TUGAS AKHIR

PERANCANGAN SISTEM MONITORING SUHU RUANGAN JARAK JAUH TANPA KABEL MENGGUNAKAN TRW 2.4 G TRANSMITTER/RECEIVER

MILIK
PERPUSTAKAAN
ITN MALANG

Di Susun Oleh

Nama : Achmad Budi Santoso
NIM : 02.52.006

KONSENTRASI TEKNIK ENERGI LISTRIK
JURUSAN TEKNIK ELEKTRO D-III
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
Maret 2008

ABSTRAK

Perancangan Sistem Monitoring Suhu Ruangan Jarak Jauh Tanpa Kabel (Wireless) Menggunakan TRW 2.4G Transmitter/Receiver. Achmad Budi S, NIM 02.52. 006, Teknik Elektro D-3
Dosen Pembimbing : Komang S, ST, MT

Kata Kunci : Wireless, TRW 2.4G, LM35, Mikrokontroler.

Sistem monitoring suhu ruang memberikan manfaat kepada teknisi gedung untuk mendapatkan informasi secara *real time* keadaan suhu tiap-tiap ruangan agar dapat sesegera mungkin melakukan pengecekan pada ruangan yang sedang mengalami kondisi suhu ruang yang tidak normal. Ada beberapa hal yang dapat mengakibatkan naiknya suhu di dalam ruangan yaitu, AC rusak, kebakaran dll.

Sensor suhu LM35 mengkonversikan suhu sekitar menjadi tegangan analog dengan kenaikan 0,1 V/°C. Tegangan ini kemudian durubah menjadi data digital oleh ADC 0804. Data ini kemudian diolah oleh mikrokontroler AT89S51 menjadi nilai suhu. Selanjutnya, nilai suhu ini diumpankan ke modul transceiver untuk dipancarkan melalui gelombang radio. Adapun proses yang terjadi pada sisi receiver adalah menerima nilai data suhu tersebut melalui gelombang radio. Data ini diambil untuk ditampilkan pada LCD.

Alat ini mudah dalam pengoperasiannya dengan menggunakan tombol *push-button (menu, up, dan down)* untuk *setting*, dan menggunakan *LCD* M1632 sebagai *display*. Pada alat ini buzzer digunakan sebagai alarm terjadinya gangguan yaitu keadaan suhu yang melebihi dari batasan seting yang diinginkan.

Secara sederhana pada pengiriman data menggunakan media udara, data ditumpangkan pada gelombang pembawa melalui proses modulasi dan kemudian ditransmisikan melalui gelombang radio. Dengan memanfaatkan teknologi *wireless* tersebut maka alat ini menjadi efisien dalam penggunaannya.

KATA PENGANTAR

Dengan memanjatkan puja dan puji syukur kehadiran ALLAH S.W.T, karna hanya dengan rahmad dan hidayah-nya maka penyusunan laporan ini dapat menyelesaikan TUGAS AKHIR.

Laporan Tugas Akhir ini dibuat sebagai tugas yang berkaitan dengan syarat kelulusan yang kami tempuh pada semester-semester sebelumnya. Tujuan tugas akhir ini adalah memperdalam, membuktikan menerapkan sekaligus membuktikan Tugas Akhir yang telah disusun atau dirancang dan diharapkan berguna dan bermanfaat dalam memasuki dunia industri. Pada kesempatan ini penyusun menyampaikan rasa terimakasih yang sebesar-besarnya kepada :

1. Bapak Prof, Eng, Ir, Abraham Lomi MSEE Selaku Kepala Rektorat Institut Teknologi Nasional Malang
2. Bapak Ir. Coirul saleh MT Selaku Kepala jurusan Teknik Elektro DIII ITN Malang
3. Bapak Komang Somawwirata ST, MT selaku dosen pembimbing Tugas Akhir
4. Mama, kaka saya yang telah membantu dengan Doa sehingga Laporan dan kompre dapat berjalan dengan lancar dan bermanfaat..
5. Teman-teaman yang telah banyak membantu terselesainya laporan Tugas Akhir ini
6. Dan semua pihak yang telah membantu terselesainya laporan Tugas Akhir ini

Penulis menyadari bahwa banyak kekurangan dalam penyusunan laporan ini, karena itu kritik dan saran yang bersifa membangun sangat kami perlukan demi kesempurnaan laporan ini.

Dan Akhirnya berharap bahwa tulisan ini nantinya dapat berguna bagi semua dan juga untuk penulis sendiri dimana sebagai acuan untuk berikutnya

Malang Maret 2008

Penyusun

DAFTAR ISI

LEMBAR JUDUL

LEMBAR PENGESAHAN

ABSTRAKSI i

KATA PENGANTAR ii

DAFTAR ISI iii

DAFTAR GAMBAR iv

DAFTAR TABEL v

BAB I PENDAHULUAN

1.1. Latar Belakang 1

1.2. Rumusan Masalah 2

1.3. Tujuan 2

1.4. Batasan Masalah 2

1.5. Metodologi Penulisan 3

1.6. Sistematika Penulisan 3

BAB II TEORI DASAR

2.1. Mikrokontroler AT89S51 5

2.1.1. Teori umum. 5

2.1.2. Arsitektur AT89S51. 6

2.1.3. Konfigurasi Pin pada Mikrokontroler AT89S51. 7

2.1.4. Karakteristik Oscillator Inverting. 10

2.1.5. Organisasi Memory. 11

2.1.5.1. Program Memory Internal 11

2.1.5.2. Data Memory (RAM) Internal. 11

2.1.5.3. SFR (Special Function Register). 13

2.2. IC LM35 14

2.3. ADC (*Analog to Digital Conventer*) 15

2.4. Dioda 16

2.4.1. Dioda Penyearah 16

2.5. Kapasitor 17

2.6. Resistor 19

2.7. LCD (Liquid Crystal Display) M1632 21

2.7.1. Sinyal interface M1632 21

2.7.2. Mengatur tampilan M1632 25

2.8. Modul TRW 2.4 G 30

BAB III PERENCANAAN DAN PEMBUATAN ALAT

3.1. Diagram Blok 32

3.3. Mikrokontroller AT89S51 34

3.2. Prinsip Kerja Alat 34

3.3.1. Pemetaan *Memory* 35

3.3.2. Rangkaian *Clock* 36

3.3.3. Rangkaian *Reset* 36

3.3.4. Hubungan *Pin* Pada AT89S51 (pada rangkaian transmiter) 38

3.3.5. Hubungan *Pin* Pada AT89S51 (pada rangkaian receiver) 39

3.4. Perancangan Rangkaian TRW 24-G 40

3.5. Rangkaian Sensor 41

3.6. Pengkondisian Sinyal 42

3.6.1. *Non Inverting Amplifier (penguat tak membalik)* 42

3.7. Perancangan *Analog To Digital Converter* (ADC) 0804 43

3.8. Perancangan Rangkaian LCD 45

3.9. Perangkat Lunak 49

BAB IV PENGUJIAN ALAT

4.1. Pengujian Rangkaian *Power Supply* 51

4.2. Pengujian Rangkaian TRW 2.4G 54

4.3. Pengujian Rangkaian Sensor 55

4.3.1 Analisa Data Hasil Perhitungan Rangkaian Sensor Suhu 57

4.3.2 Kesalahan Relatif 59

4.3.3 Hasil perhitungan validasi sensor suhu dengan alat lain 60

4.4 Pengujian Rangkaian ADC 0804 62

4.4.1. Tujuan. 62

4.4.2. Peralatan Yang Digunakan. 62

4.4.3. Langkah Pengujian. 63

4.4.4. Hasil Pengujian 64

4.5. Pengujian Terhadap Rangkaian *Push-button* 64

4.6. Pengujian Rangkaian *Driver Buzzer* 65

BAB V PENUTUP

5.1. Kesimpulan 67

5.2. Saran 67

DAFTAR PUSTAKA

DAFTAR GAMBAR

Gambar 2.1. Diagram Blok AT89S51 5

Gambar 2.2. Konfigurasi Pin-Pin AT89S51 7

Gambar 2.3. Karakteristik Oscilator 10

Gambar 2.4. Organisasi RAM Internal 12

Gambar 2.5. IC LM35 14

Gambar 2.6. ADC 0804 16

Gambar 2.7. Simbol Dioda 17

Gambar 2.8. Simbol Kapasitor 18

Gambar 2.11. Mengirim/Mengambil Data Ke/Dari M1632 22

Gambar 2.13. Rangkaian LCD M1632 27

Gambar 2.14. Modul TRW 2.4G 31

Gambar 3.1. Diagram Blok 32

Gambar 3.2. Rangkaian *Clock* AT89S51 Gambar 3.2. Rangkaian *Clock* AT89S51 36

Gambar 3.3. Rangkaian *Reset* 37

Gambar 3.4. Hubungan *Pin* Pada AT89S51 38

Gambar 3.5. Hubungan *Pin* Pada AT89S51 39

Gambar 3.6. Rangkaian Level Converter 41

Gambar 3.7. Rangkaian Sensor LM35 42

Gambar 3.8. Penguat tegangan tak membalik. 44

Gambar 3.9. Rangkaian ADC 0804 46

Gambar 3.10. Rangkaian LCD Pada *Mikrokontroller* 49

Gambar 3.11 Flowchart Program alat 51

Gambar 4.1. Cara pengukuran tegangan pada rangkaian catu daya 53

Gambar 4.2. Diagram Blok Pengujian TRW 2.4G 56

Gambar 4.3. Grafik pengambilan data temperatur 58

Gambar 4.4. Diagram Blok Pengujian ADC 0804 61

Gambar 4.5. Rangkaian Pengujian ADC 0804 61

Gambar 4.6. Cara Pengukuran Tegangan Pada Pengujian Rangkaian *Push-Button*..63

Gambar 4.7. Rangkaian Pengujian Rangkaian *Driver Buzzer* 64

DAFTAR TABEL

Tabel 2-1. Pengaturan RS0-RS1 Untuk Select Register Bank 12

Tabel 2-2. 128 Byte Special Function Register 13

Tabel 2-3. Tabel Gelang Warna Resistor 19

Tabel 2-4. Fungsi Pin – Pin LCD 28

Tabel 4-1. Hasil Pengukuran Rangkaian *Power Supply* 54

Tabel 4-2. Hasil Pengukuran *Vout* Sensor 57

Tabel 4-3. Perbandingan Hasil Pengukuran dan Perhitungan V_{out} 60

Tabel 4-4. Hasil Pengujian ADC 0804 dengan LED indikator. 62

Tabel 4-5. Hasil Pengukuran Pada Rangkaian *Push-Button* 63

Tabel 4-6. Hasil Pengujian Rangkaian *Driver Buzzer* 64

BAB I

PENDAHULUAN

1.1. Latar Belakang

Sistem monitoring suhu ruangan merupakan suatu aplikasi yang dapat memonitor besaran nilai suhu °C dalam ruangan. Dengan mengetahui besaran nilai suhu tersebut kita dapat mengetahui kondisi suhu didalam ruangan tersebut apakah normal atau tidak. Pada suatu gedung bertingkat baik itu mall, hotel, apartemen dan lain-lainnya, sistem monitoring suhu ruang pada tiap-tiap ruangan adalah merupakan hal yang sangat penting untuk dapat menjaga kenyaman bagi para penghuni gedung tersebut. Sistem monitoring suhu ruang memberikan manfaat kepada teknisi gedung untuk mendapatkan informasi secara *real time* keadaan suhu tiap-tiap ruangan agar dapat sesegera mungkin melakukan pengecekan pada ruangan yang sedang mengalami kondisi suhu ruang yang tidak normal. Ada beberapa hal yang dapat mengakibatkan naiknya suhu di dalam ruangan yaitu, AC rusak, kebakaran dll.

Apabila jumlah ruang pada suatu gedung sangat banyak, maka dapat dipastikan jumlah kabel yang dibutuhkan untuk aplikasi ini akan menjadi sangat banyak pula. Dan proses pengkabelan pun menjadi lebih sulit karena kalau tidak dilakukan dengan sangat rapi maka dapat menggangu pemandangan gedung tersebut. Berdasarkan uraian tersebut maka penulis mengambil judul Tugas Akhir "Perancangan Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver*". Secara sederhana pada pengiriman data menggunakan media udara, data ditumpangkan pada gelombang

pembawa melalui proses modulasi dan kemudian ditransmisikan melalui gelombang radio. Dengan memanfaatkan teknologi *wireless* tersebut maka alat ini menjadi efisien dalam penggunaannya.

1.2. Rumusan Masalah

Dari uraian tersebut diatas maka timbul beberapa permasalahan diantaranya adalah:

1. Bagaimana merencanakan dan membuat *hardware* dan *software* untuk Perancangan Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver* ?
2. Bagaimana merancang sensor suhu?
3. Bagaimana menampilkan kerja alat pada LCD ?

1.3. TUJUAN

Perancangan Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver* bertujuan untuk memberikan kemudahan-kemudahan dalam proses monitoring suhu ruangan secara terpusat menggunakan teknologi *wireless*.

1.4. BATASAN MASALAH

Agar permasalahan tidak meluas maka penulis membatasi permasalahan sebagai berikut:

1. Jarak / jangkauan mengontrol sensor suhu maksimal 20meter.
2. Tidak membahas komunikasi data melalui RF (Radio Frekuensi).
3. Tidak membahas bahasa pemrograman ASCII

1.5. Metodologi Penulisan

Metodologi yang dilakukan dalam penulisan tugas akhir Perancangan Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver* adalah sebagai berikut:

1. Kajian pustaka, meliputi konsep dasar dan teori yang digunakan
2. Perancangan dan pembuatan aplikasi meliputi perancangan model, proses pembuatan disertai diagram alirnya
3. Pengujian tentang aplikasi yang sedang dibuat
4. Membuat kesimpulan serta saran dari hasil pengujian aplikasi

1.6. SISTEMATIKA PENULISAN

Untuk mempermudah dan memperjelas pembahasan dari laporan tugas akhir ini penulis menerapkan system penulisan seperti dibawah ini :

BAB I : PENDAHULUAN

Berisi latar belakang permasalahan, rumusan masalah, tujuan pembahasan, batasan masalah, metodologi penulisan dan sistematika penulisan.

BAB II : DASAR TEORI

Membahas tentang teori dasar rangkaian yang digunakan, mikrokontroler, hardware dan teori dasar alat-alat pendukung lainnya.

BAB III : PERENCANAAN DAN PEMBUATAN ALAT

Perancangan Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver*.

BAB IV : PENGUJIAN ALAT

Berisi tentang uji coba alat yang telah dibuat, pengoperasian dan spesifikasi alat.

BAB V : PENUTUP

Merupakan kesimpulan dari pembahasan pada bab–bab sebelumnya dan kemungkinan pengembangan alat.

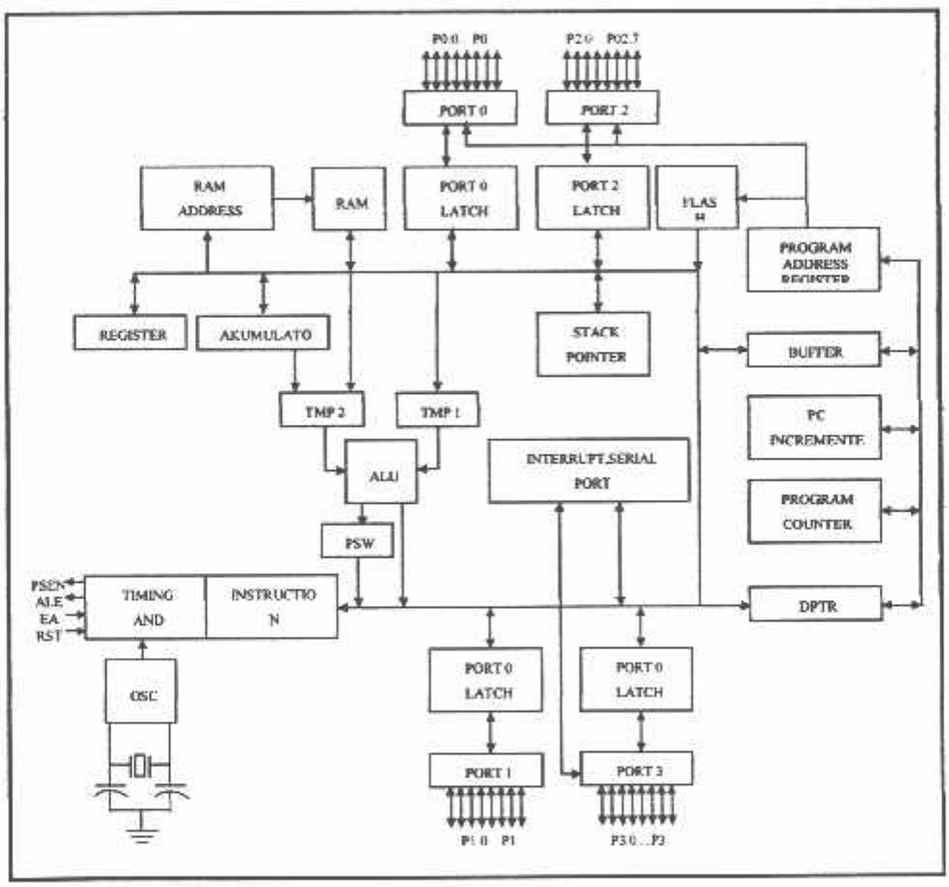
BAB II

TEORI DASAR

Bab ini akan menguraikan tentang dasar-dasar teori dasar yang menunjang dalam perancangan dan pembuatan alat. Uraian teori dalam bab ini meliputi teori IC AT89S51, LM35, LCD M1632, TRW 2.4 G, ADC 0804 dan perangkat-perangkat pendukung lainnya.

2.1. Mikrokontroler AT89S51

2.1.1. Teori umum.



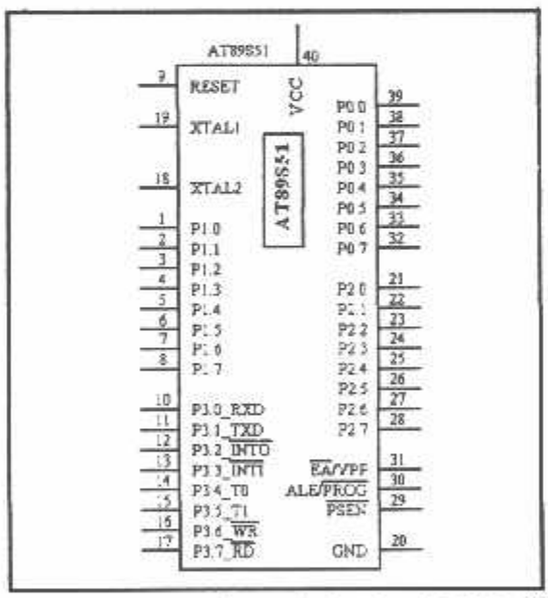
Gambar 2.1. Diagram Blok AT89S51 [5]

- 4 bank register. masing-masing berisi 8 register.
- 16 Byte yang dapat dialamati pada bit level.
- 80 Byte general purpose memory data.

- 32 pin input-output tersusun atas P0-P3 masing-masing 8-bit.
  - 2 Buah 16-bit timer counter.
  - Receiver Register, yaitu : TCON, TMOP, SCON, IP, dan IE.
  - 5 Buah sumber interupt (2 buah sumber interupt eksternal dan 3 buah sumber interupt internal).
  - Oscilator dan Clock internal.

2.1.3. Konfigurasi Pin pada Mikrokontroler AT89S51.

Konfigurasi kaki-kaki mikrokontroler AT89S51 terdiri dari 40 pin, seperti terlihat pada gambar 2.2 sebagai berikut.



Gambar 2.2. Konfigurasi Pin-Pin AT89S51 [5]

AT89S51 merupakan sebuah mikrokontroller 8-bit CMOS, *Low Power* dengan 4Kb flash Programmable and *Erasable Read Only Memory* (PEROM). IC ini dibuat sesuai dengan standart industri konfigurasi pin dan intruction set dari MCS-51.

- 4Kb Flash Memory.
- 128 Byte Internal RAM.
- 32 I/O Lines.
- 2 Timer/Counter 16-Level.
- 1 Serial Port Full Duplex.
- On Chip Osilator.

AT89S51 mempunyai dua buah *Power-Saving mode* yang dapat diatur melalui software, yaitu: IDE Mode yang akan menghentikan CPU sebagai RAM, Timer/Counter, Serial Port dan Interrupsi system tetap berfungsi. Power Down Mode yang akan menyimpan ini di RAM, tetapi menahan Oscilator untuk tidak mengaktifkan chip yang lain sampai terjadi reset secara *hardware*.

2.1.2. Arsitektur AT89S51.

Arsitektur Mikrokontroler AT89S51 adalah sebagai berikut:

- CPU (Central Processing Unit) 8-bit dengan register A (*Accumulator*) dan B.
- 16-Bit Program Counter (PC) dan data pointer (DPTR).
- 8-Bit Program status word (PSW).
- 4-Bit Stack pointer (SP).
- 4 Kbyte internal EPROM
- 128 byte internal RAM.

Fungsi dari tiap-tiap pin adalah sebagai berikut:

1. VCC (Supply tegangan).
2. GND (Ground).
3. Port 0

   Merupakan port input-output dua arah dan dikonfigurasikan sebagai multiplex dua bus alamat rendah (A0 – A7) dan data selama pengaksesan program memory dan data memory internal.
4. Port 1

   Merupakan port input-output dua arah dengan internal pull-up.
5. Port 2

   Merupakan port input-output dua arah dengan internal pull-up. Mengeluarkan address tinggi selama pengambilan (fetch) program memory internal dan selama pengaksesan ke data memory port 2 mengeluarkan isi P2SFR (*Special Function Register*) menerima addres tinggi dan beberapa sinyal kontrol selama pemograman dan veritifikasi.
6. Port 3.

   Merupakan port input-output dua arah dengan internal pull-up. Port 3 juga memiliki fungsi khusus, yaitu:

   - RXD (P3.0) : Port input serial.
   - TXD (P3.1) : Port out-put serial.
   - INTO (P3.2) : Interrup 0 external.
   - INTI (P3.3) : Internal 1 external.
   - TO (P3.4) : Input external timer 0.
   - TXD (P3.1) : Port out-put serial.

- INTO (P3.2) : Interrupt 0 external.
- INTI (P3.3) : Interrupt 1 external.
- TO (P3.4) : Input external timer 0.
- T1 (P3.5) : Input external timer 1.
- WR (P3.6) : Strobe tulis data memory external.
- RD (P3.7) : Strobe baca data memory external.

7. RST

   Input reset.

8. ALE\PROG

   Pulsa output ALE digunakan untuk proses "*latching*" byte addres rendah (A0 - A7) selama pengaksesan ke external memory. Pin ini juga untuk memasukkan pulsa program selama pemrograman.

   Pada operasi normal ALE mengeluarkan rate konstant yaitu 16 frekuensi osilasi dan boleh digunakan untuk timing external.

9. PSEN

   Merupakan strobe baca ke program memory external.

10. EA\VPP

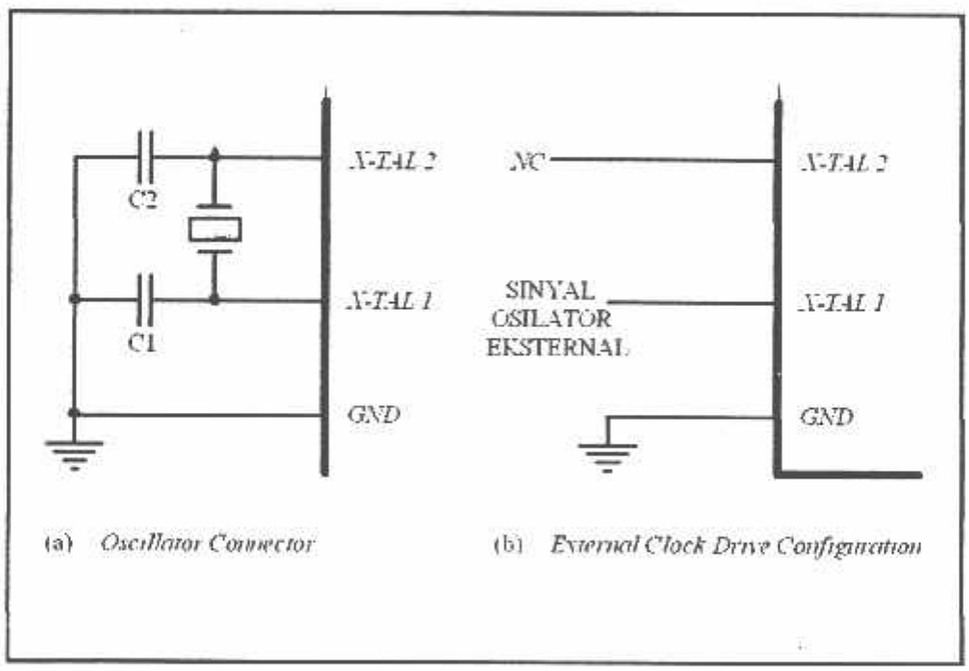
    External addres enable EA digroundkan jika mengakses memory external. Untuk mengakses memory internal maka dihubungkan ke VCC.

11. XTAL 1 dan XTAL 2.

    Kaki ini dihubungkan dengan kristal bila menggunakan osilator internal. XTAL 1 merupakan input inverting osilator amplifier sedangkan XTAL 2 merupakan out-put inverting osilator amplifier.

2.1.4. Karakteristik Oscillator Inverting.

X-TAL 1 dan X-TAL 2 secara berurutan merupakan input dan output dari sebuah *inverting amplifier* yang dapat dikonfigurasikan penggunaannya sebagai on chip oscillator seperti yang ditunjukkan pada gambar 2-3a. X-TAL 1 dan X-TAL 2 ini dapat menggunakan sebuah *kristal quartz* maupun resonator keramik.Untuk memberikan AT89S51 dari sumber clock external. Maka pin X-TAL 2 dibiarkan tidak berhubungan dan X-TAL 1 dihubungkan dengan sumber clock external seperti pada gambar 2-3b. Rangkaian ini tidak melakukan duty cycle dari setiap sinyal clock internal, karena input bagi masukan rangkaian clock internal dihubungkan ke flip-flop pembagi dua, tetapi spesifikasi nilai tegangan pada saat tinggi dan rendah, maksimum dan minimumnya harus diberikan.



Gambar 2.3. Karakteristik Oscilator [5]

2.1.5. Organisasi Memory.

Didalam AT89S51 memiliki ruangan alamat telah dibedakan untuk program memory dan data memory. Pemisahan memori program dan data tersebut membolehkan memori data diakses dengan alamat 8-bit, sehingga dapat dengan cepat dan mudah disimpan dan dimanipulasi oleh CPU 8-bit. Namun demikian, alamat memori data 16-bit bisa juga dihasilkan melalui register DPTR.

2.1.5.1. Program Memory Internal

AT89S51 memiliki program memory internal sebesar 4Kbyte dengan ruang alamat 0000H-0FFFH. Jika alamat-alamat program lebih tinggi daripada 0FFFH, yang melebihi kapasitas ROM\Fash memory internal menyebabkan AT89S51 secara otomatis mengambil Code Byte dari program memory external. Code Byte juga dapat diambil hanya dari external memory dengan alamat 0000H-FFFFH dengan cara menghubungkan Pin EA ke Ground.

2.1.5.2. Data Memory (RAM) Internal.

Ruang alamat bawah memory data (RAM) internal dengan kapasitas 128 byte yaitu 00H-7FH yang terbagi atas 3 daerah, yaitu :

- 4 Bank Register

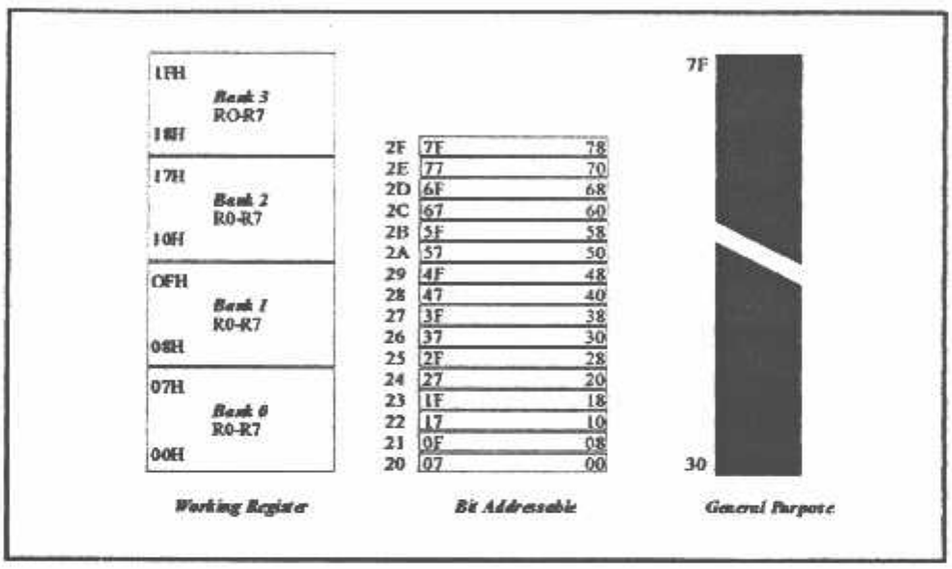
  Setiap bank terdiri dari 8 register (R0-R7). Sehingga jumlah register untuk keempat bank register (bank 0 - bank 3) menjadi 32 buah register yang menempati ruang alamat 00H-1FH. Mengaktifkan salah satu bank register dapat dilakukan dengan mengatur RSO-RSI pada PSW (Program Status Word).

- Bit Addressable.

  Terdiri dari 16 byte yang berada pada alamat 20H-2FH. Masing-masing 128 bit lokasi ini dapat di alamatkan secara langsung.

- General Purpose.

  Terdiri dari 80 byte yang menempati alamat 30H-7FH. yang dapat dialamati secara langsung dan dapat digunakan untuk keperluan umum (General Purpose RAM). Misalkan digunakan untuk lokasi stack.



Gambar 2.4. Organisasi RAM Internal [5]

Tabel 2-1. Pengaturan RS0-RS1 Untuk Select Register Bank [5]

RS1	RS0	Select Register Bank
0	0	Bank 0
0	1	Bank 1
1	0	Bank 2
1	1	Bank 3

2.1.5.3. SFR (Special Function Register).

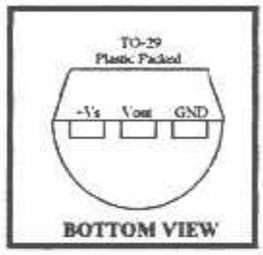
Untuk mengoperasikan AT89S51 yang tidak menggunakan alamat internal RAM (00-7FH) dilakukan oleh SFR yang beraddres 80H-FFH, tetapi tidak semua address tersebut digunakan sebagai SFR. Fungsi-fungsi SFR dijelaskan oleh tabel 2 - 2 sebagai berikut.

Tabel 2-2. 128 Byte Special Function Register [5]

SYMBOL	NAME	ADDRESS
ACC	ACCUMULATOR	0E0H
B	B REGISTER	0F0H
PSW	PROGRAM STATUS WORD	0D0H
SP	STACK POINTER	81H
DPTR	DATA POINTER 2 BYTE	
DPL	LOW BYTE	82H
DPH	HIGH BYTE	83H
P0	PORT 0	80H
P1	PORT 1	90H
P2	PORT 2	0A0H
P3	PORT 3	080H
IP	INTERUPT PERIORITY CONTROL	088H
IE	INTERUPT ENABLE CONTROL	0ABH
TMOD	TIMER / COUNTER MODE CONTROL	89H
TCON	TIMER/COUNTER CONTROL	88H
TH0	TIMER/COUNTER 0 HIGH CONTROL	8CH
TL0	TIMER/COUNTER 0 LOW CONTROL	8DH
TH1	TIMER/COUNTER 1 HIGH CONTROL	8DH
TL1	TIMER/COUNTER 1 LOW CONTROL	8CH
SCON	SERIAL CONTROL	98H
SBUF	SERIAL DATA BUFFER	99H
PCON	POWER CONTROL	87H

2.3. IC LM35

IC LM35 merupakan seri integrasi yang digunakan sebagai sensor temperatur dengan output sebanding dengan derajat celcius dari temperatur. IC LM35 berisi zener yang sensitif terhadap suhu, dan tegangan zenernya berubah linier dari suhu sekelilingnya. Pada IC LM35 menghasilkan tegangan output sebesar 10 mV untuk setiap kenaikan suhu sebesar 1^{0}C. Gambar IC LM35 yang digunakan pada rangkaian pendeteksi suhu ruangan.



Gambar 2.5. IC LM35[13]

IC LM35 memiliki kelebihan dengan kelinieran suhu yang cukup tinggi yang disesuaikan dengan derajat kelvin, sehingga dalam penggunaannya tidak harus mengurangi besar tegangan konstan dari tegangan output pada skala yang tepat. Dalam pemakaiannya IC ini tidak banyak kesulitan dalam pengkalibrasian secara ekstra dan memberikan keakurasian khusus pada temperatur $\pm \frac{1}{4}{}^{0}$C dan $\pm \frac{3}{4}{}^{0}$C. Dengan *range* yang lebih besar yaitu antara -55^{0}C sampai + 150^{0}C. IC LM35 mempunyai impedansi output yang rendah, output temperatur yang *linier* dapat diberikan *supply* tegangan tunggal maupun tegangan *simetris* (tegangan *plus* dan tegangan *mimus*).

IC LM35 memiliki beberapa macam tipe, antara lain LM35A, LM35C, LM35CA, LM35D. dari macam-macam tipe tersebut tersedia dalam beberapa package yaitu, TO-46 yang bentuknya seperti transistor, *package* TO-29 berupa

package plastic seperti transistor dan dalam bentuk IC 8 pin dengan permukaan agak menonjol.

Karakteristik dari tranduser suhu ini sebagai berikut :

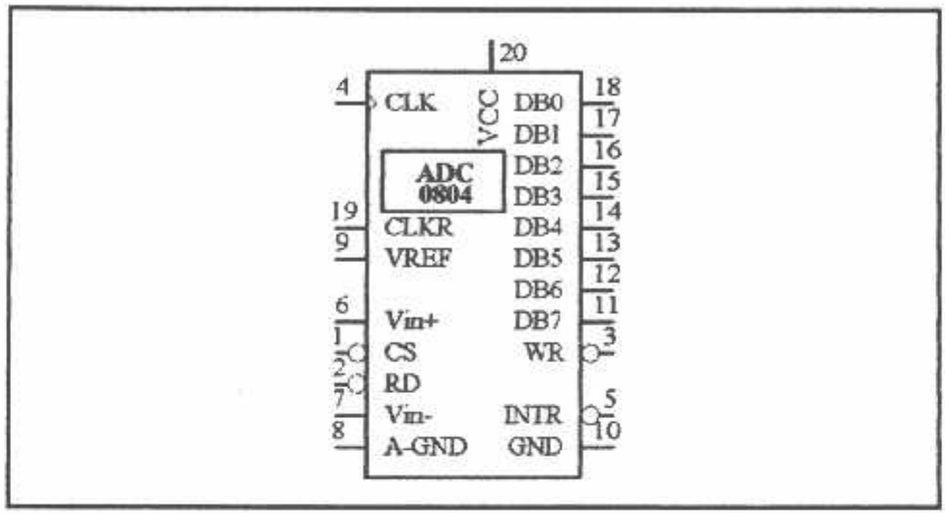
- Penunjukan kalibrasi dalam derajat celcius (^{0}C).
- Skala *linier* 10mV/ ^{0}C.
- Memiliki range temperatur antara -55 ^{0}C sampai +150 ^{0}C.
- Dioperasikan dengan tegangan 4 - 30 Volt.
- Minimum pemanasan sendiri dalam udara 0,08 ^{0}C.
- Keakuratan yang digunakan 0,5 ^{0}C (di + 25 ^{0}C).
- Minimum impedansi output 0,1 Ohm untuk 1 mA.

2.3. ADC (*Analog to Digital Conventer*)

Analog to Digital Conventer (ADC) adalah sebuah perangkat elektronik yang berfungsi untuk merubah besaran analog (tegangan/arus) menjadi data digital. ADC digunakan sebagai interface untuk untuk perangkat digital untuk komunikasi dengan perangkat analog. Dalam tugas akhir ini digunakan ADC 0804 yang merupakan sebuah input dengan data keluaran 8 buah. ADC 0804 merupakan 8 bit conventer A/D dengan output digital paralel 8 bit. Dan input dapat dimultiplex secara aksternal menjadi 8 inputan analog.

ADC 0804 memiliki keunggulan yaitu kompitibel dengan mikrokontroller dan mikroprocesor, memiliki input tegangan differensial, dapat dioperasikan pada range tegangan input sebesar 0 - 5 Vdc dengan tegangan referensi sebesar 2,5 Vdc, mempunyai resolusi 8 bit, mudah dalam interface ke mikrokontroller

maupun mikroprocesor. Gambar dari ADC 0804 dapat dilihat pada gambar sebagai berikut:



Gambar 2.6. ADC 0804 [13]

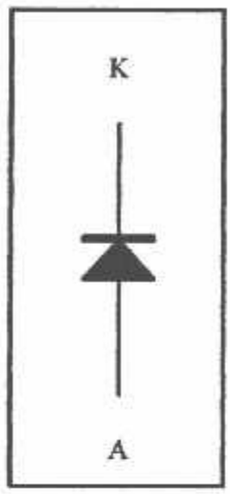
2.4. Dioda

Semua dioda prinsip kerjanya adalah sebagai penyearah. Tetapi karena proses pembuatan, bahan dan penerapannya yang berbeda-beda, maka namanya pun bebeda-beda serta memiliki karakteristik yang berbeda pula. Macam-macam dioda antara lain: Dioda Penyearah, Dioda Zener, Dioda Tunnel, Dioda Varaktor dan Dioda Photo.

2.4.1. Dioda Penyearah

Kebanyakan dioda penyearah dibuat dari bahan silicon, mengingat kemampuan dan keandalannya yang tinggi. Mampu dilewati arus yang besar dan dapat bekerja pada suhu yang tinggi. Keuntungan ini dimungkinkan karena elektron tersebut baru bias terlepas dari intinya oleh kekuatan atau energi yang

sangat besar. Penyearah disini berarti merubah arus bolak-balik (AC) menjadi arus searah (DC) yang berfrekuensi rendah. Gambar simbol dioda seperti dibawah ini:



Gambar 2.7. Simbol Dioda [8]

2.5. Kapasitor

Kapasitor adalah komponen elektronika yang terdiri dari plat konduktor yang satu disekat terhadap yang lainnya dengan isolator (dielektrik) dan berfungsi sebagai pelawan untuk arus bolak-balik, filter, tuning, kopel antar rangkaian, pembangkit gelombang sinus dan konduktor khusus. Untuk menentukan banyaknya muatan listrik yang dapat disimpan oleh kapasitor, dapat dirumuskan sebagai berikut:

$Q = C \times V$ .....(2-3)

Dimana :

Q = Banyaknya muatan dalam Coulomb

C = Kapasitor dalam farad

V = Tegangan yang ada diantara keping keping dalam Volt

Gambar 2.8. Simbol Kapasitor [8]

Sebuah kapasitor dikatakan mempunyai kapasitas satu farad apabila muatan sebesar satu couloumb membuat tegangan naik sebesar satu volt, apabila farad ternyata terlampau besar sehingga digunakan mikrofarad dan satuan lainnya yaitu nano farad dan pikofarad. Kapasitor yang berfungsi sebagai penyimpan tenaga listrik dalam praktek dapat digunakan untuk pewaktu, untuk pewaktu kapasitor dengan resistor yang disebut sebagai rangkaian sebagai rangkaian RC.

Rumus pada rangkaian RC sebagai berikut:

$$RC = R \times C \quad \text{.....(2-4)}$$

Dimana:

R = Tahanan dalam kapasitor

C = Kapasitor dalam farad

RC = Konstanta waktu dalam detik

2.6. Resistor

Pada dasarnya semua bahan memiliki sifat resistif namun beberapa bahan seperti tembaga, perak, emas dan bahan metal umumnya memiliki resistansi yang sangat kecil. Bahan-bahan tersebut menghantar arus listrik dengan baik, sehingga dinamakan konduktor. Kebalikan dari bahan yang konduktif, bahan material seperti karet, gelas, karbon memiliki resistansi yang lebih besar menahan aliran elektron dan disebut sebagai insulator.

Tabel 2-4. Tabel Gelang Warna Resistor [10]

Warna	Nilai	Faktor Pengali	Toleransi
Hitam	0	1	
Coklat	1	10	1%
Merah	2	100	2%
Jingga	3	1.000	
Kuning	4	10.000	
Hijau	5	100.000	
Biru	6	10^6	
Violet	7	10^7	
Abu-abu	8	10^8	
Putih	9	10^9	
Emas	-	0.1	5%
Perak	-	0.01	10%

Resistor adalah komponen dasar elektronika yang digunakan untuk membatasi jumlah arus yang mengalir dalam satu rangkaian. Sesuai dengan namanya resistor bersifat resistif dan umumnya terbuat dari bahan karbon. Dari hukum Ohm diketahui, resistansi berbanding terbalik dengan jumlah arus yang mengalir melaluinya. Satuan resistansi dari suatu resistor disebut Ohm atau dilambangkan dengan simbol Ω (Omega).

Tipe resistor yang umum adalah berbentuk tabung dengan dua kaki tembaga di kiri dan kanan. Pada badannya terdapat lingkaran membentuk gelang kode warna untuk memudahkan pemakai mengenali besar resistansi tanpa mengukur besarnya dengan Ohm meter. Kode warna tersebut adalah standar manufaktur yang dikeluarkan oleh EIA (Electronic Industries Association) seperti yang ditunjukkan pada tabel 2-4.

Resistansi dibaca dari warna gelang yang paling depan ke arah gelang toleransi berwarna coklat, merah, emas atau perak. Biasanya warna gelang toleransi ini berada pada badan resistor yang paling pojok atau juga dengan lebar yang lebih menonjol, sedangkan warna gelang yang pertama agak sedikit ke dalam. Dengan demikian pemakai sudah langsung mengetahui berapa toleransi dari resistor tersebut. Kalau kita telah bisa menentukan mana gelang yang pertama selanjutnya adalah membaca nilai resistansinya.

Jumlah gelang yang melingkar pada resistor umumnya sesuai dengan besar toleransinya. Biasanya resistor dengan toleransi 5%, 10.Tetapi resistor dengan toleransi 1% atau 2% (toleransi kecil) memiliki 4 gelang (tidak termasuk gelang toleransi). Gelang pertama dan seterusnya berturut-turut menunjukkan besar nilai satuan, dan gelang terakhir adalah faktor pengalinya.

Misalnya resistor dengan gelang kuning, violet, merah dan emas. Gelang berwarna emas adalah gelang toleransi. Dengan demikian urutan warna gelang resitor ini adalah, gelang pertama berwarna kuning, gelang kedua berwana violet dan gelang ke tiga berwarna merah. Gelang ke empat tentu saja yang berwarna emas dan ini adalah gelang toleransi. Dari tabel 2-3 diketahui jika gelang toleransi berwarna emas, berarti resitor ini memiliki toleransi 5%. Nilai

resistansisnya dihitung sesuai dengan urutan warnanya. Pertama yang dilakukan adalah menentukan nilai satuan dari resistor ini. Karena resitor ini resistor 5% (yang biasanya memiliki tiga gelang selain gelang toleransi), maka nilai satuannya ditentukan oleh gelang pertama dan gelang kedua.

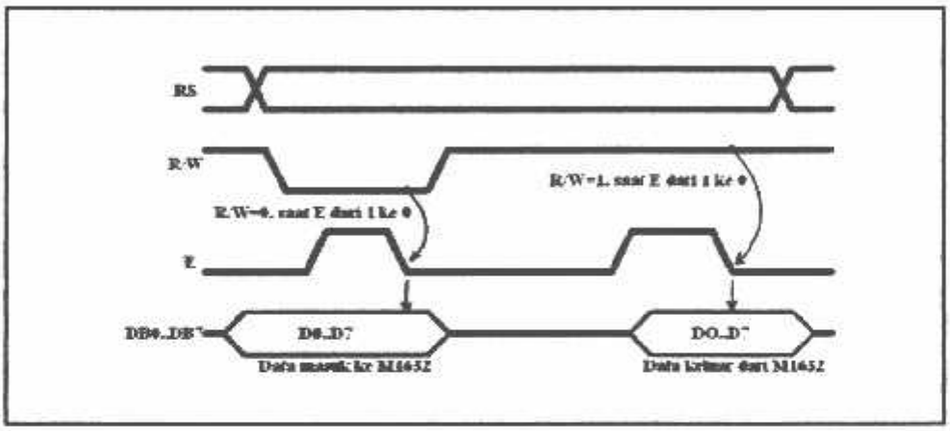
2.7. LCD (Liquid Crystal Display) M1632

LCD Display Module M1632 buatan Seiko Instrument Inc. terdiri dari dua bagian, yang pertama merupakan panel LCD sebagai media penampil informasi dalam bentuk huruf/angka dua baris, masing-masing baris bisa menampung 16 huruf/angka. Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibalik pada panel LCD, berfungsi mengatur tampilan informasi serta berfungsi mengatur komunikasi M1632 dengan mikrokontroler yang memakai tampilan LCD itu. Dengan demikian pemakaian M1632 menjadi sederhana, sistem lain yang M1632 cukup mengirimkan kode-kode ASCII dari informasi yang ditampilkan seperti layaknya memakai sebuah printer.

2.7.1. Sinyal interface M1632

Untuk berhubungan dengan mikrokontroler pemakai, M1632 dilengkapi dengan 8 jalur data (DB0..DB7) yang dipakai untuk menyalurkan kode ASCII maupun perintah pengatur kerjanya M1632. Selain itu dilengkapi pula dengan E, R/W dan RS seperti layaknya komponen yang kompatibel dengan mikroprosesor. Kombinasi lainya E dan R/W merupakan sinyal standar pada komponen buatan Motorola. Sebaliknya sinyal-sinyal dari MCS51 merupakan sinyal khas Intel dengan kombinasi sinyal WR dan RD..

RS, singkatan dari Register Select, dipakai untuk membedakan jenis data yang dikirim ke M1632, kalau **RS=0** data yang dikirim adalah perintah untuk mengatur kerja M1632, sebaliknya kalau **RS=1** data yang dikirim adalah kode ASCII yang ditampilkan. Demikian pula saat pengambilan data, saat RS=0 data yang diambil dari M1632 merupakan data status yang mewakili aktivitas M1632, dan saat **RS=1** maka data yang diambil merupakan kode ASCII dari data yang ditampilkan. Proses mengirim/mengambil data ke/dari M1632 digambarkan dalam gambar 2.9 bisa dijabarkan sebagai berikut :



Gambar 2.9. Mengirim/Mengambil Data Ke/Dari M1632 [7]

1. **RS** harus dipersiapkan dulu, untuk menentukan jenis data seperti yang telah dibicarakan di atas.
2. **R/W** di-nol-kan untuk menandakan akan diadakan pengiriman data ke M1632. Data yang akan dikirim disiapkan di **DB0..DB7**, sesaat kemudian sinyal **E** di-satu-kan dan di-nol-kan kembali. Sinyal **E** merupakan sinyal sinkronisasi, saat **E** berubah dari 1 menjadi 0 data di **DB0 .. DB7** diterima oleh M1632.

3. Untuk mengambil data dari M1632 sinyal **R/W** di-satu-kan, menyusul sinyal **E** di-satu-kan. Pada saatu **E** menjadi 1, M1632 akan meletakkan datanya di **DB0 _ DB7**, data ini harus diambil sebelum sinyal **E** di-nol-kan kembali.

Proses pengiriman data ke M1632 dijelaskan sebagai berikut :

1. Perbedaan sub-rutin **Kirim Perintah** dan **Kirim ASCII** terletak pada nilai **RS** pada saat sub-rutin itu bekerja. Sub-rutin **Kirim Perintah** bekerja dengan **RS='0'** (baris 6), data yang dikirim AT89C2051 diterima M1632 sebagai perintah untuk mengatur kerja M1632. Sub-rutin **Kirim ASCII** bekerja dengan **RS='1'** (baris 10), data yang dikirim AT89C2051 diterima M1632 sebagai kode ASCII yang akan ditampilkan
2. Sinyal **RW** di-nol-kan agar M1632 siap menerima data (baris 12), setelah itu data di akumulator **A** diletakkan di **D0_D7** (Port 1 dari AT89C2051) di baris 13, baris 14 dan 15 membangkitkan sinyal sinkronisasi **E** dengan cara membuat **P3.4** menjadi '1' dan kemudian kembali menjadi '0'. Saat sinyal **E** kembali menjadi '0' data di Port 1 akan diterima oleh M1632.
3. Selesai mengirim data, program harus menunggu sampai M1632 siap menerima data lagi. Hal ini dilakukan dengan cara mengambil Status M1632 (baris 18), dan memeriksa bit 7-nya (baris 19), selama bit 7 bernilai '1' berarti M1632 masih sibuk mengurus diri, dan program menunggunya di Tunggu Dulu.

Proses pengambilan data dari M1632 dijelaskan sebagai berikut :

1. Seperti bahasan di atas, **RS** dipakai untuk memilih Register Perintah/Status, sub-rutin **Ambil Status** bekerja dengan **RS=0** dan sub-rutin **Ambil ASCII** bekerja dengan **RS='1'**.

2. Sinyal **RW** di-satu-kan agar M1632 siap memberi data (baris 29), setelah sinyal **E** menjadi '**1**' (baris 30) M1632 akan meletakkan data di **D0 _ D7**, setelah data ini diambil (baris 31) sinyal **E** dikembalikan menjadi '**0**'

Berikut adalah Potongan Program 1 AT89C2051 dengan M1632 :

```
01:     E       bit     P3.4        ; sinyal E   di P3.4
02:     RW      bit     P3.5        ; sinyal R/W di P3.5
03:     RS      bit     P3.7        ; sinyal RS  di P3.7
04:     ;Comment In Here
05:     KirimPerintah:
06:     CLR  RS                     ; RS=0 : register perintah
07:     SJMP  OutByte
08:     ;
09:     KirimASCII:
10:     SETB  RS                    ; RS=1 : Display Data RAM
11:     OutByte:
12:     CLR  RW                     ; RW = '0', kirim data
13:     MOV  P1,A                   ; siapkan data di D0..D7
14:     SETB E                              ; buat pulsa positip
15:     CLR  E                      ; sesaat
16:     ;
17:     TungguDulu:
18:     ACALL AmbilStatus
19:     JB   A.7,TungguDulu
```

```
20:     RET
21:     AmbilStatus:
22:     CLR  RS              ; RS=0 : register status
24:     SJMP  InByte
26:     AmbilASCII:
27:     SETB  RS             ; RS=1 : Display Data RAM
28:     InByte:
29:     SETB  RW             ; RW = '1', ambil data
30:     SETB  E                      ; minta data pada M1632
31:     MOV  A,P1            ; ambil data
32:     CLR  E               ; kembalikan E ke '0'
33:     RET
```

2.7.3. Mengatur tampilan M1632

M1632 mempunyai seperangkat perintah untuk mengatur tata kerjanya, perangkat perintah tersebut meliputi perintah untuk menghapus tampilan, meletakkan kembali cursor pada baris huruf pertama baris pertama, menghidup/matikan tampilan dan lain sebagainya, semua itu dibahas secara terperinci dalam lembar data M1632.

Setelah diberi sumber daya, ada beberapa langkah persiapan yang harus dikerjakan dulu agar M1632 bisa dipakai, langkah-langkah tersebut antara lain adalah:

1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya trasfer data antar M1632 dan mikrokontroler dilakukan dengan mode 8 bit

3. Tunggu selama 4.1 mili-detik
4. Kirimkan sekali lagi perintah 30h
5. Tunggu lagi selama 100 mikro-detik

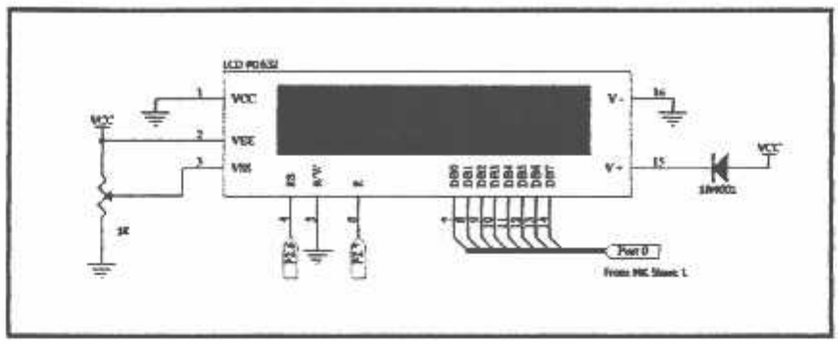
Setelah langkah-langkah tersebut di atas M1632 barulah bisa menerima data dan menampilkannya dengan baik. Pada awalnya tampilan akan nampak kacau, dengan demikian perlu segera dikirim perintah menghapus tampilan dan lain sebagainya, sesuai dengan petunjuk yang ada di Lembar Data.

Di atas dipakai AT89C2051 sebagai contoh, meskipun demikian semua yang dibahas di atas sepenuhnya bisa dipakai pada mikrokontroler MCS 51. Dalam pemakaiannya karena berbagai macam alasan, bisa saja sinyal E; **RW** dan **RS** tidak disimulasikan di **P3.4**; **P3.5** dan **P3.7**. Hal ini bisa diselesaikan dengan melakukan beberapa penyesuaian, yakni tentukan dulu perubahan rangkaian sesuai dengan keadaan yang ada, dan perubahan rangkaian itu harus di sesuaikan di baris 1 sampai 3 pada potongan program di atas.

M1632 mempunyai 8 jalur data dan memerlukan 3 jalur kontrol, dalam suatu rangkaian yang memakai banyak port dari MC-S51, bisa terjadi kekurangan port untuk menghubungkan MCS51 ke M1632. Jika sampai terjadi hal semacam ini bisa ditempuh hal hal berikut :

1. M1632 dipakai dalam mode data 4 bit, yakni hanya memakai jalur data **D0..D3**
2. Dengan sedikit tambahan rangkaian sinyal **WR** dan **RD** diubah menjadi sinyal E dan **R/W** gaya Motorola, sehingga tidak perlu menyediakan port untuk men-simulasikan sinyal-sinyal tersebut. Berikut adalah gambar rangkaian

LCD dengan komponen-komponen pendukung dengan pin-pin yang akan dihubungkan pada mikrokontroler MCS 51



Gambar 2.10. Rangkaian LCD M1632 [7]

LCD M1632 mempunyai spesifikasi sebagai berikut :

1. Memiliki 16 karakter dan dua baris tampilan yang terdiri dari 5 x 7 dot matrik ditambah dengan kursor.
2. Pembangkit karakter ROM untuk 192 jenis karakter.
3. Pembangkit karakter RAM untuk 8 jenis karakter.
4. 80 x 8 display data RAM (max 80 karakter).
5. Isolator didalam modul.
6. Memerlukan catu daya ± 5 volt.
7. Otomatis reset saat catu daya dinyalakan.

Tabel 2-5. Fungsi Pin – Pin LCD [7]

No. PIN	Nama PIN	Fungsi
1	Vss	Terminal Ground
2	Vcc	Tegangan Catu + 5 volt
3	Vee	Mengendalikan kecerahan LCD
4	RS	Sinyal pemilihan register 0 = Tulis 1 = Baca
5	R/W	Sinyal seleksi tulis atau baca 0 = Tulis 1 = Baca
6	E	Sinyal operasi awal yang mengaktif kan data tulis atau baca
7 - 14	DB0 – DB7	Merupakan saluran data berisi perin tah data yang akan ditampilkan
15	V + BL	Back Light Supply 5 Volt (Volt)
16	V - BL	Back Ligth Supply 0 (Ground)

Pada LCD juga terdapat instruksi – instruksi sebagai berikut :

- Display clear : membersihkan tampilan yang ada pada LCD.
- Cursor home : hanya membersihkan tampilan dan kursor kembali ke semula.
- Empty mode Set : layar beraksi sebagai tampilan tulis.

  S : 1/0 = menggeser layar.

  1/0 : 1 = kursor bergerak ke kanan dan layar bergerak ke kiri.

- Display On/Off kontrol.

  D : 1 = layar on

  D : 0 = layar off

  C : 1 = kursor on

  C : 0 = kursor off

  B : 1 = kursor berkedip-kedip

  B : 0 = kursor tidak berkedip – kedip

- Cursor Display Shift

  S/C : 1 = LCD diidentifikasikan sebagai layar

  S/C : 0 = LCD diidentifikasikan sebagai kursor

  R/L : 1 = menggeser satu spasi ke kanan

  R/L : 0 = menggeser satu spasi ke kiri

- Fuction Set

  DL : 1 = panjang data LCD pada 8 bit

  DL : 0 = panjang data LCD pada 4 bit

  Bit upper ditransfer terlebih dahulu kemudian diikuti dengan 4 bit lower.

  N : 1/0 = LCD menggunakan 2 atau 1 baris karakter

  P : 1/0 = LCD menggunakan 5 x 10 dot matrik

- CG RAM address set : menulis alamat RAM ke karakter

- DD RAM address set : menulis alamat RAM ke tampilan

- BF/address set : BF = 1/0, LCD dalam keadaan sibuk atau tidak sibuk.
- Data write to CG RAM or DD RAM : membaca byte dari alamat terakhir RAM yang dipilih.

2.8. Modul TRW 2.4 G

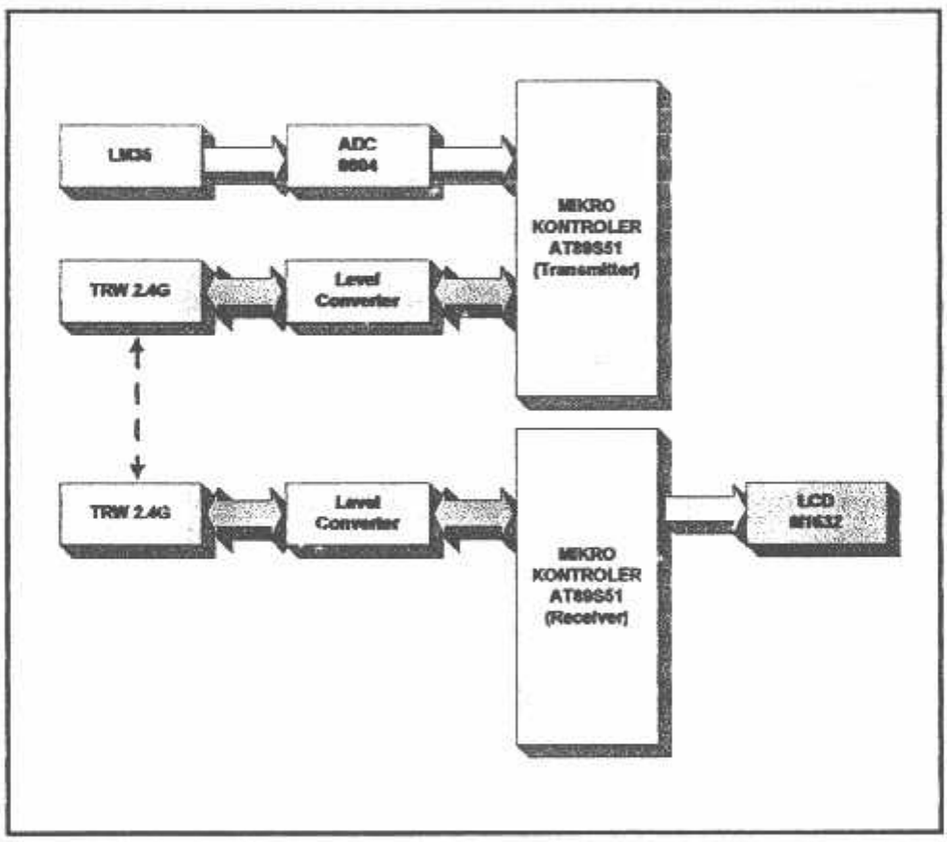
Untuk dapat mengirimkan data serial melalui udara minimal diperlukan suatu device yang dapat melakukan proses penumpangan data serial digital ke frekuensi pembawa dengan frekuensi yang lebih tinggi untuk kemudian dipancarkan ke udara. Salah satu contoh device yang dapat melakukan hal tersebut adalah modul TRW-2.4G *Wireless Transceiver*. Modul tersebut dapat mengirimkan dan menerima data serial melalui media udara, dengan kecepatan 1Mbps atau 250Kbps dengan range frekuensi 2.4-2.524 GHz ISM band. Penggunaan modul tersebut relatif cukup praktis karena dari segi ukuran cukup kecil dan sudah terdapat built-in FIFO buffer data, power amplifier dan antena. Selain ini juga terdapat built-in CRC, sehingga kesalahan data akibat gangguan ketika proses pengiriman dapat diminimalisasi. Modul tersebut bekerja dengan suplay antara 1.9 sampai 3.6VDC. Dalam 1 modul terdapat sebuah pengirim dan dua buah penerima yang dapat digunakan secara simultan. Bentuk fisik dari modul TRW-2.4G adalah seperti pada gambar 2.14. Data serial yang akan dipancarkan melalui RF diumpankan ke modul TRW-2.4G oleh mikrokontroler secara synchronous serial. Begitu pula data yang diterima, akan diambil oleh mikrokontroler secara *synchronous serial*.

BAB III

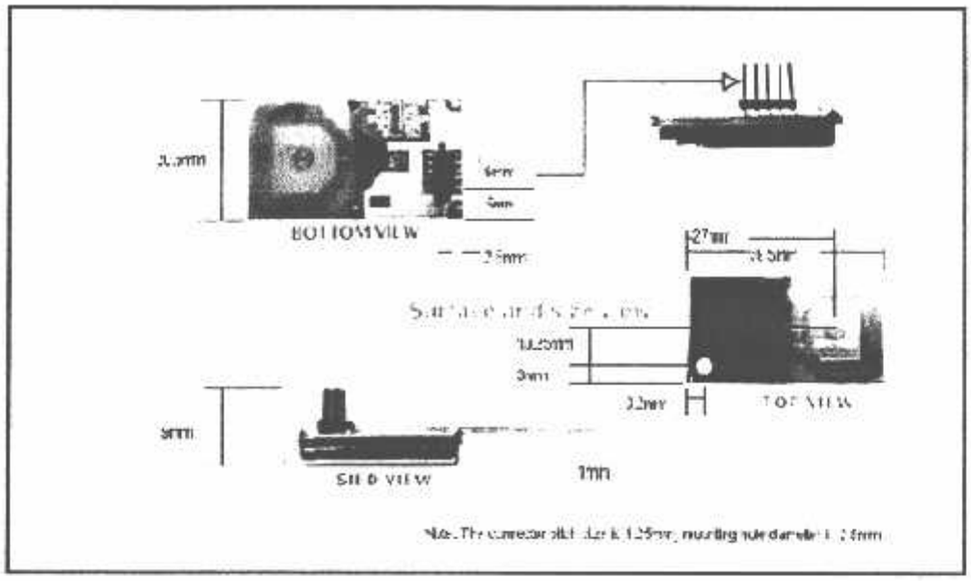
PERENCANAAN DAN PEMBUATAN ALAT

3.1. Diagram Blok

Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver* akan dibuat, digambarkan secara diagram blok sebagai berikut:



Gambar 3.1. Diagram Blok



Gambar 2.14. Modul TRW 2.4G [7]

Pada modul tersebut terdapat dua macam mode proses pengiriman dan penerimaan data serial, yaitu mode Shockburst dan mode *Direct. Mode* Shockburst adalah proses pengiriman data dengan kecepatan tinggi (1Mbps) melalui jalur 2.4GHz band tanpa memerlukan mikrokontroler berkecepatan tinggi untuk proses pengiriman data. Sedangkan *mode Direct* adalah mode pengiriman data secara konvensional, sehingga modul TRW-2.4G bekerja seperti RF device seperti pada umumnya. Pada mode Dirct, data serial yang akan dikirimkan melalui RF haruslah berkecepatan 1Mbps atau 250Kbps, agar penerima dapat mendeteksi sinyalnya, sehingga memerlukan mikrokontroler yang berkecepatan tinggi pula. Agar dapat bekerja dengan mode yang diinginkan, modul TRW-2.4G perlu dikonfigurasikan. Proses konfigurasi ini dapat dilakukan hanya dengan menggunakan 3 kabel interface, dengan kecepatan transfer datanya diatur oleh mokrokontroller, karena menggunakan transmisi data *synchronous serial*.

Rangkaian Transmitter:

- Mikrokontroler AT89S51 berfungsi sebagai Pengendali
- LM35 sebagai sensor suhu.
- ADC 0804 digunakan untuk mengubah besaran analog yang dihasilkan oleh output sensor LM35 menjadi data digital agar nantinya dapat diproses oleh mikrokontroler AT89S51 untuk dikonversi dan ditampilkan ke LCD dalam besaran °C.
- TRW 2-4G, digunakan sebagai transmitter data seting peralatan listrik melalui RF (Radio Frekuensi).
- Level Converter, digunakan agar TRW 2-4G dapat berkomunikasi dengan Mikrokontroler. TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sedangkan mikrokontroler menggunakan catu daya +5 VDC dan memiliki I/O yang bekerja pada level tegangan 5. sehingga diperlukan rangkaian Level Converter.

Rangkaian Receiver:

- Mikrokontroler AT89S51 berfungsi sebagai Pengendali
- TRW 2-4G, digunakan sebagai Receiver data seting peralatan listrik melalui RF (Radio Frekuensi).
- Level Converter, digunakan agar TRW 2-4G dapat berkomunikasi dengan Mikrokontroler. TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sedangkan

mikrokontroler menggunakan catu daya +5 VDC dan memiliki I/O yang bekerja pada level tegangan 5. sehingga diperlukan rangkaian Level Converter.

- LCD M1632 sebagai display suhu ruang dalam besaran °C.

3.2. Prinsip Kerja Alat

Sensor suhu LM35 mengkonversikan suhu sekitar menjadi tegangan analog dengan kenaikan 0,1 mV/°C. Tegangan ini kemudian durubah menjadi data digital oleh ADC 0804. Data ini kemudian diolah oleh mikrokontroler AT89S51 menjadi nilai suhu. Selanjutnya, nilai suhu ini diumpankan ke modul transceiver untuk dipancarkan melalui gelombang radio. Adapun proses yang terjadi pada sisi receiver adalah menerima nilai data suhu tersebut melalui gelombang radio. Data ini diambil untuk ditampilkan pada LCD.

3.3. Mikrokontroller AT89S51

Mikrokontroller AT89S51 dirancangkan untuk dapat berdiri sendiri, karena sudah terdapat EPROM, RAM serta Port I/O *internal*. Untuk berhubungan dengan peralatan luar *chip* dibutuhkan 3 *Bus* yaitu :

- ***Data Bus***

  Yaitu jalur untuk input - output data yang lebarnya sesuai dengan data yang diolah oleh mikrokontroller, yaitu 8 bit.

- ***Address Bus***

  Yaitu jalur input - output atau dari *memori* yang dihubungi, sehingga pada suatu saat hanya ada satu *device* yang berhubungan dengan CPU. Lebar address *bus* mikrokontroller AT89S51 adalah 16 bit (A0 - A15).

- ***Control Bus***

  Berfungsi sebagai pengatur *sinkronisasi* hubungan antara CPU dengan *device* Luar.

3.3.1. Pemetaan *Memori*

Mikrokontroller AT89S51 memiliki 16 bit address (A0 - A15) dengan demikian kapasitas maksimumnya adalah 2^{16} = 65536 *byte* =64 K*byte* dengan alamat 0000H-FFFFH. Jika alamat-alamat program lebih tinggi dari 0FFFH, yang melebihi kapasitas RAM *internal* menyebabkan mikrokontroller secara otomatis mengambil *code byte* dari program *memori* eksternal. *Code byte* juga hanya diambil dari *memori* eksternal dengan alamat 0000H-0FFFH dengan menghubungkan *Pin* EA ke ground.

Dalam perancangan ini hanya menggunakan 4 K*byte*, karena program sudah mencukupi, sehingga EA dihubungkan ke VCC. Mikrokontroller AT89S51 memiliki 4K*byte* *memori internal* yang dapat diprogram dan dihapus sesuai dengan keinginan, dan bersifat *non volatile* (tidak hilang pada saat catu daya terputus).

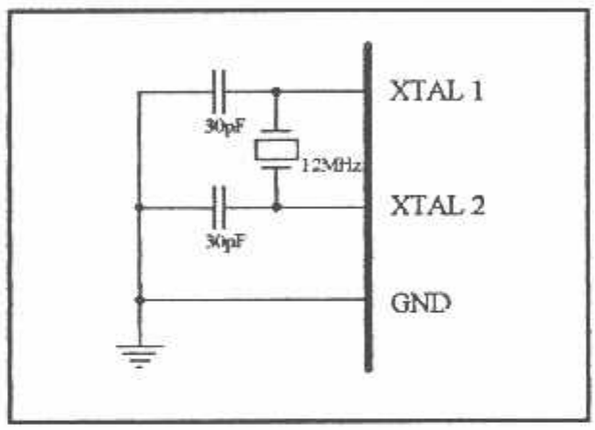
3.3.2. Rangkaian *Clock*

Mikrokontroller AT89S51 ini memiliki *internal clock*, yang berfungsi sebagai sumber *clock*, tapi masih diperlukan rangkaian tambahan untuk membangkitkan *clock* yang diperlukan.

Rangkaian ini terdiri dari 2 buah kapasitor dan sebuah kristal dengan ketentuan :

C1 dan C2 = 20 pF - 40 pF (digunakan untuk kristal)

= 30 pF - 50 pF (digunakan untuk keramik resonator)



Gambar 3.2. Rangkaian *Clock* AT89S51

3.3.3. Rangkaian *Reset*

Rangkaian *reset* bertujuan agar mikrokontroller dapat menjalankan proses mulai dari awal. Rangkaian *reset* untuk mikrokontroller dirancang agar mempunyai kemampuan *power on reset* yaitu *reset* yang terjadi pada saat sistem dinyalakan untuk pertama kalinya. *Reset* juga dapat dilakukan secara manual

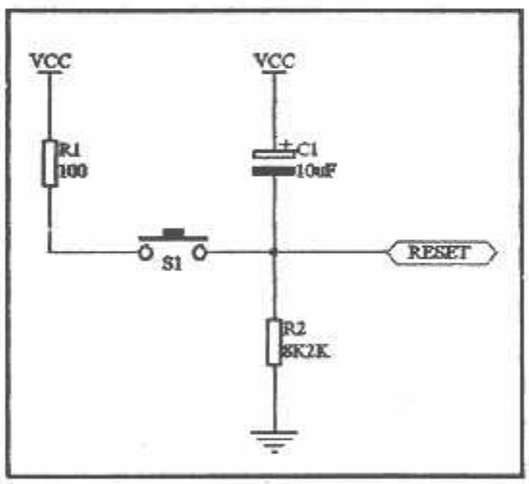
dengan menyediakan tombol yang berupa switch. Jika saklar S1 ditekan , *reset* bekerja secara manual, aliran arus akan mengalir dari VCC melalui R1 menuju kaki RST. Tegangan di RST atau VR2 akan berubah menjadi :

$$VR2 = \frac{R2 \times VCC}{R1 + R2}$$

$$= \frac{8200 \times 5}{100 + 8200}$$

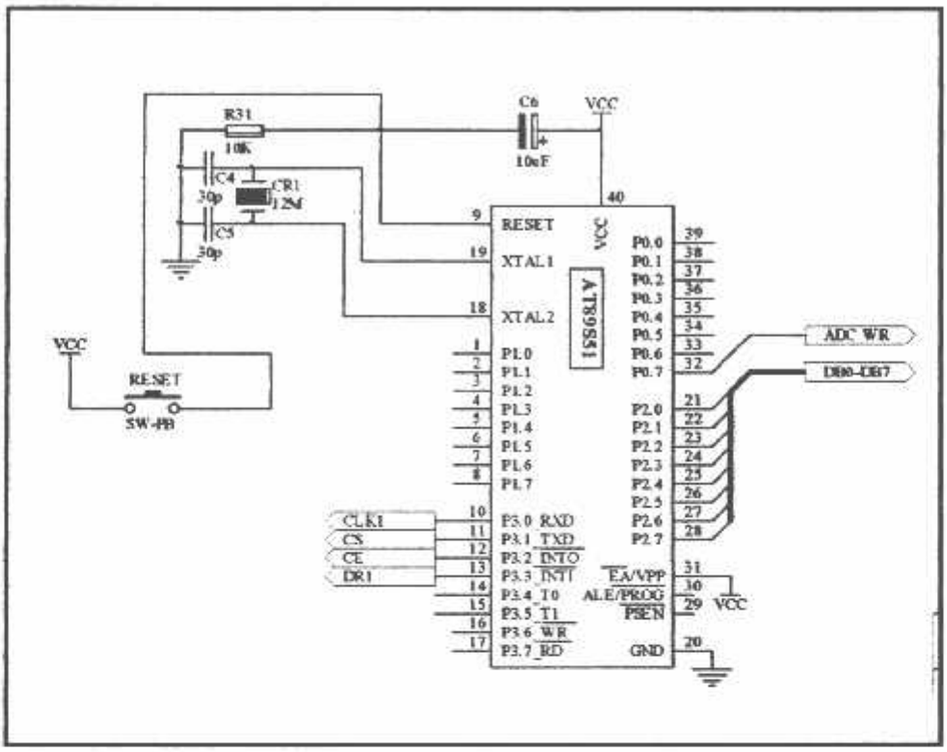
$$= 4{,}94 \text{ volt}$$

Pada tegangan 4,94 volt menyebabkan kaki RST berlogika 1 pada saat saklar ditekan. Saat saklar dilepas, aliran arus dari VCC melalui R1 akan berhenti dan tegangan pada kaki RST akan turun menuju ke nol sehingga logika berubah menjadi nol dan proses *reset* selesai. Rangkaian *reset* ini dapat dilihat seperti pada gambar berikut :



Gambar 3.3. Rangkaian *Reset*

3.3.4. Hubungan *Pin* Pada AT89S51 (pada rangkaian transmiter)

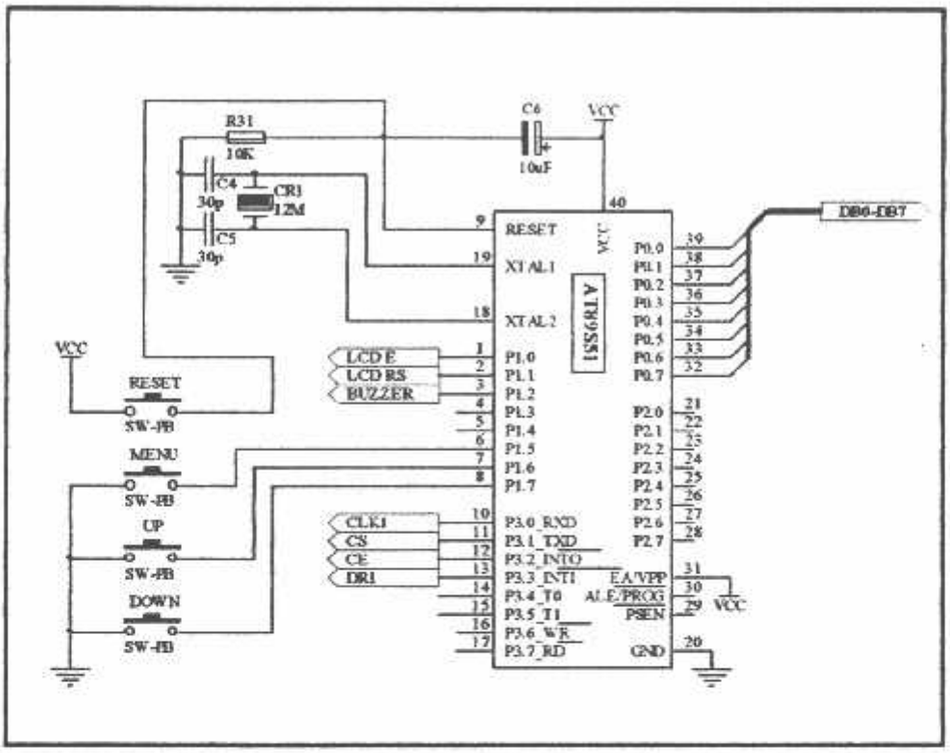


Gambar 3.4. Hubungan *Pin* Pada AT89S51

Diskripsi konfigurasi *Pin-Pin* AT89S51

- *Pin* 9 : dihubungkan Push-button Reset
- *Pin* 10-14 : dihubungkan TRW 24-G
- *Pin* 16 : dihubungkan ADC WR
- *Pin* 18-19 : dihubungkan X-tal 12M
- *Pin* 20 : dihubungkan ke ground
- *Pin* 21-28 : dihubungkan ke ADC DB0-DB7
- *Pin* 32-39 : dihubungkan ke LCD
- *Pin* 31 dan 40 : dihubungkan ke VCC

3.3.5. Hubungan *Pin* Pada AT89S51 (pada rangkaian receiver)



Gambar 3.5. Hubungan *Pin* Pada AT89S51

Diskripsi konfigurasi *Pin-Pin* AT89S51

- *Pin* 1dan 2 : dihubungkan ke LCD (E dan RS)
- Pin 5 : dihubungkan ke Buzzer
- *Pin* 6 : dihubungkan ke Push-button Menu
- *Pin* 7 : dihubungkan ke Push-button Up
- *Pin* 8 : dihubungkan Push-button Down
- *Pin* 9 : dihubungkan Push-button Reset
- *Pin* 10-14 : dihubungkan TRW 24-G

- *Pin* 18-19 : dihubungkan X-tal 12M
- *Pin* 20 : dihubungkan ke ground
- *Pin* 32-39 : dihubungkan ke LCD D0-D7
- *Pin* 31 dan 40 : dihubungkan ke VCC

3.4. Perancangan Rangkaian TRW 24-G

TRW-2.4G merupakan modul RF dengan modulasi GFSK dan memiliki 2 mode transmisi data yaitu direct dan shock burst, Dalam aplikasi ini, TRW-2.4G digunakan dalam mode shock burst agar dapat menggunakan mikrokontoler yang murah yaitu AT89S51.

Aplikasi ini membutuhkan modul / komponen berikut:

- 2 bh AT89S51
- 2 bh TRW-2.4G,
- 2 bh 74HC125,
- 4 bh transistor 2N3904,
- 6 bh resistor 10K ohm ¼ W.

TRW-2.4G memiliki spesifikasi berikut:

- Frekuensi kerja : 2,4 – 2,524 GHz ISM band.
- Tipe modulasi : GFSK.
- Catu daya : 1,9 – 3,6 V.
- RF data rate : 1 Mbps atau 250 Kbps.
- Multi channel, hingga 125 channel.
- Dual receiver.

• Jangkauan : 280m @ 250Kbps, 150m @ 1Mbps.

• Built-in antenna.

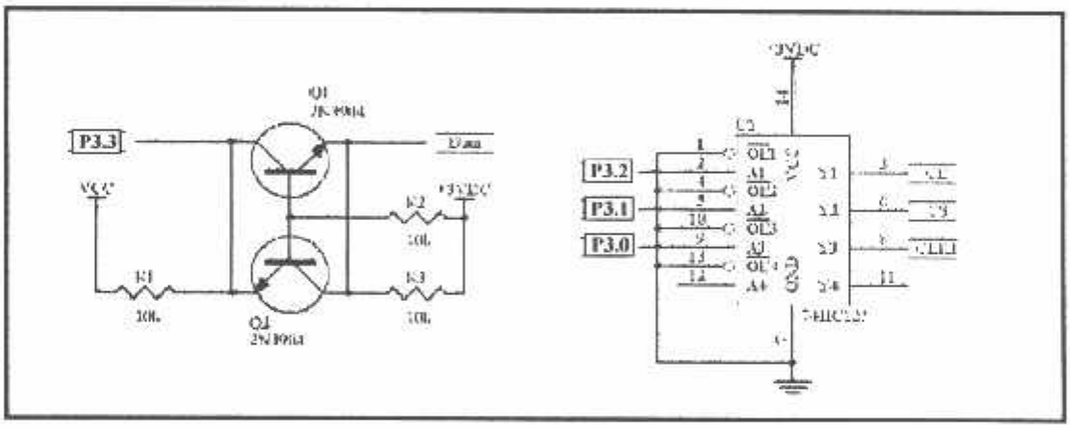
Hubungan antara modul-modul tersebut dengan mikrokontroler AT89S51 adalah sebagai berikut:

CLK1, dihubungkan P3.0 melalui rangkaian level converter

CS, dihubungkan P3.1 melalui rangkaian level converter

CE, dihubungkan P3.2melalui rangkaian level converter

TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sehingga beberapa hubungan diatas harus melalui rangkaian level converter seperti pada Gambar berikut.



Gambar 3.6. Rangkaian Level Converter

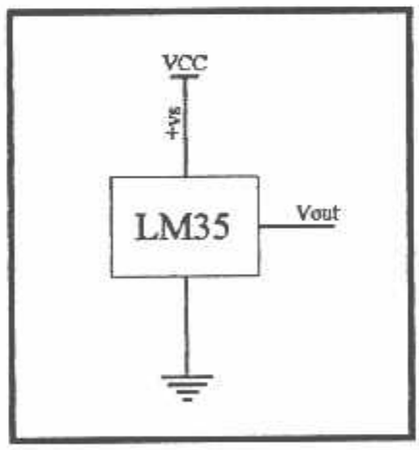
3.5. Rangkaian Sensor

Sensor temperatur yang digunakan berupa IC LM 35 yang merupakan sensor semikonduktor keluaran *National Semiconductor*. Sensor ini merupakan IC sensor temperatur yang presisi, dimana tegangan keluarannya berbanding linier dengan temperatur dalam satuan derajat Celsius. LM35 memiliki kemampuan

untuk menghasilkan tegangan keluaran sebesar 10 mV untuk setiap perubahan temperatur 1 °C. Sebagai contoh, jika suhu yang diterima sebesar 25 °C maka tegangan keluaran dari LM35 adalah sebesar 250mV.

LM35 tidak membutuhkan kalibrasi eksternal atau pengaturan untuk menghasikan akurasi yang tinggi. LM35 memiliki impedansi output yang rendah, kurang dari 0,1 Ω untuk beban operasi dari 400uA-5mA, output yang linier dan kalibrasi awal yang presisi yang membuat interfacing membaca atau pengaturannya menjadi mudah.

Sensor ini memiliki tiga terminal, positif, negatif dan v_{out} yang beroperasi layaknya sebagai sebuah dioda zener. Untuk mengoperasikannya terminal positif yang juga berfungsi sebagai terminal keluaran dibias dengan arus yang besarnya antara 400uA sampai 5mA. Tegangan *brekdown* pada terminal positif / keluaran berbanding lurus dengan temperatur lingkungan. Rangkaian sensor suhu yang digunakan adalah seperti yang terlihat pada gambar berikut ini :



Gambar 3.7. Rangkaian Sensor LM35

3.6. Pengkondisian Sinyal

Nurse Call ini dibatasi range pengukurannya pada temperatur 0 sampai 100°C. Tingkat ketelitian pengukuran temperatur yang diinginkan sebesar 0,1 °C, sehingga nantinya dapat diamati perubahan nilai suhu tubuh pada tiap perubahan temperatur.

3.6.1. *Non Inverting Amplifier (penguat tak membalik).*

Digunakan untuk mengubah range keluaran dari sensor temperatur. Sensor temperatur yang digunakan menghasilkan tegangan keluaran yang setara dengan temperatur lingkungan dalam satuan derajat Celsius. Tiap satu derajat dinyatakan dalam tegangan keluaran 10mV.

Untuk memudahkan pengambilan data temperatur, maka tegangan keluaran ini harus disesuaikan dengan resolusi ADC yang digunakan. Karena ADC yang digunakan merupakan ADC 8 bit dengan tegangan referensi sebesar 2,5 volt, maka didapatkan resolusi ADC sebesar:

$$1LSB = \frac{2.5volt}{255} = 9.803mV$$

Sehingga untuk mendapatkan perubahan 1 bit LSB pada ADC diperlukan perubahan 9.803mV pada bagian masukannya.

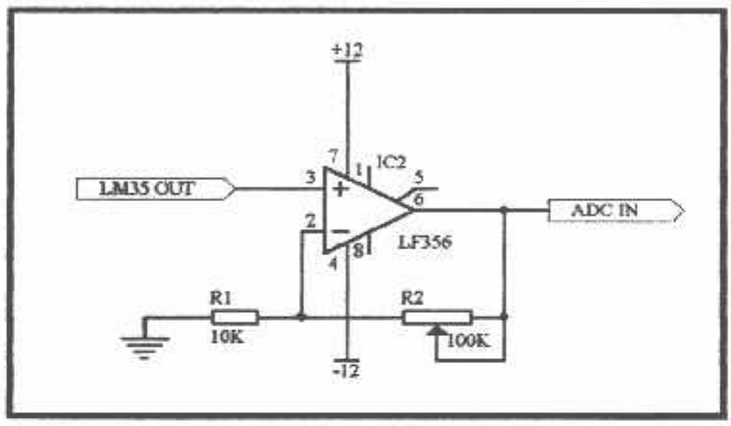
1 bit LSB ini digunakan untuk mendapatkan perubahan 1 °C, maka keluaran sensor suhu (LM 35)harus dikuatkan sebesar :

$$Av = \frac{9,803mV}{10mV} = 0.98$$

Penguat inilah yang diimplementasikan menggunakan rangkaian *non-inverting amplifier* . Penguatan yang dihasilkan ditentukan oleh nilai R1 dan R2, yaitu :

$$Av = 1 + \frac{R2}{R1}$$

Dengan mengatur nilai R2 maka akan didapatkan penguatan sebesar 0.98 kali.



Gambar 3.8. Penguat tegangan tak membalik.

3.7. Perancangan *Analog To Digital Converter* (ADC) 0804

Untuk mengubah data analog menjadi data digital dari rangkaian sensor suhu LM35, diperlukan rangkaian pengubah, yaitu mengubah dari besaran analog menjadi besaran digital dengan menggunakan ADC 0804. Untuk dikonversikan menjadi kombinasi bilangan-bilangan digital dengan tujuan untuk di-*interface*-kan ke mikrokontroler AT89S51, sebagai data biner yang dapat diolah. Agar dapat melakukan pengolahan sinyal, ADC memerlukan pengaturan tegangan referensi yaitu sebesar 2,5 volt. Jadi jika tegangannya mencapai 2,5 volt maka output

maksimumnya adalah FF heksadesimal dan naik turunnya kombinasi digital itu sesuai dengan tegangan yang masuk ke ADC, karena data biner yang diolah mempunyai lebar 8 bit (2^8).

Rangkaian ADC memerlukan pewaktu (*clock*) untuk mengatur kerjanya. Dalam hal ini untuk frekuensi *clock* ADC dapat dihitung sebesar :

Diketahui : R : 10 Kohm

C : 150 pF

$$\text{Frekuensi } clock = \frac{1}{1{,}1\ (R.C)}$$

$$= \frac{1}{1{,}1\ (10.10^{3}.150.10^{-12})}$$

$$= 606{,}06 \text{ KHz}$$

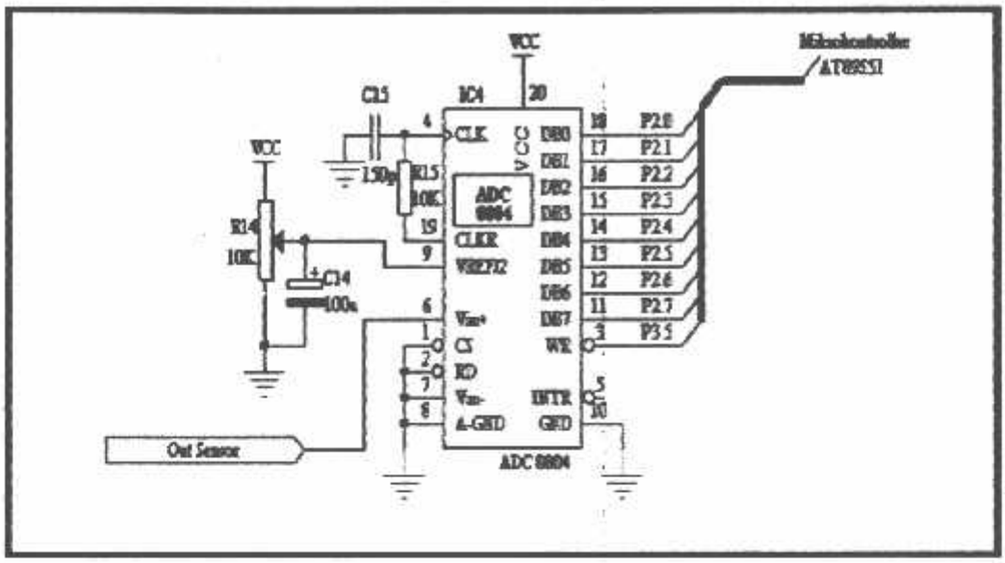
Dengan frekuensi *clock* sebesar 606 KHz, maka waktu konversi yang diperlukan *internal* ADC adalah :

frekuensi *clock* = 606 KHz

$$\text{Maka periode } clock = \frac{1}{F}$$

$$= \frac{1}{606{,}06 \text{ KHz}}$$

$$= 1{,}65 \text{ uS}$$



Gambar 3.9. Rangkaian ADC 0804

Diskripsi *Pin-Pin* ADC 0804 adalah sebagai berikut:

- *Pin* 1,2,7,8,10 : terhubung dengan ground
- *Pin* 6 : terhubung dengan out rangkaian sensor Vout 4051
- *Pin* 11-18 : terhubung dengan AT89S51 P2.0-P2.7
- *Pin* 3 : terhubung dengan AT89S51 P3.5
- *Pin* 20 : terhubung dengan Vcc

3.8. Perancangan Rangkaian LCD

LCD Display Module M1632 buatan Seiko Instrument Inc. terdiri dari dua bagian, yang pertama merupakan panel LCD sebagai media penampil informasi dalam bentuk huruf/angka dua baris, masing-masing baris bisa menampung 16 huruf/angka.

Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibalik pada panel LCD, berfungsi mengatur tampilan informasi serta berfungsi mengatur komunikasi L1632 dengan mikrokontroler yang memakai tampilan LCD itu. Dengan demikian pemakaian M1632 menjadi sederhana, sistem lain yang M1632 cukup mengirimkan kode-kode ASCII dari informasi yang ditampilkan seperti layaknya memakai sebuah printer.

. Untuk berhubungan dengan mikrokontroler pemakai, M1632 dilengkapi dengan 8 jalur data (**DB0..DB7**) yang dipakai untuk menyalurkan kode ASCII maupun perintah pengatur kerjanya M1632. Selain itu dilengkapi pula dengan **E**, **R/W** dan **RS** seperti layaknya komponen yang kompatibel dengan mikroprosesor.

Kombinasi lainya **E** dan **R/W** merupakan sinyal standar pada komponen buatan Motorola. Sebaliknya sinyal-sinyal dari MCS51 merupakan sinyal khas Intel dengan kombinasi sinyal **WR** dan **RD.**

RS, singkatan dari Register Select, dipakai untuk membedakan jenis data yang dikirim ke M1632, kalau **RS=0** data yang dikirim adalah perintah untuk mengatur kerja M1632, sebaliknya kalau **RS=1** data yang dikirim adalah kode ASCII yang ditampilkan.

Demikian pula saat pengambilan data, saat **RS=0** data yang diambil dari M1632 merupakan data status yang mewakili aktivitas M1632, dan saat **RS=1** maka data yang diambil merupakan kode ASCII dari data yang ditampilkan.

Proses mengirim/mengambil data ke/dari M1632 bisa dijabarkan sebagai berikut :

1. **RS** harus dipersiapkan dulu, untuk menentukan jenis data seperti yang telah dibicarakan di atas.
2. **R/W** di-nol-kan untuk menandakan akan diadakan pengiriman data ke M1632. Data yang akan dikirim disiapkan di **DB0..DB7**, sesaat kemudian sinyal **E** di-satu-kan dan di-nol-kan kembali. Sinyal **E** merupakan sinyal sinkronisasi, saat **E** berubah dari 1 menjadi 0 data di **DB0 .. DB7** diterima oleh M1632.
3. Untuk mengambil data dari M1632 sinyal **R/W** di-satu-kan, menyusul sinyal **E** di-satu-kan. Pada saatu **E** menjadi 1, M1632 akan meletakkan datanya di **DB0 .. DB7**, data ini harus diambil sebelum sinyal **E** di-nol-kan kembali.

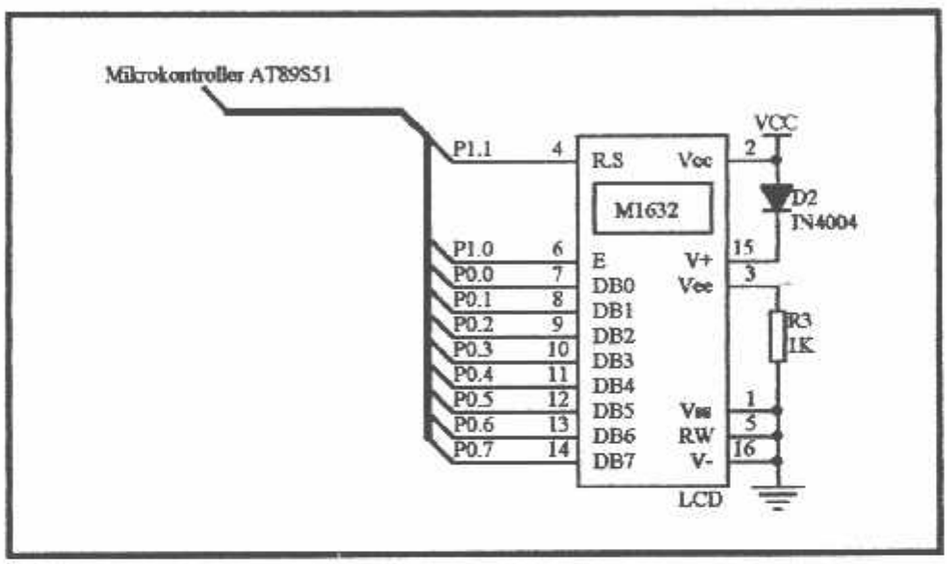
M1632 mempunyai seperangkat perintah untuk mengatur tata kerjanya, perangkat perintah tersebut meliputi perintah untuk menghapus tampilan, meletakkan kembali cursor pada baris huruf pertama baris pertama, menghidup/matikan tampilan dan lan sebagainya, semua itu dibahas secara terperinci dalam lembar data M1632.

Setelah diberi sumber daya, ada beberapa langkah persiapan yang harus dikerjakan dulu agar M1632 bisa dipakai, langkah-langkah tersebut antara lain adalah:

1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya trasfer data antar M1632 dan mikrokontroler dilakukan dengan mode 8 bit
3. Tunggu selama 4.1 mili-detik

4. Kirimkan sekali lagi perintah 30h
5. Tunggu lagi selama 100 mikro-detik

Setelah langkah-langkah tersebut di atas M1632 barulah bisa menerima data dan menampilkannya dengan baik. Pada awalnya tampilan akan nampak kacau, dengan demikian perlu segera dikirim perindah menghapus tampilan dan lain sebagainya, sesuai dengan petunjuk yang ada di lembar data.



Gambar 3.10. Rangkaian LCD Pada *Mikrokontroller*

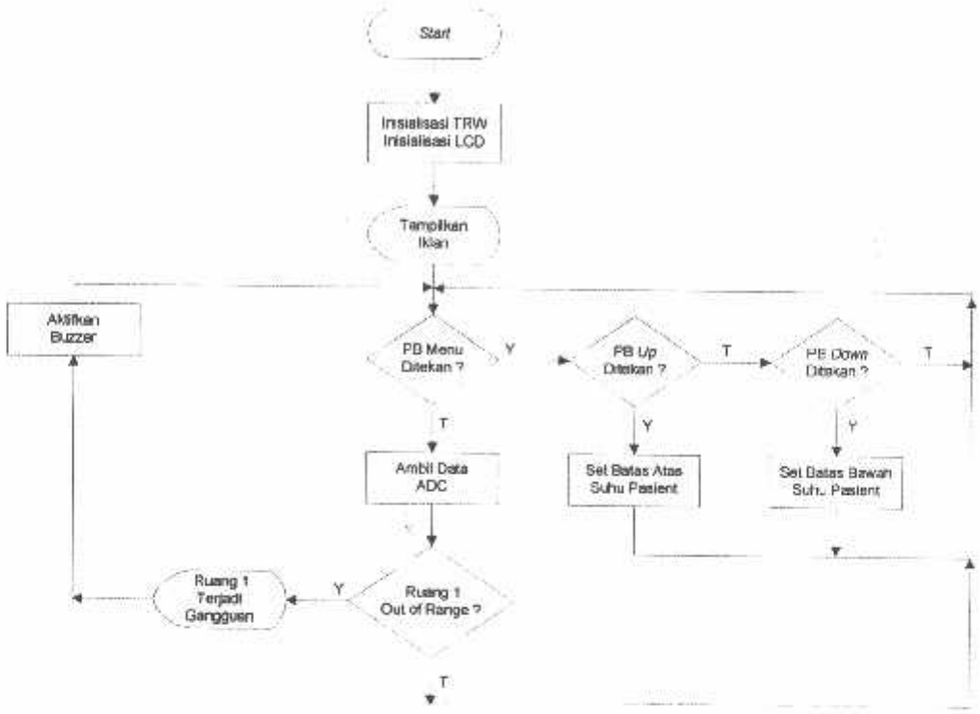
Diskripsi konfigurasi *Pin-Pin* pada LCD:

- *Pin* 1,3,5 dan 16 : dihubungkan dengan ground
- *Pin* 2 : dihubungkan dengan VCC
- *Pin* 4 : dihubungkan dengan P1.0 AT89S51
- *Pin* 6 : dihubungkan dengan P1.1 AT89S51
- *Pin* 7-14 : dihubungkan dengan AT89S51 (P0.0-P0.7)

Untuk tampilan dipergunakan LCD Dot Matrik 2 x 16 karakter. Sinyal-sinyal yang diperlukan oleh LCD adalah RS dan Enable, sinyal RS dan Enable dipergunakan sebagai input yang outputnya dipakai untuk mengaktifkan LCD. LCD akan aktif apabila mikrokontroller memberikan instruksi tulis pada LCD. Saat kondisi RS don't care dan Enable 0 maka LCD tetap pada kondisi semula, pengiriman data ke LCD dilakukan saat RS berlogika 0 dan enable berlogika 1. Instruksi dikirim pada LCD bila keadaan RS 1 dan Enable 1. Pin LCD ini untuk data terkoneksi pada *Port 0* mikrokontoler AT89S51. Kemudian untuk RS dihubungkan pada *Port* 1.1, tulis/baca (*Read/Write)* diberikan logika *low* karena disini LCD bersifat menulis data, dan yang terakhir *Enable* (E) dikendalikan dengan *Port 1.0*.

3.9. Perangkat Lunak

Perencanaan perangkat lunak sangat diperlukan untuk menjalankan sistem seperti yang diinginkan. Mikrokontroller AT89S51 tidak akan bisa dijalankan tanpa adanya *software*. Didalam perencanaan perangkat lunak ini di buat perintah-perintah yang berfungsi untuk kerja alat.



Gambar 3.11 Flowchart Program alat

BAB IV

PENGUJIAN ALAT

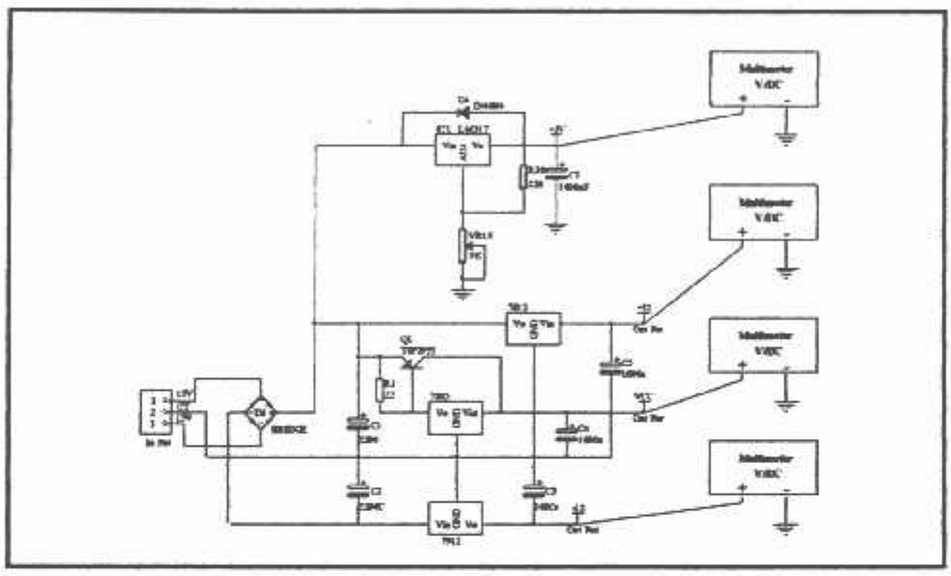
Setelah melakukan perancangan atau pembuatan alat ini maka kita perlu melakukan suatu pengujian sistem. Yang mana pengujian sistem ini bertujuan antara lain :

1. Mengetahui sejauh mana, Sistem Monitoring Suhu Ruangan Tanpa Kabel *(Wireless)* Menggunakan TRW 2.4G *Transmitter/Receiver* ini dapat berfungsi sebagaimana yang kita harapkan.
2. Mencari dan menemukan berbagai kendala yang mungkin timbul pada saat Nurse call dengan deteksi suhu tubuh pasient beroperasi untuk kemudian diperbaiki sampai pada tingkat kesalahan sistem yang sekecil mungkin sehingga didapatkan hasil yang sebaik-baiknya.
3. Memahami spesifikasi dari rangkaian kontrol dan juga rangkaian *sensor* dengan cara mengukur dan membandingkan *Vout sensor* terhadap besaran suhu °C.

4.1. Pengujian Rangkaian *Power Supply*

Pengujian ini bertujuan untuk mengetahui tegangan yang dikeluarkan oleh rangkaian catu daya yang telah dibuat. Dengan begitu dapat diketahui apakah terjadi kesalahan terhadap rangkaian atau tidak. Tegangan yang dibutuhkan untuk memberikan tegangan kerja keseluruhan rangkaian adalah +3V, +5V, +12V, dan -12V.

Untuk mengukur besarnya tegangan keluaran pada rangkaian *power supply* maka pada pengukuran ini menggunakan Multimeter Digital. Berikut cara pengukuran untuk pengambilan data besarnya tegangan pada rangkaian *power supply* yang dilakukan satu persatu pada *output regulator* tegangan LM7805, LM7812 dan LM7912, dan LM317.



Gambar 4.1. Cara pengukuran tegangan pada rangkaian catu daya

Untuk mengukur besarnya tegangan keluaran rangkaian *power supply* maka tegangan masukan pada rangkaian *power supply* juga harus tersambung dengan sumber tegangannya yaitu tegangan jala-jala dari PLN 220 V/AC. Multimeter digital di-seting pada *DC Volt* dengan batas maksimal pengukuran yaitu 20 *Volt DC*. Pengukuran tegangan dilakukan pada *output* keluaran dari *regulator* tegangan LM7805, LM7812, LM7912 dan LM317.

. Dari hasil pengukuran pada rangkaian *power supply* diperoleh data seperti pada tabel berikut ini.

Tabel 4-1. Hasil Pengukuran Rangkaian *Power Supply*

***Regulator* tegangan**	***Vout* (V)**
LM7805	+4,9
LM7812	+11,9
LM7912	-11,9
LM317	3,3V

Dari hasil pengujian yang diperoleh dapat diketahui bahwa *power supply* ini dapat memberikan tegangan kerja untuk peralatan kontrol maupun peralatan *sensor* sesuai dengan tegangan kerja yang dibutuhkan. Hasil yang di peroleh dari pengukuran tersebut, dikarenakan oleh adanya beberapa faktor yang mempengaruhi, diantaranya kualitas dari tiap-tiap komponen yang digunakan nilainya tidak murni. Selain itu, tegangan jala-jala listrik yang digunakan tidak stabil.

4.2. Pengujian Rangkaian TRW 2.4G

RW-2.4G merupakan modul RF dengan modulasi GFSK dan memiliki 2 mode transmisi data yaitu direct dan shock burst, menurut vendor-nya daya jangkau modul ini mencapai 280 meter pada kondisi open area. Dalam aplikasi ini, TRW-2.4G digunakan dalam mode *shock burst* agar dapat menggunakan mikrokontoler yang murah yaitu mikrokontroler AT89S51.

Pengujian ini membutuhkan modul / komponen berikut:

- 2 bh mikrokontroler AT89S51
- 2 bh TRW-2.4G,
- 2 bh 74HC125,
- 4 bh transistor 2N3904,
- 6 bh resistor 10K ohm ¼ W.
-1 bh LCD M1632

TRW-2.4G memiliki spesifikasi berikut:

• Frekuensi kerja : 2,4 – 2,524 GHz ISM band.

• Tipe modulasi : GFSK.

• Catu daya : 1,9 – 3,6 V.

• RF data rate : 1 Mbps atau 250 Kbps.

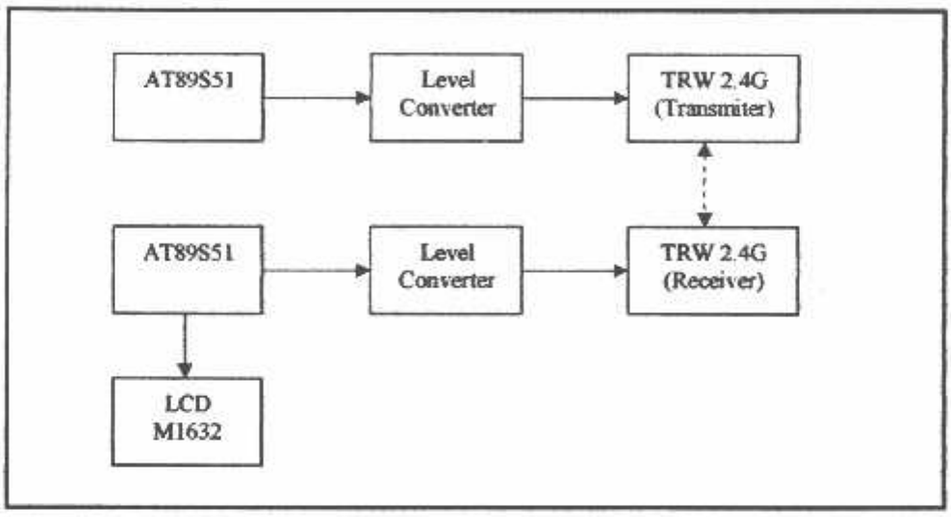
• Multi channel, hingga 125 channel.

• Dual receiver.

• Jangkauan : 280m @ 250Kbps, 150m @ 1Mbps.

• Built-in antenna.

Pengujian ini dilakukan dengan merangkai rangkaian seperti pada gambar diagaram blok dibawah ini. Dan kemudian mengisikan program cek trw yang telah penulis sertakan dalam lampiran. Apabila kondisi TRW masih baik maka pada rangkaian receiver akan menampilkan huruf 'A' pada display LCD baris pertama kolom pertama.



Gambar 4.2. Diagram Blok Pengujian TRW 2.4G

4.3. Pengujian Rangkaian Sensor

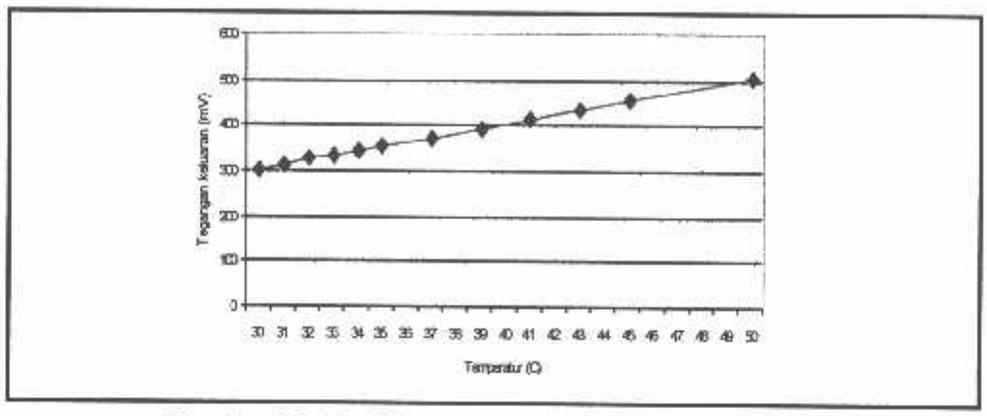
Untuk mengetahui ketelitian sensor suhu yang digunakan dalam perencanaan alat ini, maka sebelumnya data suhu hasil pengolahan dari sensor temperatur pada temperatur kamar dibandingkan dengan pembacaan termometer alkohol pada suhu dan saat yang bersamaan. Sensor ini memiliki linearitas yang baik terhadap perubahan temperatur.

Rangkaian pengkondisi sinyal LF356 pada modul digunakan sebagai *buffer* keluaran IC sensor LM35 agar tidak sampai terjadi drop tegangan. Hal ini dimaksudkan untuk mendapatkan data keluaran ADC yang sama dengan nilai temperatur saat itu.

Pengujian dilakukan dengan jalan memasukkan sensor LM35 yang telah dilindungi dengan karet *heat srink* kedalam air bersamaan dengan termometer alkohol. Air tersebut kemudian dipananaskan dengan bantuan *water heater* sampai temperaturnya mencapai 31 derajat *Celsius*.

Tabel 4-2. Hasil Pengukuran *Vout* Sensor

SUHU (^{0} C)	V_{OUT} (mVolt)
25	249
26	261
27	271
28	281
29	289
30	302
31	312



Gambar 4.3. Grafik pengambilan data temperatur

4.3.1. Analisa Data Hasil Perhitungan Rangkaian Sensor Suhu

Untuk menghitung tegangan *output* dari LM 35 menggunakan rumus :

$$\mathbf{V_{out} = suhu\ (°C)\ x\ 10\ mV\ /\ °C}$$

- Saat T = 25 ° C

$$V_{out} = 25°C \times 10mV/°C$$

$$= 250\ mV$$

- Saat T = 26 ° C

$$V_{out} = 26°C \times 10mV/°C$$

$$= 260\ mV$$

- Saat T = 27 ° C

$$V_{out} = 27°C \times 10mV/°C$$

$$= 270\ mV$$

- Saat T = 28 ° C

$$V_{out} = 28°C \times 10mV/°C$$

$$= 280\ mV$$

- Saat T = 29 ° C

$$V_{out} = 29°C \times 10mV/°C$$

$$= 290 \text{ mV}$$

- Saat T = 30 ° C

$$V_{out} = 30°C \times 10mV/°C$$

$$= 300 \text{ mV}$$

- Saat T = 31 ° C

$$V_{out} = 31°C \times 10mV/°C$$

$$= 310 \text{ mV}$$

Tabel 4-3
Perbandingan Hasil Pengukuran dan Perhitungan V_{out}

SUHU (^{0} C)	TEGANGAN KELUARAN / V_{out}	
	Pengukuran (mV)	Perhitungan (mV)
25	249	250
26	261	260
27	271	270
28	281	280
29	289	290
30	302	300
31	312	310

4.3.2. Kesalahan Relatif

Kesalahan rata-rata akan dijadikan tendensi pada setiap hasil pengukuran yang dilakukan. Oleh sebab itu untuk mencari kesalahan relatif tiap data yang diambil maka diambil rumus:

$$Kr = \frac{X - \bar{X}}{X} \times 100\%$$

Dimana:

Kr = Kesalahan Relatif

x = Hasil Rata-Rata Perhitungan

$\bar{x}$ = Hasil Rata-Rata Seharusnya

$$x = \frac{250 + 260 + 270 + 280 + 290 + 300 + 310}{7}$$

$$x = 280$$

$$\bar{x} = \frac{249 + 261 + 271 + 281 + 289 + 302 + 312}{7}$$

$$\bar{x} = 280{,}7$$

Kesalahan relatif pada rangkaian pengontrol suhu ini sebesar:

$$\textit{Kesalahan relatif} = \frac{\textit{hasil perhitungan} - \textit{hasil yang seharusnya}}{\textit{Hasil perhitungan}} \cdot 100\%$$

$$= \left| \frac{280 - 280{,}7}{280} \right| \cdot 100\%$$

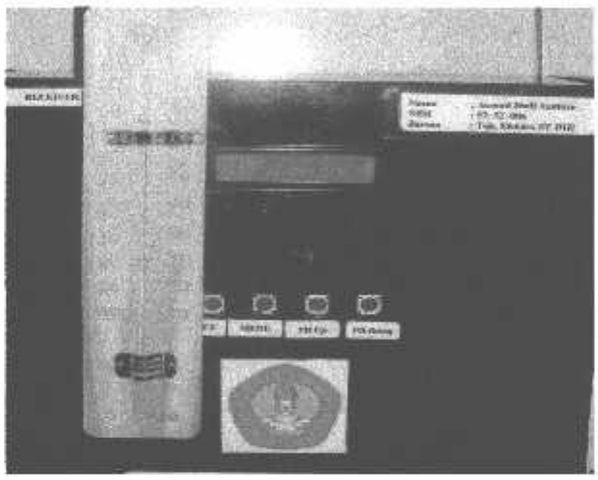
$$= 0{,}25\ \%$$

Jadi untuk rangkaian pengontrol suhu memiliki tingkat akurasi sebesar:

Tingkat akurasi = 100 % - 0,25 %

= 99,75%.

4.3.3 Hasil perhitungan Validasi sensor suhu dengan alat lain



Gambar 01

Foto hasil percobaan Validasi dengan alat lain

Dari hasil kalibrasi alat ukur suhu terdapat selisih temperatur sebesar 1.5 °C maka trdapat perbedaan arus keluar sebesar 1.5 mA denagan asumsi 1mA = 1 °C.

Oleh sebab itu untuk mencari kesalahan relatif atau validasi data yang diambil maka diambil rumus:

$$Kr = \frac{X - \overline{X}}{X} \times 100\%$$

Dimana:

Kr = Kesalahan Relatif

x = Hasil Rata-Rata Perhitungan (Termometer)

$\overline{x}$ = Hasil Rata-Rata Perhitungan (Pada Alat)

Di ket :

$x = 28.5$

$V_{out} = 28{,}5°C \times 10mV / °C$

$= 280{,}5$ mV

$\overline{x} = 27$

$V_{out} = 27°C \times 10mV / °C$

$= 270$ mV

$$x = \frac{280.5}{7}$$

$$x = 40$$

$$\bar{x} = \frac{270}{7}$$

$$\bar{x} = 38.5$$

Untuk mengetahui Kesalahan relatif pada rangkaian pengontrol suhu ini diambil rumus:

$$\textit{Kesalahan relatif} = \frac{\textit{hasil perhitungan} - \textit{hasil yang seharusnya}}{\textit{Hasil perhitungan}} \cdot 100\%$$

$$= \left|\frac{40 - 38.5}{40}\right| \cdot 100\%$$

$= 3.75\ \%$

Jadi untuk rangkaian pengontrol suhu memiliki tingkat akurasi sebesar:

Tingkat akurasi $= 100\ \% - 3{,}75\ \%$

$= 96{,}25\%.$

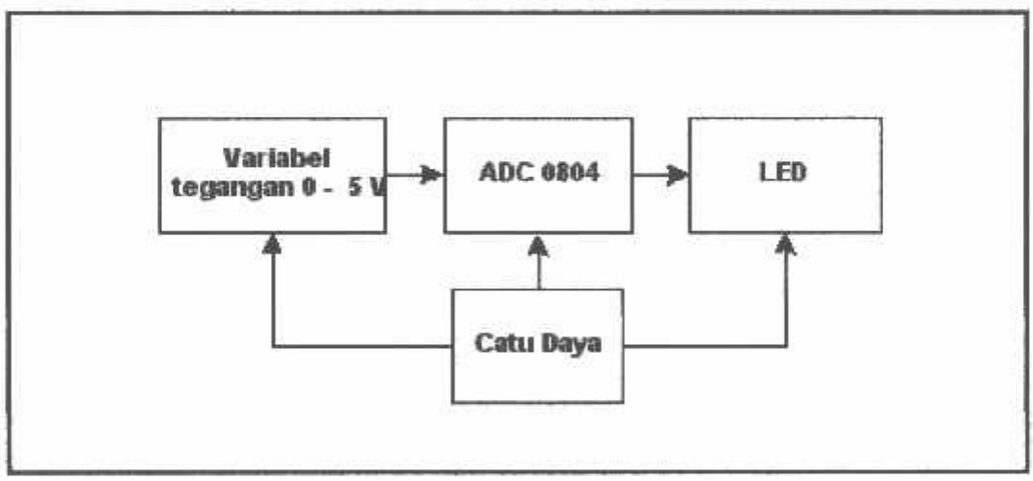
4.4. Pengujian Rangkaian ADC 0804

4.4.1. Tujuan.

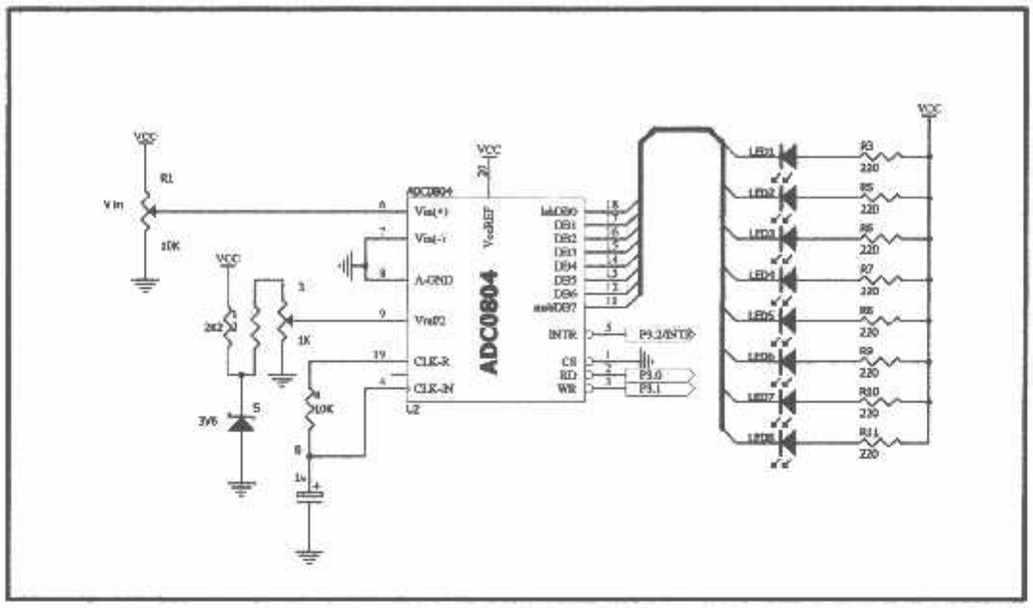
Untuk mengetahui hasil konversi tegangan analog ke digital setiap kenaikan 1 digit hasil konversi.

4.4.2. Peralatan Yang Digunakan.

1. Variabel Resistor 10 KΩ.
2. Digital Multimeter.
3. Catu Daya 5 Volt.
4. Rangkaian ADC 0804.
5. Delapan bit rangkaian indikator LED.



Gambar 4.4. Diagram Blok Pengujian ADC 0804



Gambar 4.5. Rangkaian Pengujian ADC 0804

4.4.3. Langkah Pengujian

1. Memasang 8 buah LED dengan resistor 220Ω pada keluaran ADC 0804.
2. Memberikan Catu Daya 5 Volt pada rangkaian ADC 0804.
3. Memberikan masukan analog pada tegangan 0 s.d. 5 Volt pada pin masukan ADC 0804.
4. Mengatur variabel resistor untuk mendapatkan tegangan yang diinginkan.
5. Mencatat data biner 8 bit pada keluaran ADC 0804.

4.4.4. Hasil Pengujian

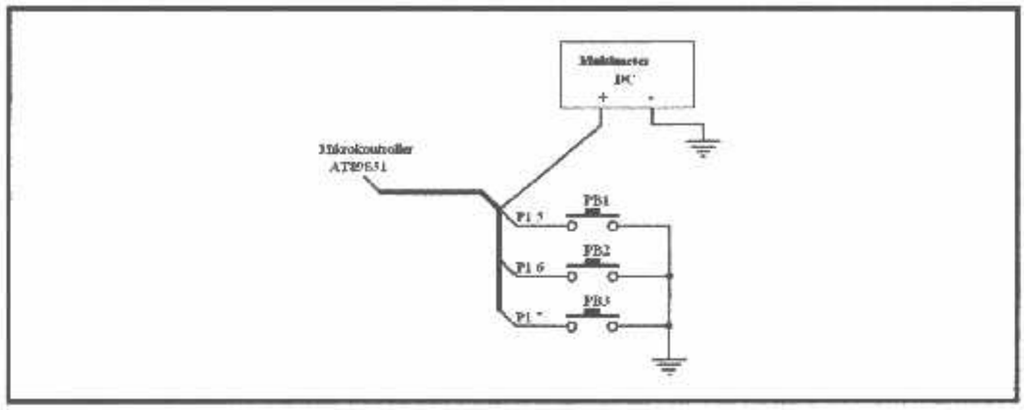
Tabel 4-4. Hasil Pengujian ADC 0804 dengan LED indikator.

Vin (Volt)	*Output*							
	D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	1	1
2	0	1	1	0	1	1	0	1
3	1	0	0	1	1	0	0	1
4	1	1	0	1	1	0	1	0
5	1	1	1	1	1	1	1	1

4.5. Pengujian Terhadap Rangkaian *Push-button*

Rangkaian *push-button* berfungsi sebagai inputan mikrokontroler untuk men-seting batasan suhu tubuh pasien yang masih diijinkan oleh Dokter. Rangkaian *push-button* diberikan inputan tegangan sebesar 4,9 *Volt* dan dirangkai dengan menggunakan *common ground.* Sehingga apabila terjadi penekanan pada tombol *push-button*, maka akan terjadi arus hubung singkat yang menyebabkan kondisi tegangan pada pin *push-button* yang terhubung dengan mikrokontroler berubah dari kondisi *"high"* 4,9 *Volt* menjadi kondisi *"low"* 0 *Volt.*

Pengujian rangkaian *push-button* dilakukan dengan cara mengukur tegangan pada *pin* mikrokontroler yang terhubung dengan rangkaian *push-button* dengan menggunakan multimeter *DC* dengan batas 20V/*DC*. Kondisi tegangan pada pin mikrokontroler yang terhubung dengan rangkaian *push-button* sebelum adanya penekanan pada tombol *push-button* adalah 4,9V. Berikut cara pengukuran tegangan pada rangkaian *push-button.*



Gambar 4.6. Cara Pengukuran Tegangan Pada Pengujian Rangkaian *Push-Button*

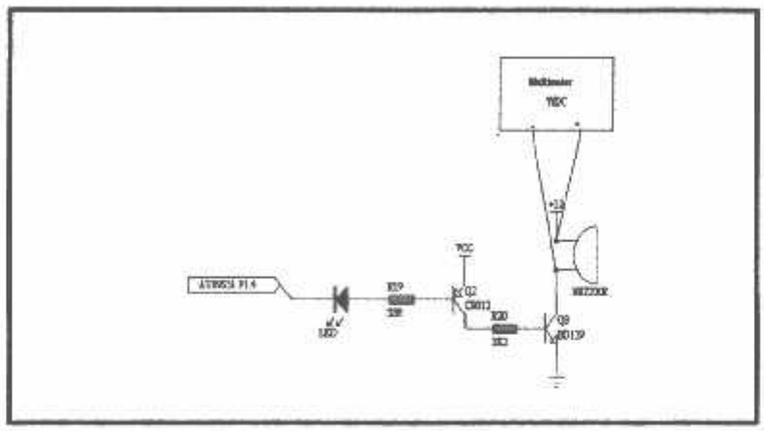
Tabel 4-5. Hasil Pengukuran Pada Rangkaian *Push-Button*

Push-Button (Tertekan)	P1.5 (V)	P1.6 (V)	P1.7 (V)
PB 1 *(Menu)*	0	4,85	4,85
PB 2 *(Up)*	4,85	0	4,85
PB 3 *(Down)*	4,85	4,85	0

4.6. Pengujian Rangkaian *Driver Buzzer*

Rangkaian *driver buzzer* berfungsi sebagai penguat arus yang diibutuhkan oleh *buzzer* agar dapat aktif sebagai sirine telah terjadinya gangguan pada daerah proteksi (beban). *Buzzer* yang digunakan adalah *Buzzer* dengan tegangan kerja 12V *DC*. Pengujian rangkaian *driver buzzer* secara *hardware* dilakukan dengan cara memberikan suatu kondisi *'low'* (0 *Volt*) dan *'high'* (5 Volt) pada *input*

driver buzzer pada Port 1.4 AT89S51, kemudian mengukur tegangan pada tegangan *output* rangkaian *driver Buzzer*.



Gambar 4.7. Rangkaian Pengujian Rangkaian *Driver Buzzer*

Tabel 4-6. Hasil Pengujian Rangkaian *Driver Buzzer*

Indek Pengujian	P1.4 (Volt/DC)	Tegangan pd *Buzzer* (Volt/DC)	Kondisi *Buzzer* (Bunyi/Diam)
1	0	11,9	Bunyi
2	4,9	-	Diam
3	0	11,9	Bunyi
4	4,9	-	Diam
5	0	11,9	Bunyi
6	4,9	-	Diam
7	0	11,9	Bunyi
8	4,9	-	Diam
9	0	11,9	Bunyi
10	4,9	-	Diam

BAB V

PENUTUP

5.1. Kesimpulan

Kesimpulan yang didapat adalah sebagai berikut :

1. Alat ini dapat bekerja baik pada ruang terbuka dengan jarak antara Transmitter danrecaiver maksimal 200 meter.
2. Sensor yang digunakan memiliki tingkat keakurasian 99.75%.
3. TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sedangkan mikrokontroler menggunakan catu daya +5 VDC dan memiliki I/O yang bekerja pada level tegangan 5. sehingga diperlukan rangkaian Level Converter.

5.2. Saran

Bila diinginkan untuk penggunaan sensor yang lebih banyak maka dapat ditambahkan rangkaian multiplexer sebagai data selector.

DAFTAR PUSTAKA

1. Andi Nalwan, Paulus. 2003. *Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrokontroler 89C51*. Jakarta: PT Elex Media Komputindo.
2. Budiharto, Widodo. 2007. *12 Proyek Mikrokontroler untuk Pemula*. Jakarta: PT Elex Media Komputindo.
3. Eko Putra, Agfianto. 2004. *Belajar Mikrokontroler AT89C51/52/55 (Teori dan Aplikasi)*. Yogyakarta: Penerbit Gava Media.
4. www.atmel.com
5. www.beyondlogic.com
6. www.fairchildsemi.com
7. www.joker-robotics.com
8. www.laipac.com
9. www.parallax.com
10. www.seiko instruments inc.com

LAMPIRAN

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ENERGI LISTRIK D III

Berita Acara Ujian Tugas akhir
Fakultas Teknologi Industri

Jurusan : TEKNIK ELEKTRO D III

Program Study : Energi Listrik

Nama : Acmad Budi Santoso

Nim : 02.52. 006

Judul Skripsi : PERANCANGAN SISTEM MONITORING SUHU RUANGAN JARAK JAUH TANPA KABEL MENGGUNAKAN TRW 2.4 G TRANSMITTER/RECEIVER

Dipertahankan didepan team penguji jenjang Diploma D III

Pada Hari :

Tanggal :

Dengan Nilai :

Panitia Tugas Akhir

Ketua

(Ir Mochtar Asroni, MS, ME)

Sekretaris

(Ir. Choirul Shaleh, MT)

Anggota Penguji

Penguji I

(Ir. H Taufik Hidayat, MT)

Penguji II

(Ir. M Abdul Hamid, MT)

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ENERGI LISTRIK D III

LEMBAR PERBAIKAN TUGAS AKHIR

Nama : **Acmad Budi Santoso**
NIM : **02.52.006**
Jurusan : **Teknik Elektro D III**
Konsentrasi : **Teknik Energi Listrik**

Judul Tugas Ahkir :**PERANCANGAN SISTEM MONITORING SUHU RUANGAN JARAK JAUH TANPA KABEL (WIRELESS) MENGGUNAKAN TRW 2.4 G TRANSMITTER/RECEIVER**

No	Materi Perbaikan	Keterangan
1	Alat Sudara harus dibandingkan dengan alat ukur lain yang di tempatkan di ruang yang sama, (Termometer Digital) Hasil pengukuran difoto dilampirkan pada hasil pengukuran.	
2	Akurasi / Validasi dan jarak monitoring?	

Dosen Pembimbing

(Komang Somawirata ST, MT)

Dosen Penguji I

(Ir. H Taufik Hidayat, MT)

Dosen Penguji II

(Ir. M Abdul Hamid, MT)

INSTITUT TEKNOLOGI NASIAONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ENERGI LISTIK DIII

LEMBAR ASISTENSI

Nama Mahasiswa : Acmad budi santoso
NIM : 02.52.006
Judul Skripsi :PERANCANGAN SISTEM MONITORING SUHU RUANGAN JARAK JAUH TANPA KABEL MENGGUNAKAN TRW 2,4G TRANSMITER / RECIVER
Dosen Pembimbing : Komang Somawinata, ST, MT

No.	Tanggal	Catatan	Tanda Tangan
1	16/07/09	Listing Asing di cetak miring (Italic), Ruangan / Sensor 71. Revisi pd PC.	
2.	25/07/09	Rangk ADC Ket IC, Bab IV Revisi. Rancang Alat + Lapangan	
3	17/08/09	Demo Alat. (perlu di tambah)	
4	19/08/09	Lapor Hasil Pengujian Lengkap Revisi Terima Jilid!	

```
;================================
;======Rangkaian Transmitter=====
;================================

ORG     000H
JMP     START

;================
;LCD KONSTANTA
;================
DISPCLR EQU 00000001B
FUNCSET EQU 00111000B
ENTRMOD EQU 00000110B
DISPON  EQU 00001100B

;==========
;PORT LCD
;==========
LCDE    BIT     P1.0
LCDRS   BIT     P1.1
PLCD    EQU     P0

;====================
;PUSH BUTTON ADDRES
;====================
TBDOWN  BIT     P1.7
TBUP    BIT     P1.6
TBMENU  BIT     P1.5

;==============
;ADC ADDRES
;==============
ADC_WR  BIT     P1.3
PADC    EQU     P2
BUZZ    BIT     P1.4

;==============
;TRW ADDRES
;==============
DATA1           BIT     P3.4
CLK1            BIT     P3.0
CS              BIT     P3.1
CE              BIT     P3.2
DR              BIT     P3.3

;=======================================
;ALAMAT VARIABLE (YG KOSONG 030H S.D 07FH)
;=======================================
PUTR            EQU     030H
Counter_5mS     EQU     31H
VR              EQU     32H
P_PA            EQU     056H
P_SA            EQU     057H
P_KA            EQU     058H

;=============
;MAIN PROGRAM
;=============
START:
        ACALL   INIT_TRW24
        LCALL   DELAY_1DETIK
        ;CALL    INITLCD
        ;CALL    IKLAN

ULANG:
        CALL    ADC
        MOV     R0,A
        ACALL   KIRIM_DATA
```

```
        MOV     A,R0
        ACALL   KIRIM_DATA
        CLR     CE
        ACALL   DELAY
        SETB    CE
;       LCALL   DELAY_100MS
        ;INC    R0
        SJMP    ULANG

MAIN_1:
        CALL    DISPLAY
        MOV     A,#086H
        CALL    LCDINS
        CALL    PULUHAN
        MOV     A,#087H
        CALL    LCDINS
        CALL    SATUAN
        MOV     A,#089H
        CALL    LCDINS
        CALL    KOMA
        JMP     MAIN_1

 PULUHAN:
        CALL    ADC
        MOV     B,#50
        DIV     AB
        MOV     P_PA,A
        CALL    TAMPIL
        RET

SATUAN:
        MOV     A,B
        MOV     B,#2
        MUL     AB
        MOV     B,#10
        DIV     AB
        MOV     P_SA,A
        CALL    TAMPIL
        RET

KOMA:
        MOV     A,B
        MOV     P_KA,A
        CALL    TAMPIL
        RET

 TAMPIL:
        ADD     A,#030H
        CALL    LCDDAT
        RET

;=================================
;PROSEDUR DISPLAY
;=================================
DISPLAY:
        MOV     A,#080H
        CALL    LCDINS
        MOV     A,#'S'
        CALL    LCDDAT
        MOV     A,#081H
        CALL    LCDINS
        MOV     A,#'U'
        CALL    LCDDAT
        MOV     A,#082H
        CALL    LCDINS
        MOV     A,#'H'
        CALL    LCDDAT
        MOV     A,#083H
```

```
        CALL    LCDINS
        MOV     A,#'U'
        CALL    LCDDAT
        MOV     A,#084H
        CALL    LCDINS
        MOV     A,#':'
        CALL    LCDDAT
        MOV     A,#088H
        CALL    LCDINS
        MOV     A,#','
        CALL    LCDDAT
        MOV     A,#08BH
        CALL    LCDINS
        MOV     A,#'C'
        CALL    LCDDAT
        RET

;=======PROSEDUR PEMBACAAN DATA ADC
ADC:
        LCALL   DELAY
        LCALL   DELAY
        SETB    ADC_WR
        CALL    DELAY
        CLR     ADC_WR
        CALL    DELAY
        MOV     A,PADC
        RET

KIRIM:
        PUSH    B
        MOV     B,#8
KIRIM8X:
        CLR     CLK1
        ACALL   DELAY
        RLC     A
        MOV     DATA1,C
        ACALL   DELAY
        SETB    CLK1
        ACALL   DELAY
        CLR     CLK1
        CLR     DATA1
        DJNZ    B,KIRIM8X
        POP     B
        RET

KIRIM_PRINTAH:
        CLR     CE
        ACALL   DELAY
        SETB    CS
        ACALL   DELAY
        SJMP    KIRIM_KAN
KIRIM_DATA:
        SETB    CE
        ACALL   DELAY
        CLR     CS
        ACALL   DELAY
KIRIM_KAN:
        ACALL   KIRIM
        RET

INIT_TRW24:
        CLR     DATA1
        CLR     CS
        CLR     CLK1
        LCALL   DELAY_100MS
        MOV     B,#18
        MOV     DPTR,#TABEL_INIT
INT_TRW24KIRIM:
```

```
        ACALL   AMBIL_TABEL
        ACALL   KIRIM_PRINTAH
        INC     DPTR
        DJNZ    B,INT_TRW24KIRIM
        RET

AMBIL_TABEL:
        MOV     A,#0
        MOVC    A,@A+DPTR
        RET

TABEL_INIT:
        DB      10001110B       ;143-136
        DB      00001000B       ;135-128
        DB      00011100B       ;127-120

        DB      00000000B       ;119-112 (DATA2_W)
        DB      00001000B       ;111-104 (DATA1_W)

        DB      00000000B       ;103-96 (ADDR2)
        DB      00000000B       ;95-88
        DB      00000000B       ;87-80
        DB      00000000B       ;79-72
        DB      00000000B       ;71 64

        DB      00000000B       ;63-56 (ADDR1)
        DB      00000000B       ;55-48
        DB      00000000B       ;47-40
        DB      00000000B       ;39-32
        DB      01000000B       ;31-24

        DB      01000011B       ;23-16 (ADDR_W,CRC_L,CRC_EN)

        DB      01101111B       ;15-8 (RX2_EN,CM,RFDR_SB,XO_F,RF_PWR)
        DB      00010100B       ;7-0 (RF_CH,RXEN)

;=======PROSEDUR CETAK STRING=========
PRINTSTRINGLOOP:
        CALL    LCDDAT
        INC     DPTR
LCDSTRING:
        CLR     A
        MOVC    A,@A+DPTR
        JNZ     PRINTSTRINGLOOP
        RET

;=======PROSEDUR KONTROL BYTE OPERASI PADA LCD
LCDINS:
        MOV     PLCD,A
        CLR     LCDRS
        SJMP    LCDOUT
LCDDAT:
        MOV     PLCD,A
        SETB    LCDRS
LCDOUT:
        SETB    LCDE
        CALL    DELAY1
        CLR     LCDE
        CALL    DELAY1
        RET

;=======PROSEDUR DELAY=========
DELAY4_1:
        DJNZ    R7,DELAY4_1
        RET

DELAY3:
```

```
        MOV     PUTR,A
MUTERZ:
        CALL    DELAY2
        DJNZ    PUTR,MUTERZ
        RET
DELAY2:
        MOV     R5,#130
MUTERX:
        MOV     R6,#250
        CALL    DELAY
        DJNZ    R6,$
        DJNZ    R5,MUTERX
        RET
DELAY1:
        MOV     R3,#08
MUTER:
        MOV     R4,#0255
        DJNZ    R4,$
        DJNZ    R3,MUTER
        RET

DELAY:
        PUSH    B
        MOV     B,#50
        DJNZ    B,$
        POP     B
        ACALL   DELAY_5MS
        RET

;* Delay 5 mS sebanyak 200 x

Delay_1detik:
        Mov     Counter_5mS,#0200

Tunggu_1detik:
        Acall   Delay_5mS
        Djnz    Counter_5mS,Tunggu_1detik
        Ret

;***********
;* Delay 5 mS sebanyak 100 x

Delay_500mS:
        Mov     Counter_5mS,#0100

Tunggu_500mS:
        Acall   Delay_5mS
        Djnz    Counter_5mS,Tunggu_500mS
        Ret

;***********
;* Delay 5 mS sebanyak 20 x

Delay_100mS:
        Mov     Counter_5mS,#020

Tunggu_100mS:
        Acall   Delay_5mS
        Djnz    Counter_5mS,Tunggu_100mS
        Ret

;***********
;* Delay 5 mS sebanyak 15 x

Delay_75mS:
        Mov     Counter_5mS,#015

Tunggu_75mS:
```

```
        Acall   Delay_5mS
        Djnz    Counter_5mS,Tunggu_75mS
        Ret

;**************
;* Delay ini bekerja hanya pada crystal 11.0592 MHz

Delay_5mS:
        Push    TMOD
        Mov     TMOD,#21H               ;Timer Mode 16 bit counter
        Mov     TH0,#0EDH
        Mov     TL0,#0FFH
        Setb    TR0

Tunggu_5mS:
        Jbc     TF0,Sudah_5mS
        Ajmp    Tunggu_5mS

Sudan_5mS:
        Clr     TR0
        Pop     TMOD
        Ret

;=======PROSEDUR INISIALISASI LCD
INITLCD:
        MOV     A,#DISPCLR
        CALL    LCDINS
        CALL    DELAY1
        MOV     A,#FUNCSET
        CALL    LCDINS
        CALL    DELAY1
        MOV     A,#DISPON
        CALL    LCDINS
        CALL    DELAY1
        MOV     A,#ENTRMOD
        CALL    LCDINS
        CALL    DELAY1
        MOV     A,#DISPCLR
        CALL    LCDINS
        CALL    DELAY2
        RET

        ;============================
        ; DAFTAR IKLAN
        ;============================
        IKLAN:
        MOV     A,#080H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_1
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_2
        CALL    LCDSTRING
        MOV     A,#03
        CALL    DELAY3
        MOV     A,#080H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_3
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_4
        CALL    LCDSTRING
        MOV     A,#03
        CALL    DELAY3
        MOV     A,#080H
        CALL    LCDINS
```

```
        MOV     DPTR,#ABOUT_5
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_6
        CALL    LCDSTRING
        MOV     A,#03
        CALL    DELAY3
        MOV     A,#080H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_7
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_8
        CALL    LCDSTRING
        MOV     A,#03
        CALL    DELAY3
        MOV     A,#DISPCLR
        CALL    LCDINS
        RET
;=============
;TEXT SECTION
;=============
;=============
;TEXT SECTION
;=============
ABOUT_1 : DB 'SISTM MONITORING',0
ABOUT_2 : DB '   SUHU RUANGAN  ',0
ABOUT_3 : DB '    TANPA KABEL  ',0
ABOUT_4 : DB '    (WIRELESS)   ',0
ABOUT_5 : DB '  Achmad Budi S ',0
ABOUT_6 : DB '  02.52. 006    ',0
ABOUT_7 : DB 'DOSEN PEMBIMBING',0
ABOUT_8 : DB 'Komang S, ST, MT',0

END
```

```
;=============================
;======Rangkaian Receiver=====
;=============================

ORG     000H
JMP     START

;===============
;LCD KONSTANTA
;===============
DISPCLR EQU 00000001B
FUNCSET EQU 00111000B
ENTRMOD EQU 00000110B
DISPON  EQU 00001100B

;==========
;PORT LCD
;==========
LCDE    BIT     P1.0
LCDRS   BIT     P1.1
PLCD    EQU     P0

;====================
;PUSH BUTTON ADDRES
;====================
TBDOWN  BIT     P1.7
TBUP    BIT     P1.6
TBMENU  BIT     P1.5
BUZZ    BIT     P1.3

;==============
;TRW ADDRES
;==============
DATA1           BIT     P3.4
CLK1            BIT     P3.0
CS              BIT     P3.1
CE              BIT     P3.2
DR1             BIT     P3.3

;=======================================
;ALAMAT VARIABLE (YG KOSONG 030H S.D 07FH)
;=======================================
PUTR            EQU     030H
Counter_5mS     EQU     31H
VR              EQU     32H
P_PA            EQU     056H
P_SA            EQU     057H
P_KA            EQU     058H
BA_P            EQU     032H
BA_S            EQU     033H
BA_K            EQU     034H
BB_P            EQU     035H
BB_S            EQU     036H
BB_K            EQU     037H
BA1_P           EQU     038H
BA1_S           EQU     039H
BA1_K           EQU     03AH
BB1_P           EQU     03BH
BB1_S           EQU     03CH
BB1_K           EQU     03DH
BB              EQU     054H
BP              EQU     055H
MA              EQU     066H

;=============
;MAIN PROGRAM
;=============
START:
```

```
        MOV     A,#DISPCLR
        CALL    LCDINS
        ACALL   INIT_TRW24
        LCALL   DELAY_1DETIK
        CALL    INITLCD
        CALL    IKLAN
        MOV     A,#DISPCLR
        CALL    LCDINS
        CALL    STATUS
        ACALL   TERIMA
        ACALL   TERIMA
        MOV     A,#DISPCLR
        CALL    LCDINS
        CALL    DISPLAY_OFF
        MOV     MA,#0
        LCALL   RST_BATASAN
ULANG:
        JB      TBMENU,ULANG1
ULANG2:
        JNB     TBMENU,$
        CALL    RST_BATAS
        MOV     A,#DISPCLR
        CALL    LCDINS
        CALL    P_TEXT1
        CALL    SET_UP1AP
        SJMP    ULANG

ULANG1:
        MOV     A,#086H
        CALL    LCDINS
        ACALL   TERIMA
        CALL    PULUHAN

        MOV     A,#087H
        CALL    LCDINS
        CALL    SATUAN

        MOV     A,#089H
        CALL    LCDINS
        CALL    KOMA
        MOV     A,MA
        CJNE    A,#0,CEK1_ATAS
        SJMP    ULANG

CEK1_ATAS:
        MOV     B,#10
        MOV     A,P_PA
        MUL     AB
        ADD     A,P_SA
        MOV     BP,A
        MOV     B,#10
        MOV     A,BA_P
        MUL     AB
        ADD     A,BA_S
        SUBB    A,BP
        JC      SET_BUZZ1
        JNZ     ULANG
        MOV     A,BA_K
        SUBB    A,P_KA
        JC      SET_BUZZ1
CEK1_BAWAH:
        MOV     B,#10
        MOV     A,BB_P
        MUL     AB
        ADD     A,BB_S
        MOV     BB,A
        MOV     B,#10
        MOV     A,P_PA
```

```
        MUL     AB
        ADD     A,P_SA
        SUBB    A,BB
        JC      SET_BUZZ1
        JNZ     ULANG
        MOV     A,P_KA
        SUBB    A,BB_K
        JC      SET_BUZZ1
        JMP     ULANG

SET_BUZZ1:
        CALL    OUT_P1
        CALL    BUUZ
        JB      TBMENU,SET_BUZZ1
        JNB     TBMENU,$
        SETB    BUZZ
        CALL    RST_BATASAN
        MOV     MA,#0
        MOV     A,#DISPCLR
        CALL    LCDINS
        CALL    DISPLAY_OFF
        JMP     ULANG
BUUZ:
        CLR     BUZZ
        CALL    DELAY
        CALL    DELAY
        CALL    DELAY
        CALL    DELAY
        SETB    BUZZ
        CALL    DELAY
        CALL    DELAY
        CALL    DELAY
        CALL    DELAY

Out_P1:
        MOV     A,#080H
        CALL    LCDINS
        MOV     DPTR,#TEXT_3
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#TEXT_4
        CALL    LCDSTRING
        RET

;=============================================================
;PROSEDUR AMBIL BATASAN
;=============================================================
SET_UP1AP:
        MOV     A,#08AH         ; SET P1 BATAS ATAS PULUHAN
        CALL    LCDINS

UP_1AP:
        JB      TBUP,DOWN_1AP
        JNB     TBUP,$
        INC     BA_P
        MOV     A,#10
        CJNE    A,BA_P,TAMPIL3
        MOV     BA_P,#0
        MOV     A,#0
        CALL    TAMPIL
        JMP     SET_UP1AP

TAMPIL3 :
        MOV     A,BA_P
        ADD     A,#030H
        CALL    LCDDAT
        JMP     SET_UP1AP
```

```
DOWN_1AP:
        JB      TBDOWN,MENU_1AS
        JNB     TBDOWN,$
        MOV     A,BA_P
        JZ      SET_UP1AP
        DEC     BA_P
        MOV     A,BA_P
        CALL    TAMPIL
        JMP     SET_UP1AP

MENU_1AS:
        JB      TBMENU,SET_UP1AP
        JNB     TBMENU,$

SET_UP1AS:
        MOV     A,#08BH         ; SET P1 BATAS ATAS SATUAN
        CALL    LCDINS

UP_1AS:
        JB      TBUP,DOWN_1AS   ; INC P1 BATAS ATAS SATUAN
        JNB     TBUP,$
        INC     BA_S
        MOV     A,#10
        CJNE    A,BA_S,TAMPIL4
        MOV     BA_S,#0
        MOV     A,#0
        CALL    TAMPIL
        JMP     SET_UP1AS

TAMPIL4:
        MOV     A,BA_S
        ADD     A,#030H
        CALL    LCDDAT
        JMP     SET_UP1AS

DOWN_1AS:
        JB      TBDOWN,MENU_1AK ; DEC P1 BATAS ATAS SATUAN
        JNB     TBDOWN,$
        MOV     A,BA_S
        JZ      SET_UP1AS
        DEC     BA_S
        MOV     A,BA_S
        CALL    TAMPIL
        JMP     SET_UP1AS

MENU_1AK:
        JB      TBMENU,SET_UP1AS
        JNB     TBMENU,$

SET_UP1AK:
        MOV     A,#08DH         ; SET P1 BATAS ATAS BELAKANG KOMA
        CALL    LCDINS
UP_1AK:
        JB      TBUP,DOWN_1AK   ; INC P1 BATAS ATAS BELAKANG KOMA
        JNB     TBUP,$
        INC     BA_K
        MOV     A,#10
        CJNE    A,BA_K,TAMPIL5
        MOV     BA_K,#0
        MOV     A,#0
        CALL    TAMPIL
        JMP     SET_UP1AK

TAMPIL5:
        MOV     A,BA_K
        ADD     A,#030H
        CALL    LCDDAT
```

```
        JMP     SET_UP1AK

DOWN_1AK:
        JB      TBDOWN,MENU_1BP ; DEC P1 BATAS ATAS BELAKANG KOMA
        JNB     TBDOWN,$
        MOV     A,BA_K
        JZ      SET_UP1AK
        DEC     BA_K
        MOV     A,BA_K
        CALL    TAMPIL
        JMP     SET_UP1AK

MENU_1BP:
        JB      TBMENU,UP_1AK
        JNB     TBMENU,$

SET_UP1BP:
        MOV     A,#0CAH         ; SET P1 BATAS BAWAH PULUHAN
        CALL    LCDINS

UP_1BP:
        JB      TBUP,DOWN_1BP
        JNB     TBUP,$
        INC     BB_P
        MOV     A,#10
        CJNE    A,BB_P,TAMPIL6
        MOV     BB_P,#0
        MOV     A,#0
        CALL    TAMPIL
        JMP     SET_UP1BP

TAMPIL6:
        MOV     A,BB_P
        ADD     A,#030H
        CALL    LCDDAT
        JMP     SET_UP1BP

DOWN_1BP:
        JB      TBDOWN,MENU_1BS
        JNB     TBDOWN,$
        MOV     A,BB_P
        JZ      SET_UP1BP
        DEC     BB_P
        MOV     A,BB_P
        CALL    TAMPIL
        JMP     SET_UP1BP

MENU_1BS:
        JB      TBMENU,UP_1BP
        JNB     TBMENU,$

SET_UP1BS:
        MOV     A,#0CBH         ; SET P1 BATAS BAWAH SATUAN
        CALL    LCDINS

UP_1BS:
        JB      TBUP,DOWN_1BS   ; INC P1 BATAS BAWAH SATUAN
        JNB     TBUP,$
        INC     BB_S
        MOV     A,#10
        CJNE    A,BB_S,TAMPIL7
        MOV     BB_S,#0
        MOV     A,#0
        CALL    TAMPIL
        JMP     SET_UP1BS

TAMPIL7:
        MOV     A,BB_S
```

```
        ADD     A,#030H
        CALL    LCDDAT
        JMP     SET_UP1BS

DOWN_1BS:
        JB      TBDOWN,MENU_1BK ; DEC P1 BATAS BAWAH SATUAN
        JNB     TBDOWN,$
        MOV     A,BB_S
        JZ      SET_UP1BS
        DEC     BB_S
        MOV     A,BB_S
        CALL    TAMPIL
        JMP     SET_UP1BS
MENU_1BK:
        JB      TBMENU,UP_1BS
        JNB     TBMENU,$

SET_UP1BK:
        MOV     A,#0CDH         ; SET P1 BATAS BAWAH BELAKANG KOMA
        CALL    LCDINS
UP_1BK:
        JB      TBUP,DOWN_1BK   ; INC P1 BATAS BAWAH BELAKANG KOMA
        JNB     TBUP,$
        INC     BB_K
        MOV     A,#10
        CJNE    A,BB_K,TAMPIL8
        MOV     BB_K,#0
        MOV     A,#0
        CALL    TAMPIL
        JMP     SET_UP1BK

TAMPIL8:
        MOV     A,BB_K
        ADD     A,#030H
        CALL    LCDDAT
        JMP     SET_UP1BK

DOWN_1BK:
        JB      TBDOWN,SET1_END ; DEC P1 BATAS BAWAH BELAKANG KOMA
        JNB     TBDOWN,$
        MOV     A,BB_K
        JZ      SET_UP1BK
        DEC     BB_K
        MOV     A,BB_K
        CALL    TAMPIL
        JMP     SET_UP1BK

SET1_END:
        JB      TBMENU,UP_1BK
        JNB     TBMENU,$
        MOV     A,#DISPON
        CALL    LCDINS
        MOV     A,#DISPCLR
        CALL    LCDINS
        MOV     MA,#1
        CALL    DISPLAY_ON
        RET

RST_BATAS:
        MOV     BA_P,#0
        MOV     BA_S,#0
        MOV     BA_K,#0
        MOV     BB_P,#0
        MOV     BB_S,#0
        MOV     BB_K,#0
        RET

PULUHAN:
```

```
        MOV     B,#50
        DIV     AB
        MOV     P_PA,A
        CALL    TAMPIL
        RET

SATUAN:
        MOV     A,B
        MOV     B,#2
        MUL     AB
        MOV     B,#10
        DIV     AB
        MOV     P_SA,A
        CALL    TAMPIL
        RET

KOMA:
        MOV     A,B
        MOV     P_KA,A
        CALL    TAMPIL
        RET

;=================================
;PROSEDUR DISPLAY
;=========================
DISPLAY_OFF:
        MOV     A,#080H
        CALL    LCDINS
        MOV     A,#'S'
        CALL    LCDDAT
        MOV     A,#081H
        CALL    LCDINS
        MOV     A,#'U'
        CALL    LCDDAT
        MOV     A,#082H
        CALL    LCDINS
        MOV     A,#'H'
        CALL    LCDDAT
        MOV     A,#083H
        CALL    LCDINS
        MOV     A,#'U'
        CALL    LCDDAT
        MOV     A,#084H
        CALL    LCDINS
        MOV     A,#':'
        CALL    LCDDAT
        MOV     A,#088H
        CALL    LCDINS
        MOV     A,#','
        CALL    LCDDAT
        MOV     A,#08BH
        CALL    LCDINS
        MOV     A,#'C'
        CALL    LCDDAT
        MOV     A,#086H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#087H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#089H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#0C0H
        CALL    LCDINS
```

```
        MOV     A,#'M'
        CALL    LCDDAT
        MOV     A,#0C1H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#0C2H
        CALL    LCDINS
        MOV     A,#'D'
        CALL    LCDDAT
        MOV     A,#0C3H
        CALL    LCDINS
        MOV     A,#'E'
        CALL    LCDDAT
        MOV     A,#0C5H
        CALL    LCDINS
        MOV     A,#'A'
        CALL    LCDDAT
        MOV     A,#0C6H
        CALL    LCDINS
        MOV     A,#'L'
        CALL    LCDDAT
        MOV     A,#0C7H
        CALL    LCDINS
        MOV     A,#'A'
        CALL    LCDDAT
        MOV     A,#0C8H
        CALL    LCDINS
        MOV     A,#'R'
        CALL    LCDDAT
        MOV     A,#0C9H
        CALL    LCDINS
        MOV     A,#'M'
        CALL    LCDDAT
        MOV     A,#0CBH
        CALL    LCDINS
        MOV     A,#'O
        CALL    LCDDAT
        MOV     A,#0CCH
        CALL    LCDINS
        MOV     A,#'F'
        CALL    LCDDAT
        MOV     A,#0CDH
        CALL    LCDINS
        MOV     A,#'F'
        CALL    LCDDAT
        RET

DISPLAY_ON:
        MOV     A,#080H
        CALL    LCDINS
        MOV     A,#'S'
        CALL    LCDDAT
        MOV     A,#081H
        CALL    LCDINS
        MOV     A,#'U'
        CALL    LCDDAT
        MOV     A,#082H
        CALL    LCDINS
        MOV     A,#'H'
        CALL    LCDDAT
        MOV     A,#083H
        CALL    LCDINS
        MOV     A,#'U'
        CALL    LCDDAT
        MOV     A,#084H
        CALL    LCDINS
        MOV     A,#':'
```

```
        CALL    LCDDAT
        MOV     A,#088H
        CALL    LCDINS
        MOV     A,#' '
        CALL    LCDDAT
        MOV     A,#088H
        CALL    LCDINS
        MOV     A,#'C'
        CALL    LCDDAT
        MOV     A,#086H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#087H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#089H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     A,#'M'
        CALL    LCDDAT
        MOV     A,#0C1H
        CALL    LCDINS
        MOV     A,#'0'
        CALL    LCDDAT
        MOV     A,#0C2H
        CALL    LCDINS
        MOV     A,#'D'
        CALL    LCDDAT
        MOV     A,#0C3H
        CALL    LCDINS
        MOV     A,#'E'
        CALL    LCDDAT
        MOV     A,#0C4H
        CALL    LCDINS
        MOV     A,#'A'
        CALL    LCDDAT
        MOV     A,#0C6H
        CALL    LCDINS
        MOV     A,#'L'
        CALL    LCDDAT
        MOV     A,#0C7H
        CALL    LCDINS
        MOV     A,#'A'
        CALL    LCDDAT
        MOV     A,#0C8H
        CALL    LCDINS
        MOV     A,#'R'
        CALL    LCDDAT
        MOV     A,#0C9H
        CALL    LCDINS
        MOV     A,#'M'
        CALL    LCDDAT
        MOV     A,#0CBH
        CALL    LCDINS
        MOV     A,#'O'
        CALL    LCDDAT
        MOV     A,#0CCH
        CALL    LCDINS
        MOV     A,#'N'
        CALL    LCDDAT
        RET

TAMPIL:
```

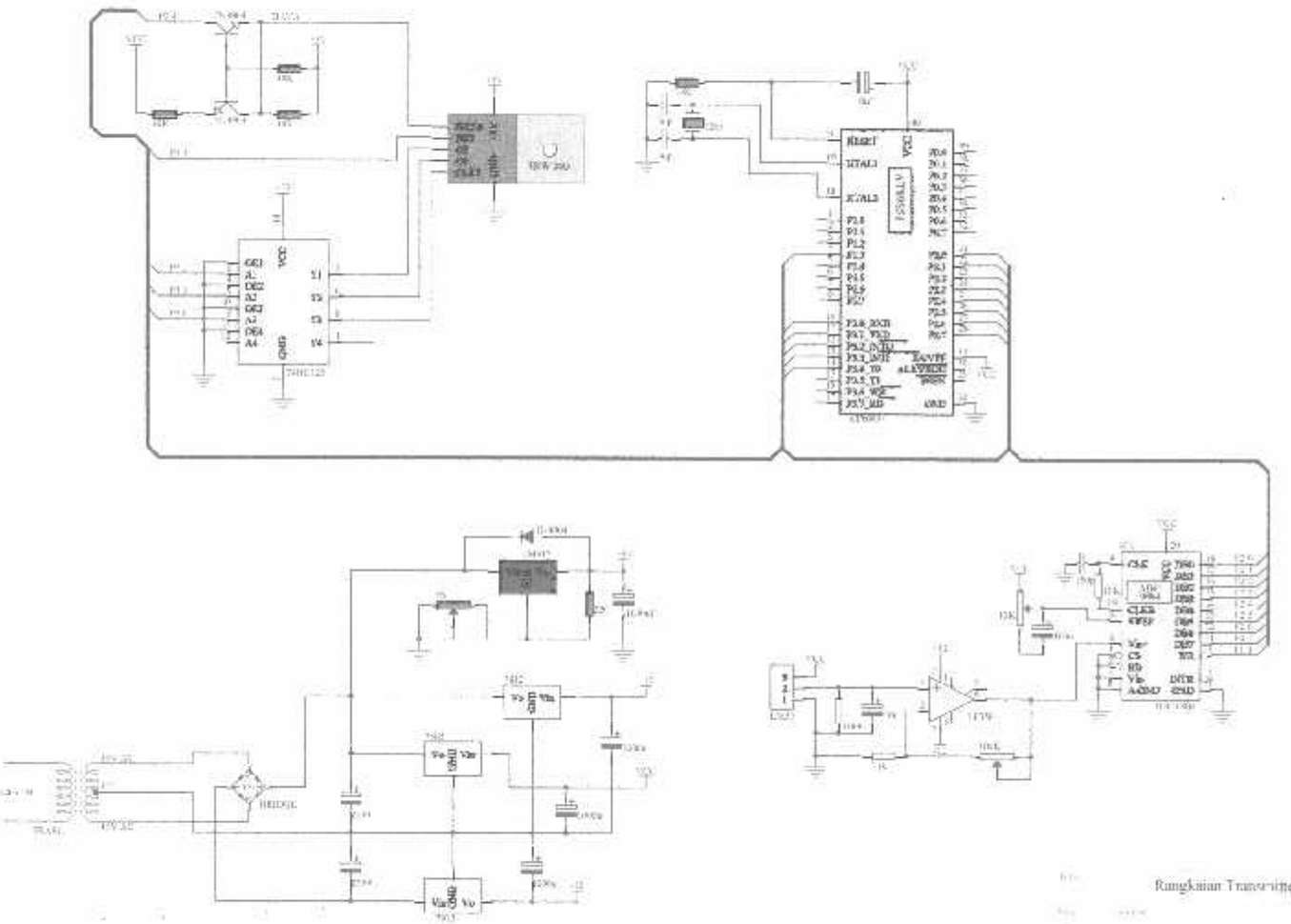
```
        ADD     A,#030H
        CALL    LCDDAT
        RET

KIRIM:
        PUSH    B
        MOV     B,#8
KIRIM8X:
        CLR     CLK1
        ACALL   DELAY
        RLC     A
        MOV     DATA1,C
        ACALL   DELAY
        SETB    CLK1
        ACALL   DELAY
        CLR     CLK1
        CLR     DATA1
        DJNZ    B,KIRIM8X
        POP     B
        RET

TERIMA:

        ACALL   DELAY
        SETB    CE
        ACALL   DELAY
        SETB    DR1
WAIT:


        JNB     TBMENU,ULANG2C
        JNB     DR1,WAIT
        PUSH    B
        MOV     B,#8
        MOV     A,#0H
TERIMA8X:
        CLR     CLK1
        ACALL   DELAY
        SETB    DATA1
        SETB    CLK1
        ACALL   DELAY
        RL      A
        MOV     C,DATA1
        MOV     ACC.0,C
        DJNZ    B,TERIMA8X
        CLR     CLK1
        POP     B
        RET
ULANG2C:

        JMP     ULANG2

KIRIM_PRINTAH:
        CLR     CE
        ACALL   DELAY
        SETB    CS
        ACALL   DELAY
        SJMP    KIRIM_KAN
KIRIM_DATA:
        SETB    CE
        ACALL   DELAY
        CLR     CS
        ACALL   DELAY
KIRIM_KAN:
        ACALL   KIRIM
        RET
```

```
INIT_TRW24:
        CLR     CE
        CLR     DATA1
        CLR     CS
        CLR     CLK1
        LCALL   DELAY_1DETIK
        MOV     B,#18
        MOV     DPTR,#TABEL_INIT
INT_TRW24KIRIM:
        ACALL   AMBIL_TABEL
        ACALL   KIRIM_PRINTAH
        INC     DPTR
        DJNZ    B,INT_TRW24KIRIM
        RET

AMBIL_TABEL:
        MOV     A,#0
        MOVC    A,@A+DPTR
        RET

TABEL_INIT:
        DB      10001110B       ;143-136
        DB      00001000B       ;135-128
        DB      00011100B       ;127-120

        DB      00000000B       ;119-112 (DATA2_W)

        DB      00001000B       ;111-104 (DATA1_W)

        DB      00000000B       ;103-96 (ADDR2)
        DB      00000000B       ;95-88
        DB      00000000B       ;87-80
        DB      00000000B       ;79-72
        DB      00000000B       ;71-64

        DB      00000000B       ;63-56 (ADDR1)
        DB      00000000B       ;55-48
        DB      00000000B       ;47-40
        DB      00000000B       ;39-32
        DB      01000000B       ;31-24

        DB      01000011B       ;23-16 (ADDR_W,CRC_L,CRC_EN)

        DB      01101111B       ;15-8 (RX2_EN,CM,RFDR_SB,XO_F,RF_PWR)
        DB      00010101B       ;7-0 (RF_CH,RXEN)

;=======PROSEDUR CETAK STRING=========
PRINTSTRINGLOOP:
        CALL    LCDDAT
        INC     DPTR
LCDSTRING:
        CLR     A
        MOVC    A,@A+DPTR
        JNZ     PRINTSTRINGLOOP
        RET

;=======PROSEDUR KONTROL BYTE OPERASI PADA LCD
LCDINS:
        MOV     PLCD,A
        CLR     LCDRS
        SJMP    LCDOUT
LCDDAT:
        MOV     PLCD,A
        SETB    LCDRS
LCDOUT:
        SETB    LCDE
        CALL    DELAY1
        CLR     LCDE
```

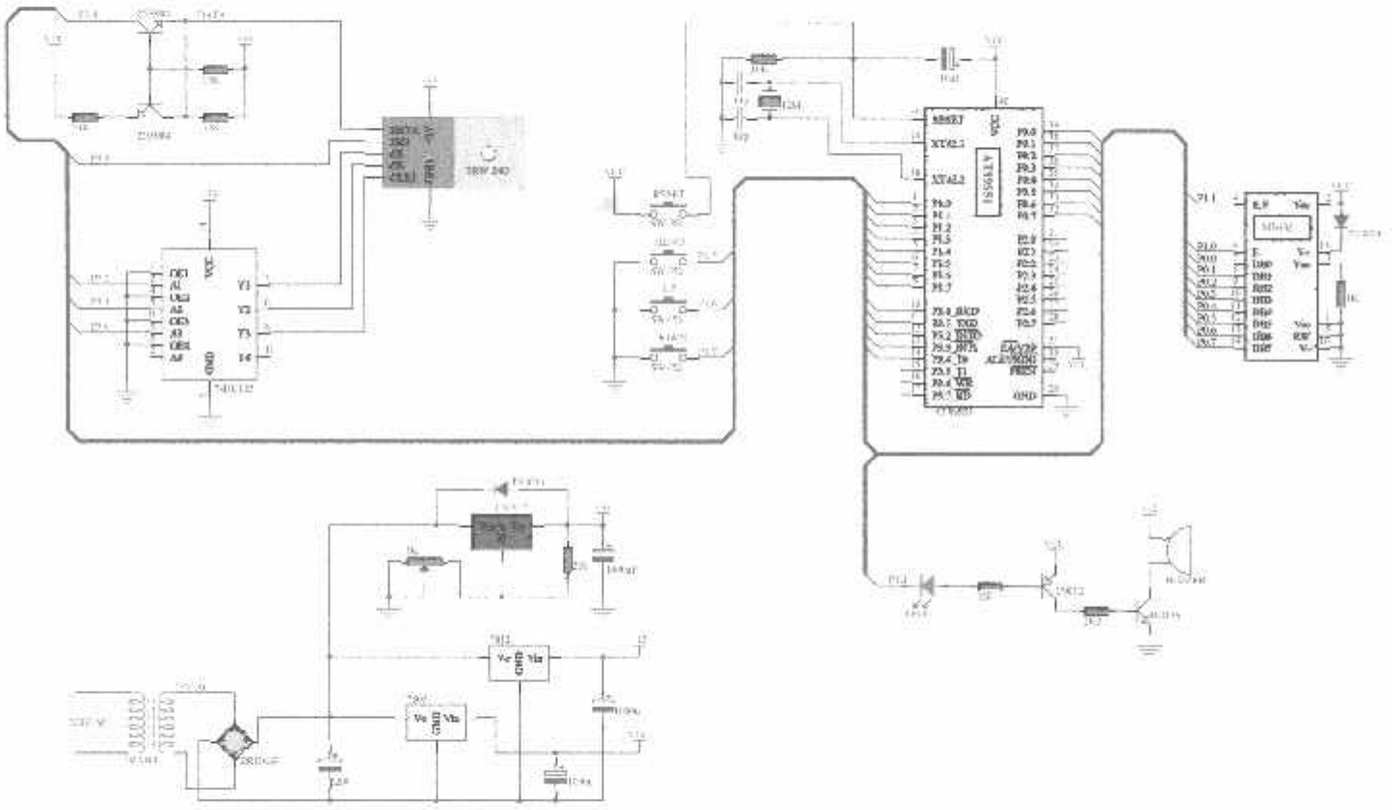
```
        CALL    DELAY1
        RET

;=======PROSEDUR DELAY=========

DELAY3:
        MOV     PUTR,A
MUTERZ:
        CALL    DELAY2
        DJNZ    PUTR,MUTERZ
        RET
DELAY2:
        MOV     R5,#130
MUTERX:
        MOV     R6,#250
        CALL    DELAY
        DJNZ    R6,$
        DJNZ    R5,MUTERX
        RET
DELAY1:
        MOV     R3,#08
MUTER:
        MOV     R4,#0255
        DJNZ    R4,$
        DJNZ    R3,MUTER
        RET

DELAY:
        PUSH    B
        MOV     B,#50
        DJNZ    B,$
        POP     B
        ACALL   DELAY_5MS
        RET

* Delay 5 mS sebanyak 200 x

Delay_1detik:
        Mov     Counter_5mS,#0200

Tunggu_1detik:
        Acall   Delay_5mS
        Djnz    Counter_5mS,Tunggu_1detik
        Ret

**********
* Delay ini bekerja hanya pada crystal 11.0592 MHz

Delay_5mS:
        Push    TMOD
        Mov     TMOD,#21H               ;Timer Mode 16 bit counter
        Mov     TH0,#0EDH
        Mov     TL0,#0FFH
        Setb    TR0

Tunggu_5mS:
        Jbc     TF0,Sudah_5mS
        Ajmp    Tunggu_5mS

Sudah_5mS:
        Clr     TR0
        Pop     TMOD
        Ret

=======PROSEDUR INISIALISASI LCD
INITLCD:
        MOV     A,#DISPCLR
        CALL    LCDINS
```

```
CALL    DELAY1
MOV     A,#FUNCSET
CALL    LCDINS
CALL    DELAY1
MOV     A,#DISPON
CALL    LCDINS
CALL    DELAY1
MOV     A,#ENTRMOD
CALL    LCDINS
CALL    DELAY1
MOV     A,#DISPCLR
CALL    LCDINS
CALL    DELAY2
RET

;============================
; DAFTAR IKLAN
;============================
IKLAN:
MOV     A,#080H
CALL    LCDINS
MOV     DPTR,#ABOUT_1
CALL    LCDSTRING
MOV     A,#0C0H
CALL    LCDINS
MOV     DPTR,#ABOUT_2
CALL    LCDSTRING
MOV     A,#03
CALL    DELAY3
MOV     A,#080H
CALL    LCDINS
MOV     DPTR,#ABOUT_3
CALL    LCDSTRING
MOV     A,#0C0H
CALL    LCDINS
MOV     DPTR,#ABOUT_4
CALL    LCDSTRING
MOV     A,#03
CALL    DELAY3
MOV     A,#080H
CALL    LCDINS
MOV     DPTR,#ABOUT_5
CALL    LCDSTRING
MOV     A,#0C0H
CALL    LCDINS
MOV     DPTR,#ABOUT_6
CALL    LCDSTRING
MOV     A,#03
CALL    DELAY3
MOV     A,#080H
CALL    LCDINS
MOV     DPTR,#ABOUT_7
CALL    LCDSTRING
MOV     A,#0C0H
CALL    LCDINS
MOV     DPTR,#ABOUT_8
CALL    LCDSTRING
MOV     A,#03
CALL    DELAY3
MOV     A,#080H
CALL    LCDINS
MOV     DPTR,#ABOUT_9
CALL    LCDSTRING
MOV     A,#0C0H
CALL    LCDINS
MOV     DPTR,#ABOUT_A
CALL    LCDSTRING
MOV     A,#03
```

```
        CALL    DELAY3
        RET
STATUS:
        MOV     A,#080H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_B
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#ABOUT_C
        CALL    LCDSTRING
        MOV     A,#03
        CALL    DELAY3
        RET

P_TEXT1:
        MOV     A,#080H
        CALL    LCDINS
        MOV     DPTR,#TEXT_1
        CALL    LCDSTRING
        MOV     A,#0C0H
        CALL    LCDINS
        MOV     DPTR,#TEXT_2
        CALL    LCDSTRING
        CALL    KRSR
        RET

KRSR:
        MOV     A,#00001101B
        CALL    LCDINS
        RET

DISPLON:
        MOV     A,#DISPON
        CALL    LCDINS
        RET
RST_BATASAN:
        MOV     BA_P,#0
        MOV     BA_S,#0
        MOV     BA_K,#0
        MOV     BB_P,#0
        MOV     BB_S,#0
        MOV     BB_K,#0
        RET
;=============
;TEXT SECTION
;=============
;=============
;TEXT SECTION
;=============
ABOUT_1 : DB 'SISTM MONITORING',0
ABOUT_2 : DB '  SUHU RUANGAN  ',0
ABOUT_3 : DB '   TANPA KABEL  ',0
ABOUT_4 : DB '   (WIRELESS)   ',0
ABOUT_5 : DB '  Achmad Budi S ',0
ABOUT_6 : DB '  02.52. 006    ',0
ABOUT_7 : DB 'DOSEN PEMBIMBING',0
ABOUT_8 : DB 'Komang S, ST, MT',0
ABOUT_9 : DB '===ITN MALANG===',0
ABOUT_A : DB '===   2008   ===',0
ABOUT_B : DB 'NO',0
ABOUT_C : DB 'CONECTION',0
TEXT_1  : DB 'Bts Panas:00,0 C',0
TEXT_2  : DB 'Bts Dgin :00,0 C',0
TEXT_3  : DB '     RUANG 1    ',0
TEXT_4  : DB 'TERJADI GANGGUAN',0

END
```



Rangkaian Transmitter



Rangkaian Receiver