INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA

SKRIPSI

PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING BERBASIS AT89S8252 YANG DIINFORMASIKAN VIA SMS

Disusun Oleh :
MOH. MUFID MURTADLO
02.17.144

MARET 2007

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

LEMBAR PERSETUJUAN

PERENCANAAN DAN PEMBUATAN
ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING
BERBASIS AT89S8252 YANG DI INFORMASIKAN VIA SMS

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik
Pada Jurusan Teknik Elektro Strata Satu (S-1) Konsentrasi Elektronika

Disusun Oleh :

MOH. MUFID MURTADLO

NIM : 02.17.144

Diperiksa dan Disetujui,

Dosen Pembimbing I	Dosen Pembimbing II
(Ir. F. Yudi Limpraptono, MT)	**(Ir. Mimien Mustikawati)**
NIP.Y. 1039500274	**NIP.P. 1030000352**

Mengatahui,
Ketua Jurusan Teknik Elektro S-1

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO
STRATA - 1

(Ir. F. Yudi Limpraptono, MT)
NIP.Y. 1039500274

KONSENTRASI ELEKTRONIKA

JURUSAN TEKNIK ELEKTRO S - 1

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

NAMA	: Moh. Mufid Murtadlo
NIM	: 02 17 144
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 9 Desember 2006 s/d 9 Juni 2007
Judul Skripsi	: Perencanaan Dan Pembuatan Alat Kontrol Suhu Pada Indukan Ayam Pedaging Berbasis AT 89S8252 Yang Diinformasikan Via SMS

Dipertahankan di hadapan Tim Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari	: Jum'at
Tanggal	: 16 Maret 2007
Nilai	: 85,5 (A)

Panitia Majelis Penguji,

Ketua

(Ir. Mochtar Asroni, MSME)
NIP.Y. 1018100036

Sekretaris

(Ir. F. Yudi Limpraptono, MT)
NIP.Y. 1039500274

Anggota Penguji

Penguji 1

(Ir. Eko Nurcahyo)
NIP.Y. 1028700172

Penguji 2

(I Komang Somawirata, ST., MT.)
NIP.P. 1030100361

ABSTRAKSI

JUDUL : PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING BERBASIS AT89S8252 YANG DI INFORMASIKAN VIA SMS

Disusun Oleh : Moh. Mufid Murtadlo, 02.17.144
Dibawah Bimbingan : Dosen Pembimbing I – Ir. F. Yudi Limpraptono, MT.
Dosen Pembimbing II – Ir. Mimien Mustikawati.

Kata Kunci : *Indukan Ayam Pedaging, Kipas, Lampu Pemanas, PWM, SMS.*

Alat kontrol suhu indukan ayam pedaging ini dibuat untuk memberikan kemudahan kepada peternak dalam menjaga temperatur induk buatan agar kebutuhan panas anak ayam pedaging dapat dipenuhi. Selain itu juga peternak tidak memerlukan banyak waktu dan tenaga untuk mengetahui kondisi suhu dari induk buatan.

Prinsip kerja dari alat ini adalah ketika induk buatan kita berikan setpoint suhu untuk minggu pertama yaitu 35 °C, suhu tersebut akan dipertahankan selama 7 hari. Ketika memasuki hari ke delapan, secara otomatis setpoint suhu akan diturunkan menjadi 32,2 °C yang mana suhu tersebut adalah suhu setpoint untuk minggu kedua. Suhu tersebut akan bertahan selama 7 hari dan pada hari ke delapan, maka suhu diturunkan menjadi 29,4 °C dan dipertahankan selama 7 hari. Kipas akan bekerja ketika suhu melebihi setpoint, dan kipas ini sendiri hanya bersifat ON / OFF saja.

Fasilitas SMS dalam sistem ini yaitu ketika tegangan jala-jala 220V putus atau lampu pemanas mati, maka alat akan mengirimkan SMS berupa SMS Peringatan kepada Ponsel Peternak. Selain itu, peternak juga bisa sewaktu-waktu memeriksa suhu induk buatan dengan mengirimkan SMS ke Ponsel Sistem dan secara otomatis Sistem akan langsung membalas dengan mengirimkan suhu aktual induk buatan pada waktu itu juga. Dari pengujian alat secara keseluruhan, dengan pengujian sebanyak 10 kali tidak terdapat kesalahan atau error = 0% pada pengujian SMS peringatan dan terdapat kesalahan relatif sebesar 0,166 % pada 15 kali pengujian SMS pengecekan suhu. Sedangkan pada pengujian rangkaian *Driver* Kipas, disimpulkan bahwa kondisi kipas sudah sesuai dengan perencanaan yaitu kipas akan berputar atau ON pada saat suhu melebihi Setpoint. Dengan melakukan pengujian sebanyak 9 kali, tidak ditemukan kesalahan atau error = 0%.

KATA PENGANTAR

Alhamdulillah, puji syukur kehadirat Allah Subhanahu Wa Ta'ala yang senantiasa mencurahkan Rahmad, Hidayah, Ridlo dan Inayah-Nya, serta anugerah kesahatan kepada penyusun sehingga skripsi yang berjudul "Perencanaan Dan Pembuatan Alat Kontrol Suhu Pada Indukan Ayam Pedaging Berbasis AT89S8252 Yang Di Informasikan Via SMS" ini dapat diselesaikan. Skripsi ini merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik Elektro Strata Satu (S-1) Institut Teknologi Nasional Malang.

Penyelesaian skripsi dan penyusunan laporan ini tidak lepas dari saran, dukungan serta bantuan dari berbagai pihak untuk mencapai kesempurnaan. Tapi penyusun menyadari, kesempurnaan hanya Milik Allah SWT dan kita selaku makhluk hanya bisa berbuat yang terbaik semampu kita. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Bapak Ir. F. Yudi Limpraptono, MT., selaku Ketua Jurusan Teknik Elektro S-1 sekaligus Dosen Pembimbing 1.
2. Ibu Ir. Mimien Mustikawati selaku Dosen Pembimbing 2.
3. Bapak, Ibu Dosen serta seluruh Staf Jurusan Teknik Elektro S-1.
4. Bapak Moh. Mukhtar dan Ibu Salimah serta keluarga yang telah memberikan do'a restu, dorongan, semangat, dan biaya.
5. Rekan Instruktur di Seluruh Laboratorium Jurusan Elektronika, dan
6. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Harapan penyusun semoga laporan skripsi ini memberikan manfaat bagi penyusun dan pembaca.

Malang, Maret 2007

Penyusun

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR GRAFIK

BAB I

PENDAHULUAN

1.1. Latar Belakang

Seiring dengan perkembangan teknologi di segala bidang, teknologi elektronika juga mengalami kemajuan yang sangat pesat sehingga hampir setiap peralatan yang digunakan manusia tidak lepas dari penggunaan berbagai macam peralatan elektronika yang ada. Salah satu bidang teknologi yang mengalami perkembangan lebih pesat adalah teknologi kontrol otomatis yang berkembang pesat seiring perkembangan jaman. Telah banyak berbagai kontrol otomatis yang dibuat dan diaplikasikan untuk keperluan industri, layanan umum maupun untuk penelitian.

Bidang peternakan tak lepas dari ide untuk melakukan penelitian yang dinilai banyak sekali peralatan-peralatan yang perlu dikembangkan untuk memudahkan aktifitas pemeliharaan maupun perawatan. Dalam hal ini topik yang diamati adalah indukan ayam pedaging, berfungsi untuk menjaga anak ayam pedaging tetap dalam suhu yang dibutuhkan.

Berdasarkan pengamatan yang ada dilapangan, pengecekan suhu indukan ayam pedaging masih dilakukan dengan manual yaitu dengan mengamati alat pengukur suhu *(termometer)*. Hal tersebut dinilai kurang efisien karena si peternak harus berada di kandang indukan ayam untuk mengetahui keadaan suhu, hal ini pasti akan menyita waktu banyak bagi peternak akibat melakukan pengecekan secara kontinyu.

1.2. Rumusan Masalah

1. Bagaimana merancang dan membuat alat yang dapat mengontrol suhu indukan ayam pedaging sesuai kebutuhan.
2. Bagaimana cara mengirimkan informasi kepada peternak via sms *(short message service)* untuk mengetahui suhu indukan anak ayam pedaging.

1.3. Tujuan

Penulisan skripsi ini bertujuan untuk merencanakan dan membuat alat kontrol suhu indukan ayam pedaging yang dilengkapi sistem informasi via sms *(short message service)*.

1.4. Batasan Masalah

Berdasarkan permasalahan yang ada, maka bahasan yang dibuat dibatasi pada hal-hal berikut:

1. Setpoint Suhu indukan adalah 35 °C, 32,2 °C dan 29,4 °C.
2. Tidak membahas kelembaban, kondisi dan jumlah anak ayam.
3. Sumber panas menggunakan lampu pijar dengan sumber tegangan dc.
4. Tidak membahas karakteristik Handphone.
5. *SMS* perintah pengecekan suhu yang di terima sistem hanya karakter "cek".
6. Yang di informasikan dari sistem hanya suhu.
7. Indukan ayam pedaging hanya bersifat miniatur atau model.

1.5. Metodologi

Metodologi yang dipakai dalam pembuatan skripsi ini adalah:

1. Studi Literatur

   Dengan mencari referensi-referensi yang berhubungan dengan perencanaan dan pembuatan alat yang akan dibuat.

2. *Field Research*

   Dengan melakukan penelitian secara langsung mengenai objek-objek yang berhubungan langsung dengan perencanaan alat yang akan dibuat.

3. *Design* dan Pembuatan Alat

   Yaitu meliputi pembuatan PCB, perakitan komponen serta penyolderan dan pembuatan perangkat lunak.

4. Pengujian Alat

   Dengan melakukan pengujian perblok rangkaian dan kerja seluruh sistem pada alat tersebut.

1.6. Sistematika Penulisan

Penulisan skripsi ini terbagi menjadi lima bab dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Membahas tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi dan sistematika penulisan pada penulisan skripsi ini.

BAB II LANDASAN TEORI

Berisikan tentang penjelasan dan teori-teori yang berhubungan dengan komponen-komponen yang digunakan dalan perancangan alat.

BAB III PERENCANAAN DAN PEMBUATAN ALAT

Membahas tentang perancangan alat yang terdiri dari perancangan perangkat keras dan perancangan perangkat lunak.

BAB IV PENGUJIAN ALAT

Membahas tentang pengujian peralatan secara keseluruhan dan analisa hasil pengujian.

BAB V PENUTUP

Berisikan kesimpulan yang didapat selama perancangan dan pembuatan alat serta saran-saran.

BAB II

LANDASAN TEORI

2.1. Pendahuluan

Untuk dapat memahami alat yang akan dirancang, maka dalam bab ini akan dijelaskan mengenai teori dasar yang akan berkaitan dengan sistem.

2.2. Mikrokontroler AT89S8252

2.2.1. Fitur Mikrokontroler AT89S8252

Perbedaan mendasar antara mikrokontroler dan mikroprosesor adalah mikrokontroler selain memiliki CPU juga dilengkapi memori dan input output yang merupakan kelengkapan sebagai sistem minimum mikrokomputer sehingga sebuah mikrokontroler dapat dikatakan sebagai mikrokomputer dalam keeping tunggal (*Single Chip Microcomputer*) yang dapat berdiri sendiri.

Mikrokontroler AT89S8252 adalah mikrokontroler ATMEL yang kompatibel penuh dengan Mikrokontroler keluarga MCS-51, membutuhkan daya rendah, memiliki performance yang tinggi dan merupakan mikrokomputer 8 bit yang dilengkapi dengan *8 Kilobyte Flash memori untuk program, 2 Kilobyte EEPROM (Electrical Eraseable and Programmable Read Only Memory) dan 256 Byte RAM internal.* Program memori yang dapat diprogram ulang dalam sistem atau menggunakan programmer nonvolative memori konvensional. Dalam sistem mikrokontroler terdapat dua hal mendasar yaitu: perangkat lunak dan perangkat keras yang keduanya saling terkait dan mendukung.

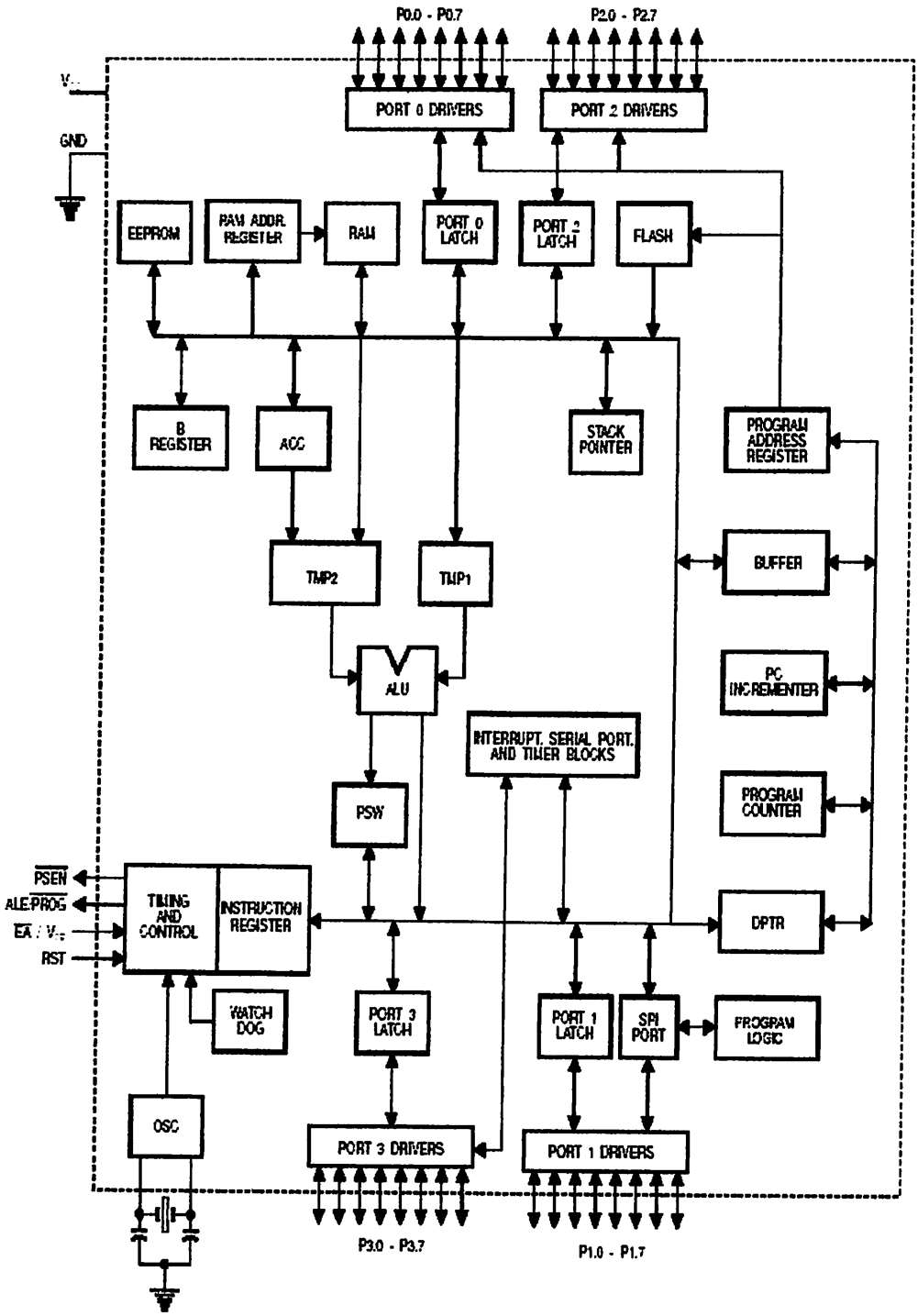
2.2.2. Arsitektur Mikrokontroler AT89S8252

Secara umum Mikrokontroler AT89S8252 memiliki :

- CPU 8 bit termasuk keluarga MCS-51
- 8 Kbyte *Flash* Memori
- 256 Byte *Internal* Memori
- 32 Port I/O, masing-masing terdiri atas 6 jalur I/O
- 3 *Timer/Counter* 16 Bit
- 2 *Serial Port Full Duplek*
- Kecepatan pelaksanaan instruksi per siklus 1 µS pada frekuensi *clock 12* Nhz
- 2 DPTR (*Data Pointer*)
- *Watchdog timer*
- Fleksibel ISP Programing

Dengan keistimewaan diatas pembuatan alat menggunakan AT89S8252 menjadi lebih sederhana dan tidak memerlukan IC pendukung yang banyak.

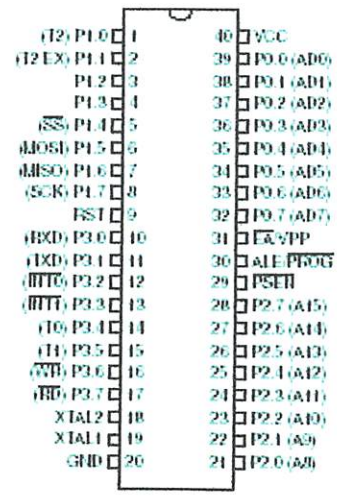
Adapun blok diagram dari Mikrokontroler AT89S8252 adalah sebagai berikut :



Gambar 2-1 Diagram Blok Mikrokontroler AT89S8252 [1]

2.2.3. Konfigurasi Pena-Pena Mikrokontroler AT89S8252

Mikrokontroler AT89S8252 terdiri dari 40 pin dengan konfigurasi sebagai berikut:



Gambar 2-2 Konfigurasi Pin-Pin AT89S8252[1]

Fungsi-fungsi tiap pinnya adalah sebagai berikut :

- VCC (Supply tegangan), pin 40
- GND (Ground) , pin 20
- Port 0, pin 32-39

  Merupakan port input-output dua arah, tanpa internal pull-up dan konfigurasi sebagai multipleks bus alamat rendah (A_0-A_7) dan data selain pengaksesan program memory dan data memory eksternal.
- Port 1, pin 1-8

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

- Port 2, pin 21-28

  Merupakan port input-output dengan internal pull-up. Mengeluarkan alamat tinggi selama pengambilan program memori eksternal.

- Port 3, pin 10-17

  Merupakan port input-output dengan internal pull-up, dimana port 3 juga memiliki fungsi khusus dan dapat dilihat pada table berikut :

Tabel 2-1 Fungsi Khusus Pada Port 3[1]

Nama Penyemat	Fungsi Khusus
Port 3.0	RxD (Port Masukan khusus)
Port 3.1	TxD (Port Keluaran Khusus)
Port 3.2	$\overline{INT0}$ (Masukan Interupsi Eksternal 0)
Port 3.3	$\overline{INT1}$ (Masukan Interupsi Eksternal 1)
Port 3.4	T0 (Masukan Pewaktu Eksternal 0)
Port 3.5	T1 (Masukan Pewaktu Eksternal 1)
Port 3.6	$\overline{WR}$ (Sinyal Tulis Memori Data Eksternal)
Port 3.7	$\overline{RD}$ (Sinyal Baca Memori Data Eksternal)

- RST (Reset), pin 9

  Input reset merupakan reset master untuk AT89S8252

- ALE /Prog (*Address Lacth Enable*), pin 30

  Digunakan untuk menahan alamat memori eksternal selama pelaksanaan intruksi.

- PSEN *(Program Store Enable)*, pin 29

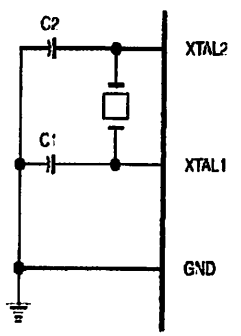
  Merupakan sinyal pengontrol yang memperbolehkan program memori eksternal masuk ke dalam bus.

- EA/VPP *(External Access)*, pin 31

  Dapat diberikan logika rendah (Ground) atau logika tinggi (+5 Volt). Jika diberikan logika tinggi maka mikrokontroler akan mengakses program dari ROM internal (EEPROM/Flash Memori), dan jika diberikan logika rendah maka mikrokontroler akan mengakses program dari memori eksternal.

- X-TAL 1 dan X-TAL 2,pin 18,19

  Pin ini dihubungkan dengan kristal bila menggunakan osilator internal. X-TAL 1 merupakan masukan ke rangkaian osilator internal sedangkan X-TAL 2 keluaran dari rangkaian osilator internal. Untuk keperluan ini diperlukan kapasitor penstabil sebesar 30 pF. Dan nilai dari X-TAL tersebut antara 4-24 Mhz. untuk lebih jelasnya dapat dilihat gambar pemasangan X-TAL serta kapasitor yang digunakannya.



Gambar 2-3 Osilator Eksternal AT89S8252[1]

2.2.4. Organisasi Memory

Organisasi memori pada mikrokontroler AT89s8252 dapat dibagi menjadi dua bagian besar yaitu memori program dan memori data. Pembagian tersebut didasarkan atas fungsi dari penyimpanan data maupun program. Memori program digunakan untuk menyimpan instruksi-instruksi yang akan dijalankan oleh mikrokontroler, sedangkan memori data digunakan sebagai tempat yang sedang diolah mikrokontroler.

Program mikrokontroler disimpan dalam memori program berupa ROM. Mikrokontroler AT89S8252 dilengkapi dengan ROM internal, sehingga untuk menyimpan program tidak digunakan ROM eksternal yang terpisah dari mikrokontroler. Agar tidak menggunakan memori program eksternal, penyemat $\overline{EA}$ dihubungkan dengan Vcc (logika 1).

Memori program mikrokontroler menggunakan alamat 16 Bit mulai dari 0000_H – $0FFF_H$ sehingga kapasitas penyimpanan program maksimal adalah 4Kbyte. Sinyal $\overline{PSEN}$ (*Program Store Enable)* tidak digunakan jika menggunakan memori program internal.

Selain program mikrokontroler AT89s8252 juga memiliki data internal sebesar 128 Byte dan mampu mengakses memori data eksternal sebear 64 Kbyte. Semua memori data internal dapat dialamati dengan data langsung atau tidak langsung. Ciri dari pengalamatan langsung adalah *operand* adalah alamat register yang berisi alamat data yang akan diolah. Sebagian memori tersebut dapat dialamati dengan pengalamatan register, dan sebagian lagi dapat dialamati dengan memori satu bit. Untuk membaca data digunakan sinyal $\overline{RD}$, sedangkan untuk menulis digunakan sinyal $\overline{WR}$.

2.2.5. SFR *(Special Fuction Register)*

Register Fungsi Khusus *(Special Fuction Register)* terletak pada 128 byte bagian atas memori data internal dan berisi register-register untuk pelayanan latch port, *timer*, *program status word* (PSW), control peripheral dan sebagainya. Alamat register fungsi khusus ini ditunjukkan pada Gambar 2-4 berikut.

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									CDF H
0D0H	PSW 00000000					SPCR 000001XX			0D7H
0C8H	T2CON 00000000	T2MOD XXXXXXXX	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			CCF H
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX						0AFH
0A0H	P2 11111111								0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111						WDCON 00000010		97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DPOL 00000000	DPOH 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

Gambar 2-4 Peta *Special Fuction Register* AT89S8252[1]

Beberapa macam register fungsi khusus yang sering digunakan adalah sebagai berikut :

- *Accumulator* (ACC) merupakan register untuk penambahan dan pengurangan, perintah Mneumonic untuk mengakses akumulator disederhanakan sebagai A.

- *Register B* merupakan register khusus yang berfungsi melayani operasi perkalian dan pembagian.
- *Stack Pointer* (SP) merupakan register 8 bit yang dapat diletakkan di alamat manapun pada RAM internal.
- *2 Data Pointer* (DPTR) terdiri atas dua register yaitu untuk byte tinggi (*data pointer high*, DPH) dan byte rendah (*data Pointer Low*, DPL) yang berfungsi untuk mengunci alamat 16 bit.
- *Port 0 sampai port 3* merupakan register yang berfungsi untuk membaca dan mengeluarkan data pada port 0,1,2,3. masing-masing register ini dapat dialamati per-byte maupun per-bit.
- *Control Register* terdiri dari register yang mempunyai fungsi control. Untuk mengontrol system interupsi, terdapat dua register khusus yaitu register IP (*Interrupt Priority*) dan register IE *(Interupt Enable*). Untuk mengontrol pelayanan timer/counter terdapat register khusus yaitu register TCON (*Timer/Counter Control*) serta pelayanan port srial menggunakan register SCON (*Serial Port Control).*

2.2.6. Sistem Interupsi

Mikrokontroler AT898252 mempunyai 5 buah sumber interupsi yang dapat membangkitkan permintaan yaitu INT0, INT1, T1, T2 dan Port Serial.

Saat terjadinya interupsi mikrokontroler secara otomatis akan menuju ke subrutin pada alamat tersebut. Setelah interupsi selesai dikerjakan, mikrokontroler akan mengerjakan program semula. Tiap-tiap sumber interupsi dapat *enable* atau *disable* secara software.

Tingkat prioritas semua sumber interrupt dapat diprogram sendiri-sendiri dengan set atau clear bit pada (*interrupt priority).* Jika dua permintaan interupsi dengan tingkat prioritas yang berbeda diterima secara bersamaan akan dilakukan polling untuk menentukan mana yang akan dilayani.

Tabel 2-2 Alamat Sumber Interupsi [1]

Sumber Interupsi	Alamat Awal
Interupt Luar 0 (INT 0)	03_H
Pewaktu/Pencacah 0 (T0)	$0B_H$
Interrupt Luar 1 (INT 0)	13_H
Pewaktu/Pencacah 0 (T0)	$1B_H$
Port Serial	23_H

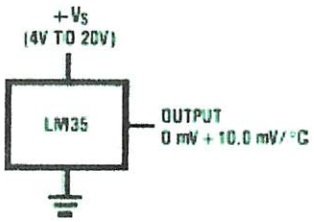
2.3. Sensor Suhu LM35

Sensor suhu yang dipakai adalah LM35 yang merupakan IC sensor temperatur yang presisi, yang mana outputnya linear dengan kenaikan temperatur dalam derajat celcius. LM35 juga tidak butuh kalibrasi lagi untuk memberikan keakuratan khusus pada temperatur ruang $\pm$ ¼ °C dan $\pm$ ¾ ° C lebih lebar yakni range temperatur antara -55 °C sampai +150 °C.

LM35 mempunyai impedansi output yang rendah, output yang linear. Sensor ini menggunakan power supply tunggal atau plus dan minus supply. Keuntungan pemakaian LM35 adalah:

1. Kalibrasi langsung dalam °C.
2. Mempunyai tegangan output linear 10 mV/°C.
3. Keakuratannya 0,5 °C pada temperatur 25 °C.

4. Menggunakan suplly dari 4 V sampai dengan 30 V.
5. Non linearitas hanya + 1/4 °C.
6. Mempunyai impedansi output yang rendah sebesar 0,1 untuk beban sebesar 1 mA.



Gambar 2-5 Sensor Suhu LM35 [2]

2.4. Sensor Cahaya / LDR

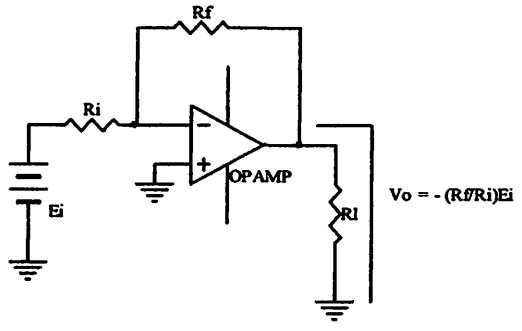
Sensor cahaya disini dipakai untuk mendeteksi keadaan lampu pijar, apakah lampu dalam keadaan hidup atau mati. Sedangkan komponen yang digunakan adalah LDR *(Light Dependent Resistor)*, yaitu resistor yang nilai resistansinya berubah-ubah tergantung pada cahaya yang diterimanya. Semakin terang cahaya yang diterimanya, maka resistansinya akan semakin kecil.

Gambar 2-6 Fitur LDR [3]

2.5. Operasional Amplifier

2.5.1. Penguatan Membalik / *Inverting Amplifier*

Penguat operasional dengan tegangan positif dihubungkan pada input pembalik.



Gambar 2-7 Rangkaian Op Amp *Inverting* dengan Ei dihubungkan pada Input Pembalik [4]

Tegangan Ei positif dihubungkan melalui tahanan masukan Ri kemasukan (-) dari Op-Amp. Umpan balik Negatif disusun oleh tahanan Rf. Tegangan antara input (+) dengan input (-) pada dasarnya nol (*virtual ground* atau *ground* semu). Karena ujung Ri yang satu berada pada *Ground* semu tersebut maka penurunan tegangan melalui Ri adalah Ei.

- Arus yang mengalir melalui Ri adalah:

$$I = \frac{Ei}{Ri} \quad \text{.....(2-9)}$$

- Untuk tegangan output Vo dapat dicari dengan rumus :

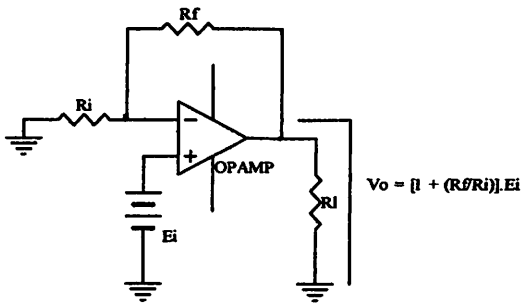
$$Vo = -Ei\frac{Rf}{Ri} \quad \text{.....(2-10)}$$

- Untuk penguatan tegangan *output* Vo adalah :

$$A = \frac{Vo}{Ei} = -\frac{Rf}{Ri} \quad \text{.....(2-11)}$$

2.5.2. Penguat Tak Membalik / *Non Inverting Amplifier*

Penguat Operasional dengan tegangan positif dihubungkan pada input tak membalik. Tegangan Vo memiliki Polaritas yang sama seperti masukan Ri. Tahanan masukan dari penguat pembalik adalah Ri.



Gambar 2-8 Rangkaian Op Amp *Non Inverting* dengan Ei dihubungkan pada Input Tak Pembalik [4]

Untuk tahanan masukan dari penguat tak membalik nilainya sangat besar, biasanya melebihi 100MΩ. Karena tegangan antara inputan (+) dan inputan (-) itu secara praktis 0, maka kedua inputan tersebut berada pada potensial Ei yang sama.

- Arus yang mengalir melalui Ri adalah:

$$I = \frac{Ei}{Ri} \quad \text{.....(2-12)}$$

- Untuk tegangan output Vo dapat dicari dengan rumus :

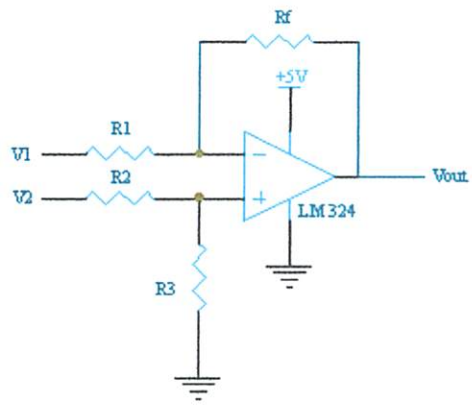
$$Vo = (1 + \frac{Rf}{Ri})\ Ei \quad \text{.....(2-13)}$$

- Untuk penguatan Tegangan output Vo adalah:

$$A = \frac{Vo}{Ei} = 1 + \frac{Rf}{Ri} \quad \text{.....(2-14)}$$

2.5.3. Penguat Differensial

Penguat differensial akan menghasilkan sinyal output yang sebanding dengan selisih pada kedua inputnya. Penguat differensial diperlihatkan pada gambar berikut:



Gambar 2-9 Penguat Differensial

Output dari rangkaian diatas dapat ditentukan dengan menggunakan prinsip superposisi. Dengan memasukkan V2 = 0, diperoleh:

$$Vout = -V1 * \left(\frac{Rf}{R1}\right)$$

Bila memasukkan V1 = 0, maka

$$Vout = \left(\frac{R3}{R2 + R3}\right) * V2\left(1\frac{Rf}{R1}\right)$$

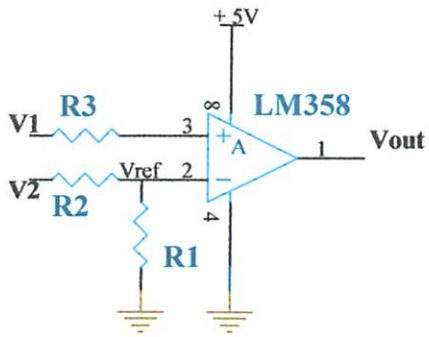
Output total Vout adalah penjumlahan kedua hasil diatas

$$Vout = -V1 * \frac{Rf}{R1} + \frac{R3}{R2 + R3} * V2\left(1\frac{Rf}{R1}\right)$$

Jika R1 = R2 = R3 = RF, maka rangkaian differensial akan berfungsi sebagai penguat selisih dengan tegangan Vout = V2 – V1.

2.5.4. Penanding / *Comparator*

Penerapan lain op-amp adalah sebagai pembanding atau *comparator*. Dalam penanding ini harus ada tegangan acuan atau V referensi, tegangan acuan ini dapat positif atau negatif.



Gambar 2-10 Penanding *(Comparator)* [5]

Dalam penerapannya, R3 boleh di tiadakan. Sedangkan untuk mengetahui besarnya nilai tegangan acuan atau Vreferensi (Vref), dapat di cari dengan menggunakan rumus pembagi tegangan sebagai berikut:

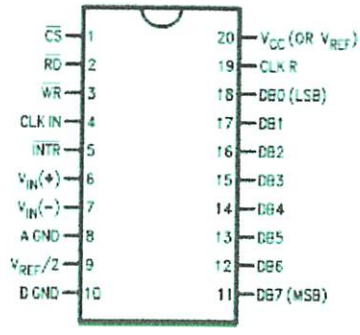
$$Vref = \frac{R1}{R1 + R2} * V2$$

Sehingga akan didapatkan Vreferensi sesuai dengan yang kita inginkan. Atau bisa juga kita menggunakan variabel resistor sebagai pengganti dari rangkaian R1 dan R2.

2.6. ADC *(Analog to Digital Converter)* 0804

Untuk dapat mengukur atau mengolah suatu variabel fisik (analog) dengan piranti digital, maka variabel tersebut harus diubah dulu menjadi variabel digital yang nilainya proporsional dengan nilai variabel yang akan diukur atau diolah tersebut. Konversi ini dilakukan oleh konverter analog ke digital (*Analog to Digital Converter*, ADC). Resolusi ADC mengacu pada jumlah bit dalam keluaran biner ADC. Spesifikasi penting lain untuk ADC adalah waktu konversi *(convertion time)*. Waktu konversi ADC adalah waktu yang diperlukan ADC untuk menghasilkan kode biner yang sahih (valid) utnuk tegangan masukan yang diberikan. Sebuah konverter disebut berkecepatan tinggi jika memiliki waktu konversi yang pendek.

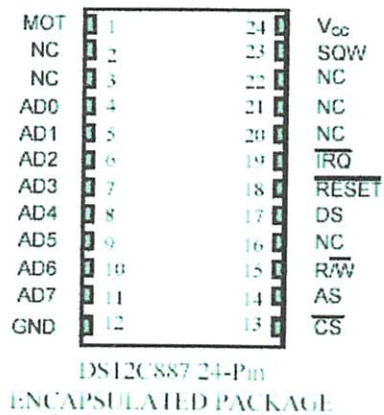
Pada perancangan alat ini menggunakan AC 0804. ADC ini merupakan jenis IC dengan keluaran 8 bit, yang mempunyai keluaran biner dan menampilkan waktu pengubah singkat. ADC 0804 ini beroperasi pada tegangan +5 Volt yang dapat mengkode tegangan analog masukan berkisar 0 – 5 Volt.



Gambar 2-11 Konfigurasi Pin ADC 0804 [6]

2.7. Serial RTC *(Real Time Clock)* DS12C887

RTC DS12887 adalah IC yang dipakai sebagai basis pewaktuan buatan Dallas Semikonduktor. Terdiri dari 24 pin dengan konfigurasi seperti ditunjukkan pada gambar berikut:



Gambar 2-12 IC Konfigurasi Pin DS12C887 [7]

Adapun karakteristik dari RTC DS12C887 antara lain:

- Osilator internal dan time base internal
- Menghitung detik, menit, jam, hari seminggu, tanggal bulan dan tahun dengan lompatan pergantian tahun
- Data tidak akan hilang maksimal sampai 10 tahun meskipun tidak ada daya.
- Data waktu kalender atau alarm dapat dipresentasikan dalam format biner/BCD
- Clock 12 atau 24 dengan AM dan PM dalam mode 12 jam-an.
- Interface dengan software lokasi RAM 64 byte
  - 14 byte clock dan register kontrol
  - 59 byte RAM untuk tujuan umum

Secara keseluruhan, fungsi pin-pin RTC DS12C887 adalah sebagai berikut:

- **GND dan VCC**

  Merupakan pin catu daya, VCC dihubungkan pada catu daya +5 volt dan GND pada ground

- **MOT – Mode Selection**

  Digunakan sebagai saklar pemilih mode diagram pewaktuan. Apabila dihubungkan pada VCC berarti sistem diagram pewaktuan Motorolla yang dipakai, jika dihubungkan dengan GND berarti sistem diagram pewaktuan Intel yang dipakai.

- **SQW – Square Wave Output**

  Merupakan pin yang dapat dipilih untuk mengeluarkan satu dari 13 frekwensi keluaran yang tersedia. Besar frekwensi keluaran SQW dapat diubah dengan diprogram pada register A. untuk mengaktifkan atau menonaktifkan sinyal SQW dipilih lewat bit SQWE pada registerB.

- **AD0-AD7 – Multiplexed Bidirectional Address / Data Bus**

  Merupakan bus alamat / data dua arah yang termultipleks. Pengiriman data maupun alamat dari maupun ke RTC dilakukan melalui bus ini.

- **AS – Address Strobe Input**

  Merupakan masukan bagi sinyal yang digunakan untuk memisahkan bus data dan bus alamat (ALE). Tepi turun AS / ALE akan menyebabkan alamat ditahan dalam RTC DS12C887.

Tepi naik berikutnya yang terjadi pada AS akan meniadakan alamat tersebut tanpa memperhatikan apakah penyemat CS aktif atau tidak.

- **DS – Data Strobe Or Read Input**

  Pin DS dapat diartikan sama seperti sinyal Output enable (OE) pada komponen memori. Dihubungkan dengan sinyal RD yang berasal dari mikrokontroller untuk melakukan proses membaca data pada RAM internal RTC.

- **R/W – Read/Write Input**

  Pin R/W dapat diartikan sama seperti sinyal Write Enable (WE) yang biasa dipakai pada komponan memori jenis RAM. Digunakan untuk proses penulisan data pada RAM internal RTC.

- **CS – Chip Select Input**

  Merupakan masukan untuk mengaktifkan periferal RTC. Sinyal CS didapat dari dekoder alamat dengan alamat tertentu.

- **IRQ – Interrupt Request Output**

  Sinyal IRQ merupakan sinyal aktif rendah, yang dapat digunakan untuk menginterupsi mikrokontroller. Keluaran IRQ tetap rendah selama status bit yang menyebabkan interupsi ada. Untuk mereset IRQ, mikrokontroller memerikan program pada register C RTC. Saat tidak terdapat interupsi, pin ini dalam kondisi impedansi tinggi (high impedance).

- **RESET – Reset Input**

  Sinyal RESET diberikan dengan memberikan logika rendah selama waktu yang dispesifikasikan, dan tidak berpengaruh pada unjuk kerja clock, kalender, dan fungsi RAM. Namun sinyal RESET dapat mengakibatkan beberapa flag direset menjadi nol.

Untuk menginisialisasi RTC, yaitu men-set waktu yang ada di RAM RTC saat pertama kali RTC diaktifkan, digunakan sinyal WR (Write) yang dihubungkan ke pin R/W, data yang diinginkan pada alamat yang bersesuaian. Sedangkan untuk membaca data dari RAM internal RTC digunakan sinyal RD (Read) yang dihubungkan pada penyemat DS (Data Strobe). Operasi pembacaan dan penulisan pada RAM internal RTC sama seperti operasi baca dan tulis pada komponen memori jenis RAM.

2.8. LCD *(Liquid Crystal Display)* M1632

LCD adalah kependekan dari *"Liquid Crystal Display"*. LCD terdiri atas tumpukan tipis atau sel dari dua lembar kaca dengan pinggiran tertutup rapat. Antara dua lembar kaca tersebut diberi bahan kristal cair *(liquid crystal),* yang tembus cahaya. Permukaan luar dari masing-masing keping kaca mempunyai lapisan penghantar tembus cahaya seperti Oxida Timah *(tin oxide)*. Sel mempunyai ketebalan sekitar 1×10^{-5} meter dan diisi dengan kristal cair.

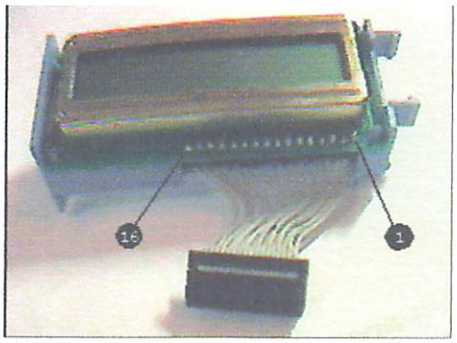
LCD memiliki keistimewaan dibandingkan tampilan yang lain seperti *sevent segment* yaitu kemampuan utnuk menampilakan karakter dan sebagai macam simbol. Salah satu jenis LCD diantaranya adalah LCD tipe dot matrik 5x7, tersusun sebanyak dua baris dan masing-masing baris terdiri atas 16 karakter.

Setiap karakter dibentuk oleh 5x7 buah titik. Sehingga jenis huruf yang mampu ditampilkan akan lebih banyak dan lebih baik dibandingkan dengan penampilan *sevent segment* atau lainnya.

Salah satu jenis LCD adalah LCD M1632 buatan Seiko Instrument, suatu jenis piranti dengan konsumsi daya rendah, disusun dari dot matrik dan dikontrol oleh ROM atau RAM generetor karakter dan RAM data display. Pengontrolan utamanya adalah pada ROM generator dan display RAM yang menghasilkan kode ASCII jika padanya diberikan input ASCII.

Adapun karakteristik dari LCD M1632 antara lain:

- Dengan 16x2 karakter dalam bentuk dot matrik 5x7 + cursor
- *Duty ratio* 1/16
- Memiliki ROM generator karakter dengan 192 tipe karakter
- 80x8 bit RAM data display
- Dapat diinterfacekan dengan MPU 8 bit/4 bit
- RAM data display dan RAM generator yang dapat dibaca oleh MPU
- +5 V single power supply
- Power on Reset
- Temperatur operasi 0°C sampai dengan 50°C



Gambar 2-13 Pin Out LCD M1632 Standard

Definisi pin yang terdapat pada LCD M1632 dijelaskan pada tabel 2-4 berikut :

Tabel 2-3 Definisi Pin LCD Module M1632 [8]

No. Pin	Simbol	Level	Fungsi	
1	Vss	-	Power Supply	0 V (GND)
2	Vcc	-		5 V ± 10%
3	Vee	-		For LCD Drive
4	RS	H/L	Sinyal Seleksi Register H; Data input [register data (write/read)] L; Instruction input [register instruksi (write), busy flag dan address counter (read)]	
5	R/W	H/L	H; Read L; Write	
6	E	H	Enable signal [sinyal penanda mulai operasi, aktif saat operasi write atau read]	
7	DB0	H/L	4 bit bus data upper 2 arah, dapat dibaca atau ditulis terhadap mikrokontroler	
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L	4 bit bus data upper 2 arah, dapat dibaca atau ditulis terhadap mikrokontroler, DB7 juga sebagai busy flag	
12	DB5	H/L		
13	DB6	H/L		
14	DB7	H/L		
15	V+BL	-	Back Light Supply	4 – 4,2 V 50 – 200 mA
16	V-BL	-		0 V (GND)

2.9. Transistor

Prinsip kerja transistor adalah arus bias base-emiter yang kecil mengatur besar arus kolektor-emiter.

Bagian penting berikutnya adalah bagaimana caranya memberi arus bias yang tepat sehingga transistor dapat bekerja optimal.

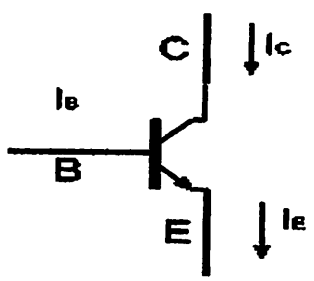
2.9.1. Arus Bias

Ada tiga cara yang umum untuk memberi arus bias pada transistor, yaitu rangkaian *CE (Common Emitter), CC (Common Collector)* dan *CB (Common Base)*. Namun dalam hal ini akan lebih detail dijelaskan bias transistor rangkaian CE. Dengan menganalisa rangkaian CE akan dapat diketahui beberapa parameter penting dan berguna terutama untuk memilih transistor yang tepat untuk berbagai aplikasi.

2.9.2. Arus *Emiter*

Dari hukum Kirchhoff diketahui bahwa jumlah arus yang masuk ke satu titik akan sama jumlahnya dengan arus yang keluar. Jika teorema tersebut diaplikasikan pada transistor, maka hukum itu menjelaskan hubungan :

$$I_E = I_C + I_B \qquad (2\text{-}2)$$



Gambar 2-14 Arus Emitor [9]

Persamanaan (2-2) tersebut mengatakan arus *emiter* I_E adalah jumlah dari arus kolektor I_C dengan arus base I_B. Karena arus I_B sangat kecil sekali atau disebutkan $I_B << I_C$, maka dapat dinyatakan :

$$I_E = I_C \quad \text{.....(2-3)}$$

2.9.3. *Alpha* (a)

Pada tabel data transistor (*databook*) sering dijumpai spesifikasi $_{dc}$ *(alpha dc)* yang tidak lain adalah :

$$a_{dc} = I_C/I_E \quad \text{.....(2-4)}$$

Definisinya adalah perbandingan arus kolektor terhadap arus emitor. Karena besar arus kolektor umumnya hampir sama dengan besar arus emiter maka idealnya besar a_{dc} adalah = 1 (satu). Namun umumnya transistor yang ada memiliki a_{dc} kurang lebih antara 0.95 sampai 0.99.

2.9.4. *Beta* (b)

Beta didefenisikan sebagai besar perbandingan antara arus kolektor dengan arus *base*.

$$b = I_C/I_B \quad \text{.....(2-5)}$$

Dengan kata lain, b adalah parameter yang menunjukkan kemampuan penguatan arus *(current gain)* dari suatu transistor. Parameter ini ada tertera di *data book* transistor dan sangat membantu para perancang rangkaian elektronika dalam merencanakan rangkaiannya.

Misalnya jika suatu transistor diketahui besar b = 250 dan diinginkan arus kolektor sebesar 10 mA, maka berapakah arus bias base yang diperlukan. Maka :

$$\mathbf{I_B = I_C/b = 10mA/250 = 40\ uA}$$

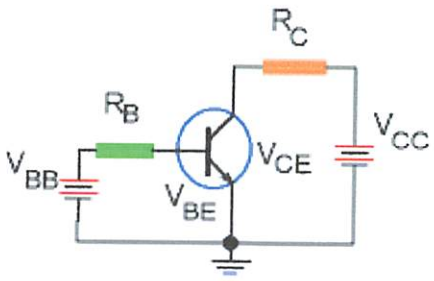
Arus yang terjadi pada kolektor transistor yang memiliki b = 200 jika diberi arus bias base sebesar 0.1 mA adalah :

$$\mathbf{I_C = b\ I_B = 200 \times 0.1mA = 20\ mA}$$

Dari rumusan ini lebih terlihat definisi penguatan arus transistor, yaitu sekali lagi, arus *base* yang kecil menjadi arus kolektor yang lebih besar.

2.9.5. *Common Emitter* (CE)

Rangkaian CE adalah rangkain yang paling sering digunakan untuk berbagai aplikasi yang mengunakan transistor. Dinamakan rangkaian CE, sebab titik *ground* atau titik tegangan 0 volt dihubungkan pada titik *emiter*.



Gambar 2-15 Rangkaian CE [9]

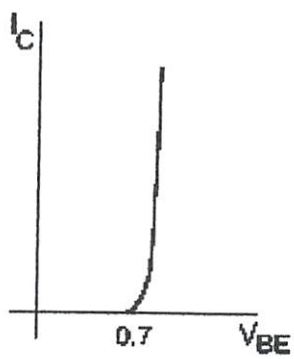
Sekilas tentang notasi, ada beberapa notasi yang sering digunakan untuk menunjukkan besar tegangan pada suatu titik maupun antar titik. Notasi dengan 1 *subscript* adalah untuk menunjukkan besar tegangan pada satu titik, misalnya V_C = tegangan kolektor, V_B = tegangan *base* dan V_E = tegangan *emiter.*

Ada juga notasi dengan 2 *subscript* yang dipakai untuk menunjukkan besar tegangan antar 2 titik, yang disebut juga dengan tegangan jepit. Diantaranya adalah :

- V_{CE} = tegangan jepit kolektor-emitor.
- V_{BE} = tegangan jepit *base*-emitor.
- V_{CB} = tegangan jepit kolektor-*base*.

Notasi seperti V_{BB}, V_{CC}, V_{EE} berturut-turut adalah besar sumber tegangan yang masuk ke titik base, kolektor dan emitor.

2.9.6. Kurva *Base*



Grafik 2-1 Kurva I_B -V_{BE} [9]

Hubungan antara IB dan VBE tentu saja akan berupa kurva dioda. Karena memang telah diketahui bahwa junction *base-emitor* tidak lain adalah sebuah dioda. Jika hukum Ohm diterapkan pada loop base diketahui adalah :

$$I_B = (V_{BB} - V_{BE}) / RB \quad (2\text{-}6)$$

V_{BE} adalah tegangan jepit dioda *junction base-emitor*. Arus hanya akan mengalir jika tegangan antara base-emitor lebih besar dari V_{BE}. Sehingga arus I_B mulai aktif mengalir pada saat nilai V_{BE} tertentu. Besar V_{BE} umumnya tercantum di dalam *databook*. Tetapi untuk penyerdehanaan umumnya diketahui V_{BE} = 0.7 volt untuk transistor silikon dan V_{BE} = 0.3 volt untuk transistor germanium.

Sampai disini akan sangat mudah mengetahui arus I_B dan arus I_C dari rangkaian berikut ini, jika diketahui besar b = 200. Katakanlah yang digunakan adalah transistor yang dibuat dari bahan silikon.

$$I_B = (V_{BB} - V_{BE}) / R_B$$

$$= (2V - 0.7V) / 100\ K$$

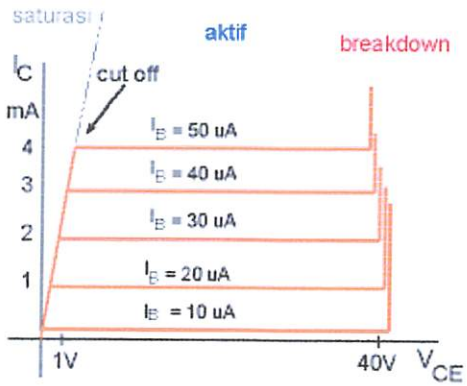
$$= 13\ uA$$

Dengan b = 200, maka arus kolektor adalah :

$$I_C = b\ I_B = 200 \times 13uA = 2.6\ mA$$

2.9.7. Kurva Kolektor

Sekarang sudah diketahui konsep arus base dan arus kolektor. Satu hal lain yang menarik adalah bagaimana hubungan antara arus base I_B, arus kolektor I_C dan tegangan kolektor-emiter V_{CE}. Pada grafik berikut telah diplot beberapa kurva kolektor arus I_C terhadap V_{CE} dimana arus I_B dibuat konstan.



Grafik 2-2 Kurva Kolektor [9]

Dari kurva ini terlihat ada beberapa *region* yang menunjukkan daerah kerja transistor. Pertama adalah daerah saturasi, lalu daerah *cut-off*, kemudian daerah aktif dan seterusnya daerah *breakdown*.

2.9.8. Daerah Aktif

Daerah kerja transistor yang normal adalah pada daerah aktif, dimana arus I_C konstan terhadap berapapun nilai V_{CE}. Dari kurva ini diperlihatkan bahwa arus I_C hanya tergantung dari besar arus I_B. Daerah kerja ini biasa juga disebut daerah linier (*linear region*).

Jika hukum Kirchhoff mengenai tegangan dan arus diterapkan pada loop kolektor (rangkaian CE), maka dapat diperoleh hubungan :

$$V_{CE} = V_{CC} - I_C R_C \quad (2\text{-}7)$$

Dapat dihitung disipasi daya transistor adalah :

$$P_D = V_{CE}\, I_C \quad (2\text{-}8)$$

Rumus ini mengatakan jumlah disipasi daya transistor adalah tegangan kolektor-emitor dikali jumlah arus yang melewatinya. Disipasi daya ini berupa panas yang menyebabkan naiknya temperatur transistor. Umumnya untuk transistor power sangat perlu untuk mengetahui spesifikasi P_{Dmax}. Spesifikasi ini menunjukkan temperatur kerja maksimum yang diperbolehkan agar transistor masih bekerja normal. Sebab jika transistor bekerja melebihi kapasitas daya P_{Dmax}, maka transistor dapat rusak atau terbakar.

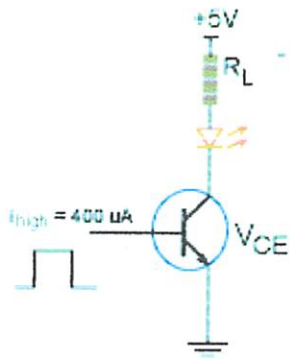
2.9.9. Daerah Saturasi

Daerah saturasi adalah mulai dari V_{CE} = 0 volt sampai kira-kira 0.7 volt (transistor silikon), yaitu akibat dari efek dioda kolektor-*base* yang mana tegangan V_{CE} belum mencukupi untuk dapat menyebabkan aliran elektron.

2.9.10. Daerah *Cut-Off*

Jika kemudian tegangan V_{CC} dinaikkan perlahan-lahan, sampai tegangan VCE tertentu tiba-tiba arus IC mulai konstan. Pada saat perubahan ini, daerah kerja transistor berada pada daerah *cut-off* yaitu dari keadaan saturasi *(OFF)* lalu menjadi aktif *(ON)*. Perubahan ini dipakai pada system digital yang hanya

mengenal angka biner 1 dan 0 yang tidak lain dapat direpresentasikan oleh status transistor *OFF* dan *ON*.



Gambar 2-16 Rangkaian *Driver* LED [9]

Misalkan pada rangkaian *driver* LED di atas, transistor yang digunakan adalah transistor dengan b = 50. Penyalaan LED diatur oleh sebuah gerbang logika (*logic gate)* dengan arus *output high* = 400 uA dan diketahui tegangan *forward* LED, V_{LED} = 2.4 volt. Kemudian pertanyaannya adalah, berapakah seharusnya resistansi R_L yang dipakai.

$$\mathbf{I_C = bI_B = 50 \times 400\ uA = 20\ mA}$$

Arus sebesar ini cukup untuk menyalakan LED pada saat transistor *cut-off*. Tegangan VCE pada saat *cut-off* idealnya = 0, dan aproksimasi ini sudah cukup untuk rangkaian ini.

$$\begin{aligned} R_L &= (V_{CC} - V_{LED} - V_{CE}) / I_C \\ &= (5 - 2.4 - 0)V / 20\ mA \\ &= 2.6V / 20\ mA \\ &= 130\ Ohm \end{aligned}$$

2.9.11. Daerah *Breakdown*

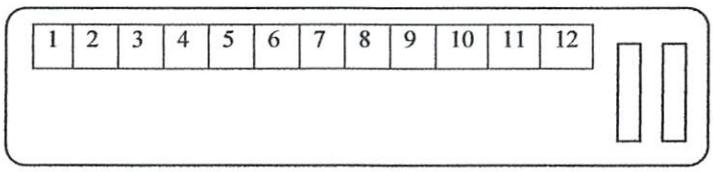
Dari kurva kolektor, terlihat jika tegangan V_{CE} lebih dari 40V, arus I_C menanjak naik dengan cepat. Transistor pada daerah ini disebut berada pada daerah *breakdown.* Seharusnya transistor tidak boleh bekerja pada daerah ini, karena akan dapat merusak transistor tersebut. Untuk berbagai jenis transistor nilai tegangan V_{CEmax} yang diperbolehkan sebelum *breakdown* bervariasi. V_{CEmax} pada *data book* transistor selalu dicantumkan juga.

2.10. Ponsel

Ponsel merupakan salah satu alat digital yang mempunyai kemampuan komunikasi yang tinggi dibanding dengan yang lain terutama untuk komunikasi data. Disamping itu pesatnya perkembangan teknologi saat ini menyebabkan ponsel tidak hanya berkomunikasi sesama ponsel saja melainkan dengan pengontrol lain seperti mikrokontroler.

2.10.1. Kabel Data

Kabel data adalah perangkat keras yang merupakan sarana penghubung antara ponsel dengan mikrokontroller agar bisa berkomunikasi. Kabel data untuk ponsel merk siemens merupakan komunikasi data serial asinkron yang memakai dua jalur komunikasi, yaitu RX dan TX. Kecepatan data untuk kirim maupun terima adalah 19200 bps.



Gambar 2-17 Konektor Ponsel Siemens C35 [10]

Koneksi ponsel / handphone Siemens pada dasarnya dapat menggunakan rangkaian yang sama dengan skematik rangkaian pada gambar 2-16 diatas. Namun perbedaannya adalah pada perangkat lunak yang berkomunikasi dengan ponsel Siemens tersebut.

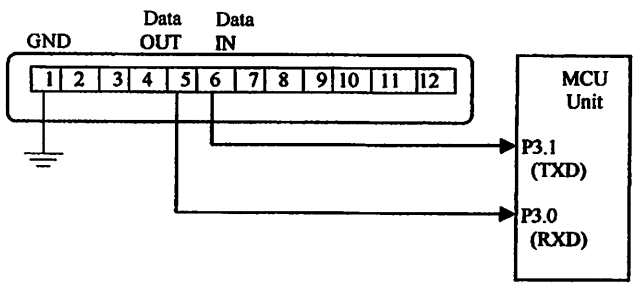
Untuk penggunaan masing-masing pin dari konektor diatas dapat dilihat pada tabel 2-4;

Tabel 2-4 Pinout Konektor Ponsel Siemens C35 [10]

Pin	Nama	Fungsi	In/Out
1	GND	Ground	
2	SELF SRVICE	Recognition/control battery charger	In/Out
3	LOAD	Charging Voltage	In
4	BATTERY	Battery	Out
5	DATA OUT	Data Sent	Out
6	DATA IN	Data Received	In
7	Z_CLK	Recognation/control accecories	
8	Z_DATA	Recognation/control accecories	
9	MICG	Ground for microphone	In
10	MIC	Microphone input	
11	AUD	Loudspeaker	Out
12	AUDG	Ground for eksternal speaker	

2.10.2. Interface Ponsel dengan Mikrokontroller

Mikrokontroller AT89S8252 dilengkapi dengan port serial. Port serial memungkinkan untuk interface dengan hardware lain dalam format serial. Komunikasi serial antara mikrokontroller AT89S8252 dengan ponsel hanya tinggal menghubungkan pin-pin serial dari masing-masing hardware. Pin Tx dari mikrokontroller akan dihubungkan dengan pin Rx yang ada di ponsel, sedangkan pin Rx yang ada di mikrokontroller dihubungkan dengan Tx di ponsel, pin ground dari mikrokontroller dihubungkan dengan pin ground dari ponsel. Gambar dibawah menunjukkan hubungan antara ponsel dengan mikrokontroller.



Gambar 2-18 Interface antara Ponsel dengan Mikrokontroller

2.11. Format Data SMS pada Siemens

Data yang mengalir ke atau dari SMS-Centre harus berbentuk PDU (Protokol Data Unit), PDU berisi bilangan-bilangan heksadesimal yang mencerminkan bahasa input output (I/O). PDU terdiri atas beberapa Header. Header untuk kirim SMS ke SMS-Centre berbeda dengan SMS terima dari SMS-Centre.

2.11.1. PDU untuk kirim SMS ke SMS-Centre

Delapan header untuk kirim SMS [11]

No SMS-Centre	Tipe SMS	No Referensi SMS	No Penerima	Bentuk SMS	Skema Enkoding	Waktu Validitas	Isi SMS

PDU untuk mengirim SMS terdiri atas delapan header, sebagai berikut:

1. **Nomor SMS-Centre**

   Header pertama ini terbagi atas tiga subheader, taitu:

   a. Jumlah pasangan Heksadesimal SMS-Centre dalam bilangan heksa.

   b. Nasional/Internasional Code.

   - Untuk Nasional, kode subheader-nya yaitu 81
   - Untuk Internasional, kode subheader-nya yaitu 91

   c. No SMS-Centre-nya sendiri, dalam pasangan heksa bibalik-balik. Jika tertinggal satu angka heksa yang tidak memiliki pasangan, angka tersebut akan dipasangkan dengan huruf F di depannya.

   Contoh: untuk nomor SMS-Centre Excelcom dapat ditulis dengan dua cara sebagai berikut:

   Cara 1:

   0818445009 diubah menjadi:

   a. 06 ⟶ ada 6 pasang

   b. 81 ⟶ 1 pasang

   c. 80-81-44-05-90 ⟶ 5 pasang

   Digabung menjadi: **<u>06818081440590</u>**

Cara 2:

62818445009 diubah menjadi:

a. 07 ⟶ ada 7 pasang

b. 91 ⟶ 1 pasang

c. 26-18-48-54-00-F9 ⟶ 6 pasang

Digabung menjadi: **07912618485400F9**

Berikut beberapa nomor SMS-Centre operator seluler di Indonesia.

Cara 1:

No	Operator Seluler	SMS-Centre No	Kode PDU
1	Telkomsel	0811000000	06818011000000
2	Satelindo	0816125	0581806121F5
3	Excelcom	0818445009	06818081440590
4	Indosat-M3	0855000000	06818055000000

Cara 2:

No	Operator Seluler	SMS-Centre No	Kode PDU
1	Telkomsel	62811000000	07912618010000F0
2	Satelindo	62816125	0591281652
3	Excelcom	62818445009	07912618485400F9
4	Indosat-M3	62855000000	07912658050000F0

2. Tipe SMS

Untuk SEND tipe SMS = 1. jadi bilangan heksanya adalah **01**

3. Nomor Referensi SMS

Nomor referensi ini dibiarkan dolo 0, jadi bilangan heksanya adalah **00**. Nanti akan diberikan sebuah nomor referensi otomatis oleh ponsel/alat SMS-gateway.

4. **Nomor Ponsel Penerima**

Sama seperti cara menulis PDU Header untuk SMS-Centre, header ini juga terbagi atas tiga bagian sebagai berikut:

- Jumlah bilangan desimal nomor ponsel yang dituju dalam bilangan heksa.
- Nasional/Internasional Code
- Untuk nasional, kode subheader-nya: 81
- Untuk Internasional, kode subheader-nya: 91
- Nomor ponsel yang dituju, dalam pasangan heksa dibalik-balik. Jika tertinggal satu angka heksa yang tidak memiliki pasangan, angka tersebut dipasangkan dengan huruf F di depannya.

Contoh:

Untuk nomor ponsel yang dituju = 628129573337 dapat ditulis dengan dua cara sebagai berikut:

Cara 1: 08129573337

a. 0B ⟶ ada 11 angka

b. 91

c. 26-18-92-75-33-73

digabungkan menjadi: **0B818021593733F7**

Cara 2: 628129573337 diubah menjadi:

a. 0C ⟶ ada 12 angka

b. 91

c. 26-18-92-75-33-73

Digabung menjadi: **0C91261892753373**

5. **Bentuk SMS,** antara lain:

   0→ 00 → dikirim sebagai SMS

   1→ 01 → dikirim sebagai telex

   2→ 02 → dikirim sebagai fax

   Dalam hal ini, untuk mengirim dalam bentuk SMS tentu saja kita memakai **00**.

6. **Skema Encoding Data I/O**

   Ada dua skema, yaitu:

   a. Skema 7 bit ⟶ ditandai dengan angka 0⟶ 00

   b. Skema 8 bit ⟶ ditandai dengan angka lebih besar dari 0 ⟶ diubah ke heksa

   ]Kebanyakan ponsel/SMS Gateway yang ada dipasaran sekarang menggunakan skema 7 bit sehingga kita menggunakan kode **00**.

7. **Jangka Waktu Sebelum SMS Expired**

   Jika bagian ini di-skip, itu berarti kita tidak membatasi waktu berlakunya SMS. Sedangkan jika isi dengan suatu bilangan integer yang kemudian diubah ke pasangan heksa tertentu, bilangan yang kita berikan tersebut akan mewakili jumlah waktu validitas SMS tersebut.

   Rumus untuk menghitung jangka waktu validitas SMS adalah sebagai berikut:

Integer (INT)	Jangka Waktu Validitas SMS
0-143	(INT + 1) x 5 menit (berarti: 5 menit s/d 12 jam)
144-167	12 jam + ((INT – 143) x 30 menit)
168-196	(INT - 166) x 1 hari
197-255	(INT - 192) x 1 minggu

Agar SMS kita pasti terkirim sampai ke ponsel penerima, sebaiknya kita tidak memberikan batasan waktu validnya.

8. Isi SMS

Header ini terdiri atas dua subheader, yaitu:

a. Panjang isi (jumlah huruf dari isi)

   Misalnya: untuk kata "hello" ada 5 huruf **05**

b. Isi berupa pasangan bilangan heksa

   Untuk ponsel/SMS Gateway berskema encoding 7 bit, jika kita mengetikkan suatu huruf dari keypad-nya, berarti kita telah membuat 7 angka I/O berurutan.

Ada dua langkah yang harus kita lakukan untuk mengonversikan isi SMS, yaitu:

Langkah Pertama : mengubahnya menjadi kode 7 bit

Langkah kedua : mengubah kode 7 bit menjadi 8 bit, yang diwakili oleh pasangan heksa.

Contoh: untuk kata "hello"

Langkah pertama:

Bit	7	1
h	110	1000
e	110	0101
l	110	1100
l	110	1100
o	110	1111

Skema 7 bit tersebut diperlihatkan pada gambar berikut.

				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	**b3**	**B2**	**b1**		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	0	0	0	**0**	@	Δ	SP	0	-	P	¨	p
0	0	0	1	**1**			!	1	A	Q	a	q
0	0	1	0	**2**	S	Φ	“	2	B	R	b	r
0	0	1	1	**3**		Γ	#	3	C	S	c	s
0	1	0	0	**4**		Λ		4	D	T	d	t
0	1	0	1	**5**		Ω	%	5	E	U	e	u
0	1	1	0	**6**		Π	&	6	F	V	f	v
0	1	1	1	**7**		Ψ	‘	7	G	W	g	w
1	0	0	0	**8**		Σ	(	8	H	X	h	x
1	0	0	1	**9**		Θ	)	9	I	Y	i	y
1	0	1	0	**10**	LF	Ξ	*	:	J	Z	j	z
1	0	1	1	**11**			+	;	K	Ä	k	ä
1	1	0	0	**12**			,	<	L	Ö	l	ö
1	1	0	1	**13**	CR		-	=	M		m	
1	1	1	0	**14**		β	.	>	N	Ü	n	ü
1	1	1	1	**15**			/	?	O		o	

Gambar 2-19 Skema 7 Bit [11]

Langkah kedua:

E 8
h 1110 1000

3 2
e 0011 0010 1

9 B
l 1001 1011 00

F D
l 1111 1101 100

0 6
o 0000 0110 1111

Oleh karena total 7 bit x 5 huruf = 35 bit, sedangkan yang kita perlukan adalah 8 bit x 5 huruf = 40 bit, maka diperlukan 5 bit dummy yang di isi dengan bilangan 0.

Setiap 8 bit mewakili suatu pasangan heksa. Tiap 4 bit mewakili suatu angka heksa, tentu saja karena secara logika 24 = 16.

Dengan demikian kata "hello" hasil konversinya menjadi : **E8329BFD06**

2.11.2. PDU untuk SMS Terima dari SMS-Centre

Delapan header untuk SMS-Terima [11]

No SMS-Centre	Tipe SMS	No Pengirim	Bentuk SMS	Skema Encoding	Tanggal & waktu SMS	Waktu Validitas	Isi SMS

Kebanyakan Header dibawah ini telah dibahas sebelumnya, kecuali beberapa yang berbeda, dijelaskan dibawah ini:

- No SMS-Centre
- Tipe SMS untuk SMS-terima = 4 **04**
- Nomor ponsel pengirim

- Bentuk SMS
- Skema Encoding
- Tanggal dan waktu SMS di-stamp di SMS-Centre

  Diwakili oleh 12 bilangan heksa (6 pasangan) yang berarti: yy/mm/dd hh:mm:ss

  Contoh: 200702251280

  02/07/22 15:32:08

  22 Juli 2002 15:32:08 WIB
- Batas waktu validitas

  jika tidak dibatasi dilambangkan dengan **00**
- Isi SMS.

2.11.3. Membedah Kedelapan Header

Setelah mengupas satu demi satu header untuk SMS-Terima ini, maka untuk PDU di bawah ini:

07912618485400F901000C9126189275337300000052E8329BFD06

Dapat kita artikan sebagai berikut:

1. SMS tersebut dikirim lewat SMS-Centre: 62855000000
2. SMS tersebut merupakan SMS-Terima
3. SMS tersebut dikirim dari ponsel no. 628561013789
4. SMS tersebut diterima dalam bentuk SMS
5. SMS tersebut memiliki skema encoding 7 bit
6. SMS sampai di SMS-Centre pada tanggal: 22-07-02, pukul: 15:32:08 WIB
7. SMS tersebut tidak memiliki batas waktu valid
8. SMS tersebut isinya adalah "hello"

2.12. PWM *(Pulse Width Modulation)*

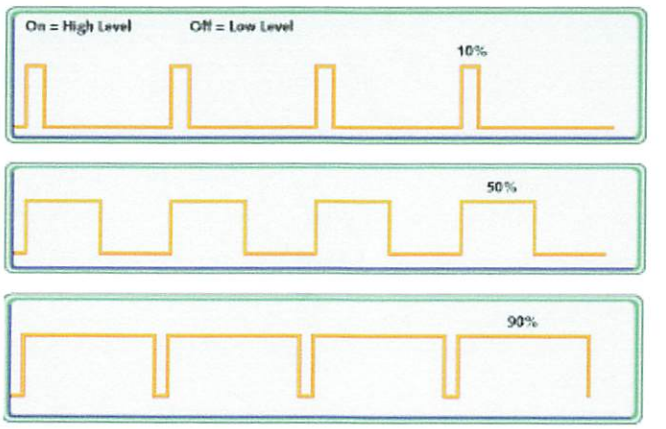
PWM biasanya digunakan untuk mungatur penggunaan daya pada beban (lampu DC) agar lebih efisien yaitu dengan cara pengalihan daya dengan menggunakan sistem lebar frekuensi, sehingga nyala DC dapat diatur.

Gambar 2-20 Rangkaian PWM sederhana [12]

Gambar 2-20 menunjukkan suatu sirkuit sederhana yang bisa digerakkan menggunakan PWM. Di gambar tersebut terdapat baterei 9V yang digunakan untuk menyalakan lampu. Jika kita menutup saklar menghubungkan lampu dan baterei selama 50 ms, lampu akan menerima 9V selama 50 ms. Jika kita membuka saklar selama 50 ms, lampu akan menerima 0V selama itu pula.

Jika kita mengulangi siklus ini 10 kali dalam satu detik, nyala lampu seolah-olah dihubungkan dengan sumber baterei 4.5V (50% dari 9 V). sehingga kita bisa katakan bahwa duty cyclenya adalah 50% dan frekwensi modulasinya adalah 10 Hz.

Gambar 2-21 dibawah ini menunjukkan tiga sinyal PWM yang berbeda, 10%, 50%, dan 90%.. Gambar 2-21a menunjukkan suatu PWM dengan keluaran duty cycle 10%. Dengan kata lain, dalam 1 periode sinyal hanya 10% yang ON/HIGH dan 90% OFF/LOW. Sebagai contoh, jika terdapat supply tegangan 9V dengan duty cycle sebesar 10%, maka sinyal analognya adalah 0.9V.



Gambar 2-21 Beberapa Contoh Sinyal PWM [12]

Duty cycle adalah istilah yang digunakan untuk menggambarkan pulsa output yang dinyatakan dalam persen (%). Persentase ini akan menunjukkan berapa besar siklus output pada saat tHigh.

Dengan perhitungan *duty cycle*;

$$DC = \frac{tHigh}{tLow + tHigh} * 100\%$$

Dimana semakin besar *duty cycle* maka semakin besar daya yang disalurkan ke beban.

2.13. Pengaturan Suhu pada Indukan Ayam Pedaging

2.13.1. Konstruksi Indukan Ayam Pedaging

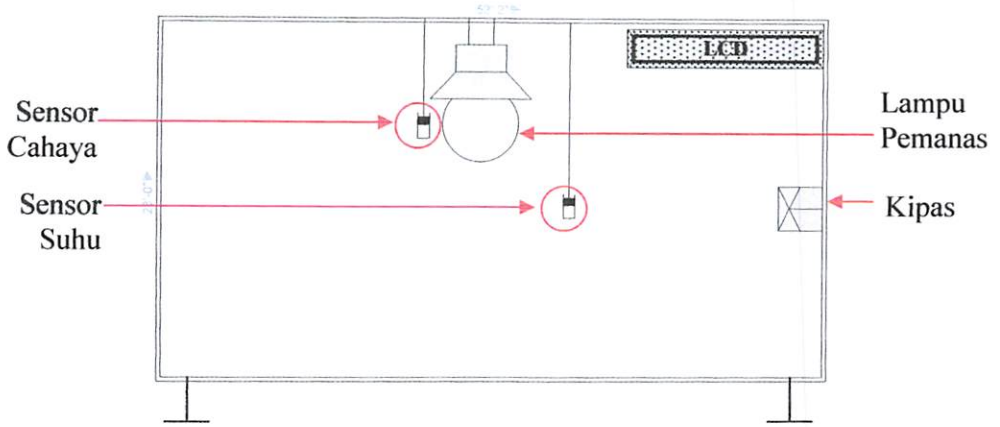
Untuk menjaga suhu indukan ayam tetap stabil, dibutuhkan pemanasan secara konstan. Pemasangan induk buatan ini untuk memenuhi kebutuhan suhu 35° C pada minggu pertama, apabila memungkinkan temperatur dapat diturunkan 5° F atau 2,8° C perminggunya.

Panas tambahan yang diberikan untuk anak ayam pedaging selama masa pertumbuhan awal tersebut dimaksudkan untuk mengimbangi suhu badan anak ayam. Apabila suhu indukan ayam tidak dinaikkan, maka akan terjadi penurunan suhu tubuh pada anak ayam yang secara langsung akan mempengaruhi aktifitas fungsi organ tubuh, ayam menjadi mudah sakit dan bahkan seringkali menyebabkan kematian. Karena itu pengaturan suhu sangat penting sehingga harus dilakukan setiap hari.

Bila suhu indukan tidak sesuai dengan kebutuhan, maka akan mengakibatkan:

- Bila suhu kurang dari kebutuhan, anak ayam akan kedinginan, masuk angin, tidak mau makan dan akhirnya menjadi sakit.

  Pada suhu rendah memudahkan virus, bakteri dan protozoa (parasit yang terdiri dari satu sel) untuk berkembang, dan dapat menyerang anak ayam sehingga menyebabkan kematian atau mempunyai pengaruh besar terhadap pertumbuhan ayam nantinya.
- Bila suhunya melebihi kebutuhan, anak ayam akan menjauh dari sumber panas. Anak ayam akan merasa selalu haus, terengah-engah dan akhirnya sakit pula.



Gambar 2-22 Konstruksi Indukan Ayam Pedaging

Pemberian cahaya lampu pada anak ayam di malam hari sungguh sangat menunjang tata laksana, sebab pemberian cahaya ini berarti memberi kesempatan makan dan minum. Suatu pertimbangan penting dalam rangka tata laksana dengan pengaturan cahaya seperti dialami di negara tropis Indonesia ini, karena pada siang hari keadaan suhu udara umumnya sangat tinggi, sehingga berakibat nafsu makan ayam menurun, tetapi keinginan minum meningkat. Karena pada siang hari ayam makanannya berkurang dan lebih banyak minum, maka untuk mengejar dan mengimbangi makanan yang masuk, ayam harus diberi kesempatan makan di waktu malam. Karena itu perlu diusahakan penerangan pada malam hari.

Pada saat lampu mati, biasanya anak ayam bertumpuk di satu sudut (akibat terkejut) untuk mendapatkan kehangatan. Hal ini dapat mengakibatkan anak ayam mati. Oleh karena itu walaupun sudah ada listrik perlu juga disediakan lampu minyak sebagai cadangan bila sewaktu-waktu listrik mati.

Cara lain untuk mengetahui anak ayam itu kedinginan atau kepanasan adalah dengan melihat sebaran anak ayam. Bila anak ayam sebagian besar menjauhi sumber panas, ini menandakan bahwa pemanas pada indukan terlalu

panas atau temperatur indukan meninggi. Bila sebagian besar anak ayam itu mendekati atau berkerumun disekitar pemanas, berarti indukan kurang panas. Sebaran anak ayam diperhatikan terutama di malam hari. Temperatur indukan diperiksa 4-5 kali semalam. Temperatur indukan ini sangat penting mengingat Indonesia berada di iklim tropis, sedangkan bibit ayam broiler berasal dari negara barat.

2.13.2. Pedoman Pengaturan Suhu Indukan Ayam Pedaging

Dua puluh empat jam sebelum anak ayam datang, penghangat pada indukan sudah dihidupkan (lampu pijar atau api sudah harus dipasang), temperatur yang dibutuhkan adalah sebagai berikut:

Tabel 2-5 Pedoman Pengaturan Suhu Pada Indukan Ayam Pedaging [13]

Umur Ayam (Minggu)	Suhu (^{o} Fahrenheit)	Suhu (^{o} Celcius)
1	95°	35°
2	90°	32,2^o
3	85°	29,4^o

Pada saat anak ayam tiba, hendaknya temperatur 95 ^{o}F sudah tercapai. Untuk keperluan itu, digunakan termometer. Biasanya skala fahrenheit lebih mudah dilihat daripada skala celciusnya. Untuk daerah pegunungan, indukan diteruskan sampai minggu ketiga, jika keadaan cuaca kurang menguntungkan karena banyak hujan atau suhu relatif rendah, maka kebutuhan panas masih bisa diperpanjang.

BAB III

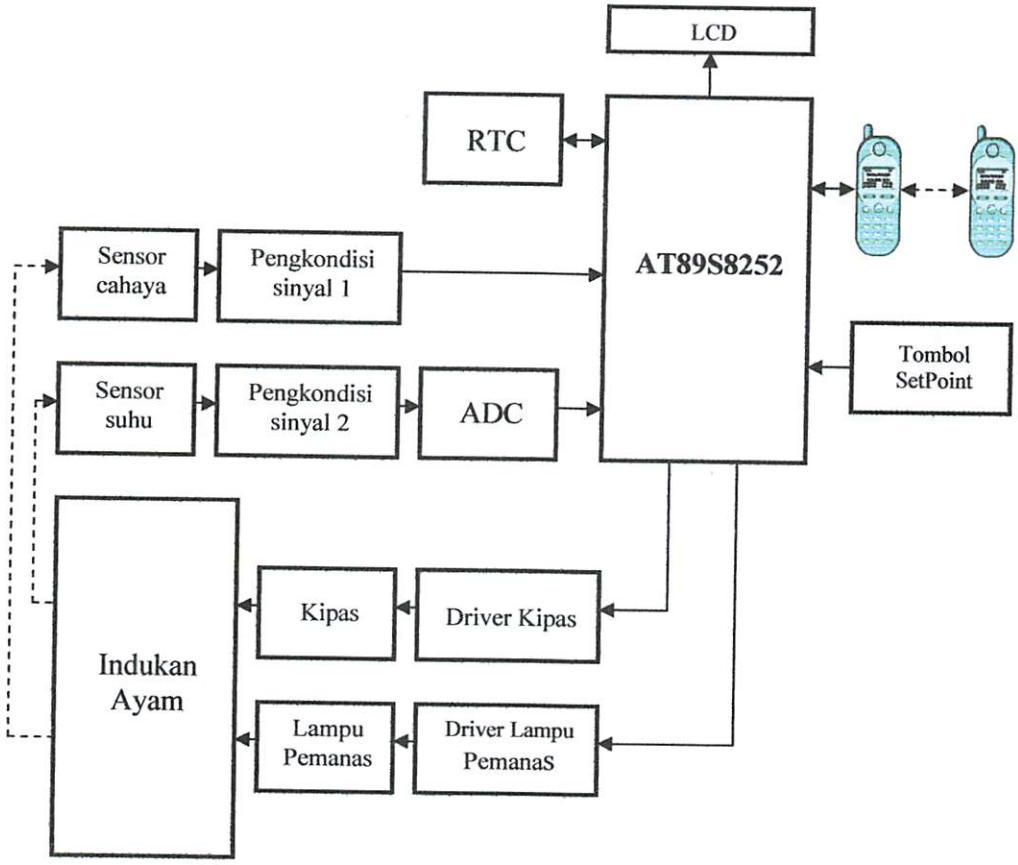
PERANCANGAN DAN PEMBUATAN ALAT

3.1. Pendahuluan

Pada bab ini membahas tentang perencanaan atau perancangan perangkat keras *(hardware)* dan perangkat lunak *(software)* yang digunakan dalam sistem pengaturan suhu yang menggunakan lampu pijar sebagai sarana pemanas pada indukan ayam pedaging yang diinformasikan via sms *(short message service)*.

3.2. Perancangan Perangkat Keras *(Hardware)*

Garis besar sistem perancangan ditunjukkan pada diagram blok dibawah:



Gambar 3-1 Diagram Blok Sistem

Keterangan fungsi dari masing-masing blok diatas adalah sebagai berikut:

1. Sensor suhu. Suhu merupakan input dari sistem, dimana dalam perancangan alat ini menggunakan IC LM35 dengan rentang suhu yang dapat dideteksi antara -55°C sampai 150°C.
2. Sensor cahaya, menggunakan LDR (light Dependent Resistor) yang berfungsi untuk mendeteksi lampu apakah dalam keadaan mati atau nyala.
3. Pengkondisi sinyal 1, difungsikan sebagai pembanding antara tegangan yang dihasilkan oleh sensor cahaya dengan tegangan referensi untuk kemudian difungsikan sebagai saklar.
4. Pengkondisi Sinyal 2 disini berfungsi untuk menguatkan tegangan output dari sensor suhu sebelum diinputkan ke ADC. Penguatan yang digunakan berupa non inverting Amplifier.
5. ADC *(Analog to Digital Converter)*

   Untuk mengubah data yang berbentuk analog menjadi data yang berbentuk digital, ADC yang digunakan adalah ADC 0804.
6. RTC *(Real Time Clock)*

   Menggunakan RTC DS12C887 yaitu IC yang dipakai sebagai basis pewaktuan sistem.
7. Mikrokontroller AT89S8252

   Berfungsi sebagai pengontrol semua sistem, dalam IC ini terdapat 8Kbytes EPROM, 256bytes RAM internal, 32 *Programmable I/O lines*, 2Kbytes EEPROM dan 3 buah timer / counter 16 bit.

8. LCD (*Liquid Crystal Display*)

   Sebagai display suhu pada indukan tersebut, bisa juga disebut termometer digital.

9. Driver kipas, yaitu sebuah rangkaian penggerak yang digunakan untuk menghidupkan kipas yang difungsikan sebagai pendingin. Kipas disini hanya bersifat on/off saja.

10. Kipas sebagai alat untuk menurunkan suhu pada indukan ayam.

11. Driver Lampu pemanas, berfungsi sebagai rangkaian pengendali lampu. Dalam hal ini, nyala lampu akan diatur oleh sistem dengan menggunakan metode PWM *(Pulse Width Modulation)*.

12. Lampu pemanas sebagai alat untuk menghasilkan suhu sesuai dengan yang diinginkan sekaligus sebagai sarana penerangan.

13. Tombol setpoint berfungsi sebagai setting point suhu.

14. *Handphone* (pada sistem)

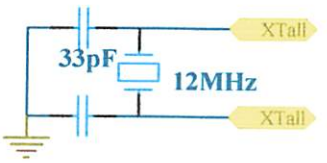
    Berfungsi sebagai sarana untuk mengirimkan informasi berupa sms kepada *handphone* peternak. *Handphone* yang digunakan adalah siemens type C35.

3.2.1. Mikrokontroler AT89S8252

3.2.1.1. Perancangan Rangkaian Clock

Kecepatan proses yang dilakukan oleh mikrokontroller ditentukan oleh sumber *clock* yang mengendalikan mikrokontroller tersebut. Sistem yang dirancang ini menggunakan osilator internal yang telah tersedia dalam *chip* AT89S8252. Untuk menentukan frekuensi osilatornya cukup dengan menghubungkan kristal dalam pin 19 (X1) dan pin 18 (X2) serta dua buah kapasitor ke *ground.*

Besarnya kapasitansinya disesuaikan dengan spesifikasi dalam lembar data AT89S8252 yaitu 33pF. Kristal yang digunakan adalah 12 MHz. gambar 3-2 memperlihatkan rangkaian *clock* yang dirancang.

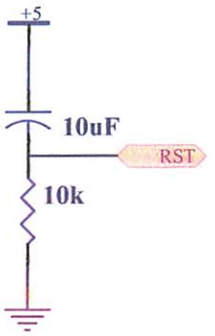


Gambar 3-2 Rangkaian Clock untuk MCU AT89S8252

Dalam sistem ini, rangkaian terdiri dari dua buah kapasitor sebesar 33 pF dan sebuah kristal sebesar 12 MHz.

3.2.1.2. Perancangan Rangkaian Reset

Pin reset pada mikrokontroller merupakan masukan aktif high (1). Pulsa transisi dari low (0) dan high (1) akan mereset mikrokontroller menuju alamat 0000H. pin reset dihubungkan dengan rangkaian power on reset yang diperlihatkan pada gambar 3-3 berikut ini:



Gambar 3-3 Rangkaian Reset untuk MCU AT89S8252

Karena kristal yang digunakan mempunyai frekuensi sebesar 11,0592 MHz maka satu periode membutuhkan waktu sebesar:

$$T = \frac{1}{f_{XTAL}}$$

$$= \frac{1}{11{,}0592MHz} S$$

$= 9{,}042 \times 10^{-8}$ S

Sehingga waktu minimal logika tinggi yag dibutuhkan untuk me-*reset* mikrokontroller adalah:

Reset (min) = T x periode yang dibutuhkan

$= 9{,}042 \times 10^{-8} \times 24$

$= 2{,}17$ µS

Mikrokontroller membutuhkan waktu minimal 2,17 µS untuk me-*reset*. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. dari persamaan diatas dengan menentukan nilai R = 10 KΩ dan C = 10 µF, maka:

$T = 0{,}357 * R * C$

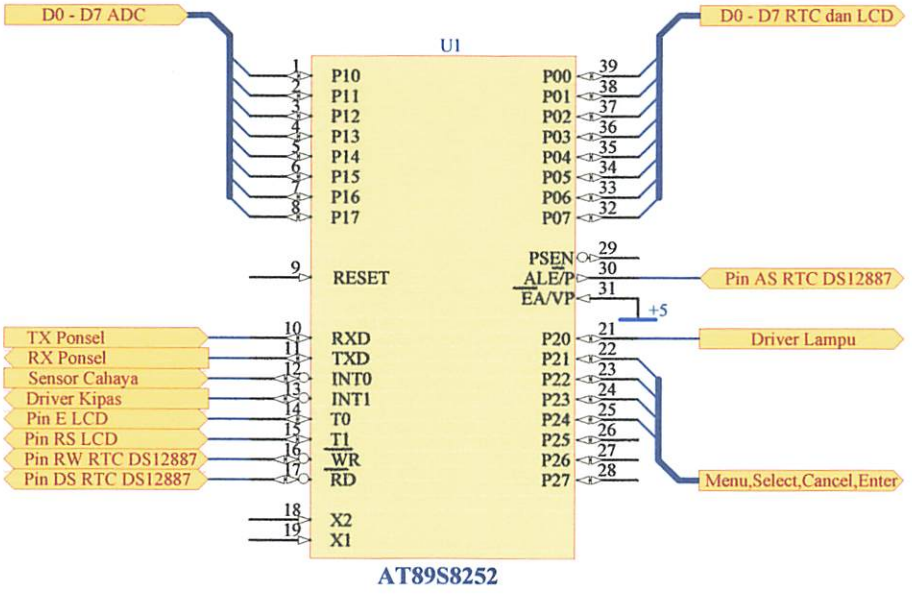
$= 0{,}357 * 10 \times 10^{6}\ \Omega * 10 \times 10^{-6}$

$= 35{,}7$ mS.

Jadi dengan nilai komponen R = 10 KΩ dan C = 10 μF dapat memenuhi syarat minimal untuk waktu yang dibutuhkan oleh mikrokontroller.

3.2.1.3. Perancangan Penggunaan Port pada Mikrokontroller AT89S8252

Dalam perancangan dan pembuatan rangkaian alat kontrol suhu pada indukan ayam pedaging ini menggunakan mikrokontroller AT89S8252 bertujuan untuk mendapatkan rangkaian yang praktis dan ringkas. IC mikrokontroller AT89S8252 ini sendiri memiliki fasilitas EEPROM sebagai memory program, sehingga sistem alat menjadi lebih sederhana. Adapun aplikasi mikrokontroller AT89S8252 dalam perancangan ini adalah sebagai berikut:



Gambar 3-4 Rangkaian MCU AT89S8252

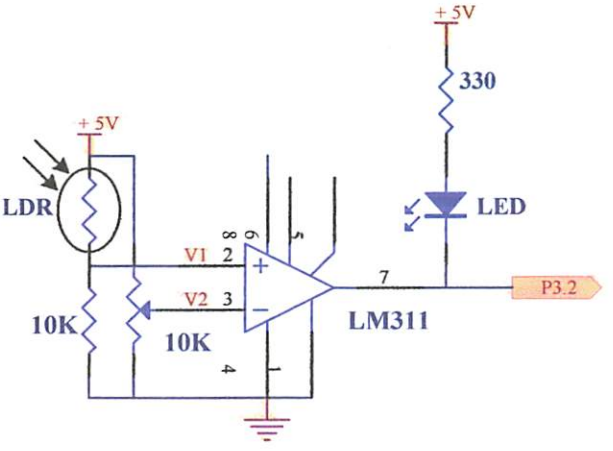
Mikrokontroller AT89S8252 memiliki 4 port paralel yang masing-masing digunakan sebagai berikut:

- Port 0 (P0.0 – P0.7) digunakan sebagai port data untuk RTC dan LCD
- Port 1.0 – 1.7 digunakan sebagai port data untuk ADC.
- Port 2.0 digunakan sebagai port untuk driver lampu pemanas.
- Port 3.0 (RXD) dan port 3.1 (TXD) digunakan sebagai komunikasi serial dengan ponsel.
- Port 3.2 digunakan sebagai port untuk sensor cahaya
- Port 3.3 digunakan sebagai port untuk driver fan / kipas.
- Port 3.4 digunakan sebagai port untuk pin E pada LCD.
- Port 3.5 digunakan sebagai port untuk pin RS pada LCD.
- Port 3.6 digunakan sebagai port untuk pin RW pada RTC.
- Port 3.7 digunakan sebagai port untuk pin DS pada RTC.
- Port 2.1 – 2.7 difungsikan sebagai port eksternal.

3.2.2. Perancangan Sensor

3.2.2.1. Sensor Cahaya / LDR

Sensor cahaya disini dipakai untuk mendeteksi keadaan lampu pijar, apakah lampu dalam keadaan hidup atau mati. Sedangkan komponen yang digunakan adalah LDR *(Light Dependent Resistor)*, yaitu resistor yang nilai resistansinya berubah-ubah tergantung pada cahaya yang diterimanya.



Gambar 3-5 Rangkaian Sensor Cahaya

LDR mempunyai resistansi yang besar pada kondisi yang gelap yaitu sekitar 1 MΩ dan resistansi ini akan turun menjadi ratusan ohm pada saat terang.

Komponen yang utama disini adalah LDR untuk mendeteksi cahaya dan sebagai rangkaian komparator digunakan IC op-amp LM311.

LDR dirangkai seri dengan tahanan 10KΩ, dengan demikian kita bisa mencari berapa besar tegangan V1 pada saat lampu mati dan V1 pada saat lampu nyala untuk menentukan berapa besarnya V2 atau V referensi.

Pada saat lampu mati, resistansi LDR bernilai sekitar 1 MΩ, maka dapat dicari V1 dengan menggunakan rumus pembagi tegangan sebagai berikut:

$$V1 = \frac{Rs}{Rs + Rldr} * Vcc$$

$$V1 = \frac{10K}{10K + 1000K} * 5$$

V1 = 0,009 * 5

V1 = 0,049 volt.

Pada saat lampu menyala, kita anggap resistansi LDR bernilai sekitar 600Ω, maka dapat dicari V1 dengan menggunakan rumus pembagi tegangan sebagai berikut:

$$V1 = \frac{Rs}{Rs + Rldr} * Vcc$$

$$V1 = \frac{10K}{10K + 600} * 5$$

V1 = 0,94 * 5

V1 = 4,7 volt.

Dengan diketahui besarnya tegangan V1 pada saat LDR terkena cahaya sebesar 0,049 volt dan V1 pada saat LDR tidak terkena cahaya sebesar 4,7 volt, maka kita dapat menentukan besarnya V2 atau V referensi. Dalam perencanaan alat ini, V2 di set pada nilai 0,1 volt.

Cara kerja rangkaian diatas adalah sebagai berikut:

- Pada saat lampu menyala dan LDR menerima cahaya, nilai V1 lebih besar dari V2, maka IC op-amp LM311 akan mengeluarkan logic 1. Karena tidak ada beda potensial tegangan antara sumber 5 Vdc dengan pin 7 / output dari LM311, maka LED mati dan P3.2 menerima logic 1.
- Pada saat lampu mati dan LDR tidak menerima cahaya, nilai V1 lebih kecil dari V2, maka IC op-amp LM311 akan mengeluarkan logic 0. Karena adanya beda potensial tegangan antara sumber 5 Vdc dengan pin 7 / output dari LM311, maka LED menyala dan P3.2 menerima logic 0.

Penggunaan resistor 330 ohm adalah sebagai pembatas arus pada LED. Kebutuhan arus pada LED adalah 15 mA, dengan tegangan Vcc 5 volt, maka dapat dicari nilai tahanannya dengan cara sebagai berikut:

$$V = I * R$$

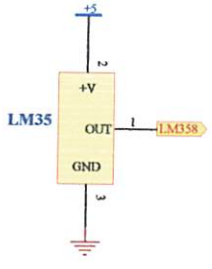
$$R = \frac{V}{I}$$

$$= \frac{5}{0.015} = 333{,}33\ \Omega$$

Jadi dari hasil perhitungan diatas dieroleh nilai R_B sebesar 333,33 Ω atau disederhanakan menjadi 330 Ω.

3.2.2.2. Sensor Suhu (LM35)

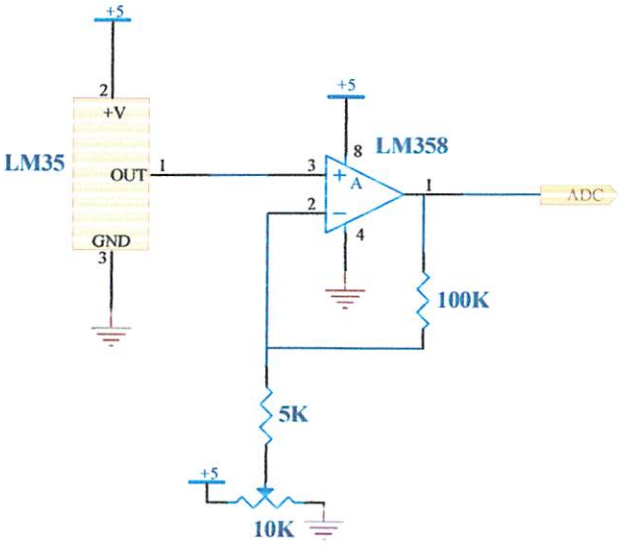
Sensor suhu yang dipakai adalah LM35 yang merupakan IC sensor temperatur yang presisi, yang mana outputnya linear dengan kenaikan temperatur dalam derajat celcius. Tegangan keluaran pada LM35 akan mengalami perubahan sebesar 10mV setiap kenaikan suhu sebesar 1° C. Sebagai contoh pada suhu 35° C maka keluaran tegangan LM35 adalah 350 mV.



Gambar 3-6 Sensor Suhu LM35

Dengan menghubungkan pin Gnd ke tanah, maka batas bawah keluarannya adalah 0 Volt untuk 0 °C sehingga keluarannya sebesar 1 Volt pada 100 °C.

3.2.2.2.1 Perancangan Pengkondisi Sinyal



Gambar 3-8 Rangkaian Pengkondisi Sinyal

Penguat Differensial digunakan untuk mengkondisikan tegangan dari sensor suhu LM35 menjadi tegangan yang berkisar antar 0 – 5 Volt, sesuai dengan tegangan input ADC 0804. Suhu yang diinginkan adalah 29,4 °C sampai 35 °C, tetapi dalam perancangan ini kita ambil range suhu antara 25 °C sampai 40 °C.

Untuk mendapatkan tegangan keluaran ini maka output dari sensor diumpankan ke suatu rangkaian penguat *differensial amplifier* dengan menggunakan IC LM358, dapat dihitung dengan menggunakan rumus:

$$Vout = AD * Vin$$

$$AD = \left(\frac{Rf}{Ri}\right)$$

$$Vout = \left(\frac{Rf}{Ri}\right) * Vin$$

Dengan:

Vin = V1 – Vref

V1 = tegangan output LM 35

Pada saat suhu 25 °C (Vout LM 35 = 0,25 Volt) tegangan output pengkondisi sinyal harus 0 Volt karena pada ADC 0804 menempati range ke 0 (range ADC antara 0 – 255), maka Vref di set pada tegangan:

$$Vout = Av * Vin$$

$$Vout = AD * (0{,}25 - Vref)$$

$$0Volt = AD * (0{,}25 - Vref)$$

$$Vref = 0{,}25\,Volt$$

Pada saat suhu 25,1 °C (Vout LM35 = 0,251 Volt) dengan Vref sebesar +0,25 Volt dan pada ADC 0804 menempati range ke 10 (tegangan input ADC = 0,02 Volt), maka penguatannya dirancang sebesar:

$$AD = \frac{Vout}{Vin}$$

$$AD = \frac{0{,}02\,V}{(0{,}251 - 0{,}25)V}$$

$$AD = \frac{0{,}02\,V}{0{,}001\,V}$$

$$AD = 20\,kali$$

Dari perancangan diatas, maka:

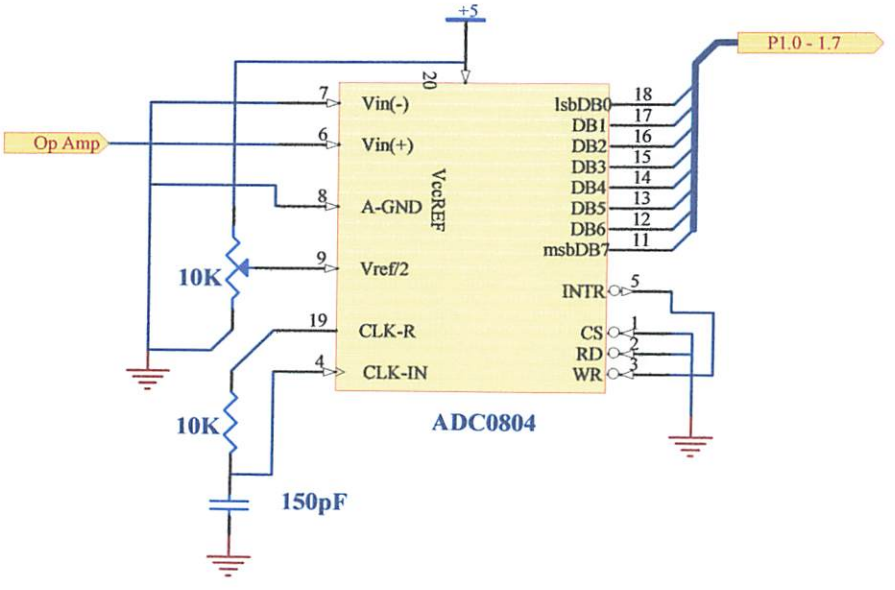
$$AD = \frac{Rf}{Ri}$$

$$20 = \frac{100\,K}{Ri}$$

$$Ri = \frac{100\,K}{20} = 5K\Omega$$

3.2.3 ADC *(Analog To Digital Converter)* 0804

Pada Perancangan alat ini menggunakan ADC 0804. ADC ini merupakan jenis IC dengan keluaran 8 bit yang mempunyai keluaran biner dan menampilkan waktu pengubah singkat. ADC 0804 ini beroperasi pada daya standard +5 Volt yang dapat mengkode tegangan analog masukan berkisar 0 – 5 Volt.



Gambar 3-9 Rangkaian ADC 0804

3.2.3.2 Clock pada ADC

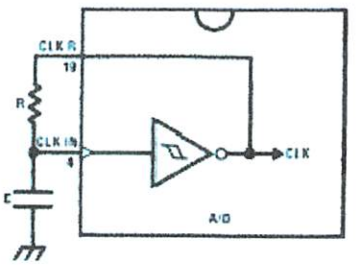
Clock pada ADC menahan pembangkit internal, sehingga hanya dibutuhkan sebuah resistor dan kapasitor yang dihubungkan pada jalur CLK R (Pin 19) dan CLK IN (Pin 4). Dengan mengambil nilai R= 10 KΩ dan C= 150 pF maka diperoleh frekwensi Clock sebesar:

$$Fclk = \frac{1}{1{,}1 * R * C}$$

$$Fclk = \frac{1}{1{,}1 * 10.10^{3} * 150.10^{-12}}$$

$Fclk = 606{,}06$ KHz

Frekwensi ini masih berada dalam batas range ADC 0804 yaitu 100 KHz sampai 1 MHz.



Gambar 3-10 Self Clocking ADC 0804

3.2.4 RTC *(Real Time Clock)* DS12887

Pada sistem ini RTC difungsikan sebagi referensi waktu (detik, menit, jam, tanggal, hari, bulan dan tahun). RTC DS12C887 mempunyai RAM internal sebesar 128 bytes yang terdiri atas 14 bytes untuk pewaktuan dan 114 bytes untuk data yang bisa diprogram oleh pemakai. Hubungan penyemat untuk RTC DS 12C887 antara lain:

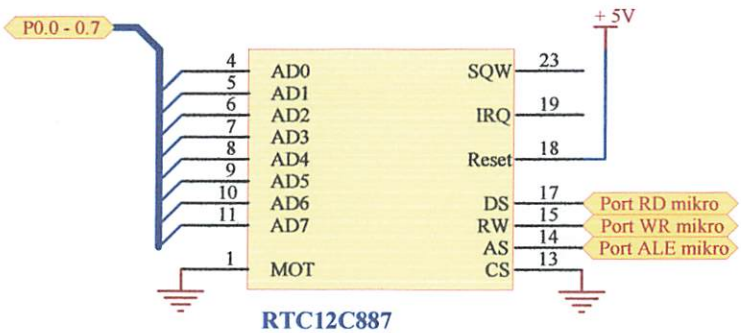
a. AS dihubungkan dengan ALE Mikrokontroller yang berfungsi untuk memisahkan saluran data dan saluran alamat yang multipleks pada saluran AD0-AD7.

b. R/W dihubungkan WR Mikrokontroller yang berfungsi sebagai kontrol untuk penulisan data pada RAM internal.

c. DS dihubungkan dengan RD Mikrokontroller yang berfungsi sebagai kontrol untuk pembacaan data pada RAM internal.

d. MOT dihubungkan ke ground untuk mendapatkan sistem bus intel.

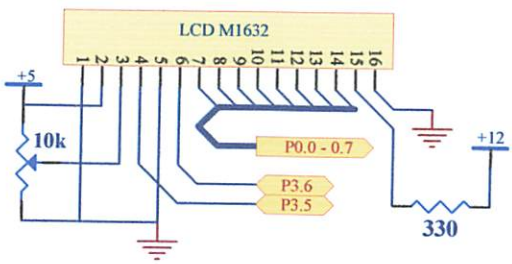
Operasi pembacaan dan penulisan pada RAM internal RTC DS 12C887 ini sama dengan operasi baca tulis pada RAM.



Gambar 3-11 Rangkaian Real Time Clock IC RTC DS12C887

3.2.5 LCD *(Liquid Crystal Display)* M1632

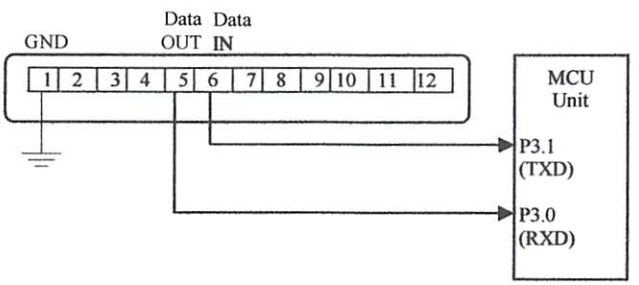
LCD terdiri atas tumpukan tipis atau sel dari dua lembar kaca dengan pinggiran tertutup rapat. Antara dua lembar kaca tersebut diberi bahan kristal cair *(liquid crystal),* yang tembus cahaya. Permukaan luar dari masing-masing keping kaca mempunyai lapisan penghantar tembus cahaya seperti Oxida Timah *(tin oxide)*. Sel mempunyai ketebalan sekitar 1 x 10^{-5} meter dan diisi dengan kristal cair.



Gambar 3-12 Rangkaian LCD M1632

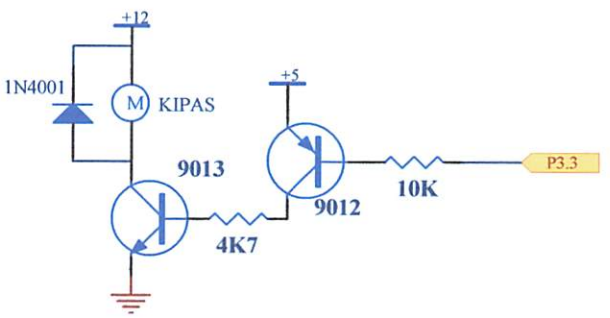
3.2.6 Perancangan Interface Ponsel dengan MCU

Mikrokontroller AT89S8252 dilengkapi dengan port serial. Port serial memungkinkan untuk interface dengan hardware lain dalam format serial. Komunikasi serial antara mikrokontroller AT89S8252 dengan ponsel hanya tinggal menghubungkan pin-pin serial dari masing-masing hardware. Pin Tx dari mikrokontroller akan dihubungkan dengan pin Rx yang ada di ponsel, sedangkan pin Rx yang ada di mikrokontroller dihubungkan dengan Tx di ponsel, pin ground dari mikrokontroller dihubungkan dengan pin ground dari ponsel. Gambar dibawah menunjukkan hubungan antara ponsel dengan mikrokontroller.



Gambar 3-13 Rangkaian Interface antara Ponsel dengan Mikrokontroller

3.2.7 Driver Kipas



Gambar 3-14 Rangkaian Driver Kipas

Kipas dalam perancangan ini berfungsi sebagai pendingin apabila suhu melebihi dari setpoint. Kipas ini hanya bersifat ON dan OFF saja.

Diketahui dari datasheet transistor 9012 besarnya I_C = 50 mA, hfe = 120, V_{BE} = 0,7 Volt, sehingga :

$$I_C = I_B * hfe$$

$$I_B = \frac{I_C}{hfe}$$

$$I_B = \frac{50mA}{120} = 0{,}416mA$$

Maka R_B adalah :

$$R_B = \frac{Vcc - Vbe}{I_B}$$

$$R_B = \frac{5 - 0{,}7}{0{,}416mA}$$

$$= \frac{4{,}3}{0{,}416mA} = 10{,}33 \text{ x } 10^3\ \Omega$$

Jadi dari hasil perhitungan diatas dieroleh nilai R_B sebesar 10,33 x 10^3 Ω atau disederhanakan menjadi 10 KΩ.

Sedangkan dari data sheet transistor 9013 besarnya I_C = 100 mA, hfe = 110, V_{BE} = 0,7 Volt, sehingga :

$$I_B = \frac{100mA}{110} = 0{,}909mA$$

Maka R_B adalah :

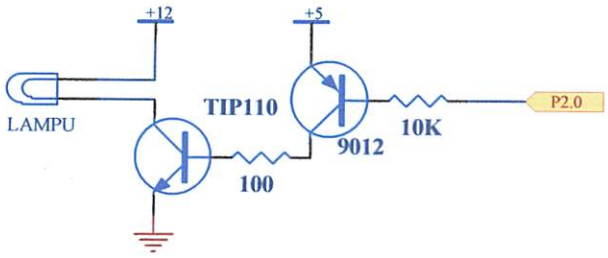
$$R_b = \frac{5 - 0{,}6 - 0{,}7}{0{,}909mA}$$

$$= \frac{3{,}7}{0{,}909mA} = 4{,}07 \text{ x } 10^3\ \Omega$$

Jadi dari hasil perhitungan diatas dieroleh nilai R_B sebesar 4,07 x 10^3 Ω, tetapi dalam perancangan ini memakai 4K7 Ω.

3.2.8 Driver Lampu Pemanas

Lampu bekerja sebagai pemanas sekaligus sebagai sarana penerangan. Intensitas cahaya lampu tergantung dari setpoint suhu yang kita berikan. Nilai suhu diambil dari kebutuhan panas anak ayam pedaging. Sementara nilai setpoint suhu didapat dengan nilai *duty cycle* yang kita berikan dan nilai *duty cycle* ini sendiri mempengaruhi intensitas cahaya lampu.



Gambar 3-14 Rangkaian Driver Lampu Pemanas

Diketahui dari data sheet transistor 9012 besarnya I_C = 50 mA, hfe = 120, V_{be} = 0,7 Volt, sehingga :

$$I_B = \frac{50mA}{120} = 0{,}416mA$$

Maka R_B adalah :

$$R_B = \frac{5-0{,}7}{0{,}416mA}$$

$$= \frac{4{,}3}{0{,}416mA} = 10{,}33 \text{ x } 10^3\ \Omega$$

Jadi dari hasil perhitungan diatas dieroleh nilai R_B sebesar 10,33 x 10^3 Ω atau disederhanakan menjadi 10 KΩ.

Sedangkan dari data sheet transistor TIP 110 besarnya I_C = 4 A, hfe = 250, V_{be} = 0,7 Volt, sehingga :

$$I_B = \frac{4A}{250} = 16mA$$

Maka R_B adalah :

$$R_b = \frac{5 - 0{,}6 - 0{,}7}{16mA}$$

$$= \frac{3{,}7}{16mA} = 231{,}25\ \Omega$$

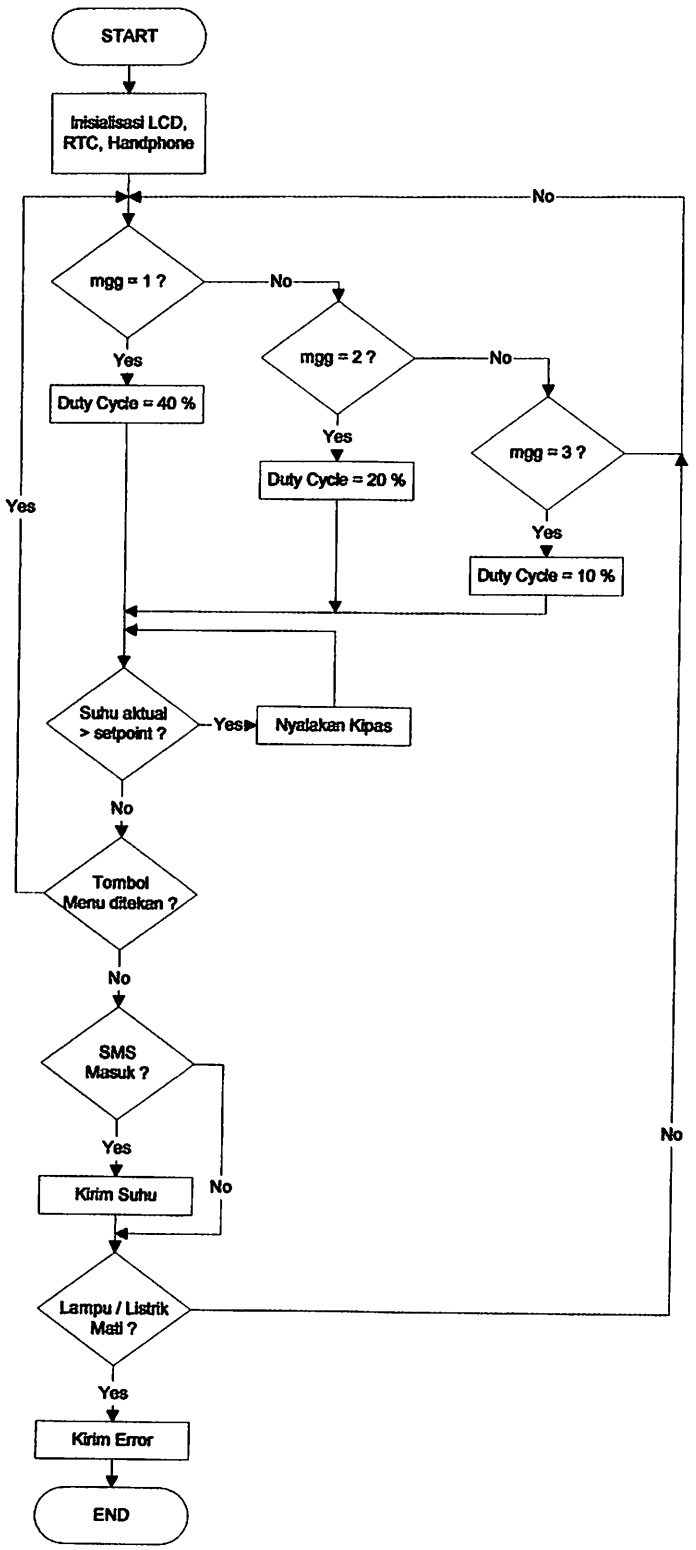
Jadi dari hasil perhitungan diatas diperoleh nilai R_B sebesar 231,25 Ω, tetapi dalam perancangan ini menggunakan 100 Ω.

3.3. Perancangan Perangkat Lunak *(Software)*

Untuk mendukung agar perangkat keras berfungsi sesuai dengan perencanaan, maka diperlukan perangkat lunak sebagai penunjangnya. Perangkat lunak ini sendiri maksudnya adalah suatu program yang kita buat yang nantinya akan ditanam kedalam mikrokontroller AT89S8252. setelah mikrokontroller tersebut diprogram, maka akan diketahui apakah program yang telah kita buat bekerja sesuai dengan yang kita rencanakan ataukah masih memiliki kesalahan.

Sistem aplikasi mikrokontroller AT89S8252 ini dapat mengatur dan mengendalikan keseluruhan sistem apabila ada urutan instruksi yang mendefinisikan secara jelas urutan tugas yang harus dikerjakan. Urutan instruksi ini sangat penting untuk didefinisikan, karena mikrokontroller dapat bekerja secara pasti sesuai dengan instruksi yang telah dibuat.

3.4. Flowchart



BAB IV

PENGUJIAN RANGKAIAN DAN ANALISA DATA

Untuk mendapatkan hasil yang maksimal setelah melaksanakan perancangan dan pembuatan alat, maka perlu dilakukan suatu pengujian terhadap alat yang telah dibuat. Pengujian ini bertujuan untuk mengetahui apakah alat yang telah dibuat dapat bekerja sesuai yang dengan perencanaan.

Bagian-bagian yang diuji dari peralatan ini adalah :

1. Terima SMS.
2. Kirim *SMS* dari Mikrokontroller.
3. Rangkaian Sensor Suhu.
4. Rangkaian Sensor Cahaya.
5. Rangkaian *Analog To Digital Converter* (ADC) 0804.
6. Rangkaian *Driver* Kipas.
7. Rangkaian *Driver* Lampu Pemanas, dan
8. Alat Secara Keseluruhan.

4.1. Pengujian Terima SMS

4.1.1. Tujuan

Untuk mengetahui bahasa PDU yang dihasilkan dari karakter yang kita ketikkan di ponsel.

4.1.2. Peralatan Yang Digunakan

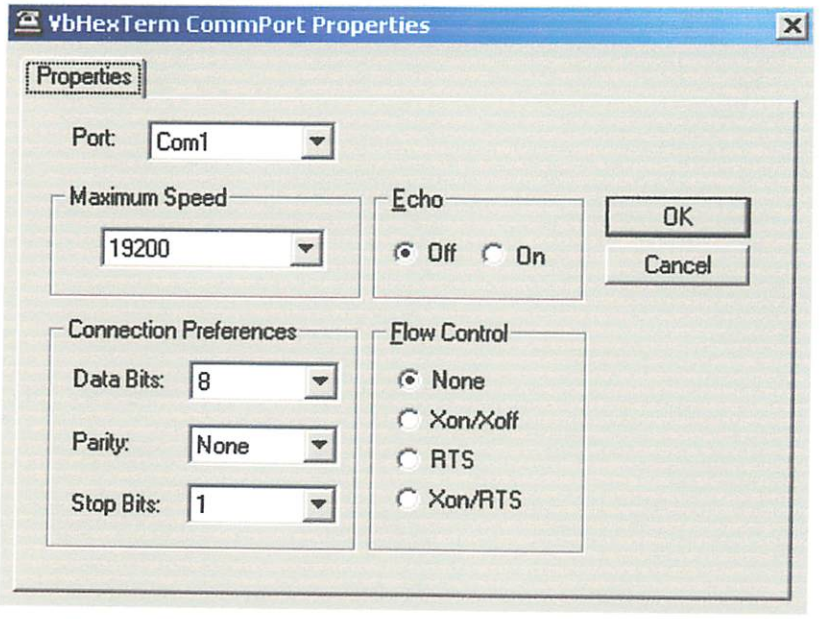
1. Ponsel Type C35.
2. Ponsel Pengirim.

3. Kabel Data Ponsel Type C35.

4. Komputer.

4.1.3. Langkah Pengujian

1. Membuka program PDUspy pada Komputer, kemudian memberikan setting baudrate 19200, 8 data bits, no parity dan 1 stop bit.



Gambar 4-1 Setting Com

2. Menghubungkan ponsel C35 dengan Komputer.

3. Mengetikkan karakter "cek" pada Ponsel Pengirim.

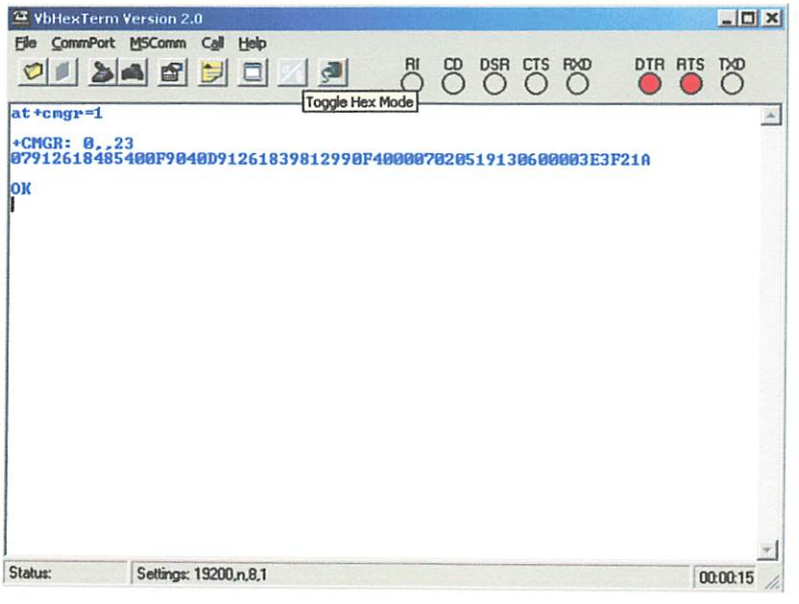
4. Mengirimkan karakter "cek" ke Ponsel C35.

5. Melihat hasilnya pada Komputer.

Gambar 4-2 Diagram Blok Pengujian Kirim SMS

4.1.4. Hasil Pengujian

Hasil pengujian dari Kirim SMS dengan mengetikkan kata "cek" adalah sebagai berikut:



Gambar 4-3 Hasil Konversi Karakter "cek"

4.1.5. Pembahasan

Dari pengujian di dapatkan PDU lengkap adalah:

07912618485400F9,04,0D91261839812990F4,00,00,702051931060,0003,E3F21A

Kemudian kita kupas satu persatu header dari delapan header untuk SMS-Terima tersebut, sehingga PDU tersebut dapat diartikan.

1. SMS tersebut dikirim lewat SMS-Centre: 62181445009, merupakan SMS-Centre untuk operator selluler Excelcom (**07912618485400F9**).
2. SMS tersebut merupakan SMS-Terima (**04**).
3. SMS tersebut dikirim dari ponsel no. 6281931892094 (**0D91261839812990F4**).
4. SMS tersebut diterima dalam bentuk SMS (**00**).
5. SMS tersebut memiliki skema encoding 7 bit (**00**).

6. SMS tersebut sampai di SMS-Sentre pada tanggal: 15-02-07, pukul 06:01:39 WIB (**702051931060**).
7. SMS tersebut memiliki batas waktu valid (**0003**).
8. SMS tersebut isinya adalah "cek" (**E3F21A**).

4.2. Pengujian Kirim SMS dari Mikrokontroller

4.2.1. Tujuan

Untuk mengetahui apakah Listing Program untuk mengirimkan SMS dari Mikrokontroller ke ponsel penerima bisa dilakukan. Pengujian ini juga untuk mengetahui apakah Mikrokontroller AT89S8252 dapat digunakan dengan baik.

4.2.2. Peralatan yang Digunakan

1. Rangkaian Mikrokontroller sederhana yang telah di isi program.
2. Catu Daya.
3. Ponsel type C35 untuk sistem.
4. Ponsel Penerima.
5. Push Button.

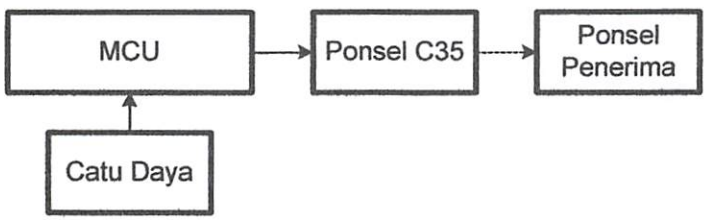
4.2.3. Langkah Pengujian

1. Membuat program kirim sms untuk di tuliskan ke mikrokontroller, sebagai berikut:

```
;==========================
;sms HALO
;==========================
;
sms HALO:     mov dptr,#HPDU_1
HALOa:        clr a
              movc a,@+dptr
              cjne a,#00h, HALOb
```

```
                mov a,#0Dh
                lcall SerOut
                ljmp HALOc
HALOb:          lcall SerOut
                inc dptr
                ljmp HALOa
HALOc:          mov dptr,#HPDU-2
HALOd:          clr a
                movc a,@a+dptr
                cjne a,#00h, HALOe
                ljmp HALOf
HALOe:          lcall SerOut
                inc dptr
                ljmp HALOd
HALOf:          mov dptr,#HPDU_3
HALOg:          clr a
                movc a,@a+dptr
                cjne a,#00h, HALOh
                ljmp HALOi
HALOh:          lcall SerOut
                inc DPTR
                ljmp HALOg
HALOi:          mov a,#1Ah
                lcall SerOut
                ret
;
HPDU_1: DB 'AT+CMGS=19',00h
HPDU_2: DB '07912618485400F9000D91261839812990F4',00h
HPDU_3: DB '004C820F309',00h
;
;HALO
;==================================================
;serial out
;==================================================
SerOut:   nop
          clr TI
          mov SBUF,A
          jnb TI,$
          clr TI
          ret
          Setb P2.4
;
Mulai:    Jb P2.4,Mulai
          Call smsHALO
          Jmp Mulai
```

2. Menghubungkan ponsel C35 dengan Mikrokontroller.
3. Memberi inputan Port 2.4 dengan menggunakan Push Button.
4. Memberikan catu daya 5 Volt pada rangkaian.
5. Melihat hasil *SMS* pada ponsel penerima.



Gambar 4-4 Diagram Blok Pengujian Kirim SMS dari Mikrokontroller

4.2.4. Hasil Pengujian

Hasil pengujian Kirim SMS dari Mikrokontroller adalah sebagai berikut:

Tabel 4-1 Hasil Pengujian Kirim SMS dari Mikrokontroller

Percobaan	Isi SMS
Dikirim	HALO
Diterima	HALO

4.3. Pengujian Rangkaian Sensor Suhu

4.3.1. Tujuan

Untuk memperoleh variasi tegangan yang dihasilkan oleh suhu yang diberikan.

4.3.2. Peralatan Yang Digunakan

1. Pemanas
2. Rangkaian Sensor Suhu.

3. Catu Daya 5 Volt.

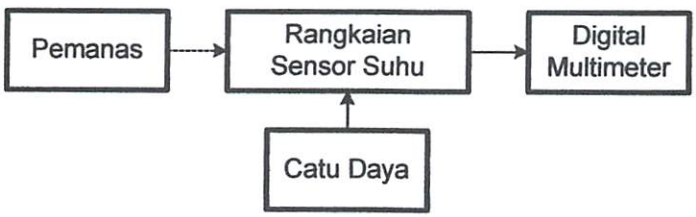
4. Digital Multimeter.

4.3.3. Langkah Pengujian

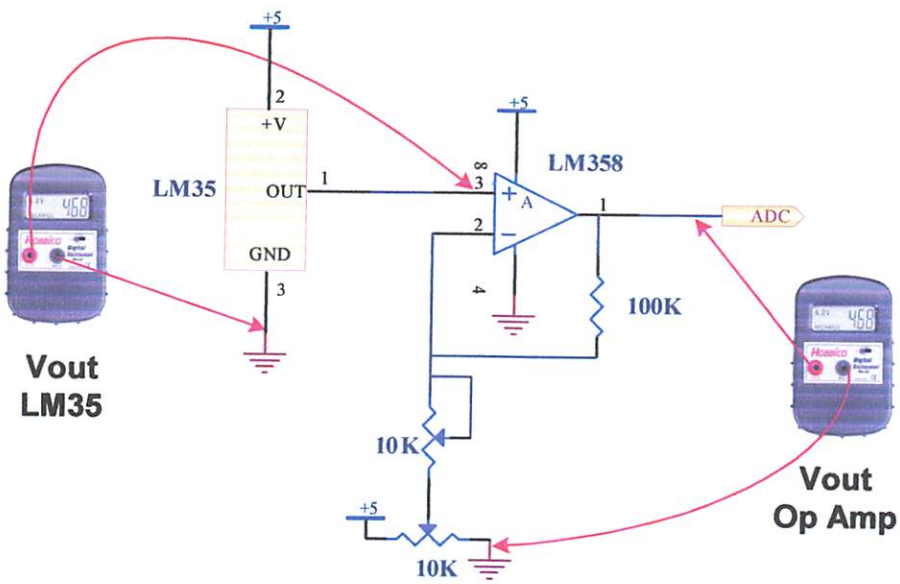
1. Menghubungkan Rangkaian Suhu dengan Catu Daya 5 Volt.

2. Memberikan Panas pada Sensor Suhu.

3. Mencatat hasil Tegangan yang dihasilkan.



Gambar 4-5 Diagram Blok Pengujian Sensor Suhu



Gambar 4-6 Rangkaian Pengujian Sensor Suhu

4.3.4. Hasil Pengujian

Hasil pengujian Rangkaian Sensor Suhu adalah sebagai berikut:

Tabel 4-2 Hasil Pengukuran Vout Sensor dan Vout Op-amp Terhadap Perubahan Suhu

Temperature (°C)	Vout Sensor (mV)	Vout Op-Amp (mV)
25	250	20
27,5	275	520
29	290	840
30	300	1,03V
32,5	325	1,55V
34	340	1,84V
36	360	2,24V
37,5	375	2,53V
39	390	2,86V
40	400	3,04V

Tabel 4-3 Perbandingan Vout Op-Amp Perhitungan Dengan Vout Op-Amp Pengukuran

Temperatur (°C)	Tegangan Output (mVolt)		% Error
	Perhitungan	Pengukuran	
25	0	20	0
27,5	500	520	4
29	800	840	5
30	1V	1,03V	3
32,5	1,5V	1,55V	3,3
34	1,8V	1,84V	2,2
36	2,2V	2,24V	1,8
37,5	2,5V	2,53V	1,2
39	2,8V	2,86V	2,1
40	3V	3,04V	1,3
Persentase (%) kesalahan rata-rata			**2,39%**

Tabel 4-4 Perbandingan Pembacaan Suhu Termometer Dengan Tampilan LCD

Temperatur (°C)	Tampilan LCD	Selisih	% Error
25	26,3	1,3	5,2
27,5	28,7	1,2	4,3
29	30,2	1,2	4,1
30	31,2	1,2	4
32,5	33,8	1,3	4
34	35,4	1,4	4,1
36	37,4	1,4	3,8
37,5	38,2	0,7	1,8
39	39,9	0,9	2,3
39,9	40,5	0,6	1,5
		Kesalahan Relatif :	**3,53%**

Untuk hasil perhitungan dari sensor suhu dapat dihitung sebagai berikut:

Vout = 10mV x T

Dimana:

10mV = ketetapan tegangan setiap kenaikan 1°C

T = perubahan *temperature*

Kesalahan dapat dicari dengan rumus sebagai berikut:

$$E = \left| \frac{Pengukuran - Perhitungan}{Perhitungan} \right| x100\%$$

4.4. Pengujian Rangkaian Sensor Cahaya

4.4.1. Tujuan

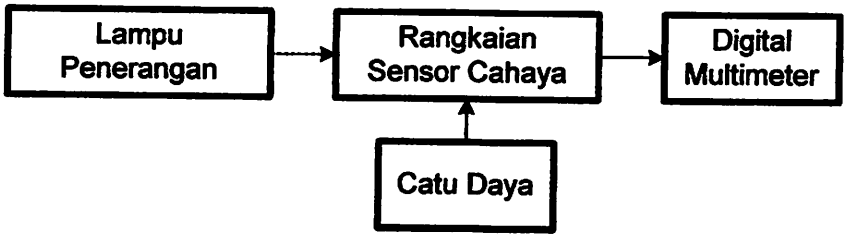
Untuk mengetahui apakah rangkaian sensor cahaya dapat bekerja sesuai dengan yang direncanakan sebagai saklar.

4.4.2. Peralatan Yang Digunakan

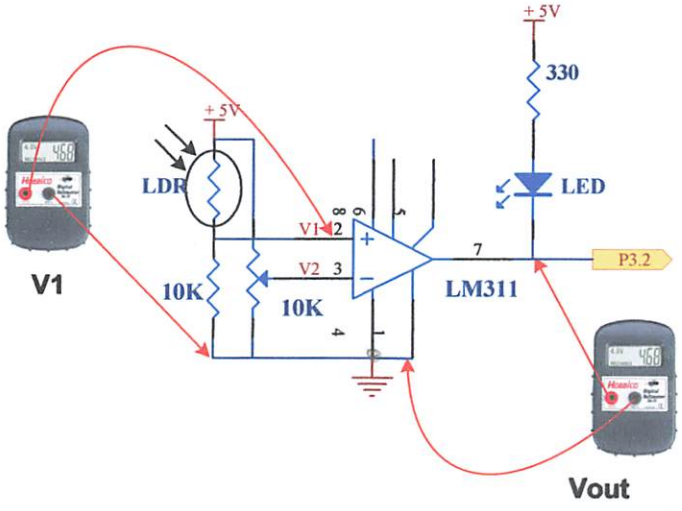
1. Lampu Penerangan.
2. Rangkaian Sensor Cahaya.
3. Catu Daya 5 Volt.
4. Digital Multimeter.
5. LED sebagai Lampu Indikator.

4.4.3. Langkah Pengujian

1. Menguhungkan Rangkaian Sensor Cahaya dengan Catu Daya 5 Volt.
2. Meletakkan Sensor Cahaya di dekat Lampu Penerangan.
3. Melihat lampu indikator dan mengukur tegangan yang dihasilkan.



Gambar 4-7 Diagram Blok Pengujian Sensor Cahaya



Gambar 4-8 Rangkaian Pengujian Sensor Cahaya

4.4.4. Hasil Pengujian

Dengan menggunakan Vreferensi atau V2 sebesar 0,1 V, didapatkan hasil pengujian Rangkaian Sensor Cahaya sebagai berikut:

Tabel 4-5 Hasil Pengujian Pada Saat Lampu Penerangan Menyala

	V1 (volt)	Vout (volt)	Kondisi LED
Perhitungan	4,7	0 V	Padam
Pengukuran	4,67 V	0 V	

Tabel 4-6 Hasil Pengujian Pada Saat Lampu Penerangan Mati

	V1 (volt)	Vout (volt)	Kondisi LED
Perhitungan	49 mV	5 V	Menyala
Pengukuran	98,5 mV	5 V	

4.5. Pengujian ADC *(Analog to Digital Converter)* 0804

4.5.1. Tujuan

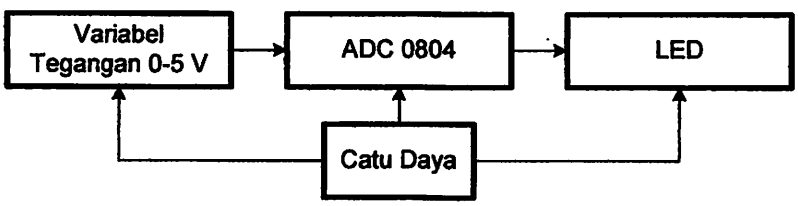
Untuk mengetahui hasil konversi tegangan analog ke digital setiap kenaikan 1 digit hasil konversi.

4.5.2. 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.

4.5.3. 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



Gambar 4-9 Diagram Blok Pengujian ADC 0804

4.5.4. Hasil Pengujian

Untuk mengetahui perubahan tegangan tiap bit ADC adalah dengan menggunakan perhitungan sebagai berikut:

Diketahui V = 5 Volt, Step ADC = 2^8 (255), maka

$$1 \text{ bit} = \frac{V}{StepADC}$$

$$= \frac{5}{255} = 19{,}6mV \approx 20 \text{ mV}$$

Tabel 4-7 Hasil Pembacaan dan Perhitungan ADC 0804

Input ADC (Volt)	Output ADC (Pembacaan)	Output ADC (Perhitungan)		
	Biner	Biner	Desimal	Heksadesimal
0	00000000	00000000	0	00h
0,25	00001101	00001101	13	0Dh
0,5	00011010	00011010	26	1Ah
0,75	00100110	00100110	38	26h
1,01	00110100	00110100	52	34h
1,26	01000000	01000000	64	40h
1,5	01001101	01001101	77	4Dh
1,75	01011001	01011001	89	59h
2,02	01100111	01100111	103	67h
2,25	01110011	01110011	115	73h
2,5	10000000	10000000	128	80h
2,75	10001100	10001100	140	8Ch
3	10011001	10011001	153	99h
3,25	10100110	10100110	166	A6h
3,5	10110011	10110011	179	B3h
3,75	10111111	10111111	191	BFh
4	11001100	11001100	204	CCh
4,25	11011001	11011001	217	D9h
4,5	11100110	11100110	230	E6h
4,75	11110010	11110010	242	F2h
4,99	11111111	11111111	255	FFh

Sedangkan untuk mengetahui data ADC ditentukan dengan perhitungan sebagai berikut :

Diketahui V analog = 0,25 volt, maka

$$\text{Data ADC} = \frac{Vana\log}{resolusi}$$

$$= \frac{0,25}{0,0196V} = 12,75 \approx 13_{(10)}$$

$$= 00001101_{(2)}$$

$$= 0D_{(16)}$$

4.6. Pengujian RTC *(Real Time Clock)* DS 12887

4.6.1. Tujuan

Untuk membandingkan tampilan waktu (jam, menit dan detik) pada alat dengan waktu sebenarnya.

4.6.2. Peralatan Yang Digunakan

1. Jam Digital sebagai penunjuk waktu.
2. Alat secara keseluruhan dengan rangkaian RTC yang sudah di program.
3. Catu Daya.

4.6.3. Langkah Pengujian

1. Menghubungkan alat dengan Catu Daya.
2. Membandingkan tampilan waktu pada alat dengan waktu sebenarnya pada jam digital.

4.6.4. Hasil Pengujian

Dari data hasil pengujian diatas, dapat disimpulkan bahwa perbandingan antara waktu pada jam digital dan waktu pada alat yang dibuat sudah sesuai atau tidak ditemukan selisih atau error = 0%.

Hasil Pengujjian RTC (Real Time Clock) DS 12C887 adalah sebagai berikut:

Tabel 4-8 Hasil Pengujian Waktu Pada Jam Digital Dengan Waktu Pada Alat

No	Waktu Pada Jam Digital	Waktu Pada Alat
1	07 : 00 : 00	07 : 00 : 00
2	08 : 00 : 00	08 : 00 : 00
3	09 : 00 : 00	09 : 00 : 00
4	10 : 00 : 00	10 : 00 : 00
5	11 : 00 : 00	11 : 00 : 00
6	15 : 00 : 00	15 : 00 : 00
7	16 : 00 : 00	16 : 00 : 00
8	17 : 00 : 00	17 : 00 : 00
9	18 : 00 : 00	18 : 00 : 00
10	19 : 00 : 00	19 : 00 : 00

4.7. Pengujian *Driver* Kipas

4.7.1. Tujuan

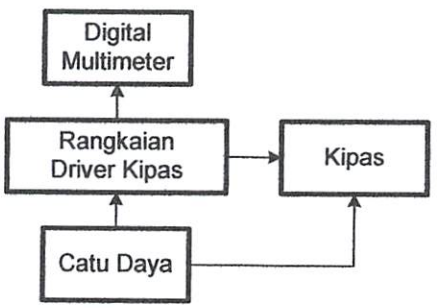
Untuk mengetahui apakah Rangkaian *Driver* Kipas yang telah dibuat dapat bekerja sesuai kondisi yang kita inginkan.

4.7.2. Peralatan Yang Digunakan

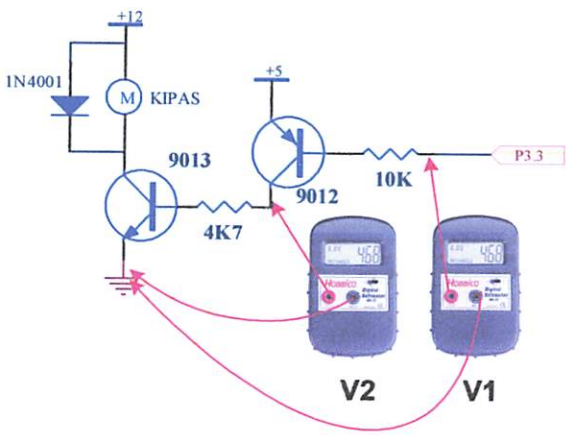
1. Rangkaian *Driver* Kipas.
2. Catu Daya 5 Volt dan 12 Volt DC.
3. Digital Multimeter.

4.7.3. Langkah Pengujian

1. Menghubungkan *Driver* Kipas dengan Catu Daya 5 Volt.
2. Menghubungkan Kipas dengan Catu Daya 12 Volt.
3. Memberikan Tegangan Saturasi pada Rangkaian.
4. Memberikan Tegangan Cut Off pada Rangkaian.
5. Mengukur output tegangan.



Gambar 4-10 Diagram Blok *Driver* Kipas



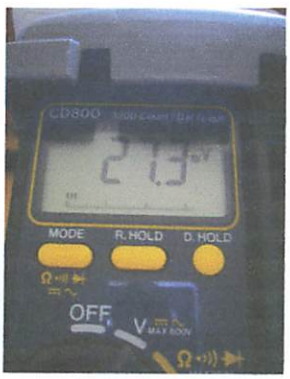
Gambar 4-11 Rangkaian Pengujian *Driver* Kipas

4.7.4. Hasil Pengujian

Hasil pengujian Rangkaian *Driver* Kipas adalah sebagai berikut:

Tabel 4-9 Hasil Pengujian Rangkaian *Driver* Kipas

Output MCU	Kondisi Transistor		V1 (Volt)	V2 (Volt)	Kondisi Kipas
	9012	9013			
0	Saturasi	Cut off	27,3 mV	4,4 V	Berputar
1	Cut off	Saturasi	4,98 V	0 V	Tidak Berputar

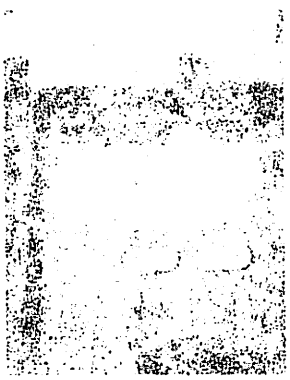


Gambar 4-12 Foto Hasil Pengukuran V1

Tabel 4-10 Hasil Pengujian *Driver* Kipas Terhadap Setpoint Suhu

Kondisi		Pengujian Ke	Hasil Pengujian	
Minggu	Setpoint Suhu (°C)		Suhu Aktual	Kondisi Kipas
Pertama	35	1	35,1	Berputar
		2	35,1	Berputar
		3	35,1	Berputar
Kedua	32,2	1	32,3	Berputar
		2	32,3	Berputar
		3	32,3	Berputar
Ketiga	29,4	1	29,5	Berputar
		2	29,5	Berputar
		3	29,5	Berputar

Datum	Kundin Transistor		V1	V2	Kundin Eigen
31.03	9012	9013	(Volt)	(Volt)	



Kundin		Frequenzen	Hauptmessungen	
Datum	Gehäuse Sockel (°C)	Kz	Sockel Nummer	Kundin Eigen
			3,0	Transistor
		2	3,0	Transistor
Kundin	30,1	1	3,0	Transistor

Dari hasil pengujian *driver* kipas terhadap setpoint suhu, disimpulkan bahwa kondisi kipas sudah sesuai dengan perencanaan dengan error = 0%.

4.8. Pengujian *Driver* Lampu Pemanas

4.8.1. Tujuan

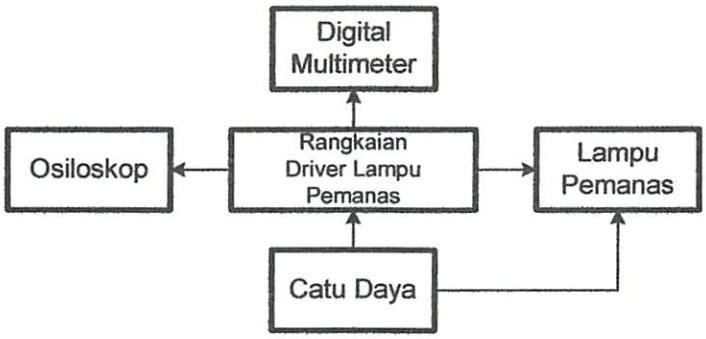
Untuk mengetahui apakah Rangkaian *Driver* Pemanas yang telah dibuat dapat difungsikan sebagai rangkaian PWM sederhana.

4.8.2. Peralatan Yang Digunakan

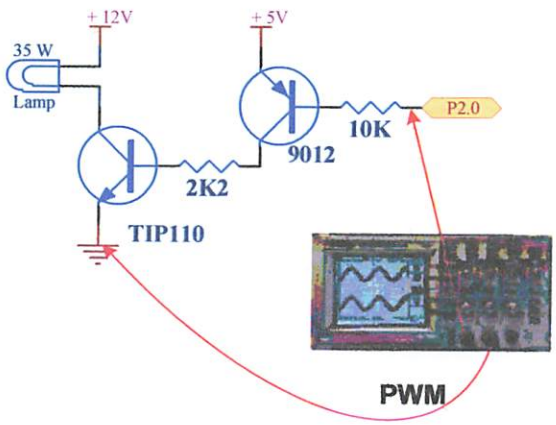
1. Rangkaian Mikrokontroller yang telah di program.
2. Rangkaian Driver Lampu Pemanas.
3. Catu Daya 5 Volt dan 12 Volt DC.
4. Digital Multimeter.
5. Osiloskop.
6. Push Button sebagai setting PWM.

4.8.3. Langkah Pengujian

1. Menghubungkan Rangkaian Mikrokontroller dengan Catu Daya 5 Volt.
2. Menghubungkan Lampu Pemanas dengan Catu Daya 12 Volt.
3. Men-setting PWM pada suhu minggu pertama, kedua dan ketiga.
4. Melihat pulsa yang dihasilkan pada masing-masing *Duty Cycle*.
5. Menghitung besarnya tegangan pada masing-masing *Duty Cycle*.



Gambar 4-13 Diagram Blok Pengujian *Driver* Lampu Pemanas



Gambar 4-14 Rangkaian Pengujian *Driver* Lampu Pemanas

4.8.4. Hasil Pengujian

Hasil pengujian Rangkaian *Driver* Lampu Pemanas adalah sebagai berikut:

Tabel 4-11 Hasil Perhitungan Rangkaian *Driver* Lampu Pemanas

Minggu	Duty Cycle (%)	Sinyal Analog (Volt)	Kondisi Lampu Pemanas
Pertama	40	4,8	Terang
Kedua	20	2,4	Sedang
Ketiga	10	1,2	Redup

Sinyal analog diatas merupakan tegangan yang dihasilkan dari besarnya *duty cycle* dalam 1 periode, besarnya tegangan tersebut dapat dicari dengan menggunakan perhitungan sebagai berikut:

V = Vcc * *Duty Cycle*

Diketahui Vcc adalah 12 Volt, maka;

Untuk minggu pertama, V = 12 * 40%

= 12 * 0,4

= 4,8 volt

Untuk minggu kedua, V = 12 * 20%

= 12 * 0,2

= 2,4 volt

Untuk minggu ketiga, V = 12 * 10%

= 12 * 0,1

= 1,2 volt

4.9. Pengujian Alat Secara Keseluruhan

4.9.1. Tujuan

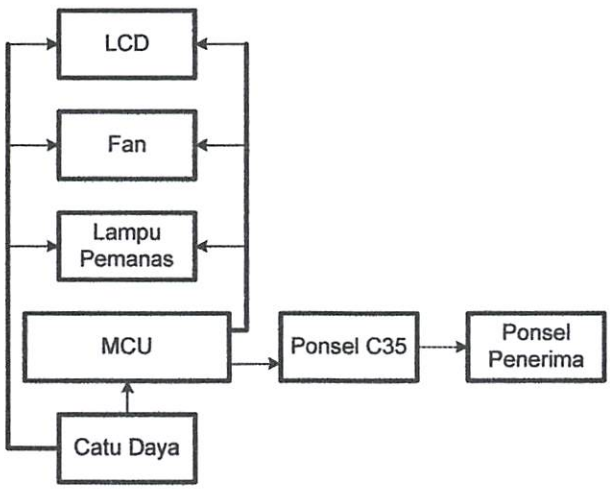
Pengujian rangkaian secara keseluruhan dilakukan untuk mengetahui keseluruhan sistem yang telah dirancang dapat bekerja dengan baik.

4.9.2. Peralatan Yang Digunakan

1. Rangkaian Sistem secara Keseluruhan.
2. Ponsel Pengirim.

4.9.3. Langkah Pengujian

1. Memberikan Catu Daya pada Rangkaian.
2. Men-setting Suhu sesuai dengan yang di inginkan.
3. Menguji Sistem SMS Peringatan dengan kondisi Lampu Pemanas Mati.
4. Menguji Sistem SMS Peringatan dengan kondisi Listrik Padam.
5. Pengecekan Suhu sesuai dengan Setpoint yang kita inginkan.



Gambar 4-15 Diagram Blok Pengujian Secara Keseluruhan

4.9.4. Hasil Pengujian

Hasil pengujian Alat secara Keseluruhan adalah sebagai berikut:

Tabel 4-12 Hasil Pengujian SMS Peringatan

Kondisi	Pengujian Ke	Tanda Peringatan	
		Status	Isi SMS
Lampu Pemanas Mati	1	Terkirim	error, lampu mati
	2	Terkirim	error, lampu mati
	3	Terkirim	error, lampu mati
	4	Terkirim	error, lampu mati
	5	Terkirim	error, lampu mati

Kondisi	Pengujian Ke	Tanda Peringatan	
		Status	Isi SMS
Listrik Padam	1	Terkirim	error, lampu mati
	2	Terkirim	error, lampu mati
	3	Terkirim	error, lampu mati
	4	Terkirim	error, lampu mati
	5	Terkirim	error, lampu mati

Dari data hasil pengujian SMS Peringatan diatas, dapat disimpulkan bahwa tidak ditemukan selisih atau error = 0%.

Tabel 4-13 Hasil Pengujian Pengecekan Suhu

Setpoint Suhu (° Celcius)		Pengujian Ke	Suhu Aktual (° Celcius)	
Minggu	Suhu yang Diinginkan		Tampilan LCD	Yang Terkirim
Pertama	35,0	1	34,8	34,8
		2	35,2	35,3
		3	35,0	35,0
		4	35,0	35,0
		5	34,9	35,0
Kedua	32,2	1	32,2	32,1
		2	31,8	31,8
		3	32,0	31,8
		4	31,8	31,8
		5	32,2	32,3
Ketiga	29,4	1	29,4	29,4
		2	29,4	29,4
		3	29,3	29,3
		4	29,2	29,2
		5	29,4	29,6

$$Kesalahanrelatif = \frac{\sum Error\%}{\sum Pengujian}$$

$$= \frac{0{,}284 + 0{,}286 + 0{,}31 + 0{,}625 + 0{,}31 + 0{,}68}{15}$$

$$= 0{,}166\ \%$$

BAB V

PENUTUP

5.1. Kesimpulan

Dari analisa dan percobaan yang dilakukan dengan alat ini, maka dapat diambil kesimpulan bahwa :

1. Ponsel Siemens type C35 dapat langsung di komunikasikan dengan Mikrokontroller AT9S8252. Komunikasi datanya adalah komunikasi data serial melalui pin TX dan RX pada masing-masing perangkat dengan memberikan setting baudrate 19200, 8 data bits, no parity dan 1 stop bit.
2. Pada ADC 0804 besarnya tegangan tiap step adalah 19,6 mV atau dibulatkan menjadi 20 mV. Sedangkan output tegangan LM35 bernilai 10 mV / °C atau 1 mV setiap 0,1 °C. Sehingga harus dikuatkan sebesar 20 kali agar setiap setiap step ADC mewakili 0,1 °C. Sedangkan pengujian suhu pada tampilan LCD dibandingkan dengan termometer air raksa diperoleh kesalahan rata-rata sebesar 3,53 %.
3. Dalam proses menampilkan waktu harus disesuaikan dengan alat penunjuk waktu. Dari hasil pengujian, yaitu perbandingan antara display waktu pada alat dengan jam digital diperoleh selama 2 hari tidak ditemukan selisih atau error = 0%.
4. Pada pengujian rangkaian *Driver* Kipas, disimpulkan bahwa kondisi kipas sudah sesuai dengan perencanaan yaitu kipas akan berputar atau ON pada saat suhu melebihi Setpoint. Dengan melakukan pengujian sebanyak 9 kali, tidak ditemukan kesalahan atau error = 0%.

5. Dari pengujian alat secara keseluruhan, dengan pengujian sebanyak 10 kali tidak terdapat kesalahan atau error = 0% pada pengujian SMS peringatan dan terdapat kesalahan relatif sebesar 0,166 % pada 15 kali pengujian SMS pengecekan suhu.

5.2. Saran

1. PWM mempunyai kelemahan yaitu timbulnya EMI *(Elektromagnetik Interference)* yang dapat mempengaruhi perangkat lain. Sebaiknya, kabel catu daya 12V untuk Lampu Pemanas yang dikendalikan oleh PWM harus di jauhkan dari kabel data Ponsel, kabel bus data LCD dan yang lebih penting lagi harus dijauhkan dari Mikrokontroller.

DAFTAR PUSTAKA

[1] www.atmel.com, AT89S8252 Datasheet, tahun 2003.

[2] www.national.com, LM35 Datasheet, tahun 2000.

[3] www.alldatasheet.com, LDR Datasheet.

[4] Caughlin, Robert F. & Driscoll, F., Penguat Operasional dan Rangkaian Terpadu Linier, Edisi Kedua, Penerbit Erlangga, Jakarta, tahun 1994.

[5] Wasito S, Vademekum Elektronika, Edisi Kedua, PT. Gramedia Pustaka Utama, Jakarta, tahun 2001.

[6] www.national.com, ADC0804 Datasheet, tahun 1995.

[7] www.alldatasheet, RTC DS12C887 Data Sheet, tahun 2000.

[8] Manual book LCD M1632, tahun 1987.

[9] www.electroniclab.com, rubrik elektronika analog, aswan hamonangan, tahun 2000.

[10] www.my-siemens.com

[11] Khang, Bustam, Trik Pemrograman Aplikasi Berbasis SMS, Jakarta, PT. Elex Media Komputindo, tahun 2002.

[12] www.Netrino.com

[13] Rasyaf, Muhammad, Beternak Ayam Pedaging, Cetakan 22, Penebar Swadaya, Jakarta, tahun 2002.

LAMPIRAN

LAMPIRAN

BISMILLAHIRRAHMAANIRRAHIIM.....

Alhamdulillah, puji syukur kehadirat Allah Subhanahu Wa Ta'ala yang senantiasa mencurahkan Rahmad, Hidayah, Ridlo dan Inayah-Nya, shalawat serta sa lam tetap tercurah kepada Junjungan nabi besar Muhammad saw..

Skripsi dan gelar sarjana ini Mufid persembahkan buat Bapak dan Ibu tercinta. Yang telah berjuang membangun keluarga dengan segala daya upaya demi mengantarkan putra putrinya merengkuh pendidikan yang layak. Beliau yakin, tidak ada warisan yang paling mulia di hadapan Allah swt. kecuali ilmu (yang bermanfaat). Dengan ilmu, seseorang bisa menjadi anak yang Shaleh, dengan ilmu seseorang bisa mengetahui mana yang hak dan mana yang bathil, dan ilmu bisa mencukupi dirinya di dunia dan akhirat. Allah Maha Tahu dengan segala maksud dan tujuan makhluk-Nya, mudah-mudahan segala usaha yang telah dilakukan bapak ibu diridloi Allah swt. dan cita-cita yang ingin dicapai senantiasa mendapat kemudahan. Terima kasih telah menjadi orang tua yang mengayomi, mendidik, menuntun serta menjadi guru, panutan, tauladan sekaligus teman yang baik buat putra putri bapak ibu. Semoga Allah swt memberikan umur yang panjang kepada bapak ibu, selalu memberikan kesehatan, kesabaran dan keikhlasan serta senantiasa menjaga bapak ibu. Amien ya Robbal 'Alamiin..

Terima kasih buat mas Misbah, mbak Ayu & si kecil Azka yang dah pinter sekarang, mudah-mudahan Allah swt selalu memberikan berkah buat keluarga mas. Makasih buat adikku tercinta, Rif, yang ngga pernah capek doain, ngingetin, nasehatin, kadang juga ngasih kuliah tujuh menit, hehe.... maaf ya.. mas belum bisa bener-bener jadi mas yang baik buat adik, tapi mas yakin, kalo mas masih belum sepenuhnya bisa jaga adik, Allah selalu menjaga adik. Amien... Makasih buat mbah Mantra yang juga ga ada capeknya doain cucu-cucunya, semoga Allah semberikan umur panjang dan kesehatan buat mbah Mantra, amien... Makasih juga buat seluruh keluarga yang telah memberikan bantuan modal, moral maupun spiritual, semoga Allah menjaga keharmonisan keluarga besar kita Amien...

nuhun pisan buat neng_Uswa, yang dah mau jadi temen, adik, kakak, pacar dan segalanya buat mas, mudah-mudahan Allah selalu ngejaga neng_Geulis, amien....

thank you so much to Eko & apid atas segalanya, dah mau ngedukung, ngasih masukan, bantuan moril spiritual, mudah-mudahan cepet lulus & merried, hehe.... okeh!!!!!

kelangkong buat temen-temen seperjuangan, Danny, Ongky, Fifi, Wawan, nanang, gondrong, Kadek, jabrik, nur, reza, sugik, agung, pakde, danang and all of you.

matur suwun sanget nggo arek-arek iwak tombro "GREATEST HOME WITH A BIG IDEA" (kambing, ali, ichang, brudin, pir, modhot, adi, donat)

terima kasih buat seluruh anak elekronika S-1 angkatan 2002 and sisa-sisa angkatan 2002 yang masih *mangkrak* di kampus (kubam, penchos, b'Doel, mas Arief, boy_oka, puput and yang laen....) semoga cepet nyusul, buat tim sepakbola ELKA 02, anak elekronika S-1 angkatan 2003, anak-anak Lab Elka & Sistem Tenaga, terima kasih.

Yang terakhir, Terima kasih buat semua baik yang sengaja, terpaksa, ikhlas, jujur, minta imbalan, banyak omong, diem gak ngenaki, tapi dah bantu Mufid tuk nyelesein kuliah dan skripsi ini. Mudah-mudahan Allah selalu dekat dengan mereka, amien..

Abu Zubar bin Bakar bercerita, Ayahku di negeri Irak berkirim surat kepadaku, "Hendaklah engaku berilmu, sebab jika sesungguhnya engkau kehilangan pekerjaan (fakir), maka ilmu itu merupakan harta bagimu. Jika engkau kaya, maka ilmu itu merupakan kebaikan bagi hidupmu"

Rasulullah saw. bersabda, "Manusia yang paling utama ialah orang yang beriman yang berilmu, yang jika dibutuhkan berkenan memberikan manfaat dan jika tidak dibutuhkan, bisa mencukupi dirinya sendiri"

Pada kesempatan lain Rasulullah saw. bersabda, "Iman itu telanjang, pakaiannya adalah taqwa, perhiasaannya adalah rasa malu, dan buahnya adalah ilmu"

Rasulullah saw. juga bersabda, "Manusia yang paling dekat dengan derajat kenabian yaitu ahli ilmudan jihad"

Al Hasan ra. Berkata, "Kelak tinta orang yang berilmu akan ditimbang dengan darah orang-orang yang mati syahid, ternyata tinta ulama (orang yang berilmu) mengungguli darah para syuhada (orang yang mati syahid)"

" Imagination is more important than knowledge.
Knowledge is limited, Imagination encircles the world " (Albert Einstein)

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1

FORMULIR BIMBINGAN SKRIPSI

Nama : Moh. Mufid Murtadlo

NIM : 02 17 144

Masa Bimbingan : 9 Juni 2006 – 9 Desember 2006

Judul Skipsi : PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING BERBASIS AT89s8252 YANG DI INFORMASIKAN VIA SMS

No.	Tanggal	Uraian	Paraf Pembimbing
1	3/8 2006	Bab I & Bab II perbaiki Rumusan	
2			
3			
4			
5			
6			
7			
8			
9			

Malang,
Dosen Pembimbing :

Ir. F. Yudi Limpraptono, MT
NIP. Y. 1039500274

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1

FORMULIR BIMBINGAN SKRIPSI

Nama : Moh. Mufid Murtadlo

NIM : 02 17 144

Masa Bimbingan : 9 Desember 2006 – 9 Juni 2007

Judul Skripsi : PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING BERBASIS AT98s8252 YANG DI INFORMASIKAN VIA SMS

No.	Tanggal	Uraian	Paraf Pembimbing
1	21/2 2007	Bab IV + V	
2	28/2 2007	Demo	
3	3/3 2007	Persiapan Konpre	
4			
5			
6			
7			
8			
9			
10			

Malang.
Dosen Pembimbing;

Ir. F. Yudi Limpraptono, MT
NIP. Y. 1039500274

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1

FORMULIR BIMBINGAN SKRIPSI

Nama : Moh. Mufid Murtadlo

NIM : 02 17 144

Masa Bimbingan : 9 Desember 2006 – 9 Juni 2007

Judul Skripsi : PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING BERBASIS AT98s8252 YANG DI INFORMASIKAN VIA SMS

No.	Tanggal	Uraian	Paraf Pembimbing
1		Bab I - II ok	
2		Bab III ok	
3		Bab IV perbaiki	
4		Bab V perbaiki	
5		Tolong tambahkan foto alat, hasil pengujian, rangkaian lengkap	
6		Seminar hasil Acc	
7		Acc kompre	
8			
9			
10			

Malang,
Dosen Pembimbing;

Ir. Mimien Mustikawati
NIP. Y. 1030000352

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1

FORMULIR BIMBINGAN SKRIPSI

Nama : Moh. Mufid Murtadlo

NIM : 02 17 144

Masa Bimbingan : 9 Juni 2006 – 9 Desember 2006

Judul Skipsi : PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING BERBASIS AT89s8252 YANG DI INFORMASIKAN VIA SMS

No.	Tanggal	Uraian	Paraf Pembimbing
1			
2			
3			
4			
5			
6			
7			
8			
9			

Malang,
Dosen Pembimbing ;

Ir. Mimien Mustikawati
NIP. P. 1030000352

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

LEMBAR BIMBINGAN SKRIPSI

NAMA	: Moh. Mufid Murtadlo
NIM	: 02 17 144
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 9 Desember 2006 s/d 9 Juni 2007
Judul Skripsi	: Perencanaan Dan Pembuatan Alat Kontrol Suhu Pada Indukan Ayam Pedaging Berbasis AT 89S8252 Yang Diinformasikan Via SMS
Dosen Pembimbing	: Ir. F. Yudi Limpraptono, MT.
Telah Dievaluasi	: 90 (Sembilan Puluh)

Diperiksa dan Disetujui,
Dosen Pembimbing I

(Ir. F. Yudi Limpraptono, MT.)
NIP.Y. 1039500274

Mengatahui,
Ketua Jurusan Teknik Elektro S-1

(Ir. F. Yudi Limpraptono, MT.)
NIP.Y. 1039500274

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

LEMBAR BIMBINGAN SKRIPSI

NAMA	: Moh. Mufid Murtadlo
NIM	: 02 17 144
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 9 Desember 2006 s/d 9 Juni 2007
Judul Skripsi	: Perencanaan Dan Pembuatan Alat Kontrol Suhu Pada Indukan Ayam Pedaging Berbasis AT 89S8252 Yang Diinformasikan Via SMS
Dosen Pembimbing	: Ir. Mimien Mustikawati
Telah Dievaluasi	: 90 (Sembilan Puluh)

Diperiksa dan Disetujui,
Dosen Pembimbing II

(Ir. Mimien Mustikawati)
NIP.P. 1030000352

Mengatahui,
Ketua Jurusan Teknik Elektro S-1

(Ir. F. Yudi Limpraptono, MT)
NIP.Y. 1039500274

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentasi T. Energi Listrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : Moh. Mujid Murtado
NIM : 02.17.144
Perbaikan meliputi :

Gambar fungsional.
- Driver motor (Roda)
- UC pin EA.!

Malang,

(Komang S.)

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

alam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentasi T. Energi
istrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

AMA : MOH MUFID
I M : 0217144
erbaikan meliputi :

Data alat daly cycle di lengkapi

Malang,

(Ekas W)

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama	: Moh. Mufid Murtadlo
NIM	: 02 17 144
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 9 Desember 2006 s/d 9 Juni 2007
Judul Skripsi	: Perencanaan Dan Pembuatan Alat Kontrol Suhu Pada Indukan Ayam Pedaging Berbasis AT 89S8252 Yang Diinformasikan Via SMS

Tanggal	Uraian	Paraf
16 Maret 2007	❖ Driver Motor (Dioda) ❖ Mikrokontroler pin EA	
	❖ Data untuk Duty Cycle dihilangkan	

Mengetahui,

Dosen Pembimbing I

(Ir. F. Yudi Limpraptono, MT.)
NIP.Y. 1039500274

Dosen Pembimbing II

(Ir. Mimien Mustikawati)
NIP.P. 1030000352

Dosen Penguji,

Penguji 1

(Ir. Eko Nurcahyo)
NIP.Y. 1028700172

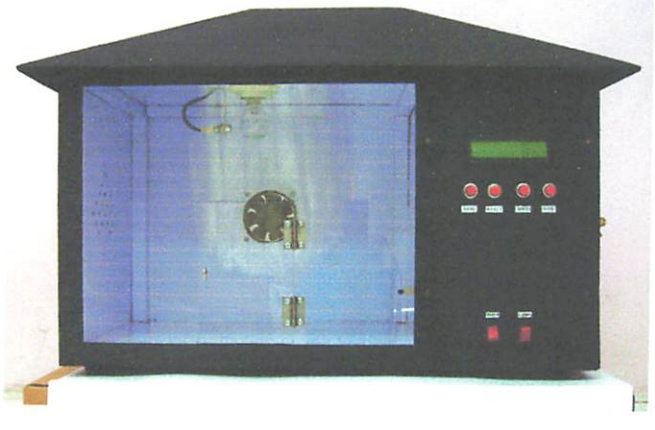
Penguji 2

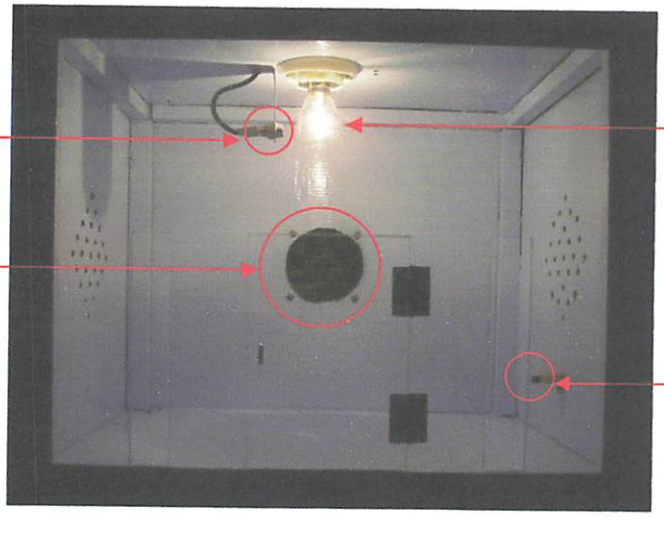
(I Komang Somawirata, ST., MT.)
NIP.P. 1030100361

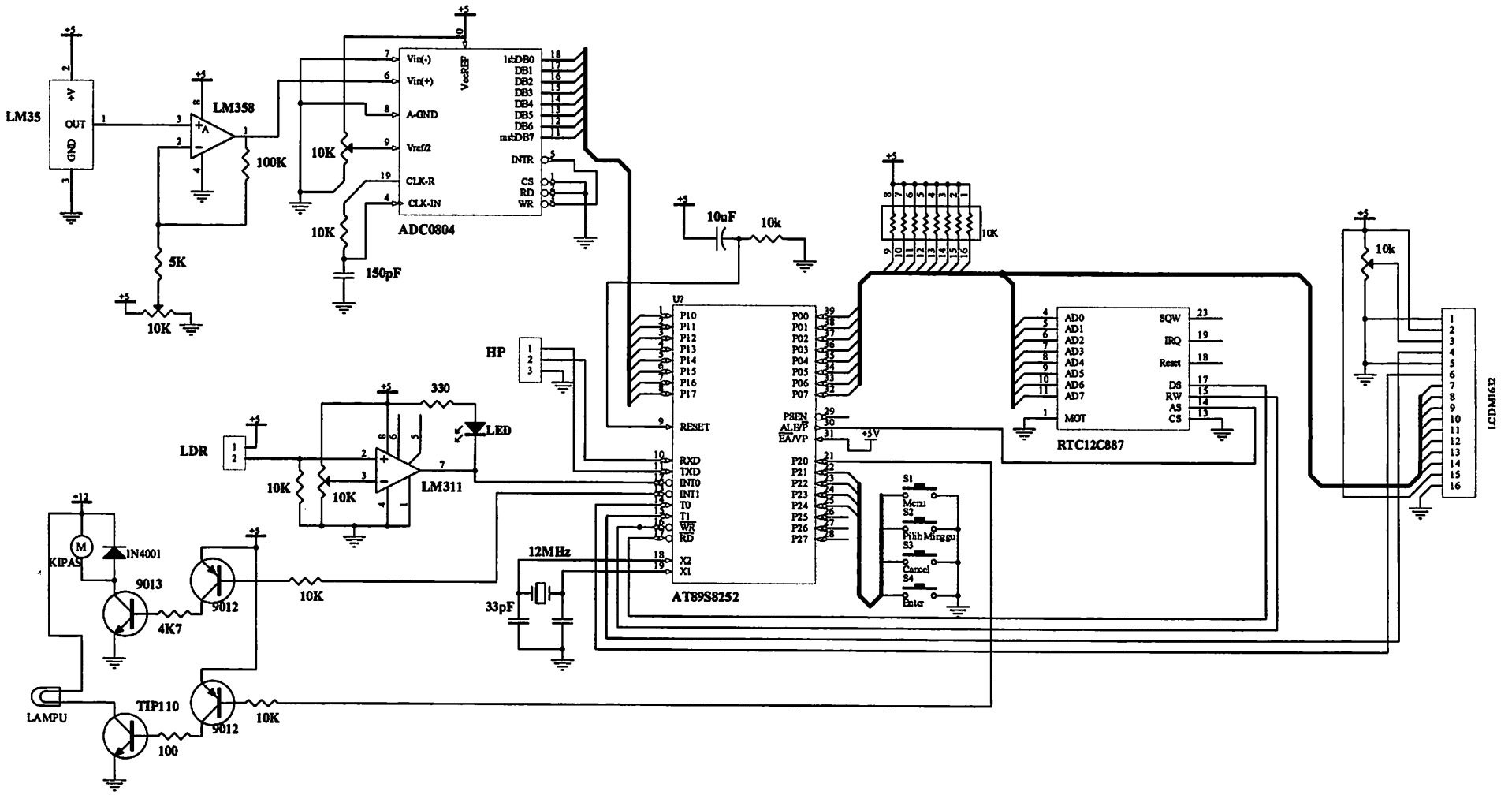
MINIATUR INDUKAN AYAM PEDAGING

Spesifikasi Alat :

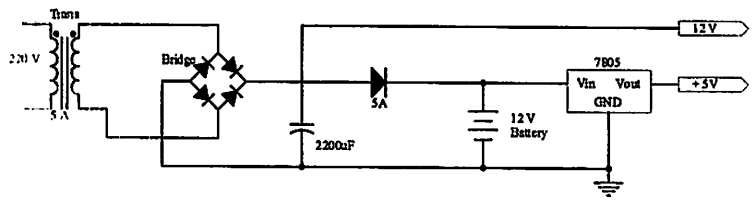
1. Input Tegangan : 220 Volt AC /60 Hz
2. Input Tegangan / Arus DC : +5 V / 500 mA , +12 V / 3 A
3. Ukuran :
   - Panjang : 50 CM
   - Lebar : 30 CM
   - Tinggi : 35 CM
4. Input :
   - Cahaya.
   - Panas.
5. Output :
   - Intensitas Lampu.
   - Exhaust Fan.
   - LCD M1632.
   - SMS.







Title	Rangkaian Lengkap		
Size B	Number		Revision
Date: 3/18/2007		Sheet of	
File: H:\Artikel\..\rangkaian lengkap.SCHDOC		Drawn By: Musd / 0217144	



Title		
Size B	Number	Revision
Date: 3/15/2007		Sheet of
File: H:\Atikeh...\ccatu daya 000.SCHDOC		Drawn By:

```
;=========================================================================
; PERENCANAAN DAN PEMBUATAN ALAT KONTROL SUHU PADA INDUKAN AYAM PEDAGING
;              BERBASIS AT89S8252 YANG DIINFORMASIKAN VIA SMS
;                     MOH. MUFID MURTADLO / 02 17 144
;=========================================================================
;
            org     0h
            ljmp    init            ; long jump ke init
;
            org     23h             ; menunjukkan alamat serial interrupt
            clr     ES
            jnb     RI,$            ; lompat ke dia sendiri jika belum 0
            clr     RI              ; jadikan R1 = 0
            mov     R1,SBUF         ; pindahkan data SBUF ke accumulator
            setb    ES              ; jadikan ES = 1
            reti                    ; kembali ke interrupt
;
            Lamp    Bit P2.0        ; port 2.0 u/ driver lampu
            Tbl0    Bit P2.1        ; port 2.1 u/ tombol 0 (menu)
            Tbl1    Bit P2.2        ; port 2.2 u/ tombol 1 (select)
            Tbl2    Bit P2.3        ; port 2.3 u/ tombol 2 (cancel)
            Tbl3    Bit P2.4        ; port 2.4 u/ tombol 3 (enter)
            In05    Bit P2.5        ; \
            In06    Bit P2.6        ; : sisa port
            In07    Bit P2.7        ; /
            Ldrs    Bit P3.2        ; port 3.2 u/ sensor cahaya
            Fans    Bit P3.3        ; port 3.3 u/ driver kipas
            Rest    Bit P3.4        ; port 3.4 reset
            Enbl    Bit P3.5        ; port 3.5 enable
            RegA    Equ 0Ah         ; register A disimpan di address 0Ah
            RegB    Equ 0Bh         ; register B disimpan di address 0Bh
            RegC    Equ 0Ch         ; register C disimpan di address 0Ch
            RegD    Equ 0Dh         ; register D disimpan di address 0Dh
            Secd    Equ 00h         ; data detik disimpan di address 00h
            Mint    Equ 02h         ; data menit disimpan di address 02h
            Hour    Equ 04h         ; data jam disimpan di address 04h
            Week    Equ 06h         ; data minggu disimpan di address 06h
            Date    Equ 07h         ; data hari disimpan di address 07h
            Mont    Equ 08h         ; data bulan disimpan di address 08h
            Year    Equ 09h         ; data tahun disimpan di address 09h
            Jam0    Equ 30h         ; data jam0 disimpan di address 30h
            Jam1    Equ 31h         ; data jam1 disimpan di address 31h
            Jam2    Equ 32h         ; data jam2 disimpan di address 32h
            Mnt0    Equ 33h         ; data menit0 disimpan di address 33h
            Mnt1    Equ 34h         ; data menit1 disimpan di address 34h
            Mnt2    Equ 35h         ; data menit2 disimpan di address 35h
            Dtk0    Equ 36h         ; data detik0 disimpan di address 36h
            Dtk1    Equ 37h         ; data detik1 disimpan di address 37h
            Dtk2    Equ 38h         ; data detik2 disimpan di address 38h
            Hri0    Equ 39h         ; data hari0 disimpan di address 39h
            Hri1    Equ 3Ah         ; data hari1 disimpan di address 3Ah
            Hri2    Equ 3Bh         ; data hari2 disimpan di address 3Bh
            Tgl0    Equ 3Ch         ; data tanggal0 disimpan di address 3Ch
            Tgl1    Equ 3Dh         ; data tanggal1 disimpan di address 3Dh
            Tgl2    Equ 3Eh         ; data tanggal2 disimpan di address 3Eh
            Bln0    Equ 3Fh         ; data bulan0 disimpan di address 3Fh
            Bln1    Equ 40h         ; data bulan1 disimpan di address 40h
            Bln2    Equ 41h         ; data bulan2 disimpan di address 41h
            Thn0    Equ 42h         ; data tahun0 disimpan di address 42h
            Thn1    Equ 43h         ; data tahun1 disimpan di address 43h
            Thn2    Equ 44h         ; data tahun2 disimpan di address 44h
            Cks0    Equ 45h         ; cks0 disimpan di address 45h
            Cks1    Equ 46h         ; cks1 disimpan di address 46h
            Cks2    Equ 47h         ; cks2 disimpan di address 47h
            Cks3    Equ 48h         ; cks3 disimpan di address 48h
            Cks4    Equ 49h         ; cks4 disimpan di address 49h
            Cks5    Equ 4Ah         ; cks5 disimpan di address 4Ah
            Dshu    Equ 4Bh         ; data suhu disimpan di address 4Bh
            Dsh0    Equ 4Ch         ; data suhu0 disimpan di address 4Ch
            Dsh1    Equ 4Dh         ; data suhu1 disimpan di address 4Dh
            Dsh2    Equ 4Eh         ; data suhu2 disimpan di address 4Eh
            Dsh3    Equ 4Fh         ; data suhu3 disimpan di address 4Fh
            Dbts    Equ 50h         ; batas suhu disimpan di address 50h
            Dbt0    Equ 51h         ; batas suhu0 disimpan di address 51h
```

```
                Hurf    Equ 52h         ; huruf disimpan di address 52h
                Cntr    Equ 53h         ; century disimpan di address 53h
                Dmgg    Equ 54h         ; data minggu disimpan di address 54h
                Dhrr    Equ 55h         ; data hari disimpan di address 55h
                Ster    Equ 56h         ; data error disimpan di address 56h
                Dly0    Equ 57h         ; delay0 disimpan di address 57h
                Dly1    Equ 58h         ; delay1 disimpan di address 58h
                Dly2    Equ 59h         ; delay2 disimpan di address 59h
                Dly3    Equ 5Ah         ; delay3 disimpan di address 5Ah
                Dly4    Equ 5Bh         ; delay4 disimpan di address 5Bh
;
;=====================================================================
;               INISIALISASI
;=====================================================================
;
init:           mov     Dmgg,#3         ; isi reg Dmgg dengan 3
                mov     Ster,#0         ; reset status error
                acall   lcd_in          ; panggil lcd_in
                acall   rtc_in          ; panggil rtc_in
                acall   srl_in          ; panggil srl_in
                acall   sms_in          ; info incoming SMS [1,1,0,0,1]
                acall   rdmghr          ; baca RTC -> status minggu & hari
;
;=====================================================================
;               TAMPILAN LOGO
;=====================================================================
;
mulai:          mov     DPTR,#logo      ; pindahkan data logo ke DPTR
                acall   line1           ; panggil line1
                mov     hurf,#16        ; isi register hurf dengan 16
                acall   tulis           ; panggil tulis
                acall   line2           ; panggil line2
                mov     hurf,#16        ; isi regster hurf dengan 16
                acall   tulis           ; panggil tulis
                acall   delay1          ; panggil delay1
                acall   line1           ; panggil line1
                mov     hurf,#16        ; isi register hurf dengan 16
                acall   tulis           ; panggil tulis
                acall   line2           ; panggil line2
                mov     hurf,#16        ; isi register hurf dengan 16
                acall   tulis           ; panggil tulis
                acall   delay1          ; panggil delay1
;
;=====================================================================
;               TAMPILAN WAKTU
;=====================================================================
;
                mov     DPTR,#tptim     ; pindahkan data tptim ke DPTR
                acall   line1           ; panggil line1
                mov     hurf,#16        ; isi register hurf dengan 16
                acall   tulis           ; panggil tulis
                acall   bc_jam          ; panggil bc_jam
                mov     DPTR,#angka     ; pindahkan angka ke DPTR
                acall   line2           ; panggil line2
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,Jam1          ; baca data Jam1, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,Jam2          ; baca data Jam2, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,#13           ; isi Akumulator dengan 13
                acall   wr_chr          ; panggil wr_chr
                mov     A,Mnt1          ; baca data mnt1, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,Mnt2          ; baca data Mnt2, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,Tgl1          ; baca data Tgl1, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,Tgl2          ; baca data Tgl2, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,#14           ; isi Akumulator dengan 14
                acall   wr_chr          ; panggil wr_chr
```

```
                mov     A,Bln1          ; baca data Bln1, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,Bln2          ; baca data Bln2, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,#14           ; isi Akumulator dengan 14
                acall   wr_chr          ; panggil wr_chr
                mov     A,Thn1          ; baca data Thn1, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,Thn2          ; baca data Thn2, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                acall   delay1          ; panggil delay1
;
;=====================================================================
;               TAMPILAN MINGGU/HARI
;=====================================================================
;
loop:           acall   rdmghr          ; panggil rdmghr/baca minggu,hari
                mov     DPTR,#mggke     ; pindahkan data mggke ke DPTR
                acall   line1           ; panggil line1
                mov     hurf,#12        ; isi register hurf dengan 12
                acall   tulis           ; panggil tulis
                mov     DPTR,#angka     ; pindahkan data angka ke DPTR
                mov     A,Dmgg          ; baca data Dmgg, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     DPTR,#harke     ; pindahkan data harke ke DPTR
                acall   line2           ; panggil line2
                mov     hurf,#11        ; isi register hurf dengan 11
                acall   tulis           ; panggil tulis
                mov     DPTR,#angka     ; pindahkan data angka ke DPTR
lmphr0:         mov     A,Dmgg          ; baca data Dmgg, simpan isinya di A
                cjne    A,#15,lmphr1    ; apakah Akumulator=15? Jika tidak lompat ke lmphr1
                acall   wr_chr          ; panggil wr_chr
                sjmp    lmphr2          ; lompat ke lmphr2
lmphr1:         mov     A,Dhrr          ; baca data Dhrr, simpan isinya di A
                acall   wr_chr          ; panggil wr_chr
lmphr2:         mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                acall   delay1          ; panggil delay1
;
;=====================================================================
;               TAMPILAN SUHU
;=====================================================================
;
                mov     DPTR,#tpshu     ; pindahkan data tpshu ke DPTR
                acall   line1           ; panggil line1
                mov     hurf,#16        ; isi register hurf dengan 16
                acall   tulis           ; panggil tulis
                acall   bc_adc          ; panggil bc_adc
                acall   cacah           ; panggil cacah
                mov     DPTR,#angka     ; pindahkan data angka ke DPTR
                acall   line2           ; panggil line2
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,#10           ; isi Akumulator dengan 10
                acall   wr_chr          ; panggil wr_chr
                mov     A,Dsh0          ; baca data Dshu, simpan isinya di A
```

```
            acall   wr_chr          ; panggil wr_chr
            mov     A,Dsh1          ; baca data Dsh1, simpan isinya di A
            acall   wr_chr          ; panggil wr_chr
            mov     A,Dsh2          ; baca data Dsh2, simpan isinya di A
            acall   wr_chr          ; panggil wr_chr
            mov     A,#11           ; isi Akumulator dengan 11
            acall   wr_chr          ; panggil wr_chr
            mov     A,Dsh3          ; baca data Dsh3, simpan isinya di A
            acall   wr_chr          ; panggil wr_chr
            mov     R0,#0DFh        ; isi register R0 dengan 0DFh
            acall   w_chr           ; panggil wr_chr
            mov     A,#12           ; isi Akumulator dengan 12
            acall   wr_chr          ; panggil wr_chr
            mov     A,#10           ; isi Akumulator dengan 10
            acall   wr_chr          ; panggil wr_chr
            mov     A,#10           ; isi Akumulator dengan 10
            acall   wr_chr          ; panggil wr_chr
            mov     A,#10           ; isi Akumulator dengan 10
            acall   wr_chr          ; panggil wr_chr
            mov     A,#10           ; isi Akumulator dengan 10
            acall   wr_chr          ; panggil wr_chr
            mov     A,#10           ; isi Akumulator dengan 10
            acall   wr_chr          ; panggil wr_chr
            acall   delay1          ; panggil delay1
            ljmp    loop            ; lompat ke loop
;
;==========================================================================
;           MENU
;==========================================================================
;
menu:       mov     DPTR,#tmenu     ; pindahkan data tmenu ke DPTR
            acall   line1           ; panggil line1
            mov     hurf,#16        ; isi register hurf dengan 16
            acall   tulis           ; panggil tulis
mnlp0:      cjne    R4,#1,mnlp1     ; apakah R4=1? Jika tidak lompat ke mnlp1
            mov     DPTR,#plmg1     ; jika ya, pindahkan data plmg1 ke register DPTR
            mov     R5,#1           ; register buffer minggu 1
mnlp1:      cjne    R4,#2,mnlp2     ; apakah R4=2? Jika tidak lompat ke mnlp2
            mov     DPTR,#plmg2     ; jika ya, pindahkan data plmg2 ke register DPTR
            mov     R5,#2           ; register buffer minggu 2
mnlp2:      cjne    R4,#3,mnlp3     ; apakah R4=3? Jika tidak lompat ke mnlp3
            mov     DPTR,#plmg3     ; jika ya, pindahkan data plmg3 ke register DPTR
            mov     R5,#3           ; register buffer minggu 3
mnlp3:      cjne    R4,#4,mnlp4     ; apakah R4=4? Jika tidak lompat ke mnlp4
            mov     DPTR,#mtsys     ; jika ya, pindahkan data mtsys ke register DPTR
            mov     R5,#15          ; isi register R5 dengan 15
mnlp4:      acall   line2           ; panggil line2
            mov     Hurf,#16        ; isi register Hurf dengan 16
            acall   tulis           ; panggil tulis
mnlp5:      acall   jeda            ; panggil jeda
            jb      Tbl1,mnlp7      ; apakah Tbl1 ditekan? Jika tidak
                                    ; cek tombol lain di mnlp7
tglps0:     acall   jeda            ; panggil jeda
            jnb     Tbl1,tglps0     ; apakah Tbl1 ditekan? Jika ya lompat ke tglps0
            inc     R4              ; jika  tidak, tambahkan isi register R4 dengan 1
            cjne    R4,#4,mnlp6     ; apakah R4=4? Jika tidak lompat ke mnlp6
            mov     R4,#1           ; jika ya, isi register R4 dengan 1
mnlp6:      sjmp    mnlp0           ; lompat ke mnlp0
mnlp7:      jb      Tbl2,mnlp8      ; apakah Tbl2 ditekan? Jika tidak
                                    ; cek tombol lain di mnlp8
tglps1:     acall   jeda            ; panggil jeda
            jnb     Tbl2,tglps1     ; apakah Tbl2 ditekan? Jika ya lompat ke tglps1
            ljmp    mulai           ; jika tidak, lompat ke mulai
mnlp8:      jb      Tbl3,mnlp5      ; apakah Tbl3 ditekan? Jika tidak
                                    ; cek tombol lain di mnlp5
tglps2:     acall   jeda            ; panggil jeda
            jnb     Tbl3,tglps2     ; apakah Tbl3 ditekan? Jika ya lompat ke tglps2
            mov     Dmgg,R5         ; jika tidak, baca data R5, simpan
                                    ; isinya di register Dmgg
            acall   wrmghr          ; panggil wrmghr
            acall   delayk          ; panggil delayk
            ljmp    mulai           ; lompat ke mulai
```

```
;
;=====================================================================
;               TULIS MINGGU/HARI
;=====================================================================
;
wrmghr:         mov     DPTR,#RegB      ; pindahkan data RegB ke DPTR
                mov     A,#96h          ; isi Akumulator dengan 96
                movx    @DPTR,A         ; salin data eksternal Akumulator ke lokasi DPTR
                mov     DPTR,#14h       ; address RTC untuk data minggu
                mov     A,Dmgg          ; baca data Dmgg, simpan isinya di A
                movx    @DPTR,A         ; salin data eksternal Akumulator ke lokasi DPTR
                mov     DPTR,#Week      ; pindahkan data Week ke DPTR
                mov     A,#01           ; isi Akumulator dengan 1
                movx    @DPTR,A         ; salin data eksternal Akumulator ke lokasi DPTR
                mov     DPTR,#RegB      ; pindahkan data RegB ke DPTR
                mov     A,#16h          ; isi Akumulator dengan 16
                movx    @DPTR,A         ; salin data eksternal Akumulator ke lokasi DPTR
                ret
;
rdmghr:         mov     DPTR,#14h       ; address RTC untuk data minggu
                movx    A,@DPTR         ; salin data eksternal DPTR ke lokasi akumulator
                mov     Dmgg,A          ; baca data akumulator, simpan isinya di register Dmgg
                mov     DPTR,#Week      ; pindahkan data Week ke DPTR
                movx    A,@DPTR         ; salin data eksternal DPTR ke lokasi akumulator
                mov     Dhrr,A          ; baca data akumulator, simpan isinya
                                        ; di register Dhrr
                ret
;=====================================================================
;               BACA SMS
;=====================================================================
;
bc_sms:         acall   delayk          ; panggil delayk
                mov     DPTR,#smbca     ; pindahkan data smsbca ke DPTR
                mov     Hurf,#9         ; isi register Hurf dengan 9
                acall   kr_ins          ; panggil kr_ins
                mov     Cntr,#82        ; isi register Cntr dengan 82
tg_sms:         acall   bc_ins          ; panggil bc_ins
                djnz    Cntr,tg_sms     ; kurangi isi Cntr, jika belum nol ulangi lagi
                acall   bc_ins          ; panggil bc_ins
                mov     Cks0,R1         ; baca data R1, simpan isinya di Cks0
                acall   bc_ins          ; panggil bc_ins
                mov     Cks1,R1         ; baca data R1, simpan isinya di Cks1
                acall   bc_ins          ; panggil bc_ins
                mov     Cks2,R1         ; baca data R1, simpan isinya di Cks2
                acall   bc_ins          ; panggil bc_ins
                mov     Cks3,R1         ; baca data R1, simpan isinya di Cks3
                acall   bc_ins          ; panggil bc_ins
                mov     Cks4,R1         ; baca data R1, simpan isinya di Cks4
                acall   bc_ins          ; panggil bc_ins
                mov     Cks5,R1         ; baca data R1, simpan isinya di Cks5
                acall   delayk          ; panggil delayk
;
;=====================================================================
;               HAPUS SMS
;=====================================================================
;
                mov     DPTR,#smhps     ; pindahkan data smhps ke DPTR
                mov     Hurf,#9         ; isi register Hurf dengan 9
                acall   kr_ins          ; panggil kr_ins
                acall   delayk          ; panggil delayk
;
;=====================================================================
;               PENGECEKAN SMS MASUK
;=====================================================================
;
ck_sm0:         mov     DPTR,#dtcek     ; pindahkan data dtcek ke DPTR
                clr     C
                mov     A,#0            ; isi Akumulator dengan 0
                movc    A,@A+DPTR       ; DPTR sebagai register tak langsung
                mov     B,Cks0          ; baca data Cks0, simpan isinya di register B
                subb    A,B             ; kurangi A dengan B
                cjne    A,#0,ck_sm1     ; apakah A=0? Jika tidak lompat ke ck_sm1
                clr     C
                mov     A,#1            ; isi Akumulator dengan 1
```

```
                movc    A,@A+DPTR       ; DPTR sebagai register tak langsung
                mov     B,Cks1          ; baca data Cks1, simpan isinya di register B
                subb    A,B             ; kurangi A dengan B
                cjne    A,#0,ck_sm1     ; apakah A=0? Jika tidak lompat ke ck_sm1
                clr     C
                mov     A,#2            ; isi Akumulator dengan 2
                movc    A,@A+DPTR       ; DPTR sebagai register tak langsung
                mov     B,Cks2          ; baca data Cks2, simpan isinya di register B
                subb    A,B             ; kurangi A dengan B
                cjne    A,#0,ck_sm1     ; apakah A=0? Jika tidak lompat ke ck_sm1
                clr     C
                mov     A,#3            ; isi Akumulator dengan 3
                movc    A,@A+DPTR       ; DPTR sebagai register tak langsung
                mov     B,Cks3          ; baca data Cks3, simpan isinya di regiser B
                subb    A,B             ; kurangi A dengan B
                cjne    A,#0,ck_sm1     ; apakah A=0? Jika tidak lompat ke ck_sm1
                clr     C
                mov     A,#4            ; isi Akumulator dengan 4
                movc    A,@A+DPTR       ; DPTR sebagai register tak langsung
                mov     B,Cks4          ; baca data Cks4, simpan isinya di
                subb    A,B             ; kurangi A dengan B
                cjne    A,#0,ck_sm1     ; apakah A=0? Jika tidak lompat ke ck_cm1
                clr     C
                mov     A,#5            ; isi Akumulator dengan 5
                movc    A,@A+DPTR       ; DPTR sebagai register tak langsung
                mov     B,Cks5          ; baca data Cks5, simpan isinya di
                subb    A,B             ; kurangi A dengan B
                cjne    A,#0,ck_sm1     ; apakah A=0? Jika tidak lompat ke ck_sm1
;
                mov     DPTR,#smk24     ; pindahkan data smk24 ke DPTR
                mov     Hurf,#10        ; isi register Hurf dengan 10
                acall   kr_ins          ; panggil kr_ins
                acall   delayk          ; panggil jeda
                acall   bc_adc          ; panggil bc_adc
                mov     R2,A            ; salin data akumulator ke R2 (data suhu -> R2)
                mov     R3,Dsh3         ; salin data Dsh3 ke R3 (angka dibelakang koma -> R3)
                acall   klbrs           ; panggil klbrs
                mov     Hurf,#50        ; isi register Hurf dengan 50
                acall   kr_dta          ; panggil kr_dta
                acall   delayk          ; panggil jeda
ck_sm1:         mov     R1,#0FFh        ; reset incoming data serial
                ljmp    mulai           ; lompat ke mulai
;
;=====================================================================
;               KIRIM ERROR
;=====================================================================
;
kr_err:         mov     A,Ster          ; status error
                cjne    A,#0,krerr      ; apakah Akumulator=0? Jika tidak lompat ke krerr
                mov     DPTR,#smk29     ; pindahkan data smk29 ke DPTR
                mov     Hurf,#10        ; isi register Hurf dengan 10
                acall   kr_ins          ; panggil kr_ins
                acall   delayk          ; panggil delayk
                mov     DPTR,#dterr     ; pindahkan data dterr ke DPTR
                mov     Hurf,#60        ; isi register Hurf dengan 60
                acall   kr_dta          ; panggil kr_dta
                acall   delayk          ; panggil delayk
                mov     Ster,#1         ; isi register Ster dengan 1
krerr:          ret
;
;=====================================================================
;               CEK BATAS SUHU
;=====================================================================
;
cekbts:         acall   bc_adc          ; panggil bc_adc
                mov     A,Dshu          ; salin isi Dshu ke Akumulator
                mov     B,Dbts          ; salin isi Dbts ke register B
                div     AB              ; bagi A dengan B
ckbt0:          cjne    A,#0,ckbt2      ; apakah A=0? Jika tidak lompat ke ckbt2
ckbt1:          setb    Fans            ; jika ya, matikan kipas
                sjmp    ckbt4           ; lompat ke ckbt4
ckbt2:          mov     A,Dshu          ; salin isi Dshu ke Akumulator
                mov     B,Dbts          ; salin isi Dbts ke register B
                subb    A,B             ; kurangi A dengan b
```

```
                cjne    A,#0,ckbt3      ; apakah A=0, jika tidak lompat ke ckbt3
                mov     A,Dsh3          ; salin data Dsh3 ke A (suhu dibelakang koma)
                mov     B,Dbt0          ; salin data Dbt0 ke B (batas suhu dibelakang koma)
                div     AB              ; bagi A dengan B
                cjne    A,#0,ckbt3      ; apakah A=0, jika tidak lompat ke ckbt3
                sjmp    ckbt1           ; lompat ke ckbt1
ckbt3:          clr     Fans            ; nyalakan kipas (aktif low)
ckbt4:          ret
;
;=====================================================================
;               BACA ADC
;=====================================================================
;
bc_adc:         mov     A,P1            ; salin isi P1 ke Akumulator
                mov     B,#10           ; suhu 25-40 -> ADC 27-177
                div     AB              ; data ADC dibagi dg 10
                mov     Dsh3,B          ; sisa = angka dibelakang koma
                mov     B,#25           ; hasil ditambah 25
                add     A,B             ; tambahkan A dengan B
                mov     Dshu,A          ; salin isi Akumulator ke Dshu
                ret
;
cacah:          mov     B,#100          ; isi register B dengan 100
                div     AB              ; bagi A dengan B (data ADC dibagi 10)
                mov     Dsh0,A          ; sisa = angka dibelakang koma
                mov     A,B             ; salin isi register B ke Akumulator
                mov     B,#10           ; isi regiater B dengan 10
                div     AB              ; bagi A dengan B
                mov     Dsh1,A          ; salin isi Akumulator ke Dsh1
                mov     Dsh2,B          ; salin isi register B ke Dsh2
hps0:           mov     A,Dsh0          ; salin isi Dsh0 ke Akumulator
                cjne    A,#0,hps1       ; apakah A=0? Jika tidak lompat ke hps1
                mov     Dsh0,#10        ; jika ya, isi Dsh0 dengan 10
hps1:           mov     A,Dsh1          ; salin isi Dsh1 ke Akumulator
                cjne    A,#0,hps2       ; apakah A=0? Jika tidak lompat ke hps2
                mov     A,Dsh0          ; jika ya, salin isi Dsh0 ke Akumulator
                cjne    A,#10,hps2      ; apakah A=10? Jika tidak lompat ke hps2
                mov     Dsh1,#10        ; jika ya, isi Dsh1 dengan 10
hps2:           ret
;
;=====================================================================
;               BACA RTC
;=====================================================================
;
bc_jam:         mov     DPTR,#Hour
                movx    A,@DPTR
                mov     Jam0,A
                mov     B,#10
                div     AB
                mov     Jam1,A
                mov     Jam2,B
bc_mnt:         mov     DPTR,#Mint
                movx    A,@DPTR
                mov     Mnt0,A
                mov     B,#10
                div     AB
                mov     Mnt1,A
                mov     Mnt2,B
bc_dtk:         mov     DPTR,#Secd
                movx    A,@DPTR
                mov     Dtk0,A
                mov     B,#10
                div     AB
                mov     Dtk1,A
                mov     Dtk2,B
bc_tgl:         mov     DPTR,#Date
                movx    A,@DPTR
                mov     Tgl0,A
                mov     B,#10
                div     AB
                mov     Tgl1,A
                mov     Tgl2,B
bc_bln:         mov     DPTR,#Mont
                movx    A,@DPTR
```

```
                mov     Bln0,A
                mov     B,#10
                div     AB
                mov     Bln1,A
                mov     Bln2,B
bc_thn:         mov     DPTR,#Year
                movx    A,@DPTR
                mov     Thn0,A
                mov     B,#10
                div     AB
                mov     Thn1,A
                mov     Thn2,B
                ret
;
;==========================================================================
;               INISIALISASI RTC
;==========================================================================
;
rtc_in:         mov     DPTR,#RegA
                mov     A,#20h
                movx    @DPTR,A
                mov     DPTR,#RegB
                mov     A,#16h
                movx    @DPTR,A
                acall   delays
                ret
;
;==========================================================================
;               SETTING RTC
;==========================================================================
;
seting:         mov     DPTR,#RegB
                mov     A,#96h
                movx    @DPTR,A
                mov     DPTR,#Hour
                mov     A,#20
                movx    @DPTR,A
                mov     DPTR,#Mint
                mov     A,#11
                movx    @DPTR,A
                mov     DPTR,#Secd
                mov     A,#00
                movx    @DPTR,A
                mov     DPTR,#Date
                mov     A,#02
                movx    @DPTR,A
                mov     DPTR,#Mont
                mov     A,#02
                movx    @DPTR,A
                mov     DPTR,#Year
                mov     A,#07
                movx    @DPTR,A
                mov     DPTR,#RegB
                mov     A,#16h
                movx    @DPTR,A
                ret
;
;==========================================================================
;               INISIALISASI SMS
;==========================================================================
;
sms_in:         mov     DPTR,#smfrm     ; pindahkan data smfrm ke DPTR
                mov     Hurf,#17        ; isi register Hurf dengan 17
                acall   kr_ins          ; panggil kr_ins
                acall   delayk          ; panggil delayk
                ret
;
;==========================================================================
;               BACA INSTRUKSI
;==========================================================================
;
bc_ins:         mov     R1,#0FFh                ; isi R1 dengan 0FFh
bc_in0:         cjne    R1,#0FFh,bc_in1         ; apakah R1=0FFh? Jika tidak lompat ke bc_ins
                sjmp    bc_in0                  ; jika ya, lompat ke bc_ins0
```

```
bc_in1:         ret
;
;=========================================================================
;               APAKAH ADA INSTRUKSI
;=========================================================================
;
ad_ins:         cjne    R1,#0FFh,ad_in0         ; apakah R1=0FFh? Jika tidak lompat ke ad_in0
                sjmp    ad_in1                  ; jika ya, lompat ke ad_in1
ad_in0:         ljmp    bc_sms                  ; lompat ke bc_sms
ad_in1:         ret
;
;=========================================================================
;               KIRIM INSTRUKSI (DIAKHIRI ENTER)
;=========================================================================
;
kr_ins:         clr     A               ; bersihkan Akumulator
                movc    A,@A+DPTR       ;
                mov     R1,A            ; salin data Akumulator ke R1
                inc     DPTR            ; naikkan isi DPTR
                acall   krmsrl          ; panggil krmsrl
                acall   jeda            ; panggil jeda
                djnz    Hurf,kr_ins     ; kurangi isi register Hurf, jika belum nol lompat
                                          ke kr_ins
                mov     R1,#0Dh         ; isi R1 dengan 0Dh
                acall   krmsrl          ; panggil krmsrl
                acall   jeda            ; panggil jeda
                ret
;
;=========================================================================
;               KIRIM DATA (DIAKHIRI CTRL+Z)
;=========================================================================
;
kr_dta:         clr     A               ; bersihkan Akumulator
                movc    A,@A+DPTR
                mov     R1,A            ; salin data Akumulator ke R1
                inc     DPTR            ; naikkan isi DPTR
                acall   krmsrl          ; panggil krmsrl
                acall   jeda            ; panggil jeda
                djnz    Hurf,kr_dta     ; kurangi isi register Hurf, jika belum nol lompat
                                          ke kr_dta
                mov     R1,#26          ; isi R1 dengan 26h
                acall   krmsrl          ; panggil krmsrl
                acall   jeda            ; panggil jeda
                ret
;
;=========================================================================
;               KIRIM SERIAL
;=========================================================================
;
krmsrl:         clr     ES
                mov     SBUF,R1         ; salin isi R1 ke SBUF
                jnb     TI,$            ;
                clr     TI
                setb    ES
                ret
;
;=========================================================================
;               INISIALISASI SERIAL
;=========================================================================
;
srl_in:         setb    EA
                mov     TMOD,#20h
                mov     TH1,#0FDH
                setb    TR1
                mov     SCON,#50h
                mov     A,#80h
                orl     87h,A                   ; double baudrates
                acall   delays
                ret
;
;=========================================================================
;               MENYIAPKAN LINE1 LCD
;=========================================================================
;
```

```
line1:          mov     R0,#80h
                acall   w_ins
                ret
;
;==================================================================
;               MENYIAPKAN LINE2 PADA LCD
;==================================================================
;
line2:          mov     R0,#0C0h
                acall   w_ins
                ret
;
;==================================================================
;               TULIS DATA PADA LCD
;==================================================================
;
tulis:          clr     A
                vc      A,@A+DPTR
                mov     R0,A
                inc     DPTR
                acall   w_chr
                djnz    Hurf,tulis
                ret
;
;==================================================================
;               TULIS CHARACTER PADA LCD
;==================================================================
;
wr_chr:         movc    A,@A+DPTR
                mov     R0,A
                acall   w_chr
                ret
;
w_ins:          clr     Enbl
                clr     Rest
                mov     P0,R0
                setb    Enbl
                clr     Enbl
                acall   jeda
                ret
;
w_chr:          clr     Enbl
                setb    Rest
                mov     P0,R0
                setb    Enbl
                clr     Enbl
                acall   jeda
                ret
;
;==================================================================
;               INISIALISASI LCD
;==================================================================
;
lcd_in:         acall   delays
                mov     R0,#03Fh
                acall   w_ins
                acall   w_ins
                mov     R0,#0Dh
                acall   w_ins
                mov     R0,#06h
                acall   w_ins
                mov     R0,#01h
                acall   w_ins
                mov     R0,#0Ch
                acall   w_ins
                acall   delays
                ret
;
;==================================================================
;               INSTRUKSI PWM
;==================================================================
;
jeda:           mov     A,Dmgg
                cjne    A,#0,jeda0
```

```
                mov     Dly0,#255
                djnz    Dly0,$
jeda0:          cjne    A,#1,jeda1
                clr     Lamp
                mov     Dly0,#100
                djnz    Dly0,$
                setb    Lamp
                mov     Dly0,#155
                djnz    Dly0,$
jeda1:          cjne    A,#2,jeda2
                clr     Lamp
                mov     Dly0,#50
                djnz    Dly0,$
                setb    Lamp
                mov     Dly0,#225
                djnz    Dly0,$
jeda2:          cjne    A,#3,jeda3
                clr     Lamp
                mov     Dly0,#25
                djnz    Dly0,$
                setb    Lamp
                mov     Dly0,#230
                djnz    Dly0,$
jeda3:          cjne    A,#4,jeda4
                mov     Dly0,#255
                djnz    Dly0,$
jeda4:          cjne    A,#15,jeda5
                mov     Dly0,#255
                djnz    Dly0,$
jeda5:          ret
;
delays:         mov     R1,#0FFh
dlys0:          acall   jeda
                acall   ad_ins
                jb      Tbl0,dlys1
tglps3:         acall   jeda
                jnb     Tbl0,tglps3
                mov     R4,#1
                ljmp    menu
dlys1:          djnz    Dly1,dlys0
                ret
;
delayl:         mov     Dly2,#20
dlyl0:          acall   delays
                mov     A,Dmgg
                cjne    A,#1,dlyl1
                mov     Dbts,#35        ; batas = 35.0
                mov     Dbt0,#0
dlyl1:          cjne    A,#2,dlyl2
                mov     Dbts,#32        ; batas = 32.2
                mov     Dbt0,#2
dlyl2:          cjne    A,#3,dlyl3
                mov     Dbts,#29        ; batas = 29.4
                mov     Dbt0,#4
dlyl3:          cjne    A,#15,dlyl4
                setb    Fans
                sjmp    dlyl5
dlyl4:          acall   cekbts
dlyl5:          jb      Ldrs,dlyl7      ; jika LDR deteksi lampu mati
                mov     A,Dmgg
                cjne    A,#15,dlyl6
                sjmp    dlyl7
dlyl6:          acall   kr_err          ; kirim sms error
dlyl7:          jnb     Ldrs,dlyl8      ; jika LDR deteksi lampu nyala
                mov     Ster,#0         ; reset status error
dlyl8:          djnz    Dly2,dlyl0
                ret
;
delayk:         mov     Dly3,#25        ; isi Dly3 dengan 25
dlyk0:          mov     Dly4,#255       ; isi Dly4 dengan 255
dlyk1:          acall   jeda            ; panggil jeda
                djnz    Dly4,dlyk1      ; kurangi Dly4, jika belum nol lompat ke dlyk1
                djnz    Dly3,dlyk0      ; kurangi Dly3, jika belum nol lompat ke dlyk0
                ret
```

```
;
;=========================================================================
;               KALIBRASI SUHU 28.0-28.9
;=========================================================================
;
klbrs:          cjne    R2,#28,klbr1    ; apakah isi R2=28? Jika tidak lompat ke klbr1
klb00:          cjne    R3,#0,klb01     ; jika ya, apakah isi R3=0? Jika tidak lompat ke klb01
                mov     DPTR,#dt280     ; jika ya, pindahkan data dt280 ke DPTR
klb01:          cjne    R3,#1,klb02     ;
                mov     DPTR,#dt281     ;
klb02:          cjne    R3,#2,klb03
                mov     DPTR,#dt282
klb03:          cjne    R3,#3,klb04
                mov     DPTR,#dt283
klb04:          cjne    R3,#4,klb05
                mov     DPTR,#dt284
klb05:          cjne    R3,#5,klb06
                mov     DPTR,#dt285
klb06:          cjne    R3,#6,klb07
                mov     DPTR,#dt286
klb07:          cjne    R3,#7,klb08
                mov     DPTR,#dt287
klb08:          cjne    R3,#8,klb09
                mov     DPTR,#dt288
klb09:          cjne    R3,#9,klbr1
                mov     DPTR,#dt289
;
;=========================================================================
;               SUHU 29.0-29.9
;=========================================================================
;
klbr1:          cjne    R2,#29,klbr2
klb10:          cjne    R3,#0,klb11
                mov     DPTR,#dt290
klb11:          cjne    R3,#1,klb12
                mov     DPTR,#dt291
klb12:          cjne    R3,#2,klb13
                mov     DPTR,#dt292
klb13:          cjne    R3,#3,klb14
                mov     DPTR,#dt293
klb14:          cjne    R3,#4,klb15
                mov     DPTR,#dt294
klb15:          cjne    R3,#5,klb16
                mov     DPTR,#dt295
klb16:          cjne    R3,#6,klb17
                mov     DPTR,#dt296
klb17:          cjne    R3,#7,klb18
                mov     DPTR,#dt297
klb18:          cjne    R3,#8,klb19
                mov     DPTR,#dt298
klb19:          cjne    R3,#9,klbr2
                mov     DPTR,#dt299
;
;=========================================================================
;               SUHU 30.0-30.9
;=========================================================================
;
klbr2:          cjne    R2,#30,klbr3
klb20:          cjne    R3,#0,klb21
                mov     DPTR,#dt300
klb21:          cjne    R3,#1,klb22
                mov     DPTR,#dt301
klb22:          cjne    R3,#2,klb23
                mov     DPTR,#dt302
klb23:          cjne    R3,#3,klb24
                mov     DPTR,#dt303
klb24:          cjne    R3,#4,klb25
                mov     DPTR,#dt304
klb25:          cjne    R3,#5,klb26
                mov     DPTR,#dt305
klb26:          cjne    R3,#6,klb27
                mov     DPTR,#dt306
klb27:          cjne    R3,#7,klb28
                mov     DPTR,#dt307
```

```
klb28:          cjne    R3,#8,klb29
                mov     DPTR,#dt308
klb29:          cjne    R3,#9,klbr3
                mov     DPTR,#dt309
;
;=========================================================================
;               SUHU 31.0-31.9
;=========================================================================
;
klbr3:          cjne    R2,#31,klbr4
klb30:          cjne    R3,#0,klb31
                mov     DPTR,#dt310
klb31:          cjne    R3,#1,klb32
                mov     DPTR,#dt311
klb32:          cjne    R3,#2,klb33
                mov     DPTR,#dt312
klb33:          cjne    R3,#3,klb34
                mov     DPTR,#dt313
klb34:          cjne    R3,#4,klb35
                mov     DPTR,#dt314
klb35:          cjne    R3,#5,klb36
                mov     DPTR,#dt315
klb36:          cjne    R3,#6,klb37
                mov     DPTR,#dt316
klb37:          cjne    R3,#7,klb38
                mov     DPTR,#dt317
klb38:          cjne    R3,#8,klb39
                mov     DPTR,#dt318
klb39:          cjne    R3,#9,klbr4
                mov     DPTR,#dt319
;
;=========================================================================
;               SUHU 32.0-32.9
;=========================================================================
;
klbr4:          cjne    R2,#32,klbr5
klb40:          cjne    R3,#0,klb41
                mov     DPTR,#dt320
klb41:          cjne    R3,#1,klb42
                mov     DPTR,#dt321
klb42:          cjne    R3,#2,klb43
                mov     DPTR,#dt322
klb43:          cjne    R3,#3,klb44
                mov     DPTR,#dt323
klb44:          cjne    R3,#4,klb45
                mov     DPTR,#dt324
klb45:          cjne    R3,#5,klb46
                mov     DPTR,#dt325
klb46:          cjne    R3,#6,klb47
                mov     DPTR,#dt326
klb47:          cjne    R3,#7,klb48
                mov     DPTR,#dt327
klb48:          cjne    R3,#8,klb49
                mov     DPTR,#dt328
klb49:          cjne    R3,#9,klbr5
                mov     DPTR,#dt329
;
;=========================================================================
;               SUHU 33.0-33.9
;=========================================================================
;
klbr5:          cjne    R2,#33,klbr6
klb50:          cjne    R3,#0,klb51
                mov     DPTR,#dt330
klb51:          cjne    R3,#1,klb52
                mov     DPTR,#dt331
klb52:          cjne    R3,#2,klb53
                mov     DPTR,#dt332
klb53:          cjne    R3,#3,klb54
                mov     DPTR,#dt333
klb54:          cjne    R3,#4,klb55
                mov     DPTR,#dt334
klb55:          cjne    R3,#5,klb56
                mov     DPTR,#dt335
```

```
klb56:          cjne    R3,#6,klb57
                mov     DPTR,#dt336
klb57:          cjne    R3,#7,klb58
                mov     DPTR,#dt337
klb58:          cjne    R3,#8,klb59
                mov     DPTR,#dt338
klb59:          cjne    R3,#9,klbr6
                mov     DPTR,#dt339
;
;==========================================================================
;               SUHU 34.0-34.9
;==========================================================================
;
klbr6:          cjne    R2,#34,klbr7
klb60:          cjne    R3,#0,klb61
                mov     DPTR,#dt340
klb61:          cjne    R3,#1,klb62
                mov     DPTR,#dt341
klb62:          cjne    R3,#2,klb63
                mov     DPTR,#dt342
klb63:          cjne    R3,#3,klb64
                mov     DPTR,#dt343
klb64:          cjne    R3,#4,klb65
                mov     DPTR,#dt344
klb65:          cjne    R3,#5,klb66
                mov     DPTR,#dt345
klb66:          cjne    R3,#6,klb67
                mov     DPTR,#dt346
klb67:          cjne    R3,#7,klb68
                mov     DPTR,#dt347
klb68:          cjne    R3,#8,klb69
                mov     DPTR,#dt348
klb69:          cjne    R3,#9,klbr7
                mov     DPTR,#dt349
;
;==========================================================================
;               SUHU 35.0-35.9
;==========================================================================
;
klbr7:          cjne    R2,#35,klbr8
klb70:          cjne    R3,#0,klb71
                mov     DPTR,#dt350
klb71:          cjne    R3,#1,klb72
                mov     DPTR,#dt351
klb72:          cjne    R3,#2,klb73
                mov     DPTR,#dt352
klb73:          cjne    R3,#3,klb74
                mov     DPTR,#dt353
klb74:          cjne    R3,#4,klb75
                mov     DPTR,#dt354
klb75:          cjne    R3,#5,klb76
                mov     DPTR,#dt355
klb76:          cjne    R3,#6,klb77
                mov     DPTR,#dt356
klb77:          cjne    R3,#7,klb78
                mov     DPTR,#dt357
klb78:          cjne    R3,#8,klb79
                mov     DPTR,#dt358
klb79:          cjne    R3,#9,klbr8
                mov     DPTR,#dt359
;
klbr8:          ret
;
;==========================================================================
;               DATA
;==========================================================================
;
logo:           DB      ' Mufid Murtadlo '
                DB      '   002.17.144   '
                DB      ' Elektronika S1 '
                DB      '  ITN Malang    '
tpshu:          DB      '   - TEMP -     '
tptim:          DB      '  - TIME -      '
angka:          DB      '0123456789 .C:/-'
```

```
tmenu:      DB    ' >    MENU    < '
plmg1:      DB    ' Pilih Minggu 1 '
plmg2:      DB    ' Pilih Minggu 2 '
plmg3:      DB    ' Pilih Minggu 3 '
mtsys:      DB    ' Matikan System '
mggke:      DB    '  Minggu Ke     '
harke:      DB    '   Hari Ke      '
smfrm:      DB    'AT+CNMI=1,1,0,0,1'
smbca:      DB    'AT+CMGR=1'
smhps:      DB    'AT+CMGD=1'
smk21:      DB    'AT+CMGS=21'
smk24:      DB    'AT+CMGS=24'
smk29:      DB    'AT+CMGS=29'
dtcek:      DB    'E3F21A'
dt280:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB0D310'        ; 50-24
dt281:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB1D310'
dt282:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB2D310'
dt283:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB3D310'
dt284:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB4D310'
dt285:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB5D310'
dt286:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB6D310'
dt287:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB7D310'
dt288:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB8D310'
dt289:      DB    '0001000D91261839812990F400000BF33ABA0E92E15CB9D310'
dt290:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB0D310'
dt291:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB1D310'
dt292:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB2D310'
dt293:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB3D310'
dt294:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB4D310'
dt295:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB5D310'
dt296:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB6D310'
dt297:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB7D310'
dt298:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB8D310'
dt299:      DB    '0001000D91261839812990F400000BF33ABA0E92E55CB9D310'
dt300:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB0D310'
dt301:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB1D310'
dt302:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB2D310'
dt303:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB3D310'
dt304:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB4D310'
dt305:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB5D310'
dt306:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB6D310'
dt307:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB7D310'
dt308:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB8D310'
dt309:      DB    '0001000D91261839812990F400000BF33ABA0E9AC15CB9D310'
dt310:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB0D310'
dt311:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB1D310'
dt312:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB2D310'
dt313:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB3D310'
dt314:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB4D310'
dt315:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB5D310'
dt316:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB6D310'
dt317:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB7D310'
dt318:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB8D310'
dt319:      DB    '0001000D91261839812990F400000BF33ABA0E9AC55CB9D310'
dt320:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB0D310'
dt321:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB1D310'
dt322:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB2D310'
dt323:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB3D310'
dt324:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB4D310'
dt325:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB5D310'
dt326:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB6D310'
dt327:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB7D310'
dt328:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB8D310'
dt329:      DB    '0001000D91261839812990F400000BF33ABA0E9AC95CB9D310'
dt330:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB0D310'
dt331:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB1D310'
dt332:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB2D310'
dt333:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB3D310'
dt334:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB4D310'
dt335:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB5D310'
dt336:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB6D310'
dt337:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB7D310'
dt338:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB8D310'
dt339:      DB    '0001000D91261839812990F400000BF33ABA0E9ACD5CB9D310'
```

```
dt340:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB0D310'
dt341:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB1D310'
dt342:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB2D310'
dt343:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB3D310'
dt344:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB4D310'
dt345:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB5D310'
dt346:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB6D310'
dt347:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB7D310'
dt348:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB8D310'
dt349:          DB      '0001000D91261839812990F400000BF33ABA0E9AD15CB9D310'
dt350:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB0D310'
dt351:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB1D310'
dt352:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB2D310'
dt353:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB3D310'
dt354:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB4D310'
dt355:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB5D310'
dt356:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB6D310'
dt357:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB7D310'
dt358:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB8D310'
dt359:          DB      '0001000D91261839812990F400000BF33ABA0E9AD55CB9D310'
dterr:          DB      '0001000D91261839812990F400001165B9FC2D6781D8E136BC0E6A87E969'
                ; 60-29
dtmsk:          DB      '07912618485400F9240D91261839812990F4000070103111517182 03E3F21A'
                ; end
```

ires

atible with MCS-51™ Products
tes of In-System Reprogrammable Downloadable Flash Memory
PI Serial Interface for Program Downloading
ndurance: 1,000 Write/Erase Cycles
tes EEPROM
ndurance: 100,000 Write/Erase Cycles
6V Operating Range
Static Operation: 0 Hz to 24 MHz
-level Program Memory Lock
8-bit Internal RAM
grammable I/O Lines
16-bit Timer/Counters
nterrupt Sources
ammable UART Serial Channel
rial Interface
ower Idle and Power-down Modes
pt Recovery From Power-down
ammable Watchdog Timer
ata Pointer
-off Flag

ription

89S8252 is a low-power, high-performance CMOS 8-bit microcomputer with s of downloadable Flash programmable and erasable read only memory and s of EEPROM. The device is manufactured using Atmel's high-density nonvol- emory technology and is compatible with the industry-standard 80C51 on set and pinout. The on-chip downloadable Flash allows the program mem- be reprogrammed in-system through an SPI serial interface or by a ional nonvolatile memory programmer. By combining a versatile 8-bit CPU vnloadable Flash on a monolithic chip, the Atmel AT89S8252 is a powerful mputer which provides a highly-flexible and cost-effective solution to many led control applications.

39S8252 provides the following standard features: 8K bytes of downloadable K bytes of EEPROM, 256 bytes of RAM, 32 I/O lines, programmable watch- er, two data pointers, three 16-bit timer/counters, a six-vector two-level architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In , the AT89S8252 is designed with static logic for operation down to zero fre- and supports two software selectable power saving modes. The Idle Mode e CPU while allowing the RAM, timer/counters, serial port, and interrupt sys- continue functioning. The Power-down mode saves the RAM contents but the oscillator, disabling all other chip functions until the next interrupt or hard- set.

vnloadable Flash can be changed a single byte at a time and is accessible the SPI serial interface. Holding RESET active forces the SPI bus into a serial ming interface and allows the program memory to be written to or read from ock Bit 2 has been activated.

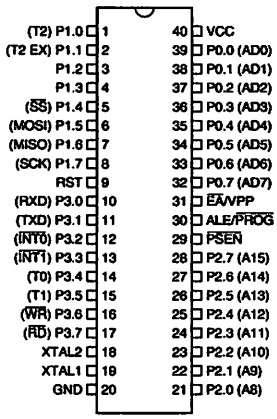
8-bit Microcontroller with 8K Bytes Flash

AT89S8252

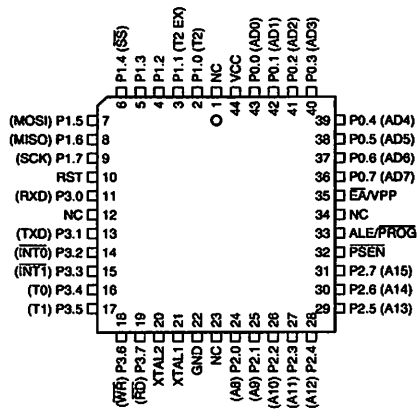
Rev. 0401E–02/00

Configurations

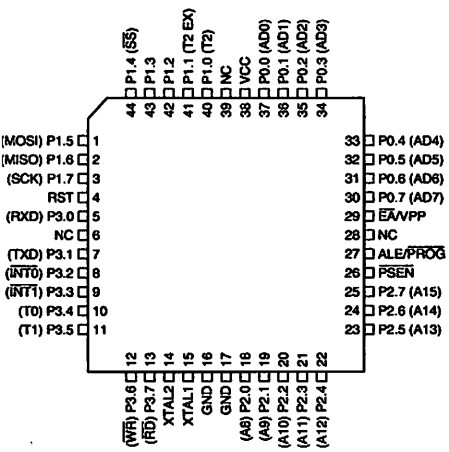
PDIP



PLCC



PQFP/TQFP



Description

voltage.

.

s an 8-bit open drain bbi-didirectional I/O port. As an port, each pin can sink eight TTL inputs. When 1s tten to port 0 pins, the pins can be used as high-nce inputs.

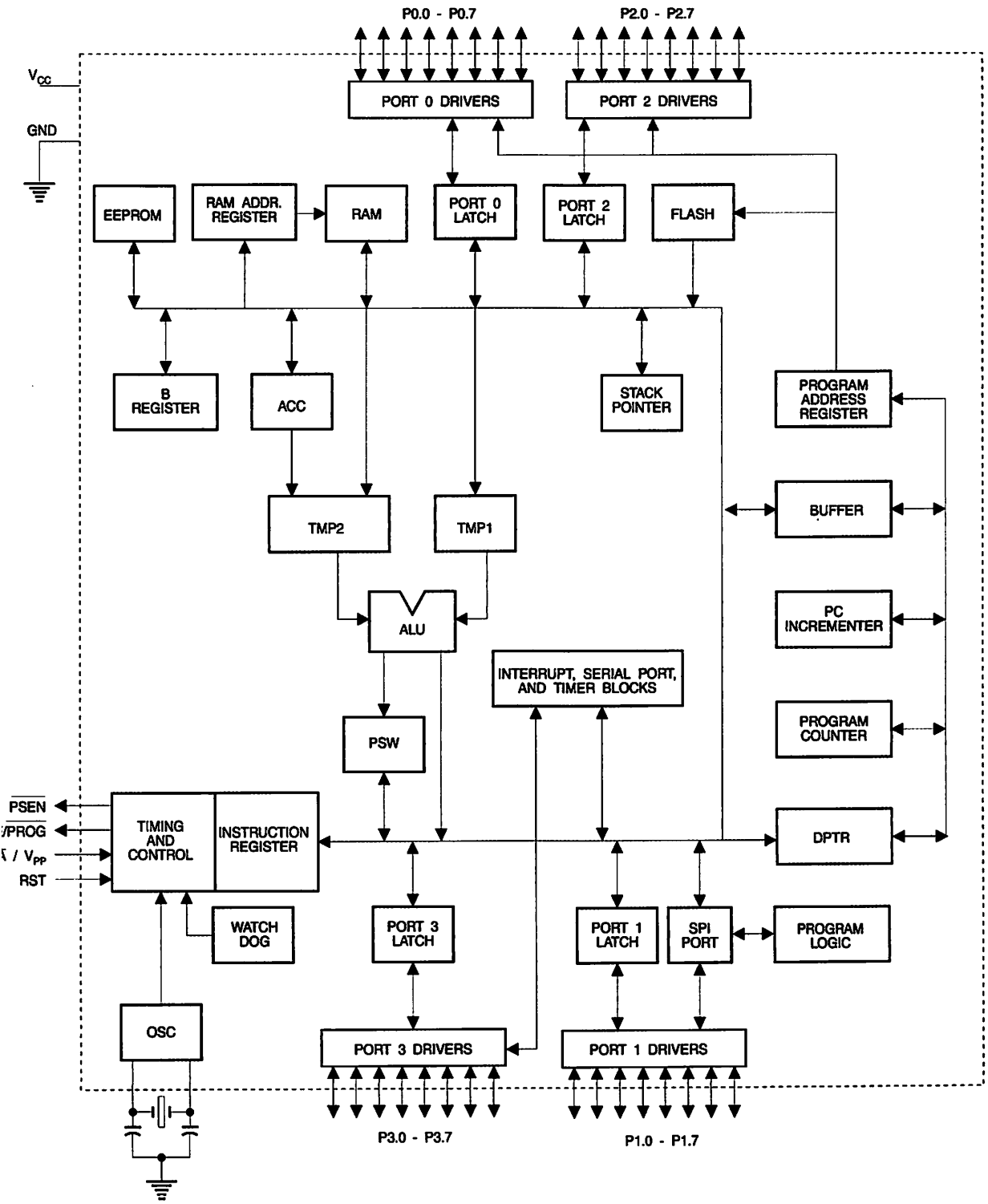
can also be configured to be the multiplexed low-address/data bus during accesses to external program and data memory. In this mode, P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

Port 1 is an 8-bit bi-directional I/O port with internal pullups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

k Diagram


P0.0 - P0.7
P2.0 - P2.7
VCC
GND
PORT 0 DRIVERS
PORT 2 DRIVERS
EEPROM
RAM ADDR. REGISTER
RAM
PORT 0 LATCH
PORT 2 LATCH
FLASH
B REGISTER
ACC
STACK POINTER
PROGRAM ADDRESS REGISTER
BUFFER
TMP2
TMP1
ALU
PC INCREMENTER
INTERRUPT, SERIAL PORT, AND TIMER BLOCKS
PROGRAM COUNTER
PSW
PSEN
/PROG
/ VPP
RST
TIMING AND CONTROL
INSTRUCTION REGISTER
DPTR
WATCH DOG
PORT 3 LATCH
PORT 1 LATCH
SPI PORT
PROGRAM LOGIC
OSC
PORT 3 DRIVERS
PORT 1 DRIVERS
P3.0 - P3.7
P1.0 - P1.7

ort 1 pins provide additional functions. P1.0 and n be configured to be the timer/counter 2 external put (P1.0/T2) and the timer/counter 2 trigger input 2EX), respectively.

escription

nore, P1.4, P1.5, P1.6, and P1.7 can be configured SPI slave port select, data input/output and shift put/output pins as shown in the following table.

n	Alternate Functions
	T2 (external count input to Timer/Counter 2), clock-out
	T2EX (Timer/Counter 2 capture/reload trigger and direction control)
	$\overline{SS}$ (Slave port select input)
	MOSI (Master data output, slave data input pin for SPI channel)
	MISO (Master data input, slave data output pin for SPI channel)
	SCK (Master clock output, slave clock input pin for SPI channel)

lso receives the low-order address bytes during ogramming and verification.

an 8-bit bi-directional I/O port with internal pullups. t 2 output buffers can sink/source four TTL inputs. s are written to Port 2 pins, they are pulled high by rnal pullups and can be used as inputs. As inputs, ins that are externally being pulled low will source (I_{IL}) because of the internal pullups.

mits the high-order address byte during fetches ternal program memory and during accesses to l data memory that use 16-bit addresses (MOVX @ In this application, Port 2 uses strong internal pul- en emitting 1s. During accesses to external data that use 8-bit addresses (MOVX @ RI), Port 2 e contents of the P2 Special Function Register.

lso receives the high-order address bits and some signals during Flash programming and verification.

an 8 bit bi-directional I/O port with internal pullups. rt 3 output buffers can sink/source four TTL inputs. s are written to Port 3 pins, they are pulled high by rnal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89S8252, as shown in the following table.

Port 3 also receives some control signals for Flash programming and verification.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$

Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

$\overline{PSEN}$

Program Store Enable is the read strobe to external program memory.

When the AT89S8252 is executing code from external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory.

$\overline{EA}$/VPP

External Access Enable. $\overline{EA}$ must be strapped to GND in order to enable the device to fetch code from external pro-

emory locations starting at 0000H up to FFFFH. ›wever, that if lock bit 1 is programmed, $\overline{EA}$ will be y latched on reset.

ıld be strapped to V_{CC} for internal program execu- 'his pin also receives the 12-volt programming voltage (V_{PP}) during Flash programming when 12- ;ramming is selected.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier.

AT89S8252 SFR Map and Reset Values

								0FFH
B 00000000								0F7H
								0EFH
ACC 00000000								0E7H
								0DFH
PSW 00000000					SPCR 000001XX			0D7H
T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
								0C7H
IP XX000000								0BFH
P3 11111111								0B7H
IE 0X000000		SPSR 00XXXXXX						0AFH
P2 11111111								0A7H
SCON 00000000	SBUF XXXXXXXX							9FH
P1 11111111						WMCON 00000010		97H
TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

ial Function Registers

of the on-chip memory area called the Special Func-
gister (SFR) space is shown in Table 1.

at not all of the addresses are occupied, and unoc-
addresses may not be implemented on the chip.
ccesses to these addresses will in general return
data, and write accesses will have an indeterminate

oftware should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Timer 2 Registers Control and status bits are contained in registers T2CON (shown in Table 2) and T2MOD (shown in Table 9) for Timer 2. The register pair (RCAP2H, RCAP2L) are the Capture/Reload registers for Timer 2 in 16 bit capture mode or 16-bit auto-reload mode.

. T2CON—Timer/Counter 2 Control Register

l Address = 0C8H Reset Value = 0000 0000B

lressable

TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
7	6	5	4	3	2	1	0

ol	Function
	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).
	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflows to be used for the receive clock.
	Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
$\overline{2}$	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

log and Memory Control Register The WMCON ' contains control bits for the Watchdog Timer in Table 3). The EEMEN and EEMWE bits are used to select the 2K bytes on-chip EEPROM, and to enable byte-write. The DPS bit selects one of two DPTR registers available.

WMCON—Watchdog and Memory Control Register

N Address = 96H

Reset Value = 0000 0010B

PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN
7	6	5	4	3	2	1	0

I	Function
	Prescaler Bits for the Watchdog Timer. When all three bits are set to "0", the watchdog timer has a nominal period of 16 ms. When all three bits are set to "1", the nominal period is 2048 ms.
E	EEPROM Data Memory Write Enable Bit. Set this bit to "1" before initiating byte write to on-chip EEPROM with the MOVX instruction. User software should set this bit to "0" after EEPROM write is completed.
I	Internal EEPROM Access Enable. When EEMEN = 1, the MOVX instruction with DPTR will access on-chip EEPROM instead of external data memory. When EEMEN = 0, MOVX with DPTR accesses external data memory.
	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1
ST SY	Watchdog Timer Reset and EEPROM Ready/$\overline{\text{Busy}}$ Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The WDTRST bit is then automatically reset to "0" in the next instruction cycle. The WDTRST bit is Write-Only. This bit also serves as the RDY/$\overline{\text{BSY}}$ flag in a Read-Only mode during EEPROM write. RDY/$\overline{\text{BSY}}$ = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/$\overline{\text{BSY}}$ bit equals "0" and is automatically reset to "1" when programming is completed.
I	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

isters Control and status bits for the Serial Periph- rface are contained in registers SPCR (shown in and SPSR (shown in Table 5). The SPI data bits ained in the SPDR register. Writing the SPI data during serial data transfer sets the Write Collision DL, in the SPSR register. The SPDR is double buff- writing and the values in SPDR are not changed by

t Registers The global interrupt enable bit and the al interrupt enable bits are in the IE register. In , the individual interrupt enable bit for the SPI is in R register. Two priorities can be set for each of the upt sources in the IP register.

Dual Data Pointer Registers To facilitate accessing both internal EEPROM and external data memory, two banks of 16 bit Data Pointer Registers are provided: DP0 at SFR address locations 82H-83H and DP1 at 84H-85H. Bit DPS = 0 in SFR WMCON selects DP0 and DPS = 1 selects DP1. The user should always initialize the DPS bit to the appropriate value before accessing the respective Data Pointer Register.

Power Off Flag The Power Off Flag (POF) is located at bit_4 (PCON.4) in the PCON SFR. POF is set to "1" during power up. It can be set and reset under software control and is not affected by RESET.

SPCR—SPI Control Register

Address = D5H

Reset Value = 0000 01XXB

SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
7	6	5	4	3	2	1	0

	Function
	SPI Interrupt Enable. This bit, in conjunction with the ES bit in the IE register, enables SPI interrupts: SPIE = 1 and ES = 1 enable SPI interrupts. SPIE = 0 disables SPI interrupts.
	SPI Enable. SPI = 1 enables the SPI channel and connects $\overline{SS}$, MOSI, MISO and SCK to pins P1.4, P1.5, P1.6, and P1.7. SPI = 0 disables the SPI channel.
	Data Order. DORD = 1 selects LSB first data transmission. DORD = 0 selects MSB first data transmission.
	Master/Slave Select. MSTR = 1 selects Master SPI mode. MSTR = 0 selects Slave SPI mode.
	Clock Polarity. When CPOL = 1, SCK is high when idle. When CPOL = 0, SCK of the master device is low when not transmitting. Please refer to figure on SPI Clock Phase and Polarity Control.
	Clock Phase. The CPHA bit together with the CPOL bit controls the clock and data relationship between master and slave. Please refer to figure on SPI Clock Phase and Polarity Control.
	SPI Clock Rate Select. These two bits control the SCK rate of the device configured as master. SPR1 and SPR0 have no effect on the slave. The relationship between SCK and the oscillator frequency, $F_{osc.}$, is as follows: SPR1 SPR0 SCK = $F_{osc.}$ divided by 0 0 4 0 1 16 1 0 64 1 1 128

SPSR – SPI Status Register

Address = AAH

Reset Value = 00XX XXXXB

SPIF	WCOL	–	–	–	–	–	–
7	6	5	4	3	2	1	0

	Function
	SPI Interrupt Flag. When a serial transfer is complete, the SPIF bit is set and an interrupt is generated if SPIE = 1 and ES = 1. The SPIF bit is cleared by reading the SPI status register with SPIF and WCOL bits set, and then accessing the SPI data register.
	Write Collision Flag. The WCOL bit is set if the SPI data register is written during a data transfer. During data transfer, the result of reading the SPDR register may be incorrect, and writing to it has no effect. The WCOL bit (and the SPIF bit) are cleared by reading the SPI status register with SPIF and WCOL set, and then accessing the SPI data register.

SPDR – SPI Data Register

Address = 86H

Reset Value = unchanged

SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
7	6	5	4	3	2	1	0

Memory – EEPROM and RAM

I9S8252 implements 2K bytes of on-chip EEPROM storage and 256 bytes of RAM. The upper 128 RAM occupy a parallel space to the Special Func- jisters. That means the upper 128 bytes have the ldresses as the SFR space but are physically sepa- ı SFR space.

n instruction accesses an internal location above 7FH, the address mode used in the instruction s whether the CPU accesses the upper 128 bytes or the SFR space. Instructions that use direct ng access SFR space.

mple, the following direct addressing instruction s the SFR at location 0A0H (which is P2).

```
0A0H, #data
```

ɔns that use indirect addressing access the upper es of RAM. For example, the following indirect ng instruction, where R0 contains 0A0H, accesses ı byte at address 0A0H, rather than P2 (whose is 0A0H).

```
@R0, #data
```

at stack operations are examples of indirect ing, so the upper 128 bytes of data RAM are avail- stack space.

chip EEPROM data memory is selected by setting MEN bit in the WMCON register at SFR address 96H. The EEPROM address range is from 000H to The MOVX instructions are used to access the M. To access off-chip data memory with the MOVX ɔns, the EEMEN bit needs to be set to "0".

MWE bit in the WMCON register needs to be set to e any byte location in the EEPROM can be written. ftware should reset EEMWE bit to "0" if no further M write is required. EEPROM write cycles in the ogramming mode are self-timed and typically take The progress of EEPROM write can be monitored ng the RDY/$\overline{BSY}$ bit (read-only) in SFR WMCON. $\overline{SY}$ = 0 means programming is still in progress and $\overline{SY}$ = 1 means EEPROM write cycle is completed ther write cycle can be initiated.

ion, during EEPROM programming, an attempted m the EEPROM will fetch the byte being written MSB complemented. Once the write cycle is com- rue data are valid at all bit locations.

ammable Watchdog Timer

grammable Watchdog Timer (WDT) operates from pendent oscillator. The prescaler bits, PS0, PS1 2 in SFR WMCON are used to set the period of the og Timer from 16 ms to 2048 ms. The available eriods are shown in the following table and the actual timer periods (at V_{CC} = 5V) are within ±30% of the nominal.

The WDT is disabled by Power-on Reset and during Power-down. It is enabled by setting the WDTEN bit in SFR WMCON (address = 96H). The WDT is reset by setting the WDTRST bit in WMCON. When the WDT times out without being reset or disabled, an internal RST pulse is generated to reset the CPU.

Table 7. Watchdog Timer Period Selection

WDT Prescaler Bits			
PS2	**PS1**	**PS0**	**Period (nominal)**
0	0	0	16 ms
0	0	1	32 ms
0	1	0	64 ms
0	1	1	128 ms
1	0	0	256 ms
1	0	1	512 ms
1	1	0	1024 ms
1	1	1	2048 ms

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S8252 operate the same way as Timer 0 and Timer 1 in the AT89C51, AT89C52 and AT89C55. For further information, see the October 1995 Microcontroller Data Book, page 2-45, section titled, "Timer/Counters."

Timer 2

Timer 2 is a 16 bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 8.

Timer 2 consists of two 8-bit registers, TH2 and TL2. In the Timer function, the TL2 register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which

sition was detected. Since two machine cycles (24 r periods) are required to recognize a 1-to-0 transi- e maximum count rate is 1/24 of the oscillator cy. To ensure that a given level is sampled at least fore it changes, the level should be held for at least machine cycle.

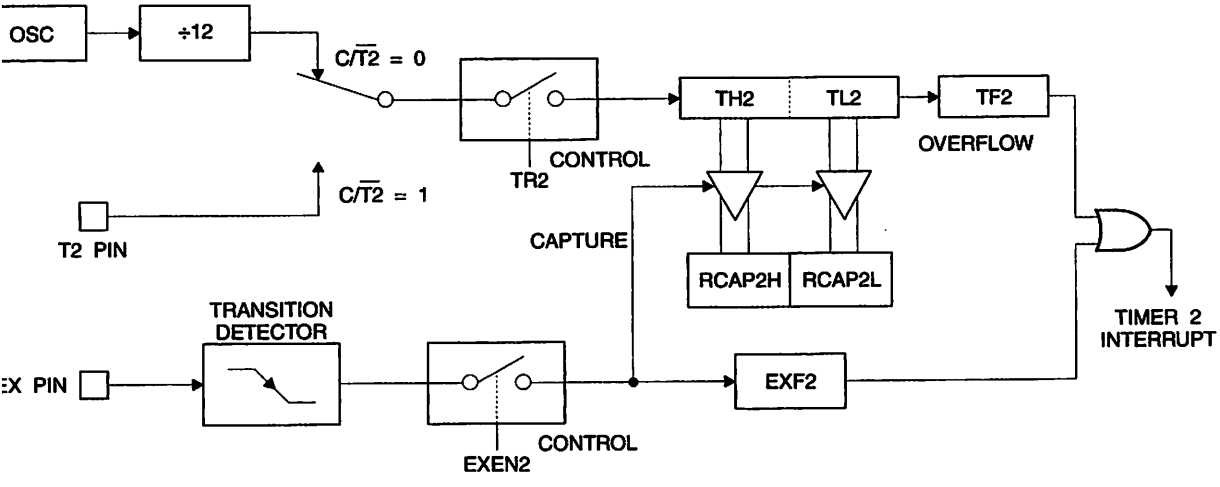
Timer 2 Operating Modes

+ TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

Capture Mode

In the capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16 bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 performs the same operation, but a l-to-0 transition at external input T2EX also causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt. The capture mode is illustrated in Figure 1.

l. Timer 2 in Capture Mode



eload (Up or Down Counter)

can be programmed to count up or down when ed in its 16 bit auto-reload mode. This feature is by the DCEN (Down Counter Enable) bit located in T2MOD (see Table 9). Upon reset, the DCEN bit 0 so that timer 2 will default to count up. When set, Timer 2 can count up or down, depending on e of the T2EX pin.

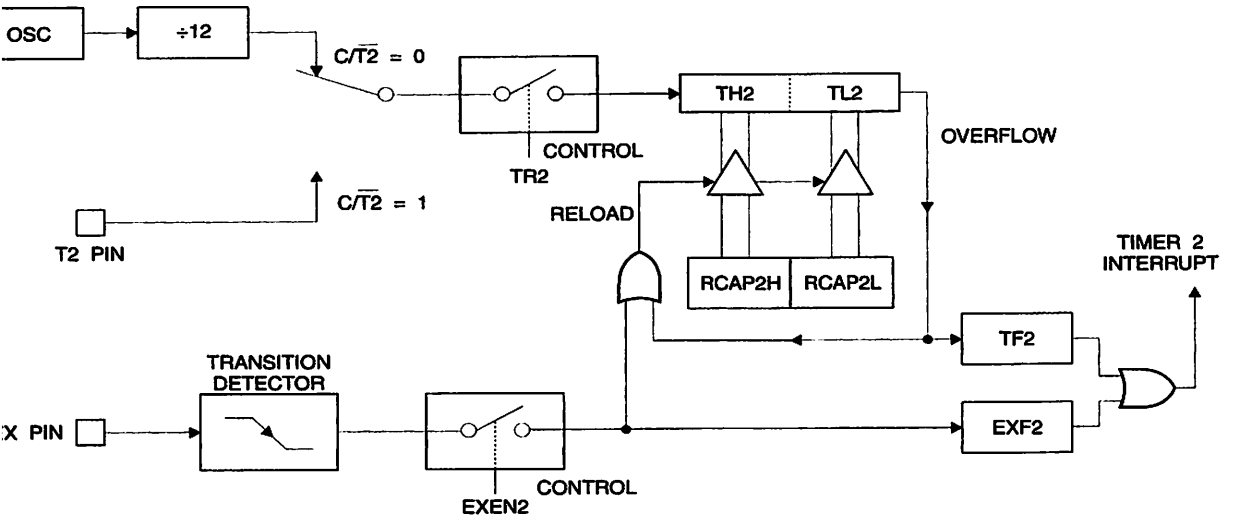
shows Timer 2 automatically counting up when 0. In this mode, two options are selected by bit in T2CON. If EXEN2 = 0, Timer 2 counts up to and then sets the TF2 bit upon overflow. The also causes the timer registers to be reloaded with it value in RCAP2H and RCAP2L. The values in and RCAP2L are preset by software. If EXEN2 = it reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 3. In this mode, the T2EX pin controls the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16 bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

A logic 0 at T2EX makes Timer 2 count down. The timer underflows when TH2 and TL2 equal the values stored in RCAP2H and RCAP2L. The underflow sets the TF2 bit and causes 0FFFFH to be reloaded into the timer registers.

The EXF2 bit toggles whenever Timer 2 overflows or underflows and can be used as a 17th bit of resolution. In this operating mode, EXF2 does not flag an interrupt.

. Timer 2 in Auto Reload Mode (DCEN = 0)



. T2MOD – Timer 2 Mode Control Register

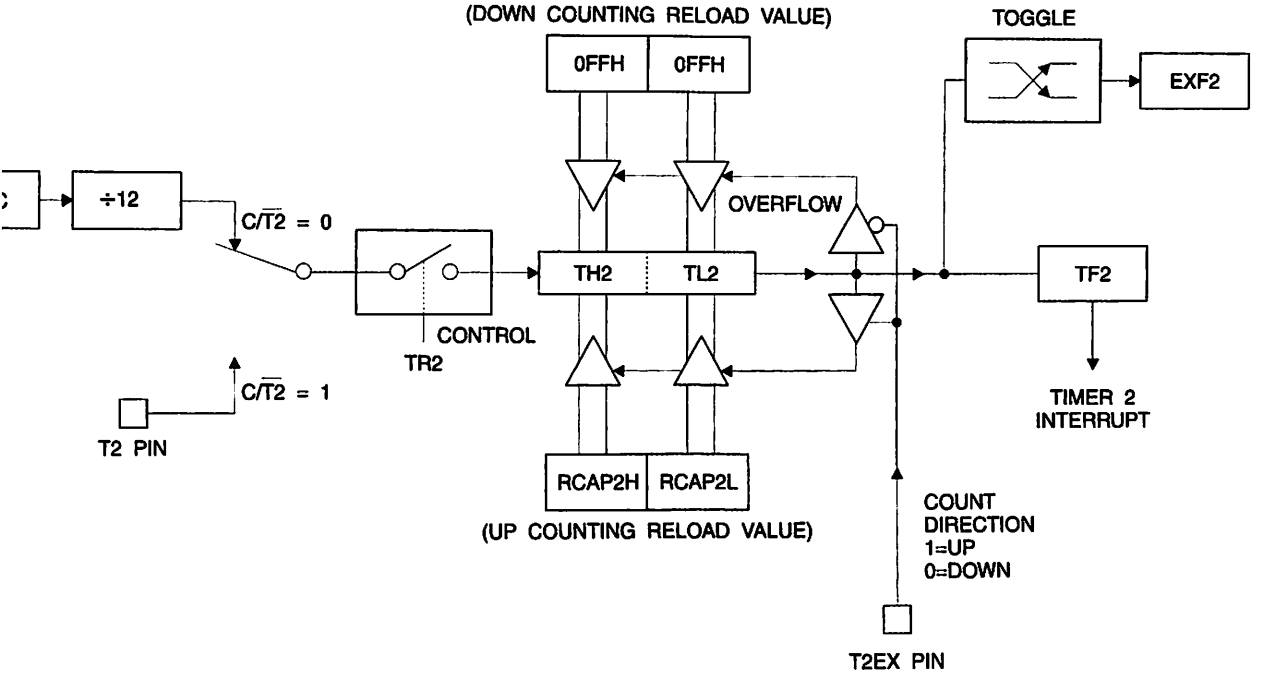
Address = 0C9H | Reset Value = XXXX XX00B

Addressable

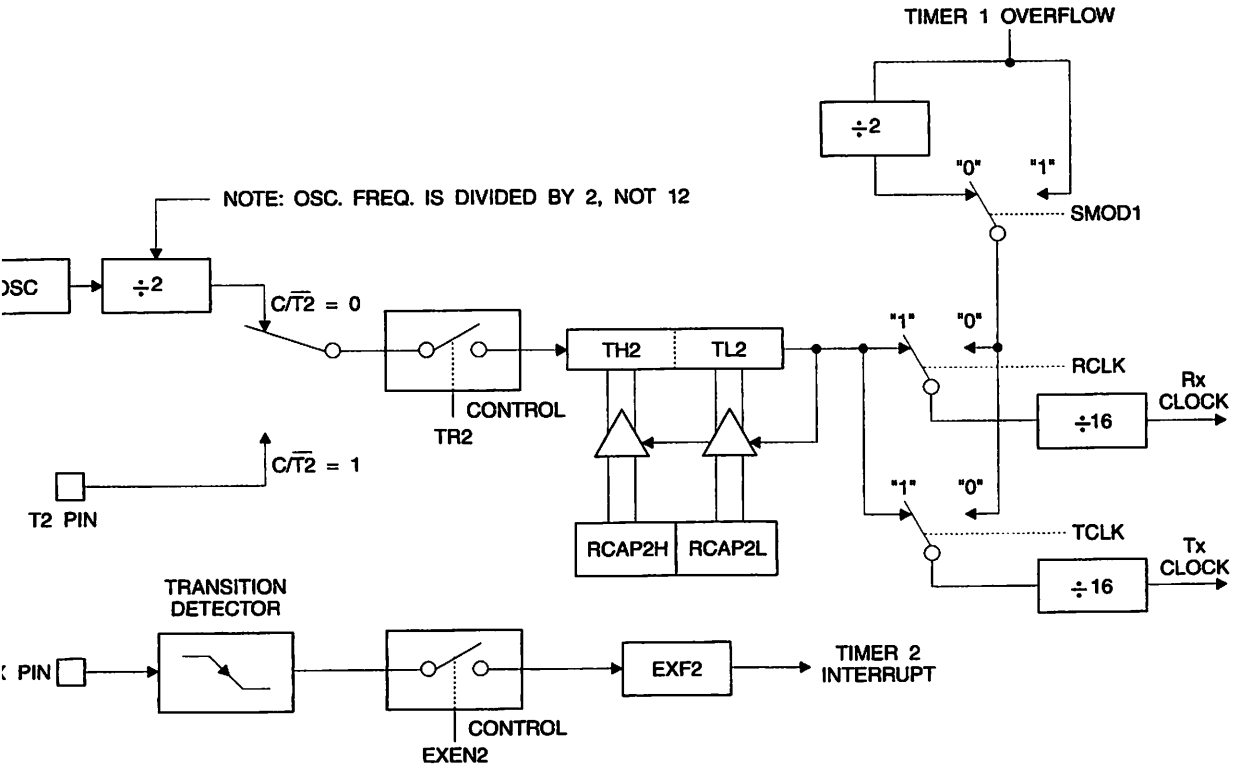
–	–	–	–	–	–	T2OE	DCEN
7	6	5	4	3	2	1	0

	Function
	Not implemented, reserved for future use.
	Timer 2 Output Enable bit.
	When set, this bit allows Timer 2 to be configured as an up/down counter.

3. Timer 2 Auto Reload Mode (DCEN = 1)



4. Timer 2 in Baud Rate Generator Mode



Rate Generator

is selected as the baud rate generator by setting nd/or RCLK in T2CON (Table 2). Note that the es for transmit and receive can be different if Timer 1 for the receiver or transmitter and Timer 1 is used other function. Setting RCLK and/or TCLK puts into its baud rate generator mode, as shown in Fig-

d rate generator mode is similar to the auto-reload that a rollover in TH2 causes the Timer 2 registers oaded with the 16 bit value in registers RCAP2H AP2L, which are preset by software.

d rates in Modes 1 and 3 are determined by Timer low rate according to the following equation.

$$\text{s 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

er can be configured for either timer or counter n. In most applications, it is configured for timer n ($CP/\overline{T2} = 0$). The timer operation is different for when it is used as a baud rate generator. Normally, er, it increments every machine cycle (at 1/12 the r frequency). As a baud rate generator, however, it nts every state time (at 1/2 the oscillator fre- . The baud rate formula is given below.

$$\frac{\text{s 1 and 3}}{\text{d Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

RCAP2H, RCAP2L) is the content of RCAP2H and . taken as a 16 bit unsigned integer.

as a baud rate generator is shown in Figure 4. This valid only if RCLK or TCLK = 1 in T2CON. Note llover in TH2 does not set TF2 and will not gener- nterrupt. Note too, that if EXEN2 is set, a 1-to-0 n in T2EX will set EXF2 but will not cause a reload CAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

Note that when Timer 2 is running (TR2 = 1) as a timer in the baud rate generator mode, TH2 or TL2 should not be read from or written to. Under these conditions, the Timer is incremented every state time, and the results of a read or write may not be accurate. The RCAP2 registers may be read but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Programmable Clock Out

A 50% duty cycle clock can be programmed to come out on P1.0, as shown in Figure 5. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed to input the external clock for Timer/Counter 2 or to output a 50% duty cycle clock ranging from 61 Hz to 4 MHz at a 16 MHz operating frequency.

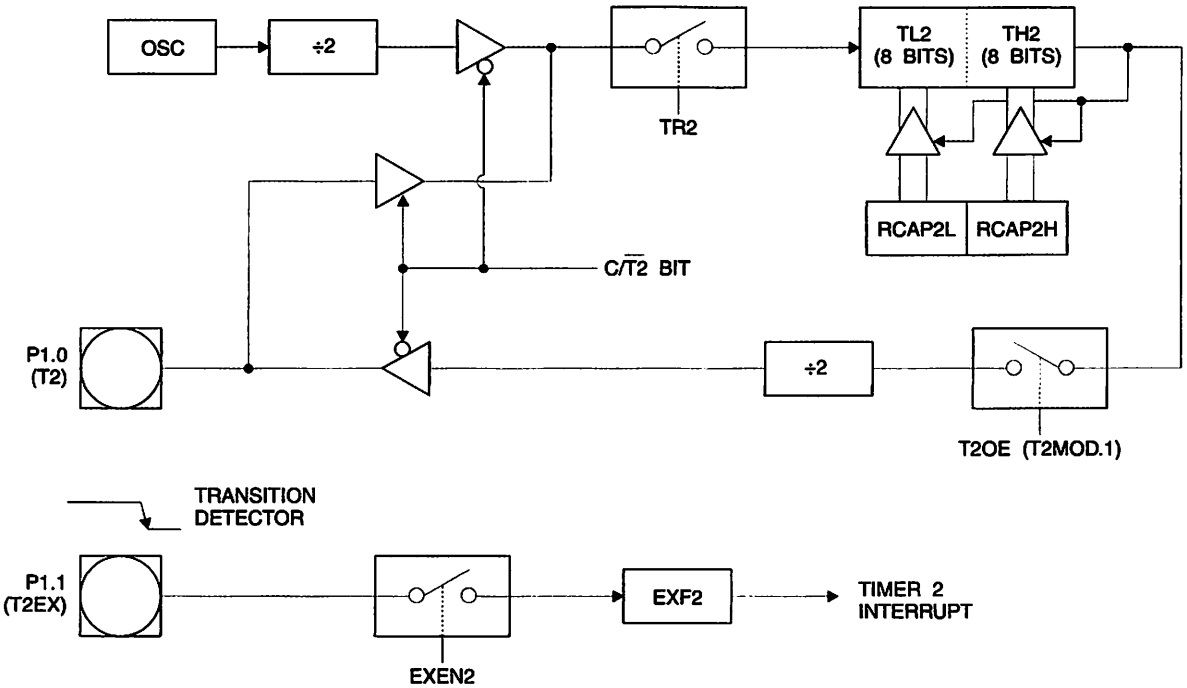
To configure the Timer/Counter 2 as a clock generator, bit $C/\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

The clock-out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L), as shown in the following equation.

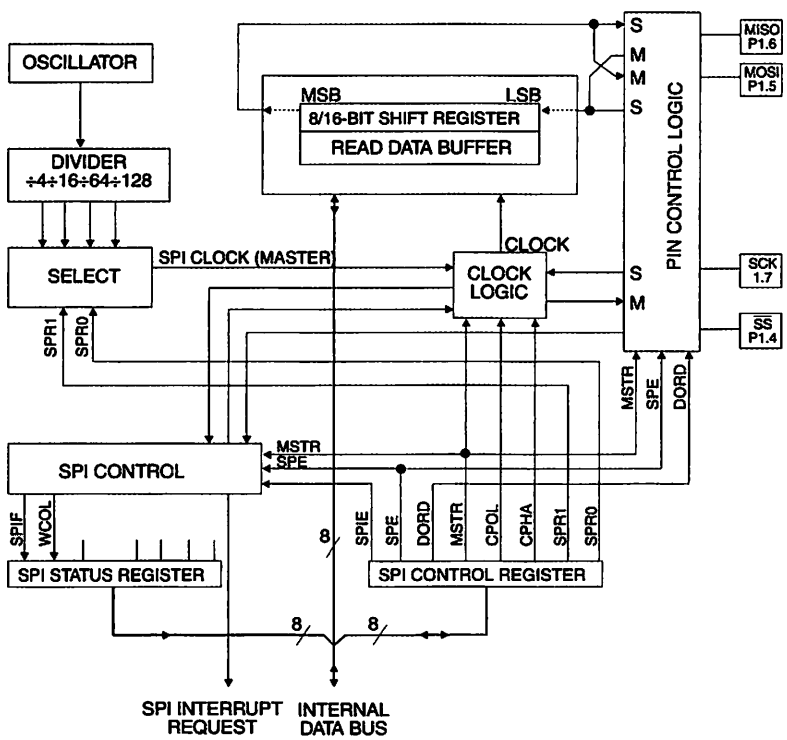
$$\text{Clock Out Frequency} = \frac{\text{Oscillator Frequency}}{4 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

In the clock-out mode, Timer 2 rollovers will not generate an interrupt. This behavior is similar to when Timer 2 is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and clock-out frequencies cannot be determined independently from one another since they both use RCAP2H and RCAP2L.

5. Timer 2 in Clock-out Mode



6. SPI Block Diagram



r

.RT in the AT89S8252 operates the same way as RT in the AT89C51, AT89C52 and AT89C55. For information, see the October 1995 Microcontroller ›ok, page 2-49, section titled, "Serial Interface."

l Peripheral Interface

ial peripheral interface (SPI) allows high-speed syn-us data transfer between the AT89S8252 and ›ral devices or between several AT89S8252 s. The AT89S8252 SPI features include the g:

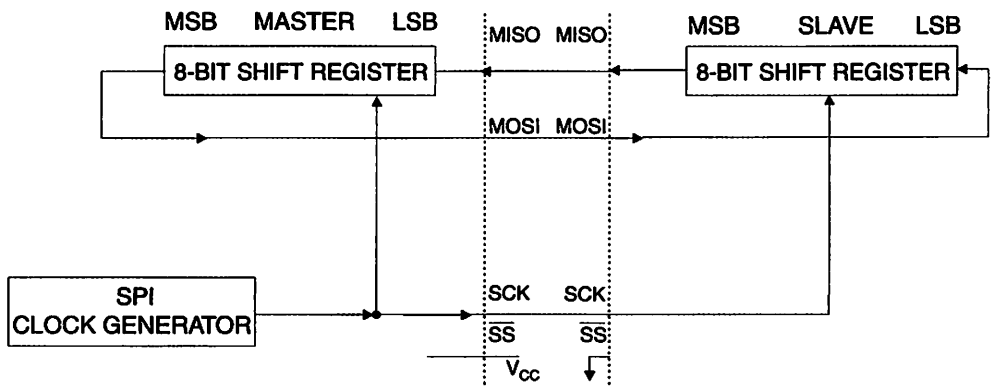
-)uplex, 3-Wire Synchronous Data Transfer
- ›r or Slave Operation
- lHz Bit Frequency (max.)
- First or MSB First Data Transfer
- Programmable Bit Rates
- ›f Transmission Interrupt Flag
- Write Collision Flag Protection
- Wakeup from Idle Mode (Slave Mode Only)

The interconnection between master and slave CPUs with SPI is shown in the following figure. The SCK pin is the clock output in the master mode but is the clock input in the slave mode. Writing to the SPI data register of the master CPU starts the SPI clock generator, and the data written shifts out of the MOSI pin and into the MOSI pin of the slave CPU. After shifting one byte, the SPI clock generator stops, setting the end of transmission flag (SPIF). If both the SPI interrupt enable bit (SPIE) and the serial port interrupt enable bit (ES) are set, an interrupt is requested.

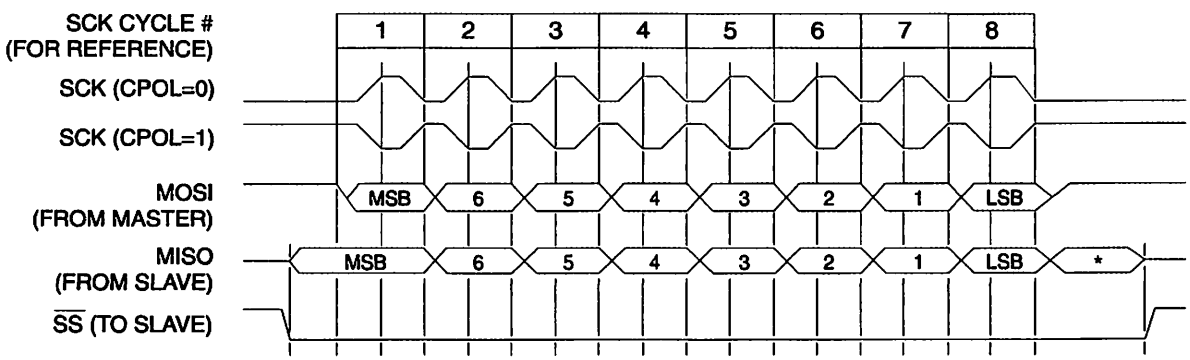
The Slave Select input, $\overline{SS}$/P1.4, is set low to select an individual SPI device as a slave. When $\overline{SS}$/P1.4 is set high, the SPI port is deactivated and the MOSI/P1.5 pin can be used as an input.

There are four combinations of SCK phase and polarity with respect to serial data, which are determined by control bits CPHA and CPOL. The SPI data transfer formats are shown in Figure 8 and Figure 9.

7. SPI Master-slave Interconnection

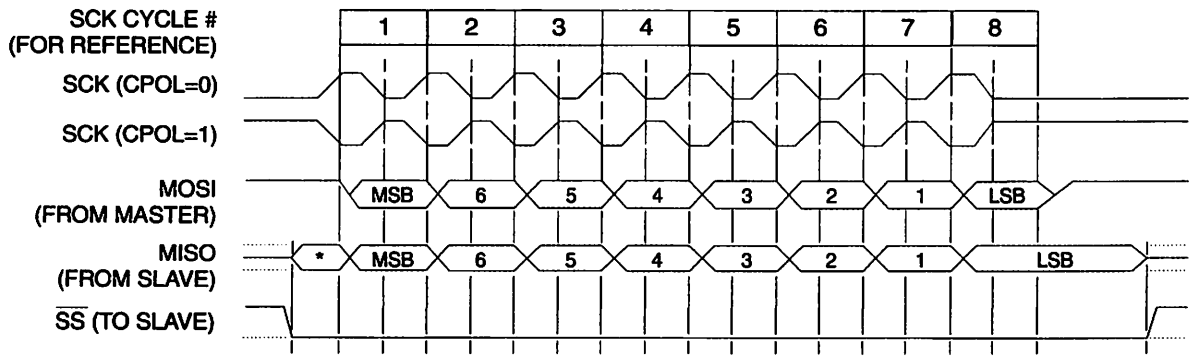


3. SPI transfer Format with CPHA = 0



ined but normally MSB of character just received

9. SPI Transfer Format with CPHA = 1



fined but normally LSB of previously transmitted character

'upts

89S8252 has a total of six interrupt vectors: two l interrupts (INT0 and INT1), three timer interrupts : 0, 1, and 2), and the serial port interrupt. These ts are all shown in Figure 10.

these interrupt sources can be individually enabled ıled by setting or clearing a bit in Special Function r IE. IE also contains a global disable bit, EA, which ; all interrupts at once.

ıt Table 10 shows that bit position IE.6 is unimple- ı. In the AT89C51, bit position IE.5 is also mented. User software should not write 1s to these itions, since they may be used in future AT89 s.

interrupt is generated by the logical OR of bits TF2 F2 in register T2CON. Neither of these flags is by hardware when the service routine is vectored ıct, the service routine may have to determine · it was TF2 or EXF2 that generated the interrupt, : bit will have to be cleared in software.

ıer 0 and Timer 1 flags, TF0 and TF1, are set at ' the cycle in which the timers overflow. The values ı polled by the circuitry in the next cycle. However, er 2 flag, TF2, is set at S2P2 and is polled in the 'cle in which the timer overflows.

Table 10. Interrupt Enable (IE) Register

(MSB)(LSB)

EA	–	ET2	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	SPI and UART interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.
User software should never write 1s to unimplemented bits, because they may be used in future AT89 products.		

10. Interrupt Sources

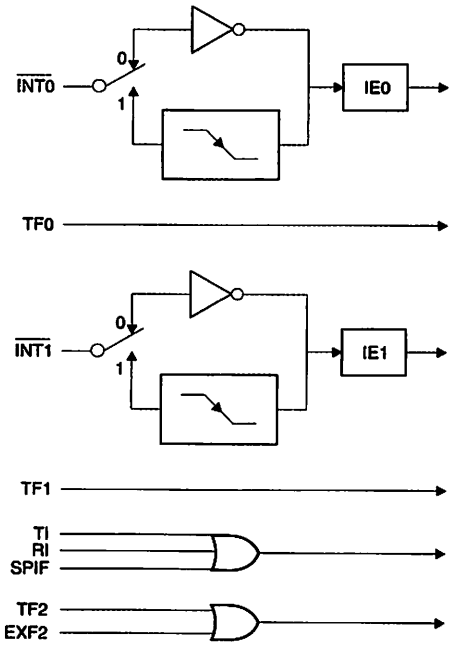
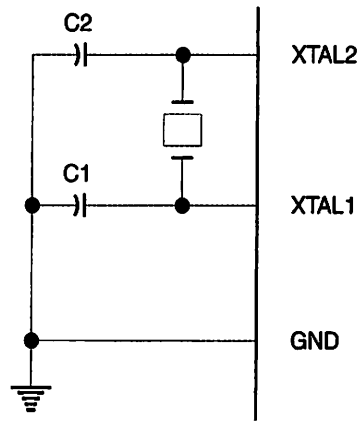
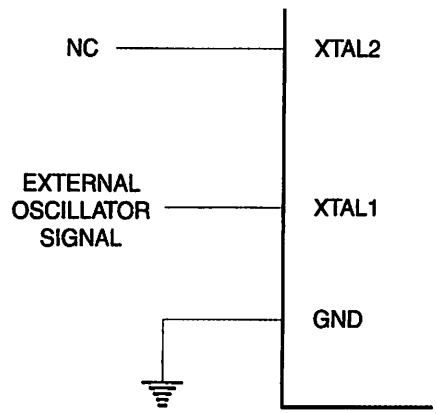


Figure 11. Oscillator Connections



Note: Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Figure 12. External Clock Drive Configuration



llator Characteristics

and XTAL2 are the input and output, respectively, ıverting amplifier that can be configured for use as hip oscillator, as shown in Figure 11. Either a quartz or ceramic resonator may be used. To drive the from an external clock source, XTAL2 should be left ected while XTAL1 is driven, as shown in Figure 12. ıre no requirements on the duty cycle of the external gnal, since the input to the internal clocking circuitry gh a divide-by-two flip-flop, but minimum and maxi- oltage high and low time specifications must be ıd.

Mode

mode, the CPU puts itself to sleep while all the on-
ripherals remain active. The mode is invoked by
e. The content of the on-chip RAM and all the spe-
ıctions registers remain unchanged during this
The idle mode can be terminated by any enabled
t or by a hardware reset.

ıat when idle mode is terminated by a hardware
he device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when idle mode is terminated by a reset, the instruction following the one that invokes idle mode should not write to a port pin or to external memory.

ıs of External Pins During Idle and Power-down Modes

	Program Memory	ALE	$\overline{PSEN}$	PORT0	PORT1	PORT2	PORT3
	Internal	1	1	Data	Data	Data	Data
	External	1	1	Float	Data	Address	Data
down	Internal	0	0	Data	Data	Data	Data
down	External	0	0	Float	Data	Data	Data

r-down Mode

ower-down mode, the oscillator is stopped and the
ion that invokes power-down is the last instruction
d. The on-chip RAM and Special Function Regis-
tain their values until the power-down mode is
ted. Exit from power-down can be initiated either by
are reset or by an enabled external interrupt. Reset
s the SFRs but does not change the on-chip RAM.
et should not be activated before V_{CC} is restored to
nal operating level and must be held active long
to allow the oscillator to restart and stabilize.

power-down via an interrupt, the external interrupt
enabled as level sensitive before entering power-
he interrupt service routine starts at 16 ms (nomi-
r the enabled interrupt pin is activated.

Program Memory Lock Bits

The AT89S8252 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

When lock bit 1 is programmed, the logic level at the $\overline{EA}$ pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value and holds that value until reset is activated. The latched value of $\overline{EA}$ must agree with the current logic level at that pin in order for the device to function properly.

Once programmed, the lock bits can only be unprogrammed with the Chip Erase operations in either the parallel or serial modes.

Bit Protection Modes(1)(2)

gram Lock Bits			
LB1	LB2	LB3	Protection Type
U	U	U	No internal memory lock feature.
P	U	U	MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory. $\overline{EA}$ is sampled and latched on reset and further programming of the Flash memory (parallel or serial mode) is disabled.
P	P	U	Same as Mode 2, but parallel or serial verify are also disabled.
P	P	P	Same as Mode 3, but external execution is also disabled.

1. U = Unprogrammed
2. P = Programmed

ıramming the Flash and EEPROM

ɜ AT89S8252 Flash Microcontroller offers 8K bytes /stem reprogrammable Flash Code memory and 2K ɔf EEPROM Data memory.

'89S8252 is normally shipped with the on-chip Flash ınd EEPROM Data memory arrays in the erased i.e. contents = FFH) and ready to be programmed. evice supports a High-voltage (12V) Parallel pro- ning mode and a Low-voltage (5V) Serial mming mode. The serial programming mode pro- convenient way to download the AT89S8252 inside ır's system. The parallel programming mode is com- with conventional third party Flash or EPROM nmers.

ıde and Data memory arrays are mapped via sepa- ldress spaces in the serial programming mode. In allel programming mode, the two arrays occupy one ıous address space: 0000H to 1FFFH for the Code nd 2000H to 27FFH for the Data array.

ıde and Data memory arrays on the AT89S8252 are nmed byte-by-byte in either programming mode. An ase cycle is provided with the self-timed program- peration in the serial programming mode. There is d to perform the Chip Erase operation to reprogram ɜmory location in the serial programming mode any of the lock bits have been programmed.

ɔarallel programming mode, there is no auto-erase o reprogram any non-blank byte, the user needs to Chip Erase operation first to erase both arrays.

l Programming Algorithm: To program and verify 39S8252 in the parallel programming mode, the fol- ɜequence is recommended:

ver-up sequence:

ɔly power between V_{CC} and GND pins.

RST pin to "H".

ɔly a 3 MHz to 24 MHz clock to XTAL1 pin and wait at least 10 milliseconds.

PSEN pin to "L"

E pin to "H"

pin to "H" and all other pins to "H".

ɔly the appropriate combination of "H" or "L" logic ɔls to pins P2.6, P2.7, P3.6, P3.7 to select one of programming operations shown in the Flash gramming Modes table.

ɔly the desired byte address to pins P1.0 to P1.7 P2.0 to P2.5.

ɔly data to pins P0.0 to P0.7 for Write Code ration.

5. Raise $\overline{EA}/V_{PP}$ to 12V to enable Flash programming, erase or verification.
6. Pulse ALE/$\overline{PROG}$ once to program a byte in the Code memory array, the Data memory array or the lock bits. The byte-write cycle is self-timed and typically takes 1.5 ms.
7. To verify the byte just programmed, bring pin P2.7 to "L" and read the programmed data at pins P0.0 to P0.7.
8. Repeat steps 3 through 7 changing the address and data for the entire 2K or 8K bytes array or until the end of the object file is reached.
9. Power-off sequence:

   Set XTAL1 to "L".

   Set RST and $\overline{EA}$ pins to "L".

   Turn V_{CC} power off.

In the parallel programming mode, there is no auto-erase cycle and to reprogram any non-blank byte, the user needs to use the Chip Erase operation first to erase both arrays.

$\overline{Data}$ Polling: The AT89S8252 features $\overline{DATA}$ Polling to indicate the end of a write cycle. During a write cycle in the parallel or serial programming mode, an attempted read of the last byte written will result in the complement of the written datum on P0.7 (parallel mode), and on the MSB of the serial output byte on MISO (serial mode). Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. $\overline{DATA}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{Busy}$: The progress of byte programming in the parallel programming mode can also be monitored by the RDY/$\overline{BSY}$ output signal. Pin P3.4 is pulled Low after ALE goes High during programming to indicate $\overline{BUSY}$. P3.4 is pulled High again when programming is done to indicate READY.

Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed Code or Data byte can be read back via the address and data lines for verification. The state of the lock bits can also be verified directly in the parallel programming mode. In the serial programming mode, the state of the lock bits can only be verified indirectly by observing that the lock bit features are enabled.

Chip Erase: Both Flash and EEPROM arrays are erased electrically at the same time. In the parallel programming mode, chip erase is initiated by using the proper combination of control signals and by holding ALE/$\overline{PROG}$ low for 10 ms. The Code and Data arrays are written with all "1"s in the Chip Erase operation.

;erial programming mode, a chip erase operation is 1 by issuing the Chip Erase instruction. In this mode, ase is self-timed and takes about 16 ms.

chip erase, a serial read from any address location ırn 00H at the data outputs.

Programming Fuse: A programmable fuse is avail- ı disable Serial Programming if the user needs ım system security. The Serial Programming Fuse y be programmed or erased in the Parallel Program- lode.

T89S8252 is shipped with the Serial Programming ınabled.

ıg the Signature Bytes: The signature bytes are y the same procedure as a normal verification of ıs 030H and 031H, except that P3.6 and P3.7 must ıd to a logic low. The values returned are as follows:

0H) = 1EH indicates manufactured by Atmel
1H) = 72H indicates 89S8252

ramming Interface

;ode byte in the Flash and EEPROM arrays can be , and the entire array can be erased, by using the riate combination of control signals. The write oper- cycle is self-timed and once initiated, will tically time itself to completion.

ɔr programming vendors offer worldwide support for ıel microcontroller series. Please contact your local nming vendor for the appropriate software revision.

l Downloading

ıe Code and Data memory arrays can be pro- ed using the serial SPI bus while RST is pulled to ə serial interface consists of pins SCK, MOSI (input) SO (output). After RST is set high, the Programming instruction needs to be executed first before pro- ase operations can be executed.

-erase cycle is built into the self-timed programming ɔn (in the serial mode ONLY) and there is no need ıxecute the Chip Erase instruction unless any of the s have been programmed. The Chip Erase opera- ıs the content of every memory location in both the ıd Data arrays into FFH.

de and Data memory arrays have separate address 0000H to 1FFFH for Code memory and 000H to 7FFH for Data memory.

Either an external system clock is supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK) frequency should be less than 1/40 of the crystal frequency. With a 24 MHz oscillator clock, the maximum SCK frequency is 600 kHz.

Serial Programming Algorithm

To program and verify the AT89S8252 in the serial programming mode, the following sequence is recommended:

1. Power-up sequence:

   Apply power between VCC and GND pins.

   Set RST pin to "H".

   If a crystal is not connected across pins XTAL1 and XTAL2, apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Enable serial programming by sending the Programming Enable serial instruction to pin MOSI/P1.5. The frequency of the shift clock supplied at pin SCK/P1.7 needs to be less than the CPU clock at XTAL1 divided by 40.
3. The Code or Data array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. The selected memory location is first automatically erased before new data is written. The write cycle is self-timed and typically takes less than 2.5 ms at 5V.
4. Any memory location can be verified by using the Read instruction which returns the content at the selected address at serial output MISO/P1.6.
5. At the end of a programming session, RST can be set low to commence normal operation.

Power-off sequence (if needed):

Set XTAL1 to "L" (if a crystal is not used).

Set RST to "L".

Turn V_{CC} power off.

Serial Programming Instruction

The Instruction Set for Serial Programming follows a 3-byte protocol and is shown in the following table:

uction Set

ıction	Input Format			Operation
	Byte 1	Byte 2	Byte 3	
ımming Enable	1010 1100	0101 0011	xxxx xxxx	Enable serial programming interface after RST goes high.
Erase	1010 1100	xxxx x100	xxxx xxxx	Chip erase both 8K & 2K memory arrays.
Code Memory	aaaa a001	low addr	xxxx xxxx	Read data from Code memory array at the selected address. The 5 MSBs of the first byte are the high order address bits. The low order address bits are in the second byte. Data are available at pin MISO during the third byte.
Code Memory	aaaa a010	low addr	data in	Write data to Code memory location at selected address. The address bits are the 5 MSBs of the first byte together with the second byte.
Data Memory	00aa a101	low addr	xxxx xxxx	Read data from Data memory array at selected address. Data are available at pin MISO during the third byte.
Data Memory	00aa a110	low addr	data in	Write data to Data memory location at selected address.
Lock Bits	1010 1100	x x111	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

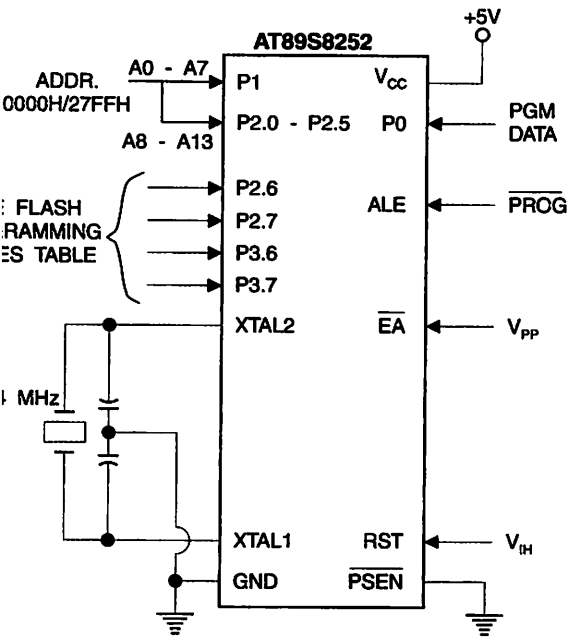
1. DATA polling is used to indicate the end of a write cycle which typically takes less than 2.5 ms at 5V.
2. "aaaaa" = high order address.
3. "x" = don't care.

n and EEPROM Parallel Programming Modes

	RST	$\overline{PSEN}$	ALE/$\overline{PROG}$	$\overline{EA}/V_{PP}$	P2.6	P2.7	P3.6	P3.7	Data I/O P0.7:0	Address P2.5:0 P1.7:0
Prog. Modes	H	h(1)	h(1)	x						
ase	H	L	⌐‾⌐ (2)	12V	H	L	L	L	X	X
10K bytes) Memory	H	L	⌐‾⌐	12V	L	H	H	H	DIN	ADDR
10K bytes) Memory	H	L	H	12V	L	L	H	H	DOUT	ADDR
ock Bits:	H	L	⌐‾⌐	12V	H	L	H	L	DIN	X
Bit - 1									P0.7 = 0	X
Bit - 2									P0.6 = 0	X
Bit - 3									P0.5 = 0	X
ock Bits:	H	L	H	12V	H	H	L	L	DOUT	X
Bit - 1									@P0.2	X
Bit - 2									@P0.1	X
Bit - 3									@P0.0	X
tmel Code	H	L	H	12V	L	L	L	L	DOUT	30H
evice Code	H	L	H	12V	L	L	L	L	DOUT	31H
rog. Enable	H	L	⌐‾⌐ (2)	12V	L	H	L	H	P0.0 = 0	X
rog. Disable	H	L	⌐‾⌐ (2)	12V	L	H	L	H	P0.0 = 1	X
erial Prog. Fuse	H	L	H	12V	H	H	L	H	@P0.0	X

1. "h" = weakly pulled "High" internally.
2. Chip Erase and Serial Programming Fuse require a 10 ms $\overline{PROG}$ pulse. Chip Erase needs to be performed first before reprogramming any byte with a content other than FFH.
3. P3.4 is pulled Low during programming to indicate RDY/$\overline{BSY}$.
4. "X" = don't care

13. Programming the Flash/EEPROM Memory



14. Verifying the Flash/EEPROM Memory

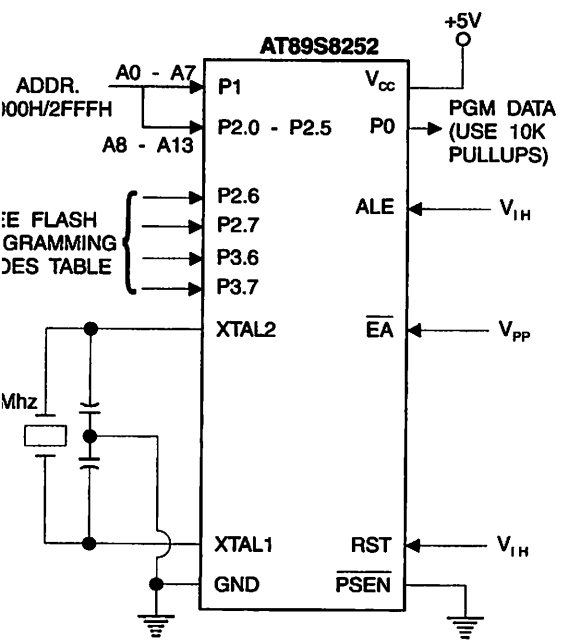
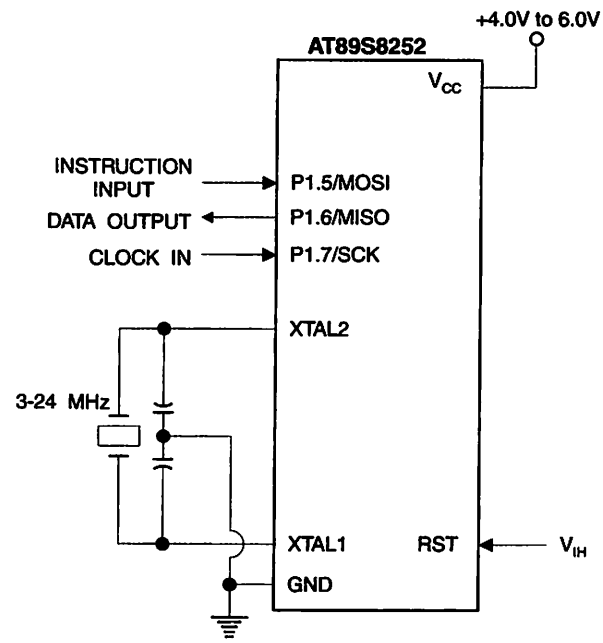


Figure 15. Flash/EEPROM Serial Downloading

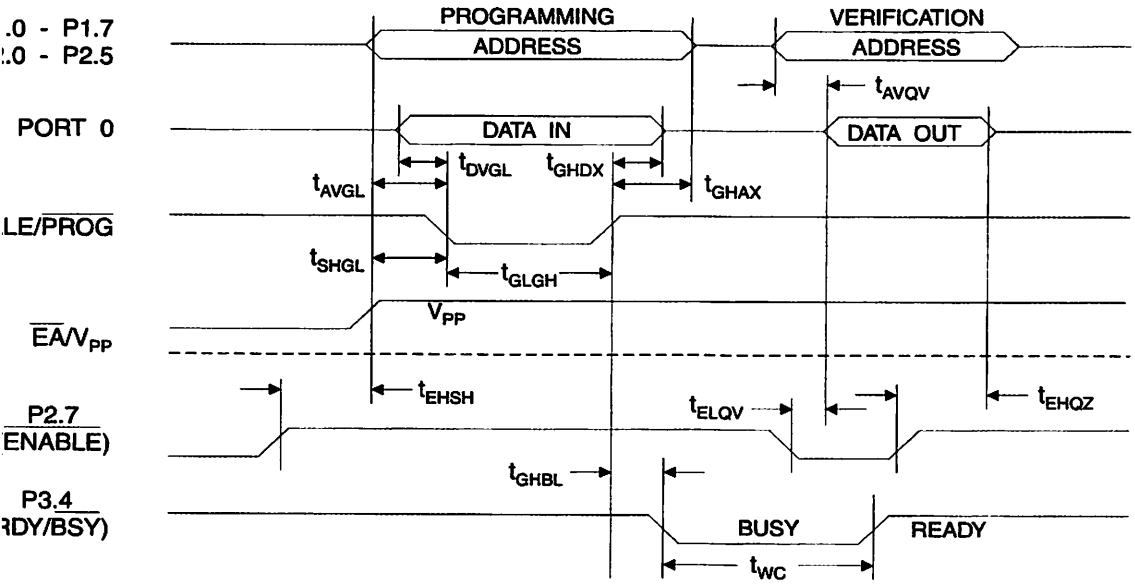


h Programming and Verification Characteristics – Parallel Mode

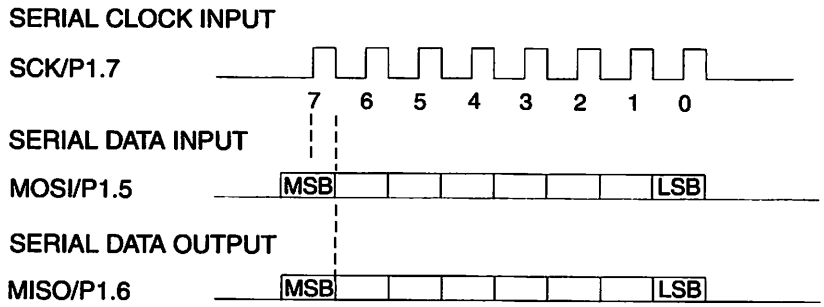
C to 70°C, V_{CC} = 5.0V ± 10%

ol	Parameter	Min	Max	Units
	Programming Enable Voltage	11.5	12.5	V
	Programming Enable Current		1.0	mA
	Oscillator Frequency	3	24	MHz
	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
	Data Hold after $\overline{PROG}$	$48t_{CLCL}$		
	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
	$\overline{PROG}$ Width	1	110	µs
	Address to Data Valid		$48t_{CLCL}$	
	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
	Data Float after $\overline{ENABLE}$	0	$48t_{CLCL}$	
	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
	Byte Write Cycle Time		2.0	ms

h/EEPROM Programming and Verification Waveforms – Parallel Mode



l Downloading Waveforms



olute Maximum Ratings*

ting Temperature	-55°C to +125°C
ge Temperature	-65°C to +150°C
e on Any Pin lespect to Ground	-1.0V to +7.0V
um Operating Voltage	6.6V
utput Current	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Characteristics

lues shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 5.0V ± 20%, unless otherwise noted.

ol	Parameter	Condition	Min	Max	Units
	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC} - 0.1	V
	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC} - 0.3	V
	Input Hifh-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
	Input Hifh-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC} + 0.5	V
	Output Low-voltage (1) (Ports 1,2,3)	I_{OL} = 1.6 mA		0.5	V
	Output Low-voltage (1) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.5	V
	Output Hifh-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
	Output Hifh-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-650	µA
	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
	Reset Pull-down Resistor		50	300	KΩ
	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode (2)	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 10 mA
   Maximum I_{OL} per 8-bit port:
   Port 0: 26 mA
   Ports 1, 2, 3: 15 mA
   Maximum total I_{OL} for all output pins: 71 mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V

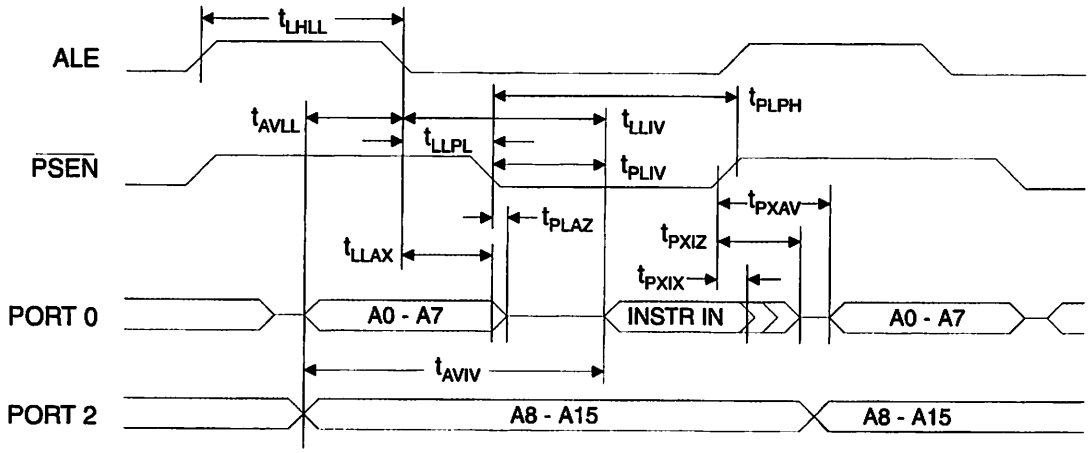
Characteristics

operating conditions, load capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; load capacitance for all other s = 80 pF.

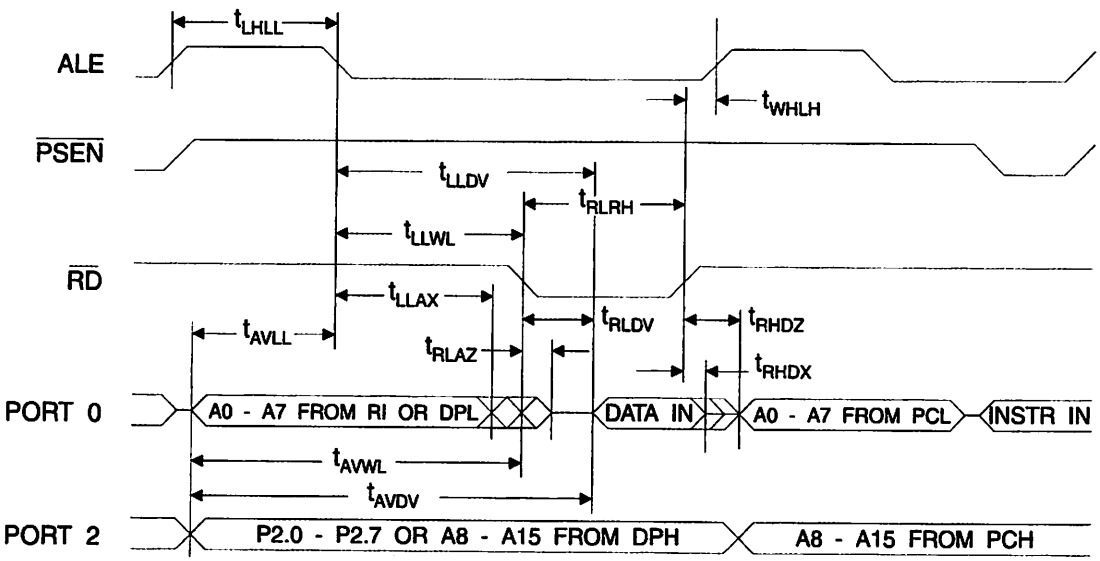
rnal Program and Data Memory Characteristics

ol	Parameter	Variable Oscillator		Units
		Min	Max	
L	Oscillator Frequency	0	24	MHz
	ALE Pulse Width	$2t_{CLCL}$ - 40		ns
	Address Valid to ALE Low	t_{CLCL} - 13		ns
	Address Hold after ALE Low	t_{CLCL} - 20		ns
	ALE Low to Valid Instruction In		$4t_{CLCL}$ - 65	ns
	ALE Low to $\overline{PSEN}$ Low	t_{CLCL} - 13		ns
	$\overline{PSEN}$ Pulse Width	$3t_{CLCL}$ - 20		ns
	$\overline{PSEN}$ Low to Valid Instruction In		$3t_{CLCL}$ - 45	ns
	Input Instruction Hold after $\overline{PSEN}$	0		ns
	Input Instruction Float after $\overline{PSEN}$		t_{CLCL} - 10	ns
	$\overline{PSEN}$ to Address Valid	t_{CLCL} - 8		ns
	Address to Valid Instruction In		$5t_{CLCL}$ - 55	ns
	$\overline{PSEN}$ Low to Address Float		10	ns
	$\overline{RD}$ Pulse Width	$6t_{CLCL}$ - 100		ns
	$\overline{WR}$ Pulse Width	$6t_{CLCL}$ - 100		ns
	$\overline{RD}$ Low to Valid Data In		$5t_{CLCL}$ - 90	ns
	Data Hold after $\overline{RD}$	0		ns
	Data Float after $\overline{RD}$		$2t_{CLCL}$ - 28	ns
	ALE Low to Valid Data In		$8t_{CLCL}$ - 150	ns
	Address to Valid Data In		$9t_{CLCL}$ - 165	ns
	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	$3t_{CLCL}$ - 50	$3t_{CLCL}$ + 50	ns
	Address to $\overline{RD}$ or $\overline{WR}$ Low	$4t_{CLCL}$ - 75		ns
	Data Valid to $\overline{WR}$ Transition	t_{CLCL} - 20		ns
	Data Valid to $\overline{WR}$ High	$7t_{CLCL}$ - 120		ns
	Data Hold after $\overline{WR}$	t_{CLCL} - 20		ns
	$\overline{RD}$ Low to Address Float		0	ns
	$\overline{RD}$ or $\overline{WR}$ High to ALE High	t_{CLCL} - 20	t_{CLCL} + 25	ns

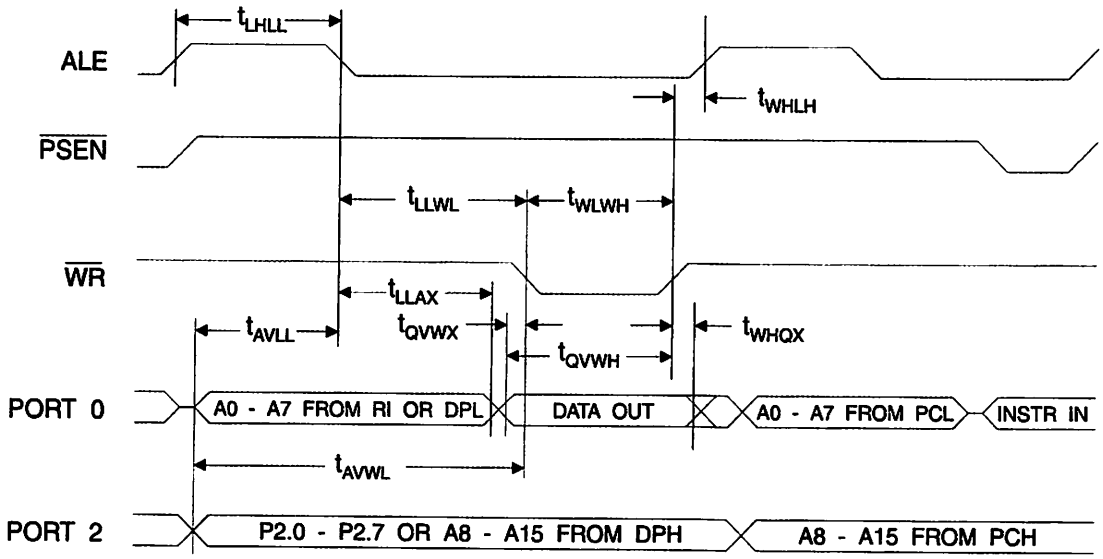
rnal Program Memory Read Cycle



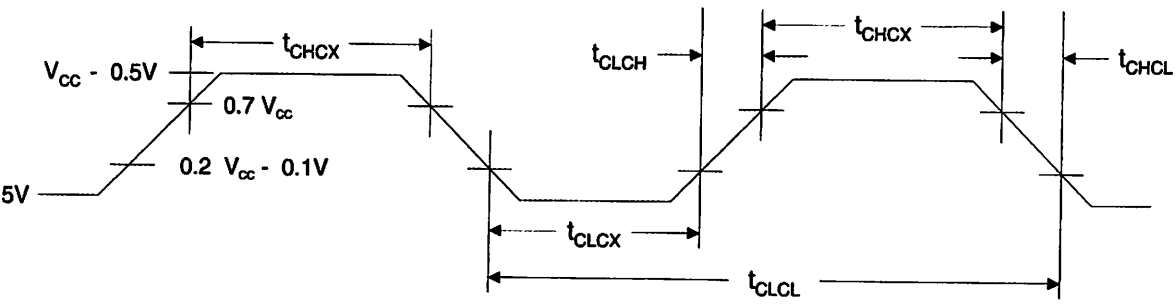
rnal Data Memory Read Cycle



ernal Data Memory Write Cycle



rnal Clock Drive Waveforms



rnal Clock Drive

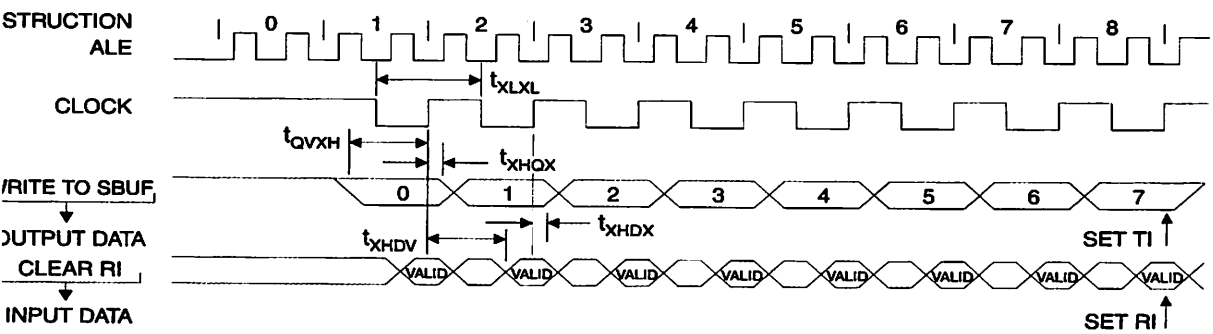
ol	Parameter	V_{CC} = 4.0V to 6.0V		Units
		Min	Max	
	Oscillator Frequency	0	24	MHz
	Clock Period	41.6		ns
	High Time	15		ns
	Low Time	15		ns
	Rise Time		20	ns
	Fall Time		20	ns

al Port Timing: Shift Register Mode Test Conditions

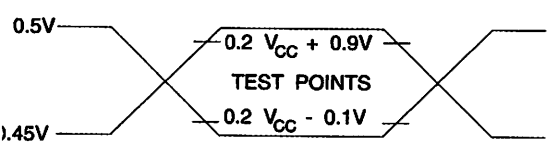
alues in this table are valid for V_{CC} = 4.0V to 6V and Load Capacitance = 80 pF.

ɔol	Parameter	Variable Oscillator		Units
		Min	Max	
	Serial Port Clock Cycle Time	$12t_{CLCL}$		µs
	Output Data Setup to Clock Rising Edge	$10t_{CLCL}$ - 133		ns
	Output Data Hold after Clock Rising Edge	$2t_{CLCL}$ - 117		ns
	Input Data Hold after Clock Rising Edge	0		ns
	Clock Rising Edge to Input Data Valid		$10t_{CLCL}$ - 133	ns

t Register Mode Timing Waveforms

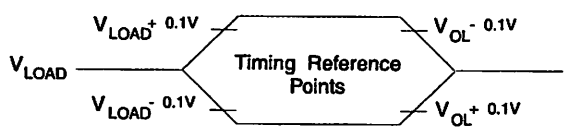


'esting Input/Output Waveforms(1)



1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

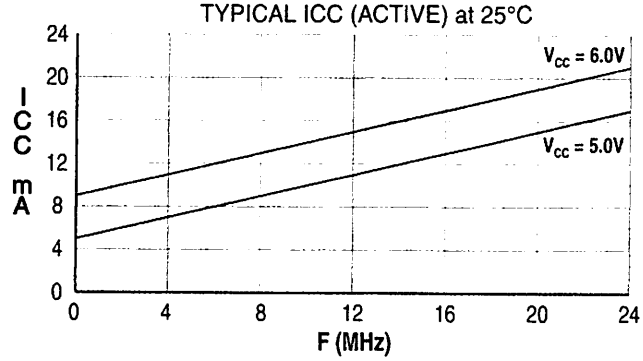
Float Waveforms(1)



Notes: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

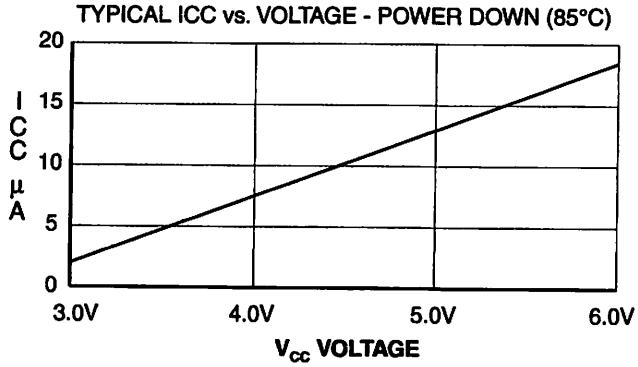
AT89S8252

TYPICAL ICC (ACTIVE) at 25°C



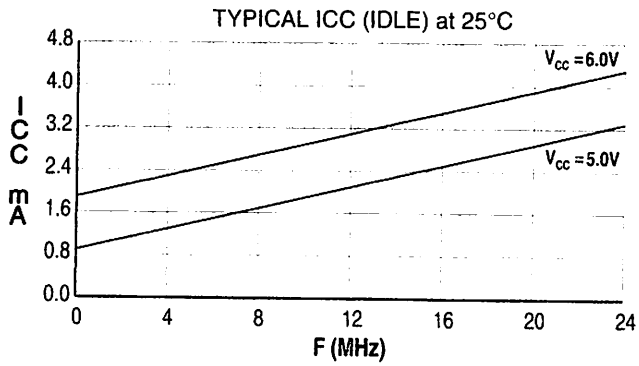
AT89S8252

TYPICAL ICC vs. VOLTAGE - POWER DOWN (85°C)



AT89S8252

TYPICAL ICC (IDLE) at 25°C



Notes: 1. XTAL1 tied to GND for Icc (power-down)
2. Lock bits programmed

ering Information

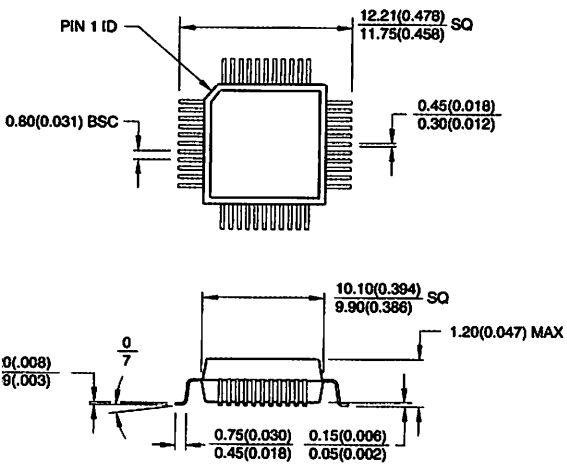
eed Hz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 6.0V	AT89S8252-24AC AT89S8252-24JC AT89S8252-24PC AT89S8252-24QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)
	4.0V to 6.0V	AT89S8252-24AI AT89S8252-24JI AT89S8252-24PI AT89S8252-24QI	44A 44J 40P6 44Q	Industrial (-40°C to 85°C)
33	4.5V to 5.5V	AT89S8252-33AC AT89S8252-33JC AT89S8252-33PC AT89S8252-33QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)

= Preliminary Information

	Package Type
	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
	44-lead, Plastic J-leaded Chip Carrier (PLCC)
	40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
	44-lead, Plastic Gull Wing Quad Flatpack (PQFP)

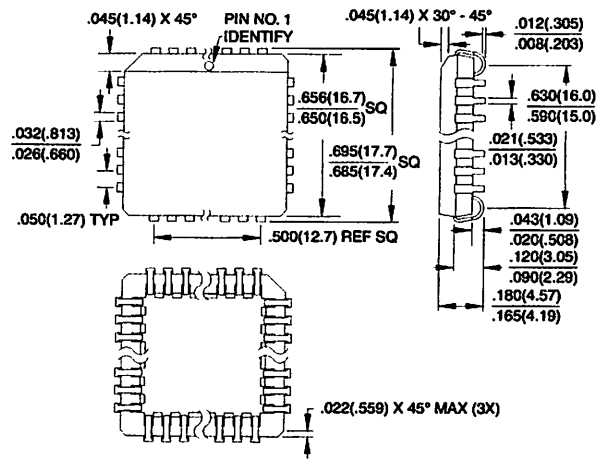
kaging Information

4A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad
latpack (TQFP)
imensions in Millimeters and (Inches)*
:DEC STANDARD MS-026 ACB

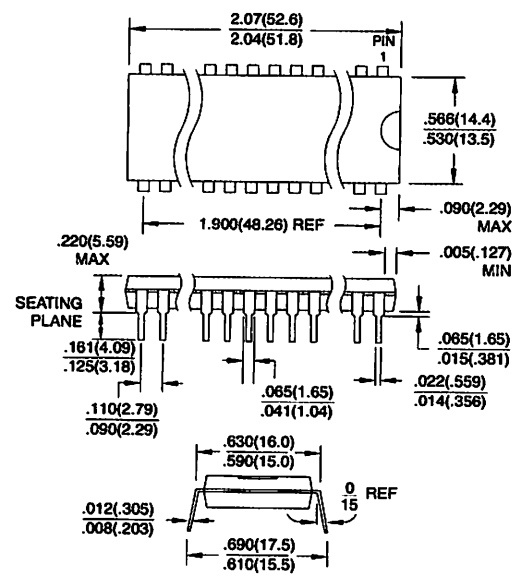


ntrolling dimension: millimeters

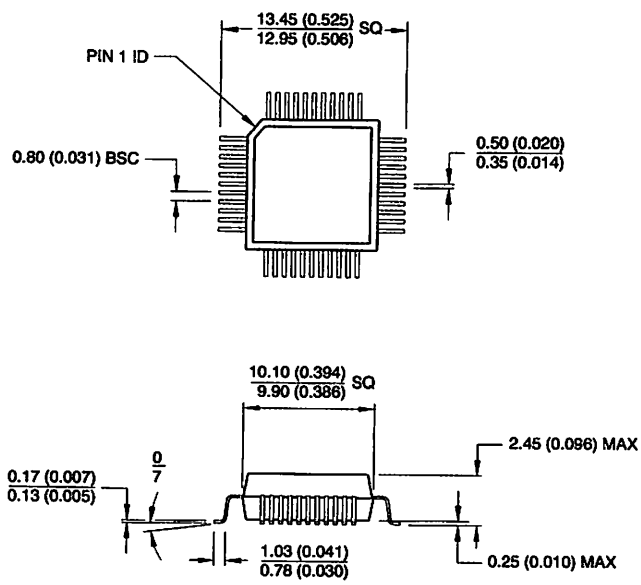
44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-018 AC



P6, 40-lead, 0.600" Wide, Plastic Dual Inline
ckage (PDIP)
nensions in Inches and (Millimeters)



44Q, 44-lead, Plastic Quad Flat Package (PQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-022 AB



Controlling dimension: millimeters

ıel Headquarters

ıorate Headquarters
25 Orchard Parkway
n Jose, CA 95131
L (408) 441-0311
X (408) 487-2600

ıpe
nel U.K., Ltd.
liseum Business Centre
'erside Way
mberley, Surrey GU15 3YL
gland
L (44) 1276-686-677
X (44) 1276-686-697

ıel Asia, Ltd.
ɔm 1219
nachem Golden Plaza
Mody Road Tsimhatsui
ɜt Kowloon
ıg Kong
. (852) 2721-9778
K (852) 2722-1369

ı
ıel Japan K.K.
Tonetsu Shinkawa Bldg.
4-8 Shinkawa
ıo-ku, Tokyo 104-0033
an
. (81) 3-3523-3551
((81) 3-3523-7581

Atmel Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

0401E–02/00/xM

Manual Reference

AT Command Set
(GSM 07.07, GSM 07.05)

for SIEMENS Mobile Phone

S25

and Derivatives

Table of Contents

1. Software Interface

1.1. Overview of the Supported AT Command Set

Page	Commands 07.07	Function	E10	S10	S10 act	Rel aun ch	C25	S25
6	AT+CGMI	Issue manufacturer ID code	✔	✔	✔	✔	✔	✔
6	AT+CGMM	Issue model ID code	✔	✔	✔	✔	✔	✔
6	AT+CGMR	Output the GSM telephone version	✔	✔	✔	✔	✔	✔
7	AT+CGSN	Output the serial number (IMEI)	✔	✔	✔	✔	✔	✔
7	AT+GSN	Output the serial number (IMEI)	✔	✔	✔	✔	✔	✔
7	AT+CHUP	Terminate call	✔	✔	✔	✔	✔	✔
7	AT+CEER	Query the reason for disconnection of last call	✔	✔	✔	✔	✔	✔
8	AT+CREG	Power status	✔	✔	✔	✔	✔	✔
8	AT+COPS	Commands concerning selection of network operator	✔	✔	✔	✔	✔	✔
9	AT+CLCK	Switch blocking on and off	✔	✔	✔	✔	✔	✔
9	AT+CPWD	Change password to a block	✔	✔	✔	✔	✔	✔
10	AT+CLIP	Display telephone number of calling party	✔	✔	✔	✔	✔	✔
10	AT+CCFC	Call forwarding	✔	✔	✔	✔	✔	✔
11	AT+CHLD	Call hold and multiparty	✔	✔	✔	✔	✔	✔
11	AT+CPAS	Query the telephone status	✔	✔	✔	✔	✔	✔
12	AT+CPIN	Enter PIN and query block	✔	✔	✔	✔	✔	✔
12	AT+CBC	Battery charge	✔	✔	✔	✔	✔	✔
13	AT+CSQ	Output signal quality	✔	✔	✔	✔	✔	✔
13	AT+CPBS	Select a telephone book	✔	✔	✔	✔	✔	✔
14	AT+CPBR	Read a telephone-book entry	✔	✔	✔	✔	✔	✔
14	AT+CPBW	Write a telephone-book entry	✔	✔	✔	✔	✔	✔
15	AT+CMEE	Expanded error messages according to GSM 07.07	✔	✔	✔	✔	✔	✔
16	AT+VTS	Send a DTMF tone	✔	✔	✔	✔	✔	✔
17	AT+VTD	Set duration of a DTMF tone	✔	✔	✔	✔	✔	✔
17	AT+WS46	Select wireless network	✔	✔	✔	✔	✔	✔
17	AT+CSCS	Select TE character set	✔	✔	✔	✔	✔	✔
18	AT+CAOC	Advice of charge	✔	✔	✔	✔	✔	✔
18	AT+CSSN	Supplementary service notifications	✔	✔	✔	✔	✔	✔
19	AT+CRSM	Restricted SIM access			✔	✔	✔	✔
19	AT+CIMI	Output of IMSI			✔	✔	✔	✔
20	AT+CACM	Accumulated call meter				✔	✔	✔
20	AT+CAMM	Accumulated call meter maximum				✔	✔	✔
21	AT+CLCC	List Current Calls						✔
22	AT+CCLK	Clock						✔
22	AT+COPN	Read operator names						✔

Page	Commands 07.05	Function	E10	S10	S10 act	Rel aun ch	C25	S25
23	AT+CSMS	Selection of message service	✔	✔	✔	✔	✔	✔
24	AT+CPMS	Selection of SMS memory	✔	✔	✔	✔	✔	✔
24	AT+CMGF	SMS format	✔	✔	✔	✔	✔	✔
25	AT+CSCA	Address of the SMS service center	✔	✔	✔	✔	✔	✔
25	AT+CNMI	Display new incoming SMS	✔	✔	✔	✔	✔	✔
27	AT+CNMA	Acknowledgment of a short message directly output	✔	✔	✔	✔	✔	✔
27	AT+CMGL	List SMS	✔	✔	✔	✔	✔	✔
28	AT+CMGR	Read in an SMS	✔	✔	✔	✔	✔	✔
28	AT+CMGS	Send an SMS	✔	✔	✔	✔	✔	✔
28	AT+CMSS	Send an SMS from the SMS memory	✔	✔	✔	✔	✔	✔
29	AT+CMGW	Write an SMS to the SMS memory	✔	✔	✔	✔	✔	✔
29	AT+CMGD	Delete an SMS in the SMS memory	✔	✔	✔	✔	✔	✔
29	AT+CSCB	Select cell broadcast messages			✔	✔	✔	✔
29	AT+CMGC	Send an SMS command			✔	✔	✔	✔

1.2. AT Command Set

Remote-control operation of the GSM mobile telephone runs via a serial interface, where AT+C commands according to GSM 07.07 and GSM 07.05 as well as several manufacturer-specific AT commands are available. These commands are described in more detail on the following pages.

The commands are entered by way of the operating functions of the respective base unit. This converts the operating functions to AT commands so that the mobile phone can execute the required action. The following should be noted:

The modem guideline V.25ter applies to the sequence of the interface commands. According to this guideline, commands should begin with the character string **"AT"** and end with **"<CR>" (= 0x0D).** The input of a command is acknowledged by the display of "OK" or "ERROR". **A command currently in process is interrupted by each additional character entered.** This means that you should not enter the next command until you have received the acknowledgment; otherwise the current command is interrupted.

The commands supported are listed in the following tables:

1.2.1. Hayes-Standard Commands

The Hayes-standard commands correspond to the commands of AT Hayes-compatible modems.

Command	Function
A/	Repeat last command
AT...	Prefix for all other commands
ATA	Accept call
ATD<str>;	Dial the dialing string <str> with the voice utility Valid dial modifiers: "T" (tone dialing), "P" (pulse dialing) is ignored. The character ";" is important, for this tells the phone that the call should be set up with the voice utility. Otherwise an attempt is made to set up a data call, which the phone immediately acknowledges with "ERROR".
ATD><n>;	Dial the telephone number from the current telephone book location number <n> The telephone book is selected with the command at+cpbs
ATD><mem> <n>;	Dial the telephone number from the telephone book <mem> location number <n>
ATDL	Dial last telephone number
ATE0	Deactivate command echo
ATE1	Activate command echo
ATH[0]	Separate connection
ATQ0	Display acknowledgments
ATQ1	Suppress acknowledgments
ATV0	Output acknowledgments as numbers

Command	Function
ATV1	Output acknowledgments as text
AT&F[0]	Reset to stored profile
AT&V	Display active and stored profiles
ATZ	Set to default configuration
AT+GCAP	Output the capabilities list

1.2.2. Acknowledgments for Normal Data Communication

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

1.3. AT Commands and Responses According to GSM 07.07 and GSM 07.05

According to GSM, it is possible to execute an AT command in various forms.

Test command	AT+CXXX=?	The telephone responds by sending the list of parameters and value ranges; these can be set using the affiliated Write command or by means of internal processes.
Read command	AT+CXXX?	This command tells you the current value setting of the parameter(s).
Write command	AT+CXXX=<...>	This command is used to set parameters that can be set.
Execute command	AT+CXXX	The Execute command reads non-settable parameters which are influenced by internal processes in the telephone.

1.3.1. AT Cellular Commands According to GSM 07.07

AT+CGMI	Issue manufacturer ID code
Test command AT+CGMI=?	Response OK
Execute command AT+CGMI	Response <manufacturer> Parameter <manufacturer> Name of manufacturer (SIEMENS) **Important**: There is a leading output prefix +CGMI in models before the S25.

AT+CGMM	Issue model ID code
Test command AT+CGMM=?	Response OK
Execute command AT+CGMM	Response <model> Parameter <model> Name of telephone (MOBILE) **Important**: There is a leading output prefix +CGMM in models before the S25.

AT+CGMR	Output the GSM telephone version
Test command AT+CGMR=?	Response OK
Execute command AT+CGMR	Response <revision> Parameter <revision> Version of the telephone software **Important**: There is a leading output prefix +CGMR in models before the S25.

AT+CGSN	Output the serial number (IMEI)
Test command AT+CGSN=?	Response OK
Execute command AT+CGSN	Response <sn> Parameter <sn> IMEI of the telephone **Important**: There is a leading output prefix +CGMI in models before the S25.

AT+GSN	Output the serial number (IMEI)
Test command AT+GSN=?	Response OK
Execute command AT+GSN	Response +GSN: <sn> Parameter <sn> IMEI of the telephone **Important**: The output prefix +GSN may be missing in future versions.

AT+CHUP	Terminate call
Test command AT+CHUP=?	Response OK
Execute command AT+CHUP	Response OK/ERROR
	Description: All active calls and all calls on hold are terminated.

AT+CEER	Query the reason for disconnection of last call
Test command AT+CEER=?	Response OK
Execute command AT+CEER	Response +CEER: <report> Parameter <report> Disconnection reason reported as number

AT+CREG	Power status
Test command AT+CREG=?	Response +CREG: (list of supported <n>s) OK/ERROR/+CME ERROR
	Parameter <n> 0 Suppresses the unexpected network-status messages 1 Displays the unexpected network-status messagesOK/ERROR/+CME ERROR
Read command AT+CREG?	Response +CREG: <n>,<stat>[,<lac>,<ci>] OK/ERROR/+CME ERROR
	Parameter <n> See Test command <stat> 0 Not checked in, not seeking 1 Checked in 2 Not checked in, but seeking a network 3 Check-in denied by network 4 Unknown 5 Registered, roaming <lac> Hexadecimal 2-byte string type of location area code <ci> Hexadecimal 2-byte string type of cell ID
Write command AT+CREG=<n>	Parameter <n> See Test command Response OK/ERROR/+CME ERROR
	Unexpected message +CREG: <stat>

AT+COPS	Commands concerning selection of network operator
Test command AT+COPS=?	Response +COPS: [list of supported (<stat>,long alphanumeric <oper>,,numeric <oper>)s][,,(list of supported <mode>s),(list of supported <format>s)] OK/ERROR/+CME ERROR Parameter <stat> 0 Unknown 1 Useful network operator 2 Used network operator 3 Prohibited network operator <oper> Operator in the format according to <mode>
Read command AT+COPS?	Response +COPS: <mode>[,<format>,<oper] OK/ERROR/+CME ERROR Parameter <mode> 0 Automatic mode 1 Manual selection of network operator 3 Setting of format 4 Automatic, manual selected <format> 0 Long alphanumeric 2 Numeric <oper> <oper> Network operator
Write command AT+COPS=<mode>[,<format>[,<oper]]	Parameter <mode> See Read command <format> See Read command If <mode> = 1, <format> can only = 2 <oper> In numeric form only Response OK/ERROR/+CME ERROR

AT+CLCK	Switch blocking on and off **Revision to GSM 07.07 according to CR TDOC ETSI/SMG4 187/96**
Test command AT+CLCK=?	Response +CLCK: (list of supported <fac>s) OK/ERROR/+CME ERROR Parameter <fac> "CS" Keyboard lock "PS" Phone locked to SIM (device code) "SC" SIM card (PIN) "FD" FDN lock "AO" BAOC (bar all outgoing calls) "OI" BOIC (bar outgoing international calls) "OX" BOIC-exHC (bar outgoing international calls except to home country) "AI" BAIC (bar all incoming calls) "IR" BIC-Roam (bar incoming calls when roaming outside the home country) "AB" All Barring services "AG" All outgoing barring services "AC" All incoming barring services
Write command AT+CLCK=<fac>, <mode>[,<passwd>[,<class>]]	Parameter <fac> See Test command <mode> 0 Cancels block 1 Activates block 2 Queries block status <passwd> Password <class> 1 Voice 2 Data 4 Fax 7 All classes (default value) Response If <mode>=2 and command is successful +CLCK: <status>[,<class1>[<CR><LF> +CLCK: <status>, class2....]] Parameter <status> 0 On 1 Off OK/ERROR/+CME ERROR

AT+CPWD	Change password to a block
Test command AT+CPWD=?	Response +CPWD: list of supported (<fac>, <pwdlength>)s OK/ERROR/+CME ERROR Parameter <fac> "P2" PIN2 otherwise See Test command for AT+CLCK command, without "FD" <pwdlength> Password length
Write command AT+CPWD=<fac>, <oldpwd>, <newpwd>	Parameter <fac> See Test command for AT+CLCK command <oldpwd>, <newpwd> Old and new password Response OK/ERROR/+CME ERROR

AT+CLIP	Display telephone number of calling party
Test command AT+CLIP=?	Response +CLIP: (list of supported <n>s) OK/ERROR/+CME ERROR Parameter <n> 0 Suppresses the unexpected messages 1 Displays the unexpected messages
Read command AT+CLIP?	Response +CLIP: <n>, <m> OK/ERROR/+CME ERROR Parameter <n> See Test command <m> 0 CLIP not booked 1 CLIP booked 2 Unknown
Write command AT+CLIP=<n>	Parameter <n> See Read command Response OK/ERROR/+CME ERROR
	Unexpected message +CLIP: <num>,<type> Telephone number of caller

AT+CCFC	Call forwarding
Test command AT+CCFC=?	Response +CCFC: (list of supported <reas>s) OK/ERROR/+CME ERROR Parameter <reas> 0 Always 1 If busy 2 If no answer 3 If not available 4 All reasons (0-3) 5 All conditional reasons (1-3)
Write command AT+CCFC=<reas>, <mode>[, <num>[, <type>[,<class> [,,,<time>]]]]	Parameter <reas> See Test command <mode> 0 Deactivate 1 Activate 2 Query 3 Install 4 Delete <num> Telephone number <type> Type of telephone number <class> 1 Voice 2 Data 4 Fax 7 All classes <time> 1-30 Time, rounded to a multiple of five seconds Response If <mode>=2 and command is successful +CCFC: <status>, <class1>[, <num>, <type>[,,, <time>]][<CR><LF>+CCFC: ....] OK/ERROR/+CME ERROR Parameter <status> 0 Not active 1 Active

AT+CHLD	Call hold and multiparty
Test command AT+CHLD=?	Response [+CHLD: (list of supported <n>s)] OK/ERROR/+CME ERROR
Write command AT+CHLD=[<n>]	Parameter <n> 0 Terminates all held calls or sets UDUB (User Determined User Busy) for a waiting call 1 Terminates all active calls (if there are any) and accepts the other call (waiting call or held call) 1X Terminates call number X (X= 1-7) 2 Puts all active calls on hold (if there are any) and accepts the other call (waiting call or held call) as active 2X Puts all active calls except call X (X= 1-7) on hold 3 Connects the call put on hold to the active call For terminating Terminating all calls except waiting calls is done with "AT+CHUP" Note: Command scope depends on the SIM clearing and/or on the network support Response OK/ERROR/+CME ERROR

AT+CPAS	Query the telephone status
Test command AT+CPAS=?	Response +CPAS: (list of supported <pas>s) OK/ERROR/+CME ERROR Parameter <pas> 0 Ready 3 Incoming call (phone is ringing) 4 Call is active
Execute command AT+CPAS	Response +CPAS: <pas> OK/ERROR/+CME ERROR Parameter <pas> See Test command OK/ERROR/+CME ERROR

AT+CPIN	Enter PIN and query block
Test command AT+CPIN=?	Response OK
Read command AT+CPIN?	Response +CPIN: <code> OK/ERROR/+CME ERROR Parameter <code> READY — No further input necessary SIM PIN — SIM PIN input necessary SIM PUK — SIM PUK input necessary PH-SIM PIN — Device-code (theft protection) input necessary PH-SIM PUK — Device-code PUK (theft protection) input necessary SIM PIN2 — PIN2, e.g. for editing the FDN book; only possible if previous command was acknowledged with +CME ERROR:17 SIM PUK2 — Only possible if previous command was acknowledged with error +CME ERROR:18 The required error message can (must) be provoked by an attempted Write command.
Write command AT+CPIN=<pin>[, <new pin>	Parameter <pin> — Password for appropriate block; if the block is a PUK, then a <new pin> is necessary. <new pin> — New password for the block Response OK/ERROR/+CME ERROR

AT+CBC	Battery charge
Test command AT+CBC=?	Response +CBC: (list of supported <bcs>s),(list of supported <bcl>s) OK/ERROR/+CME ERROR Parameter <bcs> 0 — ME is supplied from battery 1 — ME has battery but is not supplied from there 2 — ME has no battery connected 3 — Error <bcl> 0 — Battery is flat, but no more actions possible 1-100 charge in per cent
Execute command AT+CBC	Response +CBC: <bcs>,<bcl>

AT+CSQ	Output signal quality
Test command AT+CSQ=?	Response +CSQ: (list of supported <rssi>s), list of supported <ber>) OK/ERROR/+CME ERROR Parameter <rssi> Reception level: 0 -113 dBm or less 1 -111 dBm 2-30 -109 to -53 dBm 31 -51 dBm or more 99 Unknown <ber> Bit error rate: 0-7 Like RXQUAL values from Table GSM 05.08 in Section 8.2.4 99 Unknown
Execute command AT+CSQ	Response +CSQ: <rssi>, <ber> OK/ERROR/+CME ERROR Parameter <rssi> See Test command <ber> See Test command

AT+CPBS	Select a telephone book
Test command AT+CPBS=?	Response +CPBS: (list of supported <sto>s) OK/ERROR/+CME ERROR Parameter <sto> "FD" SIM fix-dialing phonebook "SM" SIM phonebook "ME" ME phonebook "DC" ME Dialled Calls List "ON" SIM (or ME) own numbers (MSISDNs) list "LD" SIM last-dialling phonebook "MC" ME missed (unanswered received) calls list "RC" ME received calls list ***For description of telephone-book features, see** *Appendix A* **Note:** "DC" and "LD" are never both available.
Read command AT+CPBS?	Response +CPBS: <sto> OK/ERROR/+CME ERROR Parameter <sto> See Test command
Write command AT+CPBS=<sto>	Parameter <sto> See Test command Response OK/ERROR/+CME ERROR

AT+CPBR	Read a telephone-book entry
Test command AT+CPBR=?	Response +CPBR: (list of supported <index>s), <nlength>, <tlength> OK/ERROR/+CME ERROR Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of text corresponding to the number
Write command AT+CPBR=<index1>[, <index2>]	Response +CPBR: <index1>, <nummer>, <typ>, <text>[<CR><LF> +CPBR: ..... +CPBR: <index2>, <nummer>, <typ>, <text>] OK/ERROR/+CME ERROR Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <nummer> Telephone number <typ> Type of number <text> Text corresponding to the telephone number **NOTE:** In models before the S25, empty phonebook records are reported as follows: **+CPBR: <index1>,empty** In S25ff, those empty entries don't produce any output.

AT+CPBW	Write a telephone-book entry
Test command AT+CPBW=?	Response +CPBW: (list of supported <index>s), <nlength>, <tlength> OK/ERROR/+CME ERROR Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of text corresponding to the number
Write command AT+CPBW=[<index>], [<nummer>, [<typ>, [<text>]]]	Parameter <index> Location number at which the entry is written <nummer> Telephone number <typ> Type of number <text> Text corresponding to the telephone number Response OK/ERROR/+CME ERROR Note: The following characters in `<text>` must be entered via the escape sequence: GSM char. / Seq. / Seq.(hex) / Note \ / \5C / 5C 35 43 / (backslash) “ / \22 / 5C 32 32 / (string delimiter) BSP / \08 / 5C 30 38 / (backspace) NULL / \00 / 5C 30 30 / (GSM null) '0' (GSM null) may cause problems on application level when using the function strlen() and should thus be represented by an escape sequence when necessary

AT+CMEE	Expanded error messages according to GSM 07.07
Test command AT+CMEE=?	Response +CMEE: (list of supported <n>s) Parameter <n> 0 Suppresses the expanded error format 1 Expanded error messages as number 2 Expanded error messages as text
Read command AT+CMEE?	Response +CMEE: <n> Parameter <n> See Read command
Write command AT+CMEE=<n>	Parameter <n> See Read command Response OK/ERROR/+CME ERROR
	Description: **The following CME errors are possible:** 0 PHONE FAILURE 1 NO CONNECTION TO PHONE 2 PH-TA LINK RESERVED 3 OPERATION NOT ALLOWED 4 OPERATION NOT SUPPORT 5 PH-SIM PIN REQUIRED 10 SIM NOT INSERTED 11 SIM PIN REQUIRED 12 SIM PUK REQUIRED 13 SIM FAILURE 14 SIM BUSY 15 SIM WRONG 16 INCORRECT PASSWORD 17 SIM PIN2 REQUIRED 18 SIM PUK2 REQUIRED 20 MEMORY FULL 21 INVALID INDEX 22 NOT FOUND 23 MEMORY FAILURE 24 TEXT TOO LONG 25 INV CHAR IN TEXT 26 DIAL STRING TOO LONG 27 INV CHAR IN DIAL 30 NO NETWORK SERVICE 31 NETWORK TIMEOUT 100 UNKNOWN 512 CALL BARRED BY BLACKLIST 513 PHONE LINK RESERVED 514 INVALID DIAL STRING 515 PHONE BUSY 550 PH-SIM PUK REQUIRED 551 NTF-SIM PIN REQUIRED

	552 NTF-SIM PUK REQUIRED 553 PH-NET PIN REQUIRED 554 PH-NET PUK REQUIRED 555 PH-SP PIN REQUIRED 556 PH-SP PUK REQUIRED 557 PH-CP PIN REQUIRED 558 PH-CP PUK REQUIRED 559 FEATURE PIN REQUIRED 560 FEATURE PUK REQUIRED **The following CMS errors have been defined for SMS:** 300 ME FAILURE 301 SMS SERVICE OF ME RESERVED 302 OPERATION NOT ALLOWED 303 OPERATION NOT SUPPORTED 304 INVALID PDU PARAMETER 305 INVALID TEXT MODE 310 SIM NOT INSERTED 311 SIM PIN NECESSARY 312 PH-SIM PIN NECESSARY 313 SIM FAILURE 314 SIM BUSY 315 SIM WRONG 320 MEMORY FAILURE 321 INVALID MEMORY FAILURE 322 MEMORY FULL 330 SMSC ADDRESS UNKNOWN 331 NO NETWORK SERVICE 332 NETWORK TIMEOUT 340 NO +CNMA ACK EXPECTED 500 UNKNOWN ERROR

AT+VTS	Send a DTMF tone
Test command AT+VTS=?	Response +VTS: <dtmf>,<duration> OK/ERROR/+CME ERROR Parameter <dtmf> 0-9,#,*,A-D, exactly one character <duration> Duration of tone in (duration/10) seconds
Write command AT+VTS= <dtmf> [,<duration>] or AT+VTS= <dtmf-string>	Parameter <dtmf> **One** character from the list, see Test command<duration> See Test command <dtmf-string> max. 29 characters in quotation marks ("..."), then a duration cannot be specified Response OK/ERROR/+CME ERROR **Important**: There is a leading output prefix +VTS in models before the S25.

AT+VTD	Set duration of a DTMF tone
Test command AT+VTD=?	Response +VTD: (list of supported <duration>s) OK/ERROR/+CME ERROR Parameter <duration> 1-255 Duration of tone in (duration/10) seconds
Read command AT+VTD?	Response +VTD: <duration> OK/ERROR/+CME ERROR
Write command AT+VTD=<duration>	Parameter <duration> See Test command Response OK/ERROR **Important**: There is a leading output prefix +VTD in models before the S25.

AT+WS46	Select wireless network
Test command AT+WS46=?	Response +WS46: (list of supported <n>s) OK
Read command AT+WS46?	Response +WS46: <n> OK/ERROR/+CME ERROR Parameter <n> Integer; WDS side stack 12 GSM digital cellular
Write command AT+WS46=[<n>]	Response OK/ERROR/+CME ERROR **Important**: There is a leading output prefix +WS46 in models before the S25.

AT+CSCS	Select TE character set
Test command AT+CSCS=?	Response +CSCS: (list of supported <chset>s) OK
Read command AT+CSCS?	Response +CSCS: <chset> OK/ERROR/+CME ERROR Parameter <chset> String; determines which TE character set is used
Write command AT+CSCS=[<chset>]	Response OK/ERROR/+CME ERROR

AT+CAOC	Advice of charge
Test command AT+CAOC=?	Response OK
Execute command AT+CAOC	Response +CAOC: <ccm> OK/ERROR/+CME ERROR Parameter <ccm> Updated hexadecimal call meter, measured in home units; coding analogous to ACMmax on the SIM

AT+CSSN	Supplementary service notifications **Revision according to GSM 07.07 Version 5.0.0**
Test command AT+CSSN=?	Response +CSSN: (list of supported <n>s), (list of supported <m>s) Parameter <n> 0 Suppresses the +CSSI messages 1 Activates the +CSSI messages <m> 0 Suppresses the +CSSU messages 1 Activates the +CSSU messages For supported +CSSI/+CSSU messages, see also **Fehler! Verweisquelle konnte nicht gefunden werden.**
Read command AT+CSSN?	Response +CSSN: <n>,<m> Parameter <n> See Test command <m> See Test command
Write command AT+CSSN=<n>[,<m>]	Parameter <n> See Read command <m> See Read command
	Unexpected message +CSSI: <code1> +CSSU: <code2> Parameter <code1> Intermediate result code 3 Waiting call is pending <code2> Unsolicited result code 5 Held call was terminated

AT+CRSM	Restricted SIM access
Test command AT+CRSM=?	Response OK
Write command +CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>]]]	Response +CRSM: <sw1>,<sw2>[,<response>] OK/ERROR/+CME ERROR Parameter <command>:178 READ RECORD 192 GET RESPONSE 214 UPDATE BINARY 220 UPDATE RECORD 242 STATUS <fileid>: Integer, identifier of the data file on the SIM, mandatory for every command except STATUS (see GSM 11.11) <P1>, <P2>, <P3>: Integer, transferal parameter from ME to SIM, mandatory for every command except GET RESPONSE,STATUS (see GSM 11.11) <data>: Hexadecimal string; information that is to be written to the SIM <sw1>, <sw2>: Integer; information from the SIM as to how/whether the command was executed <response>: Hexadecimal string; given when a command was successfully processed **Note:** The write access to CK boxes receives only limited support and differs from device to device.

AT+CIMI	Output of IMSI
Test command AT+CIMI=?	Response OK
Execute command AT+CIMI	Response <imsi> Parameter <imsi> International Mobile Subscriber Identity (IMSI)

AT+CACM	Accumulated call meter
Test command AT+CACM=?	Response OK
Read command AT+CACM?	Response +CACM: <acm> OK/ERROR/+CME ERROR Parameter <acm> Accumulated call meter in hexadecimal format, measured in home units; coding analogous to ACMmax on the SIM
Write command AT+CACM=[<passwd>]	Response OK/ERROR/+CME ERROR Parameter <passwd> String type; usually PIN2

AT+CAMM	Accumulated call meter maximum
Test command AT+CAMM=?	Response OK
Read command AT+CAMM?	Response +CAMM: <acmmax> OK/ERROR/+CME ERROR Parameter <acmmax> Accumulated call meter maximum in hexadecimal format, measured in home units; coding analogous to ACMmax on the SIM
Write command AT+CAMM=[<acmmax>[,<passwd>]]	Response OK/ERROR/+CME ERROR Parameter <acmmax> (see Read command) <passwd> String type; usually PIN2

AT+CLCC	List Current Calls
Test command AT+CLCC=?	Response OK
Execute command AT+CLCC	Response [+CLCC: <id1>,<dir>,<stat>,<mode>,<mpty>, <number>,<type>] [<CR><LF>+CLCC: <id2>,<dir>,<stat>,<mode>,<mpty>, <number>,<type> [...]]] OK/ERROR/+CME ERROR Parameter <idx>: integer type; call identification number as described in GSM 02.30 [19] subclause 4.5.5.1; this number can be used in +CHLD command operations <dir>: 0 mobile originated (MO) call 1 mobile terminated (MT) call <stat> (state of the call): 0 active 1 held 2 dialing (MO call) 3 alerting (MO call) 4 incoming (MT call) 5 waiting (MT call) <mode> (bearer/teleservice): 0 voice 1 data 2 fax 3 voice followed by data, voice mode 4 alternating voice/data, voice mode 5 alternating voice/fax, voice mode 6 voice followed by data, data mode 7 alternating voice/data, data mode 8 alternating voice/fax, fax mode 9 unknown <mpty>: 0 call is not one of multiparty (conference) call parties 1 call is one of multiparty (conference) call parties <number>: string type phone number in format specified by <type> <type>: type of address octet in integer format

AT+CCLK	Clock
Test command AT+CCLK=?	Response OK
Write command AT+CCLK=<time>	Response OK/ERROR/+CME ERROR Parameter: <time> see Test commnd

AT+COPN	Read operator names
Test command AT+COPN=?	Response OK
Execute command AT+COPN	Response +COPN:numeric <oper>,long alphanumeric <oper><CR><LF> +COPN:..... OK/ERROR/+CME ERROR Parameter <oper> Network operator in numeric and alphanumeric notation

1.3.2. AT Commands According to GSM 07.05 for SMS

The GSM 07.05 commands are used for operating the SMS functions of the GSM mobile phone. The GSM module MOBILE supports the SMS PDU mode.

AT+CSMS	Selection of message service **Revision according to GSM 07.05 Version 5.0.0**
Test command AT+CSMS=?	Response +CSMS: (list of supported <service>s) Parameter <service> 0 GSM 3.40 and 3.41 1 GSM 3.40 and 3.41 and compatibility of the AT command syntax for phase 2+ *(**NOTE:** Deactivating the phase 2+ compatibility is only possible if the direct output of short messages +CNMI=2,2 or +CNMI=2,3 is not activated. If necessary, the latter should be deactivated first).*
Read command AT+CSMS?	Response +CSMS: <service>,<mt>,<mo>,<bm> Parameter <service> 0 GSM 3.40 and 3.41 <mt> Mobile terminated messages 1 Type supported <mo> Mobile originated messages 1 Type supported <bm> Broadcast type messages 0 Type not supported
Write command AT+CSMS= <service>	Parameter <service> 0 GSM 3.40 and 3.41 Response +CSMS: <mt>,<mo>,<bm> OK/ERROR/+CMS ERROR

AT+CPMS	Selection of SMS memory **Revision according to GSM 07.05 Version 4.7.0**
Test command AT+CPMS=?	Response +CPMS: (list of supported <mem1>s),(list of supported <mem2>s) ,(list of supported <mem3>s) Parameter <mem1> Memory from which messages are read and deleted "SM" SIM-messages memory <mem2> Memory to which messages are written and sent "SM" SIM-messages memory <mem3> Memory in which received messages are stored, if forwarding to the PC is not set ("+CNMI") "SM" SIM-messages memory
Read command AT+CPMS?	Response +CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,<total2>,<mem3>,<used3>,<total3> Parameter <memx> Memory from which messages are read and deleted <usedx> Number of messages currently in <memx> <totalx> Number of storable messages in <memx>
Write command AT+CPMS=<mem1>[,<mem2>[,<mem3>]]	Parameter <mem1> See Test command <mem2> See Test command <mem3> See Test command Response +CPMS: <used1>,<total1>,<used2>,<total3>,<used3>,<total3> OK/ERROR/+CMS ERROR

AT+CMGF	SMS format
Test command AT+CMGF=?	Response +CMGF: (list of supported <mode>s) Parameter <mode>: 0 PDU mode
Read command AT+CMGF?	Response +CMGF: <mode> Parameter <mode>: 0 PDU mode
Write command AT+CMGF=[<mode>]	Parameter <mode>: 0 PDU mode Response OK/ERROR

AT+CSCA	Address of the SMS service center
Test command AT+CSCA=?	Response OK
Read command AT+CSCA?	Response +CSCA: <sca>,<tosca> Parameter <sca> Service-center address in string format <tosca> Service-center address format
Write command AT+CSCA=<sca>[,<tosca>]	Parameter <sca> Service-center address in string format <tosca> Service-center address format Response OK/ERROR

AT+CNMI	Display new incoming SMS **Revision according to GSM 07.05 Version 4.7.0**
Test command AT+CNMI=?	Response +CNMI: (list of supported <mode>s),(list of supported <mt>s),(list of supported <bm>s),(list of supported <ds>s),(list of supported <bfr>s) Parameter <mode> 0 Buffers unexpected messages (but is equivalent to rejecting; see <bfr>) 2 Buffers unexpected messages if serial interface is occupied, otherwise they are output <mt> 0 Suppresses unexpected messages for incoming short messages 1 Unexpected messages of a received short message (SMS-DELIVER) that is stored on a chip card are output in the form **+CMTI: <mem>,<index>** 2 Unexpected messages of a received short message (SMS-DELIVER) (except class 2 and the message "Waiting Indication Group: *store message*") are output in the form **+CMT: [<alpha>],<length><CR><LF><pdu>** (<alpha> is not supported) Class 2 and the message "Waiting Indication Group: *store message*" are output as <mt>=1 3 Unexpected messages of a received short message (SMS-DELIVER) class 3 are output as <mt>=2. Messages with other data coding schemes are output as <mt>=1. (***NOTE:*** *<mt>=2 and <mt>=3 are not possible unless the Phase 2+ compatibility has been activated by means of +CSMS=1*)

	<bm>	0	Suppresses unexpected messages for incoming cell broadcast messages
		2	Outputs unexpected messages for cell broadcast messages in the form **+CBM: <length><CR><LF><pdu>**
	<ds>	0	Suppresses unexpected messages for incoming SMS status reports
		2	Outputs unexpected messages for SMS status reports in the form **+CDS: <length><CR><LF><pdu>**
	<bfr>	1	Buffered unexpected messages are rejected when switching from <mode> 0 to <mode> 2.
	<mem>		See +CPMS
	<index>		Index of the record on the chip card
	<alpha>		alphanumeric representation of the sender address
	<length>		Length of <pdu>
	<pdu>		See +CMGL
Read command AT+CNMI?	Response +CNMI: <mode>,<mt>,<bm>,<ds>,<bfr> Parameter <mode> See Test command <mt> See Test command <bm> See Test command <ds> See Test command <bfr> See Test command		
Write command AT+CNMI=[<mode>[,<mt>[,<bm>[,<ds>[,<bfr>]]]]]	Parameter <mode> See Test command <mt> See Test command <bm> See Test command <ds> See Test command <bfr> See Test command Response OK/ERROR/+CMS ERROR		
	Unexpected message +CMTI: <mem>,<index>		Indication that new message has arrived
	+CMT: ,<length><CR><LF><pdu>		Direct output of the short message
	+CDS: <length><CR><LF><pdu>		Direct output of the status report
	+CBM: <length><CR><LF><pdu>		Direct output of the cell broadcast message

AT+CNMA	Acknowledgment of a short message directly output (without storing on the chip card) **Revision according to GSM 07.05 Version 5.0.0** ***(NOTE:** This command is not possible unless the Phase 2+ compatibility has been activated by means of +CSMS=1)*
Test command AT+CNMA=?	Response +CNMA: (list of supported <n>s) Parameter <n> 0 Mode of functioning analogous to GSM 07.05 text mode
Write command AT+CNMA[=<n>]	Parameter <n> See Test command Response OK/ERROR/+CMS ERROR: <err>

AT+CMGL	List SMS **Revision according to GSM 07.05 Version 4.7.0**
Test command AT+CMGL=?	Response +CMGL: (list of supported <stat>s) Parameter <stat> 0 "REC UNREAD": received unread messages (default) 1 "REC READ": received read messages 2 "STO UNSENT": stored unsent messages 3 "STO SENT": stored sent messages 4 "ALL": all messages
Write command AT+CMGL[=<stat>]	Parameter <stat> See Test command Response **If PDU mode (+CMGF=0) and command are successful:** +CMGL: <index>,<stat>,[<alpha>],<length><CR><LF><pdu> [<CR><LF>+CMGL: <index>,<stat>,[alpha],<length><CR><LF><pdu> [...]] <pdu> The PDU begins with the service-center address (according to GSM04.11), followed by the TPDU according to GSM03.40 in hexadecimal format **otherwise:** +CMS ERROR: <err>

AT+CMGR	Read in an SMS **Revision according to GSM 07.05 Version 4.7.0**
Test command AT+CMGR=?	Response OK
Write command AT+CMGR=<index>	Parameter <index> Index of message in selected memory <mem1> Response **If PDU mode (+CMGF=0) and command are successful:** +CMGR: <stat>,[<alpha>],<length><CR><LF><pdu> <pdu> Siehe "AT+CMGL" **otherwise:** +CMS ERROR: <err>

AT+CMGS	Send an SMS
Test command AT+CMGS=?	Response OK
Write command **If PDU mode (+CMGF=0)** +CMGS=<length><CR>***PDU is given*** <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response **If sending is successful:** +CMGS: <mr> **If sending is not successful:** +CMS ERROR: <err>

AT+CMSS	Send an SMS from the SMS memory
Test command AT+CMSS=?	Response OK
Write command +CMSS=<index>[,<da>[,<toda>]]	Parameter <index> Index of message in selected memory <mem1> <da> Destination address in string format <toda> Format of destination address <mr> Message reference Response **If sending is successful:** +CMSS: <mr> **If sending is not successful:** +CMS ERROR: <err>

AT+CMGW	Write an SMS to the SMS memory
Test command AT+CMGW=?	Response OK
Write command **If PDU mode (+CMGF=0)** AT+CMGW=<length>[,<stat>]<CR>***PDU is given*** <ctrl-Z/ESC>	Parameter <length> Length of PDU <stat> See command +CMGL <pdu> See "AT+CMGL" <index> Index of message in selected memory <mem1> Response +CMGW: <index> +CMS ERROR: <err>

AT+CMGD	Delete an SMS in the SMS memory
Test command At+CMGD=?	Response OK
Write command AT+CMGD=<index>	Parameter <index> Index of message in the selected memory <mem1> Response OK/ERROR/+CMS ERROR

AT+CSCB	Select cell broadcast messages
Test command AT+CSCB=?	Response +CSCB: (list of supported <mode>s) Parameter <mode> 0 Accepts messages that are defined in <mids> and <dcss> 1 Does not accept messages that are defined in <mids> and <dcss>
Read command AT+CSCB? Write command AT+CSCB=[<mode>[,<mids>[,<dcss>]]]	Response +CSCB: <mode>,<mids>,<dcss> Parameter <mode> See Test command <mids> String type; combinations of CBM message Ids <dcss> String type; combinations of CBM data coding schemes

AT+CMGC	Send an SMS command
Test command AT+CMGC=?	Response OK
Write command **If PDU mode (+CMGF=0)** +CMGC=<length><CR>***PDU is given*** <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response **If sending is successful:** +CMGC: <mr> **If sending is not successful:** +CMS ERROR: <err>

Appendix A

Features of the Telephone-Book Memory

Name	Description	Category / Access	Write	Delete completely
FD	Fix-dialing number (SIM fix-dialing telephone book)	GSM 07.07 / +CPBS	Allowed (PIN2 required)	
SM	Abbreviate dialing number (SIM telephone book)	GSM 07.07 / +CPBS	Allowed (device code required if FDN replacement is active)	
DC (MD)	Mobile last dialing number (last number redial memory; only if "LD" is not available)	GSM 07.07 / +CPBS	Not allowed	
ON (OW)	Own Numbers (SIM own telephone numbers)	GSM 07.07 (Siemens) / +CPBS (historical)	Allowed	
LD	SIM last dialing number (last number redial memory on SIM)	GSM 07.07 / +CPBS	Not allowed	
ME	Mobile-equipment telephone book (ME dialing numbers)	GSM 07.07 / +CPBS	Allowed (device code required if FDN replacement is active)	
MC (MS)	Missed dialing numbers (unanswered calls)	GSM 07.07 (Siemens) / +CPBS	Not allowed	
RC (CD)	Callback dialing numbers (answered calls)	GSM 07.07 (Siemens) / +CPBS	Not allowed	

Writing to the FDN Phonebook / FDN Replacement

Writing to the fix-dialing number phonebook is protected by PIN2.
A Write sequence (to e.g. record 5) runs as follows:

```
AT+CMEE=2                    //Activate expanded error message
OK
```

```
AT+CPBS=?                          // Listing of available telephone books
+CPBS: "FD","SM","LD"
OK

AT+CPBS="FD"                       // Selection of the FDN telephone book
OK

AT+CPBW=5,1234,,"test"             // A Write to record 5 is attempted...
+CME ERROR: SIM PIN2 REQUIRED      // ... PIN2 is required for this purpose

AT+CPIN?                           // Query of the PIN status...
+CPIN: SIM PIN2                    // ... PIN2 is to be entered

AT+CPIN=12345678                   // Input of PIN2
OK

AT+CPBW=5,1234,,"test"             // A Write to record 5 is attempted...
OK                                 // PIN2 remains active as long as you use the commands
                                   // RCCL3_CMD_CPIN, RCCL3_CMD_CPBS,
                                   // RCCL3_CMD_CPBR, RCCL3_CMD_CPBW,
                                   // RCCL3_CMD_SPIC.
                                   // If you use other commands or if none of the
                                   // above commands are executed within five
                                   // minutes, the validity of PIN2 is voided.

AT+CPBW=6,5678,,"new test"         // A Write to record 6 is attempted...
OK

...
```

In addition, if there is no FDN phonebook available on the SIM, it is possible to activate a feature which activates FDN-like behavior for the "SM" and "ME" phonebooks (FDN replacement). (Currently this feature can only be activated via the MMI block/device block/excluding telephone book.)
In this case, the Write to the "SM" and "ME" phonebooks is ensured by the device code (PH-SIM PIN and PH-SIM PUK, respectively).
The sequence for entering the device code is analogous to the above example.

DALLAS SEMICONDUCTOR

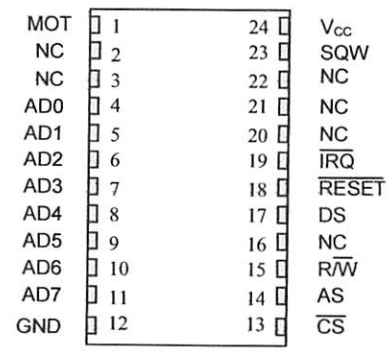
DS12C887
Real Time Clock

w.dalsemi.com

ATURES

- Drop-in replacement for IBM AT computer clock/calendar
- Pin compatible with the MC146818B and DS1287
- Totally nonvolatile with over 10 years of operation in the absence of power
- Self-contained subsystem includes lithium, quartz, and support circuitry.
- Counts seconds, minutes, hours, days, day of the week, date, month, and year with leap year compensation valid up to 2100
- Binary or BCD representation of time, calendar, and alarm
- 12– or 24–hour clock with AM and PM in 12–hour mode
- Daylight Savings Time option
- Selectable between Motorola and Intel bus timing
- Multiplex bus for pin efficiency
- Interfaced with software as 128 RAM locations
  - 15 bytes of clock and control registers
  - 113 bytes of general purpose RAM
- Programmable square wave output signal
- Bus–compatible interrupt signals (IRQ)
- Three interrupts are separately software maskable and testable
  - Time–of–day alarm once/second to once/day
  - Periodic rates from 122 ms to 500 ms
  - End of clock update cycle
- Century register

PIN ASSIGNMENT



DS12C887 24-Pin
ENCAPSULATED PACKAGE

PIN DESCRIPTION

AD0-AD7	-	Multiplexed Address/Data Bus
NC	-	No Connect
MOT	-	Bus Type Selection
$\overline{CS}$	-	RTC Chip Select Input
AS	-	Address Strobe
R/$\overline{W}$	-	Read/Write Input
DS	-	Data Strobe
$\overline{RESET}$	-	Reset Input
$\overline{IRQ}$	-	Interrupt Request Output
SQW	-	Square Wave Output
V_{CC}	-	+5 Volt Main Supply
GND	-	Ground

SCRIPTION

DS12C887 Real Time Clock plus RAM is designed as a direct upgrade replacement for the DS12887 isting IBM compatible personal computers to add hardware year 2000 compliance. A century byte added to memory location 50, 32h, as called out by the PC AT specification. A lithium energy e, quartz crystal, and write-protection circuitry are contained within a 24–pin dual in-line package. ich, the DS12C887 is a complete subsystem replacing 16 components in a typical application. The ions include a nonvolatile time-of-day clock, an alarm, a one-hundred-year calendar, programmable rupt, square wave generator, and 113 bytes of nonvolatile static RAM. The real time clock is ctive in that time-of-day and memory are maintained even in the absence of power.

'ERATION

: block diagram in Figure 1 shows the pin connections with the major internal functions of the 12C887. The following paragraphs describe the function of each pin.

;NAL DESCRIPTIONS

D, V_{CC} - DC power is provided to the device on these pins. V_{CC} is the +5 volt input. When 5 volts are lied within normal limits, the device is fully accessible and data can be written and read. When V_{CC} is ›w 4.25 volts typical, reads and writes are inhibited. However, the timekeeping function continues ffected by the lower input voltage. As V_{CC} falls below 3 volts typical, the RAM and timekeeper are :ched over to an internal lithium energy source. The timekeeping function maintains an accuracy of ±1 ute per month at 25°C regardless of the voltage input on the V_{CC} pin.

'T (Mode Select) – The MOT pin offers the flexibility to choose between two bus types. When ıected to V_{CC}, Motorola bus timing is selected. When connected to GND or left disconnected, Intel timing is selected. The pin has an internal pull-down resistance of approximately 20KΩ.

W (Square Wave Output) – The SQW pin can output a signal from one of 13 taps provided by the nternal divider stages of the Real Time Clock. The frequency of the SQW pin can be changed by ;ramming Register A as shown in Table 1. The SQW signal can be turned on and off using the SQWE n Register B. The SQW signal is not available when V_{CC} is less than 4.25 volts typical.

)-AD7 (Multiplexed Bidirectional Address/Data Bus) – Multiplexed buses save pins because ess information and data information time share the same signal paths. The addresses are present ıg the first portion of the bus cycle and the same pins and signal paths are used for data in the second ion of the cycle. Address/data multiplexing does not slow the access time of the DS12C887 since the change from address to data occurs during the internal RAM access time. Addresses must be valid · to the falling edge of AS/ALE, at which time the DS12C887 latches the address from AD0 to AD6. d write data must be present and held stable during the latter portion of the DS or WR pulses. In a cycle the DS12C887 outputs 8 bits of data during the latter portion of the DS or RD pulses. The read : is terminated and the bus returns to a high impedance state as DS transitions low in the case of ›rola timing or as RD transitions high in the case of Intel timing.

Address Strobe Input) – A positive going address strobe pulse serves to demultiplex the bus. The ıg edge of AS/ALE causes the address to be latched within the DS12C887. The next rising edge that rs on the AS bus will clear the address regardless of whether $\overline{CS}$ is asserted. Access commands ld be sent in pairs.

Data Strobe or Read Input) – The DS/RD pin has two modes of operation depending on the level e MOT pin. When the MOT pin is connected to V_{CC}, Motorola bus timing is selected. In this mode s a positive pulse during the latter portion of the bus cycle and is called Data Strobe. During read s, DS signifies the time that the DS12C887 is to drive the bidirectional bus. In write cycles the ıg edge of DS causes the DS12C887 to latch the written data. When the MOT pin is connected to ', Intel bus timing is selected. In this mode the DS pin is called Read(RD). RD identifies the time d when the DS12C887 drives the bus with read data. The RD signal is the same definition as the ut Enable (OE) signal on a typical memory.

$\overline{W}$ **(Read/Write Input)** – The R/$\overline{W}$ pin also has two modes of operation. When the MOT pin is nected to V_{CC} for Motorola timing, R/W is at a level which indicates whether the current cycle is a l or write. A read cycle is indicated with a high level on R/W while DS is high. A write cycle is cated when R/$\overline{W}$ is low during DS. When the MOT pin is connected to GND for Intel timing, the $\overline{W}$ signal is an active low signal called WR. In this mode the R/$\overline{W}$ pin has the same meaning as the te Enable signal (WE) on generic RAMs.

(Chip Select Input) – The Chip Select signal must be asserted low for a bus cycle in the DS12C887 e accessed. $\overline{CS}$ must be kept in the active state during DS and AS for Motorola timing and during RD WR for Intel timing. Bus cycles which take place without asserting $\overline{CS}$ will latch addresses but no ess will occur. When V_{CC} is below 4.25 volts, the DS12C887 internally inhibits access cycles by rnally disabling the $\overline{CS}$ input. This action protects both the real time clock data and RAM data during er outages.

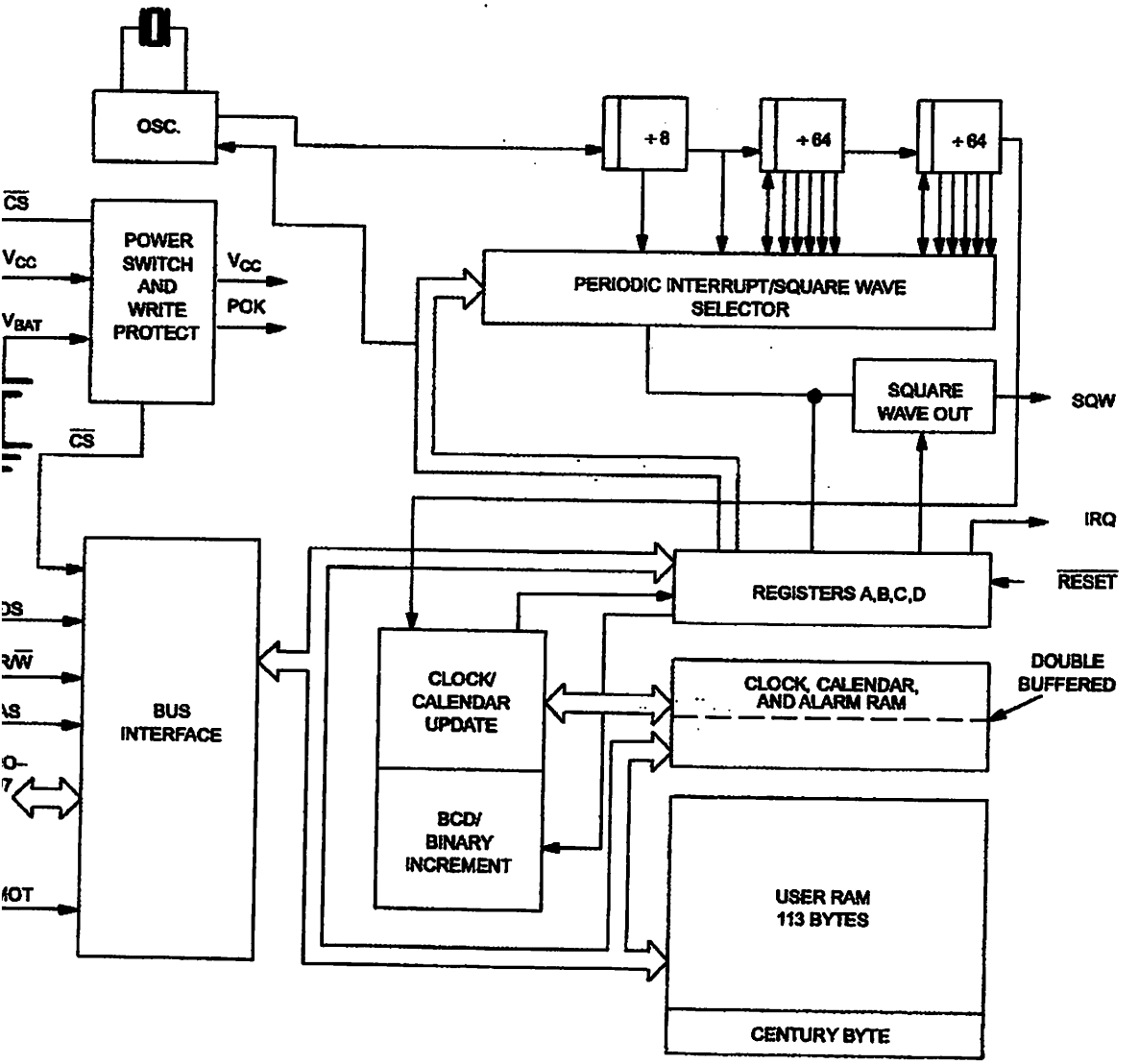
(Interrupt Request Output) - The $\overline{IRQ}$ pin is an active low output of the DS12C887 that can be l as an interrupt input to a processor. The $\overline{IRQ}$ output remains low as long as the status bit causing the rupt is present and the corresponding interrupt-enable bit is set. To clear the $\overline{IRQ}$ pin the processor gram normally reads the C register. The RESET pin also clears pending interrupts. When no interrupt litions are present, the $\overline{IRQ}$ level is in the high impedance state. Multiple interrupting devices can be nected to an $\overline{IRQ}$ bus. The IRQ bus is an open drain output and requires an external pull-up resistor.

$\overline{ET}$ **(Reset Input)** – The $\overline{RESET}$ pin has no effect on the clock, calendar, or RAM. On power-up the $\overline{SET}$ pin can be held low for a time in order to allow the power supply to stabilize. The amount of that $\overline{RESET}$ is held low is dependent on the application. However, if $\overline{RESET}$ is used on power–up, ime $\overline{RESET}$ is low should exceed 200 ms to make sure that the internal timer that controls the 2C887 on power-up has timed out. When $\overline{RESET}$ is low and V_{CC} is above 4.25 volts, the following rs:

A. Periodic Interrupt Enable (PEI) bit is cleared to zero.
B. Alarm Interrupt Enable (AIE) bit is cleared to zero.
C. Update Ended Interrupt Flag (UF) bit is cleared to zero.
D. Interrupt Request Status Flag (IRQF) bit is cleared to zero.
E. Periodic Interrupt Flag (PF) bit is cleared to zero.
F. The device is not accessible until $\overline{RESET}$ is returned high.
G. Alarm Interrupt Flag (AF) bit is cleared to zero.
H. $\overline{IRQ}$ pin is in the high impedance state.
I. Square Wave Output Enable (SQWE) bit is cleared to zero.
J. Update Ended Interrupt Enable (UIE) is cleared to zero.

ypical application $\overline{RESET}$ can be connected to V_{CC}. This connection will allow the DS12C887 to and out of power fail without affecting any of the control registers.

12C887 BLOCK DIAGRAM Figure 1



WER-DOWN/POWER-UP CONSIDERATIONS

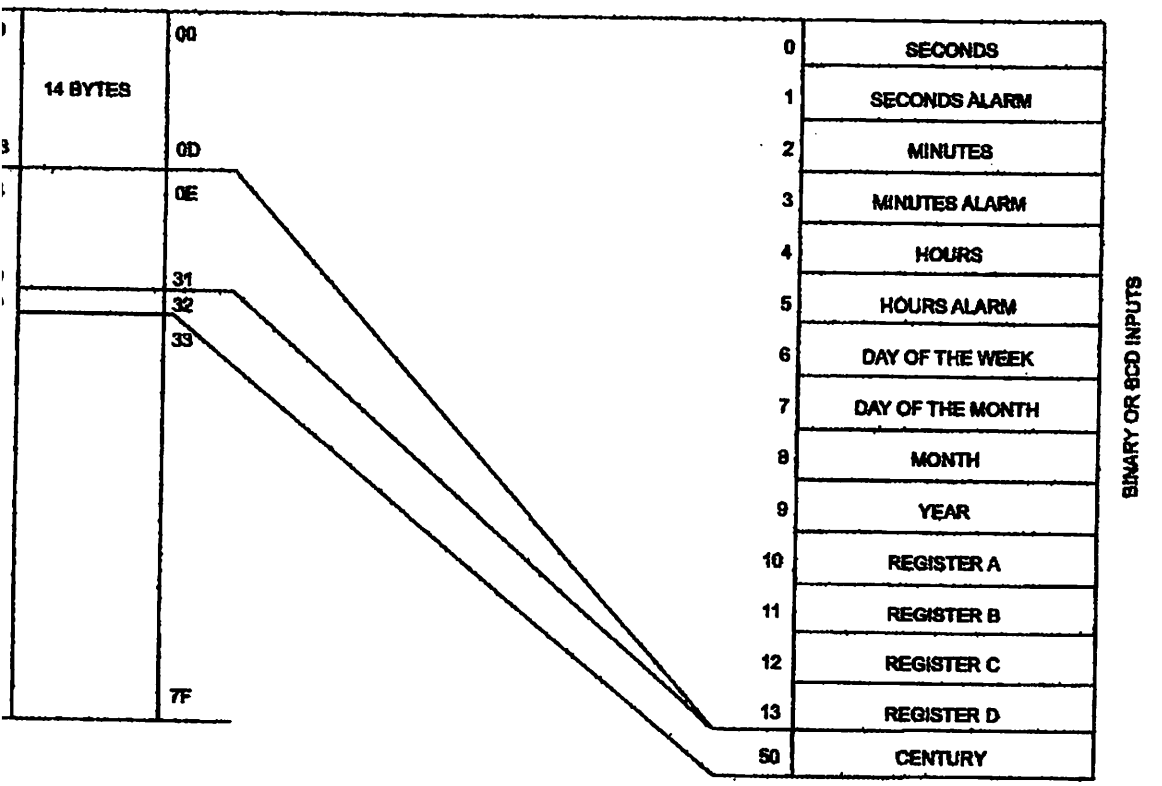
Real Time Clock function will continue to operate and all of the RAM, time, calendar, and alarm ıory locations remain nonvolatile regardless of the level of the V_{CC} input. When V_{CC} is applied to the 2C887 and reaches a level of greater than 4.25 volts, the device becomes accessible after 200 ms, ided that the oscillator is running and the oscillator countdown chain is not in reset (see Register A). time period allows the system to stabilize after power is applied. When V_{CC} falls below 4.25 volts, hip select input is internally forced to an inactive level regardless of the value of $\overline{CS}$ at the input pin. DS12C887 is, therefore, write-protected. When the DS12C887 is in a write-protected state, all inputs gnored and all outputs are in a high impedance state. When V_{CC} falls below a level of approximately lts, the external V_{CC} supply is switched off and an internal lithium energy source supplies power to :eal Time Clock and the RAM memory.

: ADDRESS MAP

address map for the DS12C885 is shown in Figure 2. The address map consists of 113 bytes of user I, 11 bytes of RAM that contain the RTC time, calendar, and alarm data, and 4 bytes which are used ontrol and status. All 128 bytes can be directly written or read except for the following:

1. Registers C and D are read-only.
2. Bit-7 of Register A is read-only.
3. The high order bit of the seconds byte is read-only.

2C887 REAL TIME CLOCK ADDRESS MAP Figure 2



IE, CALENDAR AND ALARM LOCATIONS

time and calendar information is obtained by reading the appropriate memory bytes. The time, ıdar, and alarm are set or initialized by writing the appropriate RAM bytes. The contents of the ten , calendar, and alarm bytes can be either Binary or Binary-Coded Decimal (BCD) format. Before ng the internal time, calendar, and alarm registers, the SET bit in Register B should be written to a : one to prevent updates from occurring while access is being attempted. In addition to writing the ten , calendar, and alarm registers in a selected format (binary or BCD), the data mode bit (DM) of ster B must be set to the appropriate logic level. All ten time, calendar, and alarm bytes must use the : data mode. The set bit in Register B should be cleared after the data mode bit has been written to v the real time clock to update the time and calendar bytes. Once initialized, the real time clock es all updates in the selected mode. The data mode cannot be changed without reinitializing the ten bytes. Table 2 shows the binary and BCD formats of the ten time, calendar, and alarm locations. The 2 bit cannot be changed without reinitializing the hour locations. When the 12–hour format is ted, the high order bit of the hours byte represents PM when it is a logic one. The time, calendar, ılarm bytes are always accessible because they are double buffered. Once per second the eleven bytes dvanced by one second and checked for an alarm condition. If a read of the time and calendar data rs during an update, a problem exists where seconds, minutes, hours, etc. may not correlate. The ıbility of reading incorrect time and calendar data is low. Several methods of avoiding any possible rect time and calendar reads are covered later in this text. The three alarm bytes can be used in two . First, when the alarm time is written in the appropriate hours, minutes, and seconds alarm ions, the alarm interrupt is initiated at the specified time each day if the alarm enable bit is high . The ıd use condition is to insert a "don't care" state in one or more of the three alarm bytes. The "don't code is any hexadecimal value from C0 to FF. The two most significant bits of each byte set the 't care" condition when at logic 1. An alarm will be generated each hour when the "don't care" bits et in the hours byte. Similarly, an alarm is generated every minute with "don't care" codes in the : and minute alarm bytes. The "don't care" codes in all three alarm bytes create an interrupt every ıd.

E, CALENDAR AND ALARM DATA MODES Table 1

ADDRESS LOCATION	FUNCTION	DECIMAL RANGE	RANGE	
			BINARY DATA MODE	BCD DATA MODE
O	Seconds	0-59	00-3B	00-59
1	Seconds Alarm	0-59	00-3B	00-59
2	Minutes	0-59	00-3B	00-59
3	Minutes Alarm	0-59	00-3B	00-59
4	Hours 12-hr, Mode	1-12	01-0C AM, 81-8C PM	01-12 AM, 81-92 PM
	Hours 24-hr, Mode	0-23	00-17	00-23
5	Hours Alarm 12-hr, Mode	1-12	01-0C AM, 81-8C PM	01-12 AM, 81-92 PM
	Hours Alarm 24-hr, Mode	0-23	00-17	00-23
6	Day of the week Sunday=1	1-7	01-07	01-07
7	Date of Month	1-31	01-1F	01-31
8	Month	1-12	01-0C	01-12
9	Year	0-99	00-63	00-99
50	Century	0-99	NA	19,20

NTROL REGISTERS

DS12C887 has four control registers which are accessible at all times, even during the update cycle.

GISTER A

3 LSB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
UIP	DV2	DV1	DV0	RS3	RS2	RS1	RS0

- The Update In Progress (UIP) bit is a status flag that can be monitored. When the UIP bit is a 1, the te transfer will soon occur. When UIP is a 0, the update transfer will not occur for at least 244µs. time, calendar, and alarm information in RAM is fully available for access when the UIP bit is 0. The bit is read-only and is not affected by $\overline{\text{RESET}}$. Writing the SET bit in Register B to a 1 inhibits any te transfer and clears the UIP status bit.

, DV1, DV0 - These three bits are used to turn the oscillator on or off and to reset the countdown ı. A pattern of 010 is the only combination of bits that will turn the oscillator on and allow the RTC ep time. A pattern of 11X will enable the oscillator but holds the countdown chain in reset. The next te will occur at 500 ms after a pattern of 010 is written to DV0, DV1, and DV2.

RS2, RS1, RS0 - These four rate-selection bits select one of the 13 taps on the 15-stage divider or le the divider output. The tap selected can be used to generate an output square wave (SQW pin) ır a periodic interrupt. The user can do one of the following:

1. Enable the interrupt with the PIE bit;
2. Enable the SQW output pin with the SQWE bit;
3. Enable both at the same time and the same rate; or
4. Enable neither

1 lists the periodic interrupt rates and the square wave frequencies that can be chosen with the RS These four read/write bits are not affected by $\overline{\text{RESET}}$.

GISTER B

B

							LSB
3IT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SET	PIE	AIE	UIE	SQWE	DM	24/12	DSE

' - When the SET bit is a 0, the update transfer functions normally by advancing the counts once per nd. When the SET bit is written to a 1, any update transfer is inhibited and the program can initialize ime and calendar bytes without an update occurring in the midst of initializing. Read cycles can be uted in a similar manner. SET is a read/write bit and is not affected by $\overline{\text{RESET}}$ or internal functions e DS12C887.

- The Periodic Interrupt Enable bit is a read/write bit which allows the Periodic Interrupt Flag (PF) ı Register C to drive the $\overline{\text{IRQ}}$ pin low. When the PIE bit is set to 1, periodic interrupts are generated riving the $\overline{\text{IRQ}}$ pin low at a rate specified by the RS3-RS0 bits of Register A. A 0 in the PIE bit cs the $\overline{\text{IRQ}}$ output from being driven by a periodic interrupt, but the Periodic Flag (PF) bit is still set e periodic rate. PIE is not modified by any internal DS12C887 functions but is cleared to 0 on $\overline{\text{ET}}$.

- The Alarm Interrupt Enable (AIE) bit is a read/write bit which, when set to a 1, permits the Alarm (AF) bit in register C to assert $\overline{\text{IRQ}}$. An alarm interrupt occurs for each second that the 3 time bytes l the 3 alarm bytes including a "don't care" alarm code of binary 11XXXXXX. When the AIE bit is ı 0, the AF bit does not initiate the $\overline{\text{IRQ}}$ signal. The internal functions of the DS12C887 not affect IE bit.

- The Update Ended Interrupt Enable (UIE) bit is a read/write bit that enables the Update End Flag bit in Register C to assert $\overline{\text{IRQ}}$. The $\overline{\text{RESET}}$ pin going low or the SET bit going high clears the ›it.

E - When the Square Wave Enable (SQWE) bit is set to a 1, a square wave signal at the frequency ' the rate-selection bits RS3 through RS0 is driven out on the SQW pin. When the SQWE bit is set he SQW pin is held low. SQWE is a read/write bit and is cleared by $\overline{\text{RESET}}$. SQWE is set to a 1 V_{CC} is powered up.

The Data Mode (DM) bit indicates whether time and calendar information is in binary or BCD t. The DM bit is set by the program to the appropriate format and can be read as required. This bit modified by internal functions or $\overline{\text{RESET}}$. A 1 in DM signifies binary data while a 0 in DM ies Binary Coded Decimal (BCD) data.

- The 24/12 control bit establishes the format of the hours byte. A 1 indicates the 24-hour mode and licates the 12-hour mode. This bit is read/write and is not affected by internal functions or $\overline{\text{RESET}}$.

The Daylight Savings Enable (DSE) bit is a read/write bit which enables two special updates when s set to 1. On the first Sunday in April the time increments from 1:59:59 AM to 3:00:00 AM. On ;t Sunday in October when the time first reaches 1:59:59 AM it changes to 1:00:00 AM. These l updates do not occur when the DSE bit is a zero. This bit is not affected by internal functions or $\overline{\text{T}}$.

GISTER C

B

LSB

3IT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RQF	PF	AF	UF	0	0	0	0

)F - The Interrupt Request Flag (IRQF) bit is set to a 1 when one or more of the following are true:
= PIE = 1
= AIE = 1
= UIE = 1

IRQF = (PF • PIE) + (AF • AIE) + (UF • UIE)

time the IRQF bit is a 1, the $\overline{IRQ}$ pin is driven low. Flag bits PF, AF, and UF are cleared after ster C is read by the program or when the $\overline{RESET}$ pin is low.

The Periodic Interrupt Flag (PF) is a read-only bit which is set to a 1 when an edge is detected on the ted tap of the divider chain. The RS3 through RS0 bits establish the periodic rate. PF is set to a 1 pendent of the state of the PIE bit. When both PF and PIE are 1's, the $\overline{IRQ}$ signal is active and will ie IRQF bit. The PF bit is cleared by a software read of Register C or a $\overline{RESET}$.

A 1 in the Alarm Interrupt Flag (AF) bit indicates that the current time has matched the alarm time. AIE bit is also a 1, the $\overline{IRQ}$ pin will go low and a 1 will appear in the IRQF bit. A $\overline{RESET}$ or a read gister C will clear AF.

The Update Ended Interrupt Flag (UF) bit is set after each update cycle. When the UIE bit is set to 1 in UF causes the IRQF bit to be a 1, which will assert the $\overline{IRQ}$ pin. UF is cleared by reading ster C or a $\overline{RESET}$.

3 THROUGH BIT 0 - These are unused bits of the status Register C. These bits always read 0 and ot be written.

GISTER D

LSB

IT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RT	0	0	0	0	0	0	0

- The Valid RAM and Time (VRT) bit indicates the condition of the battery connected to the V_{BAT} his bit is not writeable and should always be a 1 when read. If a 0 is ever present, an exhausted al lithium energy source is indicated and both the contents of the RTC data and RAM data are onable. This bit is unaffected by $\overline{RESET}$.

THROUGH BIT 0 - The remaining bits of Register D are not usable. They cannot be written and, read, they will always read 0.

NTURY REGISTER

century register at location 32h, is a BCD register designed to automatically load the BCD value 20 ıe year register changes from 99 to 00. The MSB of this register will not be affected when the load of ccurs and will remain at the value written by the user.

NVOLATILE RAM

113 general purpose nonvolatile RAM bytes are not dedicated to any special function within the 2C887. They can be used by the processor program as nonvolatile memory and are fully available ıg the update cycle.

ERRUPTS

RTC plus RAM includes three separate, fully automatic sources of interrupt for a processor. The n interrupt can be programmed to occur at rates from once per second to once per day. The periodic rupt can be selected for rates from 500ms to 122µs. The update-ended interrupt can be used to ate to the program that an update cycle is complete. Each of these independent interrupt conditions is ribed in greater detail in other sections of this text.

processor program can select which interrupts, if any, are going to be used. Three bits in Register B le the interrupts. Writing a logic 1 to an interrupt-enable bit permits that interrupt to be initiated ı the event occurs. A zero in an interrupt-enable bit prohibits the $\overline{IRQ}$ pin from being asserted from interrupt condition. If an interrupt flag is already set when an interrupt is enabled, $\overline{IRQ}$ is ediately set at an active level, although the interrupt initiating the event may have occurred much r. As a result, there are cases where the program should clear such earlier initiated interrupts before enabling new interrupts.

ı an interrupt event occurs, the relating flag bit is set to logic 1 in Register C. These flag bits are set ıendent of the state of the corresponding enable bit in Register B. The flag bit can be used in a ıg mode without enabling the corresponding enable bits. The interrupt flag bit is a status bit which are can interrogate as necessary. When a flag is set, an indication is given to software that an upt event has occurred since the flag bit was last read; however, care should be taken when using the ıits as they are cleared each time Register C is read. Double latching is included with Register C so ıits which are set remain stable throughout the read cycle. All bits which are set (high) are cleared read and new interrupts which are pending during the read cycle are held until after the cycle is leted. One, two, or three bits can be set when reading Register C. Each utilized flag bit should be ined when read to ensure that no interrupts are lost.

econd flag bit usage method is with fully enabled interrupts. When an interrupt flag bit is set and the ıponding interrupt enable bit is also set, the $\overline{IRQ}$ pin is asserted low. $\overline{IRQ}$ is asserted as long as at ıne of the three interrupt sources has its flag and enable bits both set. The IRQF bit in Register C is whenever the $\overline{IRQ}$ pin is being driven low. Determination that the RTC initiated an interrupt is ıplished by reading Register C. A logic one in bit 7 (IRQF bit) indicates that one or more interrupts been initiated by the DS12C887. The act of reading Register C clears all active flag bits and the bit.

;CILLATOR CONTROL BITS

en the DS12C887 is shipped from the factory, the internal oscillator is turned off. This feature vents the lithium energy cell from being used until it is installed in a system. A pattern of 010 in bits 4 ugh 6 of Register A will turn the oscillator on and enable the countdown chain. A pattern of 11X will the oscillator on, but holds the countdown chain of the oscillator in reset. All other combinations of 4 through 6 keep the oscillator off.

UARE WAVE OUTPUT SELECTION

teen of the 15 divider taps are made available to a 1-of-15 selector, as shown in the block diagram of ire 1. The first purpose of selecting a divider tap is to generate a square wave output signal on the V pin. The RS0–RS3 bits in Register A establish the square wave output frequency. These uencies are listed in Table 1. The SQW frequency selection shares its 1-of-15 selector with the odic interrupt generator. Once the frequency is selected, the output of the SQW pin can be turned on off under program control with the square wave enable bit (SQWE).

RIODIC INTERRUPT SELECTION

periodic interrupt will cause the $\overline{IRQ}$ pin to go to an active state from once every 500ms to once y 122µs. This function is separate from the alarm interrupt which can be output from once per second ice per day. The periodic interrupt rate is selected using the same Register A bits which select the re wave frequency (see Table 1). Changing the Register A bits affects both the square wave iency and the periodic interrupt output. However, each function has a separate enable bit in Register he SQWE bit controls the square wave output. Similarly, the periodic interrupt is enabled by the PIE i Register B. The periodic interrupt can be used with software counters to measure inputs, create it intervals, or await the next needed software function.

RIODIC INTERRUPT RATE AND SQUARE WAVE OUTPUT FREQUENCY

le 2

XT. REG. B	SELECT BITS REGISTER A				t_{PI} PERIODIC INTERRUPT RATE	SQW OUTPUT FREQUENCY
E32K	RS3	RS2	RS1	RS0		
0	0	0	0	0	None	None
0	0	0	0	1	3.90625 ms	256 Hz
0	0	0	1	0	7.8125 ms	128 Hz
0	0	0	1	1	122.070 µs	8.192 kHz
0	0	1	0	0	244.141 µs	4.096 kHz
0	0	1	0	1	488.281 µs	2.048 kHz
0	0	1	1	0	976.5625 µs	1.024 kHz
0	0	1	1	1	1.953125 ms	512 Hz
0	1	0	0	0	3.90625 ms	256 Hz
0	1	0	0	1	7.8125 ms	128 Hz
0	1	0	1	0	15.625 ms	64 Hz
0	1	0	1	1	31.25 ms	32 Hz
0	1	1	0	0	62.5 ms	16 Hz
0	1	1	0	1	125 ms	8 Hz
0	1	1	1	0	250 ms	4 Hz
0	1	1	1	1	500 ms	2 Hz

DATE CYCLE

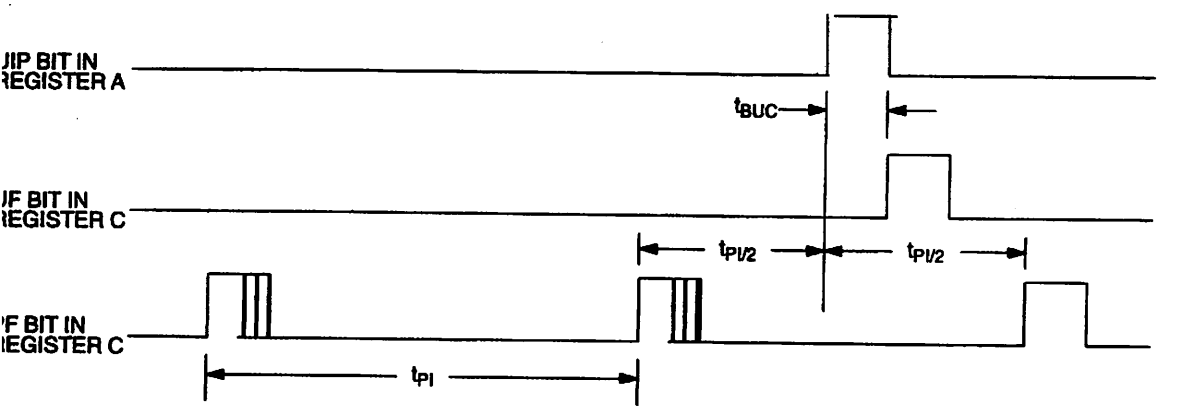
DS12C887 executes an update cycle once per second regardless of the SET bit in Register B. When SET bit in Register B is set to one, the user copy of the double buffered time, calendar, and alarm s is frozen and will not update as the time increments. How-ever, the time countdown chain continues ɔdate the internal copy of the buffer. This feature allows time to maintain accuracy independent of ing or writing the time, calendar, and alarm buffers and also guarantees that time and calendar ·mation is consistent. The update cycle also compares each alarm byte with the corresponding time and issues an alarm if a match or if a "don't care" code is present in all three positions.

e are three methods that can handle access of the real time clock that avoid any possibility of ssing inconsistent time and calendar data. The first method uses the update-ended interrupt. If led, an interrupt occurs after every up date cycle that indicates that over 999 ms are available to read l time and date information. If this interrupt is used, the IRQF bit in Register C should be cleared re leaving the interrupt routine.

cond method uses the update-in-progress bit (UIP) in Register A to determine if the update cycle is in ress. The UIP bit will pulse once per second. After the UIP bit goes high, the update transfer occurs s later. If a low is read on the UIP bit, the user has at least 244µs before the time/calendar data will ıanged. Therefore, the user should avoid interrupt service routines that would cause the time needed ıd valid time/calendar data to exceed 244µs.

hird method uses a periodic interrupt to determine if an update cycle is in progress. The UIP bit in ster A is set high between the setting of the PF bit in Register C (see Figure 3). Periodic interrupts occur at a rate of greater than t_{BUC} allow valid time and date information to be reached at each rence of the periodic interrupt. The reads should be complete within 1 ($t_{PI/2}$ + t_{BUC}) to ensure that is not read during the update cycle.

)ATE-ENDED AND PERIODIC INTERRUPT RELATIONSHIP Figure 3



ı = PERIODIC INTERRUPT TIME INTERNAL PER TABLE 1
ıUC = DELAY TIME BEFORE UPDATE CYCLE = 244 µs

SOLUTE MAXIMUM RATINGS*

age on Any Pin Relative to Ground	-0.3V to +7.0V
rating Temperature	0° to 70°C
ıge Temperature	-40°C to +70°C
ering Temperature	260°C for 10 seconds (See Note 7)

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

COMMENDED DC OPERATING CONDITIONS (0°C to 70°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
ver Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
ıt Logic 1	V_{IH}	2.3		V_{CC}+0.3	V	1
ıt Logic 0	V_{IL}	-0.3		0.8	V	1

ELECTRICAL CHARACTERISTICS (0°C to 70°C; VCC = 5.0V ± 10%)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
rage V_{CC} Power Supply rent	I_{CC1}		7	15	mA	2
ıt Leakage	I_{IL}	-1.0		+1.0	µA	3
Leakage	I_{OL}	-1.0		+1.0	µA	4
ıt Current	I_{MOT}	-1.0		+500	µA	3
ɔut @ 2.4V	I_{OH}	-1.0			mA	1,5
ɔut @ 0.4V	I_{OL}			4.0	mA	1
e Protect Voltage	V_{TP}	4.0	4.25	4.5	V	

'ACITANCE (tA = 25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
t Capacitance	C_{IN}			5	pF	
ut Capacitance	C_{OUT}			7	pF	

ELECTRICAL CHARACTERISTICS (0°C to 70°C; VCC = 5.0V ± 10%)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
cle Time	t_{CYC}	385		DC	ns	
se Width, DS/E Low or /$\overline{WR}$ High	PW_{EL}	150			ns	
se Width, DS/E High or /$\overline{WR}$ Low	PW_{RH}	125			ns	
ut Rise and Fall	t_R, t_F			30	ns	
$\overline{V}$ Hold Time	t_{RWH}	10			ns	
$\overline{V}$ Setup Time Before DS/E	t_{RWS}	50			ns	
p Select Setup Time Before , $\overline{WR}$, or $\overline{RD}$	t_{CS}	20			ns	
p Select Hold Time	t_{CH}	0			ns	
d Data Hold Time	t_{DHR}	10		80	ns	
te Data Hold Time	t_{DHW}	0			ns	
x'ed Address Valid Time to E Fall	t_{ASL}	30			ns	
sed Address Hold Time to E Fall	t_{AHL}	10			ns	
ay Time DS/E to AS/ALE	t_{ASD}	20			ns	
e Width AS/ALE High	PW_{ASH}	60			ns	
ay Time, AS/ALE to DS/E	t_{ASED}	40			ns	
put Data Delay Time from E or $\overline{RD}$	t_{DDR}	20		120	ns	6
Setup Time	t_{DSW}	100			ns	
t Pulse Width	t_{RWL}	5			µs	
Release from DS	t_{IRDS}			2	µs	
Release from $\overline{RESET}$	t_{IRR}			2	µs	

ES:

ll voltages are referenced to ground.

ll Outputs are open.

e MOT pin has an internal pull-down of 20KΩ.

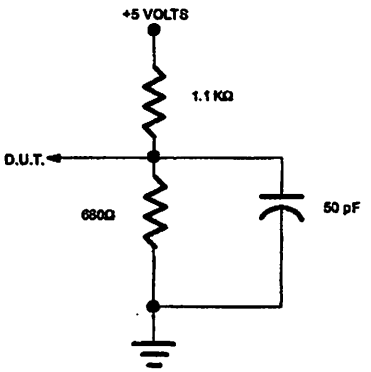
pplies to the AD0-AD7 pins, the $\overline{IRQ}$ pin, and the SQW pin when each is in a high impedance state.

e $\overline{IRQ}$ pin is open drain.

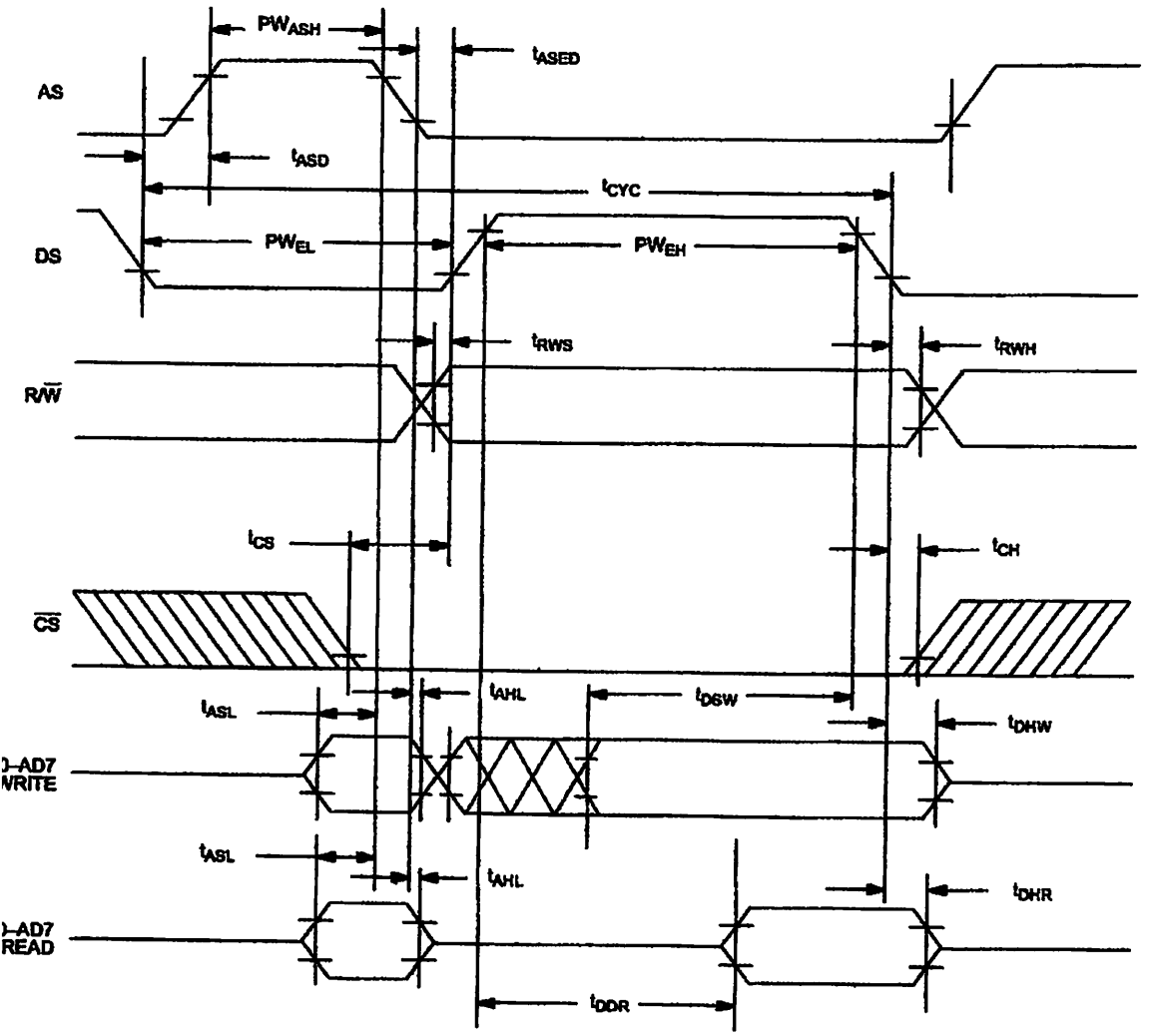
easured with a load as shown in Figure 4.

al-Time Clock Modules can be successfully processed through conventional wave-soldering chniques as long as temperature exposure to the lithium energy source contained within does not ceed +85°C. Post solder cleaning with water washing techniques is acceptable, provided that rasonic vibration is not used. Such cleaning can damage the crystal.

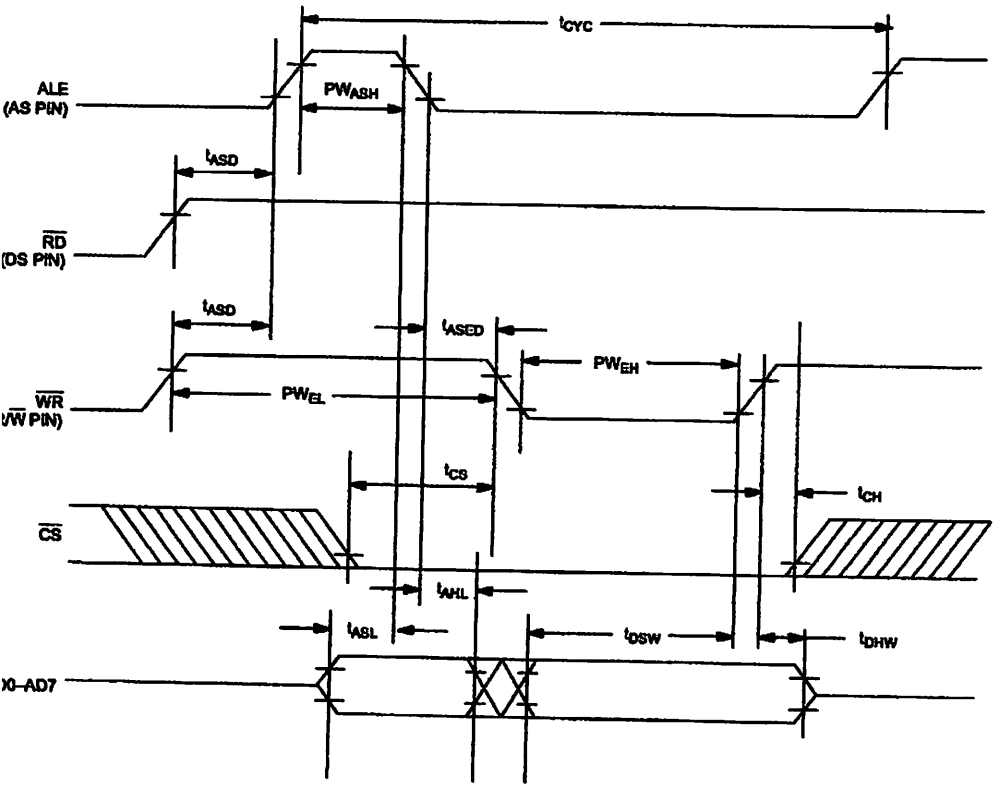
TPUT LOAD Figure 4



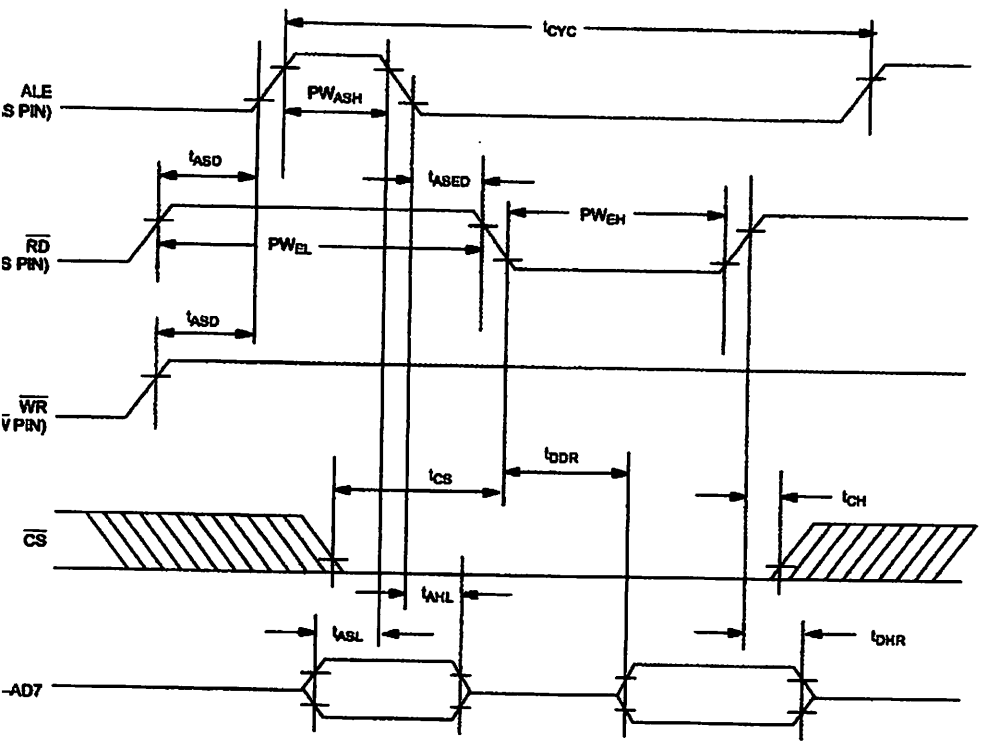
12C887 BUS TIMING FOR MOTOROLA INTERFACE



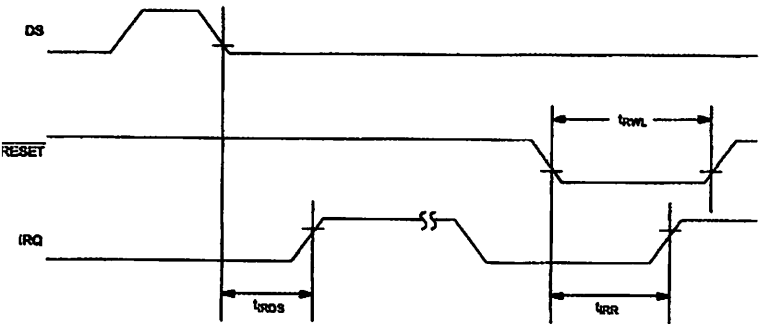
I2C887 BUS TIMING FOR INTEL INTERFACE WRITE CYCLE



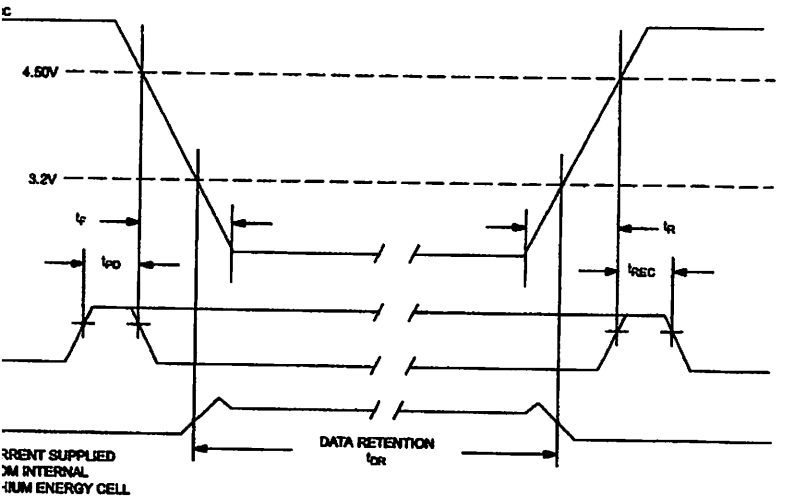
2C887 BUS TIMING FOR INTEL INTERFACE READ CYCLE



12C887 IRQ RELEASE DELAY TIMING



WER DOWN / POWER UP TIMING



WER DOWN / POWER UP TIMING

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
at V_{IH} before Power-wn	t_{PD}			0	µs	
slew from 4.5V to 0V at V_{IH})	t_F $4.0 \leq V_{CC} \leq 4.5V$	300			µs	
slew from 0V to 4.5V at V_{IH})	t_R	100			µs	
at V_{IH} after Power-Up	t_{REC}	20		200	ms	
ected Data Retention	t_{DR}	10			years	10,11

(tA=25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
ected Data Retention	t_{DR}	10			years	10,11

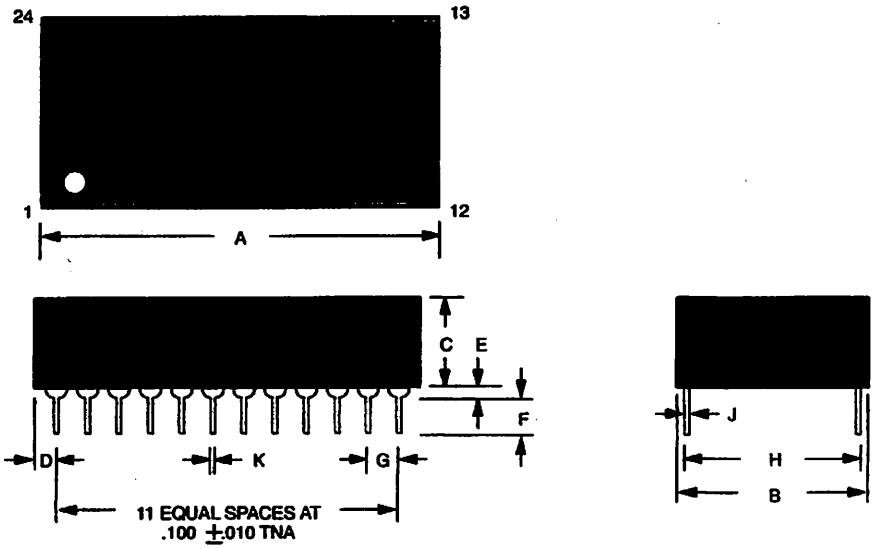
e:

eal time clock will keep time to an accuracy of ±1 minute per month during data retention time for eriod of t_{DR}.

ning:

r no circumstances are negative undershoots, of any amplitude, allowed when device is in battery up mode.

12C887 REAL TIME CLOCK PLUS RAM



TE:

2, 3, 16, 20, 21 and 22 are missing by design

	24-PIN	
	MIN	MAX
	1.320 33.53	1.335 33.91
	0.675 17.15	0.700 17.78
	0.345 8.76	0.370 9.40
	0.100 2.54	0.130 3.30
	0.015 0.38	0.030 0.76
	0.110 2.79	0.140 3.56
	0.090 2.29	0.110 2.79
	0.590 14.99	0.630 16.00
	0.008 0.20	0.012 0.30
	0.015 0.38	0.021 0.53

Plastic Medium-Power Complementary Silicon Transistors

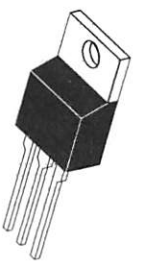
. . designed for general–purpose amplifier and low–speed switching applications.

- High DC Current Gain —
  h_{FE} = 2500 (Typ) @ I_C = 1.0 Adc
- Collector–Emitter Sustaining Voltage — @ 30 mAdc
  $V_{CEO(sus)}$ = 60 Vdc (Min) — TIP110, TIP115
  = 80 Vdc (Min) — TIP111, TIP116
  = 100 Vdc (Min) — TIP112, TIP117
- Low Collector–Emitter Saturation Voltage —
  $V_{CE(sat)}$ = 2.5 Vdc (Max) @ I_C = 2.0 Adc
- Monolithic Construction with Built–in Base–Emitter Shunt Resistors
- TO–220AB Compact Package

NPN
TIP110
TIP111*
TIP112*
PNP
TIP115
TIP116*
TIP117*

*Motorola Preferred Device

DARLINGTON
2 AMPERE
COMPLEMENTARY SILICON
POWER TRANSISTORS
60–80–100 VOLTS
50 WATTS



CASE 221A–06
TO–220AB

MAXIMUM RATINGS

Rating	Symbol	TIP110, TIP115	TIP111, TIP116	TIP112, TIP117	Unit
Collector–Emitter Voltage	V_{CEO}	60	80	100	Vdc
Collector–Base Voltage	V_{CB}	60	80	100	Vdc
Emitter–Base Voltage	V_{EB}	5.0			Vdc
Collector Current — Continuous Peak	I_C	2.0 4.0			Adc
Base Current	I_B	50			mAdc
Total Power Dissipation @ T_C = 25°C Derate above 25°C	P_D	50 0.4			Watts W/°C
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	2.0 0.016			Watts W/°C
Unclamped Inductive Load Energy — Figure 13	E	25			mJ
Operating and Storage Junction	T_J, T_{stg}	−65 to +150			°C

THERMAL CHARACTERISTICS

Characteristics	Symbol	Max	Unit
Thermal Resistance, Junction to Case	$R_{\theta JC}$	2.5	°C/W
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	62.5	°C/W

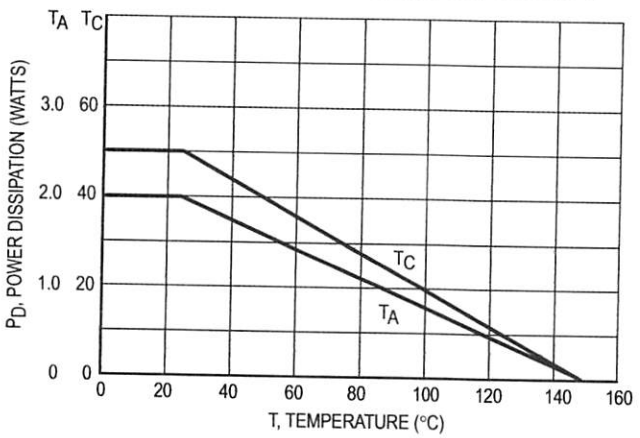


Figure 1. Power Derating

ferred devices are Motorola recommended choices for future use and best overall value.

EV 1

ELECTRICAL CHARACTERISTICS (T_C = 25°C unless otherwise noted)

Characteristic		Symbol	Min	Max	Unit
OFF CHARACTERISTICS					
Collector–Emitter Sustaining Voltage (1) (I_C = 30 mAdc, I_B = 0)	TIP110, TIP115 TIP111, TIP116 TIP112, TIP117	$V_{CEO(sus)}$	60 80 100	— — —	Vdc
Collector Cutoff Current (V_{CE} = 30 Vdc, I_B = 0) (V_{CE} = 40 Vdc, I_B = 0) (V_{CE} = 50 Vdc, I_B = 0)	TIP110, TIP115 TIP111, TIP116 TIP112 ,TIP117	I_{CEO}	— — —	2.0 2.0 2.0	mAdc
Collector Cutoff Current (V_{CB} = 60 Vdc, I_E = 0) (V_{CB} = 80 Vdc, I_E = 0) (V_{CB} = 100 Vdc, I_E = 0)	TIP110, TIP115 TIP111, TIP116 TIP112, TIP117	I_{CBO}	— — —	1.0 1.0 1.0	mAdc
Emitter Cutoff Current (V_{BE} = 5.0 Vdc, I_C = 0)		I_{EBO}	—	2.0	mAdc
ON CHARACTERISTICS (1)					
DC Current Gain (I_C = 1.0 Adc, V_{CE} = 4.0 Vdc) (I_C = 2.0 Adc, V_{CE} = 4.0 Vdc)		h_{FE}	1000 500	— —	—
Collector–Emitter Saturation Voltage (I_C = 2.0 Adc, I_B = 8.0 mAdc)		$V_{CE(sat)}$	—	2.5	Vdc
Base–Emitter On Voltage (I_C = 2.0 Adc, V_{CE} = 4.0 Vdc)		$V_{BE(on)}$	—	2.8	Vdc
DYNAMIC CHARACTERISTICS					
Small–Signal Current Gain (I_C = 0.75 Adc, V_{CE} = 10 Vdc, f = 1.0 MHz)		h_{fe}	25	—	—
Output Capacitance (V_{CB} = 10 Vdc, I_E = 0, f = 0.1 MHz)	TIP115, TIP116, TIP117 TIP110, TIP111, TIP112	C_{ob}	— —	200 100	pF

(1) Pulse Test: Pulse Width ≤ 300 µs, Duty Cycle ≤ 2%.

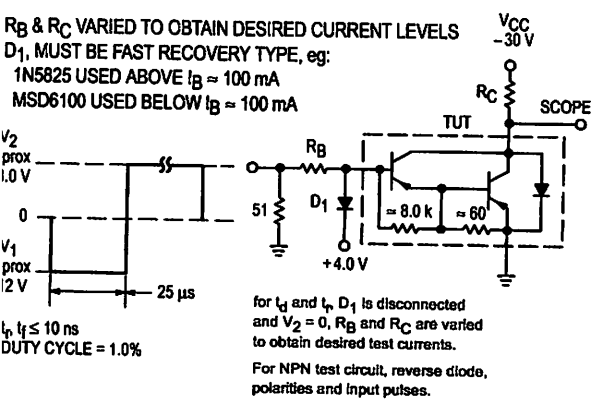


Figure 2. Switching Times Test Circuit

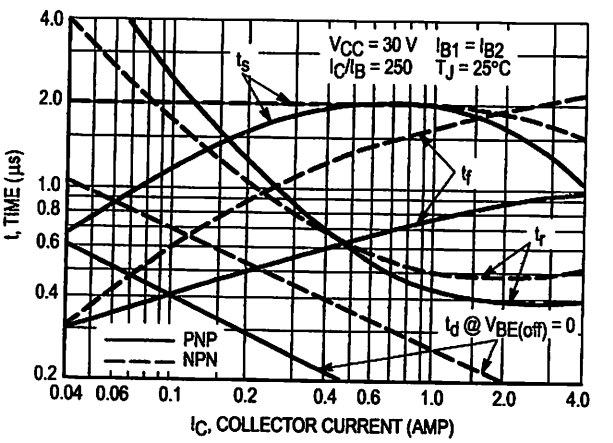


Figure 3. Switching Times

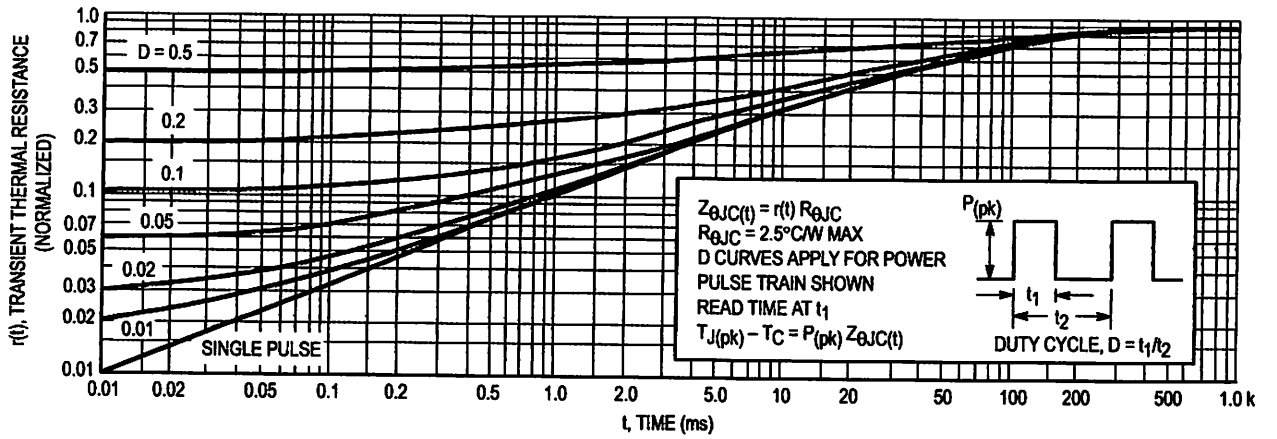


Figure 4. Thermal Response

ACTIVE–REGION SAFE–OPERATING AREA

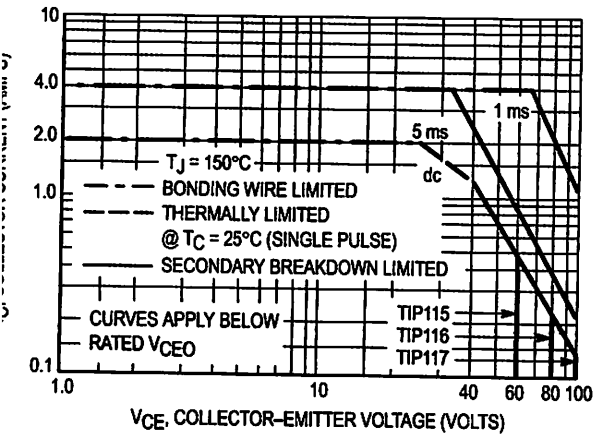


Figure 5. TIP115, 116, 117

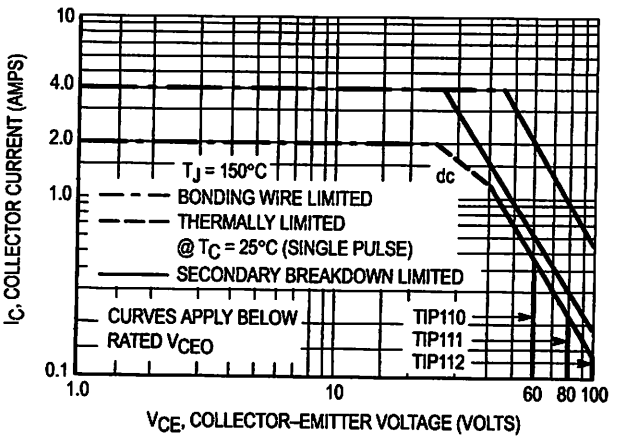


Figure 6. TIP110, 111, 112

There are two limitations on the power handling ability of a ınsistor: average junction temperature and second break-ıwn. Safe operating area curves indicate $I_C - V_{CE}$ limits of ɜ transistor that must be observed for reliable operation; ., the transistor must not be subjected to greater dissipa-n than the curves indicate.

The data of Figures 5 and 6 is based on $T_{J(pk)}$ = 150°C; ; is variable depending on conditions. Second breakdown lse limits are valid for duty cycles to 10% provided $T_{J(pk)}$ l50°C. $T_{J(pk)}$ may be calculated from the data in Figure 4. high case temperatures, thermal limitations will reduce the wer that can be handled to values less than the limitations ɔosed by second breakdown.

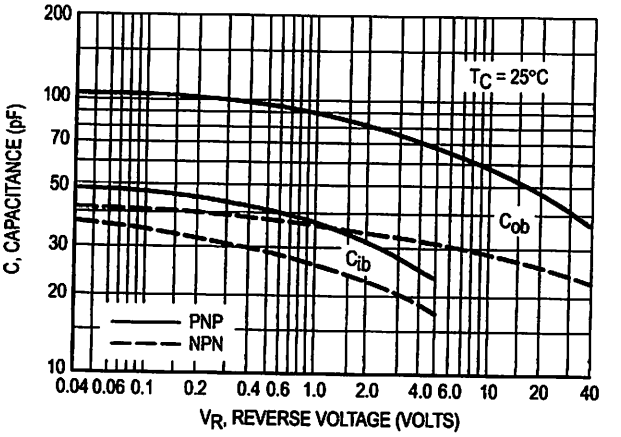


Figure 7. Capacitance

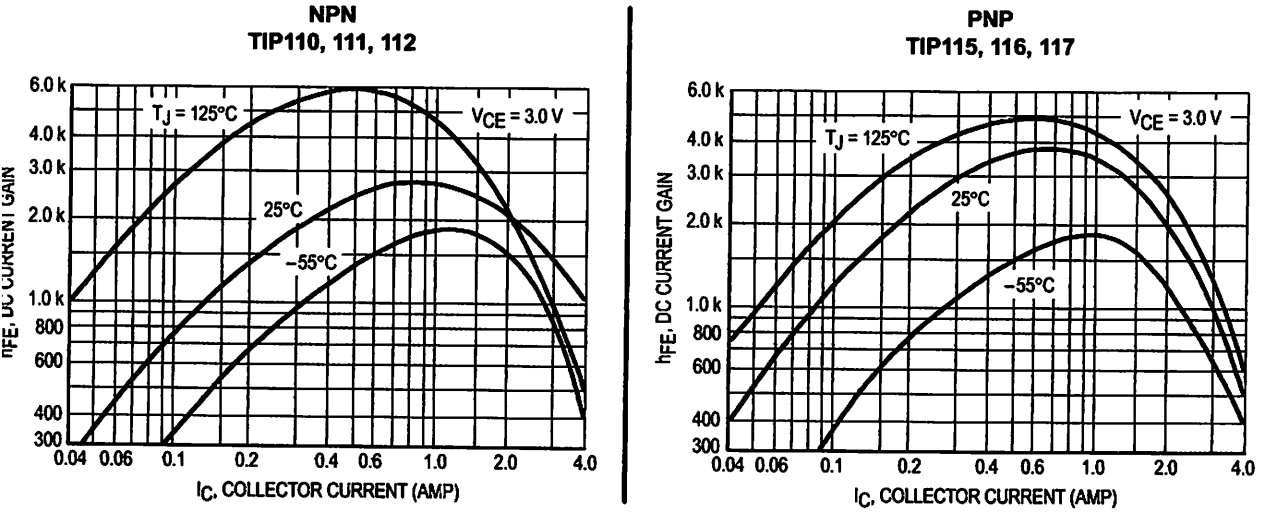


Figure 8. DC Current Gain

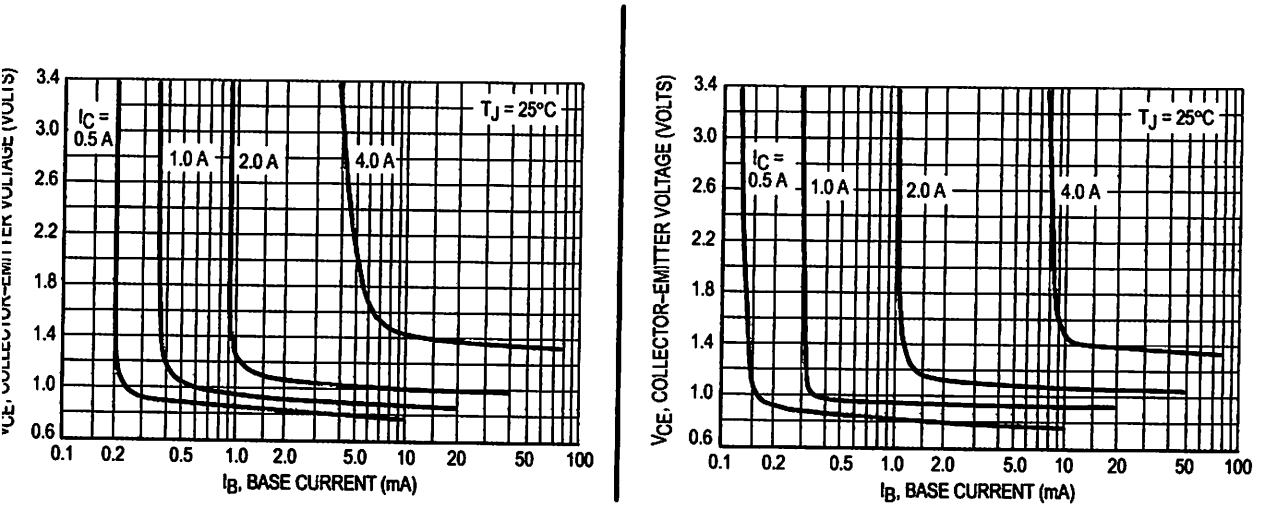


Figure 9. Collector Saturation Region

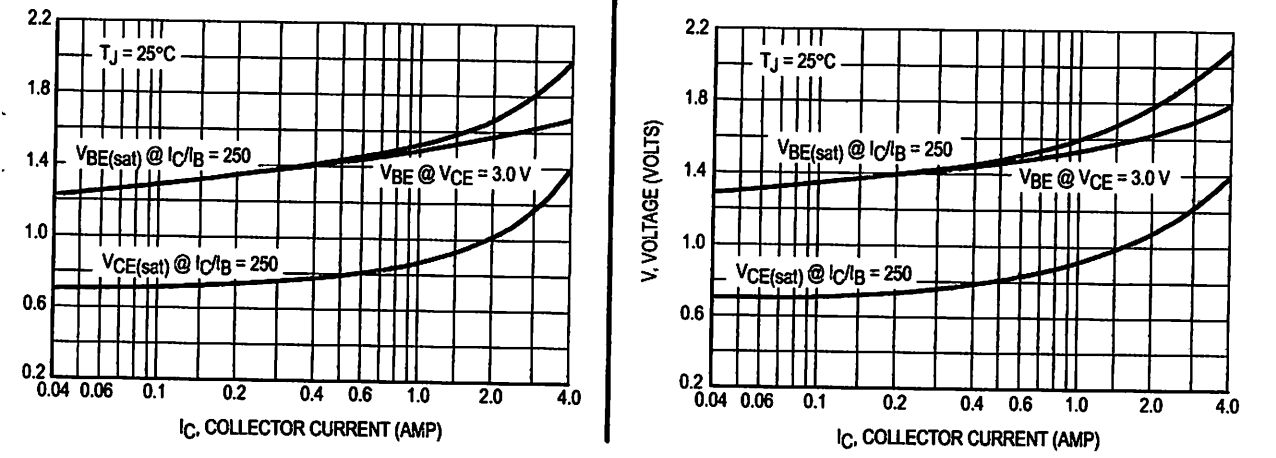


Figure 10. "On" Voltages

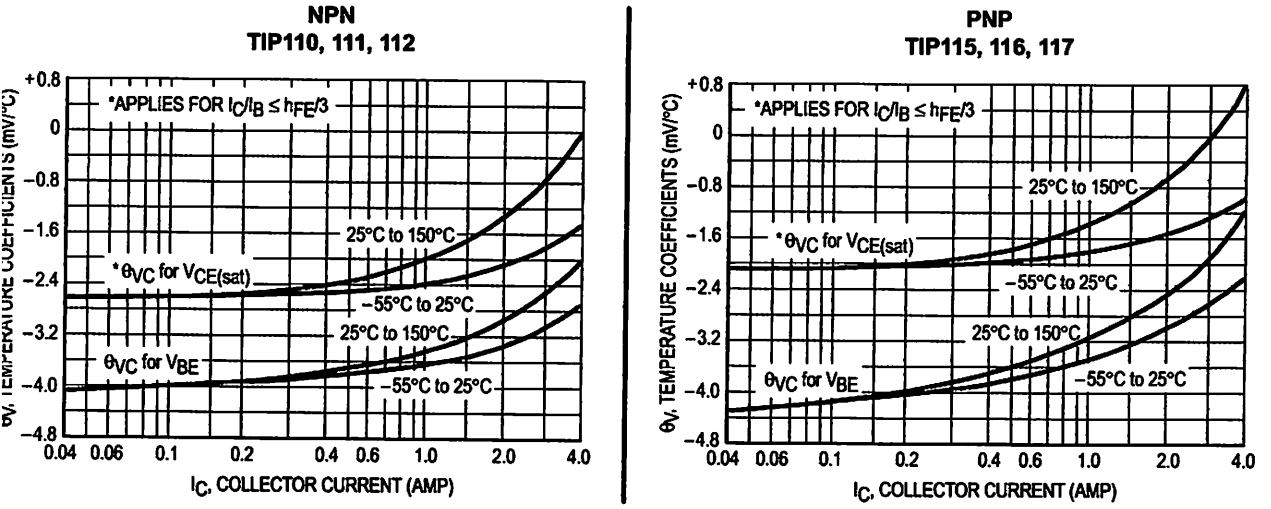


Figure 11. Temperature Coefficients

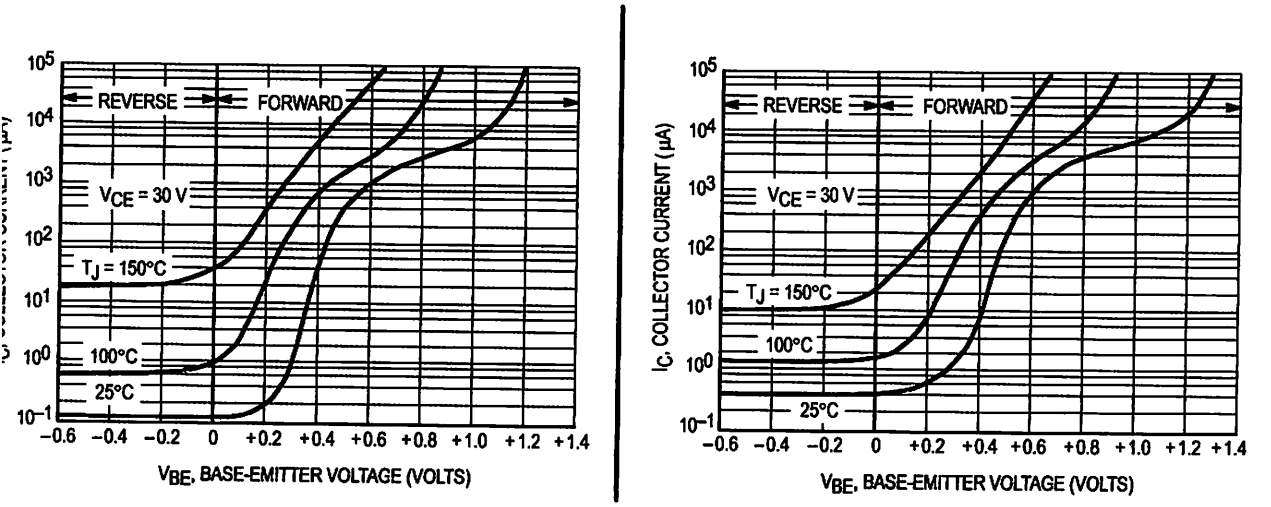
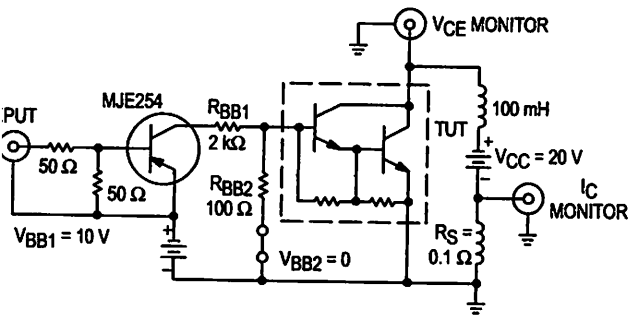
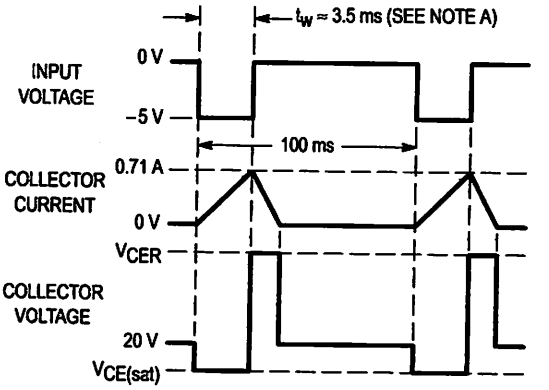


Figure 12. Collector Cut-Off Region

TEST CIRCUIT



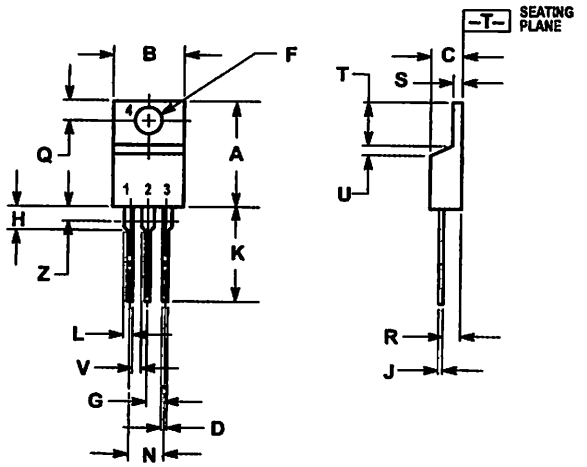
VOLTAGE AND CURRENT WAVEFORMS



Note A: Input pulse width is increased until I_{CM} = 0.71 A, NPN test shown; for PNP test reverse all polarity and use MJE224 driver.

Figure 13. Inductive Load Switching

PACKAGE DIMENSIONS



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION Z DEFINES A ZONE WHERE ALL BODY AND LEAD IRREGULARITIES ARE ALLOWED.

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.570	0.620	14.48	15.75
B	0.380	0.405	9.66	10.28
C	0.160	0.190	4.07	4.82
D	0.025	0.035	0.64	0.88
F	0.142	0.147	3.61	3.73
G	0.095	0.105	2.42	2.66
H	0.110	0.155	2.80	3.93
J	0.018	0.025	0.46	0.64
K	0.500	0.562	12.70	14.27
L	0.045	0.060	1.15	1.52
N	0.190	0.210	4.83	5.33
Q	0.100	0.120	2.54	3.04
R	0.080	0.110	2.04	2.79
S	0.045	0.055	1.15	1.39
T	0.235	0.255	5.97	6.47
U	0.000	0.050	0.00	1.27
V	0.045	—	1.15	—
Z	—	0.080	—	2.04

STYLE 1:
PIN 1. BASE
2. COLLECTOR
3. EMITTER
4. COLLECTOR

CASE 221A–06
TO–220AB
ISSUE Y

w to reach us:
A / EUROPE: Motorola Literature Distribution;
). Box 20912; Phoenix, Arizona 85036. 1–800–441–2447

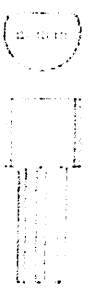
AX: RMFAX0@email.sps.mot.com – TOUCHTONE (602) 244–6609
'ERNET: http://Design–NET.com

JAPAN: Nippon Motorola Ltd.; Tatsumi–SPD–JLDC, Toshikatsu Otsuki, 6F Seibu–Butsuryu–Center, 3–14–2 Tatsumi Koto–Ku, Tokyo 135, Japan. 03–3521–8315

HONG KONG: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852–26629298

9013 NPN SILICON TRANSISTOR

TO—92



1.EMITTER
发 射 极
2.BASE
基 极
3.COLLECTOR
集 电 极

1 2 3

EATURES
征

Power dissipation （最大耗散功率）
P_{CM} : 0.625 W（Tamb=25℃）

Collector current （最大集电极电流）
I_{CM} : 0.5 A

Collector-base voltage （集电极--基极击穿电压）
$V_{(BR)CBO}$: 45 V

ECTRICAL CHARACTERISTICS（Tamb=25℃ unless otherwise specified）
电 特 性 （环境温度 除 非 另 有 规 定）

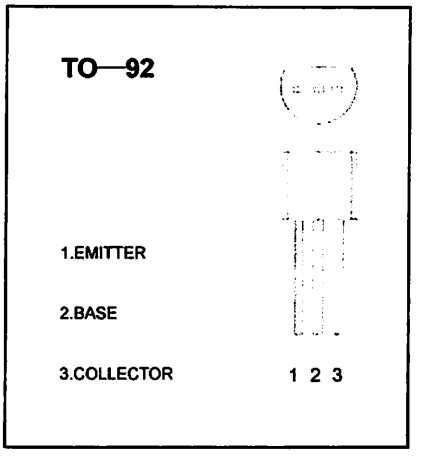
Parameter 参 数	Symbol 符 号	Test conditions 测 试 条 件	MIN 最小值	TYP 典型值	MAX 最大值	UNIT 单位
llector-base breakdown voltage 电 极 - - 基 极 击 穿 电 压	$V(BR)_{CBO}$	Ic= 100 μ A, I_E=0	45			V
llector-emitter breakdown voltage 电 极 - - 发 射 极 击 穿 电 压	$V(BR)_{CEO}$	Ic= 0. 1 mA, I_B=0	25			V
itter-base breakdown voltage 射 极 - - 基 极 击 穿 电 压	$V(BR)_{EBO}$	I_E= 100 μ A, I_C=0	5			V
llector cut-off current 电 极 - - 基 极 截 止 电 流	I_{CBO}	V_{CB}= 40 V I_E=0			0.1	μ A
llector cut-off current 电极--发射极截止电流	I_{CEO}	V_{CE}= 20 V I_B=0			0.1	μ A
itter cut-off current 射极--基极截止电流	I_{EBO}	V_{EB}= 5 V, I_C=0			0.1	μ A
current gain(note) 流 电 流 增 益	$H_{FE(1)}$	V_{CE}= 1 V, I_C= 50 mA	64		300	
	$H_{FE(2)}$	V_{CE}= 1V, I_C =500 mA	40			
lector-emitter saturation voltage 电 极 - - 发 射 极 饱 和 压 降	V_{CE}(sat)	I_C= 500 mA, I_B=50 mA			0.6	V
e-emitter saturation voltage 极--发射极饱和压降	V_{BE}(sat)	I_C= 500mA, I_B= 50 mA			1.2	V
e-emitter voltage 极--发射极正向电压	V_{BE}	I_E=100mA			1.4	V
sition frequency 征 频 率	f_T	V_{CE}= 6 V, I_C= 20 mA f =30MHz	150			MHz

SSIFICATION OF $H_{FE(1)}$ （分类）

ak 类	D	E	F	G	H	I
ge 围	64-91	78-112	96-135	112-166	144-220	190-300

hing Computer Components Co., (H.K.)Ltd.
age: http://www.wingshing.com
Tel:(852)2341 9276 Fax:(852)2797 8153
E-mail: wsccltd@hkstar.com

9012 PNP SILICON TRANSISTOR



FEATURES

Power dissipation

P_{CM} : 0.625 W (Tamb=25℃)

Collector current

I_{CM} : -0.5 A

Collector-base voltage

$V_{(BR)CBO}$: -40 V

ELECTRICAL CHARACTERISTICS (Tamb=25℃ unless otherwise specified)

Parameter	Symbol	Test conditions	MIN	TYP	MAX	UNIT
Collector-base breakdown voltage	$V(BR)_{CBO}$	Ic= -100 μ A, I_E=0	-40			V
Collector-emitter breakdown voltage	$V(BR)_{CEO}$	Ic= -0. 1 mA, I_B=0	-20			V
Emitter-base breakdown voltage	$V(BR)_{EBO}$	I_E= -100 μ A, I_C=0	-5			V
Collector cut-off current	I_{CBO}	V_{CB}=-40 V I_E=0			-0.1	μ A
Collector cut-off current	I_{CEO}	V_{CE}=- 20 V I_B=0			-0.2	μ A
Emitter cut-off current	I_{EBO}	V_{EB}= - 5 V, I_C=0			-0.1	μ A
DC current gain(note)	$H_{FE(1)}$	V_{CE}= -1 V, I_C= -50 mA	64		300	
	$H_{FE(2)}$	V_{CE}= -1V, I_C =-500 mA	40			
Collector-emitter saturation voltage	V_{CE}(sat)	I_C=-500 mA, I_B=-50 mA			-0.6	V
Base-emitter saturation voltage	V_{BE}(sat)	I_C=-500mA,I_B=-50 mA			-1.2	V
Base-emitter voltage	V_{EB}	I_E=-100mA			-1.4	V
Transition frequency	f_T	V_{CE}=- 6 V, I_C= -20 mA f =30MHz	150			MHz

CLASSIFICATION OF $H_{FE(1)}$

Rank	D	E	F	G	H	I
Range	64-91	78-112	96-135	112-166	144-202	190-300

Wing Shing Computer Components Co., (H.K.)Ltd. Tel:(852)2341 9276 Fax:(852)2797 8153
Homepage: http://www.wingshing.com E-mail: ic@wingshing.com