PENYIMPAN DATA PADA SISTEM TELEMETRI STASIUN PENGAMAT CUACA DILENGKAPI DENGAN SENSOR KELEMBABAN, TEMPERATUR, CURAH HUJAN, KECEPATAN ANGIN DAN ARAH ANGIN DIAKSES MELALUI INTERNET

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik

Disusun oleh:

ERDY PRASETIYA KUSUMA
9901060311-63

DEPARTEMEN PENDIDIKAN NASIONAL

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2007

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PENYIMPAN DATA PADA SISTEM TELEMETRI STASIUN PENGAMAT CUACA DILENGKAPI DENGAN SENSOR KELEMBABAN, TEMPERATUR, CURAH HUJAN, KECEPATAN ANGIN DAN ARAH ANGIN DIAKSES MELALUI INTERNET

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

Disusun oleh:

ERDY PRASETIYA KUSUMA
9901060311-63

Dosen Pembimbing

Ir. Bambang Siswojo
NIP. 131 759 588

Ir. Muhammad Aswin
NIP. 131 879 045

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PENYIMPAN DATA PADA SISTEM TELEMETRI STASIUN PENGAMAT CUACA DILENGKAPI DENGAN SENSOR KELEMBABAN, TEMPERATUR, CURAH HUJAN, KECEPATAN ANGIN DAN ARAH ANGIN DIAKSES MELALUI INTERNET

Disusun oleh:

ERDY PRASETIYA KUSUMA
9901060311-63

Skripsi ini telah diuji dan dinyatakan lulus

Tanggal 24 Desember 2005

Dosen Penguji

Ir. Heru Nurwarsito
NIP. 131 879 033

Ir. Ponco Siwindarto, MS
NIP. 131 837 966

Ir. Nanang Sulistyanto, MSc
NIP. 132 090 389

Ir. Nurussa'adah
NIP. 131 994 339

Mengetahui

Ketua Jurusan Teknik Elektro

Ir. HERU NURWARSITO
NIP. 131 879 033

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

ABSTRAK

ERDY PRASETIYA KUSUMA, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya, Oktober 2005, *Penyimpan Data Pada Sistem Telemetri Stasiun Pengamat Cuaca Dilengkapi Dengan Sensor Kelembaban, Temperatur, Curah Hujan, Kecepatan Angin dan Arah Angin Diakses Melalui Internet.*
Dosen Pembimbing : Ir. Bambang Siswoyo dan Ir. Muhammad Aswin

Indonesia memiliki 2 macam musim yaitu penghujan dan kemarau dengan rentang waktu kurang lebih 6 bulan masing-masing musim. Tetapi pada akhir-akhir ini dapat kita rasakan bahwa keteraturan antara musim hujan dan musim kemarau mulai terganggu, sehingga pengamatan terhadap cuaca harus lebih diperhatikan.

Berdasarkan alasan di atas maka disusun sebuah alat yang diharapkan dapat menyimpan data parameter cuaca secara sementara dalam kurun waktu tertentu sebelum dikirim pada stasiun pusat.

Penyimpan Data (*data logger*) berfungsi sebagai pengumpul data sementara (*polling data*) sebelum data tersebut di kirim pada Stasiun Pusat sehingga unit Pengontrol Sensor dapat bekerja lebih efisien dan *real time.* Dengan adanya pembagian tugas diatas, Unit Pengontrol Sensor bertugas mengambil data cuaca kemudian mengirimkan data tersebut pada Unit Penyimpan data yang akan menyimpan data tersebut dalam sebuah memori dan dalam jangka waktu tertentu kumpulan data tersebut akan dikirim pada Stasiun Pusat. Pengiriman data yang berkala juga dapat menghemat pemakaian catu daya daripada pengiriman data secara langsung.

Data parameter cuaca akan disimpan pada Serial EEPROM dengan kapasitas 512 kilobit atau 64 kilobyte. Data yang diterima penyimpan data (*Data Logger*) dari pengontrol sensor dalam sekali pengiriman sebanyak 14 byte, yang terdiri dari 5 byte data dari RTC (*Real Time Clock*) berisi data menit, jam, tanggal, bulan, dan tahun. Sedangkan 9 byte data yang lain berupa 2 byte data temperatur, 2 byte data kelembaban, 2 byte data curah hujan, 2 byte data kecepatan angin, 1 byte data arah angin.

UNIVERSITAS BRAWIJAYA

PENGANTAR

Alhamdulillah, Segala puji syukur ke hadirat Allah SWT atas rahmat, hidayah serta karunia–Nya, sehingga kami dapat menyelesaikan skripsi yang berjudul "PENYIMPAN DATA PADA SISTEM TELEMETRI STASIUN PENGAMAT CUACA DILENGKAPI DENGAN SENSOR KELEMBABAN, TEMPERATUR, CURAH HUJAN, KECEPATAN ANGIN DAN ARAH ANGIN DIAKSES MELALUI INTERNET" dengan baik dan lancar.

Penyusunan skripsi ini diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Teknik, yang merupakan salah satu syarat yang harus ditempuh oleh setiap mahasiswa Jurusan Teknik Elektro di lingkungan Fakultas Teknik Universitas Brawijaya Malang.

Dengan kesungguhan dan rasa rendah hati, penyusun mengucapkan terima kasih kepada :

1. Ir. Heru Nurwarsito selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang.
2. Rudy Yuwono, ST selaku Sekertaris Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang.
3. Ir. Ponco Siwindarto, MS selaku KKDK Elektronika Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
4. Bapak Ir. Bambang Siswojo selaku Dosen pembimbing yang telah banyak memberikan ide, bimbingan, motivasi, pengarahan serta saran dalam penyusunan skripsi ini.
5. Bapak Ir. Muhammad Aswin selaku Dosen pembimbing yang telah banyak memberikan saran serta motivasi, masukan serta pengarahan dalam penyusunan skripsi.
6. Bapak, Ibu dosen serta segenap staf dan karyawan Jurusan Teknik Elektro baik secara langsung maupun tidak langsung yang telah banyak membantu dalam menyelesaikan skripsi ini.
7. Bapak, Ibu, Adikku Ertha Kusuma Dewi, Bojoku Ike Kusuma Dewi, Pak No, Bu Yuli, Benny (ndendotz) serta seluruh keluarga besar Tirtodjoyo dan Suparno yang tiada henti mendukung dengan segenap upaya dan doa yang tiada putusnya.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

8. Teman-teman angkatan 99, para konco plek Senosuke, Iman, Angki Sandjaya, Hendra, Wisnoe, Santi, jeng Elis, Shifit, Sahir, Gundul, Syarief , bogel, Serta 140 Teman 99ku, Mario, Hendro, Heru, Kilil, Warih, Rekan P Bmb, Konco Sumbersari Gg IV/ 261 C, Konco KK21, Konco Kesumba 2B(andre, ical, tomi, novi, G man.dll) suwun atas semuanya rek
9. Teman – teman asisten Lab. Sistem Digital angkatan 1997, 1998, 1999, 2000, 2001 dan tak lupa Mas Dulhadi yang telah banyak memberikan dukungan dan motivasi..
10. Semua pihak yang tidak dapat kami sebutkan satu persatu, sehingga skripsi ini dapat terselesaikan dengan sebaik – baiknya.

Penyusun menyadari bahwa skripsi yang kami susun masih banyak kekurangan serta masih jauh dari sempurna, hal ini tidak lain karena keterbatasan materi dan pengetahuan yang dimiliki penyusun. Oleh karena itu kritik serta saran sangat kami harapkan demi kesempurnaan penyusunan skripsi ini.

Akhir kata, penyusun mengharapkan semoga skripsi ini dapat bermanfaat bagi rekan – rekan mahasiswa dan bagi seluruh pembaca pada umumnya.

Malang, Juli 2007

Penyusun

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR ISI

Halaman

ABSTRAK i
PENGANTAR ii
DAFTAR ISI iv
DAFTAR TABEL vii
DAFTAR GAMBAR viii
DAFTAR LAMPIRAN ix
I. PENDAHULUAN 1
1.1. Latar Belakang 1
1.2. Rumusan Masalah 2
1.3. Ruang Lingkup 3
1.4. Tujuan 3
1.5. Sistematika Penulisan 3
II. DASAR TEORI 5
2.1. Umum 5
2.2. Mikrokontroler AT89C51 6
2.2.1. Arsitektur Mikrokontroler AT89C51 6
2.2.2. Susunan dan Fungsi Pin pada AT89C51 8
2.2.3. Register-register pada AT89C51 10
2.2.4. Interupsi (*Interrupt*) 13
2.3. Media Penyimpan Data 13
2.3.1. Memori Magnetik 13
2.3.2. Memori Semikonduktor 14
2.3.2.1. RAM (*Random Access Memory*) 14
2.3.2.2. ROM (*Read Only Memory*) 14
2.4. EEPROM Serial 16
2.4.1. Protokol I^2C 17
2.4.2. Pengalamatan EEPROM Serial 18
2.4.3. Operasi Penulisan 19
2.4.4. Operasi Pembacaan 20
2.5. Komunikasi Data Serial 22
2.5.1. Transmisi Sinkron 22

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2.5.2. Transmisi Asinkron 23
2.6. Basis Pewaktuan *Real Time Clock* (RTC) IC DS12887 24
2.6.1. Catu Daya RTC 27
2.6.2. Peta Memori 28
2.6.2.1. REGISTER A 29
2.6.2.2. REGISTER B 30
2.6.2.3. REGISTER C 30
III. METODOLOGI 30
3.1. Studi Literatur 32
3.2. Interview 32
3.3. Perancangan dan Pembuatan Alat 32
3.3.1. Spesifikasi Alat 32
3.3.2. Perancangan Alat 33
3.3.3. Pembuatan Alat 33
3.4. Pengujian dan Analisis Data 33
IV. PERENCANAAN DAN PEMBUATAN ALAT 34
4.1. Gambaran Umum 34
4.2. Spesifikasi Alat 35
4.3. Perencanaan Perangkat Keras 36
4.3.1. Sistem Mikrokontroler AT89C51 36
4.3.2. Rangkaian Antarmuka Modul RTC 38
4.3.3. Serial EEPROM 39
4.3.3.1. Operasi Penulisan 41
4.3.3.2. Operasi Pembacaan 42
4.4. PERANCANGAN PERANGKAT LUNAK 43
4.6.1. Perangkat Lunak Mikrokontroler 43
V. PENGUJIAN DAN ANALISA 46
5.1. Pengujian Rangkaian 46
5.1.1. Pengujian Komunikasi Serial 46
5.1.1.1. Tujuan 46
5.1.1.2. Alat dan Bahan 46
5.1.1.3. Prosedur Pengujian 46
5.1.1.4. Data Hasil Pengujian 48

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.1.1.5. Analisa Hasil Pengujian 48
5.1.2. Pengujian Rangkaian RTC 48
5.1.2.1. Tujuan 48
5.1.2.2. Alat dan Bahan 48
5.1.2.3. Prosedur Pengujian 49
5.1.2.4. Data Hasil pengujian 50
5.1.2.5. Analisa Hasil Pengujian 50
5.2. Pengujian Rangkaian Keseluruhan 50
5.2.1. Tujuan 50
5.2.2. Alat dan Bahan 50
5.2.3. Prosedur Pengujian 51
5.2.4. Data Hasil Pengujian 51
5.2.5. Analisa Hasil Pengujian 52
VI. KESIMPULAN DAN SARAN 54
6.1. Kesimpulan 54
6.2. Saran 54
DAFTAR PUSTAKA 56
LAMPIRAN 57

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR TABEL

Tabel	Judul	Halaman
2.1.	Tabel Fungsi Khusus Port 3 Mikrokontroller AT89C51	9
2.2.	Keterangan *Pin Out* RTC DS12887	24
4.1.	Data register pengaturan port serial dan besar nilainya	38
4.2.	Tabel Fungsi Pin IC AT24C512	40

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR GAMBAR

Gambar Judul Halaman

2.1. Blok diagram sistem telemetri stasiun pengamat cuaca 5
2.2. Arsitektur AT89C51 7
2.3. Susunan pin *AT89C51* 8
2.4. *Timing diagram* protokol I^2C 18
2.5. Operasi penulisan pada EEPROM serial AT24C512 20
2.6. Operasi pembacaan pada EEPROM serial AT24C512 21
2.7. Transmisi sinkron 22
2.8. Pengiriman data serial tak sinkron 23
2.9. Konfigurasi penyemat IC DS12887 25
2.10. Peta alamat *RAM internal* real time clock *DS12887* 29
2.11. Denah register A 29
2.12. Denah register B 30
2.13. Denah register C 31
4.1. Blok diagram sistem penyimpan data 34
4.2. Rangkaian mikrokontroler AT89C51 pada sistem penyimpan data 37
4.3. Rangkaian antarmuka RTC dengan sistem AT89C51 39
4.4. IC SEEPROM AT24C512 39
4.5. SEEPROM AT24C512 dengan mikrokontroller AT89C51 40
4.6. Operasi penulisan data pada SEEPROM 41
4.7. Operasi pembacaan data dari SEEPROM 42
4.8. Diagram alir program utama 44
4.9. Diagram alir program subrutin koneksi server 45
5.1. Diagram blok pengujian komunikasi serial antara mikrokontroler dengan Komputer 46
5.2. Diagram flowchart program pengujian komunikasi serial pada mikrokontroler 47
5.3. Hasil pengujian komunikasi serial menggunakan hyperterminal 48
5.4. Diagram blok pengujian rangkaian RTC 49
5.5. Diagram Flowchart pengujian rangkaian RTC 49
5.6. Hasil pengujian RTC menggunakan hyperterminal 50

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.7. Blok pengujian alat penyimpan data pada sistem telemetri stasiun pengamat Cuaca 51
5.8. Tampilan program hyper terminal pada saat pengujian alat keseluruhan 52

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR LAMPIRAN

Lampiran	Judul	Halaman
1 : RANGKAIAN SKEMATIK ALAT		
	Rangkaian skematik alat	L1 – 1
2 : LISTING PROGRAM		
	Listing Program	L2 – 1
3 : FOTO ALAT		
	Foto tampilan pada PC saat data dalam memori dibaca	L3 – 1
	Foto letak komponen rangkaian yang direncanakan	L3 – 1
4 : DATASHEET KOMPONEN		
	2-wire Serial EEPROM 512K (65536x8) AT24C512	L4 – 1
	Interfacing AT24CXX Serial EEPROMs with AT89CX051Microcontrollers	L4 – 2
	Dallas Semiconductor DS12887 Real-Time Clock	L4 – 3
	8 Bit Microcontroller with 4K Bytes Flash - AT89C51	L4 – 4

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB I
PENDAHULUAN

1.1 Latar Belakang

Indonesia sebagai negara yang terletak didaerah khatulistiwa memiliki iklim tropis, sehingga budaya rakyat Indonesia berkaitan erat dengan iklim tropis dan cuaca. Indonesia memiliki 2 macam musim yaitu penghujan dan kemarau dengan rentang waktu kurang lebih 6 bulan masing-masing musim. Tetapi pada akhir-akhir ini dapat kita rasakan bahwa keteraturan antara musim hujan dan musim kemarau mulai terganggu, sehingga pengamatan terhadap cuaca harus lebih diperhatikan.

Telemetri adalah metode yang digunakan untuk mengetahui kondisi atau nilai suatu besaran di tempat lain yang jaraknya cukup jauh. Pada sistem telemetri, secara garis besar terdapat dua alat yang terpisah, yaitu *Remote Telemetry Unit* (RTU) dan *Base Station*. RTU merupakan suatu alat yang berada pada satu tempat yang sama dengan besaran yang diamati, dan bertugas mengirimkan data hasil pengamatan ke *Base Station*. Sedangkan *Base Station* adalah unit yang berada pada tempat yang terpisah dengan besaran yang diamati, dan bertugas menerima dan mengolah data yang diterima dari RTU. Umumnya, sistem telemetri menggunakan beberapa RTU dan sebuah *Base Station*.

Pada telemetri terdapat beberapa medium yang dapat digunakan sebagai medium pengiriman data hasil pengukuran. Medium tersebut umumnya adalah kabel dan gelombang radio. Sarana internet bisa menggunakan kedua medium tesebut, dilengkapi dengan berbagai kemudahan yang ditawarkan protokol TCP/IP (*Transmission Control Protocol/Internet Protocol*).

Seiring dengan perkembangan teknologi, dalam usaha untuk mengamati perubahan cuaca, diperlukan sebuah stasiun pengamat cuaca yang bekerja secara otomatis untuk mengamati cuaca yang meliputi suhu udara, kelembaban udara, kecepatan angin, arah angin, serta curah hujan pada suatu wilayah tertentu. Hasil pengamatan tersebut dapat disimpan dan atau dikirimkan pada stasiun pusat pengamat cuaca.

Secara keseluruhan, penelitian ini terbagi menjadi tiga pokok bahasan, yaitu bagian pengendali sensor, *data logger*, dan *base station*. Pengendali sensor berfungsi sebagai pengontrol sensor temperatur, kelembaban, arah angin, kecepatan angin dan curah hujan. Pengontrol inilah yang mengatur pengubahan besaran fisik alam menjadi besaran data digital. Kemudian data hasil pembacaan dikirimkan ke *Data Logger*. *Data Logger* berfungsi untuk menyimpan data sementara dengan format tertentu selama kurun waktu

tertentu pula sebelum data tersebut ditransfer ke *Base Station*. Dalam *data logger* disediakan sejumlah memory untuk menyimpan data selama kurun waktu tertentu. *Base Station* berfungsi sebagai supervisor dalam sistem untuk mentransfer data, mengolah data, dan menampilkan data.

Skripsi ini membahas tentang Penyimpan Data (*data logger*) pada sistem telemetri stasiun pengamat cuaca. Pada bagian ini, Penyimpan Data (*data logger*) bertugas sebagai pengumpul data sementara (*polling data*) sebelum data tersebut di kirim pada Stasiun Pusat sehingga unit Pengontrol Sensor dapat bekerja lebih efisien dan *real time*. Dengan adanya pembagian tugas diatas, Unit Pengontrol Sensor bertugas mengambil data cuaca kemudian mengirimkan data tersebut pada Unit Penyimpan data yang akan menyimpan data tersebut dalam sebuah memori dan dalam jangka waktu tertentu kumpulan data tersebut akan dikirim pada Stasiun Pusat.

1.2 Rumusan Masalah

Dalam tugas akhir ini adalah bagaimana merencanakan dan membuat sistem Penyimpan Data (*Data Logger*) sebagai kesatuan judul pada Sistem Telemetri Stasiun Pengamat Cuaca Dilengkapi Sensor Kelembaban, Temperatur, Curah Hujan, Kecepatan Angin dan Arah Angin diakses melalui Internet.

Berdasarkan uraian di atas, masalah pada penelitian ini dapat dirumuskan sebagai berikut:

- Bagaimana menentukan jenis memori yang cocok untuk aplikasi penyimpanan data pada telemetri stasiun pengamat cuaca?
- Bagaimana merealisasikan sebuah penyimpan data yang dapat menyimpan data secara seri dan bisa menyimpan data secara permanen dalam jangka waktu tertentu?
- Bagaimana mengatur komunikasi antara unit penyimpan (*data logger*) dengan unit pengukur dan unit pusat pengontrol (*base station*)?
- Bagaimana mengatur dan menyiapkan data sehingga siap ditransmisikan ke pusat pengontrol (*base station*).

1.3 Ruang Lingkup

Agar diperoleh hasil pembahasan yang sesuai dengan tujuan yang diharapkan, maka perlu diberikan pembatasan masalah, sebagai berikut:

- Membahas tentang penyimpanan data yang berasal dari unit pengukur.

- Membahas tentang komunikasi antara unit pengukur dan pusat pengontrol.
- Membahas tentang pengolahan data sehingga data siap ditransmisikan.
- Tidak membahas tentang pentransmisian data.
- Catu daya tidak dibahas.

1.4 Tujuan

Penelitian ini bertujuan untuk merancang dan membuat penyimpan data pada sistem telemetri yang menyimpan data dari unit pengukur parameter cuaca pada *remote telemetry unit* secara permanen dalam jangka waktu tertentu dan menyiapkan data tersebut sehingga siap untuk ditransmisikan ke pusat pengontrol (*base station*) untuk diolah lebih lanjut sehingga stasiun pusat (*Base Station*) dan *Remote Telemetry Unit* dapat bekerja sama dalam membentuk sebuah sistem telemetri.

1.5 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan laporan tugas akhir ini adalah sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, tujuan, rumusan masalah, batasan masalah dan sistematika penulisan.

BAB II Teori Penunjang

Konsep dasar teori-teori yang mendukung perancangan dan pembuatan alat pengontrol sensor pada sistem telemetri pengamat cuaca.

BAB III Metodologi

Membahas tentang langkah-langkah perencanaan dan pembuatan sistem.

BAB IV Perencanaan dan Pembuatan Alat

Membahas tentang perancangan dan perealisasian Penyimpan Data pada Sistem Telemetri Stasiun Pengamat Cuaca dilengkapi dengan Sensor Kelembaban, Temperatur, Curah Hujan, Kecepatan Angin dan Arah Angin diakses melalui Internet.

BAB V Pengujian

Memuat hasil pengujian terhadap alat yang telah direalisasikan.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB VI Penutup

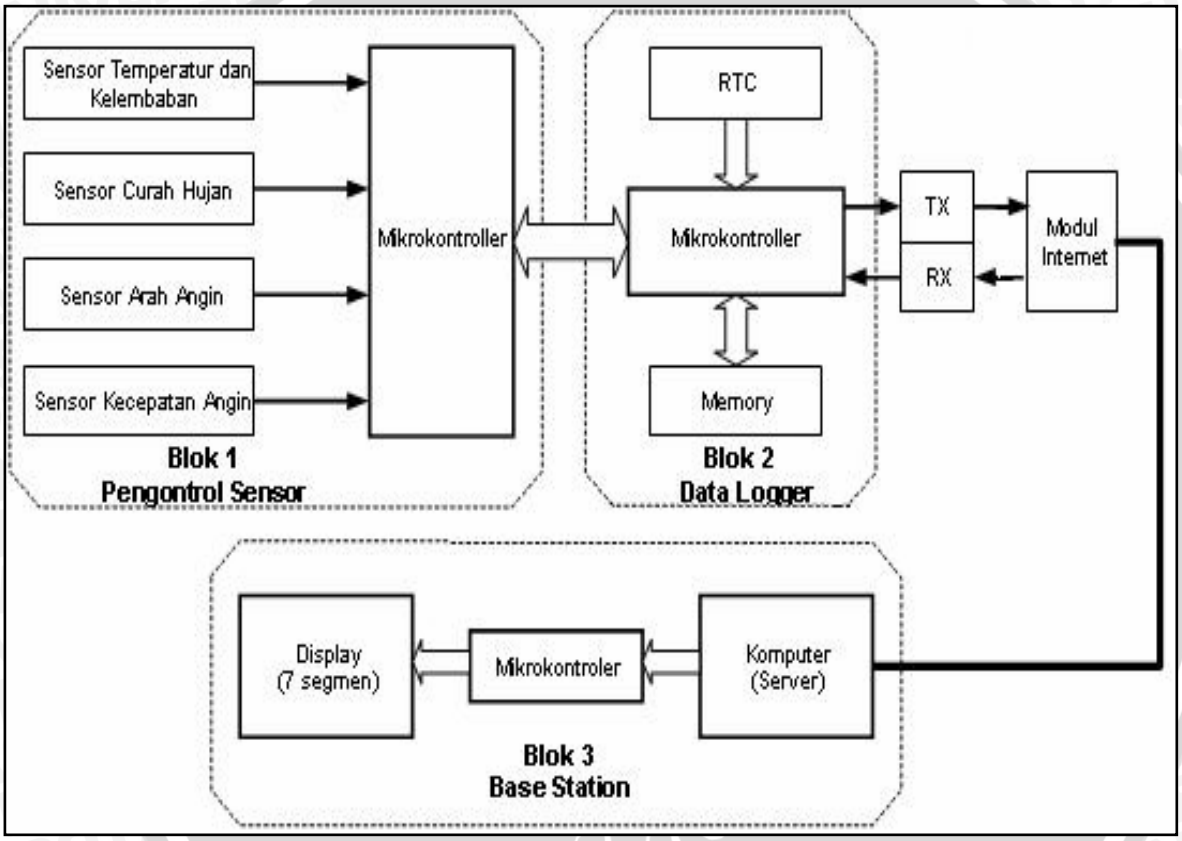
Memuat kesimpulan dan saran-saran dari sistem yang dianalisis pada tugas akhir ini.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB II
TINJAUAN PUSTAKA

2.1 Umum

Sistem Telemetri Stasiun Pengamat Cuaca yang dilengkapi dengan sensor kelembaban, temperatur, curah hujan, kecepatan dan arah angin diakses melalui internet ini meliputi tiga bagian, yaitu bagian pengendali sensor, penyimpan data (*data logger*) dan stasiun pusat (*Base Station*) pengolah data. Sistem telemetri stasiun pengamat cuaca dapat ditunjukkan seperti dalam Gambar 2.1.



Gambar 2.1.Blok diagram sistem telemetri stasiun pengamat cuaca

Sistem telemetri stasiun pengamat cuaca secara keseluruhan terdiri atas tiga bagian yaitu:

1. **Pengendali Sensor**

   Pengendali sensor berfungsi sebagai pengontrol sensor temperatur, kelembaban, arah angin, kecepatan angin dan curah hujan. Kontroler inilah yang mengatur pengubahan besaran fisik alam menjadi besaran data digital. Kemudian data hasil pembacaan dikirimkan ke Penyimpan Data (*Data Logger*) yang ditunjukkan dalam Blok 2.

2. **Penyimpan Data (*Data Logger*).**

   Penyimpan Data (*Data Logger*) berfungsi untuk menyimpan data sementara dengan format tertentu selama kurun waktu tertentu pula sebelum data tersebut ditransfer ke stasiun pusat (*Base Station*) yang ditunjukkan dalam Blok 3. Dalam Penyimpan Data (*Data Logger*) disediakan sejumlah memori untuk menyimpan data selama kurun waktu tertentu.

3. **Stasiun Pusat (*Base Station*).**

   Stasiun pusat (*Base Station*) berfungsi sebagai *supervisor* dalam sistem untuk mentransfer data, mengolah data, dan menampilkan data dalam bentuk web, yang bisa diakses melalui internet, menggunakan *Web-browser*.

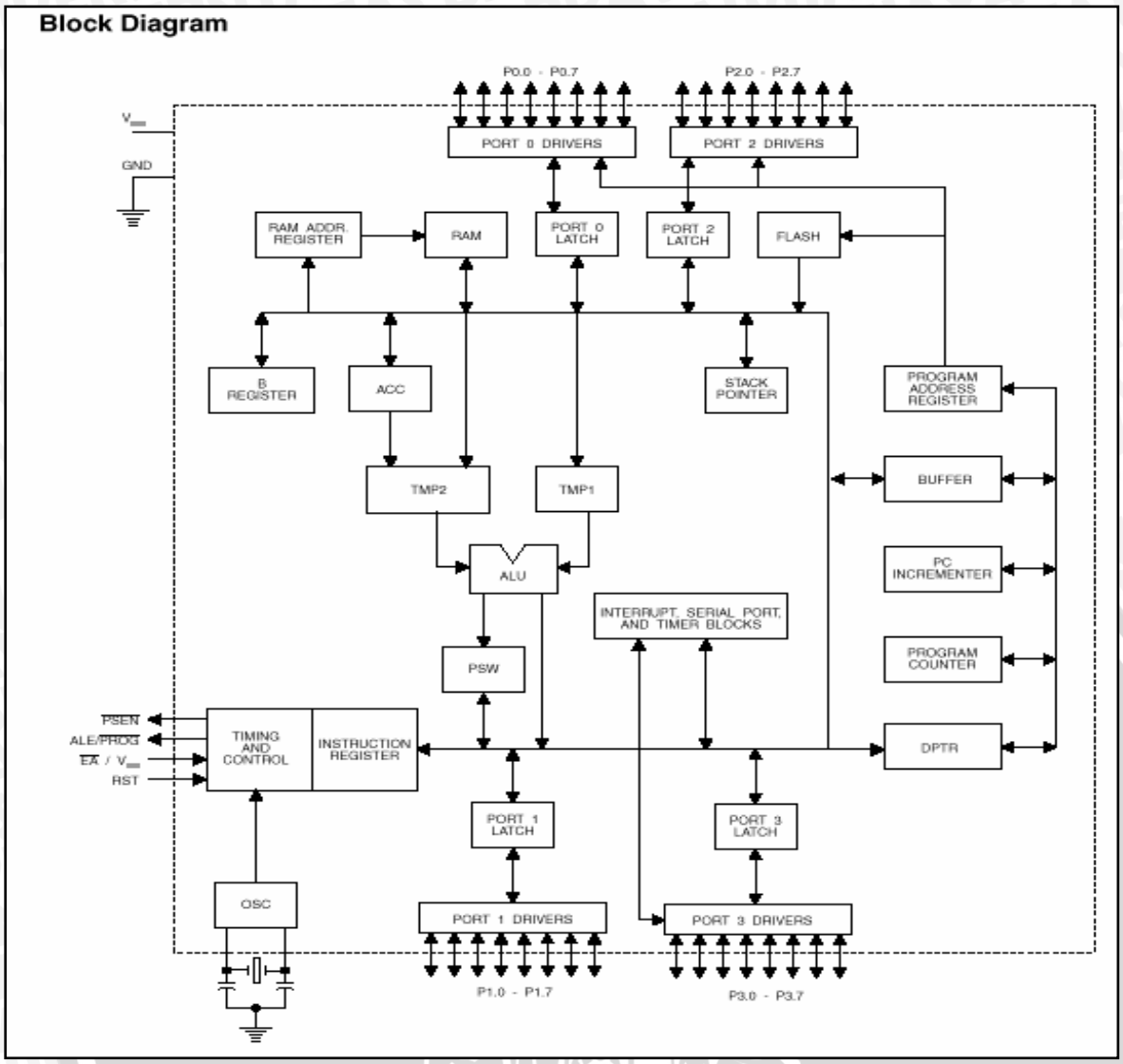
Skripsi ini membahas tentang penyimpan data pada sistem telemetri stasiun pengamat cuaca, seperti yang digambarkan pada Blok 2 dalam Gambar 2.1.

Dalam merencanakan dan merealisasikan alat Penyimpan data pada sistem telemetri stasiun pengamat cuaca dilengkapi dengan sensor kelembaban, temperatur, curah hujan, kecepatan angin dan arah angin diakses melalui internet dibutuhkan pemahaman tentang berbagai hal yang mendukung. Pemahaman ini akan bermanfaat untuk merancang perangkat keras dan perangkat lunak sistem yang dirancang. Pengetahuan yang mendukung perencanaan dan realisasi alat meliputi Mikrokontroler AT89C51, Media Penyimpan Data, EEPROM Serial, Konsep Komunikasi Data, RTC (*real time clock*).

2.2 Mikrokontroler AT89C51

2.2.1 Arsitektur Mikrokontroler AT89C51

Mikrokontroller AT89C51 produksi Atmel merupakan salah satu mikrokontroller 8 bit keluarga MCS-51. AT89C51 adalah *flash* mikrokontroler yang memiliki memori program dalam satu chipnya *(internal chip program memory)* dengan kapasitas memori 4 Kbyte. Selain itu AT89C51 mempunyai 128 byte RAM internal, empat buah I/O port 8 bit, dua timer 16 bit, *interface* untuk komunikasi serial dan mempunyai ruang sebesar 64 Kbyte untuk eksternal data dan kode memori. AT89C51 dapat dipakai dalam mode *single chip* atau mode *expand*. Untuk menggunakan AT89C51 tinggal menambahkan kristal untuk pewaktu *(clock)* dan jika diperlukan *power on reset.* Diagram blok AT89C51 dapat dilihat dalam Gambar 2.2.



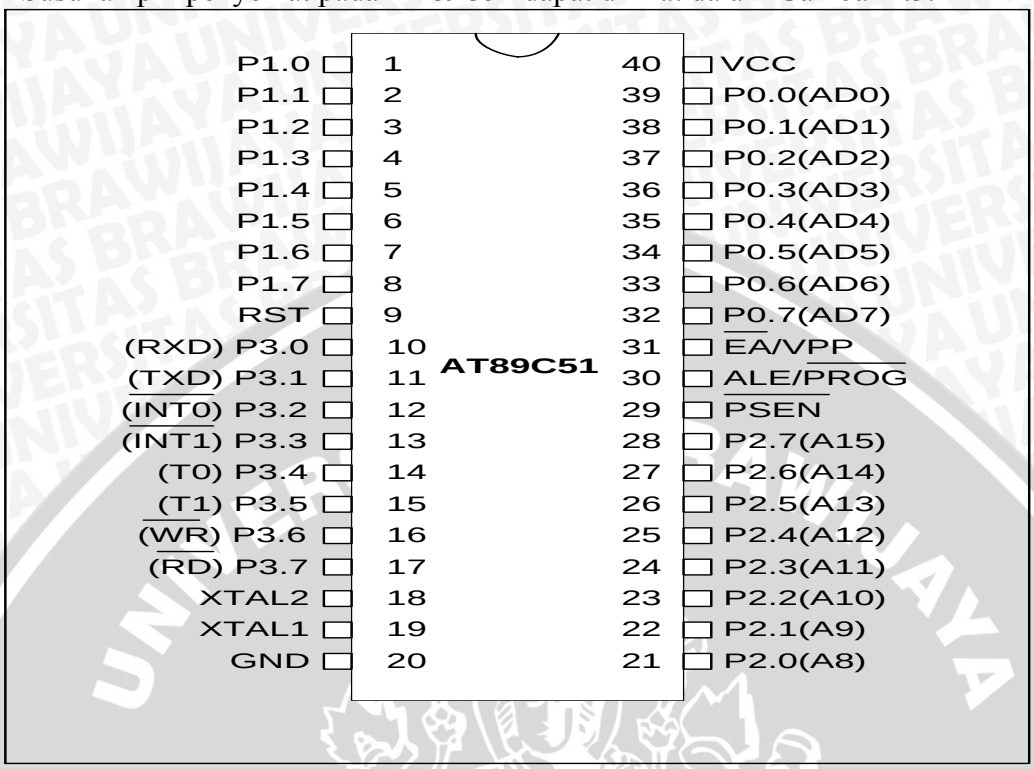
Gambar 2.2. Arsitektur AT89C51
Sumber : Annonymous, 2002 :2

Spesifikasi teknis dari mikrokontroler AT89C51 produksi Atmel adalah sebagai berikut :

- Internal memori program 4 kByte *(on chip)*
- RAM 128 Byte *(on mchip)*
- Mampu beroperasi dengan clock hingga 24 MHz
- Memiliki 6 line interup
- 32 bit *input output* terprogram
- *Full Duplex* komunikasi serial (UART)

2.2.2 Susunan dan Fungsi Pin pada AT89C51

Susunan pin penyemat pada AT89C51 dapat dilihat dalam Gambar 2.3.



Gambar 2.3. Susunan pin AT89C51
Sumber : Moh Ibnu Malik, 2003:1

Fungsi dari pin penyemat pada AT89C51 adalah sebagai berikut :

- V_{cc} (pin 40)

  Merupakan saluran catu daya +5 volt.

1 GND (pin 20)

  Merupakan saluran yang dihubungkan ke *ground* catu daya.

2 Port 0 (pin 32-39)

  Merupakan port I/O *(Input/Output)* paralel 8 saluran dua arah (*bidirecttional*).

  Berfungsi untuk *multiplexing* data dan alamat untuk memori eksternal.

3 Port 1 (pin 1-8)

  Merupakan port I/O *bidirectional* yang dapat dialamati per bit untuk berbagai keperluan.

4 Port 2 (pin 21-28)

  Merupakan port paralel 8 bit yang digunakan sebagai alamat bagian tinggi (*high address*) untuk mengakses memori eksternal.

5 Port 3 (pin 10-17)

Merupakan port I/O dua arah yang juga memiliki fungsi-fungsi khusus dan dapat dialamati per bit.

Tabel 2.1 menunjukkan fungsi-fungsi khusus dari port 3.

Tabel 2.1. Tabel Fungsi Khusus port 3 Mikrokontroller AT89C51

Bit	Nama	Alamat	Fungsi
Port3.0 (P3.0)	RxD	B0H	Menerima data serial
Port3.1 (P3.1)	TxD	B1H	Mengirim data serial
Port3.2 (P3.2)	INT0	B2H	*External interrupt* 0
Port3.3 (P3.3)	INT1	B3H	*External interrupt* 1
Port3.4 (P3.4)	T0	B4H	Timer / *counter* 0 *input*
Port3.5 (P3.5)	T1	B5H	Timer / *counter* 1 *input*
Port3.6 (P3.6)	WR	B6H	Sinyal tulis untuk data memori *external*
Port3.7 (P3.7)	RD	B7H	Sinyal baca untuk data memori *external*

Sumber: Annonymous, 2002:3

Reset (pin 9)

Merupakan masukan untuk mereset seluruh pengoperasian AT89C51

PSEN (pin 29):

PSEN (*Program Store Enable*) mengirimkan sinyal pengontrol sinkronisasi pengambilan memori program eksternal. Pada saat pengambilan memori program eksternal maka pin ini mempunyai kondisi *low* sehingga biasanya pin ini dihubungkan dengan pin *output enable* (OE) pada memori.

ALE (pin 30):

ALE (*Address Latch Enable*) mengirimkan sinyal pengontrol untuk menahan alamat memori eksternalnya selama pelaksanaan instruksi.

EA/V_{pp} (pin 31):

EA (*External Acces*) dihubungkan ke *ground* untuk pemakaian memori program eksternal. Untuk penggunaan memori program internal EA dihubungkan ke catu daya +5 volt. Dan untuk memprogram *flash* PEROM dihubungkan ke catu daya +12 volt.

XTAL2 dan XTAL1 (pin 18 dan pin 19):

Merupakan masukan untuk pewaktu sistem yang dapat dilakukan secara internal maupun eksternal.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2.2.3 Register-register pada AT89C51

AT89C51 ini mempunyai 8 bit register R0-R7 dengan 4 buah tempat penampungan berbeda alamat yang disebut *bank* dan mempunyai 21 buah SFR (*special function register*) yang terletak pada alamat 80H-FFH, semua register terletak pada internal RAM.

1. Register R0-R7

   Setelah reset *default* register terletak pada *bank* 0 yaitu alamat 00H – 07H. Register ini mempunyai 4 *bank, bank* 0 – *bank* 3 yang masing-masing *bank* mempunyai ruang 8 byte. Jadi alamat *bank* untuk register ini 00H –1FH. Register R0 dan R1 adalah register istimewa karena hanya register ini saja yang dapat dioperasikan dalam berbagai macam mode pengalamatan.

2. *Accumulator* (Acc):

   Register ini merupakan register serba guna dan umumnya digunakan untuk operasi matematika dan logika. Register ini memiliki panjang 8 bit dan terletak pada alamat E0H.

3. Register B:

   Register ini terletak pada alamat F0H memiliki panjang 8 bit dan biasanya digunakan bersama dengan *accumulator* untuk pengoperasian perkalian (Mul) dan pembagian (Div). Pada operasi perkalian, hasil perkalian *low byte* diletakkan pada *accumulator* sedang *high byte* pada register B, pada operasi pembagian, *accumulator* dibagi dengan register B dan hasil pembagian ada di *accumulator* sedang sisanya pada register B.

4. *Stack pointer* (SP) :

   Stack pointer merupakan register 8 bit yang terdapat pada alamat 81H, saat operasi *push*, data ke *stack* akan ditambah sebelum data ditulis, saat operasi *pop* data dari *stack* diambil setelah itu *stack* dikurangi. Biasanya pemakai menginisialisasi ulang *stack* tetapi jika tidak diinisialisasi ulang maka *default* dari *stack* adalah $07, ini berarti *bank* 1- *bank* 3 digunakan oleh *stack* sehingga register R0 –R7 harus diletakkan pada bank 0 sesuai dengan *default*nya.

5. *Data Pointer* (DPTR) :

   Data Pointer merupakan register 16 bit yang dibagi menjadi DPL alamat 82H untuk byte rendah (*low-byte*) dan DPH alamt 83H untuk byte tinggi (*high-byte*).

Fungsi utama dari DPTR digunakan untuk mengakses *eksternal data* atau kode memori dan juga menampung data 2 byte.

6.*Port Register*

AT89C51 memiliki register *port* 0 pada alamat 80H, register *port* 1 pada alamat 90H, register *port* 2 alamat A0H, register *port* 3 pada alamat B0H.

7.*Timer register*

AT89C51 memiliki 2 buah *timer* 16 bit. Register *timer* 0 berada pada alamat 8AH untuk TL0 (*low byte*) dan alamat 8CH untuk TH0 (*high byte*). Register *timer* 1 terletak pada lamat 8BH untuk TL1 dan alamat 8DH untuk TH1. Operasi dari *timer* ditentukan oleh register TMOD (*Timer/Counter Mode Control*), dengan alamat 89H untuk menentukan mode operasi dan register TCON (*Timer/Counter Control*), dengan alamat 88H sebagai kontrol operasi.

8.*Serial Port Register*

AT89C51 memiliki kaki untuk komunikasi serial oleh sebab itu diperlukan sebuah register untuk menampung data komunikasi. Register tersebut adalah *Serial Data Buffer* (SBUFF) pada alamat 99H. Register ini menampung data untuk mengirim ataupun data yang diterima. Berbagai macam mode operasi untuk serial komunikasi terletak pada *Serial Port Control Register* (SCON) pada alamat 98H.

9.*Interrupt Register*

AT89C51 memiliki 5 sumber interupsi (*serial port interrupt*, *timer* 0 dan *timer 1 interrupt, external interrupt*). Interupsi tidak aktif setelah sistem reset, maka interupsi harus diaktifkan dengan *software* pada *Interrupt Enable Register* (IE) alamat A8H. Prioritas interupsi dapat diatur pada *Interrupt Priority Register* (IP) B8H.

10. *Power Control Register* (PCON):

Didalam register PCON terdapat kontrol bit untuk catu daya IC sehingga dapat diatur pemakaian dayanya. Didalam register terdapat control bit SMOD untuk menggandakan *baud rate* serial *port* mode 1,2,3 dan control bit untuk *power down* (PD) juga untuk *idle mode* (IDL).

11. *Program status Word* (PSW):

Terletak pada alamat D0H dan merupakan bit status yang menunjukkan keadaan CPU pada saat itu.

Status terdiri atas:

a)*Carry Flag* (CY)

Bit ini akan bernilai satu apabila bit 8 terlampaui saat dilakukan operasi penjumlahan atau nilai pengurang lebih besar saat operasi pengurangan.

b)*Auxiliary Carry Flag* (AC)

Bit ini akan bernilai satu jika nilai suatu data diluar *range* $00 -$09 saat dilakukan operasi penjumlahan bilangan BCD.

c)Flag 0

Bit ini disediakan untuk status bit pemakai untuk berbagai keperluan. *Register Bank Select Bits* : 2 bit ini digunakan untuk memilih *bank* mana yang akan dipakai oleh register R0-R7 dengan memilih kombinasi dari RS0 dan RS1. Jumlah kombinasinya sesuai dengan jumlah *bank* yang ada. *Default* dari RS0 dan RS1 adalah bank 0.

d)Overflow Flag

Bit akan bernilai satu jika terjadi nilai *overflow* dari operasi penjumlahan ataupun pengurangan.

Parity Bit

Bit ini secara otomatis bernilai satu atau nol untuk menentukan nilai paritas.

2.2.4 Interupsi (*Interrupt*)

Saat CPU pada mikrokontroller AT89C51 sedang melaksanakan suatu program kita dapat menghentikan pelaksanaan program tersebut secara sementara dengan meminta interupsi. Bila CPU mendapat permintaan interupsi *program counter* (PC) akan diisi dengan alamat dari vektor interupsi. CPU kemudian melaksanakan rutin pelayanan interupsi mulai dari alamat tersebut. Bila rutin pelayanan interupsi selesai dilaksanakan, CPU AT89C51 kembali melaksanakan program utama yang ditinggalkan.

2.3 Media Penyimpan Data

Media penyimpan data yang digunakan adalah Serial EEPROM. Serial EEPROM digunakan karena sifatnya yang baik sebagai penyimpan data yang sifatnya tetap. Dengan digunakannya komponen ini data yang sudah terekam tidak akan hilang meskipun catu dayanya dihilangkan/dimatikan, dengan bentuknya berupa chip yang mempunyai dimensi kecil sehingga tidak membutuhkan ruang yang besar untuk penempatan dalam sistem pengamatan cuaca. Tetapi pada prinsipnya media penyimpan ini tidak harus berupa Serial EEPROM. Jenis media penyimpan yang lain juga dapat digunakan. Berikut ini akan diuraikan beberapa jenis media penyimpan data.

2.3.1 Memori Magnetik

Memori magnetik menyimpan data secara magnetik pada media peka magnet. Memori magnetik mempunyai kapasitas penyimpanan yang besar dan bersifat tidak mudah terhapus (*non volatile*). Contoh memori jenis ini adalah disket dan pita kaset.

Untuk penerapan sebagai media penyimpan data sistem, disket tidak layak digunakan karena memerlukan piranti pengendali yang rumit dan mahal. Piranti penggerak pita kaset lebih sederhana dan murah, tetapi adanya proses mekanis pada piranti tersebut menyebabkan perlunya perawatan yang terus menerus. Sebagai contoh kepala rekam pita kaset harus dibersihkan secara teratur. Mekanisme pemutar pita kaset juga harus diperiksa secara teratur untuk menjamin kerja piranti yang benar. Lagipula media penyimpan dengan pita kaset sekarang ini dianggap ketinggalan jaman.

2.3.2 Memori Semikonduktor

2.3.2.1 RAM (*Random Access Memory*)

RAM (*Random Access Memory* = memori yang diakses acak) adalah jenis memori baca/tulis (*read/write memory*). Akses acak berarti waktu akses tidak tergantung pada lokasi dimana data akan dituliskan/dibaca. Jadi waktu akses adalah sama untuk semua alamat memori.

Kelemahan utama sebuah RAM adalah sifatnya yang mudah terhapus (*volatile*), artinya bila catu daya dimatikan data yang tersimpan dalam RAM akan hilang. Terdapat dua jenis RAM, yaitu RAM statik dan RAM dinamik.

RAM Statik

RAM statik menyimpan satu bit data dalam sebuah flip-flop. Dengan demikian, sesuai dengan sifat flip-flop, isi sebuah RAM statik tetap stabil selama catu daya tersedia. Karena konsumsi dayanya yang rendah (Zaks, 1981), RAM statik dapat digunakan sebagai media penyimpan data pada sistem pengamat cuaca dengan melengkapinya dengan baterai. Tetapi cara ini menyebabkan keamanan data (dari kemungkinan terhapus) sepenuhnya bergantung pada keandalan baterainya dan memerlukan penanganan yang hati-hati agar RAM tidak pernah lepas dari baterai pendukungnya.

RAM Dinamik

RAM dinamik menyimpan 1 bit data sebagai muatan pada kapasitansi gerbang substrat sebuah transistor efek medan. Teknik ini menghasilkan kerapatan yang lebih tinggi dibandingkan dengan RAM statik, karena satu sel kapasitor ini lebih kecil dari satu flip-flop pada RAM statik. Kerugian RAM dinamik adalah keharusan untuk mengisi ulang kapasitor (*refresh*) untuk mengganti muatan kapasitor yang hilang karena kebocoran.

2.3.2.2 ROM (*Read Only Memory*)

ROM (*read only memory*) adalah memori yang hanya dapat dibaca tanpa dapat ditulisi. ROM bersifat tidak mudah terhapus (*non volatile*), data tetap tersimpan walaupun chip ROM tidak mendapat catu daya. ROM terdiri dari berbagai jenis yaitu: ROM murni, PROM (*user programmable ROM* = ROM yang dapat diprogram oleh pemakai), EPROM (*erasable* PROM = PROM yang dapat dihapus), EAROM (*electrically alterable* ROM = ROM yang dapat diubah secara elektris), EEPROM (*electrically erasable* PROM = PROM yang dapat diprogram secara elektris), dan SEEPROM (*serial* EEPROM = EEPROM yang masukan datanya berupa data serial).

ROM murni dan PROM tidak dapat dipakai sebagai media penyimpan data karena ROM murni dan PROM hanya dapat ditulisi/diprogram satu kali. Sekali diprogram, isi ROM dan PROM tidak dapat diubah lagi. Jenis yang lain, EAROM, EPROM, EEPROM, dan SEEPROM dapat ditulisi/diprogram kembali berulang-ulang.

EAROM

EAROM sebenarnya adalah jenis memori yang dapat dibaca maupun ditulisi. Tetapi karena operasi baca dan operasi tulis pada jenis memori ini jauh berbeda (operasi tulis jauh lebih lambat dan kompleks dibanding operasi baca), memori ini digolongkan pada jenis operasi baca saja.

EAROM menggunakan teknologi yang rumit dan memerlukan banyak tingkatan tegangan dan mahal. Penerapannya hanya terbatas pada penerapan kritis dibidang industri dan militer. Contohnya adalah pada sistem peluru kendali dimana keadaan tidak mudah terhapus. Ukuran kecil adalah hal yang utama, bukannya harga (Zaks, 1981).

EPROM

Seperti EAROM, EPROM juga ditulis/diprogram berulang-ulang. Tetapi EPROM harus dihapus dengan sinar ultra ungu sebelum dapat diprogram. Proses pemrograman memerlukan tegangan pemrograman yang tertentu (12 sampai 50 volt).

Sel penyimpan pada EPROM adalah transistor efek medan dengan gerbang silikon mengambang (*floating gate*) yang terisolisir secara listrik. Dengan memberikan pulsa pemrograman tegangan pemrograman, elektron-elektron berenergi tinggi di injeksikan ke gerbang silikon. Muatan elektron pada gerbang akan membuat transistor on, dan bit yang bersangkutan akan menjadi berlogika 1. setelah pemrograman selesai, muatan elektron akan tetap berada pada gerbang, karena tidak ada jalur untuk pengosongan muatan. Sekali diprogram, sebuah sel memori hanya dapat dihapus dengan menyinarinya dengan sinar ultra ungu atau frekuensi tinggi (karena itu sebuah EPROM selalu mempunyai "jendela" tembus cahaya untuk melewatkan sinar penghapus). Foton-foton sinar yang menumbuk gerbang mengambang ke substrat silikan, sehingga muatan gerbang menjadi berlogika 0. proses ini pada dasarnya analogis dengan efek foto listrik. Jadi data yang tersimpan dalam EPROM relatif permanen, asalkan EPROM terlindung dari smber sinar ultra ungu atau sinar frekuensi tinggi lain (misalnya sinar matahari, lampu *fluorescent*). Hal ini dapat dilakukan dengan mudah, yakni dengan menutup jendela EPROM dengan penutup yang tak tembus cahaya.

EEPROM

EEPROM merupakan perkembangan EPROM, dimana data yang tersimpan di dalam EEPROM dapat dihapus dengan tegangan listrik. Apabila dibandingkan dengan penghapusan pada EPROM yang menggunakan penyinaran lampu *ultraviolet*, penghapusan dengan dengan tegangan listrik lebih praktis. Karena penghapusan dengan cahaya *ultraviolet* memerlukan waktu yang lama (± 15 –20 menit). Penghapusan menggunakan tegangan listrik lebih aman dan lebih praktis bila dibandingkan dengan penghapusan dengan sinar *ultraviolet*, karena mekanisme penghapusannya dapat dilakukan satu persatu data. Sehingga data yang berada diluar alamat yang tidak dihapus tetap

tersimpan dalam sel penyimpanannya. Dalam hal ini bukan berarti penghapusan pada EEPROM tidak dapat dilakukan secara serentak seperti pada EPROM, karena penghapusan pada EEPROM juga menawarkan penghapusan serentak yang dilakukan dengan pemberian tegangan tertentu pada saluran kontrol keluaran ($\overline{OE}$). Dengan sistem penghapusan menggunakan tegangan listrik ini, memungkinkan EEPROM untuk dapat dihapus dalam rangkaian tanpa melakukan pemindahan komponen.

Pada EEPROM membutuhkan 8 jalur data, dan juga jalur alamat yang sebanding dengan kapasitas memori tersebut. Sebagai contoh EEPROM dengan kapasitas 8 kilobyte membutuhkan 13 jalur alamat.sehingga EEPROM dengan kapasitas 8 kilobyte membutuhkan 21 jalur data dan 2 pin Vcc dan *ground*. Sehingga membutuhkan ruang yang besar dan tidak praktis. Dengan keterbatasan jalur alamat pada mikrokontroler maka kapasitas memori EEPROM yang bisa digunakan adalah 16 jalur alamat, kapasitasnya sebesar 64 kilobyte.

2.4 EEPROM Serial

EEPROM serial dikembangkan untuk mengatasi keterbatasan jumlah I/O pada mikroprosesor/mikrokontroller. Yang membedakan antara EEPROM serial dengan EEPROM paralel adalah antar muka EEPROM serial yang diakses secara serial. Sebuah IC memori dengan kapasitas 2 kByte yang dibentuk dengan teknik transfer data secara pararel paling tidak mempunyai 24 kaki, 8 kaki untuk jalur data, 11 kaki untuk jalur alamat, 3 kaki untuk jalur kontrol, 2 kaki untuk catu daya. Memori yang sama kalau dibentuk dengan teknik transfer data secara seri mempunyai 8 kaki, dan hanya 2 atau 3 kaki yang perlu dihubungkan ke mikroprosesor/mikrokontroller. Keunggulan lainnya adalah daya yang dibutuhkan lebih rendah, 2 mA untuk operasi pembacaan dan 3 mA untuk operasi penulisan, dan ukuran fisik yang lebih kecil.

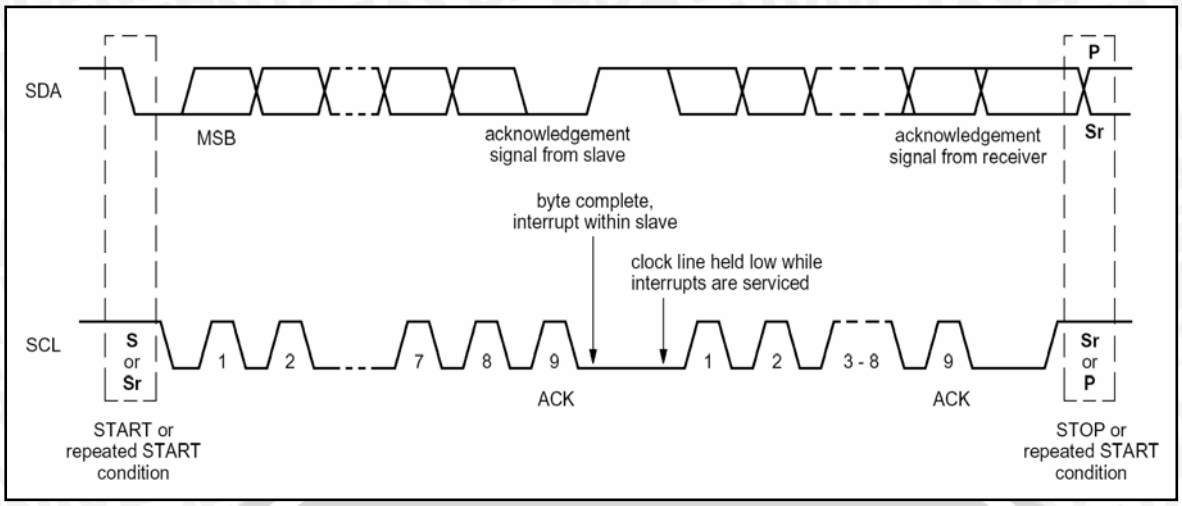
Teknik transfer data secara seri antar IC dikembangkan oleh 3 perusahaan IC, yang pertama adalah teknik I^2C (*Inter Integrated Circuit*) yang dikenalkan oleh Philips, teknik SPI (*Serial Peripheral Interface*) dari Motorola dan teknik MicroWire dari National Semiconductor. Protokol I^2C memakai 2 jalur untuk keperluan transfer data secara seri, sedangkan SPI dan MicroWire memakai 3 jalur. Semua protokol mempunyai 1 jalur untuk *clock*, I^2C hanya punya satu jalur data 2 arah, sedangkan SPI dan MicroWire mempunyai 2 jalur data satu arah, masing-masing untuk jalur data masuk dan jalur data keluar. Protokol yang paling sering digunakan pada EEPROM serial adalah I^2C.

2.4.1 Protokol I^2C

I^2C versi 1.0 dikenalkan oleh Philips pada tahun 1992, direvisi menjadi versi 2.0 pada tahun 1998, kemudian direvisi lagi menjadi versi 2.1 pada tahun 2000. Komunikasi data secara I^2C dilakukan melalui dua saluran, masing-masing adalah saluran data secara seri (SDA) dan saluran *clock* (SCL), kedua saluran ini dikenal sebagai bus I^2C yang dipakai menghubungkan banyak IC I^2C untuk berbagai macam keperluan. IC-IC I^2C itu dibedakan menjadi *master* dan *slave*. yang dimaksud dengan *master* adalah peralatan I^2C yang memulai transfer data dan yang membangkitkan *clock* (SCK). Yang bertindak sebagai *master* umumnya adalah mikroprosesor/mikrokontroler yang bertugas mengendalikan bus I^2C.

Pada kondisi normal, SDA hanya boleh berubah selama SCL *low*. Pengecualian kondisi ini adalah saat kondisi *start* dan *stop*. Kondisi *start* menandakan *master* akan memulai proses pertukaran data dengan ditandai transisi *high* ke *low* pada saluran SDA saat saluran SCL *high*. Sedangkan kondisi *stop* menandakan *master* akan menghentikan proses pertukaran data dengan ditandai transisi *low* ke *high* pada saluran SDA saat saluran SCL *high*.

Lebar data yang dikirimkan melalui SDA harus sebanyak 8 bit atau 1 byte. Banyaknya byte yang dapat dikirimkan tidak dibatasi. Masing masing byte harus diikuti bit *acknowledge* oleh pihak penerima data dengan menahan saluran SDA pada kondisi *low* saat pulsa *clock acknowledge* sedang dalam periode *high*. Jika pihak penerima tidak memberikan *acknowledge*, berarti pihak penerima tidak sedang dalam kondisi siap bertukar data. Data ditransfer dengan MSB terlebih dahulu. Jika *slave* tidak dapat menerima atau mengirimkan byte data selama melakukan beberapa fungsi lain, sebagai contoh melayani interupt internal, *slave* dapat memaksa *master* ke kondisi *wait* dengan menahan saluran SCL pada kondisi *low*. Perpindahan data kemudian dilanjutkan jika *slave* siap untuk bertukar data dan melepasan saluran SCL. Proses pengiriman data dapat dihentikan dengan memberikan kondisi *stop* walaupun proses pengiriman belum lengkap satu byte. Dalam kasus ini, tidak ada sinyal *acknowledge*. *Timing diagram* protokol I^2C dapat dilihat dalam Gambar 2.4.



Gambar 2.4. *Timing diagram* protokol I^2C
Sumber : Philips, 2000 : 10

2.4.2 Pengalamatan EEPROM Serial

Karena IC I^2C hanya dikendalikan lewat kaki SDA dan SCK saja, tidak ada sarana lainnya dari *master* yang bisa dipakai untuk mengendalikan I^2C, maka alamat yang dipakai untuk memilih isi IC I^2C dikirimkan secara serial pula, persis seperti halnya pengiriman data.

Pengalamatan dasar IC I^2C dilakukan dengan nomor group dan nomor chip. Nomor group adalah nomor yang diberikan oleh Philips sebagai pencipta I^2C pada kelompok-kelompok IC I^2C. Sebagai contoh nomor group untuk EEPROM serial adalah 1010 (biner). Nomor *chip* adalah nomor yang diberikan pada masing-masing *chip* lewat kaki A0, A1, dan A2 dari masing-masing IC. Dalam IC I^2C tertentu, A0..A2 tidak dihubungkan ke kaki IC, tapi dipakai didalam IC untuk menomori *register* atau memori di dalam IC bersangkutan.

Setelah *master* I^2C mengirimkan sinyal *start*, byte pertama yang dikirim berisi nomor group, nomor *chip* dan 1 bit lagi sebagai penentu arah data (*low* = *write*, *high* = *read*), seperti yang digambarkan dalam Gambar 2.5. Setelah *master* mengirimkan byte pertama, IC I^2C yang memiliki nomor group dan *chip* yang bersesuaian menyisipkan sinyal *acknowledge* sebagai kesiapannya bertukar data. Selanjutnya *master* akan melakukan pertukaran data dengan *slave* sesuai dengan bit penentu arah data. Jika pertukaran data telah selesai, *master* mengakhiri hubungan dengan mengirimkan sinyal *stop*.

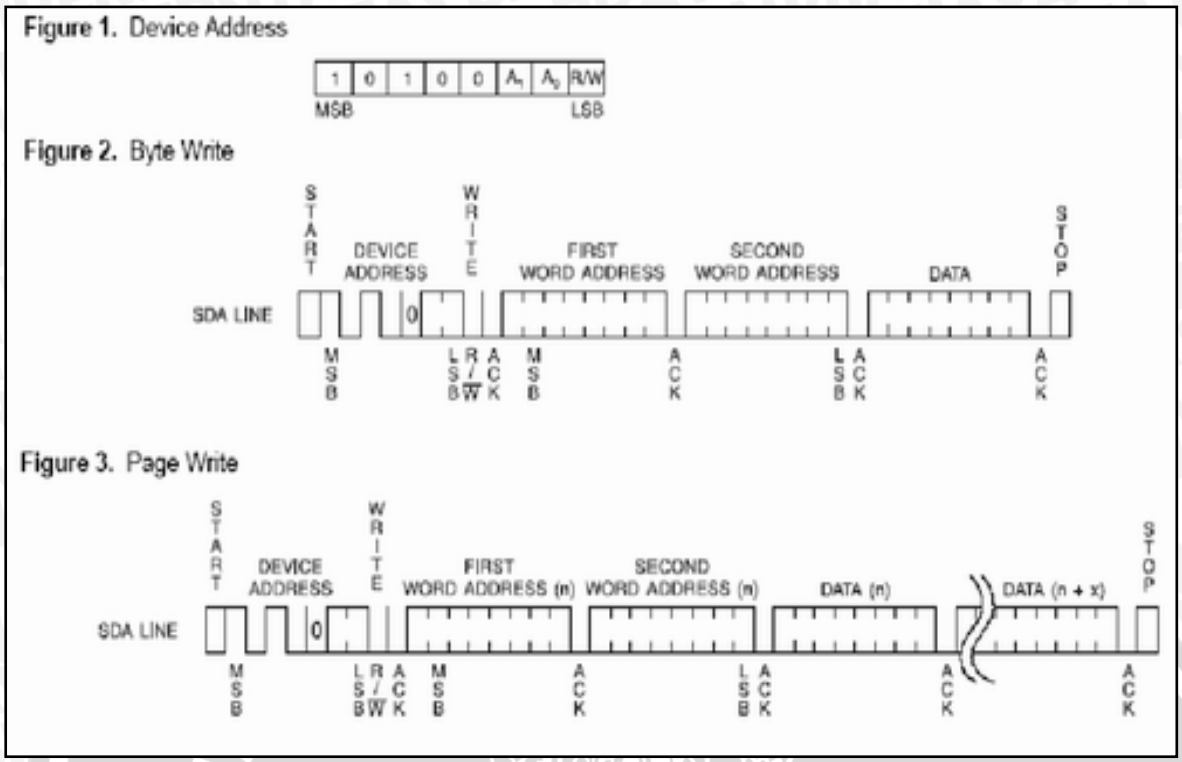
Dengan cara pengalamatan diatas *master* dapat mengendalikan 128 (7 bit) alamat. Untuk EEPROM serial, dimana dibutuhkan jumlah alamat yang lebih banyak, digunakan cara pengalamatan yang lain. EEPROM dengan kapasitas 2 kB kebawah, misal 24C16

yang memiliki kapasitas 2 kB, menggunakan 3 bit nomor *chip* dan 1 byte alamat yang dikirim setelah byte pertama sehingga dapat mengalamati 11 bit alamat pada EEPROM. Sedangkan EEPROM dengan kapasitas lebih dari 2 kB, misal 24C256 yang memiliki kapasitas 32 kB, menggunakan 2 byte alamat yang masing-masing dikirim setelah byte pertama.

2.4.3 Operasi Penulisan

Operasi penulisan pada EEPROM serial dapat dibagi menjadi dua jenis, yaitu penulisan perbyte dan penulisan perhalaman. Pada penulisan perbyte, 8 bit data dikirimkan setelah *master* mengirimkan kondisi *start*, 4 bit alamat group, 3 bit alamat *chip* atau alamat EEPROM, bit "0" untuk mode *write, acknowledge* dari EEPROM, 8 bit alamat EEPROM, dan *acknowledge* dari EEPROM. Setelah 8 bit data diterima EEPROM akan memberikan *acknowledge* dan *master* mengakhiri operasi penulis dengan kondisi *stop*. Setelah kondisi *stop*, EEPROM memasuki siklus penulisan ke memori *nonvolatile* selama waktu siklus tulis. Semua masukan tidak berfungsi selama siklus penulisan dan EEPROM tidak akan menjawab panggilan sampai proses penulis selesai. Proses penulisan perbyte dapat dilihat dalam Gambar 2.5.*figure* 2.

Langkah-langkah penulisan perhalaman hampir sama dengan penulisan perbyte, namun setelah data 8 bit dikirim dan mendapat *acknowledge* dari EEPROM, *master* tidak mengirimkan kondisi *stop* tetapi mengirimkan data yang akan di tulis pada alamat selanjutnya. Pada akhir pengiriman masing-masing byte EEPROM memberikan *acknowledge*. Masing-masing tipe EEPROM serial biasanya mempunyai ukuran halaman tertentu, misalnya AT24C64 memiliki halaman sebesar 32 byte. Setiap selesai menerima data, empat bit LSB alamat EEPROM otomatis dinaikkan, sedangkan bit yang lain tetap. Setelah batas akhir halaman terlampaui, alamat akan kembali ke awal halaman. Proses penulisan perhalaman dapat dilihat dalam Gambar 2.5 (*figure* 3).



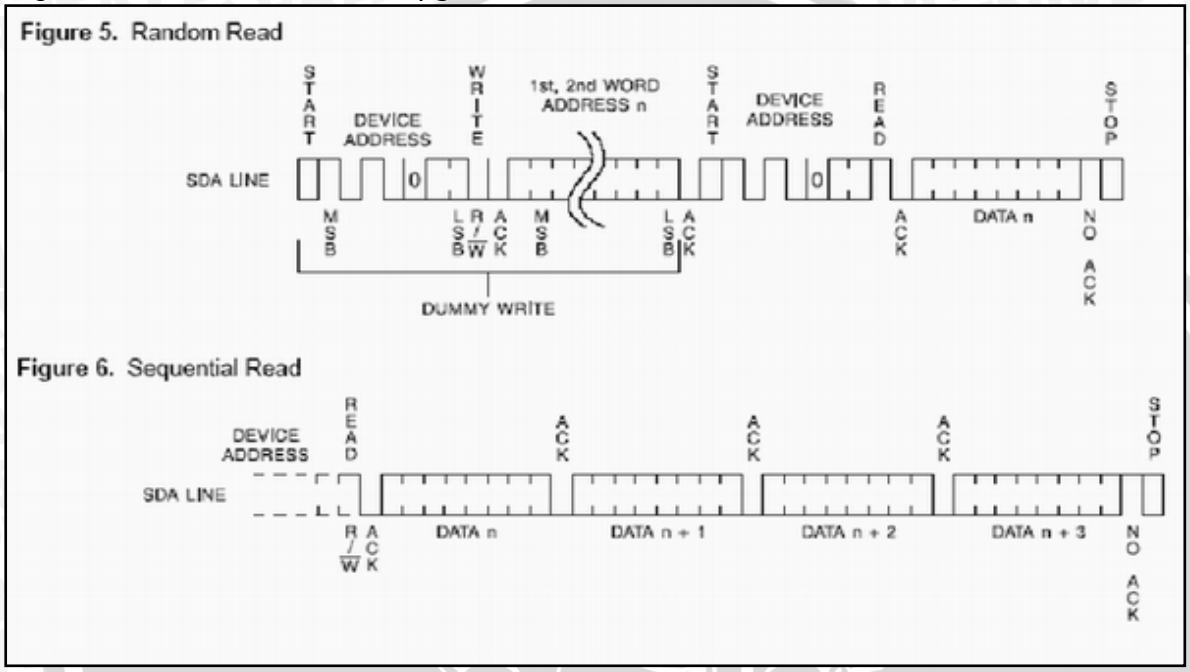
Gambar 2.5. Operasi penulisan pada EEPROM serial AT24C512
Sumber: Annonymous, 2004:11

2.4.4 Operasi Pembacaan

EEPROM serial memiliki *counter* alamat internal yang menyimpan satu alamat sesudah alamat terakhir yang diakses dengan operasi *write* atau *read*. Alamat pada *counter* alamat internal akan tetap valid selama EEPROM masih mendapat catu daya. Untuk memulai operasi pembacaan, *master* mengirimkan kondisi *start* diikuti byte pertama. Setelah mengirimkan *acknowledge*, EEPROM mengirimkan data ke *master*. Untuk mengakhiri hubungan, *master* tidak memberikan *acknowledge* setelah bit data terakhir dikirimkan oleh EEPROM, kemudian disusul dengan mengirimkan kondisi *stop*.

Untuk melakukan pembacaan secara acak (*random read*), dibutuhkan urutan penulisan semu untuk memuat alamat ke *counter* alamat EEPROM. *Master* memulai dengan urutan operasi penulisan. Setelah alamat EEPROM dikirim dan mendapat *acknowledge* dari EEPROM, *master* tidak mengirimkan data, tetapi mengirimkan kondisi *start* baru dan memulai proses pembacaan. Proses pembacaan secara acak dapat dilihat dalam Gambar 2.6.*figure*5.

Pembacaan juga dapat dilakukan secara berurutan (*sequential read*) dengan memberikan *acknowledge* setelah EEPROM mengirimkan byte data. Secara otomatis EEPROM menaikkan alamat *counter* internal dan mengirimkan data pada alamat berikutnya. Sama seperti proses pembacaan diatas, untuk mengakhiri pembacaan, *master* tidak memberikan *acknowledge* setelah bit data terakhir dikirimkan oleh EEPROM, kemudian disusul dengan mengirimkan kondisi *stop*. Proses pembacaan secara berurutan dapat dilihat dalam Gambar 2.6.*figure*6.



Gambar 2.6. Operasi pembacaan pada EEPROM serial AT24C512
Sumber: Annonymous, 2004:12

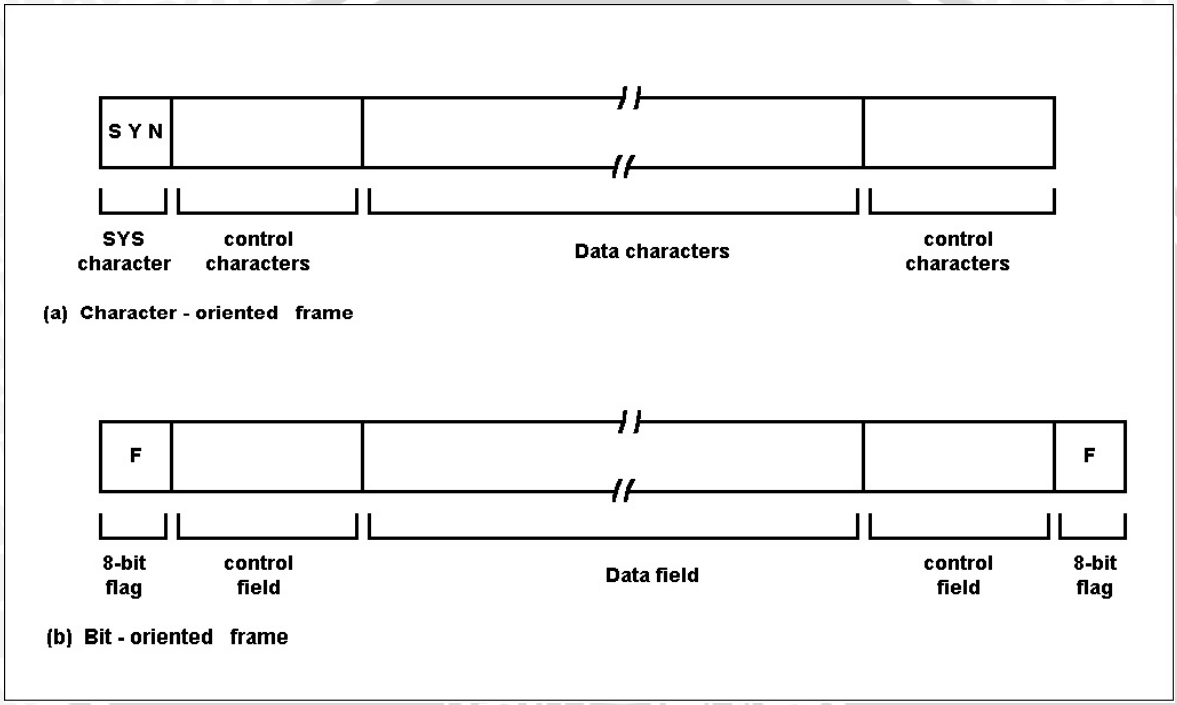
2.5 Komunikasi Data Serial

Ada dua cara untuk mentransfer data serial berdasarkan pewaktuannya, yaitu transmisi sinkron dan transmisi asinkron (Beyda, 1996: 43).

2.5.1 Transmisi Sinkron

Dalam transmisi sinkron, paket karakter dan bit dikirimkan tanpa *start bit* dan *stop bit*, serta pengiriman dan penerimaan masing-masing bit dapat diperkirakan. Untuk mencegah penyimpangan waktu antara pengirim dan penerima, *clock* keduanya harus disinkronkan, diantaranya dengan menggunakan *clock* yang berbeda antara pengirim dan penerima, namun informasi *clock* harus disertakan dalam sinyal data.

Pada transmisi sinkron, dibutuhkan tingkat pen-sinkronan lainnya, yang akan memungkinkan penerima menentukan awal dan akhir blok data. Untuk menentukan hal ini, masing-masing blok memulai dengan pola *preamble bits* (bit-bit pembuka) dan mengakhirinya dengan pola *postamble bits* (bit-bit penutup). Pola-pola ini berupa informasi kontrol. Gabungan informasi kontrol dan data disebut *frame*. Format sebuah *frame* tergantung apakah pola transmisinya berupa karakter (*character-oriented scheme*) atau bit (*bit-oriented scheme*) (William Stalling, 1991:136).



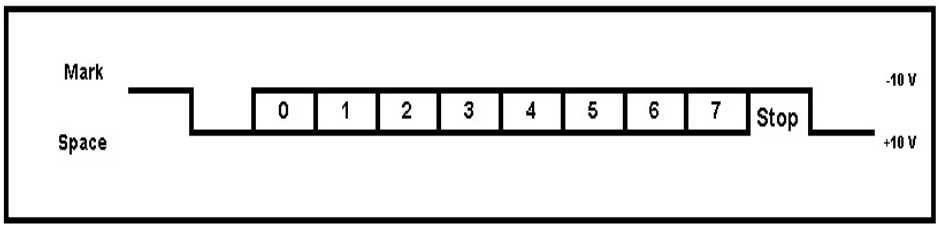
Gambar 2.7. Transmisi sinkron
Sumber : William Stalling, 1991:137

Pada transmisi *character-oriented*, blok data diperlakukan sebagai rangkaian karakter (biasanya 8 bit karakter). Semua informasi dalam bentuk karakter. *Frame* diawali dengan *synchronization characters*, yang disebut juga SYN, yaitu pola bit yang unik yang memberitahu penerima bahwa ini adalah awal dari blok. Dalam hal ini, *postamble* berupa karakter unik lainnya dalam beberapa pola. Bagian penerima siap menerima blok data dengan ditandai adanya SYN kemudian menerima data hingga ada karakter penutup.

Pada transmisi *bit-oriented*, blok data diperlakukan sebagai rangkaian bit. Baik data maupun informasi kontrol harus diterjemahkan dalam kesatuan karakter 8 bit. Suatu pola bit khusus menandakan awal blok. Pada transmisi ini, *preamble* berupa bit-bit yang panjangnya 8 bit dan disebut flag. Flag yang sama juga digunakan sebagai *postamble* (William Stalling, 1991:137).

2.5.2 Transmisi Asinkron

Pada pengiriman data serial tak sinkron, setiap karakter dikirimkan sebagai satu kesatuan bebas, yang berarti bahwa waktu antara pengiriman bit terakhir dari sebuah karakter dan bit pertama dari karakter berikutnya tidak tetap. Data dikirimkan satu karakter dalam satu waktu. Pewaktuan atau sinkronisasi dipertahankan dalam masing-masing karakter, sehingga penerima mempunyai kesempatan untuk men-sinkronkan pada awal masing-masing karakter. Setiap karakter yang dikirimkan selalu diikuti dengan bit awal (*start bit*) dengan nilai biner 0, dan bit akhir (*stop bit*) dengan nilai biner 1 (William Stalling, 1991:134).



Gambar 2.8. Pengiriman data serial tak sinkron
Sumber: Dwi Sutadi, 2002:58

Dalam Gambar 2.8 menggambarkan pulsa serial port dengan format data 8N1; yakni 8 bit data, ***No** parity*, dan 1 stop bit. Pada keadaan "menganggur", serial port berlogika *Mark* (logika 1). Jika ada transmisi data maka diawali dengan start bit yang berlogika 0. Berikutnya setiap bit sepanjang karakter dikirim satu per satu. Pengiriman data secara serial yakni dengan mengirimkan LSB (*Least Significant Bit*) terlebih dahulu, lalu diikuti MSB (*Most Significant Bit*). Untuk mengakhiri transmisi, *stop* bit berlogika 1 ditambahkan di akhir paket data.

Ada tiga cara untuk mentransfer data serial berdasarkan jalur komunikasinya, yaitu komunikasi *simplex*, *half duplex*, dan *full duplex* (Beyda, 1996: 47).

Komunikasi *simplex*

Komunikasi *simplex* adalah komunikasi satu arah. Pengirim dan penerima tugasnya tetap; satu mengirim data, yang lainnya bertugas menerima data. Contoh komunikasi *simplex* ini adalah siaran stasiun radio komersial.

Komunikasi *half duplex*

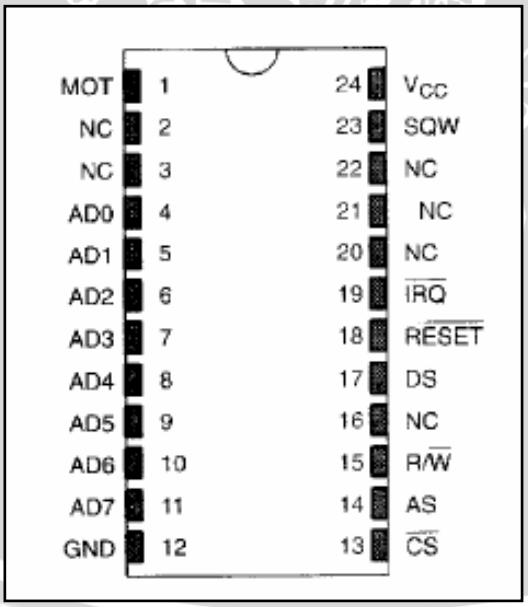
Transfer data pada komunikasi ini dilakukan dua arah, tetapi data yang dikirimkan hanya satu arah pada satu waktu. Contoh penggunaannya adalah pada *handy talkie*. Komunikasi *half duplex* disebut juga komunikasi dua kabel (*two-wire communications*), karena pada mulanya sepasang kabel digunakan untuk transmisi *half duplex*.

Komunikasi *full duplex*

Merupakan komunikasi dua arah, masing-masing peralatan dapat mengirim dan menerima data dalam satu waktu. Contoh penggunaan metode ini adalah pada percakapan telepon. Komunikasi ini sering disebut dengan komunikasi empat kabel (*four-wire communications*).

2.6 Basis Pewaktuan *Real Time Clock* (RTC) IC DS12887

DS12887 adalah IC yang dipakai sebagai basis pewaktuan buatan *Dallas Semiconductor*. Terdiri atas 24 penyemat dengan konfigurasi penyemat seperti ditunjukkan dalam Gambar 2.9.



Gambar 2.9. Konfigurasi penyemat IC DS12887
Sumber: Annonymous:1

Keistimewaan IC DS12887 antara lain adalah:

Osilator internal dan *time base* internal.

1. Menghitung detik, menit, jam dalam sehari-hari.
2. Menghitung hari dalam setiap minggu, tanggal, bulan dan tahun.
3. Seratus tahun kelender.
4. Mempunyai catu daya *back-up.*
5. Dapat tetap beroperasi selama lebih dari 10 tahun tanpa kehadiran catu daya eksternal (karena adanya baterai cadangan di dalam RTC).
6. RAM internal sebesar 64 byte, terdiri atas 14 byte untuk *clock* dan register kontrol, sedangkan 50 byte lainnya dapat digunakan oleh pemakai.

Secara keseluruhan, fungsi penyemat-penyemat RTC DS12887 adalah sebagai berikut :

Tabel 2.2 Keterangan *Pin Out* RTC DS12887

Nomor Pin	Nama Pin	Keterangan
12 dan 24	GND dan Vcc	Merupakan penyemat catu daya. Vcc dihubungkan pada catu daya +5 volt dan GND pada *ground*.
1	MOT (Motel)	Digunakan sebagai saklar pemilih mode diagram pewaktuan. Apabila dihubungkan pada Vcc berarti sistem diagram pewaktuan Motorola yang dipakai, jika dihubungkan dengan GND berarti sistem diagram pewaktuan Intel yang dipakai.
4 . . . 11	AD0–AD7 *Multiplexed Bidirectional Address/Data Bus*	Merupakan bus alamat/data dua arah yang termultipleks. Pengiriman data maupun alamat dari maupun ke RTC dilakukan melalui bus ini.
13	$\overline{CS}$ *Chip Select Input*	Merupakan masukan untuk mengaktifkan periferal RTC. Sinyal didapat dari dekoder alamat dengan alamat tertentu.
14	AS — Address Strobe Input	Merupakan masukan bagi sinyal yang digunakan untuk memisahkan bus data dan bus alamat (ALE). Tepi turun AS/ALE akan menyebabkan alamat ditahan dalam DS12887. Tepi naik berikutnya yang terjadi pada AS akan meniadakan alamat tersebut tanpa

		memperhatikan apakah penyemat $\overline{CS}$ aktif atau tidak.
15	*READ* / $\overline{WRITE}$ *Input*	Penyemat $R/\overline{W}$ dapat diartikan sama seperti sinyal *Write Enable* ($\overline{WE}$) yang biasa dipakai pada komponen memori jenis RAM. Digunakan untuk proses penulisan data pada RAM internal RTC.
17	DS — *Data Strobe or Read Input*	Penyemat DS dapat diartikan sama seperti sinyal *Output Enable* (OE) pada komponen memori. Dihubungkan dengan sinyal $\overline{WR}$ yang berasal dari mikrokontroler untuk melakukan proses membaca data pada RAM internal RTC.
18	$\overline{\textbf{RESET}}$ — *Reset Input*	Sinyal $\overline{RESET}$ diberikan dengan memberikan logika rendah selama waktu yang dispesifikasikan, dan tidak berpengaruh pada unjuk kerja *clock*, kalender, dan fungsi RAM. Namun sinyal $\overline{RESET}$ dapat mengakibatkan beberapa flag di-*reset* menjadi nol.
19	$\overline{\textbf{IRQ}}$ — *Interrupt Request Output*	Sinyal $\overline{IRQ}$ merupakan sinyal aktif rendah, yang dapat digunakan untuk menginterupsi mikrokontroler. Keluaran $\overline{IRQ}$ tetap rendah selama status bit yang menyebabkan interupsi ada. Untuk me-*reset* $\overline{IRQ}$, mikrokontroler membaca register C RTC atau me-*reset* RTC. Saat tidak terdapat interupsi, penyemat ini akan dalam kondisi impedansi tinggi (*high-impedance*).
23	SQW — *Square Wave Output*	Merupakan penyemat yang dapat dipilih untuk mengeluarkan satu dari 13 frekuensi keluaran yang tersedia. Besar frekuensi keluaran SQW dapat diubah

		dengan diprogram pada Register A. Untuk mengaktifkan atau me-nonaktifkan sinyal SQW dipilih lewat bit SQWE pada Register B.

Sumber: Annonymous:4-5

2.6.1 Catu Daya RTC

Berkat adanya baterai *Lithium* dalam DS12887, IC ini akan tetap mencatat waktu meskipun catu daya yang disalurkan lewat kaki Vcc (kaki 24) terputus. Dalam keadaan kehilangan Vcc, informasi yang tersimpan dalam CMOS RAM tetap bertahan dengan baik. Saat mulai menerima catu daya, dalam waktu 200 milidetik setelah tegangan Vcc mencapai lebih dari 4,25 volt, catatan waktu dan informasi yang disimpan di CMOS RAM dalam DS12887 bisa diambil.

Jika tegangan pada Vcc kurang dari 4,25 volt, CS *(chip select*) di dalam IC dipaksa berlogika tinggi tanpa mempedulikan keadaan tegangan pada kaki CS (kaki nomor 13). Dalam keadaan semacam ini informasi tidak bisa diisikan ke dalam IC. Mekanisme semacam ini sangat penting untuk menjamin isi IC tidak berubah pada saat catu daya dalam keadaan tidak stabil. Saat tegangan Vcc kurang dari 3 volt, catu daya yang disalurkan lewat kaki Vcc dilepas, dan IC bekerja dengan sumber daya dari baterai Lithium. RTC DS 12887 membutuhkan arus maksimal 15 mA, sedangkan arus *typical*-nya sekitar 7 mA.

2.6.2 Peta Memori

Dari sudut pandang pemrograman, IC DS12887 terlihat sebagai 64 byte memori data. Memori data tersebut dipakai untuk mencatat waktu, penentu alarm, dan register pengatur kerja.

Waktu dicatat dalam DETIK (memori-data nomor 0), MENIT (nomor 2), JAM (nomor 4), HARI (nomor 6), TANGGAL (nomor 7), BULAN (nomor 8), TAHUN (nomor 9) dan ABAD (nomor 50). Data catatan waktu bisa diambil dalam format data biner maupun dalam format data BCD (*Binary Coded Decimal*), misalnya bulan Januari selalu dinyatakan dengan 01h, bulan Oktober dinyatakan dengan 0Ah dalam format biner, dan dinyatakan dengan 10h dalam format BCD.

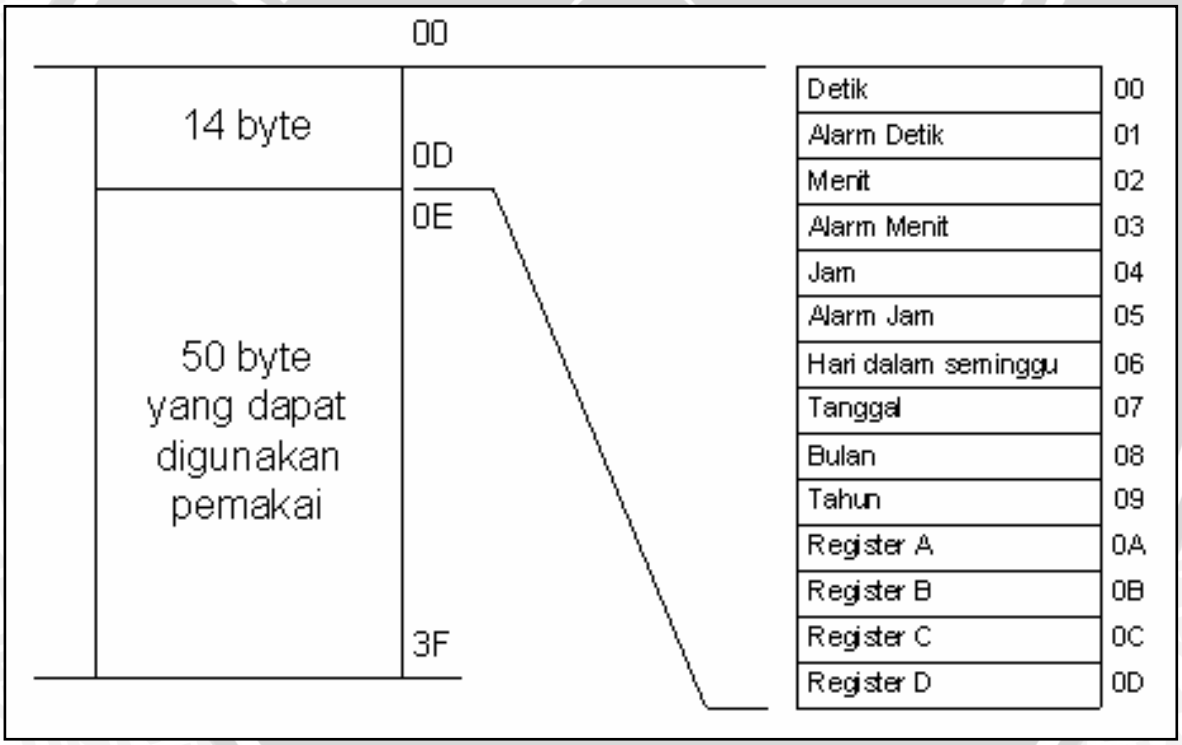
Pencatatan JAM bisa dilakukan dalam format 24 jam per hari, atau 12 jam am/pm. Tidak ada tanda khusus untuk menyatakan am/pm, nilai 01-12 (BCD) dipakai untuk

menyatakan jam 1 am sampai 12 am, sedangkan jam 1 pm dinyatakan dengan 81h (BCD) dan jam 12 pm dinyatakan dengan 92h (BCD).

Penentuan waktu alarm diatur lewat ALARM DETIK (memori-data nomor 1), ALARM MENIT (nomor 3), dan ALARM JAM (nomor 5). Jika DETIK, MENIT dan JAM sama nilainya dengan waktu alarm yang dikehendaki, kaki $\overline{IRQ}$ menjadi nol, dan dipakai sebagai sinyal permintaan Interupsi bagi prosesor.

Memori nomor 10 sampai 13 dipakai sebagai Register pengatur kerja DS12887 dan register pemantau kerja dari DS12887. Register-register tersebut dinamakan sebagai Register A sampai Register D. Pemakaian masing-masing bit dalam register-register tersebut dirinci dalam Gambar 2.11 sampai 2.13.

Memori data yang masih tersisa merupakan memori bebas yang bisa dipakai untuk segala macam keperluan. Peta alamat RAM IC DS12887 ditunjukkan dalam Gambar 2.10.



Gambar 2.10. Peta alamat RAM internal r*eal time clock* DS12887
Sumber: Annonymous:6

2.6.2.1 REGISTER A

Bit 7 dari register A dinamakan sebagai UIP (*Update In Progress*). Nilai 1 pada UIP menyatakan data catatan waktu dalam DS12887 segera akan diperbarui. Jadi pengambilan data terkini seharusnya dilakukan pada saat UIP='0'.

DV2, DVI dan DVO (bit 6 sampai 4) dipakai untuk mengatur kerja osilator kristal. Nilai '010' pada tiga bit ini menghidupkan rangkaian osilator dan mengaktifkan sistem pencatatan waktu. Nilai '110' atau '111' menghidupkan rangkaian osilator, tetapi sistem pencatat waktu tidak bekerja. Nilai-nilai lainnya mematikan rangkaian osilator. Setelah diproduksi, osilator dalam IC dimatikan, agar baterai Lithium tidak terpakai sia-sia.

RS3, RS2, RS1, dan RSO (bit 3 sampai 0) berfungsi untuk menentukan frekuensi sinyal kotak yang dibangkitkan pada kaki SQW (kaki nomor 23). Frekuensi yang dibangkitkan sesuai dengan kombinasi biner dari keempat bit ini.

Register A (CMOS RAM No. 10)

D7	D6	D5	D4	D3	D2	D1	D0
UIP	DV2	DV1	DV0	RS3	RS2	RS1	RS0

Gambar 2.11. Denah register A
Sumber: Annonymous:11

2.6.2.2 REGISTER B

SET (bit 7) dipakai untuk menghentikan pencatatan waktu. Saat SET bernilai '0', proses pencatatan waktu berjalan secara normal, yaitu catatan waktu berubah setiap detik sekali. Pada saat catatan waktu dirubah secara manual untuk keperluan penyesuaian waktu, SET di-satu-kan. Saat penyesuaian waktu selesai, SET kembali di-nol-kan.

PIE (*Periodic Interrupt Enable*)=1 pada bit 6 mengaktifkan sistem interupsi secara periodik. Periode interupsi yang dibangkitkan ditentukan oleh kombinasi nilai biner pada RS3 sampai RS0 pada Register A. AIE (*Alarm Interrupt Enable*)=1 pada bit 5 mengaktifkan sistem interupsi alarm. UIE (*Update Ended Interrupt Enable*)=1 pada bit 4 mengaktifkan sistem interupsi karena adanya perubahan nilai pencatatan waktu. SQWE (*Square Wave Enable*)=1 pada bit 3 membangkitkan sinyal kotak pada kaki SQW (kaki nomor 23). DM (*Data Mode*) pada bit 2 menentukan format data pencatatan waktu. DM=0 pencatatan waktu dilakukan dengan format biner, DM=l pencatatan waktu dilakukan dengan format BCD. 24/12 (bit 1) menentukan format pencatatan JAM, 24/12=0 pencatatan JAM dilakukan dengan format 0..12, 24/12='l' pencatatan JAM dilakukan dengan format 0..24.

Register B (CMOS RAM No. 11)							
D7	D6	D5	D4	D3	D2	D1	D0
SET	PIE	AIE	UIE	SQWE	DM	24/12	DSE

Gambar 2.12. Denah register B
Sumber: Annonymous:12

2.6.2.3 REGISTER C

IRQF (*Interrupt Request Flag*) pada bit 7 dipakai untuk menandai DS12887 sedang mengajukan permintaan interupsi pada prosesor dengan cara me-nol-kan kaki $\overline{IRQ}$ (kaki nomor 19). Setelah prosesor membaca isi Register C, IRQF dan bit-bit lain dalam register C berubah menjadi ‘0’ dan kaki $\overline{IRQ}$ menjadi ‘1’.

PF (*Periodic Interrupt Flag*) pada bit 6 menjadi '1' secara periodik, periodenya ditentukan oleh kombinasi nilai biner pada RS3...RSO dalam Register A. Jika PIE dalam Register B bernilai ‘1’, dan saat PF bernilai '1', DS12887 mengajukan permintaan interupsi pada prosesor dengan cara me-nol-kan kaki $\overline{IRQ}$ (kaki nomor 19).

AF (*Alarm Interrupt Flag*) pada bit 5 menjadi ‘1’ jika waktu saat ini sudah sama dengan waktu alarm. Jika saat itu AIE dalam Register B bernilai ‘1’ pula, DS12887 mengajukan permintaan interupsi dengan me-nol-kan kaki $\overline{IRQ}$ (kaki nomor 19).

UF (*Update Ended Interrupt Flag*) pada bit 4 menjadi ‘1’ sekali dalam 1 detik, yakni saat ada perubahan pencatatan waktu. Jika saat itu UIE dalam Register B bernilai '1' pula, DS12887 mengajukan permintaan interupsi dengan me-nol-kan kaki $\overline{IRQ}$ (kaki nomor 19).

Register C (CMOS RAM No. 12)							
D7	D6	D5	D4	D3	D2	D1	D0
IRQF	PF	AF	UF	0	0	0	0

Gambar 2.13. Denah register C
Sumber: Annonymous:13

Untuk menginisialisasi RTC, yaitu men-*set* waktu yang ada di RAM RTC saat kali pertama RTC diaktifkan, digunakan sinyal $\overline{WR}$ (*write*) yang dihubungkan pada penyemat $R/\overline{W}$, data yang diinginkan pada alamat yang bersesuaian. Sedangkan untuk membaca data

dari RAM internal RTC digunakan sinyal $\overline{RD}$ (*read*) yang dihubungkan pada penyemat DS (*data strobe*).

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB III
METODOLOGI

Penyusunan skripsi ini didasarkan pada masalah yang bersifat aplikatif, yaitu perencanaan dan perealisasian alat agar dapat menampilkan unjuk kerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah. Data dan spesifikasi komponen yang digunakan dalam perencanaan merupakan data sekunder yang diambil dari buku data komponen elektronika. Pemilihan komponen berdasarkan perencanaan dan disesuaikan dengan komponen yang ada di pasaran.

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat adalah sebagai berikut:

3.1 Studi Literatur

Mengumpulkan dan mempelajari literatur baik bersifat primer yaitu buku-buku yang berhubungan dengan mikrokontroler AT89C51, media penyimpan data, serial EEPROM, konsep komunikasi data, RTC (*real time* clock) IC DS12887 maupun yang bersifat sekunder yaitu data-data komponen yang nantinya akan digunakan.

3.2 Interview

Melakukan tanya jawab mengenai materi yang berhubungan dengan pembuatan alat telemetri pada stasiun pengamat cuaca di Karangkates, Jawa Timur.

3.3 Perancangan dan Pembuatan Alat

3.3.1 Spesifikasi Alat

Menentukan spesifikasi alat yang akan dirancang dengan memperhatikan berbagai informasi yang telah terkumpul dalam studi literatur. Spesifikasi Alat yang akan dibuat adalah:

Memori yang dipakai adalah jenis AT24C512 dengan kapasitas 512 Kilobit atau 64 KiloBytes *compatibel* dengan semua mikrokontroller Atmel tipe 8051.

1 Data keluar atau masuk kedalam memori secara serial melalui jalur *serial data line* (SDA) dengan menggunakan protokol I2C yang dikembangkan oleh PHILIPS dan mampu mentransfer data secara dua arah (*bi-directional*).

2 Satu paket data yang disimpan dalam memori EEPROM serial berasal dari unit pengontrol sensor sejumlah 9 byte ditambah dengan data dari RTC (*real time clock*) DS12887 buatan Dallas Semiconductor sejumlah 5 byte.

3 Penyimpan data menerima data dari unit pengontrol sensor sejumlah 9 byte yang terdiri atas data arah angin satu byte, data kecepatan angin dua byte, data suhu dua byte, data kelembaban dua byte, dan data curah hujan dua byte.

4 RTC (*real time clock*) DS12887 menyediakan data waktu kapan unit pengontrol sensor mengirim data yang terdiri atas 5 byte yaitu data menit, jam, tanggal, bulan, dan tahun masing-masing satu byte.

5 Keluaran dari unit penyimpan data berupa data yang akan dikirimkan berkala pada unit stasiun pusat (*base Station*).

6 Menggunakan mikrokontroler AT89C51 sebagai pengendali utama pada Sistem Penyimpan data.

7 Sistem mikrokontroler, sistem memori,dan RTC menggunakan catu tegangan DC +5 V.

3.3.2 Perancangan Alat

Pada tahap perancangan dibuat suatu blok diagram fungsional dari rangkaian yang direncanakan. Perancangan rangkaian dilakukan pada tiap-tiap blok untuk mempermudah perancangan serta penentuan nilai komponen yang digunakan. Selanjutnya masing-masing blok digabung menjadi suatu rangkaian lengkap. Perancangan alat didasarkan pada teori yang ada dan data-data komponen yang nantinya akan digunakan dalam pembuatan sistem. Pemilihan komponen-komponen didasarkan pada perancangan dan disesuaikan dengan komponen yang ada di pasaran.

3.3.3 Pembuatan Alat

Pada tahap ini, pertama-tama dilakukan realisasi dengan menggunakan *project board*. Hal ini dilakukan untuk memastikan rangkaian dengan komponen yang ada mendekati atau sama dengan yang dirancang. Kemudian dilanjutkan dengan pembuatan PCB dan merangkai komponen-komponen di atas PCB tersebut.

3.4 Pengujian dan Analisis Data

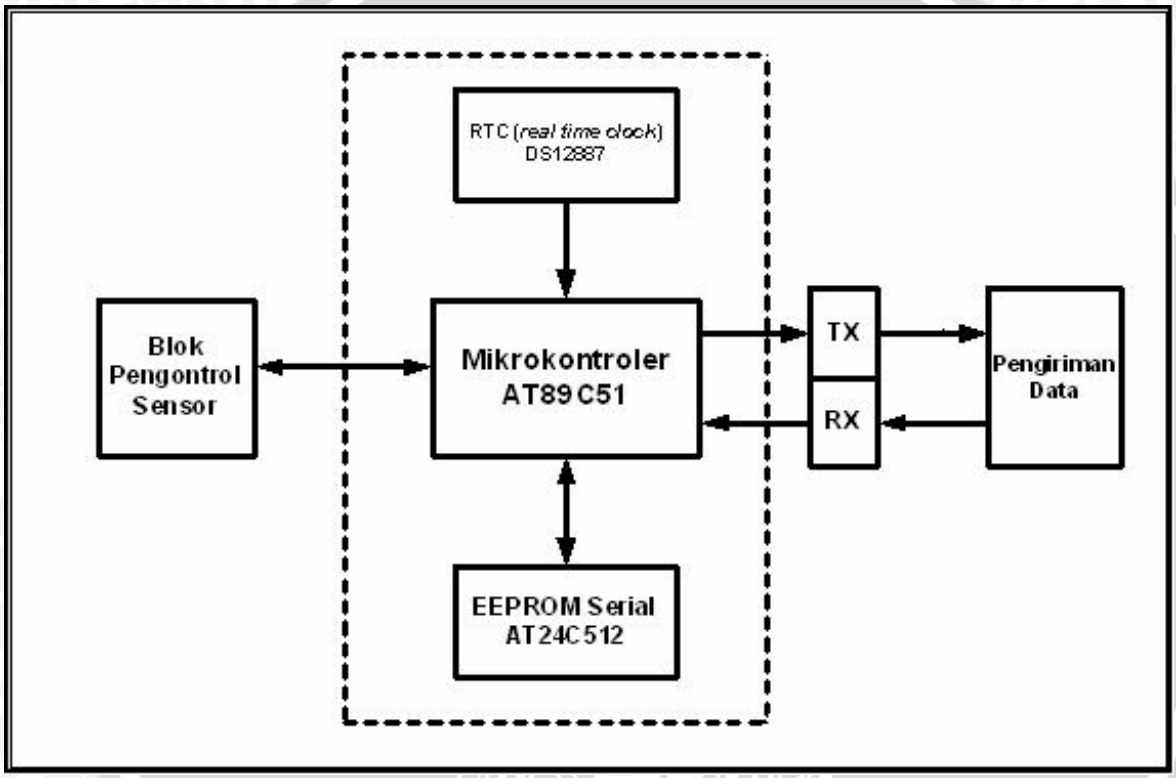
Tahap selanjutnya adalah pengujian dan analisis, yaitu menguji setiap blok, kemudian dibandingkan dengan teori yang ada sehingga didapatkan suatu kesimpulan tentang peralatan yang dibuat, selanjutnya dilakukan pengujian keseluruhan sistem untuk mengetahui hasil rancangan sesuai dengan spesifikasi atau tidak.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB IV
PERENCANAAN DAN PEMBUATAN ALAT

Bagian ini menjelaskan tentang perencanaan dan pembuatan sistem mikrokontroler AT89C51, *Real Time Clock* (RTC), EEPROM Serial AT24C512. Perencanaan dan pembuatan sistem aplikasi mikrokontroler AT89C51 meliputi perangkat keras dan perangkat lunak

4.1 GAMBARAN UMUM



Gambar 4.1. Blok diagram sistem penyimpan data

Penyimpan data pada sistem telemetri stasiun pengamat cuaca dengan sensor kelembaban, temperatur, curah hujan, arah angin, dan kecepatan angin yang dapat diakses lewat internet berfungsi untuk menyimpan data-data yang didapat dari pengontrol sensor. Penyimpan data berinteraksi dengan pengontrol sensor untuk mendapatkan data parameter cuaca. Oleh pengontrol sensor, data yang diperoleh akan diberikan ke penyimpan data secara serial dengan media penyimpanan adalah memori dengan jenis EEPROM (*electrical erasable programable read only memory*) serial. Setelah beberapa saat, penyimpan data mengirim data ke stasiun pusat, penyimpan data menyiapkan data-data yang telah

tersimpan agar siap dikirimkan lewat jaringan internet. Penyimpan data juga berperan sebagai penghubung antara stasiun pusat (*base station*) dengan pengontrol sensor.

Pada sistem terdapat RTC yang digunakan untuk mengetahui data waktu kapan data parameter cuaca yang diambil oleh pengontrol sensor dikirim ke penyimpan data. Sistem komunikasi yang dilakukan antara penyimpan data dengan pengontrol sensor adalah komunikasi data sinkron.

Fungsi dari blok-blok dalam Gambar 4.1 adalah sebagai berikut, mikrokontroler adalah bagian utama dari alat yang direncanakan. Mikrokontroler digunakan untuk meminta data dan berkomunikasi dengan unit pengontrol sensor, menyimpan data ke dalam memori, mengambil data dari RTC, mengirimkan data ke stasiun pusat.

Serial EEPROM digunakan untuk menyimpan data. Dengan menggunakan penyimpan data yang terpisah dengan mikrokontroler diharapkan mempermudah dalam mengubah/meng-*update* isinya. Cara penulisan dan pembacaan data pada memori menggunakan protokol transfer data I2C yang dikembangkan oleh PHILIPS dan mampu mentransfer data dua arah (*bi-directional*). Untuk Serial EEPROM terdapat dua standar yang harus dipilih. Standar tipe 93CXX yang menggunakan bus Microwire (3 kaki) dan standar tipe 24CXX yang menggunakan bus I2C (2 kaki). Walaupun bus Microwire yang menggunakan tiga kaki (DI, DO, dan CLK) bisa disederhanakan lagi menjadi dua kaki, namun orang lebih banyak memilih pemakaian bus I2C yang memiliki dua kaki (SDA dan SCL). Pertimbangannya karena tipe I2C memiliki varian yang banyak dengan spasi besar, juga karena support yang cukup besar, terutama dari Philips yang memiliki IC khusus untuk konverter dari bus paralel 8bit ke bus I2C 2bit. Kelebihan utama dari sistem bus I^2C ini adalah pemakaian hanya dua kabel *bidirectional*. Satu sebagai SDA (*Serial Data*) dan yang lainnya sebagai SCL (*Serial Clock*). Karena efisiensi pemakaian hanya dua kabel ini, maka sistem bus I^2C akan mengurangi spasi PCB dan pemakaian jumlah kaki IC sehingga menurunkan biaya produksi. RTC digunakan untuk memberikan data menit, jam, tanggal, bulan dan tahun.

4.2 SPESIFIKASI ALAT

Hal pertama yang perlu dilakukan sebelum melakukan perancangan dan pembuatan alat adalah penentuan spesifikasi alat yang akan dibuat. Adapun spesifikasi alat yang akan dibuat adalah sebagai berikut:

1. Memori yang dipakai adalah jenis AT24C512 dengan kapasitas 512 Kilobit atau 64

KiloBytes *compatibel* dengan semua mikrokontroller Atmel tipe 8051.

2. Data keluar atau masuk kedalam memori secara serial melalui jalur *serial data line* (SDA) dengan menggunakan protokol I2C yang dikembangkan oleh PHILIPS dan mampu mentransfer data secara dua arah (*bi-directional*).
3. Satu paket data yang disimpan dalam memori EEPROM serial berasal dari unit pengontrol sensor sejumlah 9 byte ditambah dengan data dari RTC (*real time clock*) DS12887 buatan Dallas Semiconductor sejumlah 5 byte.
4 Penyimpan data menerima data dari unit pengontrol sensor sejumlah 9 byte yang terdiri atas data arah angin satu byte, data kecepatan angin dua byte, data suhu dua byte, data kelembaban dua byte, dan data curah hujan dua byte.
6. RTC (*real time clock*) DS12887 menyediakan data waktu kapan unit pengontrol sensor mengirim data yang terdiri atas 5 byte yaitu data menit, jam, tanggal, bulan, dan tahun masing-masing satu byte.
7. Keluaran dari unit penyimpan data berupa data yang akan dikirimkan berkala pada unit stasiun pusat (*base Station*).
8. Menggunakan mikrokontroler AT89C51 sebagai pengendali utama pada Sistem Penyimpan data.
9. Sistem mikrokontroler, sistem memori, dan RTC menggunakan catu tegangan DC +5 V.

4.3 PERENCANAAN PERANGKAT KERAS

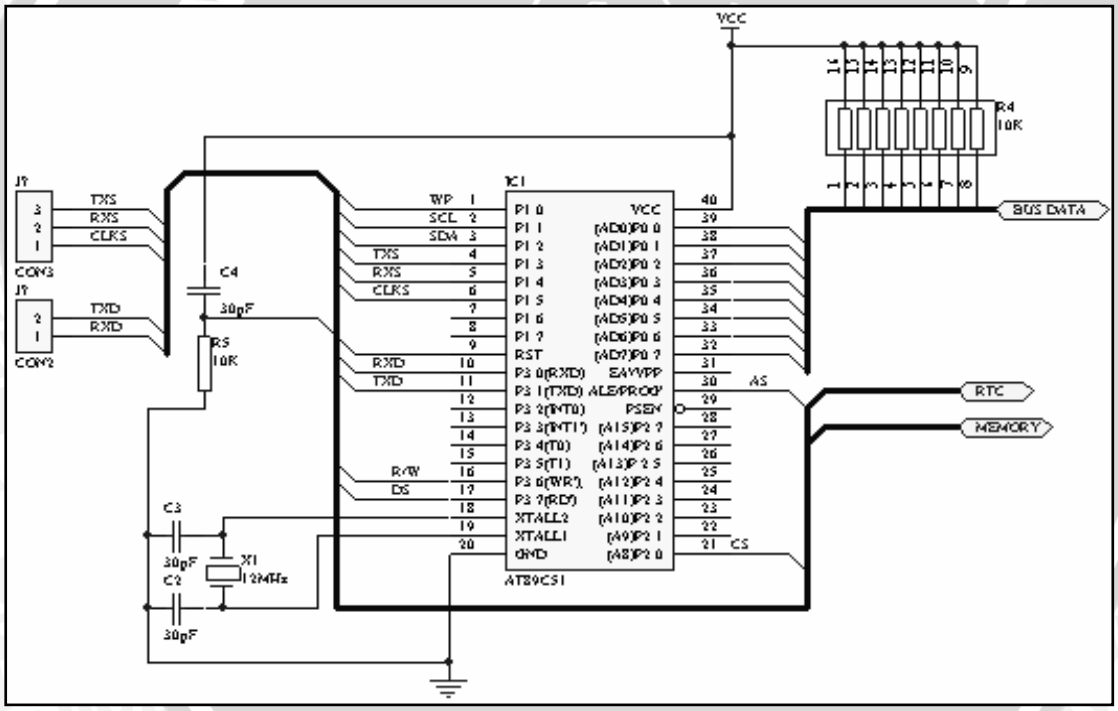
Bagian ini menguraikan rincian perencanaan perangkat keras yang meliputi perencanaan:

1 Sistem mikrokontroler AT89C51.
2 Antarmuka mikrokontroler ke RTC.
3 Antarmuka mikrokontroler dengan EEPROM serial AT24C512.

4.3.1 SISTEM MIKROKONTROLER AT89C51

Sistem mikrokontroler AT89C51 ini berfungsi sebagai pengatur aliran data dan sinyal-sinyal kontrol. Port-port mikrokontroler digunakan sebagai jalur data, jalur alamat dan sinyal-sinyal kontrol. Mikrokontroler ini menggunakan mode *single chip operation*, sehingga tidak memerlukan *external memory* untuk menyimpan perangkat lunak.

Pada perancangan sistem mikrokontroler, port 0 digunakan sebagai jalur data untuk mengakses RTC, dengan cara menghubungkan pin pada port 0 dengan AD0-AD7 pada RTC. RTC juga memerlukan sinyal-sinyal kontrol yaitu CS (*chip select*) yang dihubungkan dengan pin P2.0, pin ALE pada mikrokontroler dihubungkan dengan AS (*address strobe*), pin P3.6 (WR') dihubungkan dengan R/W pada RTC, dan pin P3.7 (RD') dihubungkan dengan DS (*data strobe*) pada RTC. Mikrokontroler menyediakan 3 pin yaitu pin P1.3, pin P1.4, dan pin P1.5 untuk komunikasi dengan unit pengontrol sensor, yaitu TX, RX, dan Clock. Untuk mengakses memori baik menulis data maupun membaca data, mikroprosessor menyediakan 2 pin, P1.1 dan P1.2. Untuk P1.1 dihubungkan dengan jalur SCL pada EEPROM Serial, sedangkan P1.2 dihubungkan dengan jalur SDA pada EEPROM Serial. Untuk mengirim data pada stasiun pusat disediakan pin P3.0 (RXD) dan P3.1 (TXD) pada mikrokontroler. Gambar 4.2 menunjukkan rangkaian mikrokontroler AT89C51pada sistem yang akan diantarmukakan ke EEPROM Serial, dan modul RTC .



Gambar 4.2. Perancangan rangkaian mikrokontroler AT89C51

Untuk membuat sistem mikrokontroler ini bekerja dibutuhkan sedikit komponen tambahan. Kristal dengan frekuensi 12 Mhz serta dua buah kapasitor 30 pF digunakan untuk menggerakkan osilator internal. Mikrokontroler menggunakan rangkaian *power-on reset*, yang me-*reset* mikrokontroler secara otomatis setiap kali catu daya dinyalakan. Rangkaian *power-on reset* tersusun dari sebuah kapasitor 10 uF dan sebuah resistor 10 KΩ. Saat catu daya dinyalakan, rangkaian *reset* akan menahan logika tinggi untuk beberapa saat

pada pin RST. Logika tinggi pada pin RST harus bertahan selama lebih dari dua siklus mesin untuk menjamin keabsahan.

Port serial pada mikrokontroler AT89C51 bersifat *full duplex*, artinya port serial dapat menerima dan mengirim pada waktu bersamaan. Penerimaan dan pengiriman data port serial melalui register SBUF. Pembacaan dan penulisan pada register tersebut melalui *Accumulator*. Port serial harus diberikan pengaturan mode kerja terlebih dahulu. Pada sistem ini port serial menggunakan timer 1 mode 2. Pengaktifan mode ini melalui register TMOD diberikan nilai 20H. Tabel 4.1. menunjukkan pengaturan register dan nilai-nilai yang diberikan.

Tabel 4.1. Data register pengaturan port serial dan besar nilainya

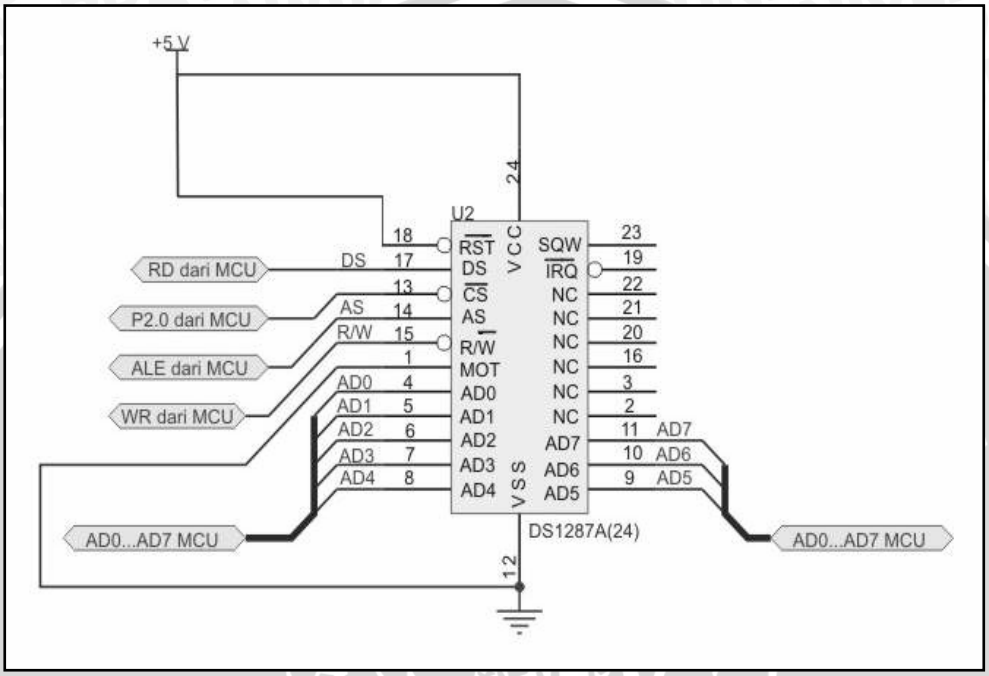
Register	*Nilai*
TMOD	20H
TH1	0FDH
SCON	50H

Pemberian nilai 50H pada register SCON dimaksudkan untuk mengaktifkan bit SM0=0 dan SM1=1, maka port serial bekerja pada mode *UART* 8-bit. Pemberian nilai 1 pada bit *REN* akan mengaktifkan port serial agar dapat memulai penerimaan data. TH1 diberikan nilai sebesar 0FDH dimaksudkan agar port serial bekerja dalam baudrate sebesar 9600 bps. Selama program berjalan pada saat penerimaan data dapat diketahui dengan adanya interrupt. Ketika Interrupt, program mikrokontroler akan menuju vektor interrupt pada alamat 23H dan menjalankan instruksi-instruksi selnjutnya mulai pada alamat 23H tersebut. Ketika terjadi interrupt tersebut bit *RI* pada register *SCON* berubah menjadi '1'. Setelah data yang diterima pada register *SBUF* diambil, maka bit *RI* harus diberikan nilai '0'. Fungsi pengiriman data melalui port serial, mikrokontroler memerikan nilai atau data dari *Accumulator* ke register *SBUF* dan memberikan nilai '1' pada bit *TI* pada register *SCON*. Proses pengiriman data telah selesai jika isi bit *TI* sama dengan '0'.

4.3.2 RANGKAIAN ANTARMUKA MODUL RTC

Sistem mikrokontroler yang digunakan adalah mikrokontroler buatan Intel, jadi pin MOT pada RTC dihubungkan ke GND. Bus data dan bus alamat RTC dihubungkan ke bus data sistem mikrokontroler AT89C51 yaitu pada port 0. Bus kontrol $R/\overline{W}$ RTC

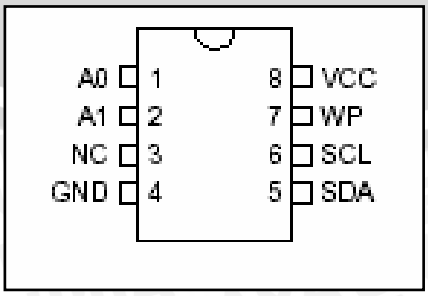
dihubungkan ke pin $\overline{WR}$ mikrokontroler AT89C51, pin DS RTC dihubungkan ke pin $\overline{RD}$ mikrokontroler AT89C51, dan pin AS RTC dihubungkan ke pin ALE mikrokontroler AT89C51. *Chip select* ($\overline{CS}$) RTC dihubungkan ke pin P2.0 mikrokontroler AT89C51. Pin $\overline{RST}$ RTC dihubungkan dengan VCC karena $\overline{RST}$ aktif pada logika rendah. Antarmuka RTC dengan sistem mikrokontroler AT89C51 ditunjukkan dalam Gambar 4.3.



Gambar 4.3. Rangkaian antarmuka RTC dengan sistem AT89C51

4.3.3 Serial EEPROM

SEEPROM digunakan untuk menyimpan data. Data tersebut diperoleh dari unit pengontrol sensor. Data yang disimpan dalam SEEPROM adalah data parameter cuaca yang meliputi kelembaban, temperatur, curah hujan, kecepatan angin, dan arah angin dalam bentuk ASCII. SEEPROM yang digunakan adalah IC AT24C512. Gambar 4.4. menunjukkan IC SEEPROM AT24C512.



Gambar 4.4. IC SEEPROM AT24C512
Sumber: Annonymous, 2004:1

Tabel 4.2. Tabel Fungsi Pin IC AT24C512

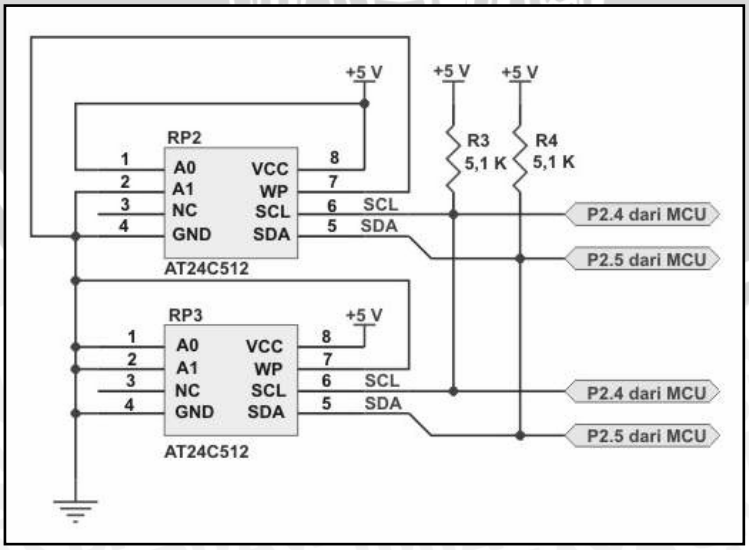
Pin Name	Function
A0 – A1	Address Inputs
SDA	Serial Data
SCL	Serial Clock Input
WP	Write Protect

Sumber: Annonymous, 2004:1

AT24C512 memiliki kapasitas penyimpanan sebesar 524288 bit yang terbagi menjadi 65536 byte. Pada jenis memori serial SEEPROM AT24C512 maksimal bisa menggunakan empat buah IC dalam pemakaian bersama bus SDA (data) dan SCL (clock). Sedangkan gambaran fungsi pin–pin pada AT24C512 adalah sebagai berikut.

- *Serial Clock* (SCL) : Sinyal clock SCL tepi naik digunakan untuk menuliskan data ke dalam EEPROM sedangkan sinyal SCL tepi turun digunakan untuk membaca data dari EEPROM.
- *Serial Data* (SDA) : Pin SDA merupakan pin dua arah digunakan untuk transfer data.
- *Device/Page Addresses* (,A1,A0) : dua buah pin ini digunakan sebagai alamat EEPROM. Apabila ketiga buah pin ini tidak dihubungkan maka akan dianggap mempunyai logika 0.
- *Write Protect* (WP): Apabila ingin menulis data ke dalam EEPROM maka pin ini harus dihubungkan dengan GND, sedangkan apabila pin ini dihubungkan dengan Vcc maka proses penulisan akan terhalangi/ tidak dapat dilakukukan.

Gambar 4.5 menunjukkan rangkaian serial EEPROM AT24C512 dengan pin-pin mikrokontroller AT89C51.

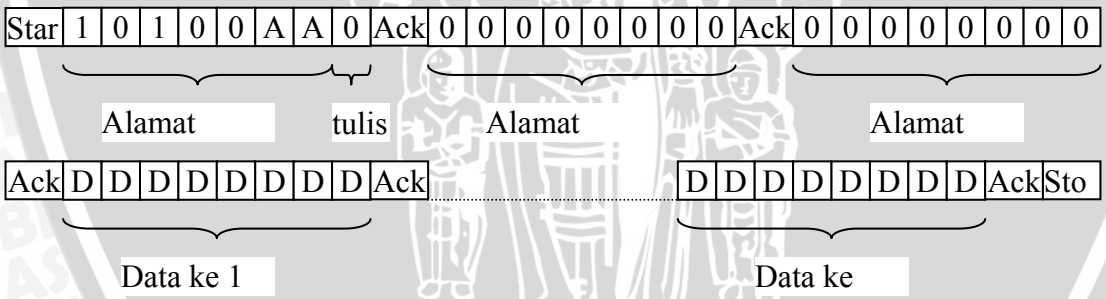


Gambar 4.5. SEEPROM AT24C512 dengan pin-pin mikrokontroller AT89C51

Dalam perencanaan alat pin 1 dan 2 pada SEEPROM RP3 dihubungkan dengan *ground*,sedang pada SEEPROM yang lainnya pin 1 dihubungkan dengan VCC,sedang pin 2 dihubungkan dengan *ground*. Dengan demikian SEEPROM yang pertama mempunyai alamat 00, sedang SEEPROM kedua mempunyai alamat 10. alamat dua IC perlu dibedakan agar tidak terjadi kesalahan dalam penulisan pada memori. Pada SEEPROM, pin WP dihubungkan dengan *ground*, sehingga akan selalu berada pada level logika 0. Dengan demikian alamat SEEPROM ini adalah 00 dan 10 serta selalu dapat ditulisi karena pin *write protect* (WP) dihubungkan dengan *ground*. Pin SCL pada SEEPROM dihubungkan dengan pin P1.1pada mikrokontroler, sedangkan pin SDA pada SEEPROM dihubungkan dengan pin P1.2pada mikrokontroler. Pin SCL serta SDA memerlukan resistor *pull-up*, hal ini dikarenakan karakteristik dari pin – pin tersebut yang merupakan pin dengan tipe *open drain*. Pada perancangan resistor *pull-up* R_3 serta R_4 bernilai 5,1 kΩ.

4.3.3.1 Operasi Penulisan

Untuk melakukan operasi penulisan data, mikrokontroller harus mengirimkan alamat dengan format seperti yang ditunjukkan dalam Gambar 4.6, dari gambar tesebut mikrokontroler harus mengirimkan sinyal – sinyal berikut :



Gambar 4.6. Operasi penulisan data pada SEEPROM
Sumber: Annonymous, 2004:11

Keterangan :

Start : Merupakan sinyal tepi turun SDA pada saat SCL tinggi

Ack : Sinyal logika rendah yang dikirim oleh serial EEPROM ke mikrokontroler

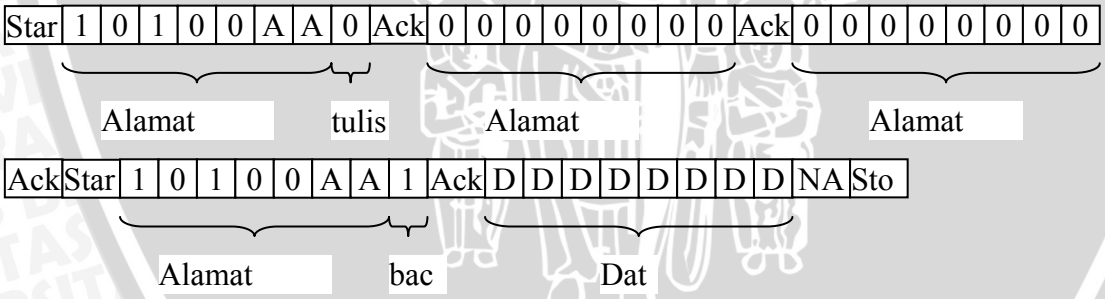
Stop : Merupakan sinyal tepi naik SDA pada saat SCL tinggi

A0,A1 : Merupakan alamat dari IC SEEPROM

Dari Gambar 4.6 dapat kita uraikan cara penulisan data ke dalam SEEPROM AT24C512 dengan memberi bit *start* yang merupakan sinyal tepi turun pada jalur SDA pada saat jalur SCL berlogika tinggi. Setelah sinyal *start* terkirim, kita kirim 8 bit yang berisi 4 bit msb pertama sebagai alamat *device* (alat) dimana SEEPROM memiliki kode alat 1010. 3 bit berikutnya sebagai pemilihan *chip device*, karena SEEPROM AT24C512 maksimal 4 buah *device* dalam 1 pemakaian *bus*, maka AT24C512 hanya memiliki A0 dan A1 untuk pemilihan *chip*. Bit kelima diberi logika 0, sedang bit 6 dan bit 7 digunakan untuk pemilihan *chip* A0 dan A1. bit LSB digunakan untuk menentukan apakah proses selanjutnya adalah proses baca atau tulis. Untuk proses tulis membutuhkan logika 0, sedangkan untuk proses baca adalah logika 1. Sinyal Ack dikirim oleh SEEPROM ke mikrokontroler. Mikrokontroler kemudian mengirimkan 2 kali 8 bit alamat data dalam memori. Setelah Ack diterima mikrokontroler, data yang akan dituliskan ke memori bisa ditulis.

4.3.3.2 Operasi Pembacaan

Untuk melakukan operasi pembacaan data, mikrokontroller harus mengirimkan alamat dengan format seperti yang ditunjukkan dalam Gambar 4.7, dari gambar tesebut mikrokontroler harus mengirimkan sinyal – sinyal berikut :



Gambar 4.7. Operasi pembacaan data pada SEEPROM

Sumber: Annonymous, 2004:12

Keterangan :

Start : Merupakan sinyal tepi turun SDA pada saat SCL tinggi

Ack : Sinyal logika rendah yang dikirim oleh serial EEPROM ke mikrokontroler

Stop : Merupakan sinyal tepi naik SDA pada saat SCL tinggi

A0,A1 : Merupakan alamat dari IC SEEPROM

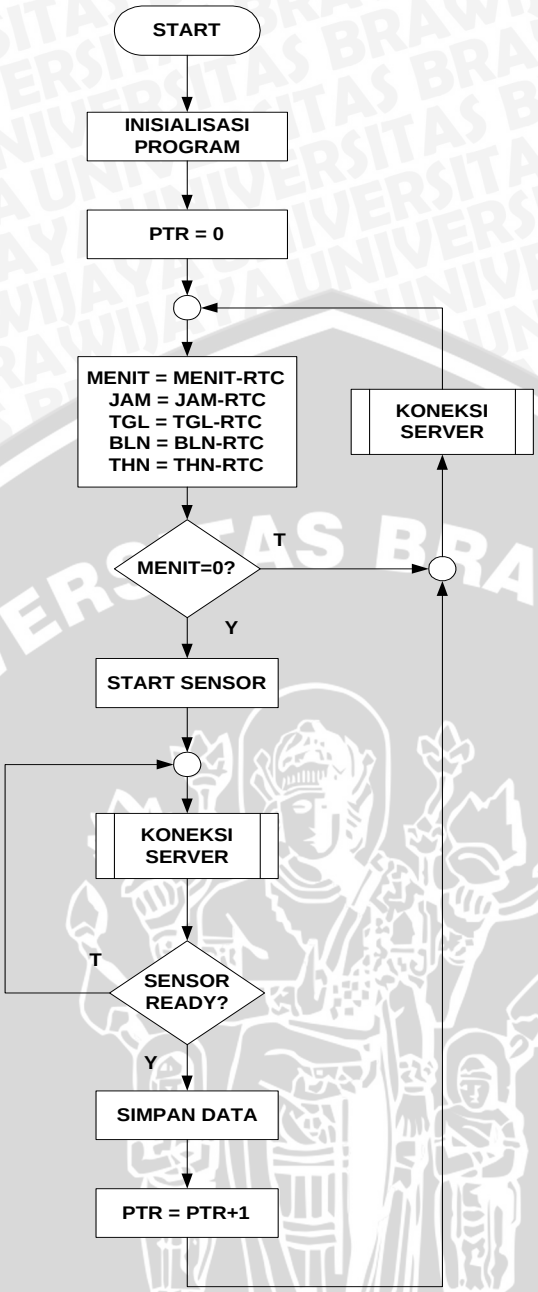
Pada proses baca data, pertama kita harus menuliskan alamat data yang akan kita baca pada memori seperti pada proses penulisan data diatas. Setelah Ack diterima mikrokontroler, sinyal start kita berikan lagi, diikuti dengan 8 bit pengalamatan *device* tapi untuk tanda baca/tulis kita beri logika 1 untuk pembacaan data. Setelah sinyal Ack diterima mikrokontroler, memori akan mengirimkan data yang dibaca.

4.4 PERANCANGAN PERANGKAT LUNAK

4.4.1 PERANGKAT LUNAK MIKROKONTROLER

Untuk pemakaian mikrokontroler di dalam suatu sistem perlu direncanakan perangkat lunak mikrokontroler yang dapat mengatur sistem tersebut. Perangkat lunak disini adalah susunan perintah-perintah (program) di dalam memori yang harus dilaksanakan mikrokontroler.

Program utama Penyimpan Data Pada Sistem Telemetri Stasiun Pengamat Cuaca Dilengkapi Dengan Sensor Kelembaban, Temperatur, Curah Hujan, Kecepatan dan Arah Angin dapat dilihat pada gambar 4.8. Program meng-inisialisasi RTC, Memori dan Serial port. Setelah itu, program akan mereset pointer menjadi 0. Mikrokontroler lalu mulai membaca menit, jam, tanggal, bulan dan tahun dari RTC. Mikrokontroler lalu mengecek apakah menit yang terbaca dari RTC menunjukkan nilai 0. Jika belum menunjukkan nilai 0, maka mikrokontroler akan menunggu sampai menit menunjukkan nilai 0 sambil melakukan subrutin koneksi Server Pusat Pengolah Data.

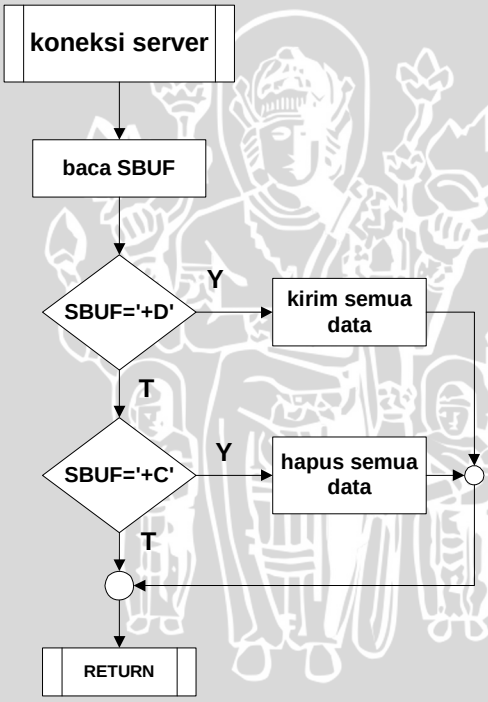


Gambar 4.8. Diagram alir program utama

Saat menit RTC terbaca 0, mikrokontroler lalu memberi perintah pada Pengontrol Sensor untuk memulai pengambilan data cuaca. Sambil menunggu Pengontrol Sensor mengambil data cuaca, mikrokontroler melakukan subrutin koneksi server pada Pusat Pengolah Data. Ketika Pengontrol Sensor sudah siap mengirimkan data pada mikrokontroler, maka Pengontrol Sensor mengirimkan pemberitahuan bahwa Pengontrol Sensor siap mengirimkan data (*Sensor ready*). Data cuaca kemudian disimpan di memori

SEEPROM AT24C512 oleh mikrokontroler, dan mikrokontroler menambah pointer satu. Mikrokontroler kemudian melakukan proses pengiriman data ke Pusat Pengolah Data dengan melakukan koneksi server. Setelah data berhasil dikirim ke Pusat Pengolah Data mikrokontroler kembali mengecek apakah menit RTC menunjukkan nilai 0 lagi.

Subrutin koneksi server adalah subrutin yang dikerjakan mikrokontroler untuk berkomunikasi dengan server pada base station. Pada subrutin koneksi server mikrokontroler akan membaca SBUF, apakah server mengirim perintah dengan kode '+D' atau '+C'. Jika perintah '+D' yang ada pada SBUF maka mikrokontroler akan mengirimkan semua data yang ada dalam memori. Apabila pada SBUF terbaca perintah '+C' maka mikrokontroler akan menghapus semua data yang ada pada memori.



Gambar 4.9. Diagram alir program subrutin koneksi server

BAB V
PENGUJIAN DAN ANALISA

Pengujian dilakukan untuk mengetahui tingkat keberhasilan perencanaan dan pembuatan alat. Bab ini akan dibahas pengujian alat Penyimpan Data Pada Sistem Telemetri Stasiun Pengamat Cuaca Dilengkapi Dengan Sensor Kelembaban, Temperatur, Curah Hujan, Kecepatan Angin dan Arah Angin Diakses Melalui Internet.

Dalam tahap pengujian metode yang digunakan adalah pengujian dilakukan setiap bagian sistem dan secara keseluruhan alat, kemudian dilakukan perbandingan antara hasil pengujian dengan teori.

5.1 Pengujian Rangkaian

5.1.1 Pengujian Komunikasi Serial

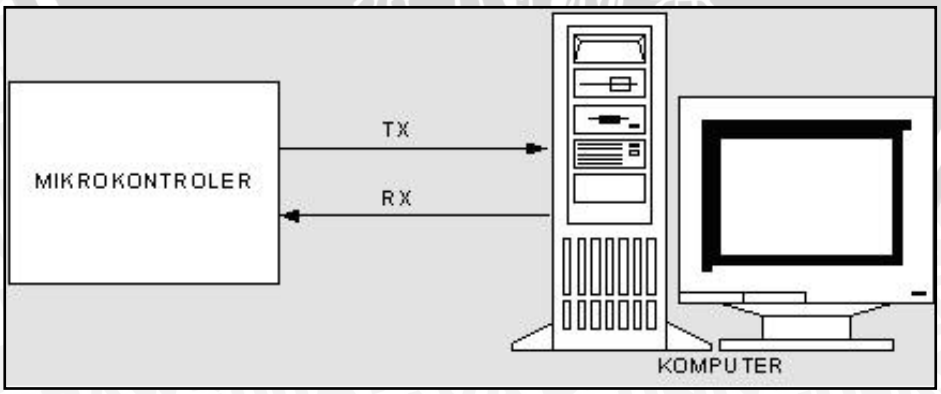
5.1.1.1 Tujuan

Tujuan pengujian rangkaian untuk sistem komunikasi serial agar dapat mengetahui bagaimana melakukan komunikasi secara serial menggunakan mikrokontroler dengan komputer.

5.1.1.2 Alat dan Bahan

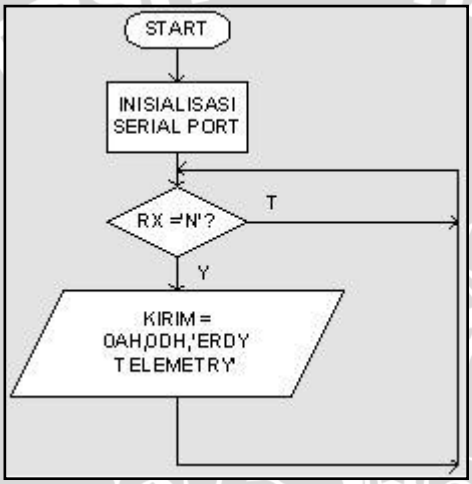
- Antar muka MAX 232
- Komputer
- Kabel Serial
- Catu daya
- Program Hyper Terminal

5.1.1.3 Prosedur Pengujian



Gambar 5.1. Diagram blok pengujian komunikasi serial antara mikrokontroler dengan komputer

Pengujian komunikasi serial antara mikrokontroler dan komputer ini juga bertujuan untuk mengetahui bagaimana cara mikrokontroler mengirimkan dan menerima data-data secara serial dengan komputer. Gambar 5.1. menunjukkan blok diagram pengujian sistem komunikasi serial tersebut. Mikrokontroler memerlukan perangkat lunak untuk dapat melaksanakan proses komunikasi data. Flowchart program mikrokontroler ditunjukkan dalam Gambar 5.2. Pertama kali mikrokontroler melaksanakan proses inisialisasi untuk port serial. Mikrokontroler menggunakan port 3.0 dan port 3.1. Inisialisasi dilaksanakan untuk mengatur kecepatan data yang digunakan dengan melakukan pemberian nilai pada register TMOD, SCON dan TIMER. Kecepatan data yang digunakan adalah 9600 bps.

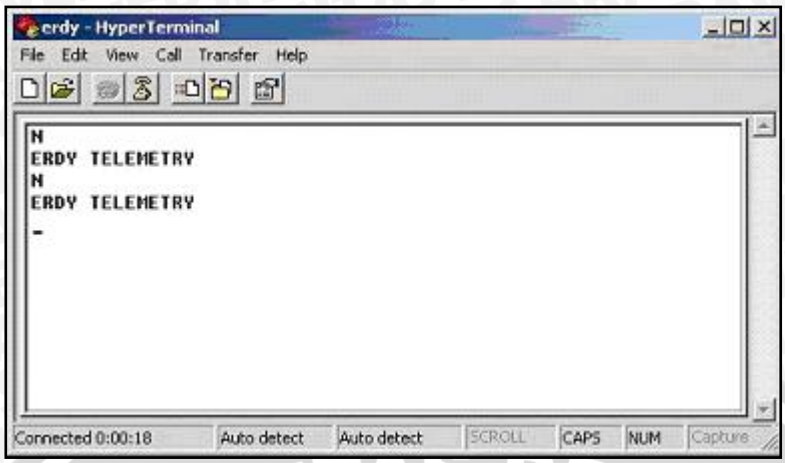


Gambar 5.2. Diagram flowchart program pengujian komunikasi serial pada mikrokontroler

Program kemudian melaksanakan proses menunggu adanya perintah dari komputer. Komputer memanfaatkan program hyperterminal untuk melaksanakan proses pengujian komunikasi serial dengan mikrokontroler. Ketik karakter “N” yang berarti komputer mengirimkan data pada mikrokontroler berupa data ASCII karakter “N”.

Mikrokontroler menerima data dari komputer dan melakukan pemeriksaan. Jika data yang diterima adalah 4EH yang berarti karakter “N”, maka mikrokontroler akan mengrimkan data-data 0AH,0DH, dan karakter “ERDY TELEMETRY”. Data-data yang dikirimkan oleh mikrokontroler tersebut diterima dan ditampilkan oleh komputer. Jika data yang diterima dari komputer bukan karakter “N”, maka mikrokotoler akan mengabaikan dan kembali melaksanakan proses penerimaan data selanjutnya dari komputer. Gambar 5.3. menunjukkan hasil pengujian sistem komunikasi serial antara mikrokontroler dengan komputer yang ditampilkan dengan hyperterminal.

5.1.1.4 Data Hasil Pengujian



Gambar 5.3. Hasil pengujian komunikasi serial menggunakan hyperterminal

5.1.1.5 Analisa Hasil Pengujian

Pengujian komunikasi serial dengan menggunakan fasilitas hyperterminal berjalan dengan baik. Saat karakter "N" diberikan oleh komputer, mikrokontroller kemudian mengirimkan mengirim balik pada komputer karakter "ERDY TELEMETRY" yang dapat dilihat pada tampilan hyperterminal diatas. Pada pengujian kedua dengan perlakuan yang sama, mikrokontroller memberikan karakter yang sama pada percobaan sebelumnya, sehingga dapat disimpulkan bahwa pengujian komunikasi antara komputer dengan mikrokontroller berrjalan dengan baik.

5.1.2 Pengujian Rangkaian RTC

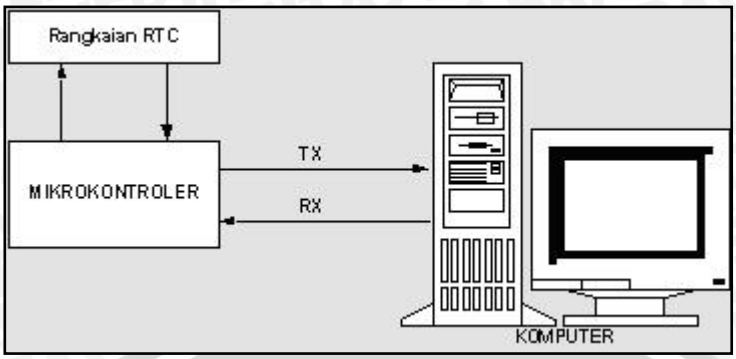
5.1.2.1 Tujuan

Tujuan pengujian rangkaian untuk sistem rangkaian RTC agar dapat mengetahui bagaimana membaca fungsi pewaktu pada IC RTC DS1287A menggunakan mikrokontroler.

5.1.2.2 Alat dan Bahan

- Antar muka MAX 232
- Komputer
- Kabel Serial
- Catu daya
- Program Hyper Terminal

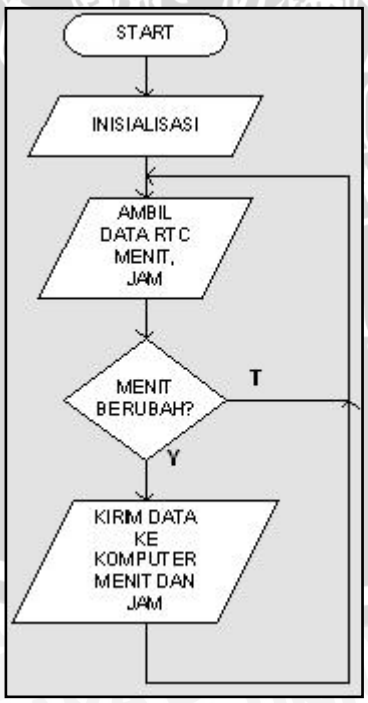
5.1.2.3 Prosedur Pengujian



Gambar 5.4. Diagram blok pengujian rangkaian RTC

Pengujian rangkaian RTC ditunjukkan dalam Gambar 5.4. RTC DS 12887 dihubungkan dengan mikrokontroler. Mikrokontroler dapat mengambil data-data yang ada pada memori RTC berupa detik, menit, jam, tanggal, bulan, tahun dan abad. Dalam pengujian RTC ini, hanya diambil data menit dan jam yang kemudian data tersebut dikirimkan oleh mikrokontroler pada komputer. Diagram flowchart ditunjukkan dalam Gambar 5.5.

Komputer hanya sebagai penerima data dan menampilkan data-data tersebut dengan program hyperterminal.

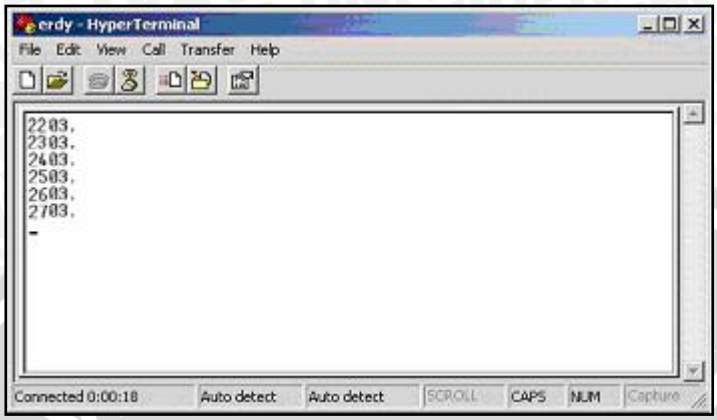


Gambar 5.5. Diagram Flowchart pengujian rangkaian RTC

Mikrokontroler melaksanakan inisialisasi pada port serial dan untuk RTC. Inisialisasi pada RTC dilaksanakan untuk menjalankan fungsi osilator dengan memberikan

nilai 20H pada register A RTC. Secara berkelanjutan mikrokontroler melakukan pemeriksaan data menit. Jika menit berubah, maka mikrokontroler mengirimkan data menit dan jam pada komputer. Hasil pengujian rangkaian RTC ditunjukkan dalam Gambar 5.6.

5.1.2.4 Hasil Pengujian Rangkaian RTC



Gambar 5.6. Hasil pengujian RTC menggunakan hyperterminal

5.1.2.5 Analisa Hasil Pengujian

Pada pengujian RTC, komputer hanya sebagai penerima data dari mikrokontroller. Saat mikrokontroller dihubungkan dengan komputer, pada Gambar 5.6 menunjukkan komputer menerima data pertama yaitu jam 03 menit 22 dan data berikutnya muncul tiap pergantian menit menjadi jam 03 menit 23. Pengujian dilaksanakan dalam tempo 6 menit dimana RTC memberikan 6 data menit dan jam. Dari tampilan hyperterminal diatas dapat disimpulkan bahwa RTC bekerja dengan baik.

5.2 Pengujian Rangkaian Keseluruhan

5.2.1 Tujuan

Tujuan pengujian rangkaian secara keseluruhan adalah untuk mengetahui tingkat keberhasilan alat yang direncanakan serta untuk mendapatkan kesimpulan serta hasil analisa dari data yang diperoleh.

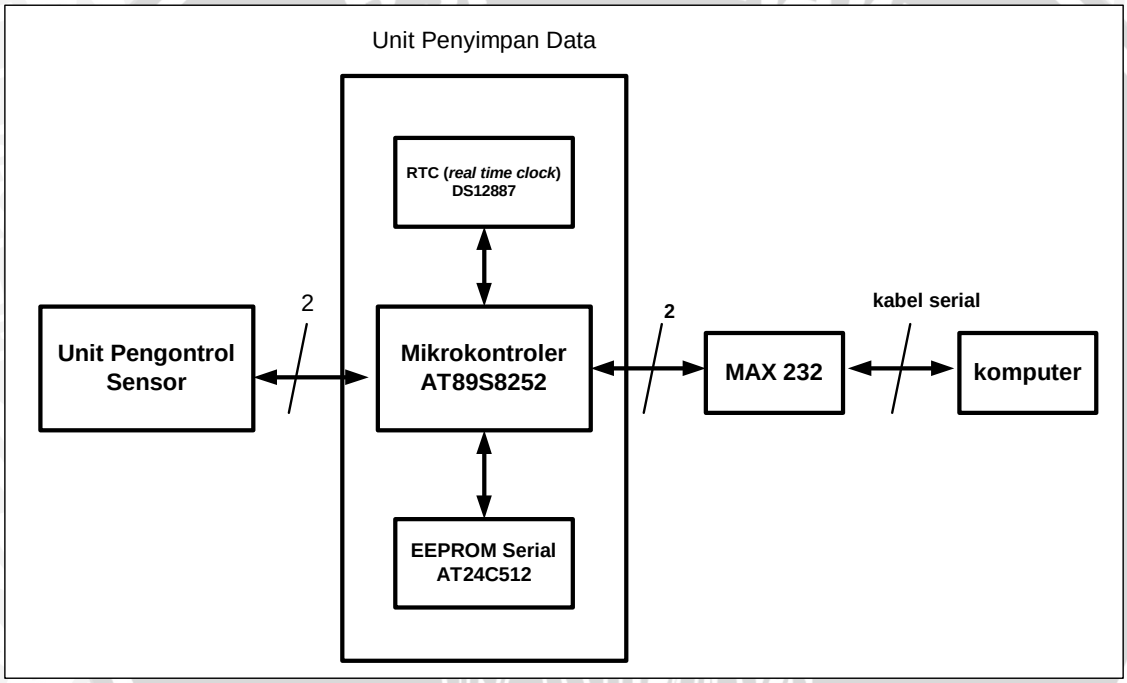
5.2.2 Alat dan Bahan

- Alat Penyimpan Data Pada Sistem Telemetri Stasiun Pengamat Cuaca
- Alat Pengontrol Sensor Pada Sistem Telemetri Stasiun Pengamat Cuaca
- Antar muka MAX 232
- Komputer
- Kabel Serial
- Catu daya

- Program Hyper Terminal

5.2.3 Prosedur Pengujian

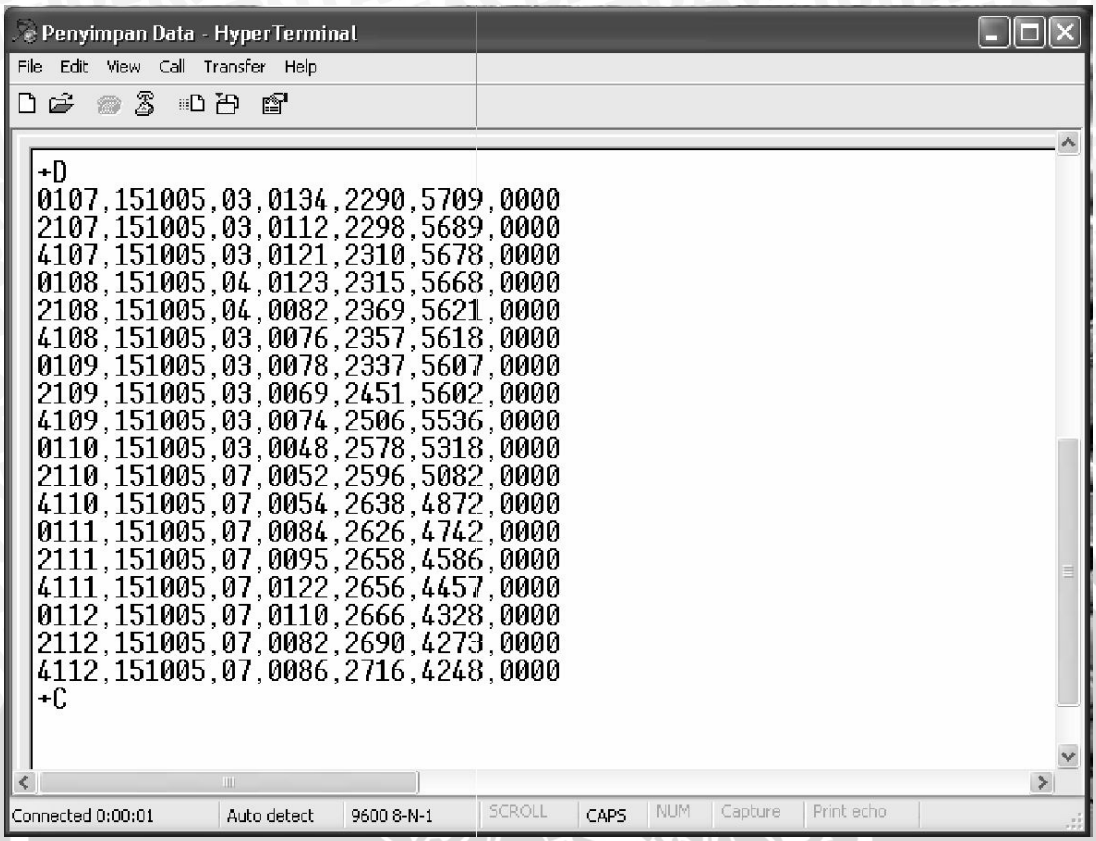
Dalam pengujian alat ini, yang dilakukan adalah membuat program pengujian alat secara keseluruhan, menghubungkan alat penyimpan data dengan alat pengontrol sensor dan komputer melalui antarmuka mikrokontroler dan komputer, menghidupkan catu daya. Program pengujian merupakan simulasi yang menyerupai kerja dari sistem secara keseluruhan. Saat kita ketik perintah '+D' maka mikrokontroler akan mengirimkan semua data yang ada dalam memori. Untuk menghapus isi dari memori kita mengetik perintah'+C' maka semua isi memori akan dihapus.



Gambar 5.7. Blok pengujian alat penyimpan data pada sistem telemetri stasiun pengamat cuaca

5.2.4 Data Hasil Pengujian

Pada saat kode mulai (+D) maka mikrokontroler akan memberi data yang akan disimpan pada memori. Setelah semua data dalam memori ditampilkan, kode untuk menghapus isi memori kita berikan (+C). Hasil pengujian ditunjukkan pada Gambar 5.8.



Gambar 5.8. Tampilan program hyper terminal pada saat pengujian alat keseluruhan

5.2.5 Analisa Hasil Pengujian

Pada pengujian alat keseluruhan yang sekaligus pengujian memori dapat diketahui apakah memori bekerja dengan baik saat penyimpanan maupun saat pembacaan. Pada pangujian alat seperti ditunjukkan pada Gambar 5.8, komputer memberikan perintah pada mikrokontroller untuk mengirimkan data yang terdapat pada memori dengan memberikan karakter “+D”. Mikrokontroller akan bekerja untuk mendapatkan data dari sensor-sensor yang terdapat pada Blok Pengontrol Sensor. Setelah mikrokontroller dihubungkan dengan komputer dan karakter “+D” diberikan pada mikrokontroller dengan cara mengetik “+D” pada keyboard seperti pada tampilan hyperterminal Gambar 5.8, maka mikrokontroller akan memberikan data yang ada pada memori yang dapat dilihat pada Gambar 5.8. Data pengujian dari blok pengontrol sensor diambil mulai jam 07.00 sampai 12.41 saat mikrokontroller menerima perintah “+D”. Data yang diambil dari blok Pengontrol Sensor terdiri dari paket-paket data. 1 paket data pada tampilan hyperterminal diatas sebagai contoh adalah “0107,151005,03,0134,2290,5709,0000” dapat dijabarkan sebagai berikut: 4 karakter pertama merupakan data menit 01 dan jam 07 (07.00 WIB). 6 karakter berikutnya adalah data tanggal 15, bulan 10 dan tahun 05 (2005). 2 karakter berikutnya adalah data

arah angin 03 yang menunjukkan arah timur, dimana utara ditunjukkan oleh karakter 01, timur laut 02, timur 03 sampai barat laut pada karakter 08. Data 0134 pada karakter berikutnya menunjukkan data kecepatan arah angin dengan nilai 01,34 dalam satuan m/menit. Karakter 2290 menunjukkan data temperatur dengan besar nilai 22,90° dalam satuan Celcius. Karakter 5709 menunjukkan data kelembaban udara dengan nilai 55,09 dalam satuan % (persen). Karakter 0000 menunjukkan data curah hujan dengan satuan mm. Pada pengujian diatas mikrokontroller mengirimkan 18 paket data yang terdapat dalam isi memori pada pengujian dengan rentang waktu antara pukul 07.00 sampai dengan 12.41. Dapat kita ambil kesimpulan bahwa memori dapat menyimpan data yang berasal dari Blok Pengontrol Sensor dan dapat dibaca saat ada perintah untuk mengirimkan data pada komputer. Secara keseluruhan alat bisa bekerja sesuai dengan yang diharapkan.

BAB VI
KESIMPULAN DAN SARAN

6.1 Kesimpulan

Dari hasil pengujian tiap blok rangkaian dan pengujian sistem keseluruhan yang telah dilakukan pada bab V, dapat ditarik kesimpulan sebagai berikut :

1. Jenis memori yang cocok untuk aplikasi penyimpanan data pada sistem telemetri stasiun pengamat cuaca adalah EEPROM serial yang memakai protokol I2C. Jenis memori ini Cuma membutuhkan empat buah pin sehingga memiliki dimensi yang kecil. Kapasitas memori SEEPROM akan terus berkembang karena alamat memori diberikan secara serial.
2. Unit Pengontrol Sensor memberikan data sebanyak 9 byte yang terdiri dari arah angin 1 byte, kecepatan angin 2 byte, suhu 2 byte, kelembaban 2 byte, dan curah hujan 2 byte dan dari RTC sebanyak 5 byte yang terdiri atas menit, jam, tanggal, bulan, dan tahun masing-masing 1 byte yang disimpan dalam EEPROM serial AT24C512 dalam sekali penerimaan data.
3. Data yang diterima dari unit pengontrol sensor dan dari RTC akan digabung menjadi Satu paket data sebanyak 14 byte. Semua data yang ada dalam memori akan dikirim ke stasiun pusat (*base station*) menunggu perintah dari stasiun pusat (*base station*).
4. Rangkaian bisa bekerja dengan baik, RTC bisa memberikan data waktu, Memori SEEPROM bisa menyimpan data dan membaca data yang telah tersimpan, sistem bisa berkomunikasi dengan blok pengontrol sensor dan komputer secara serial.

6.2 Saran

Saran yang dapat diberikan dalam perancangan serta pembuatan penyimpan data pada sistem telemetri stasiun pengamat cuaca dilengkapi dengan sensor kelembaban, temperatur, curah hujan, kecepatan angin dan arah angin diakses melalui internet adalah sebagai berikut :

1. Perlu dilakukan penelitian serta pengembangan lebih lanjut agar kapasitas memori yang dipakai lebih besar sehingga jangka waktu sampai memori penuh terisi yang lebih lama.
2. Perlu dilakukan penelitian dan pemilihan komponen yang memiliki konsumsi daya rendah sehingga pemakaian daya semakin efisien.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

3. Alat yang dibuat masih memiliki dimensi ukuran yang relatif besar, untuk itu diperlukan pengembangan agar alat yang dihasilkan mempunyai dimensi yang cukup kecil.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR PUSTAKA

Annonymous. 1998. *Interfacing AT24CXX Serial EEPROM's With AT89CXX Microcontollers*. Atmel Inc; http://www.atmel.com.

Annonymous. 2002. *8 Bit Microcontroller with 4K Bytes Flash - AT89C51*. USA: Atmel Inc.; http://www.atmel.com.

Annonymous. 2004. *2-Wire Serial EEPROM 512K AT24C512*. Atmel Inc; http://www.atmel.com.

Annonymous. *DS 12887 Real-Time Clock*. Dallas Semiconductor; http://www.maxim-ic.com.

Malik, Moh. Ibnu. 2003. Belajar Mikrokontroler ATMEL AT89S8252. Jakarta : Penerbit Gava Media.

Mismail, Budiono. 1992. *Rekayasa Data Logger Sistem Tenaga Listrik Mikrohidro Dengan Sistem Mikroprosesor*. Universitas Brawijaya. Malang.

Nalwan, Paulus Andi. 2003. *Teknik antar muka dan pemrograman mikrokontroller AT89C51*. Jakarta : Elexkomputindo.

Setiawan, Arief Candra. 2003. *Otomatisasi Stasiun Cuaca Untuk Menunjang Kegiatan Pertanian*. Program Pasca Sarjana / S3. Institut Pertanian Bogor.

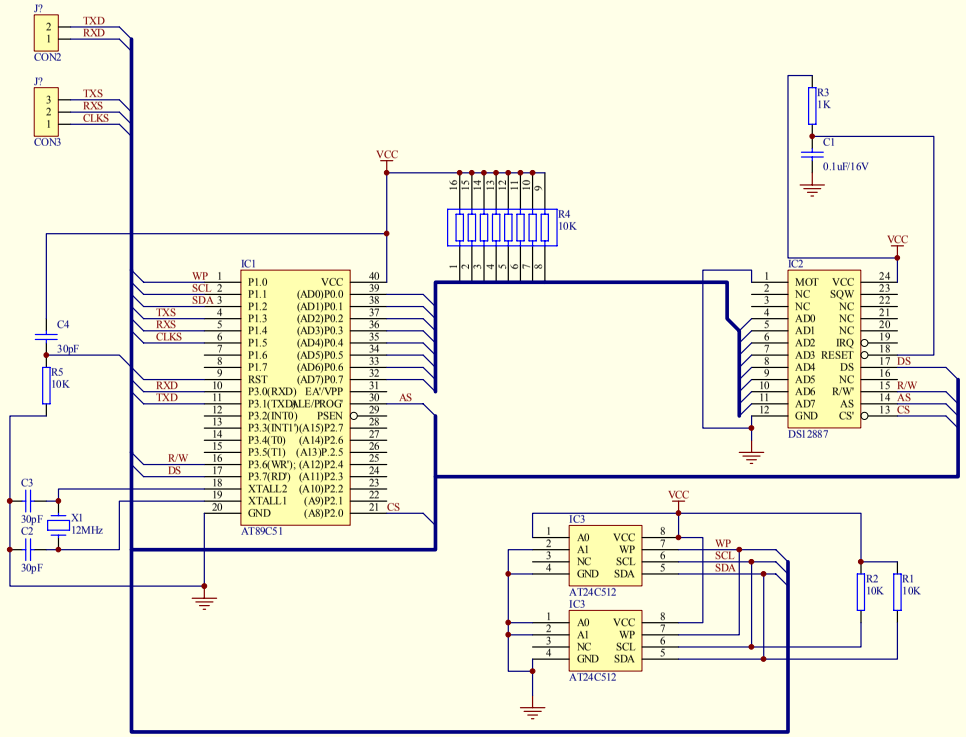
Tokheim, Roger L. 1995. *Elektronika Digital*. Jakarta : Erlangga.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

LAMPIRAN

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

UNIVERSITAS BRAWIJAYA
repository.ub.ac.id



Title		
Size B	Number	Revision
Date: 18-May-2007	Sheet of	
File: G:\YANRI\ASMDAT~1\ERDY1.SCH	Drawn By:	

```
;***************************
; PROGRAM MIKROKONTROLER
; BAGIAN PERANGKAT
; PENYIMPAN DATA PADA SISTEM TELEMETRI STASIUN
; PENGAMAT CUACA DILENGKAPAI DENGAN SENSOR
; KELEMBABAN, TEMPERATUR, CURAH HUJAN, KECEPATAN ANGIN
; DAN ARAH MATA ANGIN DIAKSES MELALUI INTERNET
; OLEH : ERDY PRASETYA KUSUMA
;*******************************************************
;
SDA             EQU     P1.2
SCL             EQU     P1.1
TXSENSOR        EQU     P1.3
RXSENSOR        EQU     P1.4
CLKSENSOR       EQU     P1.5
;
AD_RTC          EQU     P0              ;BUS DATA RTC
CS_RTC          EQU     P2.0
;
DELAY1          EQU     09H
DELAY2          EQU     0AH
DELAY3          EQU     0BH
DATASER         EQU     0CH
PTRMEN          EQU     0DH             ;MEMORY MENIT
PTRJAM          EQU     10H             ;MEMORY JAM
PTRM1           EQU     11H             ;POINTER ALAMAT SEEPROM LOW
PTRM2           EQU     12H             ;POINTER ALAMAT SEEPROM HIGH
PTRM3           EQU     13H             ;POINTER ALAMAT SELECT SEEPROM
DATAM           EQU     14H
DATAS           EQU     20H             ;MEMORY SEMENTARA
;
                ORG     00H
                LJMP    INISIALISASI
;
                ORG     23H
                LJMP    SERIALINT
;
                ORG     30H
INISIALISASI:
                MOV     SP,#44H         ;stack pointer mulai alamat 44h
                LCALL   INITSERIAL      ;panggil subprogram inisialisasi serial
                MOV     PTRJAM,#00H
                MOV     PTRMEN,#00H
                MOV     PTRM1,#00H
                MOV     PTRM2,#00H
                MOV     PTRM3,#01H
                CLR     SCL
                CLR     SDA
                SETB    TXSENSOR        ;memberikan nilai awal pada port serial untuk komuniaksi sensor
                SETB    RXSENSOR
                SETB    CLKSENSOR
                MOV     R4,#03H
                MOV     PTRM3,#00H
                MOV     R0,#PTRM1
                MOV     DPH,#00
                MOV     DPL,#00
INBAL:
                LCALL   MEMREAD                 ;panggil subporgram bca data di memori serial
                MOV     A,DATAM
                MOV     @R0,A
                INC     R0
                DJNZ    R4,INBAL
;
;*****************************************
; PROGRAM UTAMA
;*****************************************
;
UTAMA:
                MOV     A,PTRJAM
```

```
                CJNE    A,#05H,UTLAN1
                LCALL   CEKSERVER
                CJNE    A,#05H,UTLAN2
                LCALL   KIRIMDATASENSOR ;proses pengirimn data-data sensor
UTLAN2:
                MOV     PTRJAM,#00H
                LCALL   CEKPTRADR
                LJMP    UTAMA
UTLAN1:
                LCALL   TAKERTC         ;ambil data di rtc
                LCALL   TAKESENSOR1     ;ambil data sensor1
                LCALL   TAKESENSOR2     ;ambil data sensor2
                LCALL   TAKESENSOR3     ;ambil data sensor3
                LCALL   TAKESENSOR4     ;ambil data sensor4
                LCALL   TAKESENSOR5     ;ambil data sensor5
                LCALL   SIMPANSEPROM    ;SIMPAN DATA SI MEMORI
                INC     PTRJAM
                LJMP    UTAMA
;
;*************************************
;* INISIALISASI SERIAL PORT
;*************************************
;
INITSERIAL:
                MOV     TMOD,#20H
                MOV     TH1,#0FDH
                MOV     SCON,#50H
                SETB    TR1
                SETB    ES              ;enable serial intrrupt
                SETB    EA              ;enable all interrupt
                RET
;
;*************************************
; PERIKSA SERVER
;*************************************
;
CEKSERVER:
                MOV     A,#05H
                LCALL   KIRIMSERIAL
                CJNE    A,#01H,CSOFF
                LCALL   PERIKSAJAWABAN
                CJNE    A,#01H,CSOFF
                MOV     A,#05H
;               LJMP    CSBAR
CSOFF:

                RET
;
;*************************************
; PROGRAM KIRIM SERIAL
;*************************************
;
KIRIMSERIAL:
                MOV     SBUF,A
                SETB  TI
CEKTI:
                SETB    TI              ;PROSES KIRIM DATA SERIAL DIMULAI
                JB      TI,CEKTI  ;JIKA TI=0, DATA TELAH SELESAI DIKIRIM
KSBAR:
                RET
;
;******************************
; PERIKSA JAWABAN CEK SERVER,A
;******************************
;
PERIKSAJAWABAN:
                MOV     DATASER,#00H
                MOV     DELAY1,#0FFH
                MOV     DELAY2,#0FFH
                MOV     DELAY3,#0FFH
PJBAL1:
```

```
                MOV     A,DATASER
                CJNE    A,#05H,PJDEL
                MOV     A,#01H
                LJMP    PJBAR
PJDEL:
                DJNZ    DELAY1,PJBAL1
                DJNZ    DELAY2,PJBAL1
                DJNZ    DELAY3,PJBAL1
                MOV     A,#00H
PJBAR:
                RET
;
;***********************************
; PROGRAM KIRIM DATA SENSOR
; TIAP 1 JAM, A,B,R7,DPTR
;***********************************
;
KIRIMDATASENSOR:
                MOV     A,#36
                MOV     B,#06
                MUL     AB
                MOV     R7,A
                MOV     DPH,PTRM2
                MOV     DPL,PTRM1
KDSUL1:
                DEC     DPL
                JC      KDKUR1
                LJMP    KDLAN
KDKUR1:
                DEC     DPH
KDLAN:
                DJNZ    R7,KDSUL1
KDSBAL:
                LCALL   BACASEPROM
                LCALL   KIRIMSERIAL
                INC     DPTR
                MOV     A,DPL
                DEC     A
                CJNE    A,PTRM1,KDSBAL
                MOV     A,DPH
                DEC     A
                CJNE    A,PTRM1,KDSBAL

                RET
;
;***********************************
; SUBPROGRAM BACA DATA PADA SERIAL
; EEPROM PADA ALAMAT TERTENTU
;***********************************
;
BACASEPROM:
                LCALL   STARTWR                 ;PANGGIL STARTWR
                LCALL   CEKACK                  ;FUNGSI PERIKA SINYAL ACKNOWLEDGE
                CJNE    A,#01H,BACASEPROM
BSEBAL:
                LCALL   SETADR                  ;ATUR ALAMAT
                LCALL   CEKACK
                LCALL   STARTRD                 ;ATUR SINYAL MULAI PEMBACAAN
                LCALL   CEKACK
                CJNE    A,#01,BSEBAL
                LCALL   RDBIT                   ;PEMBACAAN DATA 8 BIT
                LCALL   STOPSE                  ;MENGHENTIKA SERIAL EEPROM
                RET
;
;**************************
; SUBPROGRAM START TO WRITE
;**************************
;
STARTWR:
                CLR     SDA
                NOP
```

```
                SETB    SCL
                SETB    SDA     ;START KODE
                CLR     SCL
                SETB    SDA     ;1
                LCALL   CLOCKON
                CLR     SDA     ;0
                LCALL   CLOCKON
                SETB    SDA     ;1
                LCALL   CLOCKON
                CLR     SDA     ;0
                LCALL   CLOCKON
                MOV     A,PTRM3
                CJNE    A,#01H,SW0
                CLR     SDA
                LCALL   CLOCKON
                SETB    SDA
                LCALL   CLOCKON
                LJMP    SWBWR
SW0:
                SETB    SDA
                LCALL   CLOCKON
                CLR     SDA
                LCALL   CLOCKON
SWBWR:
                CLR     SDA
                LCALL   CLOCKON
                RET
;
;**********************************
; CLOCK SER
;**********************************
;
CLOCKON:
                SETB    SCL     ;mengeluarkan sinyal clock
                CLR     SCL
                RET
;
;*********************
; CEK SINYAL ACKNOWLEDGE
;************************
;
CEKACK:
                LCALL   CLOCKON
                JB      SDA,CA1
                MOV     A,#01H
                LJMP    CABAR
CA1:
                MOV     A,#00H
CABAR:
                RET
;
;*******************************
; SUBPROGRAM ATUR ALAMAT
;*******************************
;
SETADR:
                MOV     R6,#08
                MOV     A,DPH
SA1BAL:
                CLR     C
                RLC     A
                JNC     SAC0
                SETB    SDA
                LJMP    SANEX
SAC0:
                CLR     SDA
SANEX:
                LCALL   CLOCKON
                DJNZ    R6,SA1BAL
;
                MOV     R6,#08
```

```
                MOV     A,DPL
SA2BAL:
                CLR     C
                RLC     A
                JNC     SAC1
                SETB    SDA
                LJMP    SANEX1
SAC1:
                CLR     SDA
SANEX1:
                LCALL   CLOCKON
                DJNZ    R6,SA2BAL

                RET
;
;**********************************
; SUBPRGRAM MULAI PEMBACAAN KODE
;**********************************
;
STARTRD:
                CLR     SDA
                NOP
                SETB    SCL
                SETB    SDA     ;START KODE
                CLR     SCL
                SETB    SDA     ;1
                LCALL   CLOCKON
                CLR     SDA     ;0
                LCALL   CLOCKON
                SETB    SDA     ;1
                LCALL   CLOCKON
                CLR     SDA     ;0
                LCALL   CLOCKON
                MOV     A,PTRM3
                CJNE    A,#01H,SW01
                CLR     SDA
                LCALL   CLOCKON
                CLR     SDA
                LCALL   CLOCKON
                SETB    SDA
                LCALL   CLOCKON
                LJMP    SWBWR1
SW01:
                CLR     SDA
                LCALL   CLOCKON
                SETB    SDA
                LCALL   CLOCKON
                CLR     SDA
                LCALL   CLOCKON
SWBWR1:
                SETB    SDA     ;SET READ SINYAL
                LCALL   CLOCKON
                RET
;
;**********************************
; BACA BIT DATA
;**********************************
;
RDBIT:
                MOV     A,#08H
                MOV     R6,A            ;SET PERULANGAN 8 BIT
                MOV     A,#00H
RBBAL:
                LCALL   CLOCKON
                JNB     SDA,RBSDA0      ;DATA 1/0
                SETB    C
                LJMP    RBNEX
RBSDA0:
                CLR     C
RBNEX:
                RLC     A
```

```
                DJNZ    R6,RBBAL
                RET
;
;**********************************
; SINYAL STOP SERIAL EEPROM
;**********************************
;
STOPSE:
                LCALL   CLOCKON
                SETB    SDA
                SETB    SCL
                CLR     SDA
                CLR     SCL
                RET
;
;**********************************
; SUBPROGRAM MEMERIKSA ALAMAT MEMORY
; UNTUK PEMBAGIAN PENEMPATAN DATA
; PADA MEMORY
;**********************************
;
CEKPTRADR:
                MOV     A,PTRM3
                CJNE    A,#01H,CPAKE2
                MOV     A,PTRM1
                CJNE    A,#0FH,CPABAR
                MOV     A,PTRM2
                CJNE    A,#0F9H,CPABAR
                LJMP    CPASETK2
CPAKE2:
                MOV     A,PTRM1
                CJNE    A,#0F0H,CPABAR
                MOV     A,PTRM2
                CJNE    A,#0FFH,CPABAR
                MOV     PTRM1,#03H
                MOV     PTRM2,#00H
                MOV     PTRM3,#00H
                LJMP    CPABAR
CPASETK2:
                MOV     PTRM1,#00H
                MOV     PTRM2,#00H
                MOV     PTRM3,#00H
CPABAR:
                MOV     R7,#03H
                MOV     R0,#PTRM3
                MOV     DPTR,#0000H
CPABAR1:
                LCALL   STARTWR
                LCALL   CEKACK
                LCALL   SETADR
                LCALL   CEKACK
                MOV     A,@R0
                LCALL   WRBIT
                LCALL   CEKACK
                LCALL   STOPSE
                INC     DPTR
                DEC     R0
                DJNZ    R7,CPABAR1
                RET
;
;*********************************
; SUBPROGRAM PENULISAN DATA
; PADA SEEPROM
;*********************************
;
WRBIT:
                MOV     R6,#08
WRBBAL:
                CLR     C
                RLC     A
                JNC     WBSET0
```

```
                SETB    SDA
                LJMP    WBLAN1
WBSET0:
                CLR     SDA
WBLAN1:
                LCALL   CLOCKON
                DJNZ    R6,WRBBAL
                RET
;
;*********************************
; SUBPROGRAM AMBIL DATA RTC
;*********************************
;
TAKERTC:
                PUSH    DPL
                PUSH    DPH
                MOV     DPH,#00H
                MOV     A,DATAS
                CJNE    A,#00H,TRLAN1
                LCALL   AMBILWAKTU
                MOV     PTRMEN,DATAS
                LJMP    TRBAR
TRLAN1:
                LCALL   AMBILWAKTU
                MOV     A,DATAS
                CLR     C
                SUBB    A,PTRMEN
                JC      LBKECIL
                LJMP    TRBDG1
LBKECIL:
                MOV     A,DATAS
                ADD     A,#60
                CLR     C
                SUBB    A,PTRMEN
TRBDG1:
                CJNE    A,#10,TRLAN1
                LCALL   AMBILWAKTU

TRBAR:
                POP     DPH
                POP     DPL
                RET
;
;*********************************
; SYBPROGRAM AMBIL DATA WAKTU
; MENIT,JAM,TANGGAL
;*********************************
;
AMBILWAKTU:
                MOV     DPL,#02H
                MOV     R0,#DATAS
                MOVX    A,@DPTR ;AMBILDATA MENIT
;               PUSH    ACC
                MOV     B,#10
                DIV     AB
                ORL     A,#30H
                MOV     @R0,A
                INC     R0
;               POP     ACC
                MOV     A,B
                ORL     A,#30H
                MOV     @R0,A
                INC     R0
                ORL     B,#30H
                MOV     @R0,B
                MOV     DPL,#04
                INC     R0
                MOVX    A,@DPTR         ;AMBIL DATA JAM
                MOV     B,#10
                DIV     AB
                ORL     A,#30H
```

```
                MOV     @R0,A
                INC     R0
                MOV     A,B
                ORL     A,#30H
                MOV     @R0,A
                INC     R0
                ORL     B,#30H
                MOV     @R0,B
                INC     R0
                MOV     A,#','
                MOV     @R0,A
                MOV     DPL,#07H
                INC     R0
                MOVX    A,@DPTR         ;AMBIL DATA TANGGAL
                MOV     B,#10
                DIV     AB
                ORL     A,#30H
                MOV     @R0,A
                INC     R0
                ORL     B,#30H
                MOV     @R0,B
                INC     DPTR
                INC     R0
                MOVX    A,@DPTR         ;AMBILDATA BULAN
                MOV     B,#10
                DIV     AB
                ORL     A,#30H
                MOV     @R0,A
                INC     R0
                ORL     B,#30H
                MOV     @R0,B
                INC     DPTR
                INC     R0
                MOVX    A,@DPTR         ;AMBILDATA TAHUN
                MOV     B,#10
                DIV     AB
                ORL     A,#30H
                MOV     @R0,A
                INC     R0
                ORL     B,#30H
                MOV     @R0,B
                INC     R0
                MOV     A,#','
                MOV     @R0,A
                RET
;
;*******************************
; SUBPROGRAM AMBILDATA SENSOR 1
; SENSOR ARAH ANGIN
;*******************************
;
TAKESENSOR1:
                MOV     R0,#2BH
                MOV     A,#01H
                LCALL   OUTSENSOR
                MOV     R7,#02H
TS1BAL:
                LCALL   INSENSOR
                MOV     @R0,A
                INC     R0
                DJNZ    R7,TS1BAL
                MOV     @R0,#','
                RET
;
;*******************************
; SUBPROGRAM INSTRUKSI TX SENSOR
;*******************************
;
OUTSENSOR:
                MOV     R6,#08H
OSBAL1:
```

```
                CLR     C
                RLC     A
                JC      OSSET1
                CLR     TXSENSOR
                LJMP    PROSCLK
OSSET1:
                SETB    TXSENSOR
PROSCLK:
                SETB    CLKSENSOR
                LCALL   DELAYSENSOR
                CLR     CLKSENSOR
                DJNZ    R6,OSBAL1
                RET
;
;******************************
; SUBPROGRAM TERIMA DATA SENSOR
;******************************
;
INSENSOR:
                JB      RXSENSOR,ISLAN
                MOV     A,#30H
                LJMP    ISBAR
ISLAN:
                MOV     R6,#08H
ISLAN1:
                MOV     DELAY1,#0FFH
                MOV     DELAY2,#0FFH
                MOV     DELAY3,#0FFH

ISCLKC:
                JB      CLKSENSOR,ISDEL1
                DJNZ    DELAY1,ISCLKC
                DJNZ    DELAY2,ISCLKC
                DJNZ    DELAY3,ISCLKC
                MOV     A,#30H
                LJMP    ISBAR
ISDEL1:
                JNB     RXSENSOR,ISINSET0
                SETB    C
                LJMP    ISSIMA
ISINSET0:
                CLR     C
ISSIMA:
                RLC     A
                JB      CLKSENSOR,$
                DJNZ    R6,ISLAN1
ISBAR:
                RET
;
;*********************************
; AMBIL DATA SENSOR 2
; KECEPATAN ANGIN
;*********************************
;
TAKESENSOR2:
                MOV     A,#02H
                LCALL   OUTSENSOR
                MOV     R7,#04H
                MOV     R0,#2EH
TS2BAL:
                LCALL   INSENSOR
                MOV     @R0,A
                INC     R0
                DJNZ    R7,TS2BAL
                MOV     A,#','
                MOV     @R0,A
                RET
;
;*******************************
; SUBPROGRAM AMBIL DATA SENSOR 3
; SUHU
```

```
;*******************************
;
TAKESENSOR3:
                MOV     A,#03
                LCALL   OUTSENSOR
                MOV     R7,#04H
                MOV     R0,#33H
TS3BAL:
                LCALL   INSENSOR
                MOV     @R0,A
                INC     R0
                DJNZ    R7,TS3BAL
                MOV     @R0,#','
                RET
;
;*******************************
; SUBPROGRAM AMBIL DATA SENSOR 4
; KELEMBABAN
;*******************************
;
TAKESENSOR4:
                MOV     A,#04
                LCALL   OUTSENSOR
                MOV     R7,#04H
                MOV     R0,#38H
TS4BAL:
                LCALL   INSENSOR
                MOV     @R0,A
                INC     R0
                DJNZ    R7,TS4BAL
                MOV     @R0,#','
                RET
;
;*******************************
; SUBPROGRAM AMBIL DATA SENSOR 5
; CURAH HUJAN
;*******************************
;
TAKESENSOR5:
                MOV     A,#05
                LCALL   OUTSENSOR
                MOV     R7,#04H
                MOV     R0,#33H
TS5BAL:
                LCALL   INSENSOR
                MOV     @R0,A
                INC     R0
                DJNZ    R7,TS5BAL
                MOV     @R0,#13
                INC     R0
                MOV     @R0,#10
                RET
;
;******************************
;PROGRAM SIMPAN DATA KE EEPROM
;******************************
;
SIMPANSEPROM:
                MOV     R0,#DATAS
                MOV     DPH,PTRM2
                MOV     DPL,PTRM1
SIMSBAL:
                LCALL   STARTWR
                LCALL   CEKACK
                LCALL   SETADR
                LCALL   CEKACK
                MOV     A,@R0
                LCALL   WRBIT
                LCALL   CEKACK
                LCALL   STOPSE
                MOV     A,@R0
```

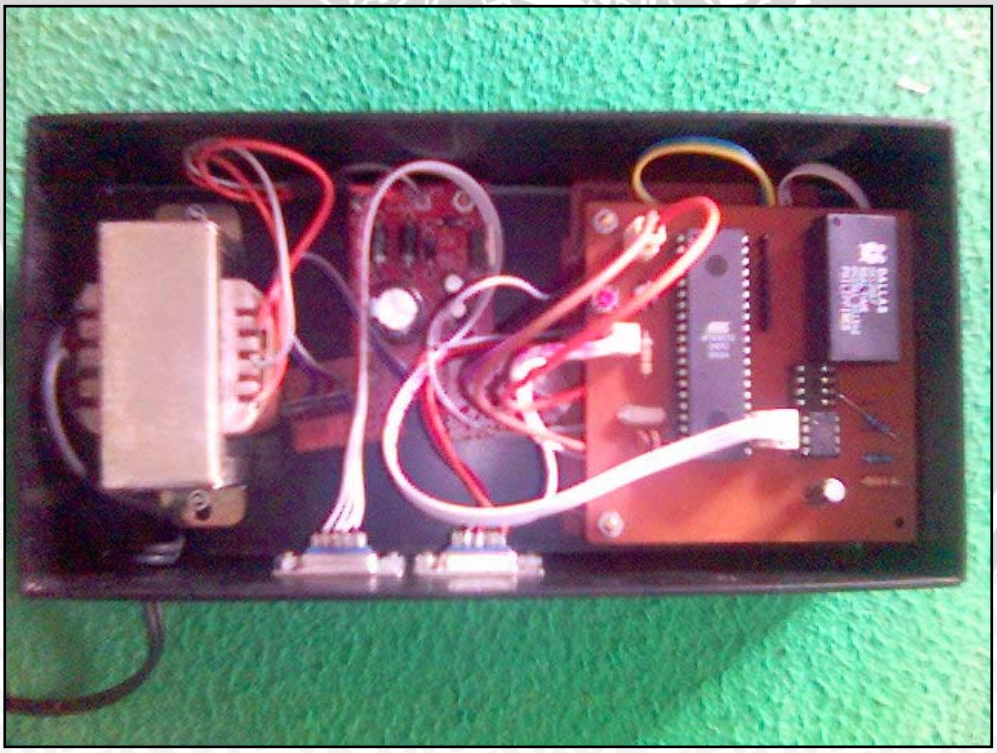
```
                CJNE    A,#10,SIMSLAN
                LJMP    SIMSBAR
SIMSLAN:
                INC     DPTR
                INC     R0
                LJMP    SIMSBAL
SIMSBAR:
                MOV     PTRM1,DPL
                MOV     PTRM2,DPH
                MOV     DPTR,#0000H
                LCALL   STARTWR
                LCALL   CEKACK
                LCALL   SETADR
                MOV     A,PTRM1
                LCALL   WRBIT
                LCALL   CEKACK
                MOV     A,PTRM2
                LCALL   WRBIT
                LCALL   CEKACK
                MOV     A,PTRM3
                LCALL   WRBIT
                LCALL   CEKACK
                LCALL   STOPSE
                RET
;
;**********************************
; SUBPROGRAM TERIMA DATA SERIAL
;**********************************
;
SERIALINT:
                PUSH    PSW             ;simpan di stack pointer
                PUSH    ACC             ;psw,accumulator,dptr
                PUSH    DPH
                PUSH    DPL
                MOV     A,R0
                PUSH    ACC
                JB      RI,SERTAKE      ;ADA DATA BARU?YA! AMBIL DATA TSB
                LJMP    SERBAR
SERTAKE:
                MOV     A,SBUF          ;ambil data serial
                CJNE    A,#'+',SERCEKD  ;isi data='C'?
                MOV     A,DATASER
                CJNE    A,#00H,SERBAR
                MOV     DATASER,#'+'
                LJMP    SERBAR
SERCEKD:
                CJNE    A,#'D',SERCEKC
                MOV     A,DATASER
                CJNE    A,#'+',SERBAR
                LCALL   KIRIMDATASEMUA
                MOV     DATASER,#00H
                LJMP    SERBAR
SERCEKC:
                CJNE    A,#'C',SERBAR
                MOV     A,DATASER
                CJNE    A,#'+',SERBAR
                LCALL   HAPUSDATASEMUA
                MOV     DATASER,#00H
SERBAR:
                POP     ACC
                MOV     R0,ACC
                POP     DPL
                POP     DPH
                POP     ACC
                POP     PSW
                RETI
;
;**********************************
;PENGIRIMAN SEMUA DATA
;**********************************
;
```

```
KIRIMDATASEMUA:
        MOV     DPTR,#0003H
        MOV     R5,PTRM3
        MOV     PTRM3,#00H
        LCALL   STARTWR
        LCALL   CEKACK
        LCALL   SETADR
        LCALL   CEKACK
        LCALL   STARTRD
        LCALL   CEKACK
KIRDSBAL1:
        LCALL   RDBIT
        PUSH    ACC
        LCALL   CEKACK
        POP     ACC
        LCALL   KIRIMSERIAL
        INC     DPTR
        CJNE    A,#0FFFFH,CEKLAN
        INC     PTRM3
        MOV     A,PTRM3
        CJNE    A,#02,CEKLAN
        LJMP    KIRDSBAR
CEKLAN:
        MOV     A,PTRM1
        CJNE    A,DPL,KIRDSBAL1
        MOV     A,PTRM2
        CJNE    A,DPH,KIRDSBAL1
        MOV     A,R5
        CJNE    A,PTRM3,KIRDSBAL1
KIRDSBAR:
        MOV     PTRM3,R5
        RET
;
;****************************************
; HAPUS DATA SEMUA
;****************************************
;
HAPUSDATASEMUA:
        MOV     PTRM1,#03H
        MOV     PTRM2,#00H
        MOV     PTRM3,#00H
        MOV     DPTR,#0000H
        LCALL   STARTWR
        LCALL   CEKACK
        MOV     A,PTRM1
        LCALL   WRBIT
        LCALL   CEKACK
        MOV     A,PTRM2
        LCALL   WRBIT
        LCALL   CEKACK
        MOV     A,PTRM3
        LCALL   WRBIT
        LCALL   CEKACK
        LCALL   STOPSE
        RET
;
;******************************
; MEMEORY READ
;******************************
;
MEMREAD:
        MOV     DPTR,#0000H
        LCALL   STARTWR
        LCALL   CEKACK
        LCALL   SETADR
        LCALL   CEKACK
        LCALL   STARTRD
        LCALL   CEKACK
        LCALL   RDBIT
        PUSH    ACC
        LCALL   CEKACK
```

```
                POP     ACC
                MOV     DATAM,A
                NOP
                LCALL   STOPSE
                RET
;
;**************************
; DELAY SENSOR
;*************************
;
DELAYSENSOR:
                MOV     DELAY1,#0FFH
                MOV     DELAY2,#0FFH
DSKENE:
                DJNZ    DELAY1,DSKENE
                DJNZ    DELAY2,DSKENE
                RET
;
                END
```

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tampilan pada PC pada saat data dalam memori dibaca



Letak komponen rangkaian yang direncanakan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA