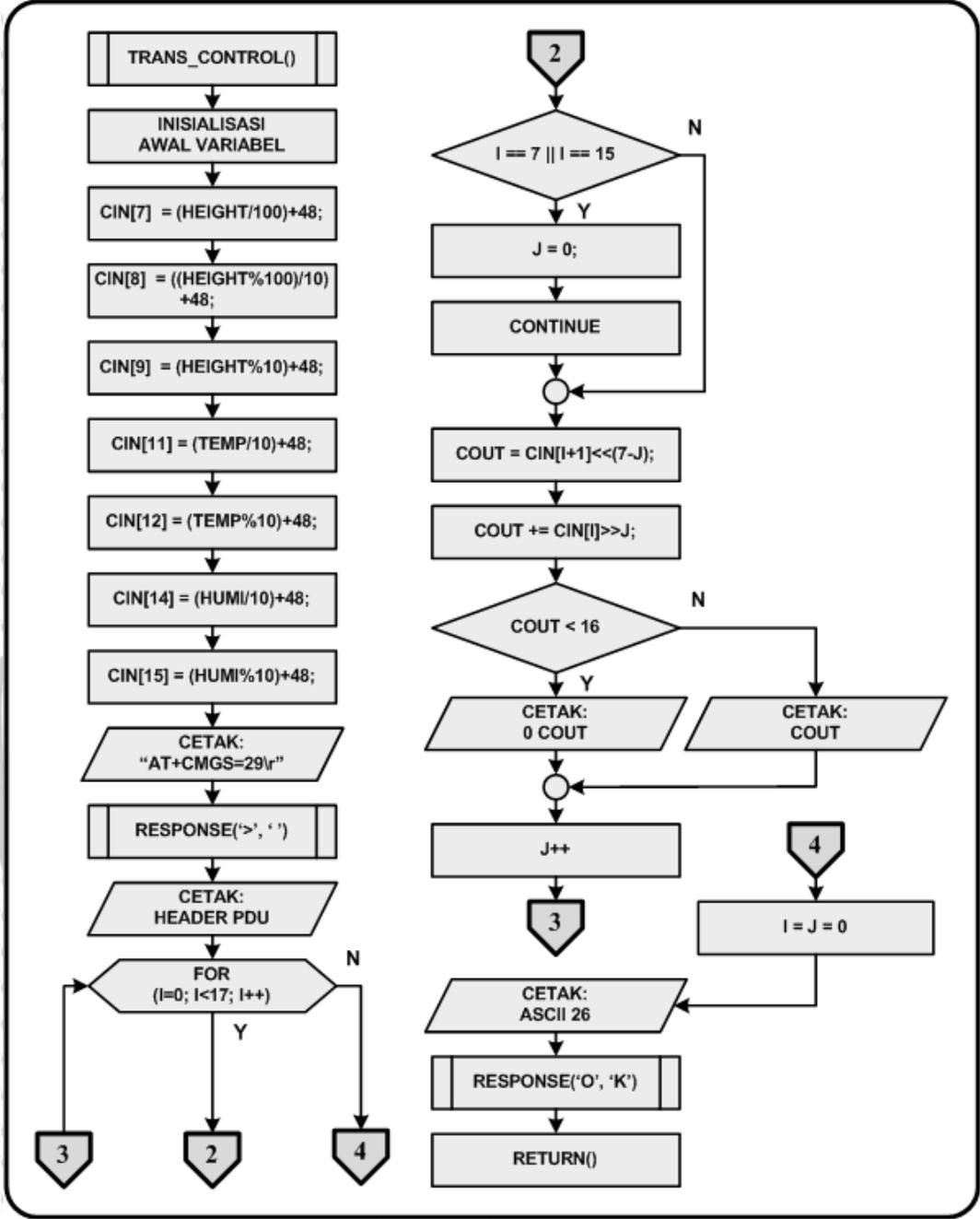
*Handphone* SIEMENS C45 sebagai media transmisi mengirimkan data melalui SMS (*Short Message Service*) dengan format data PDU (*Protocol Data Unit*). Mikrokontroler ATMega8 berkomunikasi secara serial dengan *handphone* SIEMENS C45 untuk memberikan perintah-perintah transmisi dalam *AT-Command*. Transmisi data melalui SMS oleh RTU (*Remote Telemetry Unit*) ditujukan ke *Server* dengan nomor *handphone* sebagai berikut :

**Nomor Handphone Server Dalam Format Internasional : +6285755337973**

Jumlah data yang akan dikirim ke *Server* tersusun atas 17 karakter. Data kirim sejumlah 17 karakter tersebut harus terlebih dahulu dikonversi ke dalam format PDU agar dapat dikirim melalui SMS oleh *handphone* SIEMENS C45. Algoritma subrutin TRANS\_CONTROL ditunjukkan dalam Gambar 4.25.

Algoritma subrutin TRANS\_CONTROL diawali dengan proses inisialisasi awal terhadap nilai-nilai variabel yang ada dalam program dan dilanjutkan dengan partisi terhadap masing-masing bilangan yang ada dalam variabel HEIGHT, TEMP, dan HUMI menjadi masing-masing karakter angka penyusunnya (misalkan bilangan seratus dua puluh tiga, tersusun atas karakter angka ‘1’, ‘2’, dan ‘3’).





Gambar 4.25 Algoritma Subrutin TRANS\_CONTROL

Proses partisi dilakukan dengan cara membagi atau memoduluskan bilangan-bilangan yang ada pada variabel HEIGHT, TEMP, dan HUMI dengan 100 untuk mendapatkan angka ratusan, dengan 10 untuk mendapatkan angka puluhan, dan sisa operasinya yang menyatakan angka satuan.

Untuk memperoleh karakter ASCII dari masing-masing angka hasil partisi, maka angka hasil partisi tersebut dijumlahkan dengan 48. Misalkan angka 1 memiliki nilai karakter ASCII '1' sebesar 49. Karakter hasil partisi disimpan dalam array CIN.



Variabel CIN[0], CIN[1], CIN[2], CIN[3], dan CIN[4] menyimpan karakter penyusun *string* “Data\*”. Variabel CIN[5] menyimpan karakter penyusun kode RTU. Variabel CIN[7], CIN[8], dan CIN[9] menyimpan karakter penyusun variabel HEIGHT. Variabel CIN[11], CIN[12] menyimpan karakter penyusun variabel TEMP. Variabel CIN[14], CIN[15] menyimpan karakter penyusun variabel HUMI. Misalkan data dikirim dari RTU dengan kode RTU 1 dimana ketinggian air 320 cm, suhu 28°C, kelembaban 77%RH, maka hasil *mapping* karakternya ditunjukkan dalam Tabel 4.3.

Tabel 4.3 *Mapping* Karakter pada Data Kirim RTU

<b>Array Variabel</b>	CIN[0]	CIN[1]	CIN[2]	CIN[3]	CIN[4]
Identifier Data	D	a	t	a	*
<b>Array Variabel</b>	CIN[5]	CIN[6]			
Kode RTU	1	*			
<b>Array Variabel</b>	CIN[7]	CIN[8]	CIN[9]	CIN[10]	
HEIGHT	3	2	0	*	
<b>Array Variabel</b>	CIN[11]	CIN[12]	CIN[13]		
TEMP	2	8	*		
<b>Array Variabel</b>	CIN[14]	CIN[15]	CIN[16]		
HUMI	7	7	*		

Setelah partisi selesai, maka subrutin dilanjutkan dengan mengirimkan *string* “AT+CMGS = 29” untuk menginstruksikan pengiriman SMS. Selanjutnya karakter “>” dan ” ” dikirimkan ke subrutin RESPONSE untuk dicek masukan sebagai respon atas *AT-Comand*. Jika *handphone* SIEMENS C45 telah siap menerima oktet PDU, maka subrutin akan mengirimkan *header* oktet PDU sebagai berikut :

**0791265805000F021000D91265857357379F300011**

*Header* PDU berisi data-data yang diperlukan untuk mengirim SMS ke suatu nomor *handphone* yang dituju, yaitu nomor SMS *center*, tipe SMS, nomor referensi SMS, nomor *handphone* penerima (nomor *handphone Server*), bentuk SMS, skema *encoding* data I/O, dan jangka waktu SMS (jangka waktu SMS tidak dibatasi) yang dinyatakan dalam format oktet PDU.

Ke 17 karakter hasil partisi yang telah tersimpan pada *array* variabel CIN tersebut selanjutnya akan dikonversi ke format oktet PDU dengan cara menggeser bit pada variabel CIN[I+1] ke kiri sejauh (7-J) kemudian dijumlahkan dengan hasil penggeseran bit pada variabel CIN[I] ke kanan sejauh (J) dan hasil akhirnya disimpan ke variabel COUT dengan lebar data data 8 bit.

Jika COUT bernilai kurang dari 16 maka subrutin akan mencetak heksa dari variabel COUT diawali dengan karakter '0' (misalkan nilai 10 memiliki heksa 0A). Jika COUT lebih dari 16 maka subrutin akan mencetak keseluruhan karakter heksa dari variabel COUT (misalkan nilai 17 memiliki heksa 11).

Setelah selesai maka variabel J akan di *increment*. Proses konversi ke format oktet PDU diatas akan diulang sebanyak 17 kali dengan catatan untuk nilai I sebesar 7 dan nilai J sebesar 15, proses akan di lewati (*skip*) dengan menggunakan perintah *continue* dan variabel J diisi nilai 0.

Setelah proses konversi ke format oktet PDU selesai, maka barisan oktet PDU tersebut akan dikirimkan ke *handphone* SIEMENS C45. Kemudian subrutin akan mengirimkan karakter ASCII dengan nilai 26 untuk memberi tanda ke *handphone* SIEMENS C45 bahwa seluruh kode oktet PDU telah dikirimkan.

Sebagai penanda bahwa operasi pengiriman SMS telah berhasil, maka *handphone* SIEMENS C45 akan mengirimkan *string* "OK". Pengecekan terhadap *string* "OK" dilakukan dengan mengirimkan karakter 'O' dan 'K' ke subrutin RESPONSE. Setelah subrutin selesai maka program akan dikembalikan ke fungsi pemanggilnya.

**4.3.9 Perancangan Subrutin REPORT\_CHECK**

Subrutin REPORT\_CHECK berfungsi sebagai pemeriksa SMS *report* atas transmisi data terakhir ke *Server*. Jika SMS *report* telah diterima maka sistem akan kembali melanjutkan operasinya, sebaliknya jika SMS *report* belum diterima selama waktu tunggu 30 detik maka transmisi akan diulang dengan *update* data terbaru.

Setiap kali SMS *report* diterima, maka akan terdapat 64 karakter yang akan menyusun PDU dari SMS *report* tersebut. Diantara ke 64 karakter tersebut terdapat 3 bagian yang akan selalu memiliki nilai PDU yang sama, yaitu PDU SMS *Center*, PDU nomor tujuan, dan status dari SMS. Konfigurasi ke 64 karakter dengan 3 bagian blok PDU yang selalu sama ditunjukkan sebagai berikut :

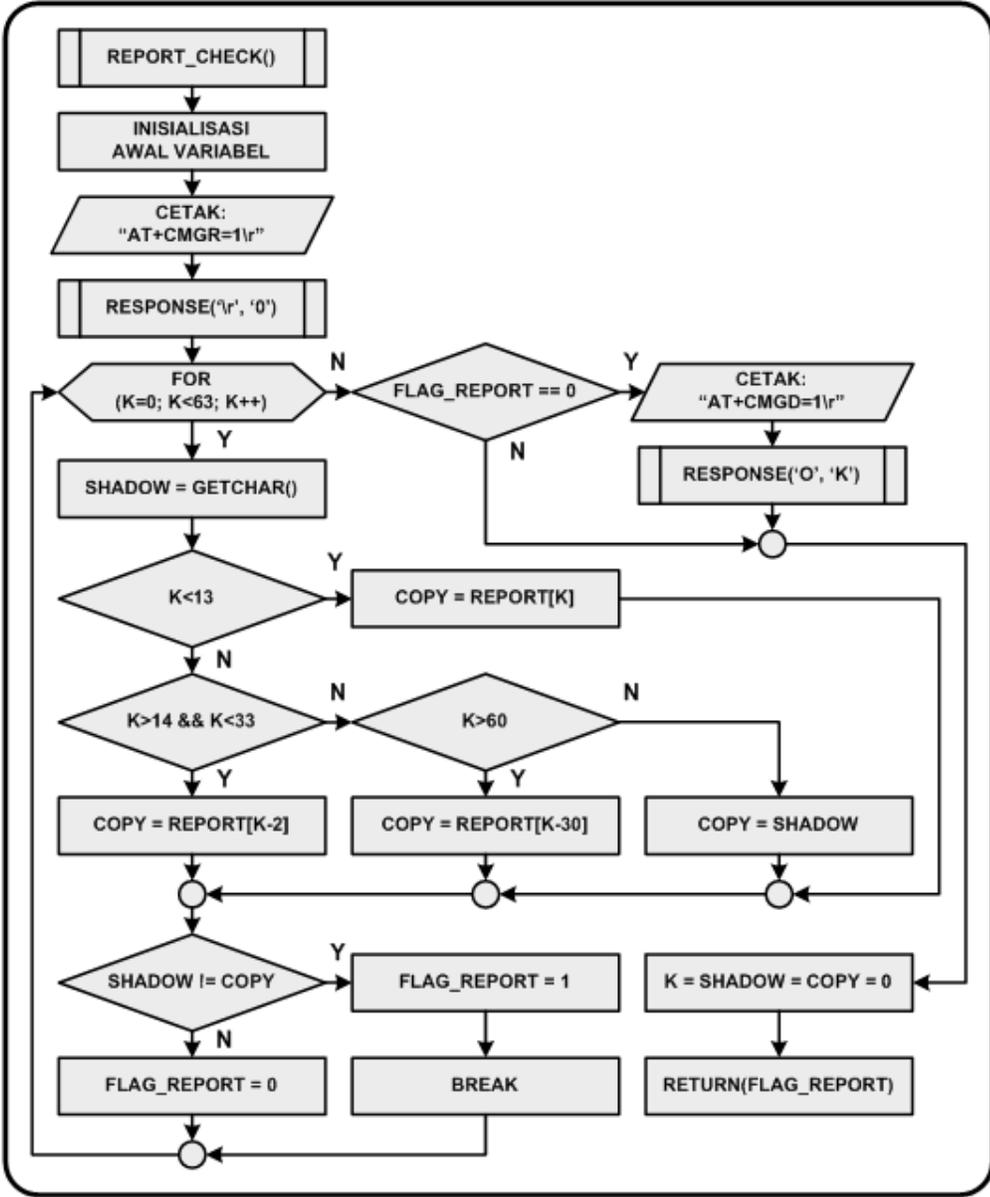
```
05912618164206XX0D91265857357379F3XXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXX00
```

Bagian yang tertulis merah merupakan PDU dari SMS *Center* dengan nomor +62816124, PDU dari nomor tujuan yaitu +6285755337973, dan PDU dari status SMS (00 berarti SMS telah terkirim) yang selalu bernilai sama.



Fungsi dari subrutin REPORT\_CHECK adalah mendeteksi eksistensi 3 bagian blok PDU yang selalu memiliki nilai sama (PDU dari SMS Center, PDU dari nomor tujuan, dan PDU dari status SMS) pada tiap SMS report yang masuk. Proses pembacaan PDU dari SMS report dilakukan dengan mengirimkan string AT-Command "AT+CMGR = 1" ke handphone SIEMENS C45.

Apabila report terdeteksi maka subrutin REPORT\_CHECK akan mengembalikan nilai 0, sebaliknya apabila report belum terdeteksi maka subrutin REPORT\_CHECK akan mengembalikan nilai 1. Algoritma subrutin REPORT\_CHECK ditunjukkan dalam Gambar 4.26.



Gambar 4.26 Algoritma Subrutin REPORT\_CHECK



Algoritma subrutin REPORT\_CHECK diawali dengan inialisasi awal atas nilai variabel yang ada dalam program, kemudian dilanjutkan dengan mengirimkan *string* "AT+CMGR = 1" yang menandakan perintah untuk membaca SMS pada *handphone* SIEMENS C45 pada memori dengan indeks 1. Proses dilanjutkan dengan pendeteksian karakter '0' dan '\r' melalui subrutin RESPONSE. Jika karakter '0' dan '\r' telah terdeteksi maka proses dilanjutkan dengan membaca karakter berikutnya yang masuk.

Pembacaan karakter masuk dilakukan melalui instruksi *getchar()*, kemudian hasil pembacaan disimpan ke variabel SHADOW dan dilanjutkan dengan seleksi terhadap nilai variabel K. Variabel K menentukan iterasi dan proses komparasi karakter yang masuk terhadap *string* REPORT yang berkapasitas 34 karakter. Susunan *string* REPORT ditunjukkan sebagai berikut :

59126181642060D91265857357379F300

Selama variabel K bernilai kurang dari 13, maka karakter yang sedang dibaca berada pada bagian PDU SMS *Center* sehingga nilai variabel COPY diisi sama dengan nilai karakter REPORT[K]. Selama variabel K berada pada rentang 14 hingga 33, maka karakter yang sedang dibaca berada pada bagian PDU SMS nomor tujuan sehingga nilai variabel COPY diisi sama dengan nilai karakter REPORT[K-2]. Selama variabel K bernilai lebih dari 60, maka karakter yang sedang dibaca berada pada bagian status dari SMS sehingga nilai variabel COPY diisi sama dengan nilai karakter REPORT[K-30]. Diluar seleksi kondisi yang ada, maka nilai variabel COPY diisi sama dengan nilai karakter yang terbaca oleh instruksi *getchar()*.

Jika nilai variabel SHADOW dan COPY sama, maka proses akan diteruskan hingga iterasi ke 68 dengan nilai FLAG\_REPORT = 0, jika tidak maka proses akan keluar dari iterasi dengan nilai FLAG\_REPORT = 0. Jika pada akhir iterasi FLAG\_REPORT bernilai 0, maka *AT-Command* yang berupa *string* "AT+CMGD = 1" dikirimkan untuk menghapus SMS *report* pada memori dengan indeks 1 dan dilanjutkan dengan pendeteksian respon *string* "OK" yang dilakukan dengan mengirimkan karakter 'O' dan 'K' ke subrutin RESPONSE. Setelah *string* "OK" terdeteksi, maka proses dilanjutkan dengan mengisikan nilai 0 pada variabel COPY, SHADOW, dan K. Pada akhir proses, subrutin akan mengembalikan nilai FLAG\_REPORT ke program pemanggilnya. Jika operasi berhasil maka nilai yang akan dikembalikan adalah 0, sebaliknya jika operasi gagal maka nilai yang akan dikembalikan adalah 1.

## BAB V PENGUJIAN DAN ANALISIS

Pengujian dan analisis dilakukan untuk mengetahui apakah sistem telah bekerja sesuai perancangan. Pengujian dilakukan per blok kemudian secara keseluruhan. Adapun pengujian yang perlu dilakukan sebagai berikut :

- 1). Pengujian Konverter AC ke DC.
- 2). Pengujian *Charger* Baterai *Automatic*.
- 3). Pengujian Rangkaian *Switching*.
- 4). Pengujian Sensor Ultrasonik SRF02.
- 5). Pengujian Sensor Suhu-Kelembaban SHT11.
- 6). Pengujian LCD 4X16 Karakter.
- 7). Pengujian Transmisi Data Melalui SMS.
- 8). Pengujian Sistem Secara Keseluruhan.

### 5.1 Pengujian Rangkaian Konverter AC ke DC

Pengujian rangkaian konverter AC ke DC bertujuan untuk mengetahui kesesuaian tegangan keluaran dari konverter AC ke DC sebesar DC 12 V untuk masukan tegangan AC 220V dari jala-jala PLN. Diagram blok pengujian rangkaian konverter AC ke DC ditunjukkan dalam Gambar 5.1.

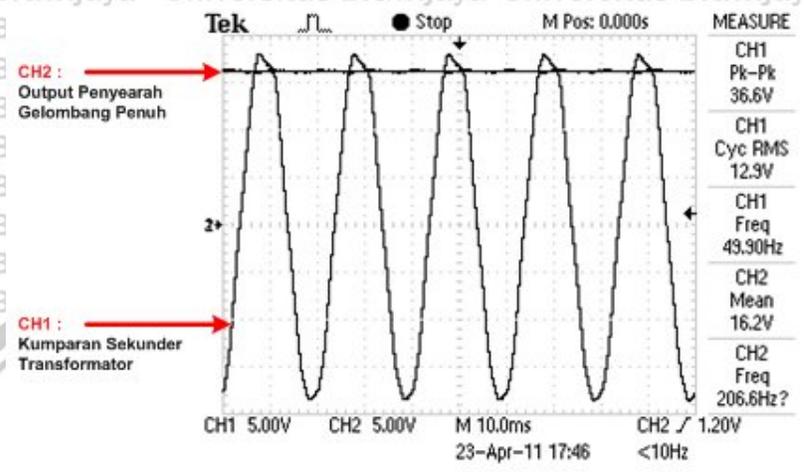


Gambar 5.1 Diagram Blok Pengujian Rangkaian Konverter AC ke DC

Pada tahap pertama akan diuji tegangan dan tegangan keluaran dari penyearah gelombang, sedangkan pada tahap ke dua akan diuji tegangan keluaran dari *fixed output regulator* 7812 (tegangan DC) pada rangkaian konverter AC ke DC.

Pengujian dilakukan dengan menggunakan osiloskop TEKTRONIX TDS-1012B. Pengujian pertama dilakukan untuk mengamati tegangan masukan dan keluaran penyearah gelombang penuh. *Channel* 1 osiloskop dihubungkan dengan kumparan sekunder transformator sebagai masukan ke penyearah gelombang penuh, sedangkan *channel* 2 osiloskop dihubungkan dengan keluaran penyearah gelombang penuh.

Hasil yang diharapkan adalah penyearah gelombang penuh mampu menyearahkan tegangan AC dari kumparan sekunder transformator menjadi tegangan DC dengan nilai minimum 14,6 V (tegangan minimum untuk mengaktifkan *charger* dan *fixed output regulator* 7812). Hasil pengujian tegangan masukan dan keluaran pada penyearah gelombang penuh ditunjukkan dalam Gambar 5.2.



Gambar 5.2 Tegangan Masukan dan Keluaran pada Penyearah Gelombang Penuh

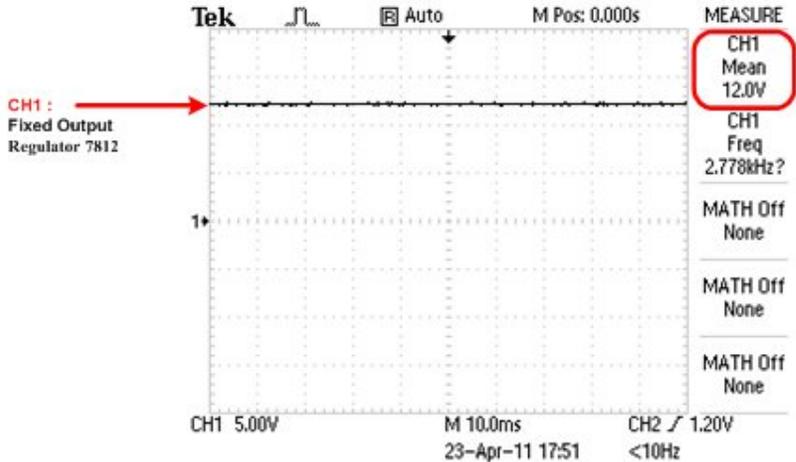
Hasil pengujian pertama menunjukkan bahwa transformator CT *step down* menurunkan tegangan AC 220 V menjadi tegangan AC 12,9 V (36,6 Vpp) dengan frekuensi yang sama yaitu 49,9 Hz. Tegangan AC sebesar 12,9 V ini merupakan nilai efektif untuk tegangan masukan penyearah gelombang penuh sehingga keluaran penyearah gelombang penuh bernilai tegangan DC 16,2 V.

Tegangan DC sebesar 16,2 V ini menjadi tegangan masukan bagi *fixed output regulator* 7812. Efek sinusoida 50 Hz berhasil diredam oleh filter kapasitor dengan nilai kapasitansi sebesar 6800  $\mu$ F sehingga tegangan *ripple* pada keluaran penyearah gelombang penuh tidak begitu nampak.

Pengujian yang kedua dilakukan terhadap *fixed output regulator* 7812, pengujian ini ditujukan untuk mengamati tegangan keluaran *fixed output regulator* 7812 pada rangkaian konverter AC ke DC. Pengujian dilakukan dengan menggunakan osiloskop TEKTRONIX TDS-1012B. *Channel* 1 osiloskop dihubungkan dengan keluaran *fixed output regulator* 7812.

Hasil yang diharapkan adalah rangkaian *fixed output regulator* 7812 mampu memberikan tegangan suplai sebesar 12 V ( $V_{CC}$ ) sebagai sumber tegangan utama untuk mengoperasikan RTU (*Remote Telemetry Unit*). Hasil pengujian tegangan keluaran pada *fixed output regulator* 7812 ditunjukkan dalam Gambar 5.3.





Gambar 5.3 Tegangan Keluaran Pada *Fixed Output Regulator* 7812

Hasil pengujian kedua menunjukkan bahwa rangkaian *fixed output regulator* 7812 berhasil meregulasi tegangan inputnya sebesar DC 16,2 V menjadi tegangan kerja RTU (*Remote Telemetry Unit*) sebesar DC 12 V dengan tegangan *ripple* yang minimum (kurang dari 3 Vpp) dalam keadaan beban penuh.

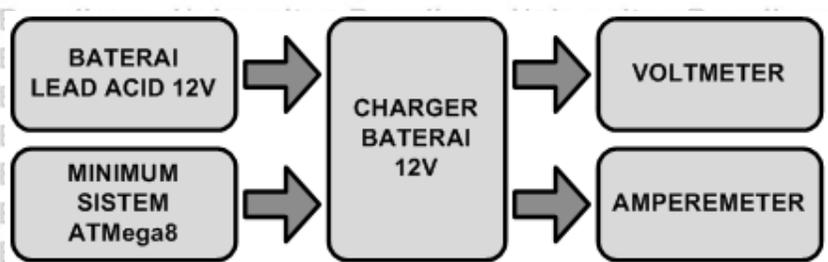
Tabel 5.1 Hasil Pengujian Rangkaian Konverter AC ke DC

	V <sub>AC</sub> Primer	V <sub>AC</sub> Sekunder	V <sub>+</sub>	V <sub>CC</sub>
<b>Pengukuran</b>	220 V	12,9 V	16,2 V	12 V
<b>Perancangan</b>	220 V	12 V	15,5 V	12 V
<b>Error</b>	0%	7,5%	4,52%	0%

Hasil pengujian rangkaian konverter AC ke DC ditunjukkan dalam Tabel 5.1. Tegangan keluaran *fixed output regulator* 7812 sebagai hasil akhir dari rangkaian konverter AC ke DC memiliki nilai yang sesuai dengan nilai perancangan yaitu 12 V.

### 5.2 Pengujian *Charger* Baterai *Automatic*

Pengujian *charger* baterai *automatic* bertujuan untuk mengetahui kinerja serta lamanya durasi *charging* oleh *charger* baterai *automatic*. Diagram blok pengujian *charger* baterai *automatic* ditunjukkan dalam Gambar 5.4.



Gambar 5.4 Diagram Blok Pengujian *Charger* Baterai *Automatic*



Pengujian dilakukan dengan menggunakan dua jenis multimeter yaitu multimeter digital dengan tipe PROTEK 506 yang difungsikan sebagai voltmeter untuk mengukur tegangan baterai dan multimeter analog dengan tipe ABB MA3E yang difungsikan sebagai amperemeter untuk mengukur arus *charging*.



**VOLTMETER : 12,05V**



**AMPEREMETER : 250mA**

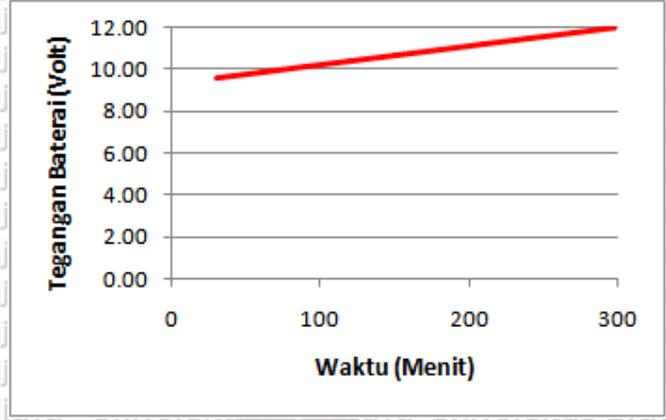
Gambar 5.5 Hasil Pengujian *Charger* Baterai *Automatic*

Hasil yang diharapkan adalah *charger* baterai *automatic* mampu mendeteksi tegangan baterai dibawah 80% kondisi idealnya kemudian mengisi kembali energi pada baterai *lead acid* 12 V hingga optimal, dan memutus proses *charging*. Hasil pengujian *charger* baterai *automatic* ditunjukkan dalam Gambar 5.5. *Track* hasil pengujian *charger* baterai *automatic* ditunjukkan dalam Tabel 5.2.

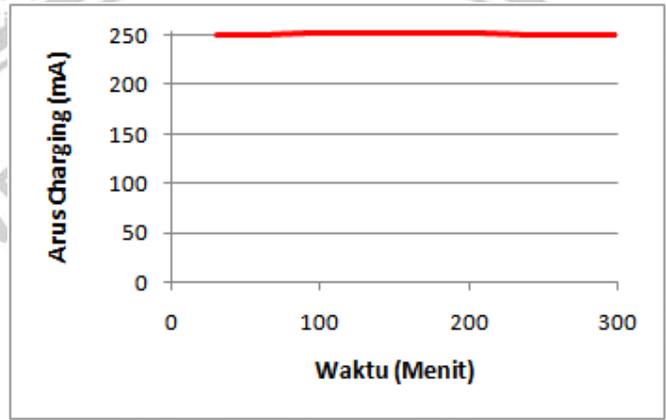
Tabel 5.2 *Track* Hasil Pengujian *Charger* Baterai *Automatic*

Waktu (Menit)	Data Kapasitas Baterai (%)	Tegangan Baterai (V)	Arus Charging (mA)
30 Menit ke-1	80,25	9,63	250
30 Menit ke-2	82,17	9.86	251
30 Menit ke-3	84,42	10,13	252
30 Menit ke-4	86,67	10,4	252
30 Menit ke-5	88,67	10,64	253
30 Menit ke-6	91,08	10,93	252
30 Menit ke-7	93,33	11,20	252
30 Menit ke-8	95,75	11,49	251
30 Menit ke-9	97,92	11,75	250
30 Menit ke-10	100	12,05	250

*Track* hasil pengujian *charger* baterai *automatic* pada Tabel 5.2 menunjukkan bahwa *charger* baterai *automatic* dapat bekerja dengan baik dalam melakukan *charging* terhadap baterai *lead acid* 1200 mAH dengan tegangan awal 9,34 V, dan memerlukan waktu *charging* selama kurang lebih 5 jam hingga tegangan pada baterai dapat dipulihkan dan mendekati kondisi optimalnya yaitu 12 V. Grafik tegangan baterai terhadap waktu ditunjukkan dalam Gambar 5.6, sedangkan grafik arus *charging* terhadap waktu ditunjukkan dalam Gambar 5.7.



Gambar 5.6 Grafik Tegangan Baterai Terhadap Waktu



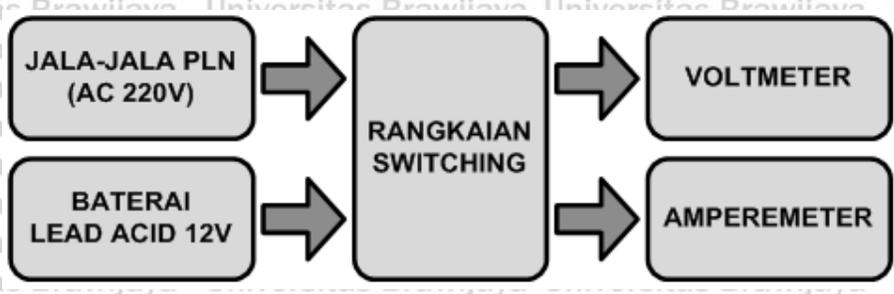
Gambar 5.7 Grafik Arus Charging Baterai Terhadap Waktu

### 5.3 Pengujian Rangkaian Switching

Pengujian rangkaian *switching* bertujuan untuk mengetahui kinerja rangkaian *switching* dalam memindahkan peran sumber daya utama sistem RTU (*Remote Telemetry Unit*) dari jala-jala PLN (AC 220 V) ke baterai (12 V) ataupun sebaliknya.

Pengujian dilakukan dengan cara mengukur arus pada kumparan *relay* serta mengukur tegangan pada jalur distribusi arus ke beban pada rangkaian *switching*.

Diagram blok pengujian rangkaian *switching* ditunjukkan dalam Gambar 5.8.



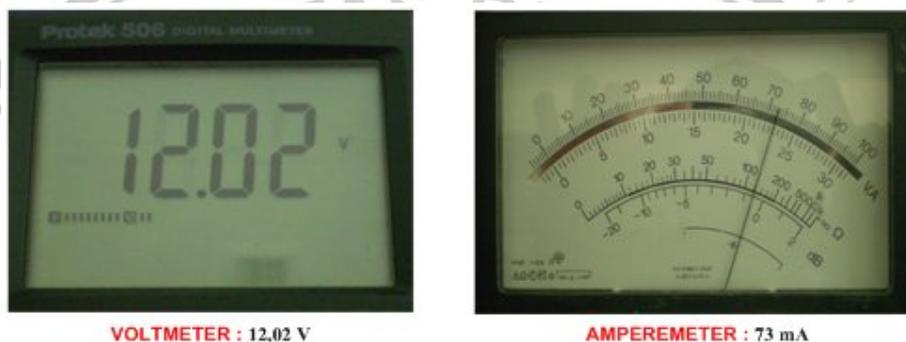
Gambar 5.8 Diagram Blok Pengujian Rangkaian Switching



Pengujian dilakukan dengan menggunakan multimeter digital tipe PROTEK 506 yang difungsikan sebagai voltmeter dan multimeter analog tipe ABB MA3E yang difungsikan sebagai amperemeter.

Pengujian pertama dilakukan untuk mengamati tegangan suplai ke beban serta arus kumparan pada *relay* untuk mode AC. Voltmeter dihubungkan paralel dengan jalur distribusi arus dari rangkaian *switching* ke beban, sedangkan amperemeter dihubungkan seri terhadap kumparan *relay* dan  $V_{CC}$ .

Hasil yang diharapkan adalah rangkaian *switching* mampu mendistribusikan tegangan dari  $V_{CC}$  ke beban serta pada kumparan *relay* dialiri arus yang cukup untuk menggerakkan koilnya. Hasil pengujian tegangan suplai beban dan arus kumparan *relay* pada rangkaian *switching* mode AC ditunjukkan dalam Gambar 5.9.



Gambar 5.9 Hasil Pengujian Tegangan Suplai Beban dan Arus Kumparan *Relay* Mode AC

Hasil pengujian pertama menunjukkan bahwa untuk mode AC, beban akan menerima tegangan suplai sebesar 12,02 V ( $V_{CC}$ ). Tegangan sebesar 12,02 V ini berasal dari keluaran *fixed output regulator* 7812 pada konverter AC ke DC dan digunakan untuk mencatu *relay*, sistem mikrokontroler, sensor, dan LCD. Nilai tegangan sebesar 12,02 V tersebut cenderung stabil selama regulator 7812 dalam kondisi baik.

Arus yang melewati kumparan *relay* sebesar 73 mA sudah cukup untuk mengaktifkan *relay* sebagai pengkondisi *switching*. *Relay* dapat bekerja dengan arus minimum sebesar 72,5 mA, dengan arus kumparan *relay* yang bernilai sebesar 73 mA akan menjamin transistor 2N2222 bekerja dalam mode saturasi dan *cut off*. Aktifnya *relay* ditandai dengan menyalnya LED sebagai *relay* indikator.

Pengujian yang kedua dilakukan untuk mengamati tegangan suplai ke beban serta arus kumparan pada *relay* untuk mode baterai. Voltmeter dihubungkan paralel dengan jalur distribusi arus dari rangkaian *switching* ke beban, sedangkan amperemeter dihubungkan seri terhadap kumparan *relay* dan  $V_{CC}$ .

Hasil yang diharapkan adalah rangkaian *switching* mampu mendistribusikan tegangan dari baterai ( $V_{BATT}$ ) ke beban serta pada kumparan *relay* tidak dialiri arus (*relay* dalam keadaan OFF). Hasil pengujian tegangan suplai beban dan arus kumparan *relay* pada rangkaian *switching* mode baterai ditunjukkan dalam Gambar 5.10.



**VOLTMETER : 11,68 V**



**AMPEREMETER : 0 A**

Gambar 5.10 Hasil Pengujian Tegangan Suplai Beban dan Arus Kumparan *Relay* Mode Baterai

Hasil pengujian kedua menunjukkan bahwa untuk mode baterai, beban akan menerima tegangan suplai dari baterai dengan nilai sebesar 11,68 V ( $V_{BATT}$ ). Tegangan sebesar 11,68 V ini secara langsung berasal dari kedua terminal baterai dan digunakan untuk mencatu sistem mikrokontroler, sensor, dan LCD. Tegangan tersebut akan berubah seiring berkurangnya energi yang tersimpan pada baterai.

Sedangkan arus yang melewati kumparan *relay* sebesar 0 A, hal ini menandakan bahwa *relay* dalam keadaan OFF (tidak ada arus yang melewati kumparan *relay*) sehingga koil *relay* akan menghubungkan  $V_{BATT}$  dengan beban. Pada mode baterai, beban akan menerima tegangan sesuai dengan tegangan terminal baterai. Aktifnya mode baterai ditandai dengan menyalnya LED sebagai baterai indikator.

Ada atau tidaknya tegangan suplai ( $V_{cc}$ ) ke beban pada mode AC ditandai dengan menyala atau tidaknya indikator tegangan suplai ( $V_{cc}$ ) berupa LED (*Light Emitting Diode*) yang terpasang pada *power board*. Hasil pengujian rangkaian *switching* ditunjukkan dalam Tabel 5.3.

Tabel 5.3 Hasil Pengujian Rangkaian *Switching*

Komparasi	Mode Kerja	Tegangan Suplai Beban	Arus Kumparan <i>Relay</i>	LED Indikator	Baterai Indikator
Pengukuran	Mode AC	12,02 V	73 mA	ON	OFF
Perancangan		12 V	72,5 mA		
Error		0,16%	0,68%		
Pengukuran	Mode Baterai	11,68 V	0 A	OFF	ON
Perancangan		12 V	0 A		
Error		2,67%	0%		

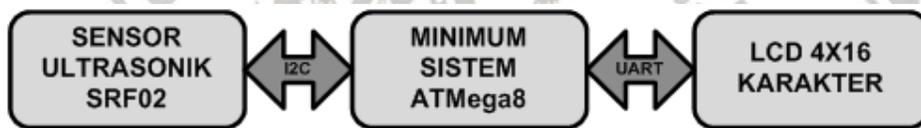
Pada mode AC, arus kumparan pada *relay* menyimpang 0,5 mA (0,68%), kesalahan sebesar 0,68% ini masih dapat ditoleransi, sebab dengan arus kumparan sebesar 73 mA *driver relay self bias* dijamin saturasi dan *relay* berada dalam keadaan aktif untuk memindahkan koilnya.

Pada mode baterai, tegangan suplai dari rangkaian *switching* ke beban menyimpang 0,32 V (2,67%) dari perancangannya sebesar 12 V. Kesalahan sebesar 2,67% ini terjadi akibat tegangan suplai akan mengikuti tegangan terminal baterai.

Mode AC dan baterai diindikasikan oleh menyalnya LED dan baterai indikator.

#### 5.4 Pengujian Sensor Ultrasonik SRF02

Pengujian sensor ultrasonik SRF02 bertujuan untuk mengetahui kemampuan serta ketepatan sensor ultrasonik SRF02 dalam mengukur ketinggian terhadap permukaan benda cair (air). Diagram blok pengujian sensor ultrasonik SRF02 ditunjukkan dalam Gambar 5.11.



Gambar 5.11 Diagram Blok Pengujian Sensor Ultrasonik SRF02

Hasil yang diharapkan adalah data ketinggian permukaan air yang diberikan oleh sensor ultrasonik SRF02 memiliki nilai kesalahan (*error* dalam %) yang minimum saat dibandingkan dengan hasil pembacaan besaran ketinggian permukaan air dari mistar ukur. Hasil pengujian sensor ultrasonik SRF02 ditunjukkan dalam Gambar 5.12.



Gambar 5.12 Hasil Pengujian Sensor Ultrasonik SRF02

Hasil pengujian sensor ultrasonik SRF02 dalam Gambar 5.12 menunjukkan bahwa pada saat pembacaan ketinggian permukaan air sebesar 30 cm, sensor ultrasonik SRF02 memiliki akurasi yang sama besar dengan mistar ukur.

Pembacaan kecepatan perubahan ketinggian air sebesar 0 cm/s terjadi akibat tidak adanya perubahan ketinggian permukaan air (air dalam kondisi tenang/stabil) yang terjadi dalam periode sampling sebesar 1 detik.

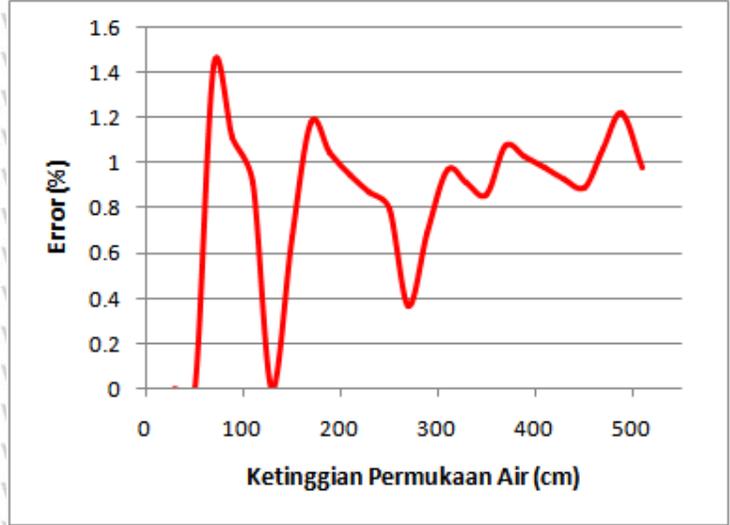
Pengujian sensor ultrasonik SRF02 dilakukan melalui pengambilan masing-masing 25 kali data ketinggian permukaan air untuk beberapa variasi kondisi ketinggian permukaan air. Hasil pengujian pengukuran ketinggian permukaan air oleh sensor SRF02 ditunjukkan dalam Tabel 5.4.

Tabel 5.4 Hasil Pengujian Pengukuran Ketinggian Permukaan Air Sensor SRF02

No.	Data Ketinggian Air SRF02 (cm)	Data Ketinggian Air Mistar Ukur (cm)	Error
1	30	30	0.00
2	50	50	0.00
3	69	70	1.43
4	89	90	1.11
5	109	110	0.91
6	130	130	0.00
7	149	150	0.67
8	168	170	1.18
9	188	190	1.05
10	208	210	0.95
11	228	230	0.87
12	248	250	0.80
13	269	270	0.37
14	288	290	0.69
15	307	310	0.97
16	327	330	0.91
17	347	350	0.86
18	366	370	1.08
19	386	390	1.03
20	406	410	0.98
21	426	430	0.93
22	446	450	0.89
23	465	470	1.06
24	484	490	1.22
25	505	510	0.98
<b>Rata-Rata Kesalahan</b>			<b>0.84</b>

Dari 25 variasi kondisi pengujian pengukuran ketinggian permukaan air sensor SRF02, nilai rata-rata kesalahan pada pengukuran ketinggian permukaan air oleh sensor ultrasonik SRF02 sebesar 0,84%. Grafik *error* pengukuran ketinggian permukaan air terhadap waktu ditunjukkan dalam Gambar 5.13.

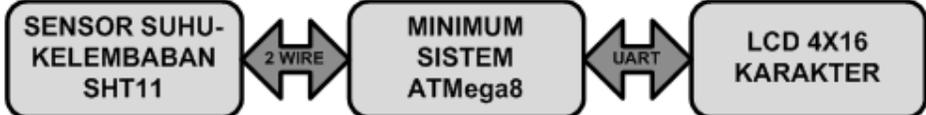




Gambar 5.13 Grafik Error Pengukuran Ketinggian Permukaan Air Terhadap Waktu

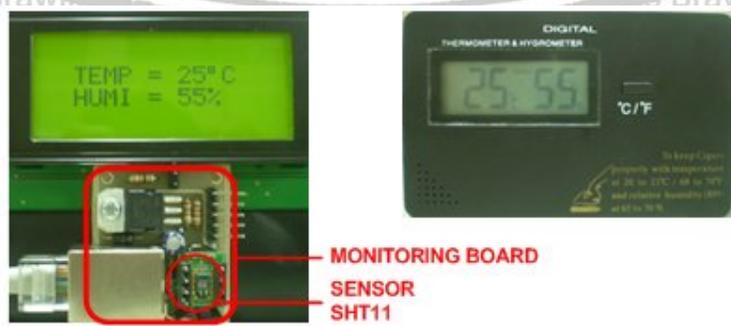
### 5.5 Pengujian Sensor Suhu-Kelembaban SHT11

Pengujian sensor suhu-kelembaban SHT11 bertujuan untuk mengetahui ketepatan sensor SHT11 dalam mengukur suhu serta kelembaban udara. Diagram blok pengujian sensor suhu-kelembaban SHT11 ditunjukkan dalam Gambar 5.14.



Gambar 5.14 Diagram Blok Pengujian Sensor Suhu-Kelembaban SHT11

Hasil yang diharapkan adalah data suhu-kelembaban dari sensor SHT11 memiliki kesalahan (*error* dalam %) minimum saat dibandingkan dengan pembacaan suhu-kelembaban dari thermometer sekaligus higrometer tipe CSONKA LR44. Hasil pengujian sensor suhu-kelembaban SHT11 ditunjukkan dalam Gambar 5.15.



Pengukuran Suhu - Kelembaban Menggunakan Sensor SHT11

Pengukuran Suhu - Kelembaban Menggunakan Thermometer & Higrometer Digital

Gambar 5.15 Hasil Pengujian Sensor Suhu-Kelembaban SHT11



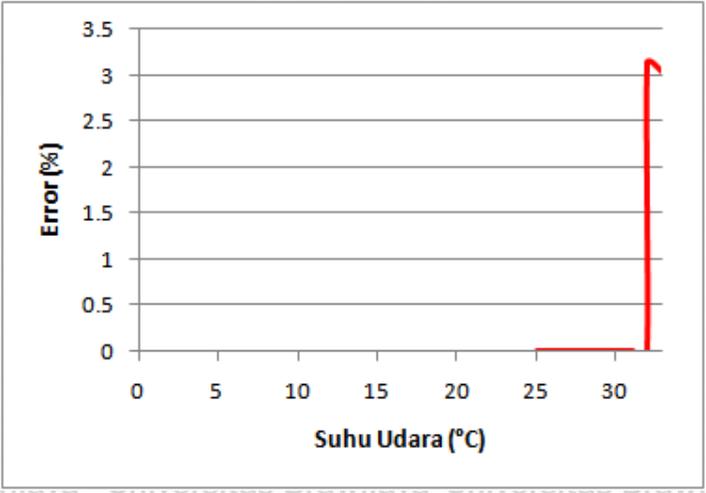
Hasil pengujian sensor suhu-kelembaban SHT11 dalam Gambar 5.15 menunjukkan bahwa pada saat pembacaan suhu udara sebesar 25°C dan kelembaban udara relatif sebesar 55%RH, sensor suhu-kelembaban SHT11 memiliki akurasi yang sama besar dengan thermometer sekaligus higrometer digital.

Pengujian pengukuran suhu oleh sensor suhu-kelembaban SHT11 dilakukan melalui pengambilan masing-masing 10 data suhu udara untuk beberapa variasi kondisi suhu udara. Hasil pengujian pengukuran suhu oleh sensor suhu-kelembaban SHT11 ditunjukkan dalam Tabel 5.5.

Tabel 5.5 Hasil Pengujian Pengukuran Suhu Sensor SHT11

No.	Data Suhu SHT11(°C)	Data Suhu Thermometer Digital (°C)	Error (%)
1	25	25	0
2	26	26	0
3	27	27	0
4	28	28	0
5	29	28	0
6	30	30	0
7	31	31	0
8	32	32	0
9	33	32	3,13
10	34	33	3,03
<b>Rata-Rata Kesalahan</b>			0,616

Error pada pengujian pengukuran suhu oleh sensor suhu-kelembaban SHT11 mulai muncul pada saat pembacaan suhu udara sebesar 32°C dengan nilai rata-rata kesalahan pembacaan suhu udara sebesar 0,616%. Grafik error pengukuran suhu udara terhadap waktu ditunjukkan dalam Gambar 5.16.



Gambar 5.16 Grafik Error Pengukuran Suhu Udara Terhadap Waktu

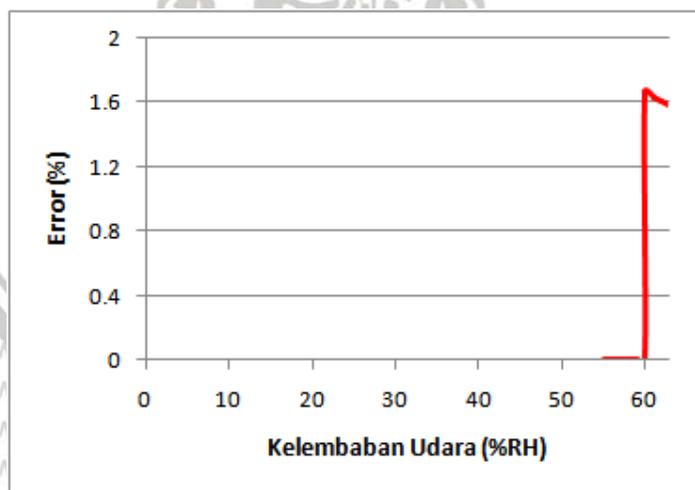


Pengujian pengukuran kelembaban oleh sensor suhu-kelembaban SHT11 dilakukan melalui pengambilan masing-masing 10 data kelembaban udara untuk beberapa variasi kondisi kelembaban udara. Hasil pengujian pengukuran kelembaban oleh sensor suhu-kelembaban SHT11 ditunjukkan dalam Tabel 5.6.

Tabel 5.6 Hasil Pengujian Pengukuran Kelembaban Sensor SHT11

No.	Data Kelembaban SHT11 (%RH)	Data Kelembaban Higrometer Digital (%RH)	Error (%)
1	55	55	0
2	56	56	0
3	57	57	0
4	58	58	0
5	59	59	0
6	60	60	0
7	61	60	1,67
8	62	61	1,63
9	63	62	1,61
10	64	63	1,58
<b>Rata-Rata Kesalahan</b>			0,649

Error pada pengujian pengukuran kelembaban oleh sensor suhu-kelembaban SHT11 mulai muncul pada saat pembacaan kelembaban udara sebesar 60%RH dengan nilai rata-rata kesalahan pembacaan kelembaban udara sebesar 0,649%. Grafik error pengukuran kelembaban udara terhadap waktu ditunjukkan dalam Gambar 5.17.



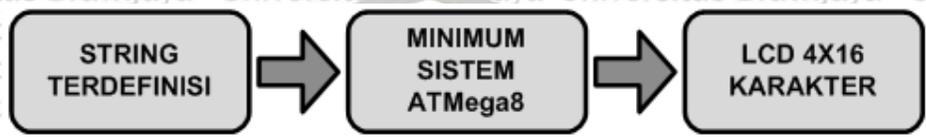
Gambar 5.17 Grafik Error Pengukuran Kelembaban Udara Terhadap Waktu

Hasil pengujian pengukuran suhu oleh sensor SHT11 dalam Tabel 5.5 menunjukkan bahwa rata-rata kesalahan pada pengukuran suhu oleh sensor SHT11 sebesar 0,616%, sedangkan rata-rata kesalahan pada pengukuran kelembaban oleh sensor suhu-kelembaban SHT11 sebesar 0,649%.



### 5.6 Pengujian LCD 4X16 Karakter

Pengujian LCD 4X16 karakter bertujuan untuk mengetahui kesesuaian antara karakter-karakter yang dikirimkan oleh program di dalam mikrokontroler ATmega8 ke LCD dengan karakter yang tertampil pada layar LCD 4X16 karakter. Pengujian dilakukan dengan menuliskan *string* yang terlebih dahulu didefinisikan untuk masing-masing baris pada LCD, kemudian dilanjutkan dengan pengecekan dan identifikasi terhadap *string* yang tertampil pada tiap baris LCD 4X16 karakter. Diagram blok pengujian LCD 4X16 karakter ditunjukkan dalam Gambar 5.18.



Gambar 5.18 Diagram Blok Pengujian LCD 4X16 Karakter

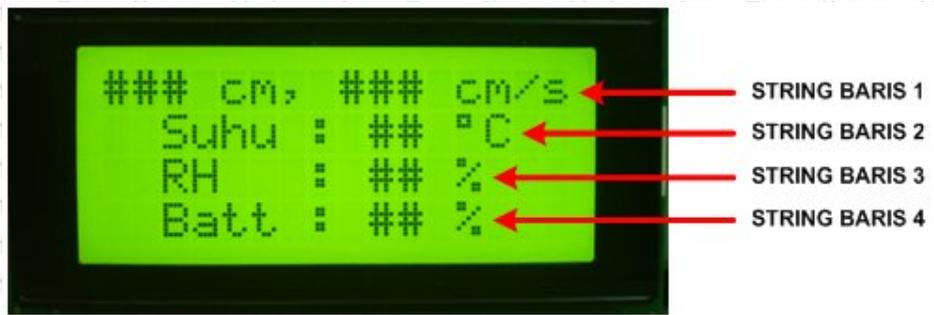
Pengujian dilakukan dengan menggunakan minimum sistem mikrokontroler ATmega8 sebagai pengirim data *string*, dan LCD 4X16 karakter dengan tipe 1604A sebagai modul penampil. Komposisi *string* terdefinisi yang akan ditampilkan pada tiap baris LCD 4X16 karakter ditunjukkan dalam Gambar 5.19.

```

    ### cm,   ### cm/s
    Suhu   : ## °C
    RH     : ## %
    Batt   : ## %
  
```

Gambar 5.19 Komposisi *String* Terdefinisi pada Baris LCD 4X16 Karakter

Hasil yang diharapkan adalah *string* yang tertampil pada modul penampil LCD 4X16 karakter memiliki nilai dan komposisi karakter yang identik dengan *string* yang sebelumnya telah terdefiniskan dalam Gambar 5.19. Hasil pengujian LCD 4X16 karakter ditunjukkan dalam Gambar 5.20.



Gambar 5.20 Hasil Pengujian LCD 4X16 Karakter



Hasil pengujian LCD 4X16 karakter dalam Gambar 5.20 menunjukkan bahwa *string* yang tertampil pada tiap baris LCD 4X16 karakter memiliki komposisi karakter yang sama persis dengan *string* yang sebelumnya telah terdefiniskan dalam Gambar 5.19. Hasil pengujian LCD 4X16 karakter menunjukkan bahwa LCD 4X16 karakter beserta rangkaian pengkondisinya dapat berfungsi dengan baik.

### 5.7 Pengujian Transmisi Data Melalui SMS

Pengujian transmisi data melalui SMS bertujuan untuk mengetahui kesesuaian informasi yang dikirimkan oleh RTU (*Remote Telemetry Unit*) sebagai *transmitter* data dengan data yang diterima oleh *Server* sebagai *receiver* data.

Pengujian transmisi data melalui SMS dilakukan dalam tiga tahap. Pada tahap pertama akan diuji rangkaian antarmuka SIEMENS C45 ke mikrokontroler ATmega8, pada tahap kedua akan diuji komunikasi serial dua arah antara mikrokontroler ATmega8 dengan PC (*Personal Computer*), pada tahap ketiga akan diuji pengiriman SMS melalui SIEMENS C45 secara *AT Command* melalui komunikasi serial.

#### 5.7.1 Pengujian Rangkaian Antarmuka SIEMENS C45

Pengujian rangkaian antarmuka SIEMENS C45 bertujuan untuk mengetahui kesesuaian level tegangan yang dikeluarkan oleh rangkaian antarmuka SIEMENS C45 dengan level tegangan yang dapat diterima oleh *handphone* SIEMENS C45 dimana nilai tegangan tersebut berkisar 0 hingga 3,6 V. Diagram blok pengujian rangkaian antarmuka SIEMENS C45 ditunjukkan dalam Gambar 5.21.

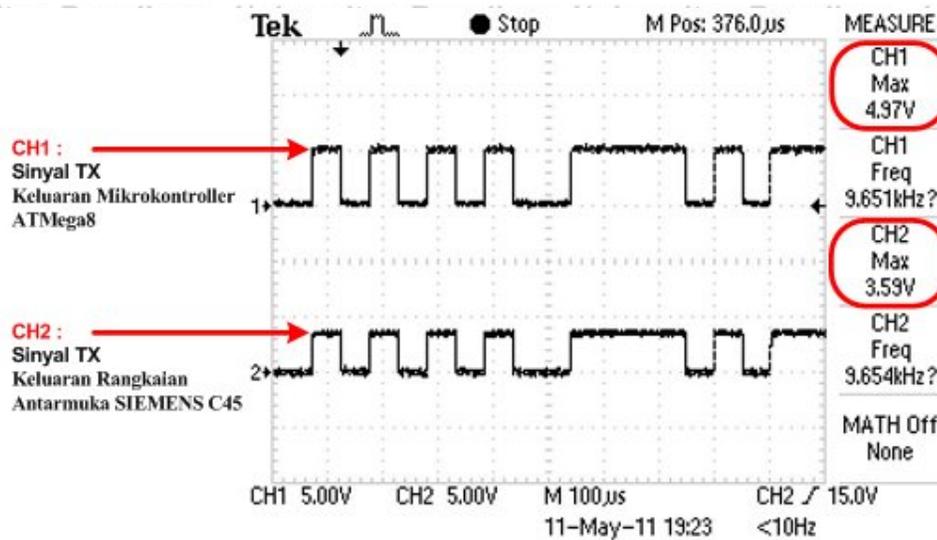


Gambar 5.21 Diagram Blok Pengujian Rangkaian Antarmuka SIEMENS C45

Pengujian dilakukan dengan menggunakan minimum sistem ATmega8 sebagai pengirim data serial, rangkaian antarmuka SIEMENS C45 sebagai pengkondisi level tegangan untuk *handphone* SIEMENS C45, dan osiloskop TEKTRONIX TDS-1012B sebagai penampil level tegangan untuk data serial.

*Channel* 1 osiloskop dihubungkan dengan PIN TX pada mikrokontroler ATmega8, sedangkan *channel* 2 osiloskop dihubungkan dengan PIN TX pada keluaran rangkaian antarmuka SIEMENS C45. *Ground* dari masing-masing minimum sistem ATmega8, rangkaian antarmuka SIEMENS C45, dan osiloskop saling terhubung.

Hasil yang diharapkan adalah level tegangan serial keluaran ATmega8 berada pada level  $-0,5 \text{ V} - 1 \text{ V}$  ( $0,2 \cdot V_{CC}$ ) untuk logika 0 dan berada pada level tegangan  $3 \text{ V}$  ( $0,6 \cdot V_{CC}$ ) -  $5,5 \text{ V}$  ( $V_{CC} + 0,5 \text{ V}$ ) untuk logika 1, sedangkan level tegangan serial keluaran rangkaian antarmuka SIEMENS C45 berada pada level tegangan  $0 - 0,4 \text{ V}$  untuk logika 0 dan berada pada level tegangan  $3,5 \text{ V} - 3,6 \text{ V}$  untuk logika 1. Hasil pengujian rangkaian antarmuka SIEMENS C45 ditunjukkan dalam Gambar 5.22.



Gambar 5.22 Hasil Pengujian Rangkaian Antarmuka SIEMENS C45

Hasil pengujian rangkaian antarmuka SIEMENS C45 dalam Gambar 5.22 menunjukkan bahwa level tegangan serial yang dikeluarkan oleh mikrokontroler ATmega8 berada pada level tegangan 0 untuk logika 0 dan berada pada level tegangan 4,97 V untuk logika 1. Level tegangan serial keluaran mikrokontroler ATmega8 berada pada level TTL (*Transistor Transistor Logic*) sesuai hasil yang diharapkan.

Level tegangan serial keluaran rangkaian antarmuka SIEMENS C45 berada pada level tegangan 0 untuk logika 0 dan berada pada level tegangan 3,59 V untuk logika 1 serta memiliki lebar pulsa pada *timing* yang sama dengan data serial pada level TTL yang dikeluarkan oleh mikrokontroler ATmega8.

### 5.7.2 Pengujian Komunikasi Serial Dua Arah

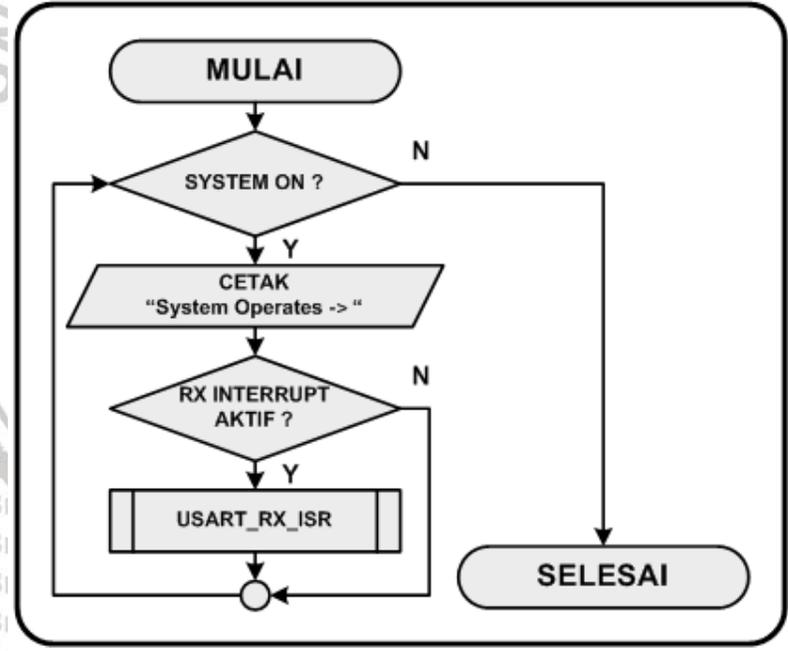
Pengujian komunikasi serial dua arah bertujuan untuk mengetahui fungsionalitas dari komunikasi serial yang dilakukan dalam dua arah oleh mikrokontroler ATmega8 bersama perangkat luar yang juga mendukung komunikasi serial dua arah, baik pada mode *transmit* ataupun *receive*. Diagram blok pengujian komunikasi serial dua arah ditunjukkan dalam Gambar 5.23.



Gambar 5.23 Diagram Blok Pengujian Komunikasi Serial Dua Arah

Pengujian dilakukan dengan menggunakan minimum sistem ATmega8 sebagai *responder* data serial, IC *level shifter* MAX 232 sebagai pengkondisi level tegangan TTL ke RS232 atau sebaliknya, PC (*Personal Computer*) dengan PORT serial sebagai pengumpan data serial, dan *software* CVAVR sebagai serial terminal.

Hasil yang diharapkan adalah mikrokontroler ATmega8 mampu mengirimkan *string* "System Operates -> " jika tidak terdeteksi input serial dari PC. Ketika mikrokontroler ATmega8 mendeteksi masukan berupa karakter angka, maka mikrokontroler ATmega8 akan mengirimkan kembali *echo* dari karakter angka ke serial terminal CVAVR sebanyak tiga kali. Algoritma program pengujian komunikasi serial dua arah ditunjukkan dalam Gambar 5.24.

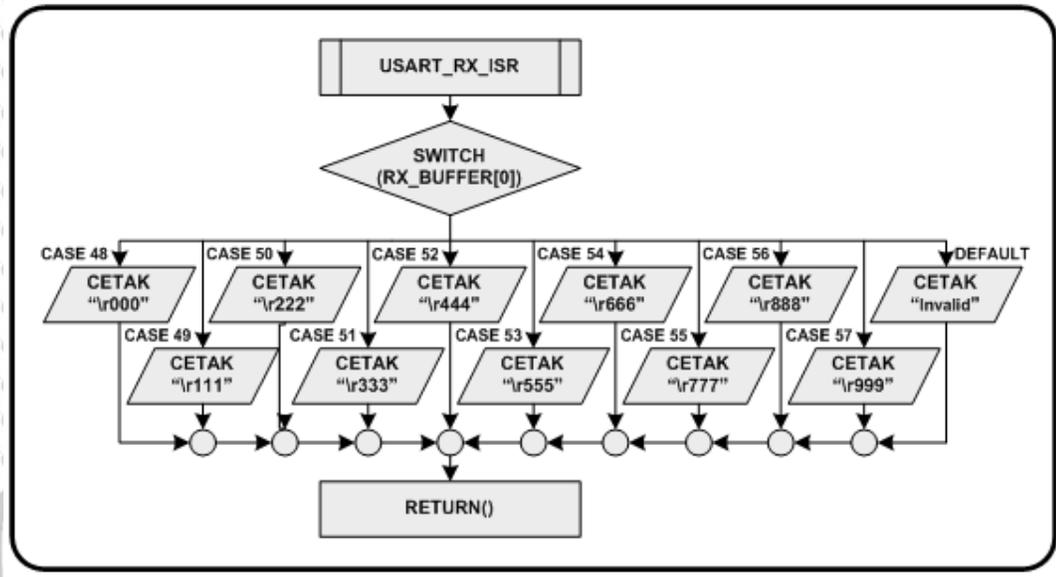


Gambar 5.24 Algoritma Program Pengujian Komunikasi Serial Dua Arah

Saat sistem diaktifkan, maka mikrokontroler ATmega8 akan mengirimkan *string* "System Operates ->" ke serial terminal CVAVR. Program akan memeriksa apakah ada masukan data serial dari PC. Jika ada data masukan serial dari PC maka RX INTERRUPT akan aktif sehingga program akan memanggil subrutin USART\_RX\_ISR. Algoritma subrutin USART\_RX\_ISR ditunjukkan dalam Gambar 5.25.

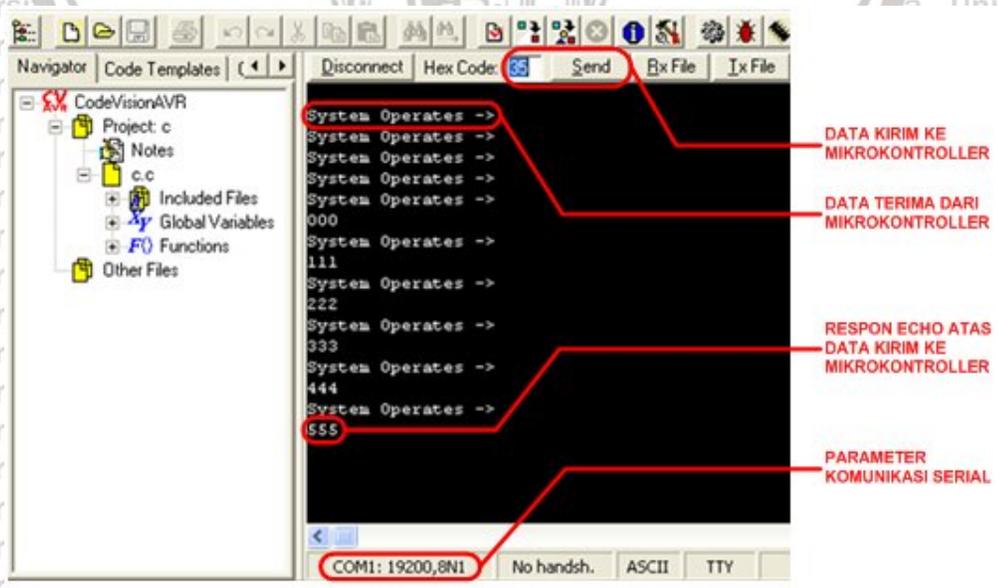


Tanpa adanya data masukan serial dari PC, *string* "System Operates ->" akan terus dikirim. Ketika mendeteksi masukan berupa karakter angka dari PC, maka mikrokontroler ATmega8 akan mengirimkan kembali *echo* dari karakter angka tersebut sebanyak 3 kali dengan diawali karakter *carriage return* (*\r*).



Gambar 5.25 Algoritma Subrutin USART\_RX\_ISR

Pengiriman karakter angka dari serial terminal CVAVR ke mikrokontroler ATmega8 dilakukan dengan memasukkan kode heksa dari karakter angka yang dimaksudkan ke *hex code bar* pada serial terminal CVAVR. Hasil pengujian komunikasi serial dua arah ditunjukkan dalam Gambar 5.26.



Gambar 5.26 Hasil Pengujian Komunikasi Serial Dua Arah



Hasil pengujian komunikasi serial dua arah dalam Gambar 5.26 menunjukkan bahwa komunikasi serial yang terjadi menggunakan parameter komunikasi serial 8 data, 1 stop, *no parity*, dengan *baudrate* sebesar 19200 bps, dimana data serial masuk ke PC dari COM1 (PORT 1 komunikasi serial pada PC).

Pada keadaan tanpa input dari PC, mikrokontroller ATmega8 akan terus menerus mengirimkan *string* "System Operates" ke PC. Saat *hex code* dengan nilai 35H (karakter '5') dikirimkan dari *hex code bar* pada serial terminal CVAVR, maka mikrokontroller ATmega8 akan balas mengirim *echo* nya yaitu *string* "555". Hasil pengujian komunikasi serial dua arah ditunjukkan dalam Tabel 5.7.

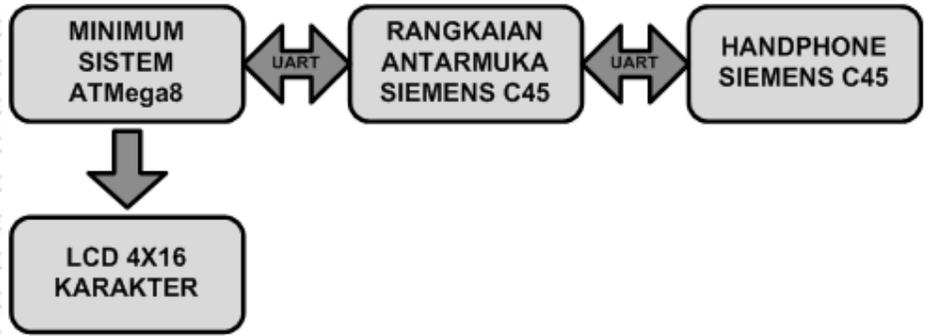
Tabel 5.7 Hasil Pengujian Komunikasi Serial Dua Arah

Data Kirim Dari Serial PC (Heksa)	Data Echo Dari Serial Mikrokontroller (String)
-	"System Operates -> "
48	"\r000"
49	"\r111"
50	"\r222"
51	"\r333"
52	"\r444"
53	"\r555"

Hasil pengujian dalam Tabel 5.7 menunjukkan bahwa telah terjalin komunikasi serial dua arah yang baik antara mikrokontroller ATmega8 dan PC dengan parameter komunikasi serial 8 data, 1 stop, *no parity*, serta *baudrate* sebesar 19200 bps.

**5.7.3 Pengujian Pengiriman SMS Melalui SIEMENS C45**

Pengujian pengiriman SMS melalui SIEMENS C45 bertujuan untuk mengetahui keaslian data yang dikirim oleh RTU dengan data yang diterima oleh *Server*, serta untuk mengetahui kinerja *report detection* atas SMS yang dikirim ke *Server*. Diagram blok pengujian pengiriman SMS melalui SIEMENS C45 ditunjukkan dalam Gambar 5.27.

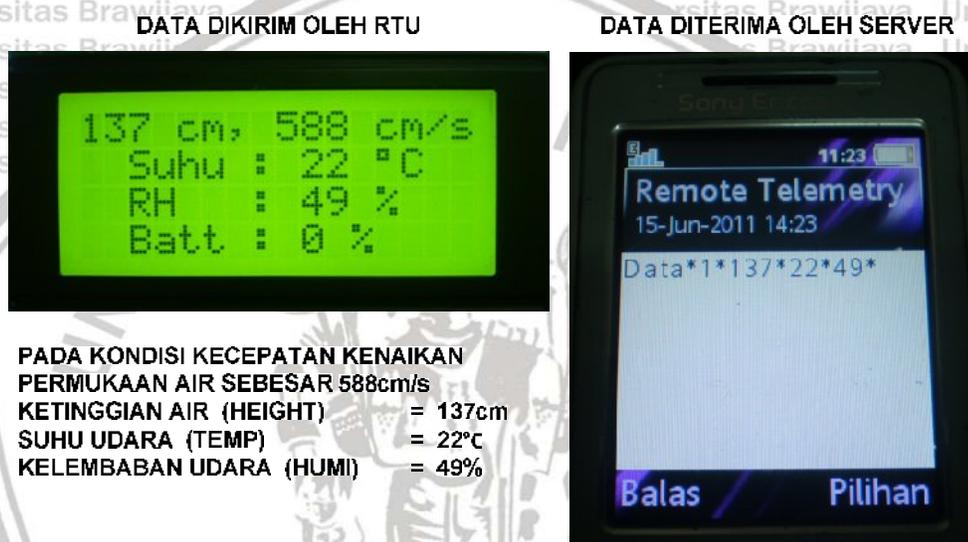


Gambar 5.27 Diagram Blok Pengujian Pengiriman SMS Melalui SIEMENS C45



Pengujian dilakukan dengan menggunakan minimum sistem ATmega8 sebagai pengirim data, rangkaian antarmuka SIEMENS C45 untuk mengkondisikan level tegangan mikrokontroler ATmega8 dengan SIEMENS C45, *handphone* SIEMENS C45 sebagai sarana pengiriman SMS, serta LCD 4X16 karakter sebagai penampil. Data HEIGHT, RAISE, TEMP, dan HUMI diambil acak menggunakan fungsi *rand()*.

Hasil yang diharapkan adalah data SMS yang dikirimkan oleh mikrokontroler ATmega8 detik dengan data SMS yang terima oleh *Server*, serta sistem dapat mendeteksi masuknya SMS *report* atas transmisi data terakhir. Hasil pengujian pengiriman SMS melalui SIEMENS C45 ditunjukkan dalam Gambar 5.28.



Gambar 5.28 Hasil Pengujian Pengiriman SMS Melalui SIEMENS C45

Hasil pengujian pengiriman SMS melalui SIEMENS C45 dalam Gambar 5.28 menunjukkan bahwa data yang akan dikirimkan ke *Server* adalah HEIGHT = 137 cm, TEMP = 22°C, dan HUMI = 49%RH, dalam keadaan kecepatan kenaikan permukaan air RAISE = 588 cm/s. Nilai RAISE sebesar 588 cm/s akan mempengaruhi durasi pengiriman data dan menyebabkan data kembali dikirim dalam durasi 94 detik kemudian (DURATION = 1800-(2,9. 588)). Hasil pengujian pengiriman SMS melalui SIEMENS C45 ditunjukkan dalam Tabel 5.8.

Hasil pengujian pengiriman SMS melalui SIEMENS C45 dalam Gambar 5.28 menunjukkan bahwa seluruh data dari variabel HEIGHT, TEMP, dan HUMI ditransmisikan via SMS ke *Server* oleh mikrokontroler ATmega8 sesuai dengan *string* format *Data\*HEIGHT\*TEMP\*HUMI\**, dimana interval pengiriman data telah mengikuti nilai variabel DURATION pada tingkat kevalidan data 100%.



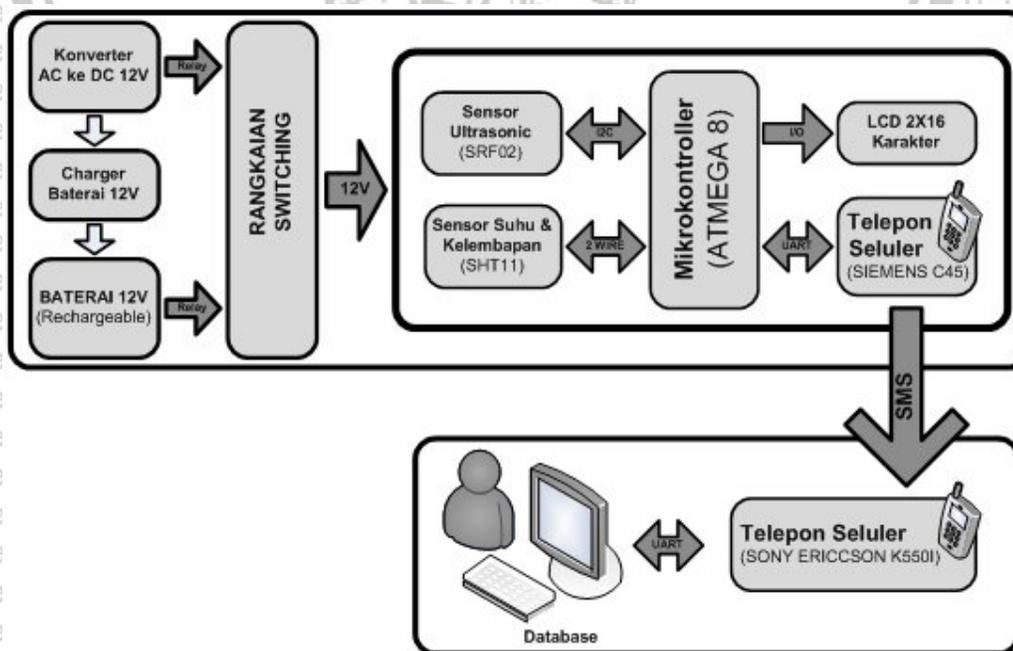
Tabel 5.8 Hasil Pengujian Pengiriman SMS Melalui SIEMENS C45

WAKTU	DATA KIRIM				DURATION (s)	DATA TERIMA	REPORT
	HEIGHT (cm)	TEMP (°C)	HUMI (%RH)	RAISE (cm/s)			
14.23	137	22	49	588	94	Data*137*22*49*	✓
14.24	312	44	85	570	147	Data*312*44*85*	✓
14.27	294	5	2	591	86	Data*294*05*02*	✓
14.28	232	79	14	568	152	Data*232*79*14*	✓
14.31	167	75	9	581	115	Data*167*75*09*	✓
14.33	301	86	49	566	158	Data*301*86*49*	✓
14.35	338	9	9	551	202	Data*338*09*09*	✓
14.39	217	58	83	557	184	Data*217*58*83*	✓
14.42	28	5	82	580	118	Data*28*05*82*	✓
14.44	122	45	50	573	138	Data*122*45*50*	✓

Report atas masing-masing data yang ditransmisikan oleh mikrokontroler ATmega8 melalui SMS dapat dideteksi dengan baik oleh subrutin REPORT\_CHECK() yang telah terprogram ke dalam mikrokontroler ATmega8, sehingga sistem dapat melakukan transmisi ulang jika SMS report belum terdeteksi selama 30 detik.

### 5.8 Pengujian Sistem Secara Keseluruhan

Pengujian sistem secara keseluruhan bertujuan untuk mengetahui kinerja sistem RTU (*Remote Telemetry Unit*) setelah seluruh bagian-bagian penyusun sistem digabungkan menjadi satu kesatuan utuh. Diagram blok pengujian sistem secara keseluruhan ditunjukkan dalam Gambar 5.29.



Gambar 5.29 Diagram Blok Pengujian Sistem Secara Keseluruhan



Pengujian dilakukan dengan menggunakan sistem RTU (*Remote Telemetry Unit*) sebagai unit pengirim data ke *Server*. Sistem RTU akan beroperasi mengambil data dengan periode sampling tiap 1 detik, data yang akan diambil antara lain ketinggian permukaan air, kecepatan perubahan ketinggian permukaan air, suhu, serta kelembaban udara, disamping itu sistem RTU juga akan mengontrol proses *charging* dengan frekuensi sampling pengambilan data level tegangan baterai tiap 1 detik. Transmisi data melalui SMS ke *Server* dilakukan secara kontinyu berdasarkan data ketinggian permukaan air. Durasi pengiriman data ke *Server* ditentukan berdasarkan fungsi alih :

$$DURATION = 1800 - \left( \frac{58}{19} \cdot HEIGHT \right)$$

Sistem RTU akan melakukan transmisi ulang data dengan *update* terbaru jika SMS *report* tidak terdeteksi dalam jangka waktu tunggu 30 detik.

Hasil yang diharapkan adalah sistem RTU dapat mengambil data-data ketinggian permukaan air, kecepatan perubahan ketinggian permukaan air, suhu, serta kelembaban udara, kemudian melakukan fungsi *charging*, serta melakukan transmisi data ke *Server* dengan durasi pengiriman tertentu. Hasil pengujian sistem secara keseluruhan ditunjukkan dalam Gambar 5.30.

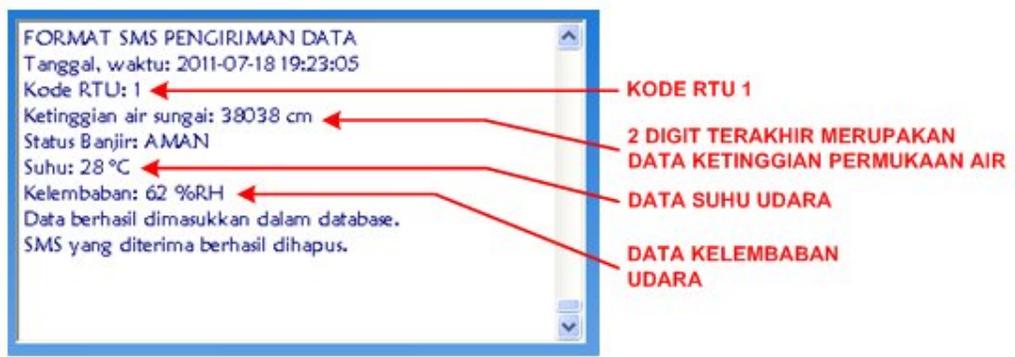


Gambar 5.30 Hasil Pengujian Sistem Secara Keseluruhan

Hasil pengujian sistem secara keseluruhan dalam Gambar 5.30 menunjukkan bahwa hasil pembacaan ketinggian permukaan air dari sensor ultrasonik SRF02 menunjukkan nilai 38 cm, dan tidak terjadi perubahan ketinggian air sehingga nilai kecepatan perubahan air adalah 0. Hasil pembacaan suhu serta kelembaban udara dari sensor suhu-kelembaban SHT11 menunjukkan nilai 28°C dan 62%RH. Level tegangan baterai menunjukkan nilai 97%, sehingga sistem belum perlu melakukan *charging*.

Transmisi data ke *Server* dilakukan dengan menggunakan kode RTU 1.

Pengujian sistem secara keseluruhan dilakukan dengan mengarahkan tujuan transmisi secara langsung ke *Server*, sehingga data-data yang dikirim RTU selama proses uji dapat disimpan oleh *Server*. Data transmisi yang diterima oleh *Server* ditunjukkan dalam Gambar 5.31. *Record* data pada *Server* selama pengujian keseluruhan sistem ditunjukkan dalam Gambar 5.32.



Gambar 5.31 Data Transmisi yang Diterima Oleh *Server*

River Flood Monitoring and Warning System: Tabel Histori Data dan Grafik Semua RTU

Data Base

Bulan: Juli Tahun: 2011 RTU: RTU 1  Tampilkan

No.	Tanggal, Waktu	RTU	Tinggi (cm)	Status	Suhu (°C)	Kelembaban (%RH)
1	7/18/2011 7:07:14 PM	1	38043	AMAN	27	62
2	7/18/2011 7:09:04 PM	1	38074	AMAN	28	65
3	7/18/2011 7:12:39 PM	1	38063	AMAN	28	65
4	7/18/2011 7:15:37 PM	1	38022	AMAN	28	62
5	7/18/2011 7:16:18 PM	1	38022	AMAN	28	62
6	7/18/2011 7:16:58 PM	1	38050	AMAN	28	62
7	7/18/2011 7:19:12 PM	1	38020	AMAN	28	63
8	7/18/2011 7:19:47 PM	1	38043	AMAN	28	62
9	7/18/2011 7:21:38 PM	1	38036	AMAN	28	63
10	7/18/2011 7:23:05 PM	1	38038	AMAN	28	62

Gambar 5.32 *Record* Data pada *Server* Selama Pengujian Keseluruhan Sistem

Berdasarkan *record* data pada *Server* selama pengujian keseluruhan sistem dalam Gambar 5.32, kesepuluh data transmisi yang dikirim oleh RTU melalui SMS seluruhnya dapat diterima dan diolah dengan baik oleh *Server*.

Dari hasil pengujian yang terlihat dalam Gambar 5.30, Gambar 5.31, dan Gambar 5.32 dapat dianalisis bahwa data-data yang dikirimkan RTU (ketinggian permukaan air, suhu udara, dan kelembaban udara) dapat diterima dengan baik oleh *Server*. Hal ini terlihat dari kecocokan antara data yang tertulis di LCD dengan data yang tertulis dalam program aplikasi yang ada di *Server*. Selain itu dalam Gambar 5.30 juga dapat dilihat bahwa data-data yang diterima oleh *Server* telah dapat disimpan ke basis data oleh program aplikasi yang dirancang. Dengan demikian dapat disimpulkan bahwa sistem secara keseluruhan dapat bekerja dengan baik.



## BAB VI KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Berdasarkan hasil perancangan dan pengujian yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut :

- 1). Perancangan sistem elektronika untuk proses akuisisi data ketinggian permukaan air digunakan modul sensor ultrasonik SRF02 yang diakses secara I2C (*Inter Integrated Circuit*). Sedangkan proses akuisisi data suhu-kelembaban udara digunakan modul sensor suhu-kelembaban SHT11 yang diakses secara *2-wire interface*.

Data ketinggian permukaan air yang diukur oleh sensor SRF02 berada pada *range* 30 cm - 600 cm dengan resolusi 1 cm dan rata-rata kesalahan 0,84%.

Data suhu yang diukur oleh sensor SHT11 berada pada *range* 0°C – 100°C dengan resolusi 1°C, dan Data kelembaban yang diukur oleh sensor SHT11 berada pada *range* 0%RH – 100%RH dengan resolusi 1%RH.

- 2). Perancangan rangkaian *switching* antara catu daya AC dari PLN dan baterai terhadap beban dilakukan dengan menggunakan *driver relay* DPDT (*Double Pole Double Throw*) *self bias*. Perancangan rangkaian *charger* baterai dilakukan menggunakan *adjustable voltage regulator* LM317, otomatisasi kerja *charger* baterai diatur oleh mikrokontroler ATmega8.

Rangkaian *switching* dan *charger* baterai *automatic* telah berfungsi dengan arus kumparan *relay* sebesar 73 mA yang menjamin *relay* DPDT untuk melakukan kontak selama proses *switching* ataupun *charging*.

Sensor tegangan pada *charger* adalah rangkaian pembagi tegangan, sehingga bersifat linear dengan perbandingan  $V_{ADC} = V_{BATT} / 3$ .

*Switching* dari catu daya AC ke baterai membutuhkan waktu transfer maksimum sebesar 20 ms.

- 3). Perancangan rangkaian antarmuka sistem mikrokontroler dengan *handphone* SIEMENS C45 dilakukan dengan menggunakan rangkaian pembagi tegangan. Perangkat lunak sistem mikrokontroler telah berfungsi dengan baik dalam mengolah data dari sensor-sensor terpasang serta melakukan transmisi data melalui SMS.

Komunikasi antara mikrokontroler AT Mega8 dan *handphone* SIEMENS

C45 dilakukan dengan menggunakan *AT Command* melalui komunikasi serial dengan *baudrate* 19200 bps.

- 4). Durasi waktu transmisi data valid tercepat untuk ketinggian permukaan air 30 cm adalah 60 detik, sedangkan durasi waktu transmisi data valid terlama untuk ketinggian permukaan air 600 cm adalah 30 menit.
- 5). Secara keseluruhan sistem dapat beroperasi dengan baik sesuai dengan fungsinya pada periode sampling sebesar 1 detik.

## 6.2 Saran

Saran-saran dalam pengimplementasian maupun peningkatan unjuk kerja sistem ini dapat diuraikan sebagai berikut :

- 1). Sistem dapat dikembangkan dengan menambahkan media penyimpan data pada RTU (*Remote Telemetry Unit*), sehingga apabila terjadi kerusakan sistem pada *Server*, data-data yang ada di RTU masih bisa diselamatkan.
- 2). Mengingat daerah operasi sistem yang berupa perairan terbuka, maka sistem dapat dikembangkan dengan menambahkan media proteksi terhadap bahaya petir dan air hujan.

## DAFTAR PUSTAKA

Atmel. 2006. *ATMEGA8/ATMEGA8L, 8-bit AVR with 8 kbytes in System Programmable*

*Flash*. <http://www.atmel.com/literatur/>. Diakses tanggal 5 Januari 2011.

Atmel. 2006. *Interfacing GSM Modems*. <http://www.atmel.com/literatur/>. Diakses tanggal 10 Januari 2011.

Bayong, Tjasyono. 2003. *Geosains*. Bandung: ITB.

Buchman, Isidor. 2003. *Charging the Lead-Acid Battery*. <http://www.batteryuniversity.com/>. Diakses tanggal 10 Januari 2011.

BMKG Karang Ploso. 2011. *Cuaca Malang Terkini*. <http://www.staklimkarangploso.net/>. Diakses tanggal 5 Januari 2011.

Devantech. 2006. *SRF02 Ultrasonic Range Finder*. [http://www.dfrobot.com/Manual/SEN0005\\_Manual\\_10\\_en.pdf/](http://www.dfrobot.com/Manual/SEN0005_Manual_10_en.pdf/). Diakses tanggal 5 Januari 2011.

Glorya Rakhmawaty, Nurussa'adah, Waru Djuriatno, 2009, *Alat Pemantau Tinggi Permukaan Air Sungai Melalui Tampilan Monitor dan Short Message Service (SMS) Berbasis Database System*, Skripsi, Teknik Elektro, UB.

Jerran. 2003. *Switching Relay*. <http://www.electro-tech-online.com/>. Diakses tanggal 5 Januari 2011.

Khairullah. 2009. *Ma'rifatullah Melalui Cuaca dan Gejala Alam*. <http://ustadzklimat.blogspot.com/>. Diakses tanggal 15 Februari 2011.

Malvino, Albert Paul. Gunawan, Hanapi. 1981. *Prinsip-Prinsip Elektronika*. Jakarta: Erlangga.

Omron. 2006. *OMRON PCB Relay*. <http://www.omron.com/ecb/products/>. Diakses tanggal 15 Februari 2011.

Pettersson. 2006. Lars. *SMS and the PDU format*. <http://dreamfabric.com/sms/>. Diakses tanggal 10 Januari 2011.

Philips Semiconductor. 2000. *The I2C-Bus Specification*. <http://www.semiconductors.philips.com/i2c/>. Diakses tanggal 15 Februari 2011.

Rod, Elliot. 2001. *Power Supply Requirements*. <http://www.sound.westhost.com/>.

Diakses tanggal 16 Januari 2011.

Romadhona Shinta, Nurussa'adah, Panca Mudjirahardjo, 2008, *Alat Pengukur Tinggi Muka Air Sungai dengan Pengiriman Data Melalui Short Message Service (SMS)*, Skripsi, Teknik Elektro, UB.

Sensirion. 2008. *Datasheet SHT1x (SHT10, SHT11, SHT15) Humidity and Temperature Sensor*. [http://www.sensirion.com/en/pdf/product\\_information/Datasheet-humidity-sensor-SHT1x.pdf/](http://www.sensirion.com/en/pdf/product_information/Datasheet-humidity-sensor-SHT1x.pdf/). Diakses tanggal 16 Januari 2011.

Siemens. 2000. *AT Command Set Reference Manual for Siemens C45*. [http://jazi.staff.ugm.ac.id/Mobile%20and%20Wireless%20Documents/s35i\\_c35i\\_m35i\\_atc\\_commandset\\_v01.pdf/](http://jazi.staff.ugm.ac.id/Mobile%20and%20Wireless%20Documents/s35i_c35i_m35i_atc_commandset_v01.pdf/). Diakses tanggal 15 Januari 2011.

Smith, Jacqueline. 2006. *The Facts On File Dictionary of Weather and Climate Revised Edition*. New York: Facts On File, Inc.

Wardhana, Lingga. 2006. *Aplikasi Sederhana dengan Mikrokontroler AVR ATmega 8535*. Jogjakarta: Andi.

Waryono dkk. 1987. *Pengantar Meteorologi Dan Klimatologi Untuk Universitas Dan Umum*. Surabaya: Bina Ilmu.



LAMPIRAN I

FOTO ALAT





Gambar 1 Unit Monitoring



Gambar 2 Unit Processing





LAMPIRAN II

GAMBAR SKEMA RANGKAIAN





LAMPIRAN III

LISTING PROGRAM MIKROKONTROLER  
ATMega8



```

#include <mega8.h> // Wait for the AD conversion to complete
#include <stdio.h> while ((ADCSRA & 0x10)==0);
#include <delay.h> ADCSRA|=0x10;
#include <SHT11/sensirion_protocol.h> return ADCH;
}

// I2C Bus functions
#asm // Declare your global variables here
.equ __i2c_port=0x15 ;PORTC bit FLAG_DELETE = 0;
.equ __sda_bit=4 unsigned char TIME_OUT = 0;
.equ __scl_bit=5
#endasm void RESPONSE (unsigned char X, unsigned char Y)
#include <i2c.h> {
while (getchar() != X) {};
while (getchar() != Y) {};
}

// Alphanumeric LCD Module functions
#asm unsigned char TEMP, HUMI;
.equ __lcd_port=0x18 ;PORTB unsigned int HEIGHT;
#endasm int RAISE;
#include <lcd.h> void SENSOR_READ (void)
{
unsigned char L_BYTE, ERROR;
unsigned int H_BYTE, RH_RAW, TEMP_RAW;

i2c_start();
i2c_write(0xE0);
i2c_write(0x00);
i2c_write(0x51);
delay_ms(70);
i2c_stop();

i2c_start();
i2c_write(0xE0);
i2c_write(2);
i2c_stop();

i2c_start();
i2c_write(0xE1);
H_BYTE = i2c_read(1);
L_BYTE = i2c_read(0);
i2c_stop();
RAISE = HEIGHT;
HEIGHT = (H_BYTE<<8)+L_BYTE;
RAISE = HEIGHT-RAISE;
ERROR = s_measure(&TEMP_RAW, 0);
}

// Timer 1 overflow interrupt service routine
bit FLAG_SEC, FLAG_DUR;
unsigned char MILISEC;
unsigned int SEC, DURATION;
interrupt [TIM1_OVF]void timer1_ovf_isr(void)
{
// Reinitialize Timer 1 value
TCNT1H=0xFF;
TCNT1L=0xC8;
// Place your code here
if (MILISEC < 100) {MILISEC++;}
else
{
MILISEC = FLAG_SEC = 0;

if (SEC < DURATION) {SEC++;}
else {SEC = FLAG_DUR = 0;};
};
}

// Read the 8 most significant bits
#define ADC_VREF_TYPE 0x60
unsigned char read_adc(unsigned char adc_input)
{
ADMUX=adc_input|ADC_VREF_TYPE;
// Start the AD conversion
ADCSRA|=0x40;
}

```



```

if (ERROR == 0)
{
    ERROR = s_measure(&RH_RAW, 1);
    TEMP = calc_sht11_temp(TEMP_RAW);
    HUMI = rhcalc_int(RH_RAW);
};
}

unsigned int DATA_ADC;
void CHARGE_CONTROL (void)
{
    DATA_ADC = read_adc(0);

    if (DATA_ADC < 163) {PORTD.7 = 1;}
    else if (DATA_ADC >= 235) {PORTD.7 = 0;};
    DATA_ADC = (DATA_ADC*100)/204;
}

void LCD_DISPLAY(void)
{
    unsigned char LINE_1[16], LINE_2[16], LINE_3[16], LINE_4[16];

    lcd_clear();
    sprintf(LINE_1, "%d cm, %d cm/s", HEIGHT, RAISE);
    sprintf(LINE_2, " Suhu : %d %c ", TEMP, 223);
    sprintf(LINE_3, " RH : %d %c ", HUMI, 37);
    sprintf(LINE_4, " Batt : %d %c ", DATA_ADC, 37);

    lcd_gotoxy(0, 0);
    lcd_puts(LINE_1);
    lcd_gotoxy(0, 1);
    lcd_puts(LINE_2);
    lcd_gotoxy(0, 2);
    lcd_puts(LINE_3);
    lcd_gotoxy(0, 3);
    lcd_puts(LINE_4);
}

unsigned char I, J, CIN[18] = "Data*1*000*00*00*";
void TRANS_CONTROL (void)
{
    unsigned char COUT;

    CIN[7] = HEIGHT/100+48;
    CIN[8] = (HEIGHT%100)/10+48;
    CIN[9] = HEIGHT%10+48;

    CIN[11] = (TEMP/10)+48;
    CIN[12] = (TEMP%10)+48;
    CIN[14] = (HUMI/10)+48;
    CIN[15] = (HUMI%10)+48;

    printf("AT+CMGS=29\r");
    RESPONSE('>', ' ');
    // Header PDU For SERVER :
    printf("07912658050000F021000D91265857357379F3000011");
    for (I=0; I<17; I++)
    {
        if (I == 7 || I == 15) {J = 0; continue;};
        COUT = CIN[(unsigned int)I+1]<<(7-J);
        COUT = COUT+(CIN[I]>>J);
        if (COUT < 16) {printf("0%x", COUT);}
        else {printf("%x", COUT);}
        J++;
    };
    I = J = 0;
    putchar(26);
    RESPONSE('O', 'K');
    delay_us(600);
}

// Header Report For SERVER :
const unsigned char REPORT[34] = "59126181642060D91265857357379F300";
unsigned char K;
unsigned char REPORT_CHECK (void)
{
    unsigned char SHADOW, COPY, FLAG_REPORT;

    printf("AT+CMGR=1\r");
    RESPONSE('\r', 'O');
    for (K=0; K<63; K++)
    {
        SHADOW = getchar();

        if (K<13) {COPY = REPORT[K];}
        else if (K>14 && K<33) {COPY = REPORT[K-2];}
        else if (K>60) {COPY = REPORT[K-30];}
        else {COPY = SHADOW;};
        if (SHADOW != COPY) {FLAG_REPORT = 1; break;}
    }
}

```



```

    else {FLAG_REPORT = 0;};
};
delay_ms(500);
if (FLAG_REPORT == 0)
{
    printf("AT+CMGD=1\r");
    RESPONSE('O', 'K');
    delay_us(600);
};
K = SHADOW = COPY = 0;
return(FLAG_REPORT);
}

void main(void)
{
    // Declare your local variables here

    // Input/Output Ports initialization
    // Port B initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
    Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTB=0x00;
    DDRB=0x00;

    // Port C initialization
    // Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
    // State6=T State5=T State4=T State3=T State2=T State1=T State0=T
    PORTC=0x00;
    DDRC=0x00;

    // Port D initialization
    // Func7=Out Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
    Func0=In
    // State7=0 State6=T State5=T State4=T State3=P State2=P State1=T
    State0=T
    PORTD=0x00;
    DDRD=0x80;

    // Timer/Counter 0 initialization
    // Clock source: System Clock
    // Clock value: Timer 0 Stopped
    TCCR0=0x00;
    TCNT0=0x00;
    // Timer/Counter 1 initialization
    // Clock source: System Clock
    // Clock value: 31.250 kHz
    // Mode: Normal top=FFFFh
    // OC1A output: Discon.
    // OC1B output: Discon.
    // Noise Canceler: Off
    // Input Capture on Falling Edge
    // Timer 1 Overflow Interrupt: On
    // Input Capture Interrupt: Off
    // Compare A Match Interrupt: Off
    // Compare B Match Interrupt: Off
    TCCR1A=0x00;
    TCCR1B=0x04;
    TCNT1H=0xFE;
    TCNT1L=0xC8;
    ICR1H=0x00;
    ICR1L=0x00;
    OCR1AH=0x00;
    OCR1AL=0x00;
    OCR1BH=0x00;
    OCR1BL=0x00;

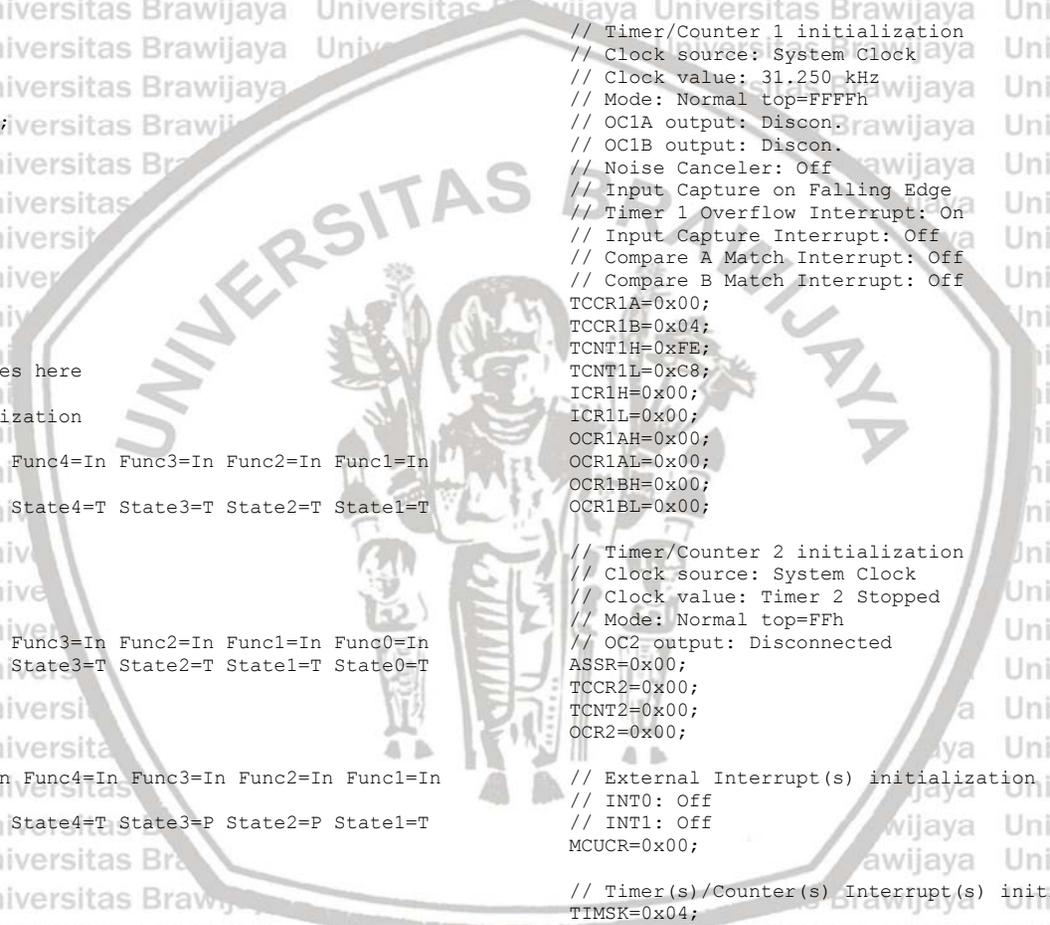
    // Timer/Counter 2 initialization
    // Clock source: System Clock
    // Clock value: Timer 2 Stopped
    // Mode: Normal top=FFh
    // OC2 output: Disconnected
    ASSR=0x00;
    TCCR2=0x00;
    TCNT2=0x00;
    OCR2=0x00;

    // External Interrupt(s) initialization
    // INT0: Off
    // INT1: Off
    MCUCR=0x00;

    // Timer(s)/Counter(s) Interrupt(s) initialization
    TIMSK=0x04;

    // USART initialization
    // Communication Parameters: 8 Data, 1 Stop, No Parity
    // USART Receiver: On
}

```



```

// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud rate: 19200
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x19;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AVCC pin
// Only the 8 most significant bits of
// the AD conversion result are used
ADMUX=ADC_VREF_TYPE;
ADCSRA=0x83;

// I2C Bus initialization
i2c_init();

// LCD module initialization
lcd_init(16);

// Global enable interrupts
#asm("sei")

lcd_clear();
lcd_gotoxy(0, 0);
lcd_putsf("REMOTE TELEMETRY");
lcd_gotoxy(0, 1);
lcd_putsf("UNIT (RTU) ");
lcd_gotoxy(0, 2);
lcd_putsf("-----");
lcd_gotoxy(0, 3);
lcd_putsf(" (CODE : 1) ");
delay_ms(2000);

lcd_clear();
lcd_gotoxy(0, 1);
lcd_putsf("PHONE CHECKING ");
        lcd_gotoxy(0, 2);
        lcd_putsf("-----");
        printf("ATE0\r");
        RESPONSE('O', 'K');
        delay_ms(1000);
        printf("AT\r");
        RESPONSE('O', 'K');
        delay_ms(1000);

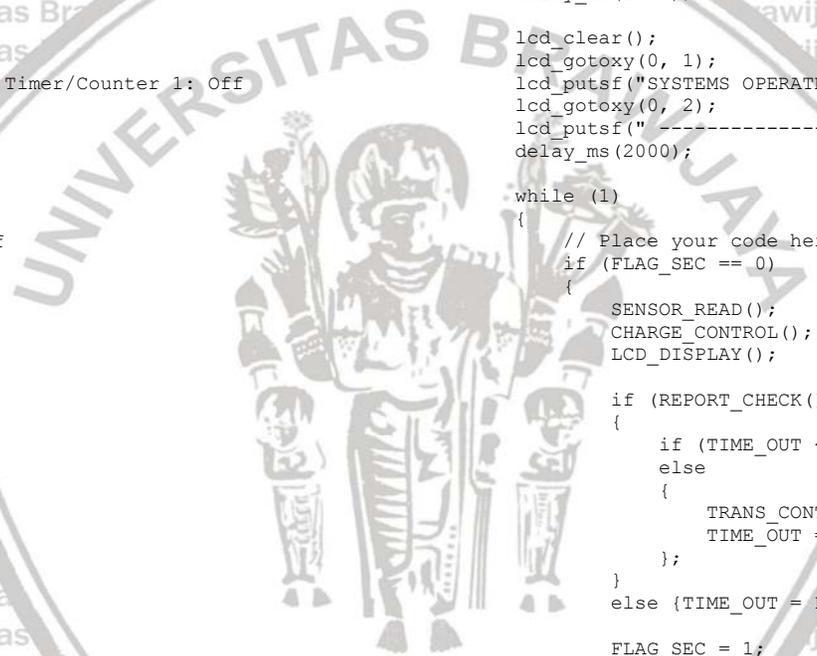
        lcd_clear();
        lcd_gotoxy(0, 1);
        lcd_putsf("SYSTEMS OPERATES");
        lcd_gotoxy(0, 2);
        lcd_putsf("-----");
        delay_ms(2000);

        while (1)
        {
            // Place your code here
            if (FLAG_SEC == 0)
            {
                SENSOR_READ();
                CHARGE_CONTROL();
                LCD_DISPLAY();

                if (REPORT_CHECK()==1 && FLAG_DELETE==1)
                {
                    if (TIME_OUT < 30) {TIME_OUT++;}
                    else
                    {
                        TRANS_CONTROL();
                        TIME_OUT = FLAG_DELETE = 0;
                    };
                }
                else {TIME_OUT = FLAG_DELETE = 0;};

                FLAG_SEC = 1;
            };

            if (FLAG_DUR == 0)
            {
                DURATION = ((HEIGHT*58)/19)-31;
                TRANS_CONTROL();
                FLAG_DELETE = FLAG_DUR = 1;
            };
        }
    
```





LAMPIRAN IV

DATASHEET

