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

SKRIPSI

PERANCANGAN DAN PEMBUATAN ALAT PENYIRAMAN AIR OTOMATIS PADA KEDELAI MENGGUNAKAN MIKROKONTROLER AT89S8252

Disusun Oleh :
ACHMAD EBIT SUGIANTO
NIM. 00.17.138

MARET 2006

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN ALAT PENYIRAM AIR OTOMATIS PADA KEDELAI MENGGUNAKAN MIKROKONTROLLER AT89S8252

SKRIPSI

Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat
Guna Mencapai Gelar Sarjana Teknik

Disusun oleh :

ACHMAD EBIT SUGIANTO

00.17138

Malang, Maret 2006

Mengetahui	Disetujui
Ketua Jurusan Teknik Elektro S-1	**Dosen Pembimbing**
Ir. F. Yudi Limpraptono, MT	**Ir. Teguh Herbasuki, MT**
NIP.Y.1039500274	**NIP.Y.1038900209**

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2006

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa	:	ACHMAD EBIT SUGIANTO
NIM	:	00 17 138
Jurusan	:	Teknik Elektro S-1
Konsentrasi	:	Teknik Elektronika
Judul Skripsi	:	

"PERANCANGAN DAN PEMBUATAN ALAT PENYIRAMAN AIR OTOMATIS PADA KEDELAI MENGGUNAKAN MIKROKONTROLLER AT89S8252"

Dipertahankan Dihadapan Team Penguji Skripsi Jenjang Strata Satu (S-1), pada :

Hari	:	Rabu
Tanggal	:	22 Maret 2006
Dengan nilai	:	79,6 (B+)

Panitia Ujian Skripsi:

Ketua	Sekretaris
(Ir. Mochtar Asroni, MSME)	(Ir.F.Yudi Limpraptono, MT)
NIP.Y. 1018100036	NIP.Y 1039500274

Anggota Penguji:

Penguji I	Penguji II
(Ir.Widodo Pudji M, MT)	(Sotyohadi, ST)

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa	:	ACHMAD ERIT SUGIANTO
NIM	:	00 17 138
Jurusan	:	Teknik Elektro S-1
Konsentrasi	:	Teknik Elektronika
Judul Skripsi	:	

"PERANCANGAN DAN PEMBUATAN ALAT PENGYIRAMAN AIR OTOMATIS PADA KEBUN MENGGUNAKAN MIKROKONTROLLER AT89S8252"

Dipertahankan Dihadapan Team Penguji Skripsi Jenjang Strata Satu (S-1), pada :

Hari	:	Rabu
Tanggal	:	22 Maret 2006
Dengan nilai	:	79,5 (B+)

Panitia Ujian Skripsi:

Ketua	Sekretaris
(Ir. Mochtar Asroni, MSME)	(Ir.F.Yudi Limpraptono, MT)
NIP.Y. 1018100036	NIP.Y. 1039500274

Anggota Penguji:

Penguji I	Penguji II
(Ir.Widodo Pudji M, MT)	(Sotyohadi, ST)

KATA PENGANTAR

Puji dan Syukur Alhamdulillah penulis panjatkan kepada Allah SWT karena berkat Rahmat dan karunia-Nya disertai segala cobaan dan rintangan serta keterbatasan waktu, yang semua itu penulis menganggap sebagai tantangan dan rintangan yang harus terjawab akhirnya penulis dapat menyelesaikan skripsi ini yang berjudul :

"PERANCANGAN DAN PEMBUATAN ALAT PENYIRAMAN AIR OTOMATIS PADA KEDELAI MENGGUNAKAN MIKROKONTROLLER AT89S8252 "

Tujuan penulisan Skripsi ini antara lain untuk memenuhi persyaratan mencapai gelar Sarjana Teknik pada Jurusan Teknik Elektro S1, Konsentrasi Teknik Elektronika, di Institut Teknologi Nasional Malang.

Sehubungan dengan itu penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada :

1. Bapak Dr. Ir. Abraham Lomi, MSME. selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME. selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir. F.Yudi Limpraptono, MT. selaku Ketua Jurusan Teknik Elektro Institut Teknologi Nasional Malang.
4. Ibu Ir. Mimien Mustikawati selaku Sekretaris Jurusan Teknik Elektro S-1 Program Studi Teknik Elektronika Institut Teknologi Nasional Malang.

5. Bapak Ir. Teguh Herbasuki, MT. selaku Dosen Pembimbing yang telah banyak meluangkan waktu, arahan dan juga tuntunan yang progresif dalam penyusunan skripsi ini sampai selesai.
6. Bapak dan Ibu dosen yang telah memberikan bekal ilmu selama penulis kuliah di ITN Malang.
7. Teman-teman seperjuangan "angkatan 2000 Elka 3-4" yang telah memberikan masukan–masukan dan saran–saran serta semangat belajar yang tidak dapat disebutkan satu persatu juga untuk semua pihak yang telah banyak membantu dalam menyelesaikan Skripsi ini.

Secara khusus, kebanggaan disertai kasih yang tulus, penulis tujukan kepada Bapak dan Ibu yang telah bersusah payah memberikan kesempatan biaya kepada penulis untuk mengikuti dan menyelesaikan studi ini.

Akhir kata penulis mohon maaf sebesar-besarnya bila ada kata-kata atau kesalahan dalam penulisan laporan ini tentunya tanpa ada unsur kesengajaan dari penulis. Penulis mengharapkan saran dan kritik yang membangun dari pembaca dan harapan saya semoga laporan Tugas Akhir ini dapat bermanfaat bagi kita semua . "Amin".

Malang, Maret 2006

Penulis

ABSTARKSI

Achmad Ebit Sugianto, November-2005, *Perancangan Dan Pembuatan Alat Penyiram air Otomatis Pada Kedelai Menggunakan Mikrokontroller AT89S8252.* Skripsi Jurursan Teknik Elektronika Fakultas Teknologi Indusri Institut Teknologi Nasional Malang. Pembimbing : Ir. Teguh Herbasuki, MT

Teknologi di bidang elektronika mempengaruhi perubahan pola kehidupan masyarakat yang dulunya memanfaatkan tenaga murni/konvensional kini berubah menggunakan suatu alat yang dapat bekerja sendiri (otomatis) yang hanya membutuhkan sedikit sentuhan tenaga manusia.

Pada era globalisasi banyak petani disibukkan dengan berbagai aktifitas dalam bidang pertanian. Seperti halnya yang dilakukan oleh petani dalam penyiraman pada kedelai secara konvensional, yakni harus disiram 3 jam sekali sampai 2 hari dari waktu persemaian. Dengan adanya alat penyiraman otomatis pada kedelai ini, diharapkan petani tidak lagi melakukan aktifitas tersebut lebih-lebih jika memiliki lahan yang luas dan dilakukan berhari-hari. Alat penyiraman otomatis ini dalam bekerjanya tidak lagi berdasarkan satuan waktu penyiraman dalam sehari sekian kali secara mutlak, tetapi lebih didasarkan pada kelembaban tanah terhadap umur tanaman kedelai tersebut. Dalam skripsi ini akan dibuat suatu alat penyiram otomatis dengan sistem pemrograman berbasis mikrokontroler AT89S8252.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan ilmu di bidang teknologi khususnya teknologi elektronika merupakan suatu kenyataan yang menunjukkan bahwa manusia banyak melakukan aktivitasnya. Sesuai dengan tuntutan jaman yang bermacam-macam tentang kebutuhan masyarakat akan sebuah hiburan ataupun alat kerja, hal ini telah membawa ide ke arah teknologi yang lebih maju, lebih bermanfaat serta memiliki nilai ekonomis.

Oleh karena itu berbagai macam sebab perkembangan teknologi khususnya bidang elektronika mempengaruhi perubahan pola kehidupan masyarakat yang dulunya memanfaatkan tenaga murni/konvensional kini berubah menggunakan suatu alat yang dapat bekerja sendiri (otomatis) yang hanya membutuhkan sedikit sentuhan tenaga manusia.

Seperti halnya yang dilakukan oleh petani dalam penyiraman pada kedelai secara konvensional, yakni harus disiram 3 jam sekali sampai 2 hari dari waktu persemaian. Dengan adanya alat penyiraman otomatis pada kedelai ini, diharapkan petani tidak lagi melakukan aktifitas tersebut lebih-lebih jika memiliki lahan yang luas dan dilakukan berhari-hari. Alat penyiraman otomatis ini dalam bekerjanya tidak lagi berdasarkan satuan waktu penyiraman dalam sehari sekian kali secara mutlak, tetapi lebih didasarkan pada kelembaban tanah terhadap umur tanaman kedelai tersebut.

Dimana alat yang menggunakan *Real Time Clock* (RTC) sebagai kontrol waktu, keypad sebagai inputan yang mempermudah dalam memasukkan waktu penyiraman atau umur tanaman dan mikrokontroler sebagai basis system pemrogramannya.

1.2 Rumusan Masalah

Permasalahan yang dihadapi dalam merencanakan dan membuat suatu alat penyiram air otomatis pada tanaman kedelai, dapat dirumuskan sebagai berikut:

1. Bagaimana membuat sensor yang mampu mendeteksi kelembaban tanah.
2. Bagaimana membuat suatu rangkaian driver yang dapat membuka kran air secara elektrik.
3. Bagaimana mengendalikan IC RTC dan menggunakannya sebagai referensi terhadap waktu yang presisi.
4. Bagaimana merangkai software yang dapat membaca sensor dan memperhitungkannya sehingga mengeluarkan data sebagai kontrol kran air otomatis.

1.3 Tujuan

Adapun tujuan pada pembuatan alat ini adalah:

Untuk mengaplikasikan sistem pemrograman dengan berbasis mikrokontroller sebagai kontrol penyiraman pada tanaman kedelai dengan referensi kelembaban tanah

1.4 Batasan Masalah

Agar permasalahan yang dibahas tidak meluas, maka penulis membatasi pembahasan hanya pada hal-hal berikut ini:

- Pengontrol utama terprogram menggunakan mikrokontroler AT 89S8252.
- Menggunakan RTC DS12C887 sebagai pengontrol waktu.
- Tidak membahas secara detail system mekanik penyiraman (debit air).
- Pembahasan aplikasi hanya ditujukan untuk tanaman kedelai.
- Tidak membahas secara detail mengenai system catu daya.

1.5 Metodologi

Untuk mendapatkan hasil yang diharapkan, maka di dalam penyusunan laporan tugas akhir ini, penulis menggunakan metode sebagai berikut :

1. Studi literatur:

   Studi literatur disini untuk mempelajari hal-hal dasar mengenahi teori tiap-tiap komponen yang digunakn sebagai berikut :

2. Perencanaan alat:
   - ✓ Melakukan perencanan untuk tiap-tiap blok diagram yang mendukung.
   - ✓ Mempersiapkan dan memilih komponen berdasarkan fungsi dan kerakteristik komponen, kemudian tersedianya dipasaran.

3. Pembuatan sekaligus pengujian alat:

   Dalam pembuatan alat disini meliputi perencanaan tata letak komponen, pembuatan layer pada PCB, pengeboran dan penyolderan, serta perakitan tiap-

tiap blok menjadi suatu rangkaian yang lengkap yang selanjutnya diuji tiap blok maupun keseluruhan.

4. Kesimpulan dan saran:

   Setelah melakukan beberapa pengujian, maka mengambil kesimpulan dan memberi saran yang diperlukan untuk pengembangan alat selanjutnya.

1.6 Sistematika Penulisan

Sistematika penulisan dalam skripsi ini terdiri dari lima (5) bab, yaitu:

BAB I : PENDAHULUAN

Berisi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penulisan, metodologi dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Membahas tentang teori-teori dasar yang berhubungan dengan alat.

BAB II : PERENCANAAN PEMBUATAN ALAT

Membahas tentang perencanaan dan pembuatan alat penyiraman air otomatis pada kedelai

BAB IV : PENGUJIAN ALAT

Membahas mengenahi pengujian dan analisis hasil pengujian.

BAB V : PENUTUP

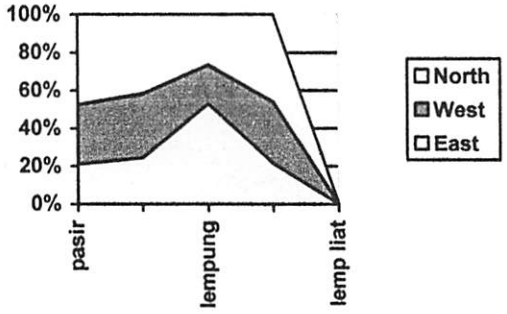
Merupakan kesimpulan hasil uji alat secara kesuluruhan dan saran – saran yang memungkinkan untuk pengembangan alat lebih lanjut.

BAB II

TINJAUAN PUSTAKA

2.1. Curah Hujan dan Kelembaban Tanah

Indikator kelembaban tanah adalah kadar air tanah yang dinyatakan dalam persen berat atau persen volume, mm air setiap m kedalaman tanah, dan tegangan air tanah (atm atau bar). Dalam praktek dikenal tiga tingkat kadar air tanah, yitu jenuh air, kapasitas lapang, dan titik layu. Kadar air tanah pada batas antara kapasitas lapang dan titik layu disebut air tersedia bagi tanaman. Akan tetapi karena kemampuan tanah memegang air tergantung pada tekstur tanah, maka kadar air tanah pada kapasitas lapang dan titik layu berbeda menurut tekstur tanah dan demikian pula air tersedia. Hubungan antara status air tanah dan tekstur tanah diilustrasikan dalam gambar 2.1. Tingkat ketersediaan air tanah tertinggi diberikan oleh tanah bertekstur halus,terutama tanah lempung debu. Makin tinggi tegangan air tanah makin rendah air tersedia pada semua tekstur tanah. Pada tegangan 15 atm (titik layu),air tanah tidak tersedia lagi bagi tanaman.



Gambar 2.1. Hubungan Antara Kadar Air Tanah Dengan Tekstur Tanah

*Sumber : Sadikin Somaatmadja, *Kedelai*, Bogor, 1995.

2.1.1. Hubungan Tanah-Air-Tanaman

Teori continuum menganggap tanaman sebagai suatu kapiler tempat air lewat, mulai dari tanah,akar,batang dan selanjutnya menguap ke udara di permukaan daun (transpirasi). Dengan teori ini Slatyer (30) memformulasikan factor-faktor yang menentukan besarnya transpirasi (T) dalam persamaan :

$$T = \frac{\Delta C}{\sum r} \approx (0{,}622\, \rho\, ud/P \times (\frac{\Delta e}{\sum r} \qquad (2\text{-}1)$$

Di mana:

ΔC = selisih konsentrasi uap air di udara dan di daun;

Σr = jumlah tahanan terhadap difusi uap air dari daun ke udara;

Δe = selisih tekanan uap air di udara dan daun;

$0{,}622\rho\ \mu d/\rho$ = factor konversi dari ΔC ke Δe;

$\rho\ \mu d$ = kerapatan udara;

ρ = tekanan udara.

Dalam keadaan kering, Ae menjadi besar dan transpirasi meningkat sampai batas dimana laju transmisi air tanah ke akar tidak dapat menandingi laju transpirasi. Dalam keadaan demikian stomata tertutup, r menjadi besar dan T akan terhambat. Di lapangan tanaman tumbuh dalam suatu komunitas yang saling mempengaruhi. Kehilangan air terjadi karena ada transpirasi dan evaporasi langsung dari permukaan tanah (evapotranspirasi). Dimana, subskripsi int berarti jalur difusi uap air dari komunitas tanaman. Dalam keadaan tanahnya basah sampai permukaan,Cint atau eint berturut-turut adalah sama dengan C atau e di permukaan tanah; rint dianggap sangat kecil,sebab itu diabaikan. Dalam keadaan

kering yang berkepanjangan,bidang penguapan (evaporation surface) secara lambat laun turun ke lapisan-lapisan daun (leaf canopy) di bawah, dan bila stomata daun tertutup karena sangat kering, bidang penguapan berada dibawah permukaan tanah dan evapotranspirasi menjadi terhambat pula.

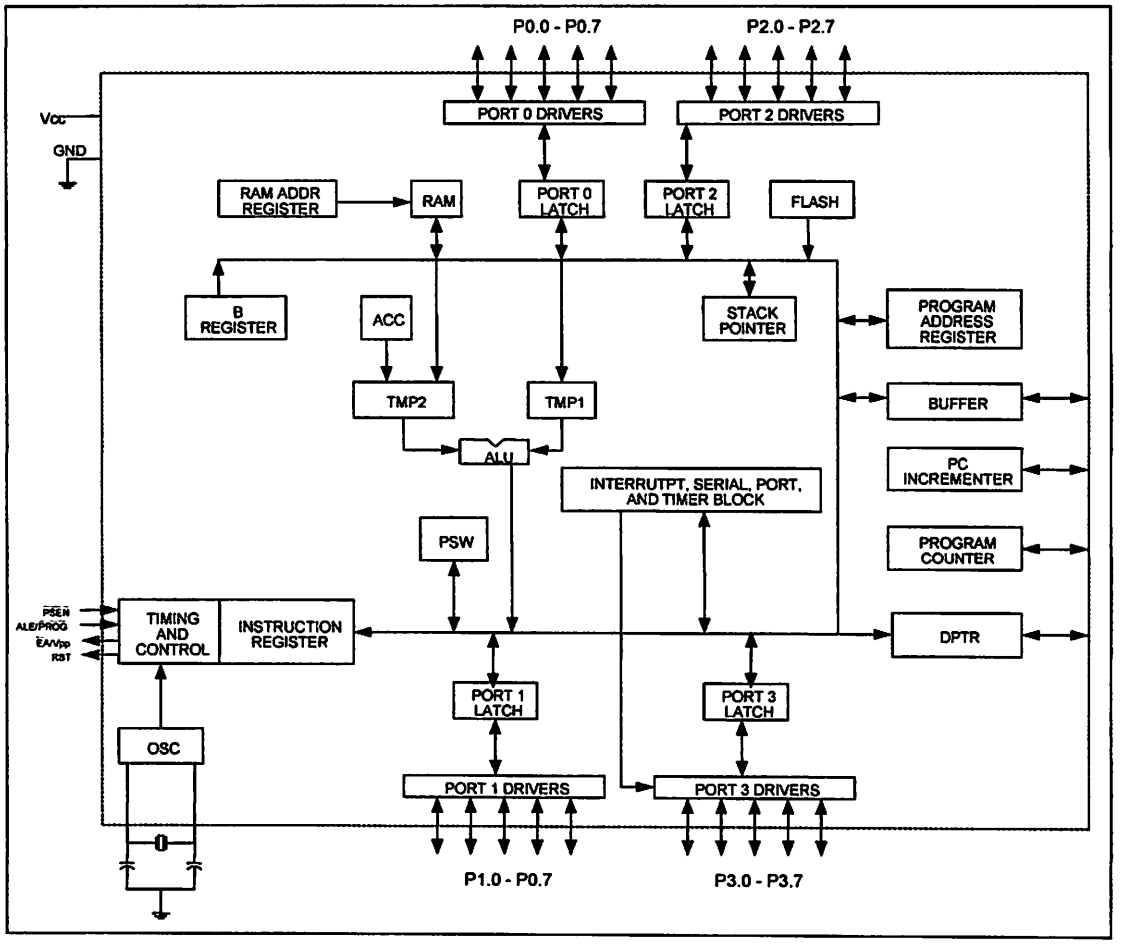
2.2. Mikrokontroler AT89S8252

2.2.1. Pendahuluan

Perbedaan mendasar antara mikrokontroler dan mikroprosesor adalah jika mikrokontroler selain memiliki CPU (*Central Processing Unit*) juga dilengkapi dengan memori dan I/O (*Input/Output*). Maka mikrokontroler dapat dikatakan sebagai *microcomputer* dalam keping tunggal (*single chip microcomputer*) yang dapat berdiri sendiri.

Mikrokontroler AT89S8252 adalah mikrokontroler keluarga MCS-51 yang membutuhkan daya rendah, memiliki kemampuan yang tinggi, dan merupakan mikrokomputer 8 bit yang dilengkapi 8K *byte Flash* PEROM (*Programmable and Erasable Read Only Memory*) yaitu ROM yang dapat ditulis ulang atau dihapus menggnakan sebuah perangkat *programmer*. *Flash* PEROM dalam AT89S8252 menggunakan *Atmel's High-Density Non Volatile Technology* yang mempunyai kemampuan untuk ditulis ulang hingga 1000 kali dan berisikan perintah *standard* MCS-51. Selain itu juga dilengkapi RAM *internal* sebesar 256 *byte*. Dalam sistem Mikrokontroler terdapat dua hal yang mendasar, yaitu: perangkat keras dan perangkat lunak yang keduanya saling terkait dan mendukung

2.2.2. Perangkat Keras Mikrokontroler AT89S8252



Gambar 2.2. Blok Diagram Mikrokontroler AT89S8252
*Sumber : Paulus, 2003 : 2

Mikrokontroler AT89S8252 secara umum memiliki:

- CPU 8 bit
- 8Kbyte FLASH PEROM
- 2 Kbyte EEPROM *internal*
- *Memory* 256 x 8 bit *Internal* RAM
- 32 Port I/O *Lines*
- 3 *Timer* dan *Counter*16 bit
- 9 Sumber Interupsi
- SPI serial *interface*

- *Oscillator* dan *clock* 12 - 24 MHz
- *Programmable Watchdog Timer*
- *Programmable UART Serial Chanel*
- *Dual Data Pointer*
- *Power-Off Flag*

2.2.3. Arsitektur AT89S8252

Arsitektur Mikrokontroler AT89S8252 adalah sebagai berikut:

1). CPU (*Central Processing Unit*) 8 bit dengan *register* A (*Accumulator*) & B.

2). 16-bit *Program Counter* (PC) dan (*Data Pointer*) DTPR.

3). 8-bit *Program Status Word* (PSW).

4). 8-bit *Stack Pointer* (SP).

5). 8 K*byte* Flash PEROM *internal.*

6). 256 *byte internal* RAM.

- 4 *bank register*, masing-masing berisi 8 *register*
- 16 *byte* yang dapat dialamati pada bit level
- 208 byte *general purpose memory data*

7). 32 pin *input-output* tersusun atas P0-P3. masing-masing 8-bit.

8). 3 buah *timer* (T0 & T1) dengan masing-masing 16-bit t*imer/ counter.*

9). *Receiver/transmiter data* secara *serial full duplex*: *Serial buffer* (SBUF).

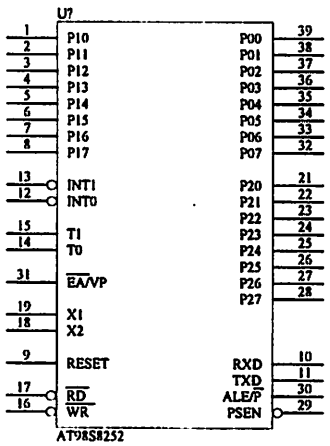
10). *Control register* : (TCON, TMOD, SCON, PCON, IP & IE)

11). 9 buah *sumber* interupsi (2 buah sumber interupsi *external* dan 3 buah sumber interupsi *internal*).

12). *Oscillator* dan c*lock internal.*

2.2.4. Konfigurasi Pin Mikrokontroler AT89S8252

Konfigurasi kaki-kaki mikrokontroler terdiri dari 40 pena, seperti terlihat pada gambar 2.3:



Gambar 2.3. Diagram Pin Mikrokontroler AT89S8252
*Sumber : Paulus, 2003 : 2

Fungsi dari tiap-tiap pena adalah sebagai berikut:

- **Pin_40: Vcc** (sumber tegangan).
- **Pin_20: GND** (*Ground*).
- **Pin_32-39: Port_0:**

  Merupakan port *input-output* dua arah multipleks dan dua *bus* alamat rendah (A0-A7) serta data selama pengaksesan program dan data memori *external*.

- **Pin_1-8: Port_1:**

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

- **Pin_21-28: Port_2:**

  Merupakan port *input-output* dengan *internal pull-up*. Mengeluarkan *address* tinggi selama pengambilan (*fetching*) *program memory external*. Selama pengaksesan ke *external data memory*, port 2 mengeluarkan isi P2 SFR

(*Special Function Register*). Menerima *address* dan beberapa sinyal kontrol selama pemprogramanan dan verifikasi.

➢ **Pin_10-17: Port_3:**

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

- RXD (P3.0) : Port *serial input.*
- TXD (P3.1) : Port *serial output.*
- INT0 (P3.2) : *external interrupt 0.*
- INT1 (P3.3) : *external interrupt 1.*
- T0 (P3.4) : *Input external timer 0.*
- T1 (P3.5) : *Input external timer 1.*
- WR (P3.6) : *Strobe* tulis data memori *external.*
- RD (P3.7) : *Strobe* baca data memori *external.*

➢ **Pin_9: RST *(Reset)*:**

Input Reset berfungsi me-*reset* CPU saat sumber tegangan dihidupkan.

➢ **Pin_30: ALE/PROG (*Address Latch Enable)/Programming*:**

Pulsa output ALE digunakan untuk proses *'latching' byte address* rendah (A0-A7) selama pengaksesan ke memori *external.* Pin ini juga digunakan untuk memasukkan pulsa program selama pemprogramanan.

➢ **Pin_29: PSEN (*Program Store Enable*):**

Merupakan *strobe* baca ke program memori *external.*

➢ **Pin_31: EA/VPP (*External Address*):**

EA *low* jika mengakses memori *external.* Untuk mengakses memori *internal* maka EA dihubungkan ke-Vcc (+5V).

➢ **Pin_18-19: XTAL1 dan XTAL2:**

Kaki ini dihubungkan dengan kristal bila menggunakan *Oscillator internal.* XTAL1 merupakan *input inverting Oscillator amplifier* sedangkan XTAL2 merupakan *output inverting Oscillator amplifier.*

2.2.5. Organisasi Memori

Dalam mikrokontroler AT89S8252 ruang alamat telah dibedakan untuk program memori dan data memori.

2.2.5.1.*Internal Program Memory*

Mikrokontroler AT89S8252 memiliki program memori *internal* sebesar 4Kbyte dengan ruang alamat 0000H-0FA0H. Jika alamat-alamat program lebih tinggi dari pada 0FA0H dimana melebihi kapasitas ROM *internal*, menyebabkan AT89S8252 secara otomatis mengambil kode byte dari program memori *external.* Kode *byte* juga dapat diambil hanya dari memori *external* dengan alamat 0000H-FFFFH dengan cara menghubungkan pin EA ke *ground.*

2.2.5.2.*Random Acsses Memory* (RAM)

Ruangan alamat memory data *internal* (RAM) dengan kapasitas 256 *byte* yaitu: 00H-FFH yang terbagi atas 3 daerah, yaitu:

1. Empat *bank register*

   Setiap *bank* terdiri dari 8 *register* (R0-R7) sehingga jumlah *register* untuk ke-empat *bank register* menjadi 32 buah *register* yang menempati ruang alamat 00H-1FH. Mengaktifkan salah satu *bank register* dapat dilakukan dengan mengatur RS0-RS1 pada *Program Status Word* (PSW).

2. Bit Addressable

Terdiri dari *16 byte* yang berada pada alamat 20H-2FH. Masing-masing bit dalam 128 bit yang lokasinya dapat dialamati secara langsung.

3. RAM Keperluan Umum

Terdiri atas 208 *byte* yang menempati alamat 30H-FFH, dan dapat dialamati secara langsung maupun tak langsung dalam penggunaan untuk keperluan umum (*general purpose*).

Tabel 2.1
Pengaturan RS0-RS1 *Bank Register*

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

*Sumber : Paulus, 2003 : 3

2.2.5.3.SFR (*Special Function Register*)

Untuk operasi AT89S8252 yang menggunakan alamat internal RAM (00H-7FH). Beberapa dari *register-register* ini juga mampu dengan pengalamatan bit sehingga dapat dioperasikan serpetri yang ada pada RAM.

2.2.5.3.1. PSW (*Program Status Word*)

Register ini terletak di alamat D0H. Cara mendefenisikannya ditunjukkan pada tabel 2.2. berikut ini :

Tabel 2.2
Pengaturan RS0-RS1 *Bank Register*

Alamat	Data	Simbol	Posisi	Fungsi /Arti
D0H	D0	P	PSW.0	*Parity flag*
	D1	-	PSW.1	*Flag* didefinisikan oleh pemakai.
	D2	OV	PSW.2	*OverflowFlag*
	D3	RS0	PSW.3	Bit pemilih *bank register.*
	D4	RS1	PSW.4	Bit pemilih *bank register.*
	D5	F0	PSW.5	*Flag 0*
	D6	AC	PSW.6	*Auxiliary CarryFlag*
	D7	CY	PSW.7	*Carry Flag*

*Sumber : Paulus, 2003 : 12

2.2.5.3.2. PCON (*Power Control*)

Register ini terletak pada alamat 87H. Cara mendefenisikannya ditunjukkan pada tabel 2.3 berikut ini :

Tabel 2.3
Skema Medefinisikan PCON

Alamat	Data	Simbol	Fungsi /Arti
87H	D0	IDL	*Idle* mode bit
	D1	PD	*Power Down* bit
	D2	GF0	Bit *flag* serbaguna.
	D3	GF1	Bit *flag* serbaguna.
	D4	-	Tidak dipakai.
	D5	-	Tidak dipakai.
	D6	-	Tidak dipakai.
	D7	SMOD	Digunakan untuk menghasilkan *baudrate* dan SMOD_1, maka *baudrate* akan *double* baik mode 0,1,2 atau 3.

*Sumber : Paulus, 2003 : 17

2.2.5.4.Sistem Interupsi

Mikrokontroler AT89S8252 mempunyai 9 buah sumber interupsi yang dapat mengakibatkan permintaan interupsi, yaitu: INT0, INT1, T0, T1 port serial dan beberapa port lainnya. Saat terjadi interupsi mikrokontroler secara otomatis akan menuju ke *subrutin* pada alamat tersebut. Setelah interupsi *service* selesai dikerjakan, Mikrokontroler akan mengerjakan program semula. Sumber interupsi

external adalah INT0, INT1, dimana kedua interupsi *external* ini akan aktif pada transisi rendah selain itu juga ada T*imer/Counter 0,* T*imer/Counter 0* dan interupsi dari port serial (*receiver*). Interupsi serial dibangkitkan dengan melakukan operasi OR pada R1 dan T1. Tiap-tiap sumber interrupsi dapat di-*enable* atau di-*disable* secara *software*. Tingkat prioritas semua sumber interupsi dapat diprogram sendiri-sendiri dengan *set* atau *clear* bit pada SFRS IP (*Interrupt Priority*).

Tabel 2.4
Alamat Sumber Interupsi

Sumber Interupsi	Alamat Awal
Power On Reset	0000h
Interrupt luar 0 (INT 0)	0003h
Pewaktu/ pencacah 0 (T0)	000Bh
Interrupt luar 1 (INT 1)	0013h
Pewaktu/ pencacah 1 (T1)	001Bh
Port I/O *Serial*	0023h

*Sumber : Paulus, 2003 : 53

Register yang berperan dalam mengatur aktif tidaknya interupsi adalah *interrupt enable register*, susunan dari bit-bit beserta kegunaannya adalah:

Tabel 2.5
Kegunaan *Interrupt Enable Register*

Alamat	Data	Simbol	Posisi	Fungsi /Arti
A8H	D0	EX0	IE.0	Diatur secara *software* untuk interupsi dari INT1.
	D1	ET0	IE.1	Diatur secara *software* untuk interupsi dari *timer/counter* 1.
	D2	EX1	IE.2	Diatur secara *software* untuk interupsi dari INT1.
	D3	ET1	IE.3	Diatur secara *software* untuk interupsi dari *timer/counter* 1.
	D4	ES	IE.4	Untuk mengatur *enable* atau *disables* uatu interupsi R1/T1.
	D5	-	IE.5	Kosong
	D6	-	IE.6	Kosong
	D7	EA	IE.7	Jika diatur 0 maka semua interupsi di-*disable*, jika diatur 1 maka interupsi diatur di-*disable* atau di-*enable* menurut masing-*masing* bit.

*Sumber : Paulus, 2003 : 51

2.2.5.5.*Timer/Counter*

Pengendalian kerja dari *timer/counter* dilakukan dengan pengaturan register yang berhubungan dengan kerja dari *timer/counter* yaitu melalui sebuah

timer/counter mode control. Untuk mengaktifkan *timer/counter* yang meliputi penentuan fungsi sebagai *timer* atau sebagai *counter* serta pemilihan *mode* operasi dapat diatur melalui TMOD yang beralamat pada 89H. Konfigurasi dari *register* TMOD seperti yang ditunjukkan pada Tabel 2.6. berikut ini :

Tabel 2.6
Register TMOD

Alamat	Data	Simbol	Posisi	Fungsi /Arti
88H	D0	IT0	TCON.0	*Interrupt* 0 *type control bit.*
	D1	IE0	TCON.1	*External interrupt* 0 *edge flag.*
	D2	IT1	TCON.2	*Interrupt type* 1 *control bit.* Diatur oleh *software* untuk menentukan aktif *low* atau *high trigger* dari *external.*
	D3	IE1	TCON.3	*External interrupt* 1 *edge flag.* Diatur oleh *hardware* ketika *external interrupt* terdeteksi dan nol-kan melalui *software* ketika *interrupt* diproses.
	D4	TR0	TCON.4	*Timer* 0 *control bit.* Diatur oleh *software* ketika *timer/ counter 0.*
	D5	TF0	TCON.5	*Timer* 0 *overflow flag control bit.* Diatur oleh *software* ketika *timer/counter 0 oferflow.*
	D6	TR1	TCON.6	*Timer* 1 *control bit.* Diatur oleh *software* ketika *timer/ counter 0.*
	D7	TF1	TCON.7	*Timer* 1 *overflow flag control bit.* Diatur oleh *software* ketika *timer/counter* 0 *oferflow.*

*Sumber : Paulus, 2003 : 33

Tabel 2.7
Timer/Counter Mode Control Register

Alamat	Data	Simbol	Fungsi /Arti
89H	D0	*Timer* 0; M0 (0)	Untuk memilih mode *timer.*
	D1	*Timer* 0; M1 (0)	Untuk memilih mode *timer.*
	D2	*Timer* 0; C/T (0)	1 = *Counter* & 0 = *Timer*
	D3	*Timer* 0; GATE (0)	*Timer* akan berjalan jika bit di *set* dan INT0 (untuk *Timer* 0) atau INT1 (untuk *Timer* 1)
	D4	*Timer* 1; M0 (1)	Untuk memilih mode *timer.*
	D5	*Timer* 1; M1 (1)	Untuk memilih mode *timer.*
	D6	*Timer* 1; C/T (0)	1 = *Counter* & 0 = *Timer*
	D7	*Timer* 1; GATE (1)	*Timer* akan berjalan jika bit di *set* dan INT0 (untuk *Timer* 0) atau INT1 (untuk *Timer* 1)

*Sumber : Paulus, 2003 : 32

Tabel 2.8
Mode Operasi *Timer/Counter*

M1	M0	*Operating Mode*
0	0	*Timer* 13 bit
0	1	*Timer/Counter* 16 bit
1	0	8 bit *Auto reload Timer /Counter*
1	1	TL0 dari *Timer* adalah 8 Bit *Timer/Counter* dikendalikan oleh kontrol bit *Timer* 0. TH0 adalah 8 bit yang dikendalikan oleh *Timer* 1 *control* bit.

*Sumber : Paulus, 2003 : 33

2.2.5.6.Metode Pengalamatan

1. Pengalamatan bit (*Direct Bit Addressing*) :

   Pengalamatan langsung tiap bit ini hanya dilakukan pada lokasi RAM *internal* yaitu 20H-2FH, dan sebagian SFR yaitu: port 0, port 1, port 2, port 3, TCON *register*, SCON *register*, IE *register*, PSW *register*, ACC dan B *register*.

2. Pengalamatan tak langsung (*Indirect Bit Addressing*):

   Pada *pengalamatan* tak langsung, instruksi menunjukkan suatu *register* yang isinya adalah alamat dari *operand, eksternal* dan *internal* RAM dapat dialamati secara tidak langsung. *Register* alamat untuk data dengan lebar 8 bit dapat berupa R0 dan R1 yang digunakan untuk memilih angka *register* atau *stack pointer*. *Register* alamat untuk data, dengan lebar 16 bit digunakan *Data Pointer* (DPTR).

3. Pengalamatan ber-indeks :

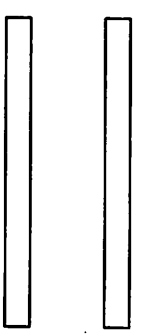
   Yang dapat diakses dengan pengalamatan berindeks hanya *memory program*. Mode ini dimaksudkan untuk membaca *look-up table program.*

4. Konstanta *immediet* :

   Pengalamatan langsung dilakukan dengan memberikan nilai ke *register* secara langsung, dilakukan dengan menggunakan tanda #,Contoh: Mov A, #100

2.3. Sensor Air (Probe)

Sensor ini berfungsi untuk mendeteksi ada tidaknya air tanah yang mengenai sensor. Sensor ini berupa probe yang diletakkan pada posisi yang berbeda dan digunakan sebagai referensi bagi input rangkaian untuk menggerakkan motor menutup atap. Besarnya resistansi probe ketika ada air adalah sekitar 10 KΩ, dan ketika tidak ada air yang mengenainya (kering) maka besar resistansinya sangat besar. Sensor air dibentuk (disusun) dari dua logam baja stenles steel dengan masing-masing memiliki ukuran panjang 5 cm, lebar 1 cm, tebal 1mm, jarak kedua logam 1,5cm dan didesain sedemikian rupa sehingga jika ada air tanah diantara kedua logam ini, maka akan menyebabkan perbedaan resistansi antar kedua logam baja tersebut. Perbedaan resistansi ini akan dikondisikan dengan rangkaian pengkondisi sinyal sehingga akan dapat digunakan untuk mendeteksi tingkat kelembaban air tanah. Gambar 2.4 dibawah ini merupakan gambar bentuk fisik sensor tersebut.



Gambar 2.4
Ukuran dan Bentuk Fisik Sensor Kelembaban Tanah
*Sumber : Perancangan

Untuk merancang sensor ini dapat dengan mudah digunakan rangkaian penguat level tegangan *output* mendeteksi objek. Rangkaian dapat dibentuk dari rangkaian (Op-Amp) dengan pengutan yang dapat diatur.

2.4. Solenoid

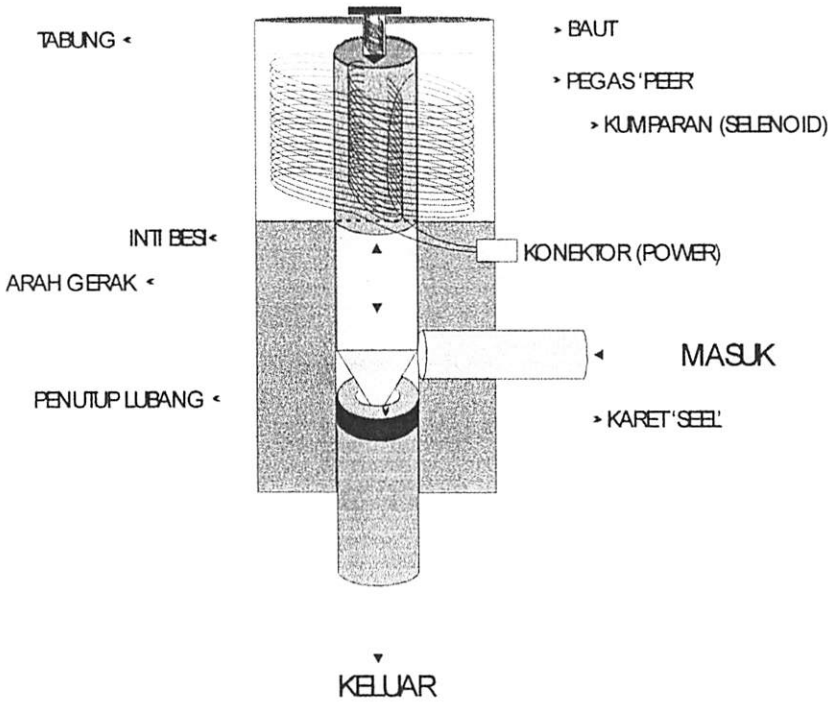
Prinsip dasar dari solenoid adalah adalah kumparan bentuk melingkar yang dialiri arus listrik, dimana bila kumparan tersebut dialiri arus I dengan jumlah lilitan n dan dengan tegangan sebesar V akan menimbulkan medan listrik di dalam kumparan tersebut. Arah gerak medan magnet tersebut ditentukan dengan kaidah tangan kiri yang berbunyi dimana arah arus listrik I tegak lurus terhadap medan listrik. Kuat medan magnet dapat ditunjukkan pada rumus :

$$H = \frac{N.i}{\ell} \qquad (2\text{-}2)$$

Dimana :

H : Kuat medan (Ampere/meter)
i : Arus listrik (A)
ℓ : Panjang jalur (m)
N : Jumlah lilitan

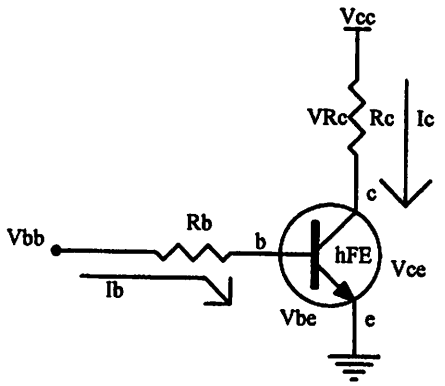
Gambar selenoid valve ditunjukkan pada gambar dibawah ini:



Gambar 2.5. Selenoid Valve

2.5. Transistor

Transistor merupakan salah satu dari komponen aktif yang sering sekali digunakan dalam setiap rangkaian elektronika. Beberapa fungsi transistor yaitu dapat digunakan sebagai rangkaian *driver* ataupun sebagai saklar. Rangkaian *driver* merupakan suatu rangkaian yang digunakan untuk menggerakkan suatu peralatan lain yang membutuhkan arus atau tegangan yang lebih besar. Dalam rangkaian *driver* ini kaki basis transistor dikontrol dengan memberi pulsa rendah dan pulsa tinggi. Rangkaian *driver* ditunjukkan dalam gambar 2.6:



Gambar 2.6. Transistor Sebagai Saklar
Sumber : Wasito S, *Vademekum*, 1984 : 91

Dalam gambar terlihat V_{bb} akan memberi tegangan maju kepada dioda emitor melelui resistor R_b. Untuk sebuah transistor *silikon* memiliki tegangan hambat pada V_{be} sebesar 0,6 Volt, sedangkan transistor *germanium* V_{be} 0,7 *Volt*. Arus basis dan kolektor ditentukan dengan persamaan:

$$I_c = \frac{V_{cc} - V_{ce}}{R_c} \ (Ampere) \qquad (2\text{-}3)$$

$$I_b = \frac{V_{bb} - V_{be}}{R_b} = \frac{I_c}{h_{fe}} \ (Ampere) \qquad (2\text{-}4)$$

Dimana:

V_{bb} : Tegangan bias basis (*Volt*)

V_{be} : Tegangan basis emitor (*Volt*)

V_{ce} : Tegangan kolektor emitor (*Volt*)

V_{cc} : Tegangan sumber (*Volt*)

R_c : Resistor pada kolektor (*Ohm*)

R_b : Resistor pada basis (*Ohm*)

I_c : Arus pada kolektor (*Ampere*)

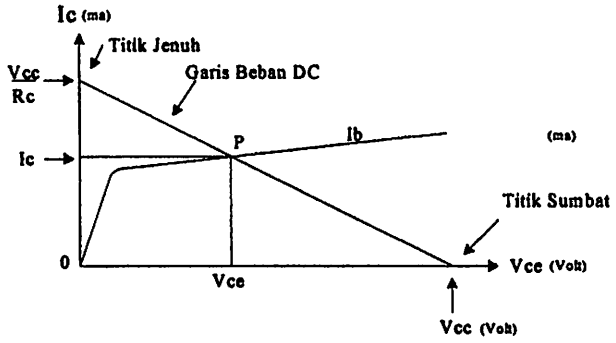
I_b : Arus pada basis (*Ampere*)

h_{fe} : Penguatan (kali/perkelipatan)

Arus kolektor yang mengalir pada resistor-nya akan menimbulkan tegangan sebesar I_c x R_c. Karena itu tegangan emitor kolektor menjadi:

$$V_{ce} = V_{cc} - I_c \cdot R_c \ (Volt) \qquad (2\text{-}5)$$

Hubungan antara I_c dan V_{ce} pada garis beban DC dari transistor pada hambatan beban tertentu ditunjukkan pada gambar 2.7:



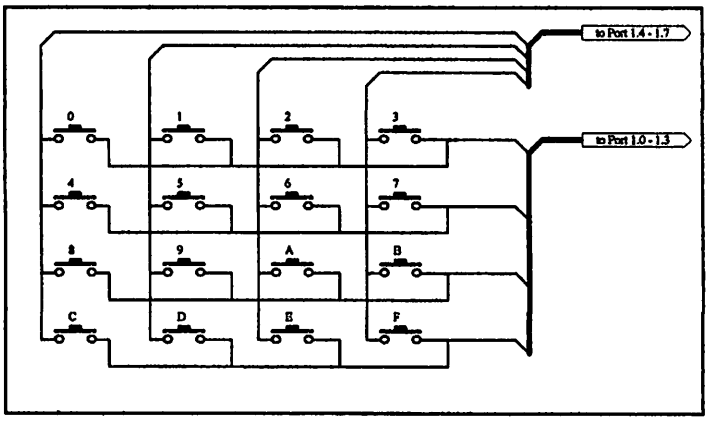
Gambar 2.7. Titik Potong Antara Garis Beban dan Arus Basis
Sumber : Wasito S, *Vademekum*, 1984 : 96

Garis beban DC tersebut mempunyai titik potong dengan sumbu tegak pada I_c di titik V_{cc} / R_c dan memotong sumbu V_{ce} pada titik V_{cc}. Titik P merupakan titik potong antara garis beban dan arus beban. Pada gambar diatas tampak bahwa titik potong antara garis beban DC dan kurva I_B pada daerah 0 atau pada sumbu V_{cc} dikenal sebagai titik *cut off* (sumbat), pada keadaan ini V_{ce}=V_{cc}. Sedangkan untuk titik jenuh adalah titik potong kurva I_B pada ujung teratas pada garis beban DC pada V_{ce} = 0 atau pada sumbu I_c.

2.6. Keypad

Teknik paling populer yang digunakan dalam perancangan *keypad* heksadesimal adalah teknik multiplekser, yaitu: pergantian atau pergeseran suatu

nilai bit dari bit tinggi menuju bit rendah atau sebaliknya dari bit rendah menuju bit tinggi, dengan empat buah jalur baris dan empat jalur kolom. Bila baris dan kolom ini disilangkan, maka akan terbentuk titik potong yang membentuk matrik 4 x 4, seperti ditunjukkan dalam gambar 2.8 berikut ini :



Gambar 2.8. *Keypad* Matrik 4 x 4
*Sumber : Paulus, 2003 : 132

Rangkaian ini dapat dianalogikan dengan tiga buah kabel terbuka dengan empat buah kabel lainnya. Bila pada suatu titik kabel yang bersilangan itu disentuhkan, maka diasumsikan bahwa tombol *keypad* pada posisi yang bersilangan tersebut ditekan. Penekanan *keypad* tersebut akan membentuk koordinat (x,y) dalam dua dimensi. Informasi posisi yang diperlukan adalah informasi tentang nilai x dan y, dalam pemrograman, nilai 'x dan y' ini dapat diperoleh melalui *scaning.* Rangkaian *keypad* ini terdiri dari 16 buah saklar *push button* yang dirangkai dalam bentuk matriks (kolom dan baris). Jalur kolom dan baris dari *keypad* tersebut nantinya akan dihubungkan pada port mikrokontroler.

2.7. Penguat Operasional

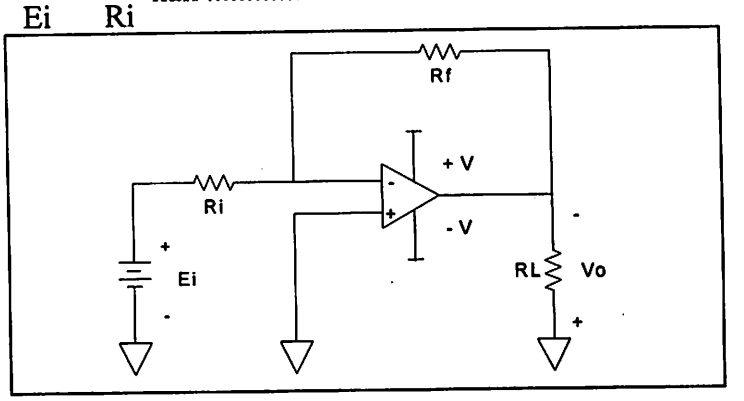
2.7.1. Penguat Pembalik

Penguat pembalik (*inverting*) adalah sebuah penguat yang tegangan keluaranya (Vo) mempunyai keluarannya yang tidak sama seperti tegangan masukkan Ei , yang diperlihatkan pada gambar 2.9. Persaman tegangan keluaran dari penguat membalik adalah sebagai berikut :

$$Vo = - \frac{Rf}{Ri} . Ei \text{ Volt} \quad \text{..... (2.6)}$$

Sehingga diperoleh persamaan *gain* tegangannya:

$$Acl = \frac{Vo}{Ei} = \frac{Rf}{Ri} \text{ kali} \quad \text{..... (2.7)}$$



Gambar 2.9. Polaritas Tegangan Untuk Penguat Pembalik
*Sumber : Robert Collinc, 1998 : 102

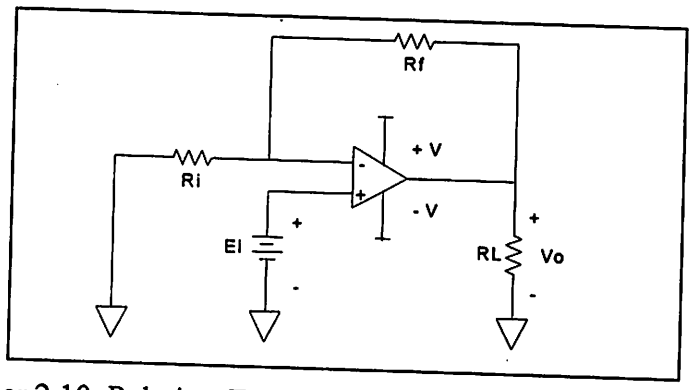
2.7.2. Penguat Tak Membalik

Penguat tak membalik *(Non Inverting)* adalah penguat yang tegangan keluaran Vo mempunyai polaritas yang sama seperti tegangan masukannya Ei, yang diperlihatkan gambar 2.10. Persamaan tegangan keluaran dari penguat tak membalik adalah sebagai berikut :

$$\left[1 + \frac{Rf}{Ri}\right]. Ei = Vo \quad \text{..... (2.8)}$$

Sehinga dapat di peroleh persaman *gain* tegangannya :

$$Acl = \frac{Vo}{Ei} \quad \text{..... (2.9)}$$



Gambar 2.10. Polaritas Tegangan Untuk Penguat Tak Membalik
*Sumber : Robert Collinc, 1998 :103

2.8.*Analog to Digital Converter* (ADC)

Sistem kontrol hanya dapat mengolah atau memproses data dalam bentuk *biner* atau lebih disebut digital saja, oleh sebab itu data *analog* yang akan diproses oleh harus diubah terlebih dahulu ke dalam bentuk kode-kode *biner* (*digital*). Jadi untuk menghubungkan sistem *analog* yang ada di luar sistem ke dalam kontrol, dibutuhkan suatu pengubah atau *analog to digital converter.*

Fungsi dasar dari pengubah *analog* ke *digital* adalah pengubah tegangan *analog* kedalam bentuk kode–kode *biner* (*digital*) sehingga dapat diproses oleh kontrol. Tegangan analog yang merupakan masukan dari ADC berasal dari pengkondisi arus, rangkaian ini mempunyai range tertentu dan disebut *analog.* Kode *biner* hasil konversi ini dipakai sebagai data untuk diolah kontrol yang kemudian ditampilkan melalui LCD .

Pengubah *analog* ke *digital* merupakan jantung dari sistem data akuisisi yang berfungsi mengubah data dalam bentuk kontinyu ke dalam *digital* yang diskrit, sehingga cocok untuk diproses oleh mikrokontroler. ADC lebih banyak variasi bentuknya bila dibandingkan dengan DAC. Karena ADC karakteristik yang dibutuhkan jauh lebih banyak. Adapun jenis ADC antara lain :

- *Single Ramp Integrating.*
- *Dual Slope Integrating .*
- *Tracking (Couenter Comparator).*
- *Single Counter.*
- *Successive Approximations.*
- *Multi Comparator (Flash).*

Ada beberapa faktor yang perlu diperhatikan dalam pemilihan komponen ADC, antara lain :

- *Resolution*

  Merupakan spesifikasi terpenting untuk ADC, Yaitu jumlah langkah dari sinyal skala penuh yang dibagi dan juga ukuran dan langkah–langkah. Boleh juga dinyatakan dalam bit yang ada dalam satu *word.* Ukuran LSB (langkah terkecil) sebagai persen dari skala penuh atau dapat juga LSB dalam mV (untuk skala penuh yang diberikan).

- *Accuracy* (Ketelitian)

  Adalah jumlah dari semua kesalahan, misalnya kesalahan *nonlinier*, skala penuh, skala nol dan lain–lain. Dapat juga menyatakan perbedaan antara tegangan *input analog* secara teoritis yang dibutuhkan untuk

menghasilkan suatu kode *biner* tertentu terhadap tegangan *input* nyata yang menghasilkan tegangan kode *biner* tersebut.

- Waktu Konversi

  Waktu yang dibutuhkan untuk mengkonversikan *analog* ke *digital* setiap sampel atau waktu yang diperlukan untuk menyelesaikan suatu konversi.

Disini tidak akan dijelaskan keseluruhan dari jenis ADC, tetapi hanya dititik-beratkan pada ADC jenis *successive approximation,* karena dalam perencanaan pembuatan alat digunakan ADC jenis ini. *Successive approximation* merupakan golongan ADC medium. ADC tipe ini dapat dikatakan merupakan perpaduan yang baik antara kecepatan (kecepatan menengah) dan tingkat kerumitan rangkaian (menengah). Waktu Konversi ADC ini selalu tetap, tidak tergantung pada besarnya sinyal analog (*input*), tetapnya waktu konversi ini merupakan salah satu keuntungan penggunaan ADC tipe ini.

2.8.1. *Successive Approximation*

ADC jenis ini dibentuk dari beberapa komponen DAC (*Digital to Analog Converter*), *Comparator* dan beberapa *register* serta rangkaian kontrol. Teknik konversi yang dipakai adalah dengan melakukan perbandingan antara tegangan *input* yang tidak diketahui dengan tegangan *input* yang dihasilkan oleh DAC. DAC diatur oleh *shift register* dan rangkaian logika untuk kontrol dan hasil konversi dapat diambil dari *register* keluaran. Konversi dimulai dengan memberikan sinyal *input start convertion.*

Dengan sinyal *start convertion* tersebut mengakibatkan *input* DAC akan dibuat aktif sehingga MSB sama dengan 1 dan bit yang lain berharga 0. *Output*

DAC akan terjadi setengah *full scale* dibanding dengan *input analog* yang dikonversi. Dari *output comparator* dapat diketahui mana yang lebih besar, bila tegangan *output* DAC lebih besar maka bit MSB tetap, sedang bila *output* DAC lebih kecil maka MSB berubah menjadi 0. Kemudian bit berikutnya (bit lebih rendah dari MSB) dibuat sama dengan 1 dengan keadaan MSB tetap seperti proses perbandingan pertama tadi.

Dari keadaan *input digital* ini akan dihasilkan *output analog* (dari DAC) yang juga akan dibandingkan lagi dengan tegangan *input* untuk menentukan apakah keadaan bit kedua akan diubah atau tidak. Proses diatas berlangsung sampai kondisi LSB, sehingga diperoleh *output* dari DAC yang terakhir. Proses pergeseran bit dari MSB sampai LSB dilakukan oleh *shift register* yang waktunya diatur oleh pulsa *clock.* Jika proses konversi tersebut telah selesai dari setiap input, maka *output* status bit pada pin 7 tersebut akan memberitahukan melalui "*End of Convertion*" yang menyatakan bahwa *output digital* ADC dapat diambil sebagai besaran *digital* yang mewakili *input analog* yang dikonversikan

2.8.2. *Analog To Digital Converter* 0809

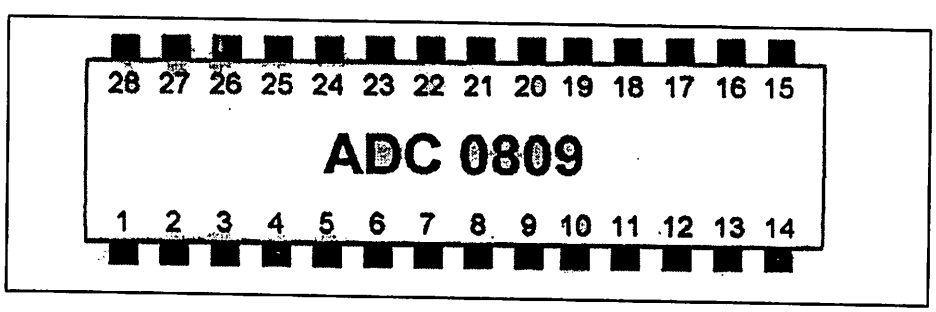
Analog To Digital Convertion 0809 dikemas dalam IC *monolitik*, ADC ini menggunakan metode pengubah berturut–turut *(successive Approximation)* dengan 8 saluran *input analog* dan dilengkapi dengan beberapa kontrol logika yang sesuai dengan *standart microprocessor.* Blok diagram ADC 0809 memperlihatkan bahwa ADC 0809 terdiri dari 3 (tiga) bagian utama yaitu :

- MUX *Analog* 8 *Channel.*
- *Analog to Digital Converter* ADC 8 bit.

- *Three – State Latch Buffer* (penyangga terkunci 3 kanal).

Multiplexer analog 8 *chanel* berfungsi untuk mengolah 8 *input* data *analog* secara bergantian. Untuk memilih *input* mana yang dikehendaki pada *output multiplekser* disediakan 3 bit kontrol pemilih saluran *input* tersedia. *Analog to Digital Converter* (ADC) merupakan *successive approximation register* 8 bit yang terdiri dari komparator, SAR dan *Clock Three – State Latch Buffer* berfungsi untuk menampung keluaran ADC 8 bit.

ADC ini merupakan *level* terbatas untuk sinyal *input*-nya yang tergantung pada *input* referensi yang digunakan. Dalam hal ini yang membatasi adalah tegangan referensi *positif* Vref (+) dan tegangan *negatif* Vref (-). Jika *input* referensi diberi tegangan *negatif* dan *input* referensi *positif* diberi tegangan *positif* sehingga tegangan *input analog* dapat berharga diantara kedua tegangan referensi tadi dan ini disebut *bipolar*, artinya kedua *input*-nya mempunyai polaritas *positif* dan *negatif*. Jika *input* referensi *positif* diberi tegangan *positif* dan referensi *negatif* diberi tegangan 0 Volt, maka dapat disebut *unipolar* karena *analog*-nya berada diatas 0 Volt (berharga *positif* saja). ADC yang dipakai dalam sistem ini dirancang untuk jenis *unipolar*. Gambar struktur pin ADC 0809 sebagai berikut:



Gambar 2.11. Struktur Pin IC ADC 0809
*Sumber : *Anonimous, National Data Sheet, 1995 : 2/56*

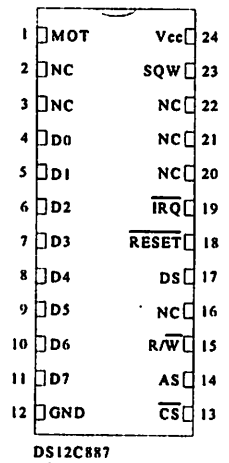
Tabel 2.9
Keterangan Pin IC ADC 0809

Pin	Nama	Keterangan
1	IN_3	Masukan 3
2	IN_4	Masukan 4
3	IN_5	Masukan 5
4	IN_6	Masukan 6
5	IN_7	Masukan 7
6	*Start*	Mulai Konversi
7	EOC	*End of Convertion*
8	2^{-5}	*Output* 5
9	*Output Enable*	Menghidupkan *Output*
10	Clock	*Clock*
11	Vcc	*Supply* tegangan
12	Vref (+)	Tegangan referensi (+)
13	GND	*Ground*
14	2^{-7}	*Output* 7
15	2^{-6}	*Output* 6
16	Vref (-)	Tegangan referensi (-)
17	2^{-8}_{LSB}	*Output* 0
18	2^{-4}	*Output* 4
19	2^{-3}	*Output* 3
20	2^{-2}	*Output* 2
21	2^{-1}_{MSB}	*Output* 1
22	ALE	*Address Latch Enable*
23	ADD_C	*Control_A*
24	ADD_B	*Control_B*
25	ADD_A	*Control_C*
26	IN_0	Masukan 0
27	IN_1	Masukan 1
28	IN_2	Masukan 2

*Sumber : *Anonimous, NATIONAL DATA SHEET, 1995 : 2/56*

2.9.RTC (*Real Time Clock*) IC DS 12C887

DS 12C887 adalah serpih yang dipakai sebagai basis pewaktuan buatan *dallas Semiconductor*. Terdiri atas 24 pin dengan konfigurasi penyemat seperti ditunjukkan dalam gambar 2.12 berikut ini :



Gambar 2.12. Konfigurasi Pin IC DS 12C887
*Sumber : Paulus, 2003 : 142

Arsitektur dari IC RTC tipe DS 12C887 antara lain adalah:

- *Oscillator internal* dan *time base internal.*
- Menghitung detik, menit, jam dalam hari.
- Menghitung hari dalam setiap minggu, tanggal, bulan dan tahun.
- Seratus tahun kalender (1 abad) samapai tahun 2100.
- Mempunyai catu daya *back-up.*
- Dapat beroperasi lebih dari 10 tahun tanpa kehadiran catu daya eksternal
- RAM internal sebesar 64 byte, terdiri atas 14 byte untuk *clock* dan *register* kontrol, sedangkan 50 *byte* lainya dapat digunakan oleh pemakai (umum).

Fungsi tiap-tiap pin IC RTC DS 12C887 adalah sebagai berikut :

➢ **Pin_1: MOT (*Motel*):**

Digunakan sebagai saklar pemilih mode. Apabila dihubungkan dengan Vcc, pewaktuan *Motorola* yang dipakai, jika dihubungkan dengan GND (*Ground*) berarti sistem diagram pewaktuan *Intel* yang dipakai.

➢ **Pin_2-3: NC (*No Conec*):**

Tidak terhubung.

➢ **Pin_3-11: AD0-AD7 (*Multiplexed Bidirectional Address/Data Bus*):**

Merupakan *bus* alamat dan data dua arah yang termultipleks.

➢ **Pin_12: GND (*Ground*):**

Merupakan penyemat catu daya yang dihubungkan dengan *ground.*

➢ **Pin_13: $\overline{CS}$ (*Chip Select Input*):**

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

➢ **Pin_14: AS (*Address* Strobe *Input*):**

Merupakan masukan bagi sinyal yang digunakan untuk memisahkan *bus* data dan *bus* alamat.

➢ **Pin_15: $R/\overline{W}$ (*Read/Write Input*):**

Penyemat $R/\overline{W}$ dapat diartikan sama seperti sinyal *Write Enable* (WE) yang dipakai dalam proses penulisan data memori jenis RAM.

➢ **Pin_16: NC (*No Conec*):**

Tidak terhubung.

➢ **Pin_17: $\overline{DS}$ (*Data Strobe or Read Input*):**

Pin DS diartikan sama seperti sinyal *Output Enable* (OE) dalam komponen memori untuk melakukan proses membaca data RAM internal RTC.

➢ **Pin_18: RESET (*Reset Input*):**

Sinyal RESET diberikan dengan memberikan logika rendah selama waktu yang di spesifikasikan.

➢ **Pin_19: IRQ (*Interrupt Request Output*):**

Sinyal IRQ merupakan sinyal aktif rendah, yang dapat digunakan untuk menginterupsi sistem kontrol. Saat tidak terdapat interupsi, pin ini dalam kondisi impedansi tinggi (*high-impedance*).

➢ **Pin_20-22: NC (*No Conec*):**

Tidak terhubung.

➢ **Pin_23: $SQ\overline{W}$ (*Square Wave Output*):**

Merupakan pin yang dapat dipilih untuk mengeluarkan satu dari 13 frekuensi keluaran yang tersedia. Besar frekuensi keluaran $SQ\overline{W}$ dapat diubah melalui pengaturan pada *register* "A". Untuk mengaktifkan dan me-nonaktifkan sinyal $SQ\overline{W}$ dipilih lewat bit $SQ\overline{W}E$ dalam *register* "B". Tabel 2.1 menunjukkan cara mengatur *register* "A" agar dihasilkan frekuensi dalam pin $SQ\overline{W}$.

➢ **Pin_24: Vcc:**

Merupakan pin catu daya yang dihubungkan dengan sumber sebesar 5Volt.

Tabel 2.10

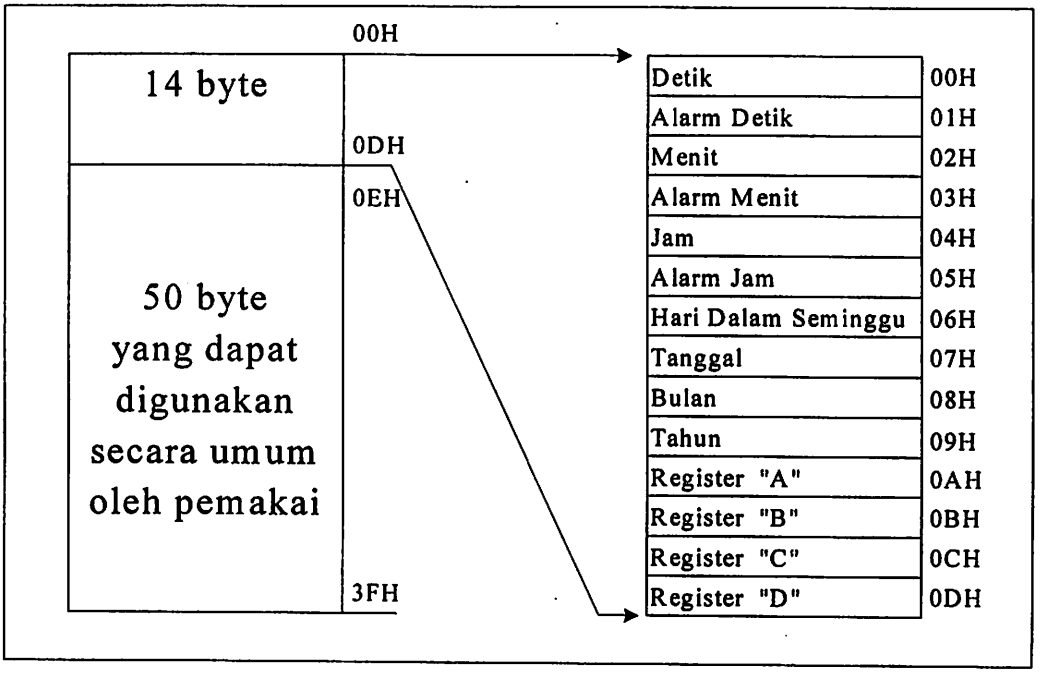
Perioda *Interrupt* dan Frekuensi Keluaran $\overline{SQW}$

SELECT BIT REGISTER A				*Tpi PERIODIC INTERRUPT RATE*	*SQW OUTPUT FREQUENCY*
RS3	RS2	RS1	RS0		
0	0	0	0	*None*	*None*
0	0	0	1	30,0517 us	32.768 Hz
0	0	1	0	61,0352 us	16.384 Hz
0	0	1	1	122,070 us	8.129 Hz
0	1	0	0	244,141 us	4.096 Hz
0	1	0	1	488,2813 us	2.048 Hz
0	1	1	0	976,5625 us	1.024 Hz
0	1	1	1	1,953125 ms	512 Hz
1	0	0	0	3,90625 ms	256 Hz
1	0	0	1	7,8125 ms	128 Hz
1	0	1	0	15,625 ms	64 Hz
1	0	1	1	31,25 ms	32 Hz
1	1	0	0	62,5 ms	16 Hz
1	1	0	1	125 ms	8 Hz
1	1	1	0	250 ms	4 Hz
1	1	1	1	500 ms	2 Hz

*Sumber : Paulus, 2003 : 144

Untuk menginisialisasi IC RTC, yaitu mengatur beberapa *register* yang ada di dalam RAM IC RTC saat kali pertama diaktifkan. Sinyal WR (*write*) sistem kontrol dihubungkan dengan penyemat R/W IC RTC, setelah data yang diinginkan dalam alamat dianggap sesuai. Sedangkan untuk membaca data RAM *internal* IC RTC, digunakan sinyal RD (*read*) pada sistem kontrol yang dihubungkan dengan pin DS (*data strobe*) pada IC RTC. Operasi pembacaan dan penulisan RAM *internal* IC RTC, sama seperti operasi baca dan tulis saat menggunakan komponen memori jenis RAM umumnya.

Peta alamat RAM *internal* IC RTC tipe DS 12C887 ditunjukkan dalam gambar 2.13 berikut ini :



Gambar 2.13 Peta Alamat RAM *Internal Real Time Clock* DS 12C887
*Sumber : Paulus, 2003 : 143

2.10. LCD (*Liquid Crystal Display*) M1632

LCD (*Liquid Cristal Display*) adalah suatu jenis piranti *output* yang menggunakan daya rendah dengan pengontrol kontras dan kecerahan. Pengontrol utamanya dan karakter ada pada ROM (*Read Only Memory*) *generator* dan *display* data RAM (*Random Access Memory*) yang akan menghasilkan *extended key codes* (kode tombol/*keyboard* standart *international* dalam Hexsa) jika padanya diberikan inputan. Untuk dapat memfungsikan dengan baik maka perlu diperhatikan proses inisialisasi yang telah ditentukan oleh pabrik pembuatnya. *Timing* penginisialisasian sangat perlu dipertimbangkan, karena jika meleset sampai orde *mili secon*, maka dapat dipastikan LCD itu tidak dapat berfungsi.

Ada dua jenis *register* yang terdapat dalam LCD M1632 ini, yaitu *data register* dan *instruction register*. Dengan menggunakan pin RS (*Register Select*)

pada LCD, pemakaian kedua *register* dapat dipilih. Pemilihan *register* pada LCD ditunjukkan pada Tabel 2.9. berikut ini :

Tabel 2.11
Pemilihan *Register* Pada LCD M1632

Nama Sinyal	No. Terminal	I/O	Tujuan	Keterangan Sinyal
RS	4	Input	MPU	0 : *Instruction register* 1 : Data *register*

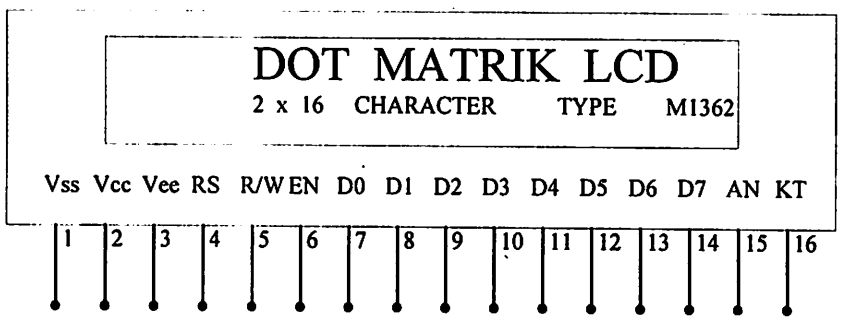
*Sumber : Seiko Data Sheet, 1987 : 2

Jika bagian yang dipilih adalah *instruction register* maka output yang dihasilkan adalah meliputi operasional dari LCD, misalnya fungsi *display clear, cursor home, entry mode set, display on/off, cursor shift* dan sejenisnya. Sebaliknya jika bagian yang dipilih adalah *data register*, *output* yang dihasilkan adalah meliputi karakter yang tabelnya terdapat pada lampiran.

Karakteristik LCD M1632 adalah :

- 16 karakter 2 baris dalam bentuk *dot matrik* 5 x 7 dan kursor.
- *Duty ratio* 1/16.
- Memiliki ROM pembangkit karakter untuk 192 jenis karakter.
- RAM untuk data *display* sebanyak 80 x 8 bit (80 karakter maksimum).
- Dapat dirangkai dengan MPU *(Microprocesor Unit)* 8 bit atau 4 bit.
- RAM data *display* dan RAM pembangkit karakter dibaca oleh *MPU.*
- Memiliki fungsi instruksi : *display ON/OFF, cursor ON/OFF, display character blink, cursor shift* dan *display shift.*
- Memiliki rangkaian *oscillator* sendiri.
- Sumber tegangan tunggal +5 Volt.
- Memiliki rangkaian *reset* otomatis pada saat catu daya dihidupkan.
- Temperatur operasi $0^{o} - 50^{o}$.

Deskripsi pin LCD ditunjukkan dalam gambar 2.14 berikut ini :



Gambar 2.14 Deskripsi Pin Pada LCD Tipe M1632
*Sumber : Seiko Data Sheet, 1987 : 2

BAB III

PERENCANAAN DAN PEMBUATAN ALAT

3.1. Pendahukuan

Perancangan dan pembuatan alat ini berdasarkan beberapa teori penunjang yang telah dijelaskan pada BAB II sebelumnya. Rangkaian yang akan dibahas meliputi rangkaian sensor kelembaban tanah, rangkaian penguat tegangan, rangkaian, konvereter *analog* ke *digital* (ADC), rangkaian *display* (LCD), rangkaian kontrol utama dengan mikrokontroler AT89S8252 dan beberapa driver selenoid sebagai pembuka kran air secara elektrik. Disamping itu juga dibahas perencanaan perangkat lunak dengan bahasa pemrograman assembler.

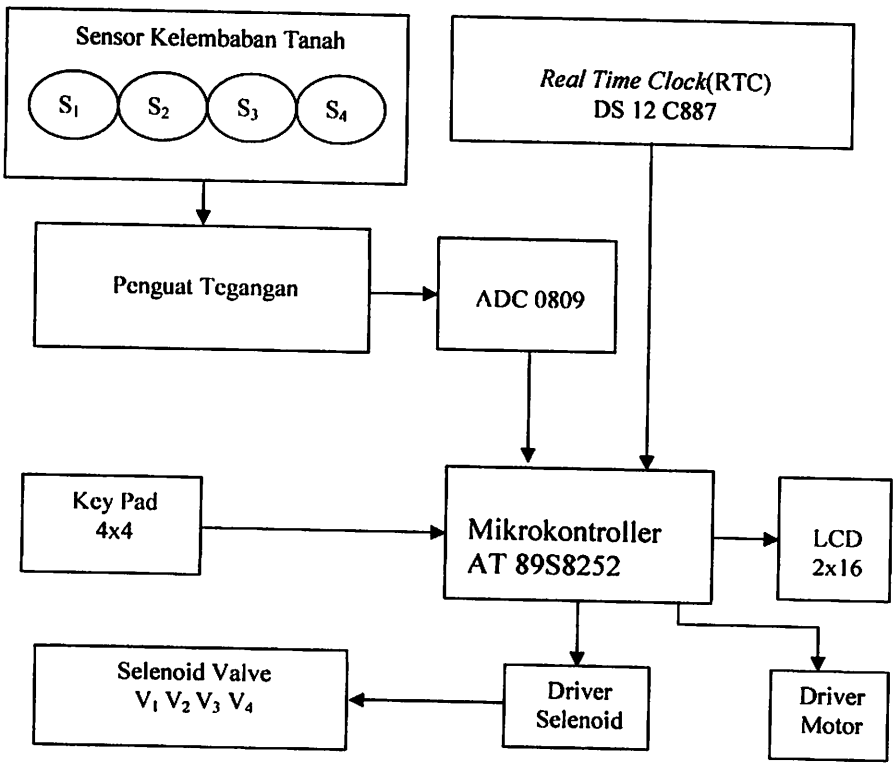
3.2. Perencanaan Perangkat Keras (*Hardware*)

Sebelum melakukan perancangan beberapa hardware, terlebih dahulu perlu adanya penyusunan blok diagram seperti ditunjukkan pada gambar 3.1. Untuk perancangan penyiraman otomatis pada tanaman kedelai berdasarkan kelembaban tanah ada beberapa fungsi dari setiap blok diagram yang ada pada gambar, antara lain:

- Sensor kelembaban tanah sebagai pendeteksi dari kelembaban tanah dengan satuan persentase dari kering hingga basah (0% - 99%).
- Rangkaian penguat tegangan yang difungsikan untuk menguatkan tegangan keluaran sensor agar dapat dibaca oleh rangkaian ADC dengan akurat.
- *Analog to Digital Convereter* (ADC) sebagai pengubah besaran *analog* dari sensor menjadi besaran *digital* agar dapat diolah oleh sistem kontrol.

- Blok *Real Time Clock* (RTC) sebagai generator pembangkit waktu yang akurat, sehingga dapat menunjukkan waktu penyiraman yang aman dan tidak berbahaya bagi daun tanaman kedelai.
- Blok kontrol yakni, yang mengendalikan hidup dan matinya kran penyiraman berdasarkan data yang diolah secara digital dari sensor kelembaban.
- Rangkaian *driver selenoid* sebagai pembuka kran air pada penyiraman tanaman kedelai secara otomatis (elektrik).
- *Display* LCD sebagai penampil data persentase dari kelembaban baik dari keypad maupun dari ADC (sensor) serta penampil data waktu dari IC RTC.

Berikut adalah gambar blok diagram sistem secara keseluruhan dari penyiraman otomatis pada tanaman kedelai.



Gambar 3.1. Blok Sistem Penyiraman Otomatis Pada Tanaman Kedelai
*Sumber : Perencanaan

3.2.1. Sensor Kelembaban Tanah

Sensor ini berfungsi untuk mendeteksi kosentrasi air dalam tanah yang menyatakan kelembaban dari suatu bidang tanah. Sensor berupa probe yang ditanam pada kedalaman tertentu (10cm-20cm). Sensor dalam perencanaan ini dibuat sebanyak empat buah, dan digunakan sebagai pengukur kondisi air tanah (kelembaban) untuk rangkaian kontrol dalam menggerakkan buka-tutup *selenoid valve* (kran listrik air).

Probe yang ditanam dalam tanah, saat mendeteksi kosentrasi kelembaban tanah tinggi memiliki resistansi sekitar ± 500 Ω, dan ketika tidak ada air (kering) maka besar resistansinya sangat besar (tak terhingga). Sensor ini dibentuk (disusun) dari dua lempeng baja yang tipis:

Data sensor :

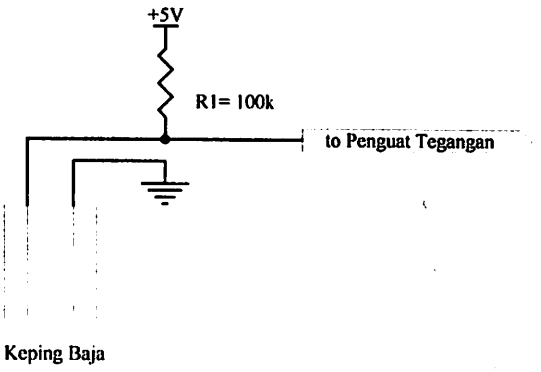
Panjang = 5cm

Lebar = 1cm

Tebal = 1mm

Jarak (*track*) = 1cm

Jika ada air tanah mengenahi probe (sensor), maka akan menyebabkan turunya nilai resistansi antara *track* (lempeng baja). *Output* dari sensor dimasukkan pada suatu rangkaian penguat tegangan, hal ini untuk menghasilkan tegangan keluaran yang sesuai dengan harga maksimal *input* ADC. Gambar 3.2 dibawah ini merupakan gambar elektrik sensor air dalam tanah (kelembaban tanah).



Gambar 3.2. Rangkaian Elektrik Sensor Kelembaban Tanah
*Sumber : Perencanaan

Rangkaian penedeteksi air tanah ini tidak berdiri sendiri tetapi masih dilengkapi reistor *pull-up* sebagai pengangkat tegangan saat tidak ada kelembaban. Besar nilai resirtor *pull-up* (R_1) ditentukan berdasarkan besar nilai arus maksimal yang mengalir, yaitu 0,05mA, dengan tegangan *output* maksimal 5*Volt*, maka R_1 adalah:

$$R_1 = \frac{Vcc}{I_{mak}} \qquad (3.1)$$

$$= \frac{5V}{0{,}5mA}$$

$$= 100k\Omega$$

Sedangkan pada saat sensor mendeteksi kelembaban tinggi, resistansi berkisar 500Ω dengan R_1 yang terpasang 100kΩ. Sehingga arus maksimal dan tegangan output maksimal dapat dicari sebagai berikut:

$$I_{mak} = \frac{Vcc}{R_{up} + R_{pb}} \qquad (3.2)$$

$$= \frac{5V}{100k + 0{,}5k}$$

$$= 0{,}049mA$$

Untuk V_{out} minimal:

$$V_{out} = \frac{R_{pb}}{R_{pb} + R_1} x Vcc \qquad (3.3)$$

$$= \frac{0{,}5k}{0{,}5k + 100k} x 5V = 0{,}024V$$

Dari perhitungan diatas didapat nilai V_{out} = 0,024V saat sensor mendeteksi kelembaban tinggi.

3.2.2. Rangkaian Penguat Tegangan Membalik (*Inverting*)

Dari *output* rangkaian sensor memberi pengertian terbalik, dengan kondisi dimana sesungguhnya jika kelembaban naik, maka prosentase akan naik demikian juga tegangan keluaran akan naik, namun tidak demikian untuk sensor diatas. Untuk menyeimbangkan pengertian diatas, maka keluaran tegangan sensor terlebih dahulu diumpankan pada rangkaian penguat membalik dengan penguatan (A=1) kali menggunakan IC Op-Amp tipe LM324. Dengan menentukan R_i = 100kΩ, maka dapat ditentukan nilai R_f sebagai berikut:

$$A = -\frac{R_f}{R_i} \qquad (3.4)$$

$$-R_f = 1 x 100k$$
$$= 100k\Omega$$

Sedangkan untuk membalik tegangan keluaran, dengan mengeser trimpot 50kΩ sampai pada tegangan (V_{ref}). Untuk menentukan nilai V_{ref} harus sebanding dengan besar nilai *output* sensor saat kelembaban rendah (kering), yaitu = $5 Volt$. Sehingga didapat V_{out} Op-Amp saat kelembaban kering adalah:

$$V_{out} = \Delta V \, x \, A \tag{3.5}$$

$$\Delta V = V^{+} - V^{-} \tag{3.6}$$

$$= 5V - 5V$$

$$= 0Volt$$

$$V_{out} = 0V \, x1$$

$$= 0Volt$$

Saat Kelembaban tinggi (tanah basah) adalah:

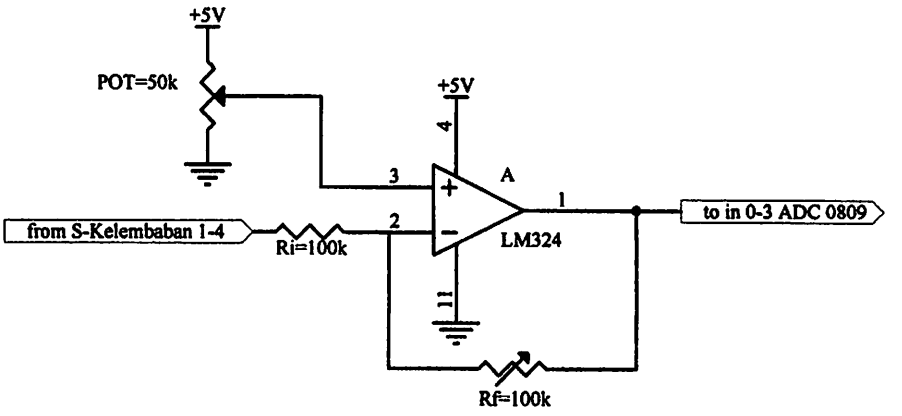
$$= 5V - 0{,}024V$$

$$= 4{,}976Volt$$

$$V_{out} = 4{,}976V \, x1$$

$$= 4{,}976Volt$$

Rangkaian penguat pembalik tegangan dari sensor kelembaban ditunjukkan pada gambar 3.3 dibawah ini:

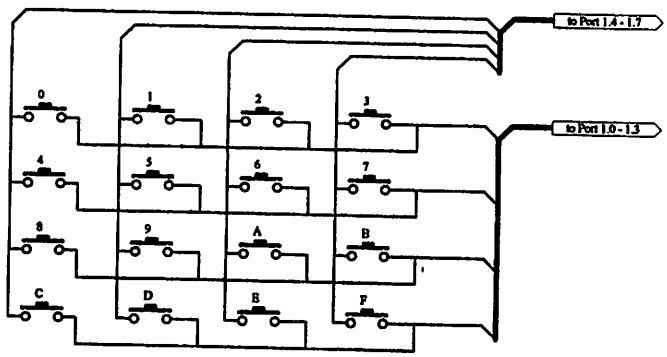


Gambar 3.3. Rangkaian Penguat Tegangan Membalik
*Sumber : Perencanaan

3.2.3. Rangkaian *Keypad* 4x4

Rangkaian *keypad* dalam perancangan menggunakan *keypad* 4 x 4 *push button*. Dari 12 tombol tersebut akan terbentuk karakter angka dan simbul (0-9 dan A-F). Teknik pembacaan pada *keypad* ini, yaitu model *scaning* 4 jalur baris

dan 4 jalur kolom. Bila baris dan kolom ini disilangkan maka akan terbentuk titik-titik potong yang membentuk matrik 4 x 4 seperti ditunjukkan pada gambar 3.4:



Gambar 3.4. Rangkaian *Keypad* Matrik 4 x 4
*Sumber : Perencanaan

Proses pembacaan *keypad* 4 x 4 model *scaning* sebagai berikut :

a) Menentukan urutan data karakter permanen *keypad* pada saat pembuatan *software*, missal (3-2-1-0-7-6-5-4-B-A-9-8-F-E-D-C). Dimana "3" yang pertama dan seterusnya "C" yang terakhir.

b) Kemudian menentukan data (*output*) untuk kolom dan baris, misal secara berurutan dari kolom teratas sampai baris paling bawah yaitu (111-1110).

c) Selanjutnya menentukan data pembanding yang ke-n, misal (0111-1110).

d) Setelah menentukan data pembanding yang ke-n, maka siap membaca data (*input*) dari penekanan tombol pada *keypad* untuk *scan* yang ke-n. Jika data *input* dan *output* sama, maka proses *scaning* dihentikan. Perlu diinggat proses dihentikan pada saat *scan* yang ke-n, maka data karakter yang diambil juga yang ke-n, misal nilai 'n =1' karakternya yaitu "3".

e) Bila proses *scaning* dihentikan atau sudah melakukan *scaning* sebanyak 16 kali, maka program keluar dari proses *scaning* menuju program selanjutnya

dalam *software*. Sebaliknya bila proses tidak dihentikan dikarenakan data *input* dan *output* tidak sama atau *scaning* belum 16 kali , maka proses menuju poin (6).

f) Jika pencocokkan antara data *input* dan *output* tidak sama, maka proses diteruskan dengan menentukan data pembanding yang ke-(n + 1), yaitu dengan cara menggeser ke kanan satu kali *logic* '0' pada kolom dan menetapkan data pada barisnya. Proses kembali pada poin (4).

g) Jika kolom sudah digeser 4 kali, maka diganti dengan menggeser satu kali ke kiri *logic* '0' pada baris dan mengembalikan data kolom pada posisi awal yaitu seperti poin 3 (0111). Sedangkan data pembanding dilanjutkan yang ke-(n + 1) dan proses kembali pada poin 4.

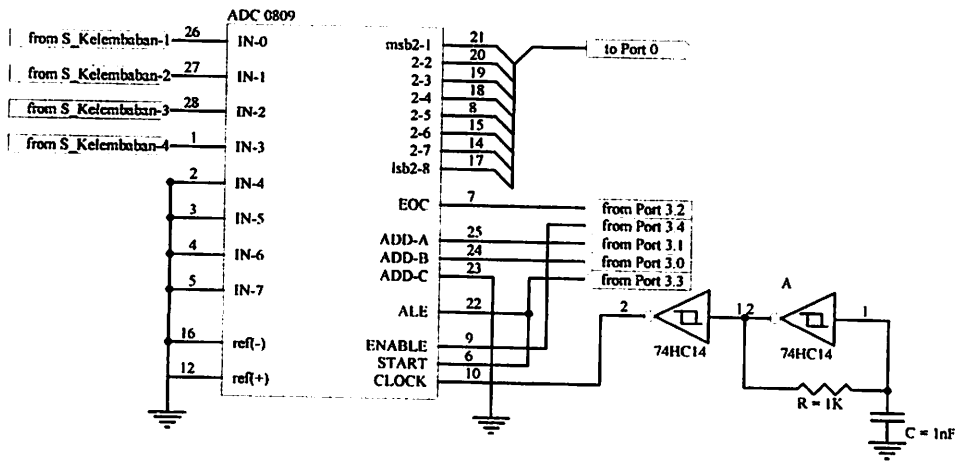
3.2.4. Rangkaian *Converter Analog to Digital* (ADC 0809)

Penggunaan rangkaian ADC 0809 yang merupakan ADC dengan sistem *Succesive Approximation*. ADC 0809 ini memiliki 8 buah *input*, tetapi dalam perancangan ini hanya menggunakan empat buah inputan yaitu D0-D3. Untuk menyeleksi empat inputannya terdapat tiga bit address yang ada pada ADC 0809 (A0, A1 & A2) dimana pin ini akan dihubungkan dengan mikrokontroller AT89S8252 untuk pengontrolannya.

Dikarenakan tegangan yang di inputkan 0-5 *Volt*, maka tingkat resolusi dari ADC 0809 diharapkan cukup kecil juga, sehingga penulis menggunakan Vref = 5*Volt*. Untuk membuat ADC 0809 dapat bekerja, maka perlu sebuah sumber

pulsa frekuensi sebesar 900Hz, yang dihasilkan dari rangkaian tangki pulsa (C & R). Rangkaian dan besar nilai "C & R" dapat di cari sebagai berkut:

$$\begin{aligned} F &= 1/1{,}1\ RC \\ 900Hz &= 1/(1{,}1 \times 1000 \times C) \\ C &= 10^{-9}\ F = 1nF \end{aligned} \qquad (3.7)$$



Gambar 3.5. Rangkaian ADC 0809
*Sumber : Perencanaan

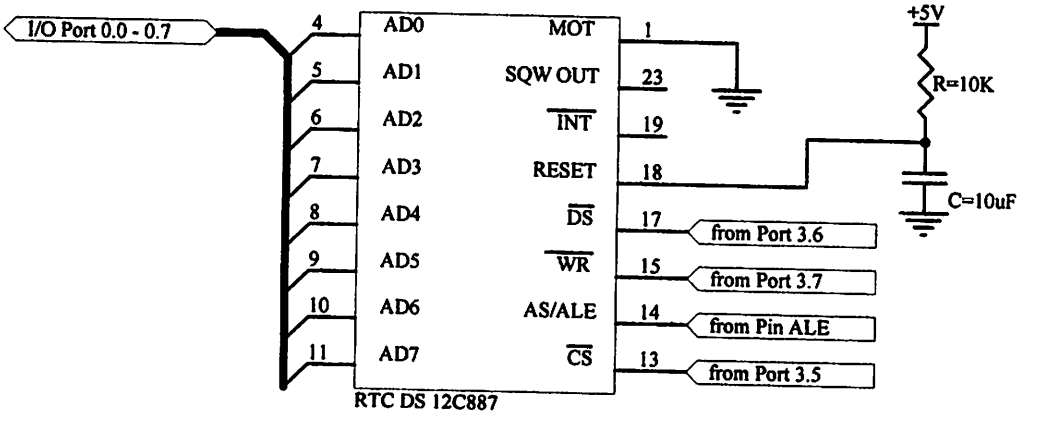
3.2.5. Rangkian *Timer* RTC DS12C887

Seperti telah dijelaskan dalam sebelumnya (tinjauan pustaka), bahwa untuk mengoperasikan IC *timer* ini diperlukan beberapa langkah, yaitu menentukan *register* A, B dan *register* C pada IC. Sedangkan untuk mengambil data dari hasil operasi *timer* RTC, dengan cara pembacaan *interrupt* pada bit "IUP" yang terletak pada *register* C. Pengaturan *interrupt* IUP langsung dikontrol oleh *register* B. Sedangkan pembacaan IUP ditentukan oleh *frekuensi counter* pada RTC, dimana *frekuensi* ini diatur pada *register* A.

Besarnya *frekuensi counter* yang dihasilkan dapat dilihat pada pin SQW, dengan *duty cycle* 50%. Sama seperti halnya IUP, SQW juga diatur oleh *register*

B untuk pengaktifkannya. Setelah bit IUP memiliki *carry flag* "1", data waktu dapat diambil. Selanjutnya sebelum ditampilkan pada layar *display* LCD, terlebih dahulu data (jam, menit, tanggal, bulan dan tahunnya) akan dicocokkan dengan data masa aktif dari masing-masing agenda yang tersimpan pada memori.

Hubungan tatap muka antara IC *timer* RTC DS 12C887 dengan sistem kontrol ditentukan oleh pin MOT pada pin 1. Bila berhubungan dengan sistem kontrol MOTOROLA maka pin MOT dihubungkan dengan Vcc, sebaliknya bila dengan sistem kontrol INTEL (mikrokontroler) pin MOT dihubungkan dengan *ground.* Rangkaian *timer* RTC nilai komponen penunjang, dicuplik dari "*Aplication Data Sheet Dallas Semicoductor*". Sedangkan untuk jalur pin data-alamat serta pin kontrol disesuaikan dengan perencanaan sistem seperti ditunjukkan pada gambar 3.6:

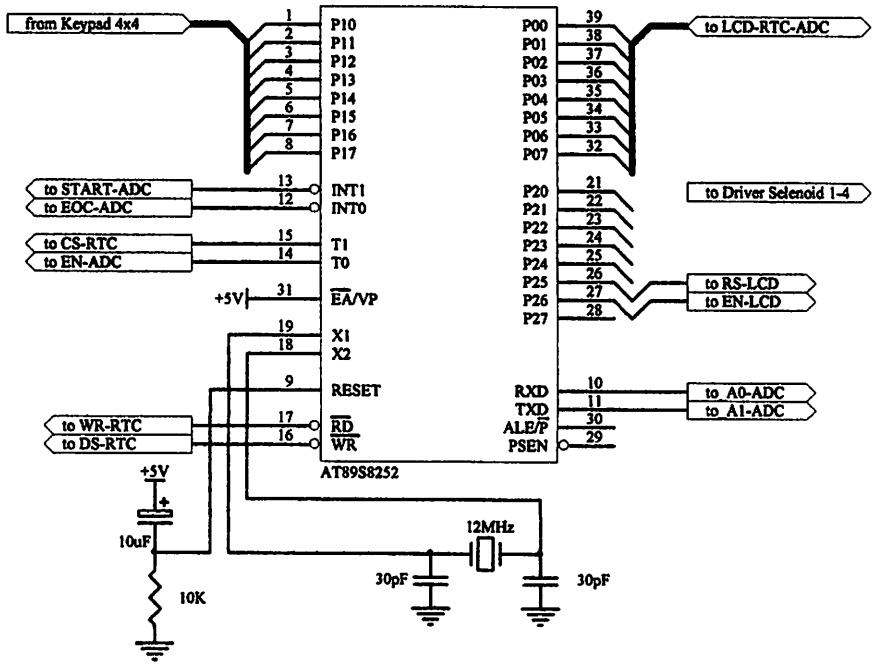


Gambar 3.6. Rangkaian *Timer* dengan IC RTC DS 12C887
*Sumber : Perencanaan

3.2.6. Rangkaian Sistem Kontrol Utama (Mikrokontrole AT89S8252)

Dalam perancanngan perangkat keras mikrokontroler AT89S8252 sebagai kontrol utama untuk menjalankan sistem, akan dibantu dengan rangkaian penunjang lainnya. Penggunaan rangkaian karena sistem kontrol beroperasi dalam sistem *digital*, sedangkan objek yang dibaca maupun yang dikontrol merupakan sistem *analog* murni. Mikrokontroler memiliki memori mpenyimpan program (Flash EPROM) dan penyimpan data dengan (EEPROM), yang sifatnya *non volatile* (tidak menguap) meskipun power dimatikan.

Pada mikrokontroler tersebut menggunkan sistem *clok* eksternal dengan frekuensi 12MHz. *Clok* ini akan disusun dari kristal 12MHz dan 2 kapasitor keramik 30pF. Gambar minimum sistem dari IC kontrol mikrokontroler AT89S8252 beserta jalur data dan kontrol ditunjukkan pada gambar 3.7 berikut:



Gambar 3.7. Rangkaian Sistem Kontrol AT89S8252
*Sumber : Perencanaan

3.2.7. Rangkaian *Driver Selenoid Valve*

Untuk menghubungkan pengontrol utama mikrokontroler dengan alat yang dikontrol, dibutuhkan suatu rangkaian penyesuai tegangan antara tegangan

pengontrol utama mikrokontroler dengan tegangan alat yang dikontrol. Penyesuai tegangan ini yang disebut rangkaian *driver*.

Selenoid pada perencanaan ini berfungsi sebagai buka-tutup kran air penyiraman. Komponen yang digunakan adalah 2 buah transistor NPN FCS 9013 dengan Hfe 120 dan TIP 41 Hfe 40. Sebuah solenoid 12V dan 1 buah diode sebagai penghantar arus diri pada kumparan *selenoid*. Dimana kedua komponen tersebut disusun dengan *driver model darlington* guna menghasilkan penguatan arus yang besar, sehingga sesuai dengan kebutuhan beban. Kerena memiliki penguatan arus yang sangat besar, maka diperlukan pembatas arus dengan resitor basis (R_b) pada basis transistor utama, agar transistor beban tidak berkerja terlalu jenuh akibat kelebihan arus basis.

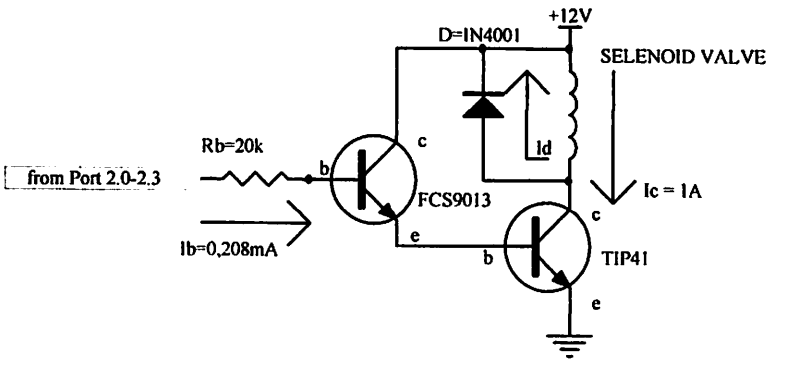
$$I_b = \frac{I_C}{h_{fe-1} \; x \; h_{fe-2}} \tag{3.8}$$

$$= \frac{1A}{120 \; x \; 40} = 0{,}208mA$$

$$R_b = \frac{V_{BB} - 2 \; x \; V_{BE}}{I_B} \tag{3.9}$$

$$= \frac{5V - 2x0{,}6}{0{,}208mA} = 18{,}269k\Omega$$

Untuk pemasangan Rb yang mendekati maka dipasang nilai 20 KΩ. Gambar dari *driver solenoide* terdapat pada gambar 3.8.



Gambar 3.8. Rangkaian *Driver Selenoid Model Darlington*
Sumber : Perencanaan

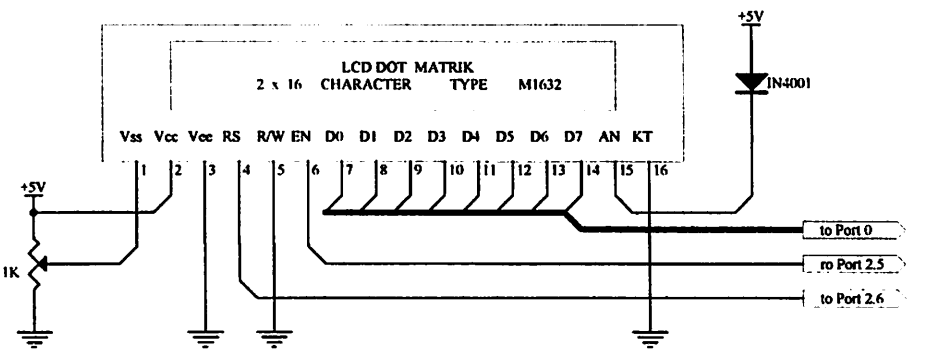
3.2.8. Rangkaian *Dispplay* LCD (M1632)

Rangkaian LCD dot matrik 2 x 16 karakter ini adalah komponen *display* yang paling umum digunakan saat ini. *Display* LCD 2 x 16 karakter ini memiliki ROM sebagai penyimpan karakter sebanyak 192 buah. Sebelum mengoperasikan LCD sebagai penampil karakter, terlebih dahulu ditentukan format penulisan LCD. Dalam format penulisan LCD ada beberapa aturan yang diberikan oleh pabrik pembuatnya (*data sheet*) yaitu:

1. Menentukan jalur bit data yang akan digunakan (4bit atau 8bit).
2. Membersihkan layar *display* dari karakter *blank.*
3. Menentukan alamat baris pertama dan baris kedua.
4. **Dalam penulisan karakter menggunakan *cursor* atau tidak.**

Jika penginisialisasian telah selesai, langkah selanjutnya menulis karakter yang diinginkan, misal tampilan yang diinginkan "123", maka format data yang ditransferkan ke jalur data adalah format data BCD angka 123 yang masing-masing disertai data posisi baris. Data yang dikirim ke LCD cukup satu kali,

selanjutnya data akan terus ditampilkan berulang-ulang oleh LCD itu sendiri selama tidak ada intruksi membersihkan layar *display*. Hubungan pin data dan pin kontrol LCD dengan mikrokontroler ditunjukkan dalam gambar 3.9 di bawah ini:



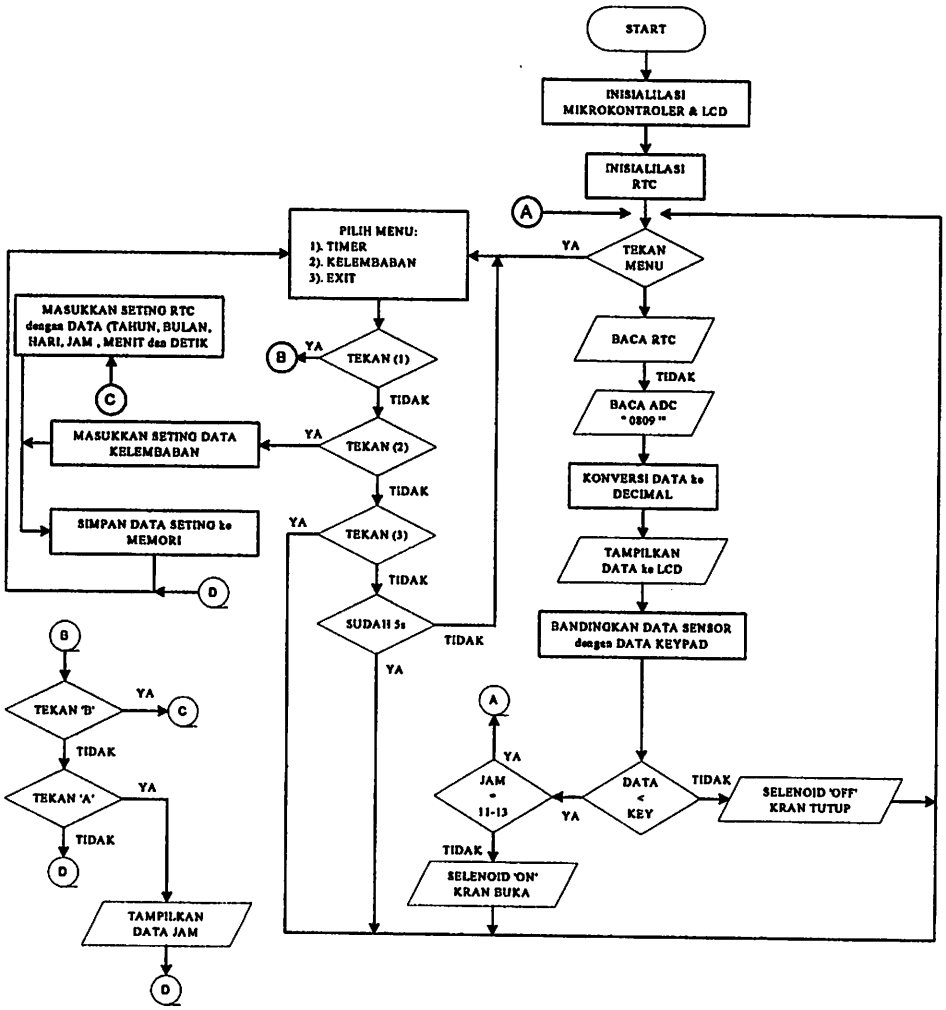
Gambar 3.9. Rangkaian LCD dengan Jalur Data dan Pin Kontrol
*Sumber : Perancangan

3.3. Perencanaan Perangkat Lunak (*Software*)

Pada sub-bab ini dibahas juga perencangan perangkat lunak secara diagram alir. Untuk pemakaian sitem kontrol digital terprogram, perlu direncanakan perangkat lunak yang dapat mengatur sistem tersebut. Perangkat lunak disini adalah susunan perintah-perintah (program) di dalam memori yang harus dilaksanakan oleh mikrokontroler.

Di dalam suatu mikrokontroler terdapat memori (Flash EPROM)) merupakan fasilitas utama untuk penyimpanan intruksi, karena disinilah disimpan perintah-perintah yang harus dikerjakan. Perencanaan perangkat lunak (*software*) didasarkan perencanaan perangkat keras yang telah dibuat sebelumnya, untuk mendapatkan sistem kerja yang diharapkan. *Software* dari alat tersebut terdapat di bagian lampiran dan diagram alir (*flow chart*).

3.3.1. Flowchart



Gambar 3.10. *Flowchart* Program Penyiraman Otomatis
*Sumber : Perencanaan

BAB IV

PENGUJIAN ALAT

Untuk mengetahui sistem penyiraman otomatis pada tanaman kedelai secara terprogram ini dapat bekerja seperti yang diharapkan, maka diperlukan pengujian-pengujian terhadap sistem. Dalam rangka pengujian tersebut, pada bab ini diuraikan sejumlah pengukuran dan percobaan yang dilakukan untuk mengetahui sistem kerja alat secara keseluruhan. Ada bebrapa cara dalam pengujian yaitu pengujian perangkat keras (*hardware*) per-blok dan alat secara keseluruhan (*hardware* dan *software*). Berikut ini prosedur pengujian perblok dan hasil pengamatan terhadap pengujian.

4.1. Pengujian Sensor Kelembaban Tanah dan Rangkaian Penguat

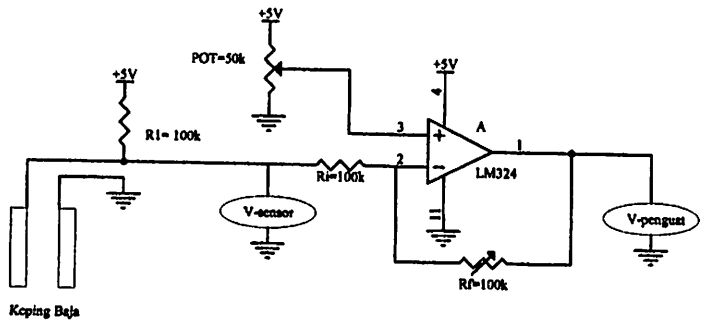
a. Tujuan:

Pada pengujian ini bertujuan untuk mengetahui kepekaan sensor kelembaban dalam mendeteksi tingkat intensitas kadar air dalam tanah dengan jenis dan kedalam tanah yang berbeda.

b. Alat dan bahan:

- Catu daya +5Volt DC
- Voltmeter *digital*
- Media tanah dan air
- Rangkaian pengujian

c. Blok gambar cara pengujian:



Gambar 4.1. Pengujian Sensor Kelembaban Tanah
Sekaligus Rangkaian Penguat Tegangan
*Sumber : Rangkaian pengujian

d. Data hasil pengukuran alat:

Tabel 4.1
Data Hasil Pengukuran Rangkaian Sensor Kelembaban Tanah

No	Jenis Tanah	*Kelembaban*	Tegangan *Output Sensor* (Volt)	Penguatan (kali)	Tegangan *Output Op-Amp* (Volt)
1	Liat	Kering	4,35	1	4,83
2	Liat	Lembab	2,78	1	2,55
3	Liat	Basah	0,43	1	0,62
4	Pasir	Kering	4,21	1	4,24
5	Pasir	Lembab	2,32	1	2,97
6	Pasir	Basah	0,22	1	0,31

*Sumber : Data hasil pengukuran

e. Analisis:

Setelah melakukan pengujian dan melihat data dari hasil beberapa pengukuran kelembaban tanah, di dapat analisis yaitu, sensor kelemaban dapat mendeteksi adanya air dalam tanah dan sensor memiliki tingkat resistansi yang berbeda untuk tiap struktur tanah yang berbeda.

4.2. Pengujian Rangkaian ADC 0809

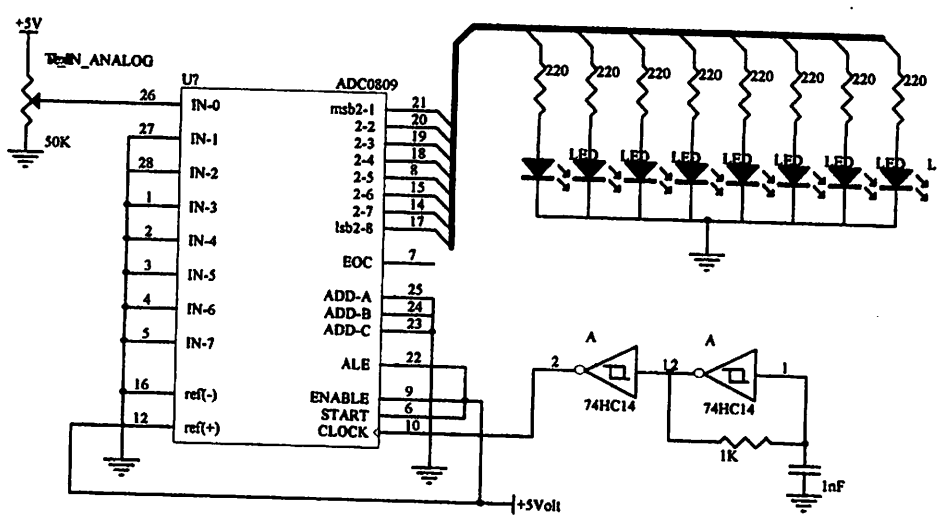
a. Tujuan:

Untuk mengetahui hasil konversi tegangan *analog* menjadi bentuk tegangan *digital* (*high* '1' & *low* '0') dari tiap volt *input* tegangan *analog*.

b. Alat dan bahan:

- Catu daya + 5Volt DC
- Voltmeter *digital*
- 8 LED masing-masing dipasang seri dengan resistor 220 Ohm.
- Variabel tegangan *input* (sebagai simulasi pengujian)
- Rangkaian pengujian.

c. Blok gambar cara pengujian:



Gambar 4.2. Pengujian Rangkaian ADC 0809
*Sumber : Rangkaian pengujian

d. Data hasil pengukuran alat:

Tabel 4.2
Data Hasil Pengukuran Rangkaian Konversi ADC 0809

Vin_0 (Volt)	Keluaran ADC						
	Perhitungan				Pengukuran		
	D7-D4	D3-D0	Des	Hex	D7-D4	D3-D0	Des
5,09	1111	1111	255	FFH	1111	1111	255
4,76	1110	1110	238	EEH	1110	1111	239
4,40	1101	1100	220	DCH	1101	1100	220
4,00	1100	1000	200	C8H	1100	1011	203
3,76	1011	1100	188	BCH	1011	1110	190
3,24	1010	0010	162	A2H	1010	0000	160
3,08	1001	1010	154	9AH	1001	1011	155
2,84	1000	1110	142	8EH	1000	1110	142
2,52	0111	1110	126	7EH	0111	1110	126
2,20	0110	1110	110	6EH	0110	1110	110
1,88	0101	1110	94	5EH	0101	1110	94
1,54	0100	1101	77	4DH	0100	0000	64
1,24	0011	1110	62	3EH	0011	0000	48
0,92	0010	1110	46	2EH	0010	0000	32
0,62	0001	1111	31	1FH	0001	1111	31
0,28	0000	1110	14	0EH	0000	1111	15
0,00	0000	0000	0	00H	0000	0000	0

*Sumber : Data pengukuran

Keterangan:

Vin = Tegangan input ADC

"1" = LED ON

"0" = LED OFF

D7-D4 = Keluaran D7, D6, D5, D4 (MSB)

D3-D0 = Keluaran D3, D2, D1, D0 (LSB)

Des = Keluaran dalam bentuk desimal

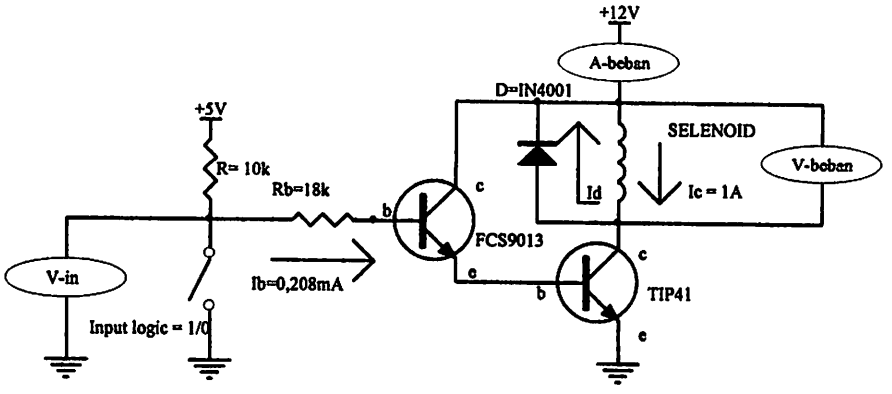
e. Analisis:

Pada percobaan ini kesalahan (*error)* masih dapat diabaikan, jika kesalahan tersebut berada pada keluaran D_0-D_1 , karena bobot nilai dari rumus (2^n) pada keluaran ini sangat rendah, yakni antara (0 – 3 desimal) saja.

4.3. Pengujian Rangkaian *Driver Selenoid*

Pengujian *driver selenoid* sebagi simulasi penggerak buka tutup kranair secara elektrik, dengan cara menghubungkan catu daya yang sesuai (+12 V) terlebih dahulu, kemudian memberi masukan pada *input logic* (1 / 0) pada basis transistor *driver*.

Dengan pengujian cara ini dapat diketahui pada saat *input* diberi masukan logika "1" maka transistor pertama (9013) akan mengalirkan arus kolektor yang lebih besar dan sebanding dari hasil kali arus basis dengan penguatan arusnya (Hfe). Kemudian arus kolektor yang mengali menuju emitor akan diumpankan pada basis transistor terakhir, dengan demikian maka, arus kolektor pada transistor (TIP 41) akan mengalir melalui selenoid. Pada kondisi ini selenoid akan membangkitkan medan listrik yang akan membuka katup penutup aliran air (sebagai kran). Kondisi sebaliknya jika *input* basis transistor pertama diberi logikia *low* '0', maka arus kolektor pada kedua transistor akan tersumbat dan menyebabkan selenoid tidak melepaskan katup penutup kran, dengan demikian kran air akan tertutup kembali.



Gambar 4.3. Pengujian *Driver Selenoid Valve*
*Sumber : Rangkaian pengujian

4.4. Pengujian *Timer* RTC DS 12C887, LCD & Keypad

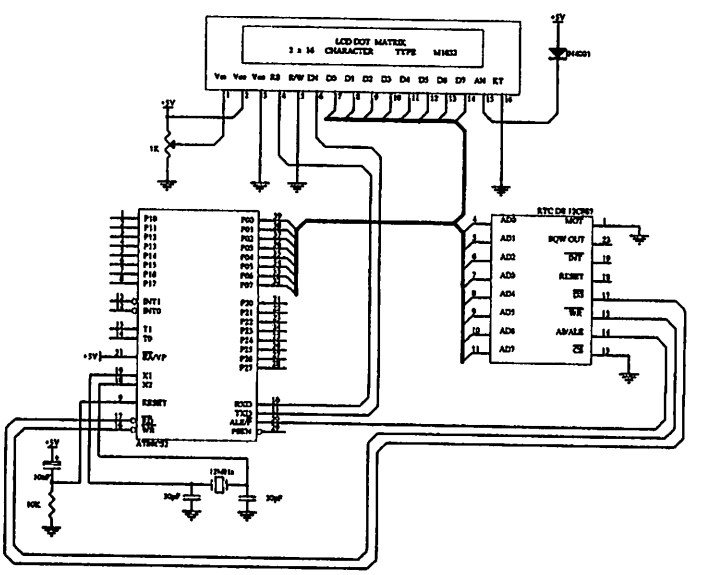
a. Tujuan :

Bagaimana memanfaatkan sebuah IC *timer* dan sekaligus RAM internalnya, yang data padanya tidak hilang walaupun catudaya dimatikan. Kemudian bagaimana menampilkan data *timer* tersebut pada *display* LCD, dimana dalam pengujian ini sudah termasuk cara penulisan data pada *display* LCD.

b. Alat dan bahan :

- Sumber tegangan +5VDC
- Program 'HB 2000' serial pada PC sekaligus *emulator*
- *Software* dengan bahasa assembler
- Rangkaian minimum sistem mikrokontroler, RTC, LCD dan keypad.

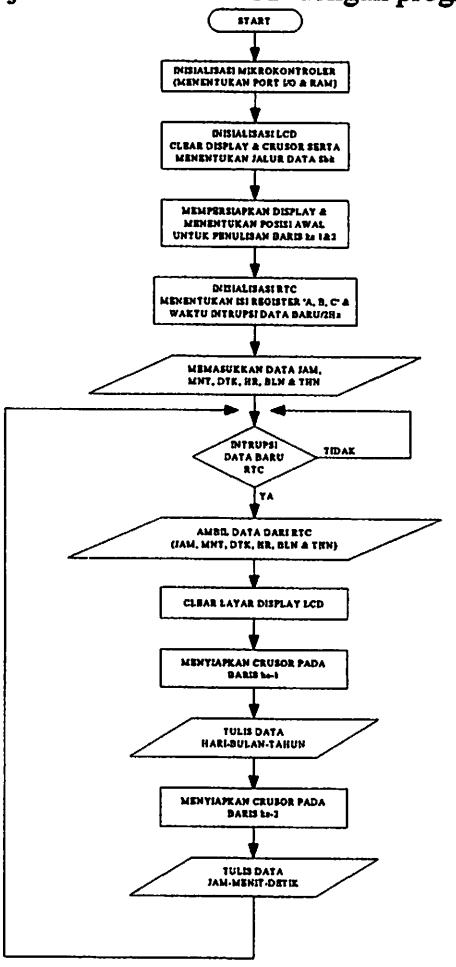
c. Gambar cara pengujian :



Gambar 4.4. Blok Cara Pengujian Rangkaian RTC dengan LCD dan Mikrokontroler AT89S8252

*Sumber : Pengujian

d. *Folwchart* pengujian IC RTC dan LCD dengan program assembler:



Gambar 4.5. *Flowchart* Penulisan dan Pengambilan Data RTC & LCD
*Sumber : Pengujian

e. Data hasil pengujian :

Tabel 4.3

Data Hasil Pengamatan Timer *RTC Pada LCD*

No	Baris LCD	Tampilan LCD	Keterangan
1	Pertama	Minggu –Oktober – 2005	Pergantian Hari/Jam 00:00:00
2	Kedua	20 : 45 : 05	*Timer-Up*/Detik

*Sumber : Pengujian

f. Analisis :

Pada pengujian blok ini tidak dapat dilakukan seperti pengujian rangkaian lainnya, karena pada blok ini masih membutuhkan perangkat lunak untuk

mengatur dan membaca data RTC tersebut. Pengujian RTC ini didukung rangkaian kontrol dan *display* LCD, karena pengujian rangkaian RTC tidak dapat berdiri sendiri. Melihat dari tabel hasil pengamatan pada pengujian RTC dan LCD, menunjukkan bahwa program penulisan dan pembacaan sekaligus menampilkan data waktu pada layar *display* LCD, berjalan sesuai yang diharapkan.

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil perancangan, pengujian dan analisis alat didapat bebarapa kesimpulan sebagai berikut :

1. Sistem hanya dilengkapi empat sensor dan empat *selenoid* pembuka kran air.
2. Waktu penyiraman dapat dilakukan sewaktu-waktu, kecuali pada jam-jam khusus jika terjadi kekeringan, sistem kontrol akan menahan pembuka kran air tetap tertutup, yakni dari jam 11 – 1 siang.
3. Penyiram akan dilakukan, jika sensor terbaca lebih kecil dari harga data pada keypad, dan selama itu kran air akan membuka.
4. Penggunaan IC RTC DS12C887 dapat memberikan pewaktuan yang tepat saat pada jam-jam khusus untuk tidak menyiram, hal ini untuk menghindari terbakarnya daun akibat dari terik sinar matahari.

5.2. Saran-Saran

1. Diperlukan penyesuaian alat (mekanik) jika untuk penggunaan yang sesungguhnya (kran air dan banyaknya sensor).
2. Sistem dapat dikembangkan dengan memberikan data kelembaban sampai pada akhir tanaman (panen).
3. Penyesuaian sensor kelembaban dapat dikembangkan menjadi model tongkat yang diberi satuan centimeter untuk mengukur kedalaman sensor.
4. Sensor dapat diperbanyak dengan cara dipararelkan.

DAFTAR PUSTAKA

- Agfianto Eko Putra, *Belajar Mikrokontroler AT89S51/52 (Teori dan Aplikasi)*, edisi ketiga, Yogyakarta: Gava Media, 2004.
- Anonimous, 2001, SS9013, TIP41 Fairchild. http://www.dzjs.com/file/9000/Ss9013.pdf#search='TRANSISTOR%209013'.html 12/03/2005.
- Coughlin, Robert F. and Frederick F. Driscoll (1982), *Penguat Operasional dan Rangkaian Terpadu Linear*, Terjemahan: Ir. Herman Widodo S., Erlangga, Jakarta, 1982.
- Edminister, Joseph A.,M.S.E., *Elektromagnetika,* Erlangga, Jakarta, 1997.
- Hayt, William H.(1979), *Elektromagnetika Teknologi*, Terjemahan: Murjono N.Sc, Erlangga, Jakarta, 1989.
- Malik, Mch. Ibnu, dan Anistradi, *Bereksperimen Dengan Mikrokontroler 8031*, PT. Elex Media Komputindo, Jakartra, 1997.
- Malvino, ***Elektronic Principle***, 3rd edition, Mc Graw Hill Inc., New York, (1984).
- National Semikonduktor Coporation, Inc., Dta Acquition, ***Operational Amplifier***, National semi konductor, Santa clara, Californa, (1995).
- Paulus Andi Nalwan, 2003, *Teknik Antar Muka dan Pemrograman Mikrokontroler AT89C51*, Jakartra: PT. Elex Media Komputindo.
- Rusdianto, Eduard,S.T., *Penerapan Konsep Dasar Listrik dan Elektronika*, Kanisius, Yogyakarta, 1999.
- Somaatmaja, Sadikin, *Kedelai,* Badan Penelitian Dan Pengembangan Pertanian,Pusat Penelitian Dan Pengembangan Tanaman Pangan, Bogor.
- Wasito, S., *Vademekum Elektronika,* PT Multimedia, Jakarta, 1985.

LAMPIRAN

INSTITUT TEKNOLOGI NASIONAL
JL. BENDUNGAN SIGURA-GURA 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

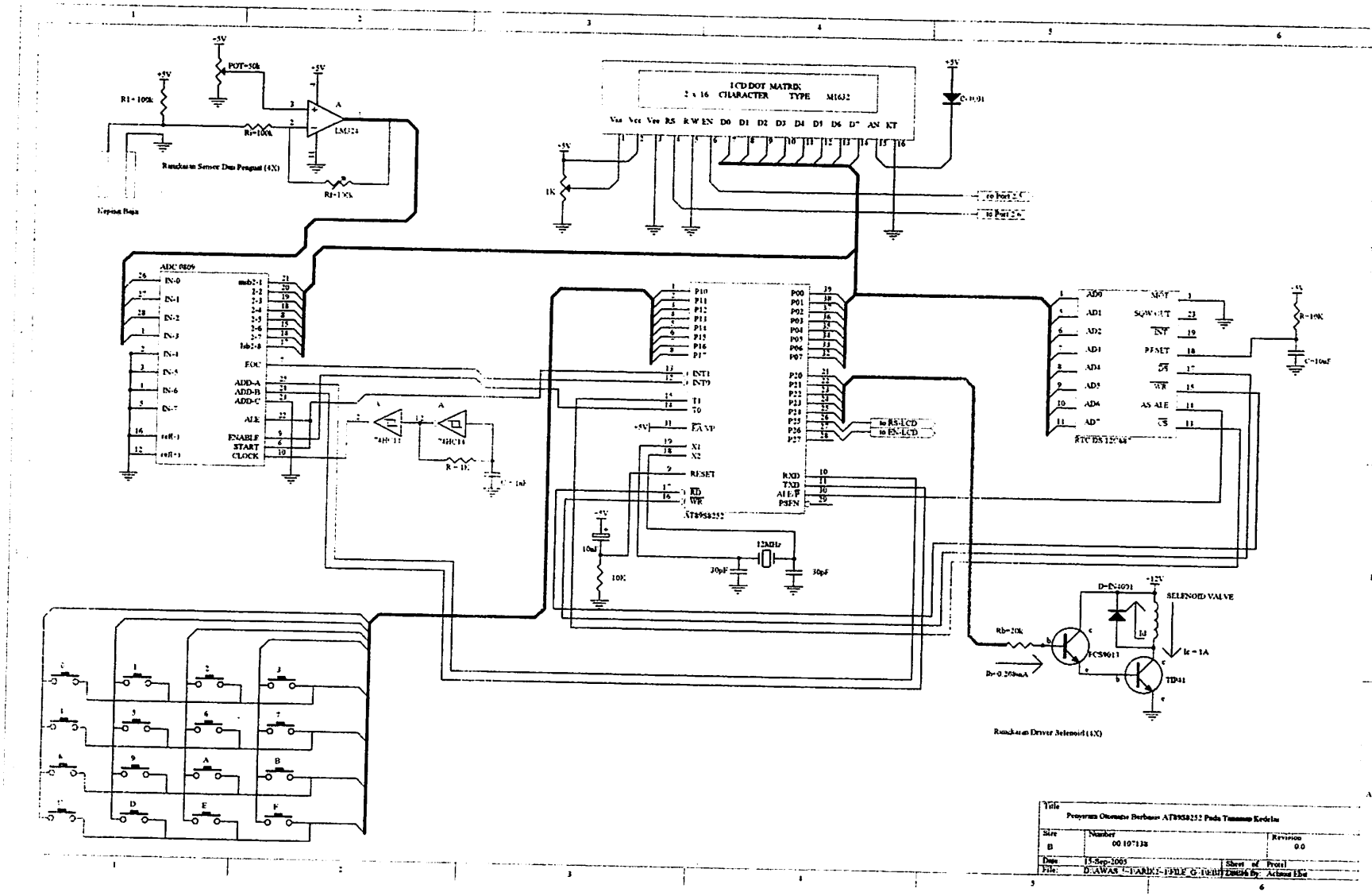
Nama : Ahmad Ebit S.
Nim : 0017138
Masa Bimbingan : 19-Jul-2005 s/d 19-Jan-2006
Judul Skripsi : Perancangan dan pembuatan alat penyiram air otomatis pada kedelai menggunakan Mikrokontroller AT89S8252

NO	Tanggal	Uraian	Paraf Pembimbing
1.	21/07 05	Konsultasi pemb. alat	
2.	07/08 05	Konsultasi lap. skripsi	
3.	02/09 05	Konsultasi bab I, II	
4.	16/09 05	Konsultasi bab III	
5.	02/10 05	Konsultasi bab IV & V	
6.	20/10 05	Percobaan Alat	
7.	11/11 05	Revisi Alat	
8.	17/10 05	ACC Ujian	
9.			
10.			

Malang, 2005
Dosen Pembimbing

Ir. Teguh Herbasuki, MT

Form. S-4a


+5V
POT=50k
R1=10k
Rf=100k
LM324
Rangkaian Sensor Dan Pengaut (4X)
Keypad Baja
1CD DOT MATRIX
2 x 16 CHARACTER TYPE M1632
Vss Vcc Vee RS R/W EN D0 D1 D2 D3 D4 D5 D6 D7 AN KT
1K
to Port 2.C
to Port 2.6
ADC 0809
IN-0
IN-1
IN-2
IN-3
IN-4
IN-5
IN-6
IN-7
EOC
ADD-A
ADD-B
ADD-C
ALE
ENABLE
START
CLOCK
AT89S8252
EA/VP
RESET
RD
WR
RXD
TXD
ALE/P
PSEN
12MHz
30pF
10uF
10K
to RS-LCD
to EN-LCD
RTC DS12C887
SQW/OUT
INT
RESET
DS
WR
AS ALE
CS
R=10K
C=10uF
D=IN4001
SELENOID VALVE
Rb=20k
FCS9013
TIP41
Ic = 1A
Rangkaian Driver Selenoid (4X)
Title
Program Otomatis Berbasis AT89S8252 Pada Tanaman Kedelai
Size
B
Number
00107138
Revision
0.0
Date
15-Sep-2005
Sheet of
File:
Drawn By:

```
;=============================================
;               KONSTANTA PORT
;=============================================
LCD             EQU   P0      ;DISPLAY
RTC             EQU   P0      ;TIMER
ADC             EQU   P0      ;ADC0809
KEY             EQU   P1      ;KEYPAD
;
SEL_1           BIT   P2.0    ;
SEL_2           BIT   P2.1    ;
SEL_3           BIT   P2.2    ;
SEL_4           BIT   P2.3    ;
MTR             BIT   P2.4    ;
RS              BIT   P2.5
EN              BIT   P2.6
CON_A           BIT   P3.0
CON_B           BIT   P3.1
START           BIT   P3.2
EOC             BIT   P3.3
CS_T            BIT   P3.4
BACA            BIT   P3.5
;=============================================
;               RAM
;=============================================
BARIS                 EQU   3BH
DATA_4                EQU   3CH
PER_BARIS             EQU   3DH
BAN_P1                EQU   3EH
COMMON                EQU   3FH
BOUNC                 EQU   35H
DAT                   EQU   36H
DATA_KEY              EQU   37H
;
REG_A                 EQU   0AH
REG_B                 EQU   0BH
REG_C                 EQU   0CH
REG_D                 EQU   0DH
;
RTC_DETIK             EQU   00H
RTC_MENIT             EQU   02H
RTC_JAM               EQU   04H
RTC_HARI              EQU   06H
RTC_TANGGAL           EQU   07H
RTC_BULAN             EQU   08H
RTC_TAHUN             EQU   09H
RTC_CENTURY           EQU   32H
;
PULUHAN               EQU   40H
SATUAN                EQU   41H
;
```

```
PUL_TAHUN       EQU  42H
SAT_TAHUN       EQU  43H
PUL_BULAN       EQU  44H
SAT_BULAN       EQU  45H
PUL_TANGGAL     EQU  46H
SAT_TANGGAL     EQU  47H
PUL_HARI        EQU  48H
SAT_HARI        EQU  49H
PUL_JAM         EQU  4AH
SAT_JAM         EQU  4BH
PUL_MENIT       EQU  4CH
SAT_MENIT       EQU  4DH
PUL_DETIK       EQU  4EH
SAT_DETIK       EQU  4FH
;
DAT_TANGGAL     EQU  50H
DAT_HARI        EQU  51H
DAT_BULAN       EQU  52H
DAT_TAHUN       EQU  53H
;
DAT_JAM         EQU  54H
DAT_MENIT       EQU  55H
DAT_DETIK       EQU  56H
;
DAT_UPP         EQU  57H
DAT_DWN         EQU  58H
DAT_ERR         EQU  59H
DAT_BUF         EQU  5AH
BUF_HARI        EQU  5BH
BUF_BULAN       EQU  5CH
DAT_KON         EQU  5DH
DAT_KONVIC      EQU  5EH
;
LEMB_1          EQU  60H
LEMB_2          EQU  61H
LEMB_3          EQU  62H
LEMB_4          EQU  63H
;
BUF_1           EQU  64H
BUF_2           EQU  65H
BUF_3           EQU  66H
BUF_4           EQU  67H
;
EROR_1          EQU  68H
EROR_2          EQU  69H
EROR_3          EQU  6AH
EROR_4          EQU  6BH
;
CAD_1           EQU  6CH
CODE_1          EQU  6DH
```

```
CODE_2                  EQU  6EH
VALUE                   EQU  6FH
;=============================================
;       PROG_AWAL
;=============================================
        ORG  00H
        SETB CS_T
        SETB SEL_1
        SETB SEL_2
        SETB SEL_3
        SETB SEL_4
        SETB MTR
        SETB EN
        CLR  BACA
        CLR  RS
;-------------------------------------
        LCALL       INIT_LCD
        LCALL       HEADO_1
        LCALL       PLEASED
;-------------------------------------
START_1:
        LCALL       AMBIL_PERSEN
        LCALL       BACA_ADC
        LCALL       TAMPIL
        LCALL       BANDING
;
        LCALL       SIRAM
;
        LCALL       KEY_SCAN
        MOV         A,DAT
        CJNE        A,#1,START_2
        MOV         DAT,#0
        MOV         A,DATA_KEY
        CJNE        A,#0FH,START_2
        LJMP        MENU
START_2:
        LCALL       TUNDA
        LJMP        START_1
;=============================================
;       PRO_AMBIL PERSEN
;=============================================
AMBIL_PERSEN:
        MOV         R0,#60H
        MOV         R1,#10H
        MOV         R7,#4
        CLR         CS_T
PERSEN:
        LCALL       DELAY
        MOVX        A,@R1
        LCALL       DELAY
```

```
        MOV  @R0,A
        INC  R0
        INC  R1
        DJNZ R7,PERSEN
        SETB CS_T
        RET
;=============================================
;       PRO_BACA_ADC
;=============================================
BACA_ADC:
        MOV         R0,#64H
        MOV         R7,#4
        MOV         CAD_1,#0
        CLR         CON_A
        CLR         CON_B
        LCALL       DELAY
BACA_Y:
        LCALL       TULISEN
        LCALL       DELAY
        LCALL       DELAY
;       JNB         EOC,$
        LCALL       GUN_1
        SETB        BACA
        LCALL       DELAY
;-------
        MOV         A,ADC
        CLR         C
        MOV         B,#25
        SUBB        A,B
        MOV         B,#24
        DIV         AB
        MOV         CODE_1,A
        MOV         A,B
        MOV         B,#10
        MUL         AB
        MOV         B,#26
        DIV         AB
        MOV         CODE_2,A
;
        MOV         A,CODE_1
        SWAP        A
        ORL         A,CODE_2
;-------
        MOV         @R0,A
        INC         R0
        INC         CAD_1
        CLR         BACA
        DJNZ        R7,BACA_1
        RET
BACA_1:
```

```
        MOV         A,CAD_1
        CJNE        A,#1,BACA_2
        SETB        CON_A
        CLR         CON_B
        LJMP        BACA_Y
BACA_2:
        CJNE        A,#2,BACA_3
        CLR         CON_A
        SETB        CON_B
        LJMP        BACA_Y
BACA_3:
        SETB        CON_A
        SETB        CON_B
        LJMP        BACA_Y
;***************"START"**********************
TULISEN:
        SETB        START
        LCALL       DELAY
        LCALL       DELAY
        CLR         START
        LCALL       DELAY
        RET
;=============================================
;       PRO_TAMPIL
;=============================================
TAMPIL:
        MOV         R1,#60H
        MOV         R7,#4
        LCALL       CLEAR_TAMPILAN
        LCALL       ROW_1
        MOV         DPTR,#MAS_3
        LCALL       WRITE
TAMPIL_3:
        MOV         A,@R1
        ANL         A,#0F0H
        SWAP        A
        CJNE        A,#0,TAMPIL_1
        LCALL       SPASI
        SJMP        TAMPIL_2
TAMPIL_1:
        LCALL       TULIS
TAMPIL_2:
        MOV         A,@R1
        ANL         A,#0FH
        LCALL       TULIS
        LCALL       STRIP
        INC         R1
        DJNZ        R7,TAMPIL_3
;-------------------------------------
        MOV         R1,#64H
```

```
        MOV     R7,#4
        LCALL   ROW_2
        MOV     DPTR,#MAS_4
        LCALL   WRITE
TIMPIL_3:
        MOV     A,@R1
        ANL     A,#0F0H
        SWAP    A
        CJNE    A,#0,TIMPIL_1
        LCALL   SPASI
        SJMP    TIMPIL_2
TIMPIL_1:
        LCALL   TULIS
TIMPIL_2:
        MOV     A,@R1
        ANL     A,#0FH
        LCALL   TULIS
        LCALL   STRIP
        INC     R1
        DJNZ    R7,TIMPIL_3
        RET
;=============================================
;       PRO_BANDING
;=============================================
BANDING:
        MOV     DAT_UPP,#60H    ;LEMB
        MOV     DAT_DWN,#64H    ;BUF
        MOV     R0,#68H
        MOV     R7,#4
BAND_1:
        MOV     A,DAT_UPP
        MOV     R1,A
        MOV     A,@R1
        MOV     B,A
        MOV     A,DAT_DWN
        MOV     R1,A
        MOV     A,@R1
        CLR     C
        CJNE    A,B,BAND_2 ;BUF/LEMB
BAND_4:
        MOV     A,#0
BAND_3:
        MOV     @R0,A
        INC     R0
        INC     DAT_UPP
        INC     DAT_DWN
        DJNZ    R7,BAND_1
        RET
BAND_2:
        JNC     BAND_4
```

```
        MOV     A,#1
        SJMP    BAND_3
;=============================================
;       PRO_PUTAR
;=============================================
SIRAM:
        CLR     CS_T
        LCALL   DELAY
        LCALL   CEK_UIP
        LCALL   DELAY
;
        MOV     R0,#RTC_JAM
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_JAM,PULUHAN
        MOV     SAT_JAM,SATUAN
        LCALL   AS_BIN
        MOV     DAT_JAM,A ;UTK_PEMBANDING
        SETB    CS_T
        LCALL   DELAY
;
        MOV     A,DAT_JAM
        CJNE    A,#0BH,TARA_1
        MOV     VALUE,#1
        SJMP    TARA_3
TARA_1:
        CJNE    A,#0CH,TARA_2
        MOV     VALUE,#1
        SJMP    TARA_3
TARA_2:
        MOV     VALUE,#0
TARA_3:
        MOV     R1,#68H
        MOV     DAT,#0
;
        MOV     A,@R1
        CJNE    A,#0,TAR_1
TER_1:
        SETB    SEL_1
        INC     R1
        SJMP    TAR_2
TAR_1:
        MOV     A,VALUE
        CJNE    A,#0,TER_1
        CLR     SEL_1
        MOV     DAT,#1
        INC     R1
TAR_2:
        MOV     A,@R1
```

```
        CJNE    A,#0,TAR_3
TER_2:
        SETB    SEL_2
        INC     R1
        SJMP    TAR_4
TAR_3:
        MOV     A,VALUE
        CJNE    A,#0,TER_2
        CLR     SEL_2
        MOV     DAT,#1
        INC     R1
TAR_4:
        MOV     A,@R1
        CJNE    A,#0,TAR_5
TER_3:
        SETB    SEL_3
        INC     R1
        SJMP    TAR_6
TAR_5:
        MOV     A,VALUE
        CJNE    A,#0,TER_3
        CLR     SEL_3
        MOV     DAT,#1
        INC     R1
TAR_6:
        MOV     A,@R1
        CJNE    A,#0,TAR_7
TER_4:
        SETB    SEL_4
        INC     R1
        SJMP    TAR_8
TAR_7:
        MOV     A,VALUE
        CJNE    A,#0,TER_4
        CLR     SEL_4
        MOV     DAT,#1
        INC     R1
TAR_8:
        MOV     A,DAT
        CJNE    A,#0,TAR_9
TAR_A:
        SETB    MTR
        RET
TAR_9:
        MOV     A,VALUE
        CJNE    A,#0,TAR_A
        CLR     MTR
```

```
        RET
;=============================================
;       PRO_SETTING_MENU
;=============================================
MENU:
        LCALL   SETT_1
        MOV     CODE_1,#40
LIEP_1:
        MOV     CODE_2,#250
LIEP_6:
        LCALL   KEY_SCAN
        MOV     A,DAT
        CJNE    A,#1,LIEP_3
        MOV     DAT,#0
        SJMP    LIEP_4
LIEP_3:
        DJNZ    CODE_2,LIEP_6
        DJNZ    CODE_1,LIEP_1
        LJMP    START_1
LIEP_4:
        MOV     A,DATA_KEY
        CJNE    A,#01H,LIEP_5
        LJMP    CEK_TIMER ;CEK_RTC
LIEP_5:
        CJNE    A,#02H,LIEP_7       ;SET_LEMBAB
        LCALL   LEMBAB
        LCALL   DELAY
        LJMP    MENU
LIEP_7:
        CJNE    A,#03H,LIEP_3       ;EXIT
        LCALL   DELAY
        LJMP    START_1
;---------------------------------------
LEMBAB:
        MOV     BUF_1,#5
        MOV     BUF_2,#0AAH
        MOV     BUF_3,#2
        LCALL   NO_MEN
NUM_1:
        LCALL   KEY_SCAN
        MOV     A,DAT
        CJNE    A,#1,NUM_1
        MOV     DAT,#0
        MOV     A,DATA_KEY
        MOV     BUF_1,A
;
        CJNE    A,#0CH,NUM_2
        RET
NUM_2:
        CLR     C
```

```
        CJNE    A,#05H,NUM_22
        SJMP    NUM_1
NUM_22:
        JNC     NUM_1
        CJNE    A,#0,NUM_3
        SJMP    NUM_1
NUM_3:
        LCALL   NO_MEN
NUM_33:
        LCALL   KEY_SCAN
        MOV     A,DAT
        CJNE    A,#1,NUM_33
        MOV     DAT,#0
        MOV     A,DATA_KEY
        CJNE    A,#0EH,NUM_33
        LCALL   WAKTU
NUM_4:
        LCALL   KEY_SCAN
        MOV     A,DAT
        CJNE    A,#1,NUM_4
        MOV     DAT,#0
        MOV     A,DATA_KEY
;
        CJNE    A,#0CH,NUM_5
        RET
NUM_5:
        CLR     C
        CJNE    A,#0AH,NUM_55
        SJMP    NUM_4
NUM_55:
        JNC     NUM_6
        MOV     A,BUF_2
        SWAP    A
        ANL     A,#0F0H
        ORL     A,DATA_KEY
        MOV     BUF_2,A
        LCALL   NO_MEN
        DJNZ    BUF_3,NUM_4
NUM_7:
        LCALL   KEY_SCAN
        MOV     A,DAT
        CJNE    A,#1,NUM_7
        MOV     DAT,#0
        MOV     A,DATA_KEY
        CJNE    A,#0EH,NUM_7
        SJMP    NUM_8
NUM_6:
        CJNE    A,#0EH,NUM_4
NUM_8:
        LCALL   WAKTU
```

```
        LCALL   TUNGGU_1
;
        MOV     A,BUF_1
        MOV     B,#0FH
        ADD     A,B
        MOV     R0,A
;
        CLR     CS_T
        MOV     A,BUF_2
        LCALL   DELAY
        MOV  x  @R0,A
        LCALL   DELAY
        SETB    CS_T
        RET
;-------------------------------------
TULIS:
        ORL     A,#30H
        MOV     LCD,A
        LCALL   LCD_CONTROL
        RET
;=====================================================
;       PRO_CEK TIMER
;=====================================================
CEK_TIMER:
        LCALL   TUNGGU_2
LIOP_1:
        LCALL   KEY_SCAN
        MOV     A,DAT
        CJNE    A,#1,LIOP_1
        MOV     DAT,#0
        MOV     A,DATA_KEY
;
        CJNE    A,#0AH,LIOP_2   ;LIHAT TIMER
        LJMP    LIHAT_RTC
LIOP_2:
        CJNE    A,#0BH,LIOP_3   ;GANTI TIMER
        LJMP    SETING_RTC
LIOP_3:
        CJNE    A,#0CH,LIOP_1   ;GANTI TIMER
        LCALL   DELAY
        RET
;=====================================================
;       PRO_LIHAT TIMER
;=====================================================
LIHAT_RTC:
        MOV     R6,#10
SEEP_1:
        LCALL   AMBIL_DATA      ;AMBIL DATA RTC
        LCALL   AMBIL_HARIAN    ;AMBIL HARI RTC
        LCALL   TAMPIL_RTC      ;TAMPIL DATA RTC
```

```
        DJNZ        R6,SEEP_2
        LJMP        MENU
SEEP_2:
        LCALL       KEY_SCAN
        MOV         A,DAT
        CJNE        A,#1,SEEP_1
        MOV         DAT,#0
        MOV         A,DATA_KEY
        CJNE        A,#0CH,SEEP_1
        LJMP        MENU
;=============================================
;       PRO_SETTING_RTC_DGN_TAMPILAN
;=============================================
SETING_RTC:
        MOV         DAT_DWN,#0
        MOV         DAT_UPP,#99
        MOV         DAT_TAHUN,#0
        MOV         DAT_BUF,#7
;
        LCALL       SET_TIMER  ;DISPLAY
        LCALL       SET_TAHUN ;DISPLAY
        MOV         A,DAT_TAHUN
        LCALL       RIBU
YES_1:
        MOV         R7,#1
        LCALL       KEY_SCAN
        MOV         A,DAT
        CJNE        A,#1,YES_1
        MOV         DAT,#0
        MOV         A,DATA_KEY
        CJNE        A,#03H,SEY_1        ;UPP(3)
        SJMP        YES_XX
SEY_1:
        CJNE        A,#07H,SEY_2        ;DWN(7)
        SJMP        YES_YY
SEY_2:
        CJNE        A,#0EH,YES_1        ;ENTER(E)
        DJNZ        DAT_BUF,YES_BUF
        LJMP        SIMPAN_RTC
;=======
YES_BUF:
        LJMP        YES_4
YES_YY:
        LJMP        YES_Y
YES_XX:
        MOV         A,DAT_BUF
        CJNE        A,#7,YEX_1
;
        MOV         A,DAT_TAHUN
        MOV         B,DAT_UPP
```

```
        CJNE    A,B,UPPER_1
        LJMP    YES_1
UPPER_1:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_TAHUN ;DISPLAY
        INC     DAT_TAHUN
        LCALL   RIBU
        LJMP    YES_1
YEX_1:
        CJNE    A,#6,YEX_2
        MOV     A,DAT_BULAN
        MOV     B,DAT_UPP
        CJNE    A,B,UPPER_2
        LJMP    YES_1
UPPER_2:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_BULAN ;DISPLAY
        INC     DAT_BULAN
        LCALL   BUL_UPDW
        LJMP    YES_1
YEX_2:
        CJNE    A,#5,YEX_3
        MOV     A,DAT_HARI
        MOV     B,DAT_UPP
        CJNE    A,B,UPPER_3
        LJMP    YES_1
UPPER_3:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_HARI  ;DISPLAY
        INC     DAT_HARI
        LCALL   HAR_UPDW
        LJMP    YES_1
YEX_3:
        CJNE    A,#4,YEX_4
        MOV     A,DAT_TANGGAL
        MOV     B,DAT_UPP
        CJNE    A,B,UPPER_4
        LJMP    YES_1
UPPER_4:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_TANGGAL     ;DISPLAY
        INC     DAT_TANGGAL
        MOV     A,DAT_TANGGAL
        LCALL   BAGI
        LJMP    YES_1
YEX_4:
        CJNE    A,#3,YEX_5
        MOV     A,DAT_JAM
        MOV     B,DAT_UPP
        CJNE    A,B,UPPER_5
```

```
        LJMP    YES_1
UPPER_5:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_JAM         ;DISPLAY
        INC     DAT_JAM
        MOV     A,DAT_JAM
        LCALL   BAGI
        LJMP    YES_1
YEX_5:
        CJNE    A,#2,YEX_6
        MOV     A,DAT_MENIT
        MOV     B,DAT_UPP
        CJNE    A,B,UPPER_6
        LJMP    YES_1
UPPER_6:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_MENIT ;DISPLAY
        INC     DAT_MENIT
        MOV     A,DAT_MENIT
        LCALL   BAGI
        LJMP    YES_1
YEX_6:
        MOV     A,DAT_DETIK
        MOV     B,DAT_UPP
        CJNE    A,B,UPPER_7
        LJMP    YES_1
UPPER_7:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_DETIK ;DISPLAY
        INC     DAT_DETIK
        MOV     A,DAT_DETIK
        LCALL   BAGI
        LJMP    YES_1
;=======
YES_Y:
        MOV     A,DAT_BUF
        CJNE    A,#7,XEY_1
        MOV     A,DAT_TAHUN
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_1
        LJMP    YES_1
DOWER_1:
        LCALL   SET_TIMER ;DIPLAY
        LCALL   SET_TAHUN ;DISPLAY
        DEC     DAT_TAHUN
        LCALL   RIBU
        LJMP    YES_1
XEY_1:
        CJNE    A,#6,XEY_2
        MOV     A,DAT_BULAN
```

```
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_2
        LJMP    YES_1
DOWER_2:
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_BULAN ;DISPLAY
        DEC     DAT_BULAN
        LCALL   BUL_UPDW
        LJMP    YES_1
XEY_2:
        CJNE    A,#5,XEY_3
        MOV     A,DAT_HARI
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_3
        LJMP    YES_1
DOWER_3:
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_HARI   ;DISPLAY
        DEC     DAT_HARI
        LCALL   HAR_UPDW
        LJMP    YES_1
XEY_3:
        CJNE    A,#4,XEY_4
        MOV     A,DAT_TANGGAL
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_4
        LJMP    YES_1
DOWER_4:
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_TANGGAL     ;DISPLAY
        DEC     DAT_TANGGAL
        MOV     A,DAT_TANGGAL
        LCALL   BAGI
        LJMP    YES_1
XEY_4:
        CJNE    A,#3,XEY_5
        MOV     A,DAT_JAM
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_5
        LJMP    YES_1
DOWER_5:
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_JAM    ;DISPLAY
        DEC     DAT_JAM
        MOV     A,DAT_JAM
        LCALL   BAGI
        LJMP    YES_1
XEY_5:
        CJNE    A,#2,XEY_6
        MOV     A,DAT_MENIT
```

```
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_6
        LJMP    YES_1
DOWER_6:
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_MENIT  ;DIPLAY
        DEC     DAT_MENIT
        MOV     A,DAT_MENIT
        LCALL   BAGI
        LJMP    YES_1
XEY_6:
        MOV     A,DAT_DETIK
        MOV     B,DAT_DWN
        CJNE    A,B,DOWER_7
        LJMP    YES_1
DOWER_7:
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_DETIK  ;DISPLAY
        DEC     DAT_DETIK
        MOV     A,DAT_DETIK
        LCALL   BAGI
        LJMP    YES_1
;=========
YES_4:
        MOV     A,DAT_BUF
        CJNE    A,#6,YES_5
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_BULAN ;DISPLAY
        MOV     DAT_DWN,#1
        MOV     DAT_UPP,#12
        MOV     DAT_BULAN,#1
        LCALL   BUL_UPDW
        LJMP    YES_1
YES_5:
        CJNE    A,#5,YES_6
        LCALL   SET_TIMER  ;DISPLAY
        LCALL   SET_HARI   ;DISPLAY
        MOV     DAT_DWN,#1
        MOV     DAT_UPP,#7
        MOV     DAT_HARI,#1
        LCALL   HAR_UPDW
        LJMP    YES_1
YES_6:
        CJNE    A,#4,YES_7
        LCALL   SET_TIMER       ;DISPLAY
        LCALL   SET_TANGGAL     ;DISPLAY
        MOV     DAT_DWN,#1
        MOV     DAT_UPP,#31
        MOV     DAT_TANGGAL,#1
        MOV     A,DAT_TANGGAL
```

```
        LCALL   BAGI
        LJMP    YES_1
YES_7:
        CJNE    A,#3,YES_8
        LCALL   SET_TIMER       ;DISPLAY
        LCALL   SET_JAM         ;DISPLAY
        MOV     DAT_DWN,#0
        MOV     DAT_UPP,#23
        MOV     DAT_JAM,#0
        MOV     A,DAT_JAM
        LCALL   BAGI
        LJMP    YES_1
YES_8:
        CJNE    A,#2,YES_9
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_MENIT ;DISPLAY
        MOV     DAT_DWN,#0
        MOV     DAT_UPP,#59
        MOV     DAT_MENIT,#0
        MOV     A,DAT_MENIT
        LCALL   BAGI
        LJMP    YES_1
YES_9:
        LCALL   SET_TIMER ;DISPLAY
        LCALL   SET_DETIK ;DISPLAY
        MOV     DAT_DWN,#0
        MOV     DAT_UPP,#59
        MOV     DAT_DETIK,#0
        MOV     A,DAT_DETIK
        LCALL   BAGI
        LJMP    YES_1
;=============================================
;       PRO_SIMPAN_JAM
;=============================================
SIMPAN_RTC:
;       CLR     EA
        LCALL   TUNGGU_1
;
        CLR     CS_T        ;>>
        LCALL   DOLAY
        LCALL   INIT_RTC_DS
        LCALL   RTC_SAVE_DS
        LCALL   DOLAY
        SETB    CS_T        ;>>
        LCALL   DELAY_TAMPILAN
;       SETB    EA
        LJMP    MENU
;=============================================
;       Inisialisasi RTC_DS
;=============================================
```

```
INIT_RTC_DS:
        LCALL   DELAY
        MOV     R1,#REG_A
        MOV     A,#2FH
        LCALL   DELAY
        MOVX    @R1,A
        LCALL   DELAY
;
        MOV     R1,#REG_B
        MOV     A,#1AH
        LCALL   DELAY
        MOVX    @R1,A
        LCALL   DELAY
;
        MOV     R1,#RTC_CENTURY
        MOV     A,#20H
        LCALL   DELAY
        MOVX    @R1,A
        LCALL   DELAY
        RET
;=============================================
;       PRO_SIMPAN_MENU_MAKANAN_DS
;=============================================
RTC_SAVE_DS:
        MOV     R1,#RTC_TAHUN
        MOV     A,DAT_TAHUN
        LCALL   SEIKO
;
        MOV     R1,#RTC_BULAN
        MOV     A,DAT_BULAN
        LCALL   SEIKO
;
        MOV     R1,#RTC_TANGGAL
        MOV     A,DAT_TANGGAL
        LCALL   SEIKO
;
        MOV     R1,#RTC_HARI
        MOV     A,DAT_HARI
        LCALL   SEIKO
;
        MOV     R1,#RTC_JAM
        MOV     A,DAT_JAM
        LCALL   SEIKO
;
        MOV     R1,#RTC_MENIT
        MOV     A,DAT_MENIT
        LCALL   SEIKO
;
        MOV     R1,#RTC_DETIK
        MOV     A,DAT_DETIK
```

```
        LCALL   SEIKO
        RET
;=============================================
;       PRO_AMBIL_DATA_RTC
;=============================================
AMBIL_DATA:
        CLR     CS_T
        LCALL   DELAY
        LCALL   CEK_UIP
        LCALL   DELAY
;
        MOV     R0,#RTC_TAHUN
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_TAHUN,PULUHAN
        MOV     SAT_TAHUN,SATUAN
        LCALL   DELAY
;
        MOV     R0,#RTC_BULAN
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_BULAN,PULUHAN
        MOV     SAT_BULAN,SATUAN
        LCALL   DELAY
;
        MOV     R0,#RTC_TANGGAL
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_TANGGAL,PULUHAN
        MOV     SAT_TANGGAL,SATUAN
        LCALL   DELAY
;
        MOV     R0,#RTC_HARI
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_HARI,PULUHAN
        MOV     SAT_HARI,SATUAN
        LCALL   DELAY
;
        MOV     R0,#RTC_JAM
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_JAM,PULUHAN
        MOV     SAT_JAM,SATUAN
        LCALL   AS_BIN
```

```
        MOV     DAT_JAM,A ;UTK_PEMBANDING
        LCALL   DELAY
;
        MOV     R0,#RTC_MENIT
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_MENIT,PULUHAN
        MOV     SAT_MENIT,SATUAN
        LCALL   AS_BIN
        MOV     DAT_MENIT,A     ;UTK_PEMBANDING
        LCALL   DELAY
;
        MOV     R0,#RTC_DETIK
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   KONVERSI
        MOV     PUL_DETIK,PULUHAN
        MOV     SAT_DETIK,SATUAN
        LCALL   AS_BIN
        MOV     DAT_DETIK,A     ;UTK_PEMBANDING
        SETB    CS_T
        LCALL   DELAY
        RET
;=============================================
;       PROG_AMBIL_HARI
;=============================================
AMBIL_HARIAN:
        MOV     A,SAT_HARI
        CJNE    A,#31H,BA_SENIN
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#MINGGU
        LCALL   WRITE
        RET
BA_SENIN:
        CJNE    A,#32H,BA_SELASA
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#SENIN
        LCALL   WRITE
        RET
BA_SELASA:
        CJNE    A,#33H,BA_RABU
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#SELASA
        LCALL   WRITE
        RET
BA_RABU:
```

```
        CJNE    A,#34H,BA_KAMIS
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#RABU
        LCALL   WRITE
        RET
BA_KAMIS:
        CJNE    A,#35H,BA_JUMAT
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#KAMIS
        LCALL   WRITE
        RET
BA_JUMAT:
        CJNE    A,#36H,BA_SABTU
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#JUMAT
        LCALL   WRITE
        RET
BA_SABTU:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#SABTU
        LCALL   WRITE
        RET
;=============================================
;       PRO_TAMPILAN_RTC
;=============================================
TAMPIL_RTC:
        LCALL   KOMA
;
        MOV     LCD,PUL_TANGGAL
        LCALL   LCD_CONTROL
        MOV     LCD,SAT_TANGGAL
        LCALL   LCD_CONTROL
;
        LCALL   STRIP
;
        MOV     LCD,PUL_BULAN
        LCALL   LCD_CONTROL
        MOV     LCD,SAT_BULAN
        LCALL   LCD_CONTROL
;
        LCALL   STRIP
;
        MOV     LCD,PUL_TAHUN
        LCALL   LCD_CONTROL
        MOV     LCD,SAT_TAHUN
        LCALL   LCD_CONTROL
```

```
;
        LCALL   ROW_2
        LCALL   DELAY
;
        SETB    RS
        LCALL   WIB
        LCALL   KOMA
;
        MOV     LCD,PUL_JAM
        LCALL   LCD_CONTROL
        MOV     LCD,SAT_JAM
        LCALL   LCD_CONTROL
;
        LCALL   TITIK
;
        MOV     LCD,PUL_MENIT
        LCALL   LCD_CONTROL
        MOV     LCD,SAT_MENIT
        LCALL   LCD_CONTROL
;
        LCALL   TITIK
;
        MOV     LCD,PUL_DETIK
        LCALL   LCD_CONTROL
        MOV     LCD,SAT_DETIK
        LCALL   LCD_CONTROL
        RET
;=============================================
;       PRO_KONVERSI_BCD_ASCII
;=============================================
KONVERSI:
        LCALL   DELAY
        MOV     DAT_KONVIC,A
        ANL     A,#0F0H
        SWAP    A
        ORL     A,#30H
        MOV     PULUHAN,A
        MOV     A,DAT_KONVIC
        ANL     A,#0FH
        ORL     A,#30H
        MOV     SATUAN,A
        RET
;=============================================
;       PRO_KONVERSI_ASCII_BINER/DESIMAL
;=============================================
AS_BIN:
        MOV     A,PULUHAN
        ANL     A,#0FH
        MOV     B,#10
        MUL     AB
```

```
        MOV     PULUHAN,A
        MOV     A,SATUAN
        ANL     A,#0FH
        MOV     B,PULUHAN
        ADD     A,B
        LCALL   DELAY
        RET
;=============================================
;       PRO_KONVERSI_BCD_BINER
;=============================================
BC_BIN:
        MOV     SATUAN,A
        ANL     A,#0F0H
        SWAP    A
        MOV     B,#10
        MUL     AB
        MOV     PULUHAN,A
        MOV     A,SATUAN
        ANL     A,#0FH
        MOV     B,PULUHAN
        ADD     A,B
        LCALL   DELAY
        RET
;=============================================
;       PRO_KONVERSI_BINER_BCD_DG_CLOK
;=============================================
SEIKO:
        MOV     B,#10
        DIV     AB
        SWAP    A
        ADD     A,B
GUCI_1:
        LCALL   DELAY
        MOVX    @R1,A
        LCALL   DELAY
        RET
;=============================================
;       PRO_CEK_'IRQF/UIP'
;=============================================
CEK_UIP:
        MOV     R0,#REG_C
        MOV     A,R0
        MOV     R1,A
SIBUK:
        LCALL   DELAY
        MOVX    A,@R0
        LCALL   DELAY
        ANL     A,#80H
        JNB     ACC.7,SIBUK
        CLR     ACC.7
```

```
        MOV     @R1,A
        LCALL   DELAY
        RET
;=============================================
;       PRO_DIS_AWAL
;=============================================
HEADO_1:
        MOV     7CH,#1
        MOV     7DH,#150
HEADO_2:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#HEADER_1
        LCALL   WRITE
        LCALL   ROW_2
        MOV     DPTR,#HEADER_2
        LCALL   ISI_DATA
        LCALL   DISPLAY
        INC     7CH
        LCALL   TUNDA
        LCALL   TUNDA
        DJNZ    7DH,HEADO_2
        LCALL   DELAY
        LCALL   DELAY
        RET
SETT_1:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#MENO_3
        LCALL   WRITE
        LCALL   ROW_2
        MOV     DPTR,#MENO_4
        LCALL   WRITE
        RET
;-----------------------------------------
PLEASED:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#PLS_1
        LCALL   WRITE
        LCALL   ROW_2
        MOV     DPTR,#PLS_2
        LCALL   WRITE
        LCALL   DELAY_TAMPILAN
        LCALL   DELAY_TAMPILAN
        RET
TUNGGU_1:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#PLS_3
```

```
        LCALL   WRITE
        LCALL   ROW_2
        MOV     DPTR,#PLS_4     ;SIMPAN PROG
        LCALL   WRITE
        LCALL   DELAY_TAMPILAN
        RET
TUNGGU_2:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#PLS_5     ;LIHAT TIMER
        LCALL   WRITE
        LCALL   ROW_2
        MOV     DPTR,#PLS_6     ;GANTI TIMER
        LCALL   WRITE
        RET
NO_MEN:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#MAS_1
        LCALL   WRITE
        MOV     A,BUF_1
        CJNE    A,#05H,MOM_1
        SJMP  MOM_2
MOM_1:
        LCALL   TULIS
MOM_2:
        LCALL   ROW_2
        MOV     DPTR,#MAS_2
        LCALL   WRITE
        MOV     A,BUF_2
        SWAP    A
        ANL     A,#0FH
        CJNE    A,#0AH,MAM_1
        MOV     A,BUF_2
        ANL     A,#0FH
        CJNE    A,#0AH,MAM_3
        SJMP    MAM_2
MAM_1:
        LCALL   TULIS
        MOV     A,BUF_2
        ANL     A,#0FH
MAM_3:
        LCALL   TULIS
MAM_2:
        RET
;-------
SET_TIMER:
        LCALL   CLEAR_TAMPILAN
        LCALL   ROW_1
        MOV     DPTR,#JAM_1
```

```
        LCALL   WRITE
        RET
SET_JAM:
        LCALL   ROW_2
        MOV     DPTR,#JAM_2
        LCALL   WRITE
        RET
SET_MENIT:
        LCALL   ROW_2
        MOV     DPTR,#JAM_3
        LCALL   WRITE
        RET
SET_DETIK:
        LCALL   ROW_2
        MOV     DPTR,#JAM_4
        LCALL   WRITE
        RET
SET_HARI:
        LCALL   ROW_2
        MOV     DPTR,#JAM_5
        LCALL   WRITE
        RET
SET_BULAN:
        LCALL   ROW_2
        MOV     DPTR,#JAM_6
        LCALL   WRITE
        RET
SET_TANGGAL:
        LCALL   ROW_2
        MOV     DPTR,#JAM_7
        LCALL   WRITE
        RET
SET_TAHUN:
        LCALL   ROW_2
        MOV     DPTR,#JAM_8
        LCALL   WRITE
        RET
KALI:
        MOV     A,#'H'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        MOV     A,#'a'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        MOV     A,#'r'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        MOV     A,#'i'
        MOV     LCD,A
        LCALL   LCD_CONTROL
```

```
        RET
WIB:
        MOV     A,#'W'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        MOV     A,#'I'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        MOV     A,#'B'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        LCALL   SPASI
        LCALL   SPASI
        LCALL   SPASI
        LCALL   SPASI
        RET
;=========
VIU:
        ORL     A,#30H
        MOV     LCD,A
        LCALL   LCD_CONTROL
        RET
;
;=========
BAGI:
        MOV     B,#10
        DIV     AB
        ORL     A,#30H
        MOV     LCD,A
        LCALL   LCD_CONTROL
        ORL     B,#30H
        MOV     LCD,B
        LCALL   LCD_CONTROL
        RET
;=========
HAR_UPDW:
        MOV     A,DAT_HARI
        MOV     B,#6
        MUL     AB
        MOV     BUF_HARI,A
        MOV     DAT_KON,#6
        MOV     DPTR,#HEAD_1
HAR_2:
        MOV     A,BUF_HARI
        MOVC    A,@A+DPTR
        MOV     LCD,A
        LCALL   LCD_CONTROL
        DJNZ    DAT_KON,HAR_1
        RET
HAR_1:
```

```
        INC     DPTR
        SJMP    HAR_2
;=============
BUL_UPDW:
        MOV     A,DAT_BULAN
        MOV     B,#9
        MUL     AB
        MOV     BUF_BULAN,A
        MOV     DAT_KON,#9
        MOV     DPTR,#HEAD_2
BUL_2:
        MOV     A,BUF_BULAN
        MOVC    A,@A+DPTR
        MOV     LCD,A
        LCALL   LCD_CONTROL
        DJNZ    DAT_KON,BUL_1
        RET
BUL_1:
        INC     DPTR
        SJMP    BUL_2
;=============
RIBU:
        MOV     LCD,#00110010B  ;'2'
        LCALL   LCD_CONTROL
        MOV     LCD,#00110000B  ;'0'
        LCALL   LCD_CONTROL
        MOV     A,DAT_TAHUN
        MOV     B,#10
        DIV     AB
        ORL     A,#30H
        MOV     LCD,A
        LCALL   LCD_CONTROL
        ORL     B,#30H
        MOV     LCD,B
        LCALL   LCD_CONTROL
        RET
;=============
SPASI:
        MOV     A,#' '
        MOV     LCD,A
        LCALL   LCD_CONTROL
        RET
STRIP:
        MOV     A,#'-'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        RET
TITIK:
        MOV     A,#':'
        MOV     LCD,A
```

```
        LCALL   LCD_CONTROL
        RET
KOMA:
        MOV     A,#';'
        MOV     LCD,A
        LCALL   LCD_CONTROL
        RET
;=============================================
;       Baca Keypad
;=============================================
KEY_SCAN:
        LCALL   BACA_TOMBOL
        MOV     A,DAT
        CJNE    A,#1,KEMBALI
LOMPAT:
        MOV     BOUNC,#5
LOMPAT1:
        MOV     P1,COMMON
        MOV     A,P1
        ORL     A,#0FH
        CJNE    A,#0FFH,LOMPAT
        DJNZ    BOUNC,LOMPAT1
KEMBALI:
        RET
;++++++++++++++++++++++++++++++++++++++++++++++
;Baca Tombol Keypad 4X4
;Jumlah baris:4 (P1.0 - P1.3)
;Jumlah kolom:4 (P1.4 - P1.7)
;++++++++++++++++++++++++++++++++++++++++++++++
BACA_TOMBOL:
        MOV  BARIS,#4
        MOV  COMMON,#11111110B      ;DATA BARIS
        MOV  DATA_4,#0
        MOV  DPTR,#ANGKA
BACA_TOMBOL1:
        MOV  BAN_P1,#01111111B      ;DATA KOLOM
        MOV  PER_BARIS,#0
BACA_TOMBOL2:
        MOV     P1,COMMON
        MOV     A,P1
        ORL     A,#0FH
        CJNE    A,BAN_P1,BACA_TOMBOL3
        MOV     A,PER_BARIS
        ADD     A,DATA_4
        MOVC    A,@A+DPTR
        MOV     DATA_KEY,A
        MOV     DAT,#1
        RET
BACA_TOMBOL3:
        INC     PER_BARIS
```

```
        MOV     A,BAN_P1
        RR      A
        MOV     BAN_P1,A
        MOV     A,PER_BARIS
        CJNE    A,#4,BACA_TOMBOL2
nextscan:
        MOV     A,DATA_4
        ADD     A,#4
        MOV     DATA_4,A
        MOV     A,COMMON
        RL      A
        MOV     COMMON,A
        DJNZ    BARIS,BACA_TOMBOL1
        MOV     P1,#0FFh
        MOV     DAT,#0
        RET
;=============================================
;       PROG_ISI DATA RAM
;=============================================
ISI_DATA:
        MOV     R1,#60H
        MOV     R3,#16
ISI_DATA1:
        MOV     A,7CH
        MOVC    A,@A+DPTR
        MOV     @R1,A
        INC     DPTR
        INC     R1
        DJNZ    R3,ISI_DATA1
        RET
;=============================================
;       PROG_DISPLAY
;=============================================
DISPLAY:
        MOV     R0,#60H
        MOV     R3,#16
        SETB    RS
DISPLAY1:
        MOV     A,@R0
        MOV     LCD,A
        LCALL   LCD_CONTROL
        INC     R0
        DJNZ    R3,DISPLAY1
        RET
;=============================================
;       Inisialisasi LCD
;=============================================
INIT_LCD:
        CLR     RS
        MOV     LCD,#3FH
```

```
        LCALL   LCD_CONTROL
        LCALL   LCD_CONTROL
        LCALL   DELAY
        MOV     LCD,#0EH
        LCALL   LCD_CONTROL
        LCALL   DELAY
        MOV     LCD,#06H
        LCALL   LCD_CONTROL
        LCALL   DELAY
        MOV     LCD,#01H
        LCALL   LCD_CONTROL
        LCALL   DELAY
        MOV     LCD,#0CH
        LCALL   LCD_CONTROL
        LCALL   DELAY
        RET
;-------
LCD_CONTROL:
        LCALL   DELAY
        SETB    EN
        CLR     EN
        LCALL   DELAY
        LCALL   DELAY
        RET
;=============================================
;       Hapus Tampilan
;=============================================
CLEAR_TAMPILAN:
        CLR     EA
        CLR     RS
        MOV     LCD,#01H
        LCALL   LCD_CONTROL
        LCALL   DELAY
        RET
;=============================================
;       Baris Pertama
;=============================================
ROW_1:
        CLR     RS
        MOV     LCD,#10H
        LCALL   LCD_CONTROL
        LCALL   DELAY
        RET
;=============================================
;       Baris Kedua
;=============================================
ROW_2:
        CLR     RS
        MOV     LCD,#11000000B
        LCALL   LCD_CONTROL
```

```
        LCALL   DELAY
        RET
;=============================================
;       Prosedur Pengiriman Data
;               ke LCD
;=============================================
WRITE:
        SETB    RS
TULIS_1:
        CLR     A
        MOVC    A,@A+DPTR
        CJNE    A,#'$',TULIS_2
        SETB    EA
        NOP
        RET
TULIS_2:
        MOV     LCD,A
        LCALL   LCD_CONTROL
        INC     DPTR
        LCALL   DELAY
        LJMP    TULIS_1
;=============================================
;       PROG_TIMER KRISTAL 12MHz
;=============================================
DOLAY:
        PUSH B
        PUSH PSW
        PUSH 02H
        PUSH 03H
        MOV  R2,#200
DOLAY_1:
        MOV  R3,#0FFH
        DJNZ R3,$
        DJNZ R2,DOLAY_1
        POP  03H
        POP  02H
        POP  PSW
        POP  B
        RET
DELAY:
        PUSH B
        PUSH PSW
        PUSH 02H
        PUSH 03H
        MOV  R2,#9
DELAY_1:
        MOV  R3,#10
        DJNZ R3,$
        DJNZ R2,DELAY_1
        POP  03H
```

```
        POP   02H
        POP   PSW
        POP   B
        RET
DELAY_TAMPILAN:
        PUSH B
        PUSH PSW
        PUSH 02H
        PUSH 03H
        PUSH 04H
        MOV  R2,#10
DELAY_TAMPILAN1:
        MOV  R3,#0FFH
DELAY_TAMPILAN2:
        MOV  R4,#0FFH
        DJNZ R4,$
        DJNZ R3,DELAY_TAMPILAN2
        DJNZ R2,DELAY_TAMPILAN1
        POP  04H
        POP  03H
        POP  02H
        POP  PSW
        POP  B
        RET
TUNDA:
        PUSH B
        PUSH PSW
        PUSH 03H
        PUSH 04H
        MOV  R4,#125
LDEL_1:
        MOV  R3,#0FFH
        DJNZ R3,$
        DJNZ R4,LDEL_1
        POP  04H
        POP  03H
        POP  PSW
        POP  B
        RET
WAKTU:                  ; 0.5_s
        PUSH B
        PUSH PSW
        PUSH 02H
WAKTU_2:
        MOV         R2,#25
WAKTU_3:
        LCALL       GAP_1
        DJNZ        R2,WAKTU_3
        POP         02H
        POP         PSW
```

```
        POP   B
        RET
GAPA_1:                         ; 1_s
        PUSH B
        PUSH PSW
        PUSH 02H
        MOV  R2,#100
GAPA_2:
        LCALL       GAP_1
        DJNZ        R2,GAPA_2
        POP         02H
        POP         PSW
        POP         B
        RET
GAP_1:                          ;"65.535 - 10000 = 55535_Dec = D8EF_Hex"
        MOV  TMOD,#1            ;(t= 0,000.100s) = 0,0001s
        MOV  TH0,#0D8H
        MOV  TL0,#0EFH
        SETB TR0
GAP_2:
        NOP
        JBC   TF0,GAP_3
        SJMP  GAP_2
GAP_3:
        RET
GUN_1:                          ;"65.535 - XXX = 65435_Dec = FF9B_Hex"
        MOV  TMOD,#1            ;(t= 0,000.100s) = 0,0001s
        MOV  TH0,#0FFH
        MOV  TL0,#9BH
        SETB TR0
GUN_2:
        NOP
        JBC   TF0,GUN_3
        SJMP  GUN_2
GUN_3:
        RET
;=======================================
;       DATA_MEMORY
;=======================================
ANGKA:
        DB
        03H,02H,01H,00H,07H,06,05H,04H,0BH,0AH,09H,08H,0FH,0EH,0DH,0CH
HEADER_1:
        DB    '"TAMPILAN UTAMA"$'
HEADER_2:
        DB    20H,20H,20H,20H,20H,20H,20H,20H
        DB    20H,20H,20H,20H,20H,20H,20H,20H
        DB    '* PENYIRAMAN OTOMATIS PADA TANAMAN KEDELAI * - Oleh:
ACHMAT EBIT - NIM: 00.107138 - '
        DB    'Fak/Jur: TEK. ELEKTRO/ELEKTRONIKA - * ITN MALANG *'
```

```
        DB      20H,20H,20H,20H,20H,20H,20H,20H
        DB      20H,20H,20H,20H,20H,20H,20H,20H
        DB      20H,20H,20H,20H,20H,20H,20H,20H
        DB      20H,20H,20H,20H,20H,20H,20H,20H
        DB      20H,20H,20H,20H,20H,20H,20H,20H
        DB      20H,20H,20H,20H,20H,20H,20H,20H
DAT_SP:
        DB      'HARI: $'
MINGGU:
        DB      'Minggu $'
SENIN:
        DB      'Senin  $'
SELASA:
        DB      'Selasa $'
RABU:
        DB      'Rabu   $'
KAMIS:
        DB      'Kamis  $'
JUMAT:
        DB      'Jum"at $'
SABTU:
        DB      'Sabtu  $'
HEAD_1:
        DB      '    MingguSenin SelasaRabu  Kamis Jum"atSabtu '
HEAD_2:
        DB      '     Januari  Februari Maret   April   Mei     Juni    Juli    Agustus
SeptemberOktober  Nopember Desember '
PLS_1:
        DB      ' * Masuk Menu * $'
PLS_2:
        DB      '-- Tekan: "F" --$'
PLS_3:
        DB      'TUNGGU!!! Sedang$'
PLS_4:
        DB      ' Simpan Program $'
PLS_5:
        DB      'Lihat Timer: "A"$'
PLS_6:
        DB      'Ganti Timer: "B"$'
MENO_3:
        DB      'MENU SET: 1.JAM $'
MENO_4:
        DB      '2.LEMBAB  3.EXIT$'
MAS_1:
        DB      'Nomor Sensor: $'
MAS_2:
        DB      'Kelembaban  : $'
MAS_3:
        DB      'KEY: $'
```

```
MAS_4:
        DB      'ADC: $'
JAM_1:
        DB      'SET TIMER "24HR"$'
JAM_2:
        DB      'JAM: $'
JAM_3:
        DB      'MENIT: $'
JAM_4:
        DB      'DETIK: $'
JAM_5:
        DB      'HARI: $'
JAM_6:
        DB      'BULAN: $'
JAM_7:
        DB      'TANGGAL: $'
JAM_8:
        DB      'TAHUN: $'
        END
```

Features

- Compatible with MCS®51 Products
- 8K Bytes of In-System Reprogrammable Downloadable Flash Memory
  - SPI Serial Interface for Program Downloading
  - Endurance: 1,000 Write/Erase Cycles
- 2K Bytes EEPROM
  - Endurance: 100,000 Write/Erase Cycles
- 4V to 6V Operating Range
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Nine Interrupt Sources
- Programmable UART Serial Channel
- SPI Serial Interface
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down
- Programmable Watchdog Timer
- Dual Data Pointer
- Power-off Flag

Description

The AT89S8252 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of downloadable Flash programmable and erasable read-only memory and 2K bytes of EEPROM. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip downloadable Flash allows the program memory to be reprogrammed In-System through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with downloadable Flash on a monolithic chip, the Atmel AT89S8252 is a powerful microcontroller, which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S8252 provides the following standard features: 8K bytes of downloadable Flash, 2K bytes of EEPROM, 256 bytes of RAM, 32 I/O lines, programmable watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S8252 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

The downloadable Flash can be changed a single byte at a time and is accessible through the SPI serial interface. Holding RESET active forces the SPI bus into a serial programming interface and allows the program memory to be written to or read from unless lock bits have been activated.

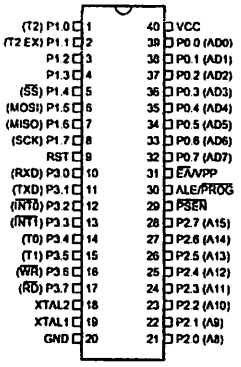
8-bit Microcontroller with 8K Bytes Flash

AT89S8252

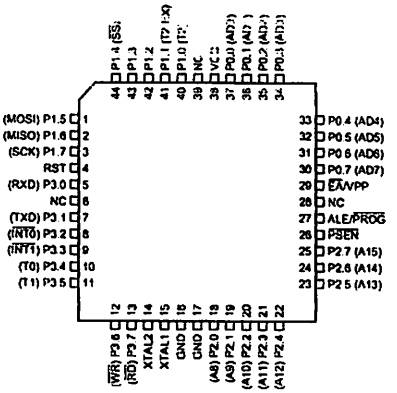
0401F–MICRO–11/03

Pin Configurations

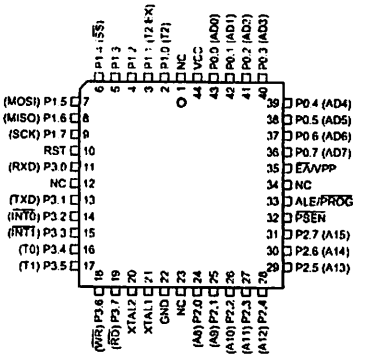
PDIP



TQFP



PLCC



Pin Description

VCC Supply voltage.

GND Ground.

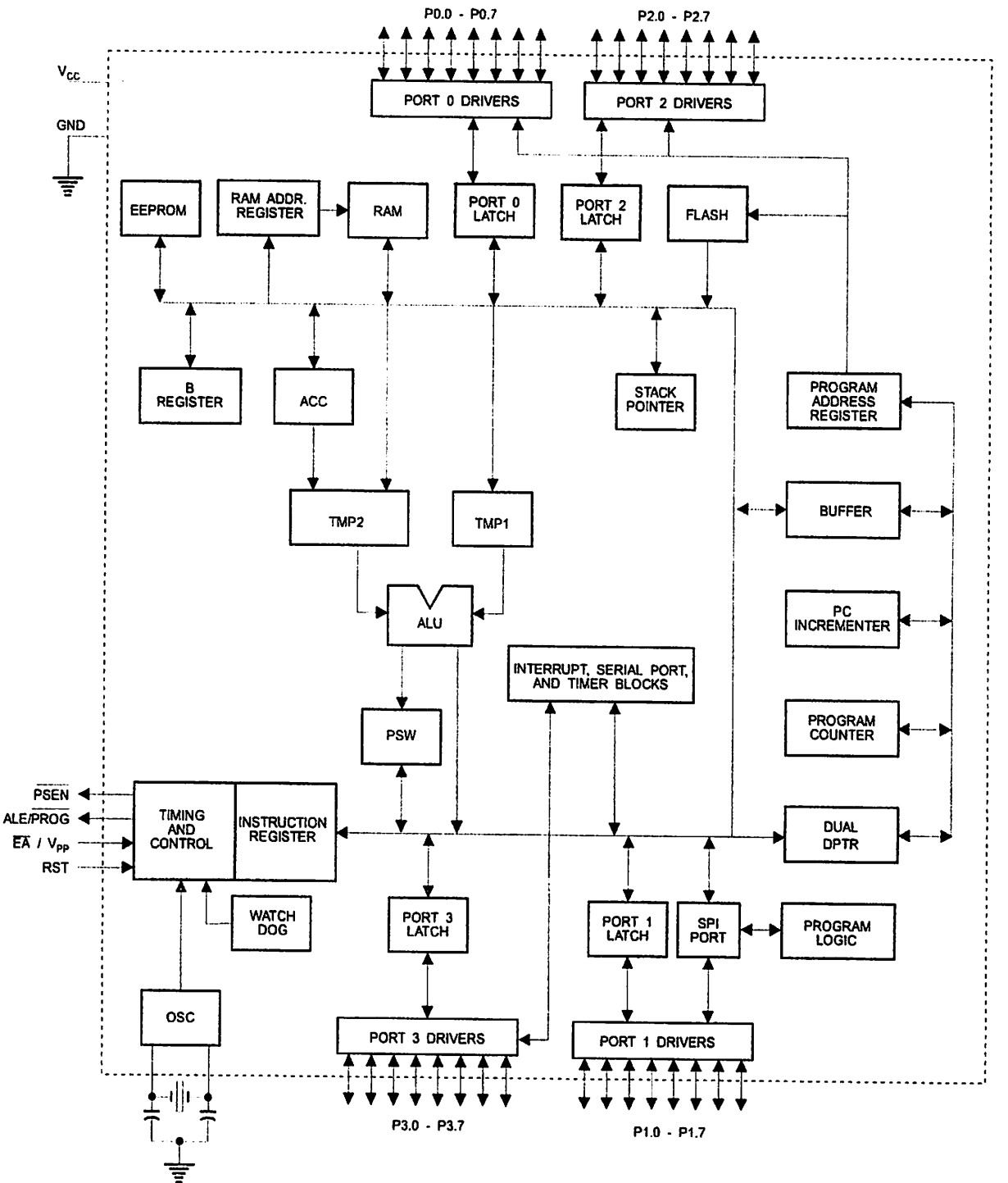
Port 0 Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups.

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

Port 1 Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. 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 pull-ups 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 pull-ups.

Block Diagram



Some Port 1 pins provide additional functions. P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX), respectively.

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

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)
P1.4	$\overline{SS}$ (Slave port select input)
P1.5	MOSI (Master data output, slave data input pin for SPI channel)
P1.6	MISO (Master data input, slave data output pin for SPI channel)
P1.7	SCK (Master clock output, slave clock input pin for SPI channel)

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pull-ups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups 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 pull-ups.

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

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

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 program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{EA}$ will be internally latched on reset.

$\overline{EA}$ should be strapped to V_{CC} for internal program executions. This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming when 12-volt programming is selected.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 1.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

User software 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.

Table 1. AT89S8252 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DF H
0D0H	PSW 00000000					SPCR 000001XX			0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CF 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						WMCON 00000010		97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

Table 2. T2CON – Timer/Counter 2 Control Register

T2CON Address = 0C8H — Reset Value = 0000 0000B

Bit Addressable

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

Symbol	Function
TF2	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.
EXF2	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).
RCLK	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.
TCLK	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.
EXEN2	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.
TR2	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
C/$\overline{T2}$	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).
CP/$\overline{RL2}$	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.

Watchdog and Memory Control Register The WMCON register contains control bits for the Watchdog Timer (shown 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.

Table 3. WMCON—Watchdog and Memory Control Register

WMCON Address = 96H — Reset Value = 0000 0010B

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

Symbol	Function
PS2 PS1 PS0	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.
EEMWE	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.
EEMEN	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.
DPS	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1
WDTRST RDY/$\overline{BSY}$	Watchdog Timer Reset and EEPROM Ready/$\overline{Busy}$ 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/$\overline{BSY}$ flag in a Read-Only mode during EEPROM write. RDY/$\overline{BSY}$ = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/$\overline{BSY}$ bit equals "0" and is automatically reset to "1" when programming is completed.
WDTEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

SPI Registers Control and status bits for the Serial Peripheral Interface are contained in registers SPCR (shown in Table 4) and SPSR (shown in Table 5). The SPI data bits are contained in the SPDR register. Writing the SPI data register during serial data transfer sets the Write Collision bit, WCOL, in the SPSR register. The SPDR is double buffered for writing and the values in SPDR are not changed by Reset.

Interrupt Registers The global interrupt enable bit and the individual interrupt enable bits are in the IE register. In addition, the individual interrupt enable bit for the SPI is in the SPCR register. Two priorities can be set for each of the six interrupt 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.

Table 4. SPCR – SPI Control Register

SPCR Address = D5H — Reset Value = 0000 01XXB

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

Symbol	Function
SPIE	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.
SPE	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.
DORD	Data Order. DORD = 1 selects LSB first data transmission. DORD = 0 selects MSB first data transmission.
MSTR	Master/Slave Select. MSTR = 1 selects Master SPI mode. MSTR = 0 selects Slave SPI mode.
CPOL	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.
CPHA	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.
SPR0 SPR1	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

Table 5. SPSR – SPI Status Register

SPSR Address = AAH — Reset Value = 00XX XXXXB

Bit	SPIF	WCOL	–	–	–	–	–	–
	7	6	5	4	3	2	1	0

Symbol	Function
SPIF	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 reading/writing the SPI data register.
WCOL	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.

Table 6. SPDR – SPI Data Register

SPDR Address = 86H — Reset Value = unchanged

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

Data Memory – EEPROM and RAM

The AT89S8252 implements 2K bytes of on-chip EEPROM for data storage and 256 bytes of RAM. The upper 128 bytes of RAM occupy a parallel space to the Special Function Registers. That means the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space.

When an instruction accesses an internal location above address 7FH, the address mode used in the instruction specifies whether the CPU accesses the upper 128 bytes of RAM or the SFR space. Instructions that use direct addressing access SFR space.

For example, the following direct addressing instruction accesses the SFR at location 0A0H (which is P2).

```
MOV 0A0H, #data
```

Instructions that use indirect addressing access the upper 128 bytes of RAM. For example, the following indirect addressing instruction, where R0 contains 0A0H, accesses the data byte at address 0A0H, rather than P2 (whose address is 0A0H).

```
MOV @R0, #data
```

Note that stack operations are examples of indirect addressing, so the upper 128 bytes of data RAM are available as stack space.

The on-chip EEPROM data memory is selected by setting the EEMEN bit in the WMCON register at SFR address location 96H. The EEPROM address range is from 000H to 7FFH. The MOVX instructions are used to access the EEPROM. To access off-chip data memory with the MOVX instructions, the EEMEN bit needs to be set to "0".

The EEMWE bit in the WMCON register needs to be set to "1" before any byte location in the EEPROM can be written. User software should reset EEMWE bit to "0" if no further EEPROM write is required. EEPROM write cycles in the serial programming mode are self-timed and typically take 2.5 ms. The progress of EEPROM write can be monitored by reading the RDY/$\overline{BSY}$ bit (read-only) in SFR WMCON. RDY/$\overline{BSY}$ = 0 means

programming is still in progress and RDY/$\overline{BSY}$ = 1 means EEPROM write cycle is completed and another write cycle can be initiated.

In addition, during EEPROM programming, an attempted read from the EEPROM will fetch the byte being written with the MSB complemented. Once the write cycle is completed, true data are valid at all bit locations.

Programmable Watchdog Timer

The programmable Watchdog Timer (WDT) operates from an independent internal oscillator. The prescaler bits, PS0, PS1 and PS2 in SFR WMCON are used to set the period of the Watchdog Timer from 16 ms to 2048 ms. The available timer periods 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 and AT89C52. For further information on the timers' operation, refer to the Atmel web site (http://www.atmel.com). From the home page, select "Products", then "Microcontrollers, then "8051-Architecture". Click on "Documentation", then on "Other Documents". Open the document "AT89 Series Hardware Description".

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 the transition was detected.

Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure that a given level is sampled at least once before it changes, the level should be held for at least one full machine cycle.

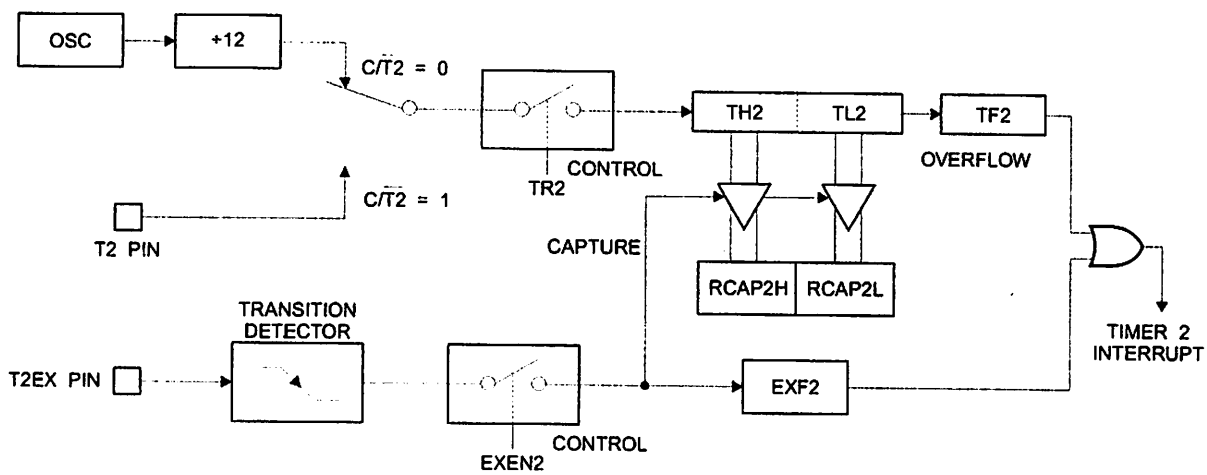
Table 8. Timer 2 Operating Modes

RCLK + 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 I-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.

Figure 1. Timer 2 in Capture Mode



Auto-reload (Up or Down Counter)

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

Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in RCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit 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.

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

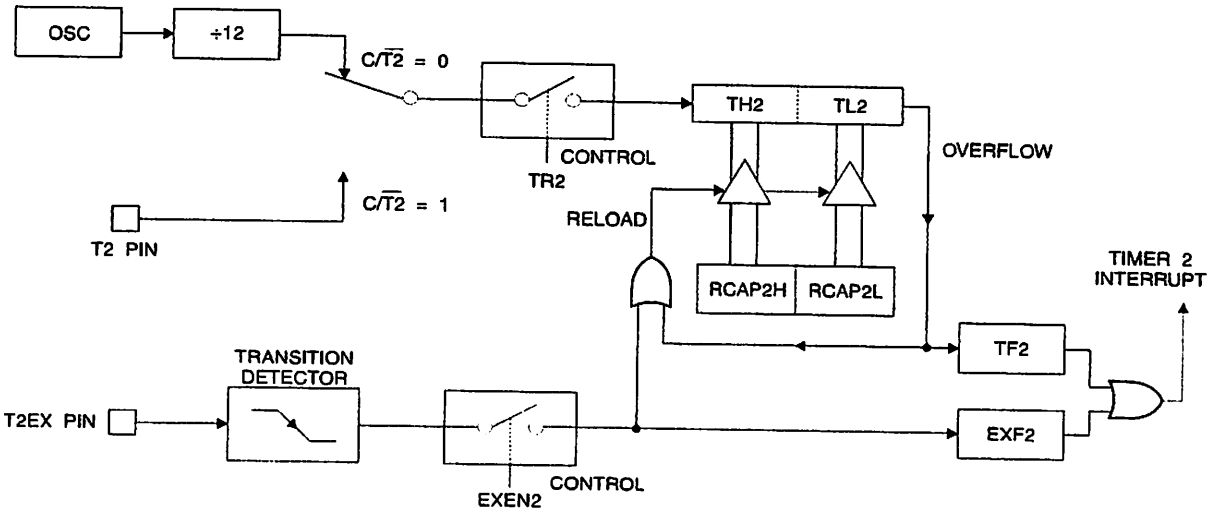


Table 9. T2MOD – Timer 2 Mode Control Register

T2MOD Address = 0C9H — Reset Value = XXXX XX00B

Not Bit Addressable

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

Symbol	Function
–	Not implemented, reserved for future use.
T2OE	Timer 2 Output Enable bit.
DCEN	When set, this bit allows Timer 2 to be configured as an up/down counter.

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

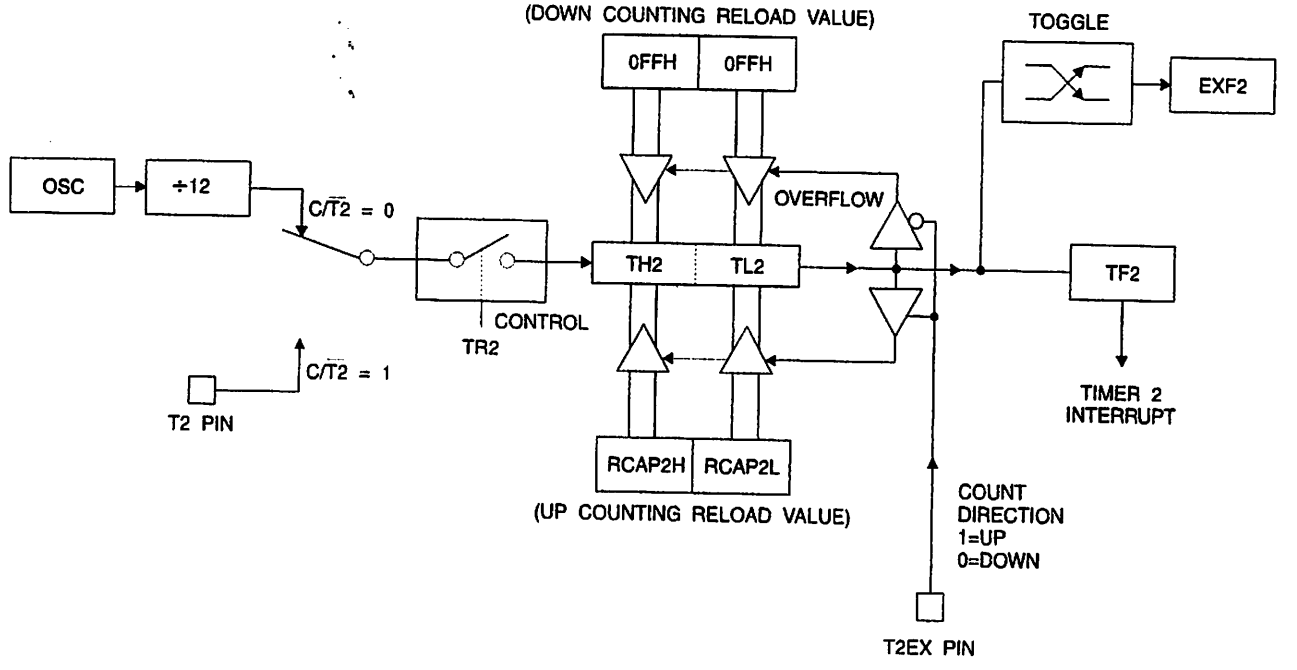
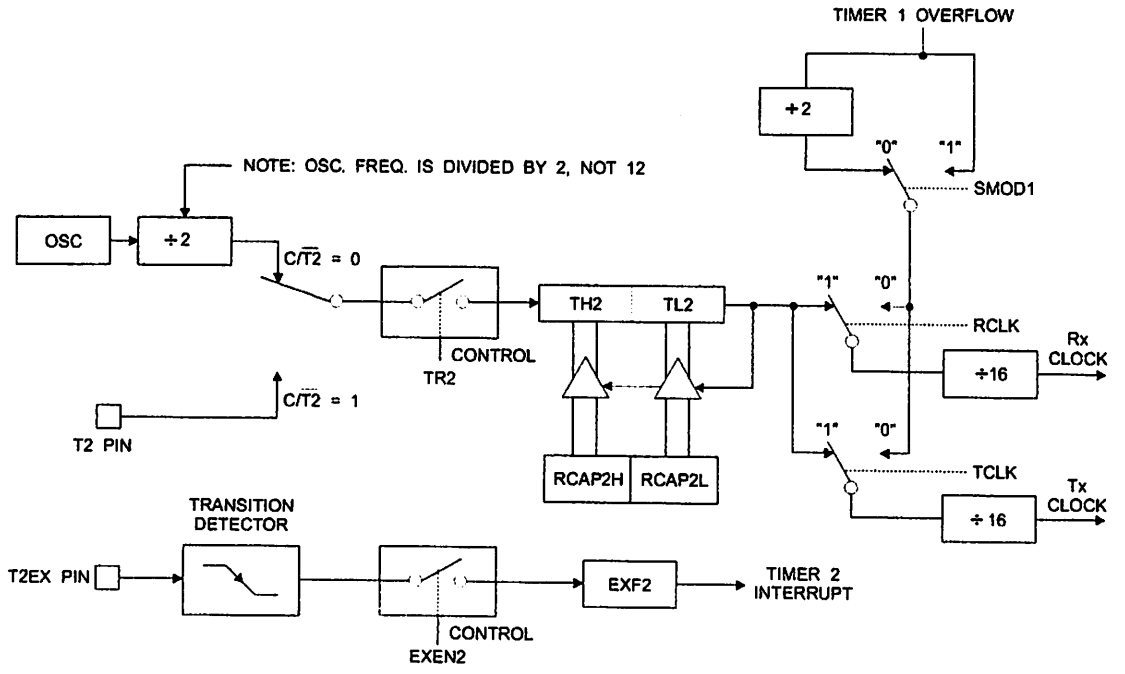


Figure 4. Timer 2 in Baud Rate Generator Mode



Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (Table 2). Note that the baud rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for the other function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate according to the following equation.

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

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

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

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

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, 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/0 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 (for 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.

Figure 5. Timer 2 in Clock-out Mode

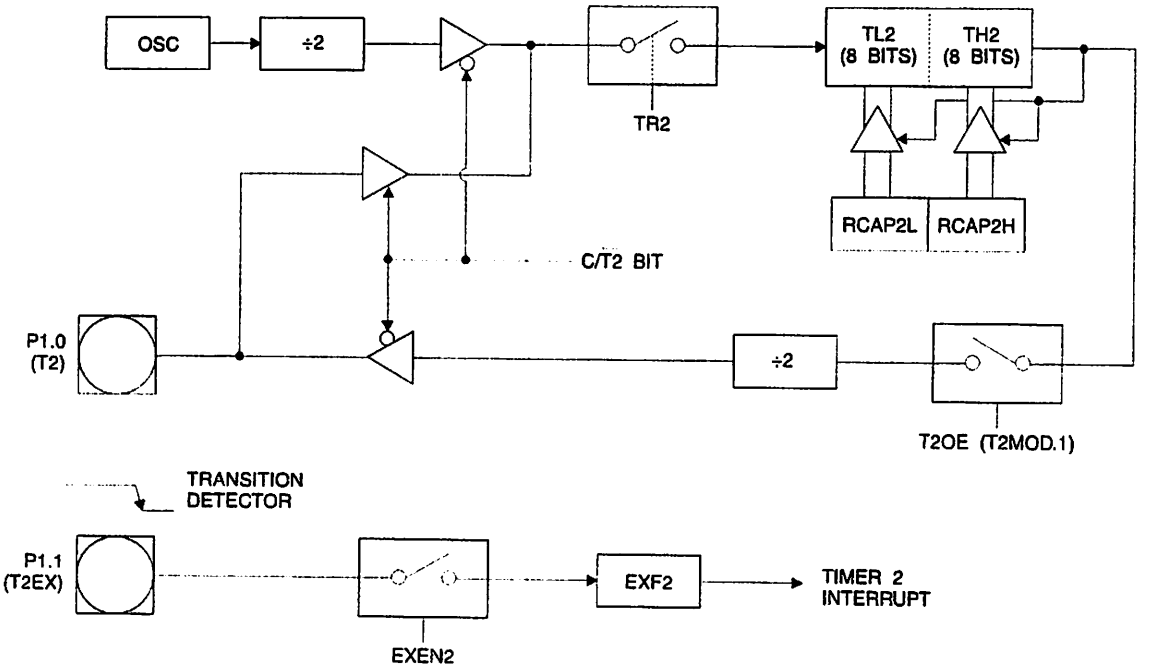
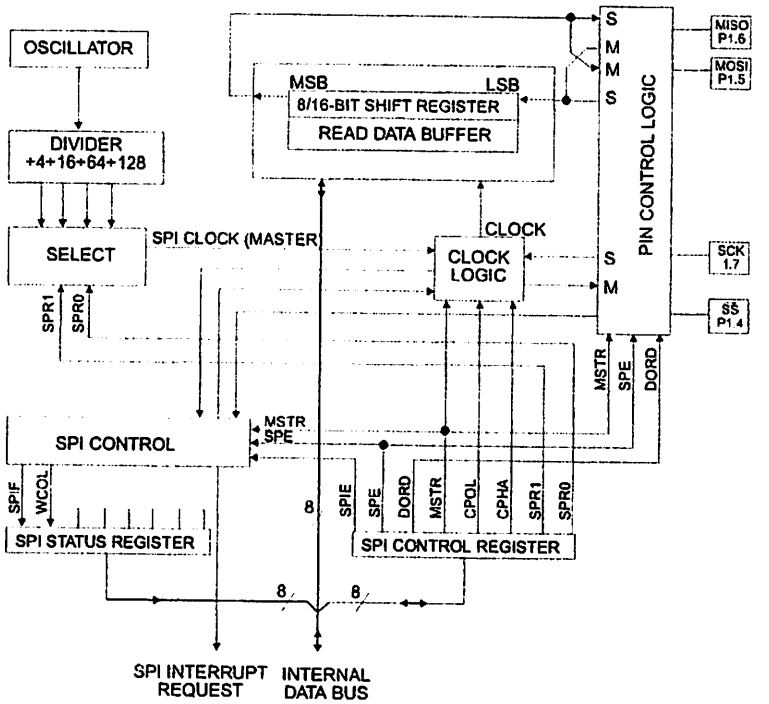


Figure 6. SPI Block Diagram



UART

The UART in the AT89S8252 operates the same way as the UART in the AT89C51 and AT89C52. For further information on the UART operation, refer to the Atmel web site (http://www.atmel.com). From the home page, select "Products", then "Microcontrollers, then "8051-Architecture". Click on "Documentation", then on "Other Documents". Open the document "AT89 Series Hardware Description".

Serial Peripheral Interface

The serial peripheral interface (SPI) allows high-speed synchronous data transfer between the AT89S8252 and peripheral devices or between several AT89S8252 devices. The AT89S8252 SPI features include the following:

- Full-Duplex, 3-Wire Synchronous Data Transfer
- Master or Slave Operation
- 1.5 MHz Bit Frequency (max.)
- LSB First or MSB First Data Transfer
- Four Programmable Bit Rates
- End of 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.

Figure 7. SPI Master-slave Interconnection

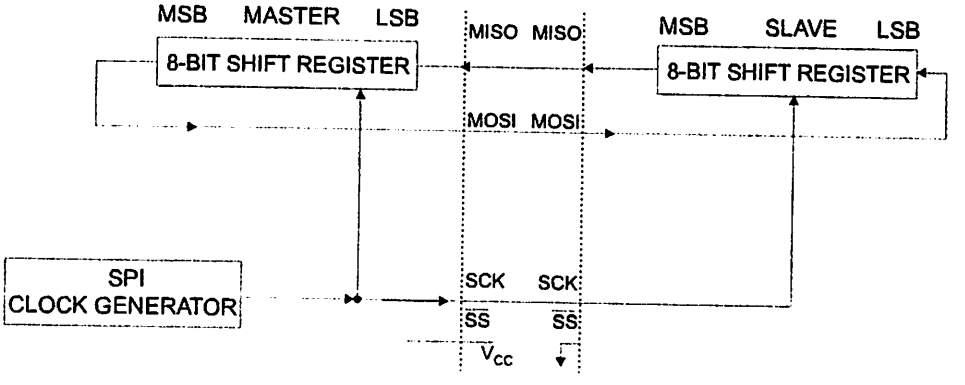
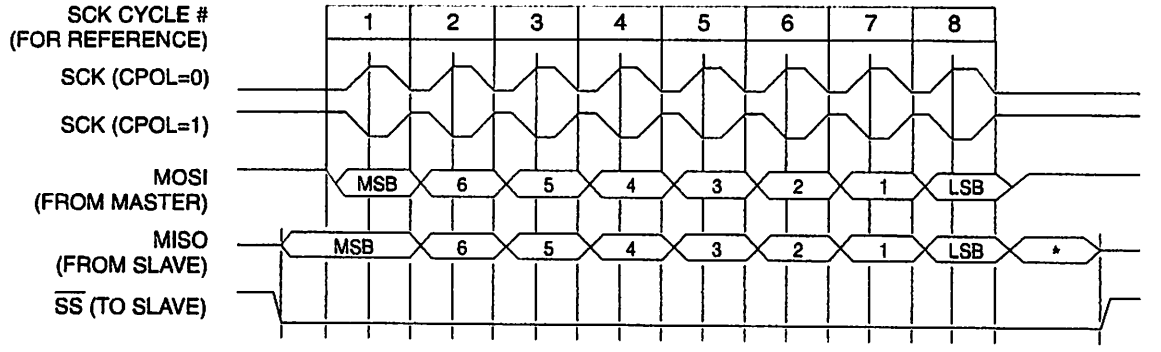
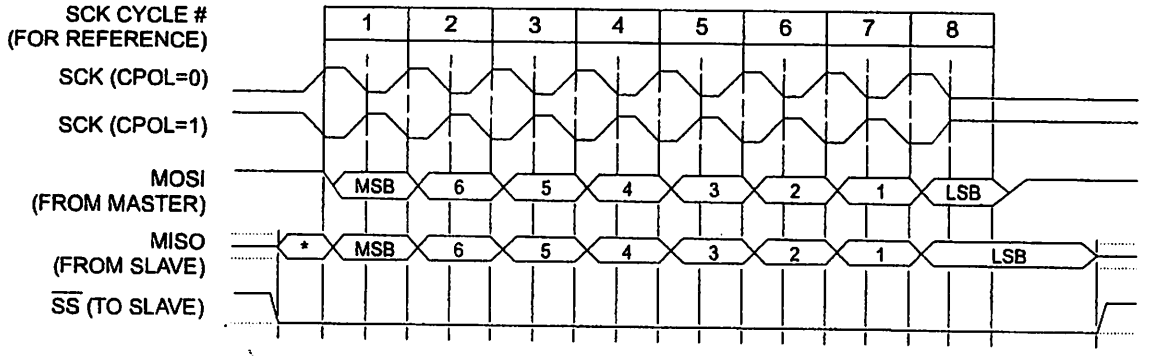


Figure 8. SPI transfer Format with CPHA = 0



Note: *Not defined but normally MSB of character just received

Figure 9. SPI Transfer Format with CPHA = 1



Note: *Not defined but normally LSB of previously transmitted character.

Interrupts

The AT89S8252 has a total of six interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (Timers 0, 1, and 2), and the serial port interrupt. These interrupts are all shown in Figure 10.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE. IE also contains a global disable bit, EA, which disables all interrupts at once.

Note that Table 10 shows that bit position IE.6 is unimplemented. In the AT89C51, bit position IE.5 is also unimplemented. User software should not write 1s to these bit positions, since they may be used in future AT89 products.

Timer 2 interrupt is generated by the logical OR of bits TF2 and EXF2 in register T2CON. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and that bit will have to be cleared in software.

The Timer 0 and Timer 1 flags, TF0 and TF1, are set at S5P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle. However, the Timer 2 flag, TF2, is set at S2P2 and is polled in the same cycle in which the timer overflows.

Table 10. Interrupt Enable (IE) Register

(MSB)(LSB)

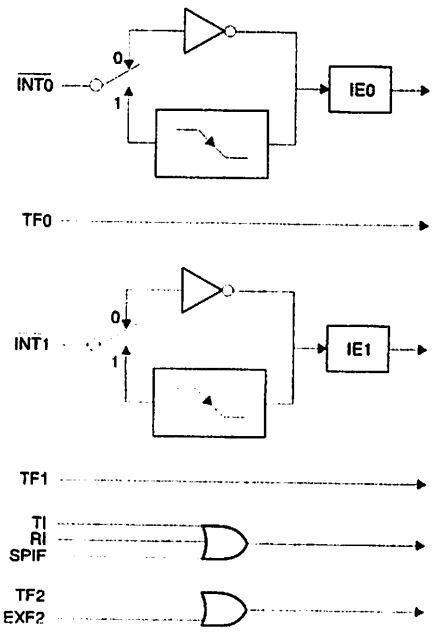
EA	–	ET2	ES	ET1	EX1	ET0	EX0

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.		

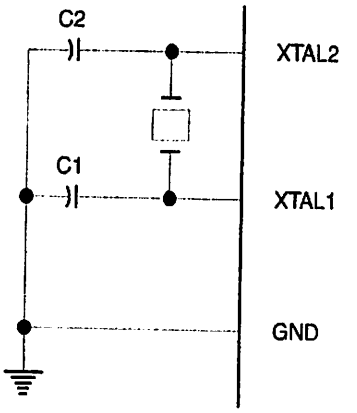
Figure 10. Interrupt Sources



Oscillator Characteristics

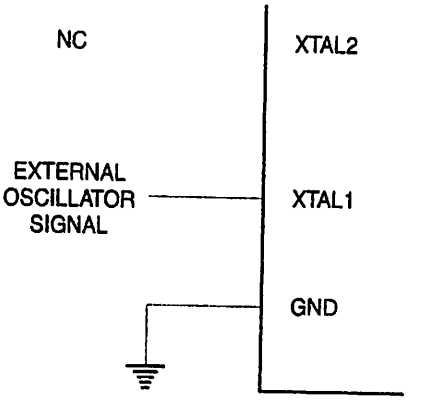
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 12. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration



Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

Note that when idle mode is terminated by a hardware reset, the 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.

Status of External Pins During Idle and Power-down Modes

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

Power-down Mode

In the power-down mode, the oscillator is stopped and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. Exit from power-down can be initiated either by a hardware reset or by an enabled external interrupt. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

To exit power-down via an interrupt, the external interrupt must be enabled as level sensitive before entering power-down. The interrupt service routine starts at 16 ms (nominal) after 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.

Lock Bit Protection Modes[1][2]

Program Lock Bits				
	LB1	LB2	LB3	Protection Type
1	U	U	U	No internal memory lock feature.
2	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.
3	P	P	U	Same as Mode 2, but parallel or serial verify are also disabled.
4	P	P	P	Same as Mode 3, but external execution is also disabled.

Notes: 1. U = Unprogrammed
2. P = Programmed

Programming the Flash and EEPROM

Atmel's AT89S8252 Flash Microcontroller offers 8K bytes of in-system reprogrammable Flash Code memory and 2K bytes of EEPROM Data memory.

The AT89S8252 is normally shipped with the on-chip Flash Code and EEPROM Data memory arrays in the erased state (i.e. contents = FFH) and ready to be programmed. This device supports a High-voltage (12-V V_{PP}) Parallel programming mode and a Low-voltage (5-V V_{CC}) Serial programming mode. The serial programming mode provides a convenient way to reprogram the AT89S8252 inside the user's system. The parallel programming mode is compatible with conventional third party Flash or EPROM programmers.

The Code and Data memory arrays are mapped via separate address spaces in the serial programming mode. In the parallel programming mode, the two arrays occupy one contiguous address space: 0000H to 1FFFH for the Code array and 2000H to 27FFH for the Data array.

The Code and Data memory arrays on the AT89S8252 are programmed byte-by-byte in either programming mode. An auto-erase cycle is provided with the self-timed programming operation in the serial programming mode. There is no need to perform the Chip Erase operation to reprogram any memory location in the serial programming mode unless any of the lock bits have been programmed.

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

Parallel Programming Algorithm: To program and verify the AT89S8252 in the parallel programming mode, the following sequence is recommended:

1. Power-up sequence:
   Apply power between V_{CC} and GND pins.
   Set RST pin to "H".
   Apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Set $\overline{PSEN}$ pin to "L"
   ALE pin to "H"
   $\overline{EA}$ pin to "H" and all other pins to "H".
3. Apply the appropriate combination of "H" or "L" logic levels to pins P2.6, P2.7, P3.6, P3.7 to select one of the programming operations shown in the Flash Programming Modes table.
4. Apply the desired byte address to pins P1.0 to P1.7 and P2.0 to P2.5.
   Apply data to pins P0.0 to P0.7 for Write Code operation.
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.

Data Polling: The AT89S8252 features DATA Polling to indicate the end of a byte write cycle. During a byte 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. DATA Polling may begin any time after a write cycle has been initiated.

Ready/Busy: The progress of byte programming in the parallel programming mode can also be monitored by the RDY/BSY output signal. Pin P3.4 is pulled Low after ALE goes High during programming to indicate 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/PROG low for 10 ms. The Code and Data arrays are written with all "1"s in the Chip Erase operation.

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

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

Serial Programming Fuse: A programmable fuse is available to disable Serial Programming if the user needs maximum system security. The Serial Programming Fuse can only be programmed or erased in the Parallel Programming Mode.

The AT89S8252 is shipped with the Serial Programming Mode enabled.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows:

(030H) = 1EH indicates manufactured by Atmel
(031H) = 72H indicates 89S8252

Programming Interface

Every code byte in the Flash and EEPROM arrays can be written, and the entire array can be erased, by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

Most worldwide major programming vendors offer support for the Atmel AT89 microcontroller series. Please contact your local programming vendor for the appropriate software revision.

Serial Downloading

Both the Code and Data memory arrays can be programmed using the serial SPI bus while RST is pulled to V_{CC}. The serial interface consists of pins SCK, MOSI (input) and MISO (output). After RST is set high, the Programming Enable instruction needs to be executed first before program/erase operations can be executed.

An auto-erase cycle is built into the self-timed programming operation (in the serial mode ONLY) and there is no need to first execute the Chip Erase instruction unless any of the lock bits have been programmed. The Chip Erase operation turns the content of every memory location in both the Code and Data arrays into FFH.

The Code and Data memory arrays have separate address spaces:

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.
6. 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:

Instruction Set

Instruction	Input Format			Operation
	Byte 1	Byte 2	Byte 3	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	Enable serial programming interface after RST goes high.
Chip Erase	1010 1100	xxxx x100	xxxx xxxx	Chip erase both 8K & 2K memory arrays.
Read 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.
Write 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.
Read 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.
Write Data Memory	00aa a110	low addr	data in	Write data to Data memory location at selected address.
Write Lock Bits	1010 1100	LB1 LB2 LB3 x x111	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

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

Flash and EEPROM Parallel Programming Modes

Mode	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
Serial Prog. Modes	H	h[1]	h[1]	x						
Chip Erase	H	L	⌐‾⌐ (2)	12V	H	L	L	L	X	X
Write (10K bytes) Memory	H	L	⌐‾⌐	12V	L	H	H	H	DIN	ADDR
Read (10K bytes) Memory	H	L	H	12V	L	L	H	H	DOUT	ADDR
Write Lock 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
Read Lock 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
Read Atmel Code	H	L	H	12V	L	L	L	L	DOUT	30H
Read Device Code	H	L	H	12V	L	L	L	L	DOUT	31H
Serial Prog. Enable	H	L	⌐‾⌐ (2)	12V	L	H	L	H	P0.0 = 0	X
Serial Prog. Disable	H	L	⌐‾⌐ (2)	12V	L	H	L	H	P0.0 = 1	X
Read Serial Prog. Fuse	H	L	H	12V	H	H	L	H	@P0.0	X

Notes:
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

Figure 13. Programming the Flash/EEPROM Memory

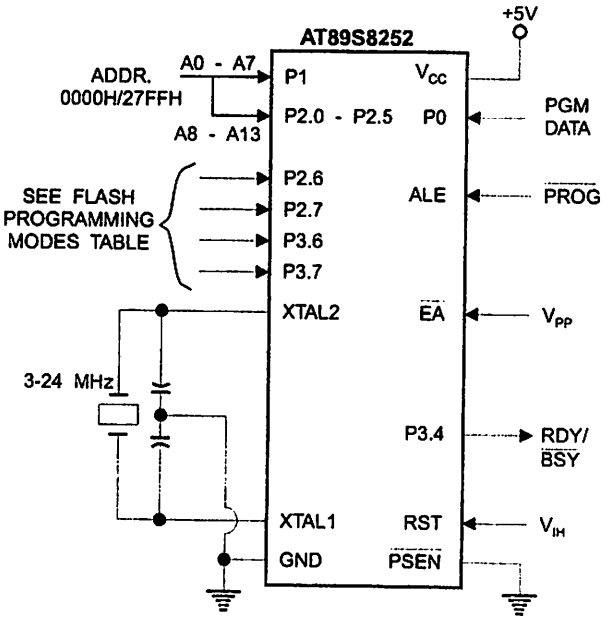


Figure 14. Verifying the Flash/EEPROM Memory

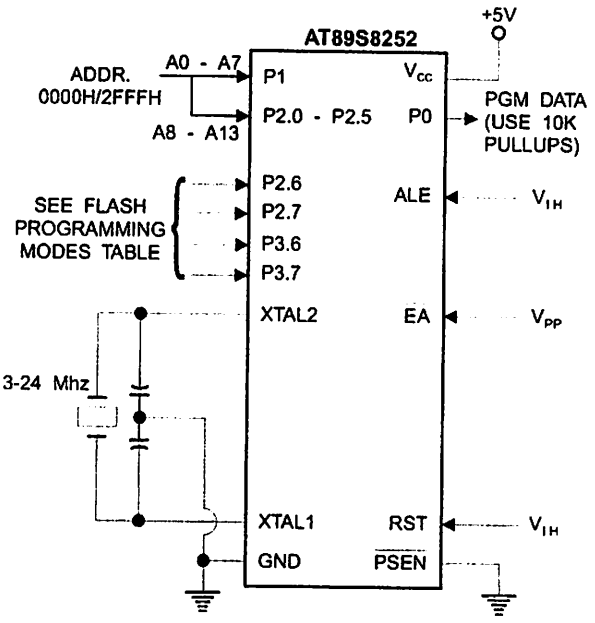
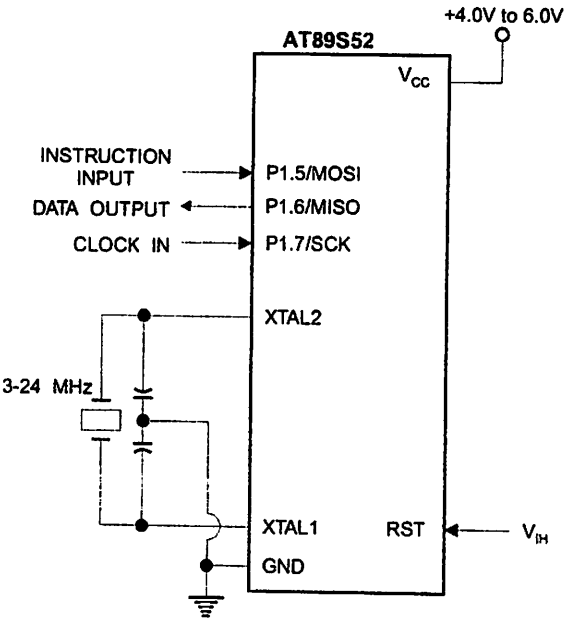


Figure 15. Flash/EEPROM Serial Downloading

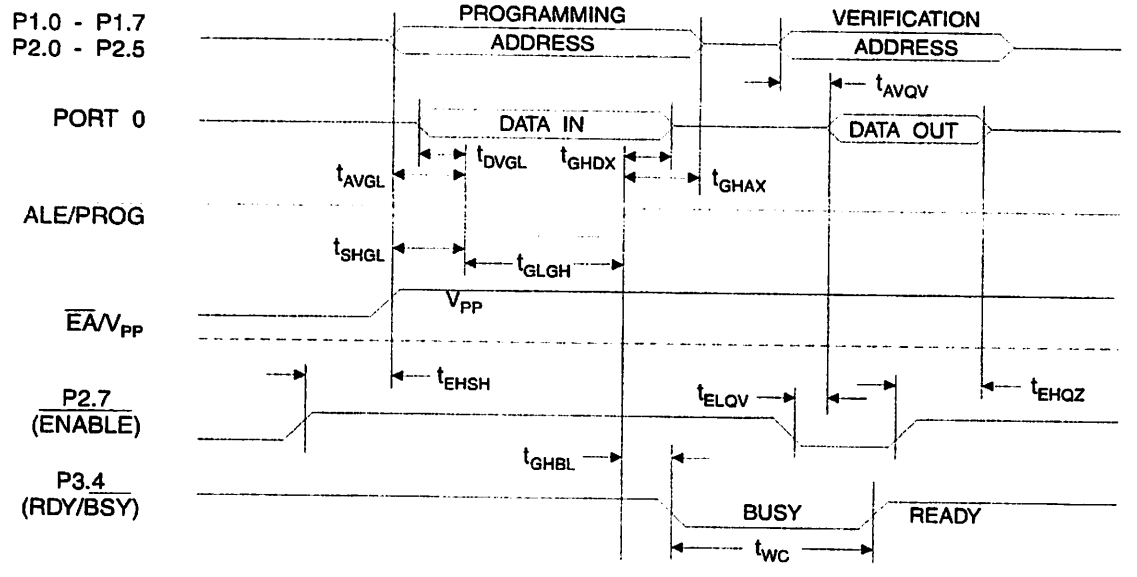


Flash Programming and Verification Characteristics – Parallel Mode

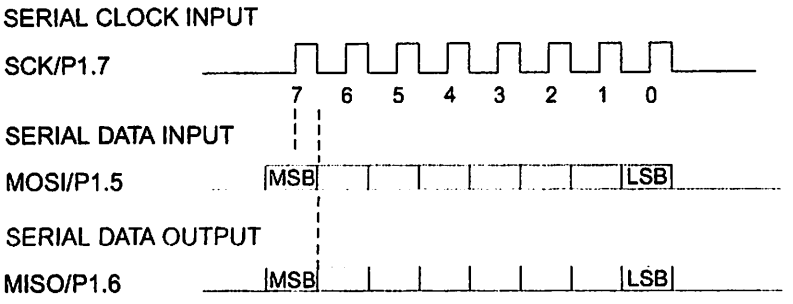
T_A = 0°C to 70°C, V_{CC} = 5.0V ±10%

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{EHSH}	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	Data Float after $\overline{ENABLE}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
t_{WC}	Byte Write Cycle Time		2.0	ms

Flash/EEPROM Programming and Verification Waveforms – Parallel Mode



Serial Downloading Waveforms



Serial Programming Characteristics

Figure 16. Serial Programming Timing

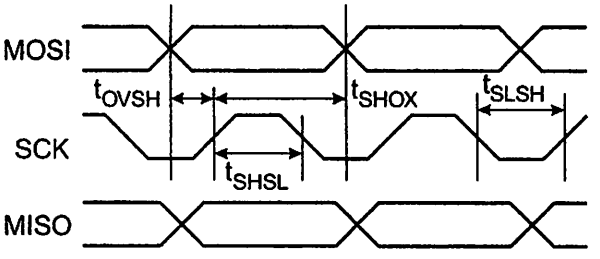


Table 11. Serial Programming Characteristics, T_A = -40° C to 85° C, V_{CC} = 4.0 - 6.0V (Unless Otherwise Noted)

Symbol	Parameter	Min	Typ	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0		24	MHz
t_{CLCL}	Oscillator Period	41.6			ns
t_{SHSL}	SCK Pulse Width High	24 t_{CLCL}			ns
t_{SLSH}	SCK Pulse Width Low	24 t_{CLCL}			ns
t_{OVSH}	MOSI Setup to SCK High	t_{CLCL}			ns
t_{SHOX}	MOSI Hold after SCK High	2 t_{CLCL}			ns

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output 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.

DC Characteristics

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

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC} - 0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC} - 0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC} + 0.5	V
V_{OL}	Output Low-voltage (1) (Ports 1,2,3)	I_{OL} = 1.6 mA		0.5	V
V_{OL1}	Output Low-voltage (1) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.5	V
V_{OH}	Output High-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
V_{OH1}	Output High-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
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ±10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pull-down Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	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

Notes:
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

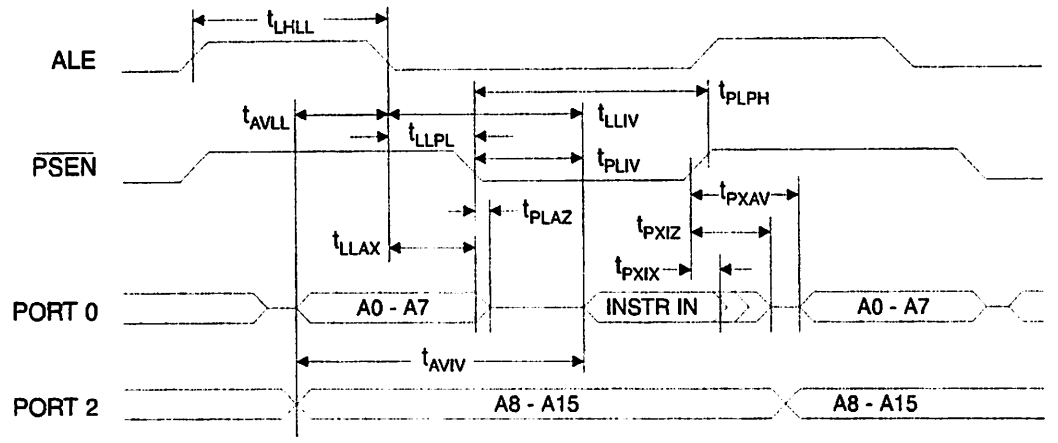
AC Characteristics

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

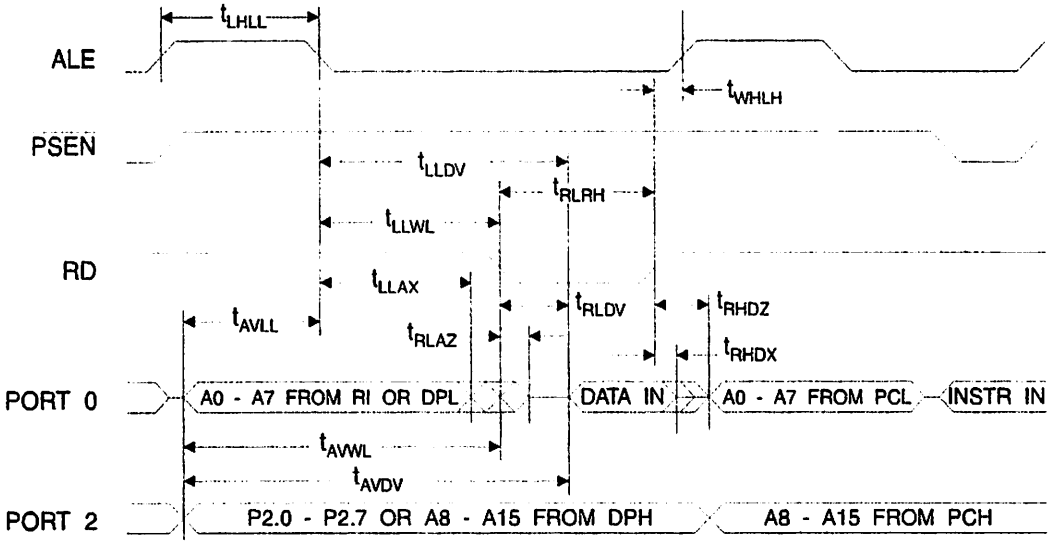
External Program and Data Memory Characteristics

Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{LHLL}	ALE Pulse Width	$2t_{CLCL}$ - 40		ns
t_{AVLL}	Address Valid to ALE Low	t_{CLCL} - 13		ns
t_{LLAX}	Address Hold after ALE Low	t_{CLCL} - 20		ns
t_{LLIV}	ALE Low to Valid Instruction In		$4t_{CLCL}$ - 65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	t_{CLCL} - 13		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	$3t_{CLCL}$ - 20		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		$3t_{CLCL}$ - 45	ns
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		ns
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		t_{CLCL} - 10	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	t_{CLCL} - 8		ns
t_{AVIV}	Address to Valid Instruction In		$5t_{CLCL}$ - 55	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	$6t_{CLCL}$ - 100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	$6t_{CLCL}$ - 100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		$5t_{CLCL}$ - 90	ns
t_{RHDX}	Data Hold after $\overline{RD}$	0		ns
t_{RHDZ}	Data Float after $\overline{RD}$		$2t_{CLCL}$ - 28	ns
t_{LLDV}	ALE Low to Valid Data In		$8t_{CLCL}$ - 150	ns
t_{AVDV}	Address to Valid Data In		$9t_{CLCL}$ - 165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	$3t_{CLCL}$ - 50	$3t_{CLCL}$ + 50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	$4t_{CLCL}$ - 75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	t_{CLCL} - 20		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	$7t_{CLCL}$ - 120		ns
t_{WHQX}	Data Hold after $\overline{WR}$	t_{CLCL} - 20		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	t_{CLCL} - 20	t_{CLCL} + 25	ns

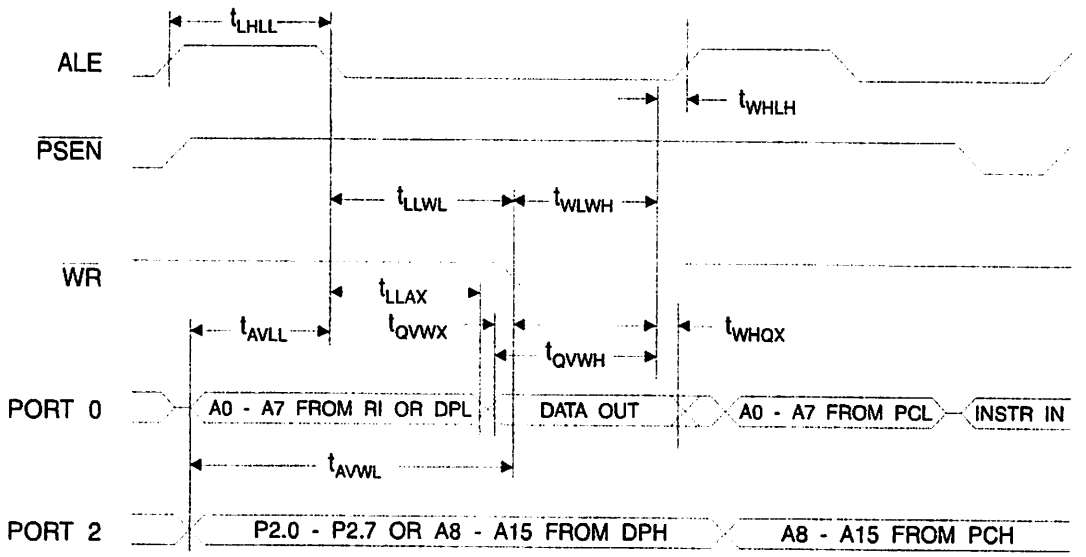
External Program Memory Read Cycle



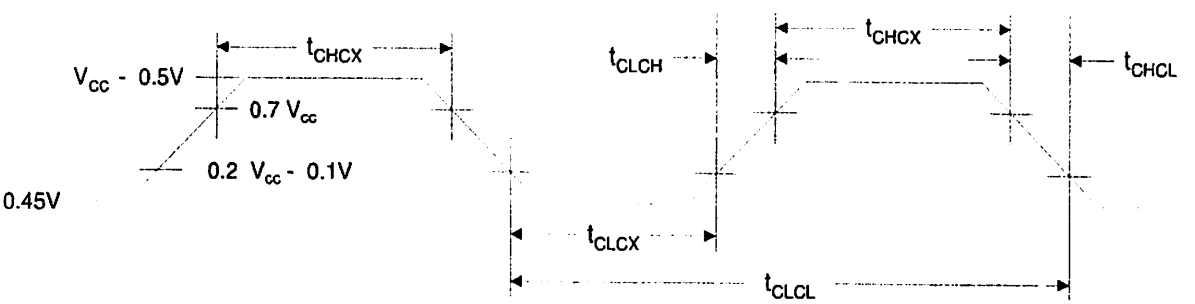
External Data Memory Read Cycle



xternal Data Memory Write Cycle



xternal Clock Drive Waveforms



xternal Clock Drive

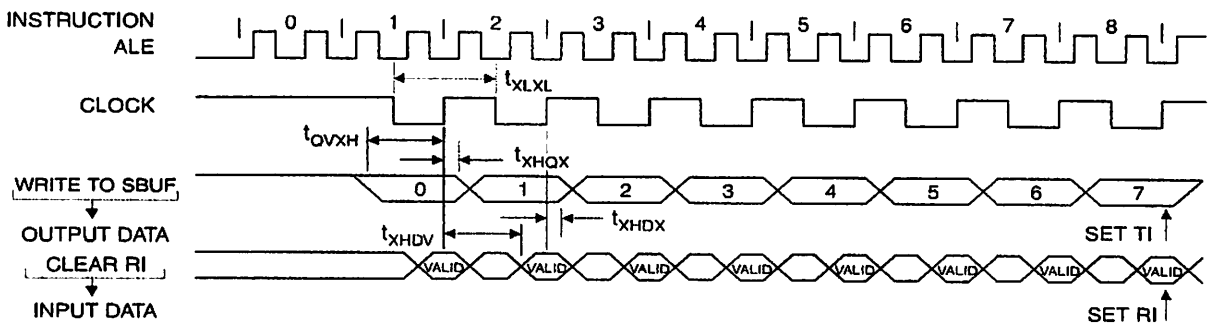
ymbol	Parameter	V_{CC} = 4.0V to 6.0V		Units
		Min	Max	
t_{CLCL}	Oscillator Frequency	0	24	MHz
LCL	Clock Period	41.6		ns
HCX	High Time	15		ns
LCX	Low Time	15		ns
LCH	Rise Time		20	ns
HCL	Fall Time		20	ns

erial Port Timing: Shift Register Mode Test Conditions

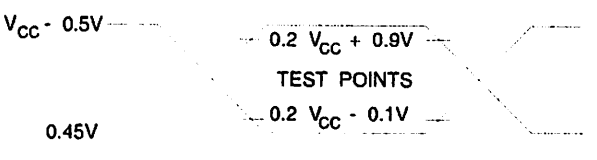
ıe values in this table are valid for V_{CC} = 4.0V to 6V and Load Capacitance = 80 pF.

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

hift Register Mode Timing Waveforms

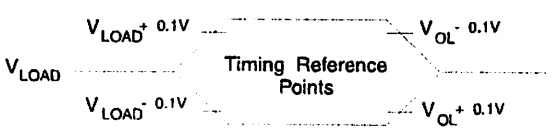


C Testing Input/Output Waveforms[1]



te: 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.

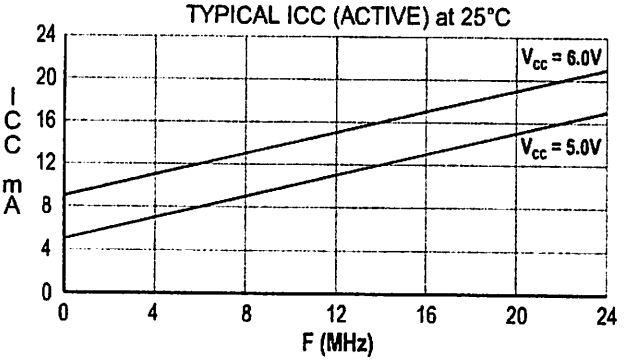
oat Waveforms[1]



te: 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.

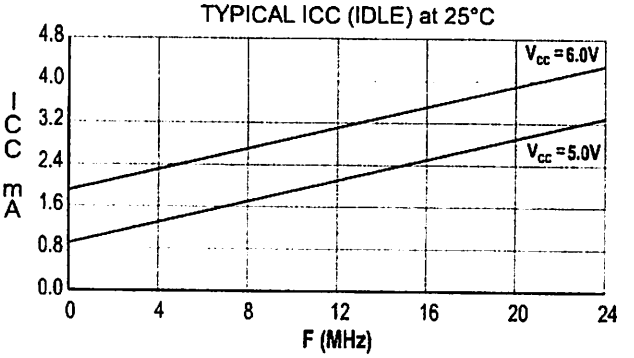
AT89S8252

TYPICAL ICC (ACTIVE) at 25°C



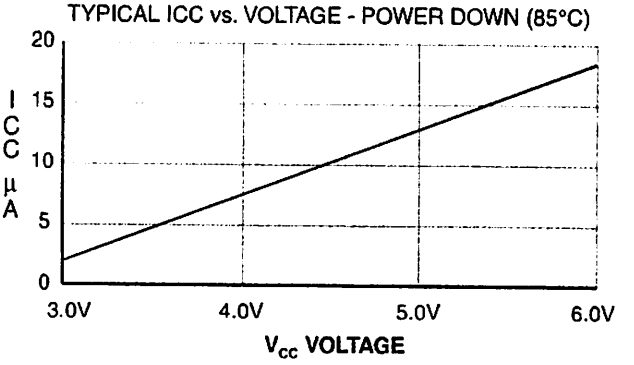
AT89S8252

TYPICAL ICC (IDLE) at 25°C



AT89S8252

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



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

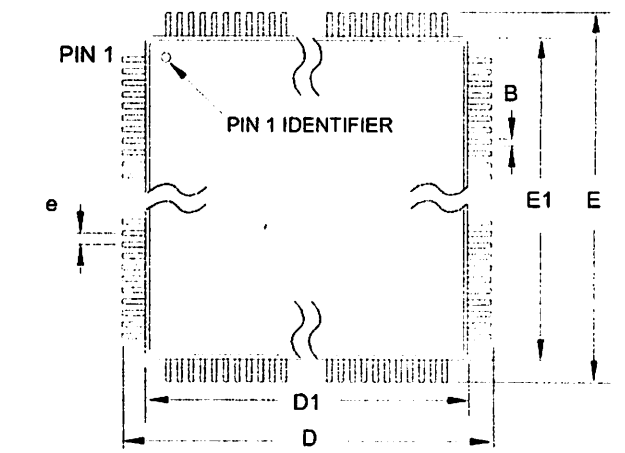
dering Information

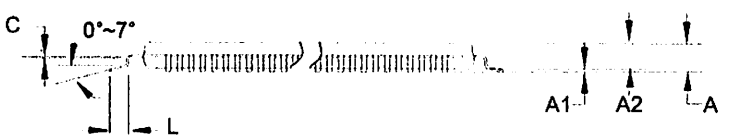
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 6.0V	AT89S8252-24AC AT89S8252-24JC AT89S8252-24PC	44A 44J 40P6	Commercial (0° C to 70° C)
	4.0V to 6.0V	AT89S8252-24AI AT89S8252-24JI AT89S8252-24PI	44A 44J 40P6	Industrial (-40° C to 85° C)

	Package Type
A	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
J	44-lead, Plastic J-leaded Chip Carrier (PLCC)
P6	40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)

ckaging Information

A – TQFP





COMMON DIMENSIONS
(Unit of Measure = mm)

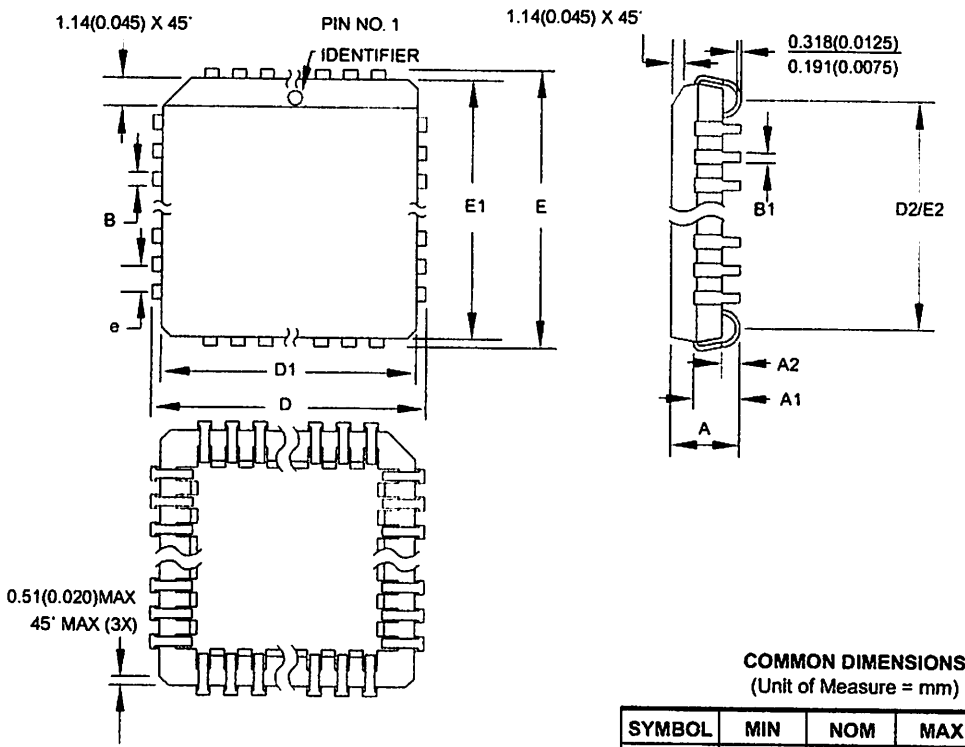
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

Notes:
1. This package conforms to JEDEC reference MS-026, Variation ACB.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25 mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10 mm maximum.

10/5/2001

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**44A**, 44-lead, 10 x 10 mm Body Size, 1.0 mm Body Thickness, 0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)	44A	B

J – PLCC



COMMON DIMENSIONS
(Unit of Measure = mm)

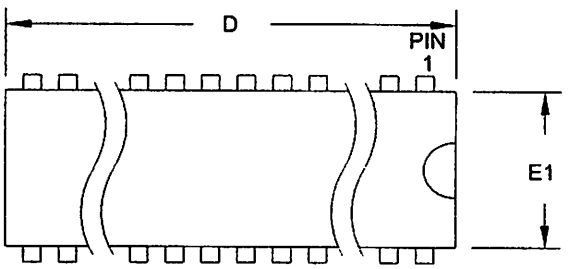
SYMBOL	MIN	NOM	MAX	NOTE
A	4.191	–	4.572	
A1	2.286	–	3.048	
A2	0.508	–	–	
D	17.399	–	17.653	
D1	16.510	–	16.662	Note 2
E	17.399	–	17.653	
E1	16.510	–	16.662	Note 2
D2/E2	14.986	–	16.002	
B	0.660	–	0.813	
B1	0.330	–	0.533	
e	1.270 TYP			

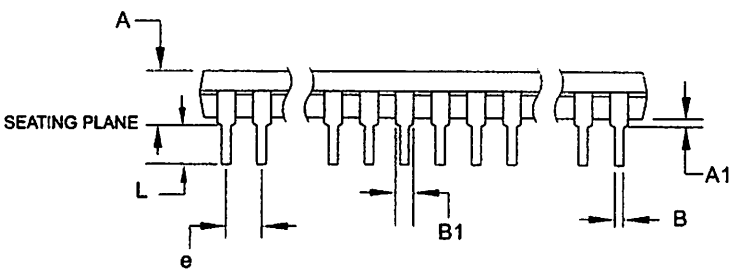
Notes:
1. This package conforms to JEDEC reference MS-018, Variation AC.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is .010"(0.254 mm) per side. Dimension D1 and E1 include mold mismatch and are measured at the extreme material condition at the upper or lower parting line.
3. Lead coplanarity is 0.004" (0.102 mm) maximum.

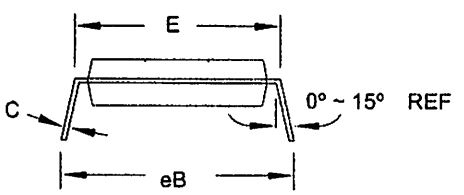
10/04/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**44J**, 44-lead, Plastic J-leaded Chip Carrier (PLCC)	44J	B

P6 – PDIP







COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	4.826	
A1	0.381	–	–	
D	52.070	–	52.578	Note 2
E	15.240	–	15.875	
E1	13.462	–	13.970	Note 2
B	0.356	–	0.559	
B1	1.041	–	1.651	
L	3.048	–	3.556	
C	0.203	–	0.381	
eB	15.494	–	17.526	
e	2.540 TYP			

Notes:
1. This package conforms to JEDEC reference MS-011, Variation AC.
2. Dimensions D and E1 do not include mold Flash or Protrusion. Mold Flash or Protrusion shall not exceed 0.25 mm (0.010").

09/28/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**40P6**, 40-lead (0.600"/15.24 mm Wide) Plastic Dual Inline Package (PDIP)	40P6	B

tmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

egional Headquarters

ırope
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

sia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

pan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

0401F–MICRO–11/03 xM

DALLAS SEMICONDUCTOR

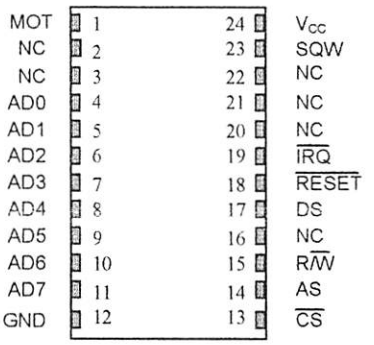
DS12C887
Real Time Clock

www.dalsemi.com

FEATURES

- 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

DESCRIPTION

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

PERATION

ıe block diagram in Figure 1 shows the pin connections with the major internal functions of the S12C887. The following paragraphs describe the function of each pin.

IGNAL DESCRIPTIONS

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

IOT (Mode Select) – The MOT pin offers the flexibility to choose between two bus types. When ınnected to V_{CC}, Motorola bus timing is selected. When connected to GND or left disconnected, Intel ıs timing is selected. The pin has an internal pull-down resistance of approximately 20KΩ.

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

D0-AD7 (Multiplexed Bidirectional Address/Data Bus) – Multiplexed buses save pins because ldress information and data information time share the same signal paths. The addresses are present ıring the first portion of the bus cycle and the same pins and signal paths are used for data in the second ɔrtion of the cycle. Address/data multiplexing does not slow the access time of the DS12C887 since the ıs change from address to data occurs during the internal RAM access time. Addresses must be valid ior to the falling edge of AS/ALE, at which time the DS12C887 latches the address from AD0 to AD6. alid write data must be present and held stable during the latter portion of the DS or WR pulses. In a ad cycle the DS12C887 outputs 8 bits of data during the latter portion of the DS or RD pulses. The read /cle is terminated and the bus returns to a high impedance state as DS transitions low in the case of lotorola timing or as RD transitions high in the case of Intel timing.

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

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

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

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

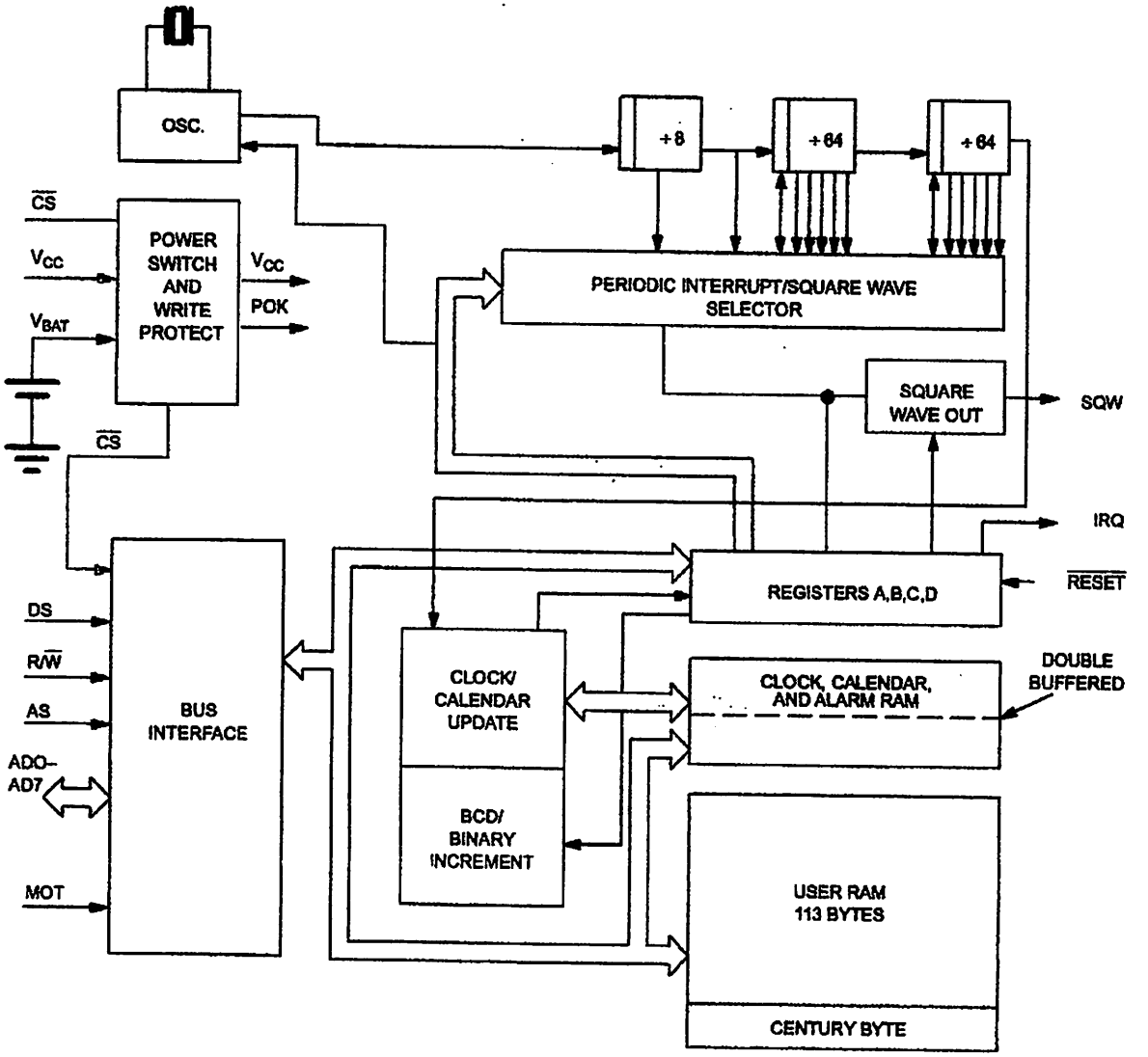
$\overline{IRQ}$ (Interrupt Request Output) - The $\overline{IRQ}$ pin is an active low output of the DS12C887 that can be used as an interrupt input to a processor. The $\overline{IRQ}$ output remains low as long as the status bit causing the interrupt is present and the corresponding interrupt-enable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

$\overline{RESET}$ (Reset Input) – The $\overline{RESET}$ pin has no effect on the clock, calendar, or RAM. On power-up the $\overline{RESET}$ pin can be held low for a time in order to allow the power supply to stabilize. The amount of time that $\overline{RESET}$ is held low is dependent on the application. However, if $\overline{RESET}$ is used on power–up, the time $\overline{RESET}$ is low should exceed 200 ms to make sure that the internal timer that controls the DS12C887 on power-up has timed out. When $\overline{RESET}$ is low and V_{CC} is above 4.25 volts, the following occurs:

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.

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

DS12C887 BLOCK DIAGRAM Figure 1



POWER-DOWN/POWER-UP CONSIDERATIONS

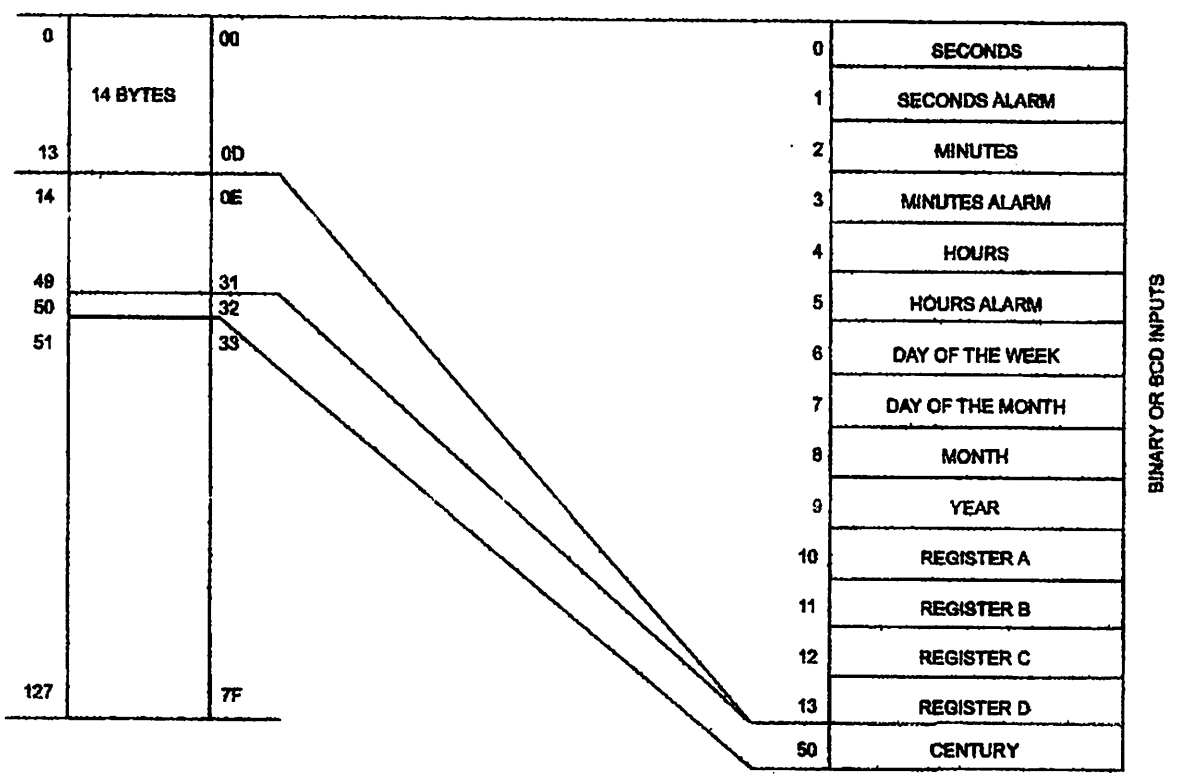
The Real Time Clock function will continue to operate and all of the RAM, time, calendar, and alarm memory locations remain nonvolatile regardless of the level of the V_{CC} input. When V_{CC} is applied to the DS12C887 and reaches a level of greater than 4.25 volts, the device becomes accessible after 200 ms, provided that the oscillator is running and the oscillator countdown chain is not in reset (see Register A). This time period allows the system to stabilize after power is applied. When V_{CC} falls below 4.25 volts, the chip select input is internally forced to an inactive level regardless of the value of $\overline{CS}$ at the input pin. The DS12C887 is, therefore, write-protected. When the DS12C887 is in a write-protected state, all inputs are ignored and all outputs are in a high impedance state. When V_{CC} falls below a level of approximately 3 volts, the external V_{CC} supply is switched off and an internal lithium energy source supplies power to the Real Time Clock and the RAM memory.

RTC ADDRESS MAP

The address map for the DS12C885 is shown in Figure 2. The address map consists of 113 bytes of user RAM, 11 bytes of RAM that contain the RTC time, calendar, and alarm data, and 4 bytes which are used for control 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.

DS12C887 REAL TIME CLOCK ADDRESS MAP Figure 2



TIME, CALENDAR AND ALARM LOCATIONS

The time and calendar information is obtained by reading the appropriate memory bytes. The time, calendar, and alarm are set or initialized by writing the appropriate RAM bytes. The contents of the ten time, calendar, and alarm bytes can be either Binary or Binary-Coded Decimal (BCD) format. Before writing the internal time, calendar, and alarm registers, the SET bit in Register B should be written to a logic one to prevent updates from occurring while access is being attempted. In addition to writing the ten time, calendar, and alarm registers in a selected format (binary or BCD), the data mode bit (DM) of Register B must be set to the appropriate logic level. All ten time, calendar, and alarm bytes must use the same data mode. The set bit in Register B should be cleared after the data mode bit has been written to allow the real time clock to update the time and calendar bytes. Once initialized, the real time clock makes all updates in the selected mode. The data mode cannot be changed without reinitializing the ten data bytes. Table 2 shows the binary and BCD formats of the ten time, calendar, and alarm locations. The 4–12 bit cannot be changed without reinitializing the hour locations. When the 12–hour format is selected, the high order bit of the hours byte represents PM when it is a logic one. The time, calendar, and alarm bytes are always accessible because they are double buffered. Once per second the eleven bytes are advanced by one second and checked for an alarm condition. If a read of the time and calendar data occurs during an update, a problem exists where seconds, minutes, hours, etc. may not correlate. The probability of reading incorrect time and calendar data is low. Several methods of avoiding any possible incorrect time and calendar reads are covered later in this text. The three alarm bytes can be used in two ways. First, when the alarm time is written in the appropriate hours, minutes, and seconds alarm locations, the alarm interrupt is initiated at the specified time each day if the alarm enable bit is high . The second use condition is to insert a "don't care" state in one or more of the three alarm bytes. The "don't care" code is any hexadecimal value from C0 to FF. The two most significant bits of each byte set the "don't care" condition when at logic 1. An alarm will be generated each hour when the "don't care" bits are set in the hours byte. Similarly, an alarm is generated every minute with "don't care" codes in the hours and minute alarm bytes. The "don't care" codes in all three alarm bytes create an interrupt every second.

TIME, CALENDAR AND ALARM DATA MODES Table 1

ADDRESS LOCATION	FUNCTION	DECIMAL RANGE	RANGE	
			BINARY DATA MODE	BCD DATA MODE
0	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

CONTROL REGISTERS

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

REGISTER A

MSB **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

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

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

RS3, RS2, RS1, RS0 - These four rate-selection bits select one of the 13 taps on the 15-stage divider or disable the divider output. The tap selected can be used to generate an output square wave (SQW pin) and/or 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

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

REGISTER B

MSB LSB

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

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

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

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

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

SQWE - When the Square Wave Enable (SQWE) bit is set to a 1, a square wave signal at the frequency set by the rate-selection bits RS3 through RS0 is driven out on the SQW pin. When the SQWE bit is set to 0, the SQW pin is held low. SQWE is a read/write bit and is cleared by $\overline{\text{RESET}}$. SQWE is set to a 1 when V_{CC} is powered up.

DM - The Data Mode (DM) bit indicates whether time and calendar information is in binary or BCD format. The DM bit is set by the program to the appropriate format and can be read as required. This bit is not modified by internal functions or $\overline{\text{RESET}}$. A 1 in DM signifies binary data while a 0 in DM specifies Binary Coded Decimal (BCD) data.

24/12 - The 24/12 control bit establishes the format of the hours byte. A 1 indicates the 24-hour mode and a 0 indicates the 12-hour mode. This bit is read/write and is not affected by internal functions or $\overline{\text{RESET}}$.

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

REGISTER C

MSB LSB

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

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

i.e., IRQF = (PF • PIE) + (AF • AIE) + (UF • UIE)

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

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

AF - A 1 in the Alarm Interrupt Flag (AF) bit indicates that the current time has matched the alarm time. If the 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 of Register C will clear AF.

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

BIT 3 THROUGH BIT 0 - These are unused bits of the status Register C. These bits always read 0 and cannot be written.

REGISTER D

MSB LSB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
VRT	0	0	0	0	0	0	0

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

BIT 6 THROUGH BIT 0 - The remaining bits of Register D are not usable. They cannot be written and, when read, they will always read 0.

CENTURY REGISTER

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

NONVOLATILE RAM

The 113 general purpose nonvolatile RAM bytes are not dedicated to any special function within the DS12C887. They can be used by the processor program as nonvolatile memory and are fully available during the update cycle.

INTERRUPTS

The RTC plus RAM includes three separate, fully automatic sources of interrupt for a processor. The alarm interrupt can be programmed to occur at rates from once per second to once per day. The periodic interrupt can be selected for rates from 500ms to 122µs. The update-ended interrupt can be used to indicate to the program that an update cycle is complete. Each of these independent interrupt conditions is described in greater detail in other sections of this text.

The processor program can select which interrupts, if any, are going to be used. Three bits in Register B enable the interrupts. Writing a logic 1 to an interrupt-enable bit permits that interrupt to be initiated when the event occurs. A zero in an interrupt-enable bit prohibits the $\overline{IRQ}$ pin from being asserted from that interrupt condition. If an interrupt flag is already set when an interrupt is enabled, $\overline{IRQ}$ is immediately set at an active level, although the interrupt initiating the event may have occurred much earlier. As a result, there are cases where the program should clear such earlier initiated interrupts before first enabling new interrupts.

When an interrupt event occurs, the relating flag bit is set to logic 1 in Register C. These flag bits are set independent of the state of the corresponding enable bit in Register B. The flag bit can be used in a polling mode without enabling the corresponding enable bits. The interrupt flag bit is a status bit which software can interrogate as necessary. When a flag is set, an indication is given to software that an interrupt event has occurred since the flag bit was last read; however, care should be taken when using the flag bits as they are cleared each time Register C is read. Double latching is included with Register C so that bits which are set remain stable throughout the read cycle. All bits which are set (high) are cleared when read and new interrupts which are pending during the read cycle are held until after the cycle is completed. One, two, or three bits can be set when reading Register C. Each utilized flag bit should be examined when read to ensure that no interrupts are lost.

The second flag bit usage method is with fully enabled interrupts. When an interrupt flag bit is set and the corresponding interrupt enable bit is also set, the $\overline{IRQ}$ pin is asserted low. $\overline{IRQ}$ is asserted as long as at least one of the three interrupt sources has its flag and enable bits both set. The IRQF bit in Register C is one whenever the $\overline{IRQ}$ pin is being driven low. Determination that the RTC initiated an interrupt is accomplished by reading Register C. A logic one in bit 7 (IRQF bit) indicates that one or more interrupts have been initiated by the DS12C887. The act of reading Register C clears all active flag bits and the IRQF bit.

OSCILLATOR CONTROL BITS

When the DS12C887 is shipped from the factory, the internal oscillator is turned off. This feature prevents the lithium energy cell from being used until it is installed in a system. A pattern of 010 in bits 4 through 6 of Register A will turn the oscillator on and enable the countdown chain. A pattern of 11X will turn the oscillator on, but holds the countdown chain of the oscillator in reset. All other combinations of bits 4 through 6 keep the oscillator off.

SQUARE WAVE OUTPUT SELECTION

Thirteen of the 15 divider taps are made available to a 1-of-15 selector, as shown in the block diagram of Figure 1. The first purpose of selecting a divider tap is to generate a square wave output signal on the SQW pin. The RS0–RS3 bits in Register A establish the square wave output frequency. These frequencies are listed in Table 1. The SQW frequency selection shares its 1-of-15 selector with the periodic interrupt generator. Once the frequency is selected, the output of the SQW pin can be turned on and off under program control with the square wave enable bit (SQWE).

PERIODIC INTERRUPT SELECTION

The periodic interrupt will cause the $\overline{IRQ}$ pin to go to an active state from once every 500ms to once every 122µs. This function is separate from the alarm interrupt which can be output from once per second to once per day. The periodic interrupt rate is selected using the same Register A bits which select the square wave frequency (see Table 1). Changing the Register A bits affects both the square wave frequency and the periodic interrupt output. However, each function has a separate enable bit in Register B. The SQWE bit controls the square wave output. Similarly, the periodic interrupt is enabled by the PIE bit in Register B. The periodic interrupt can be used with software counters to measure inputs, create output intervals, or await the next needed software function.

PERIODIC INTERRUPT RATE AND SQUARE WAVE OUTPUT FREQUENCY

Table 2

EXT. 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

JPDATE CYCLE

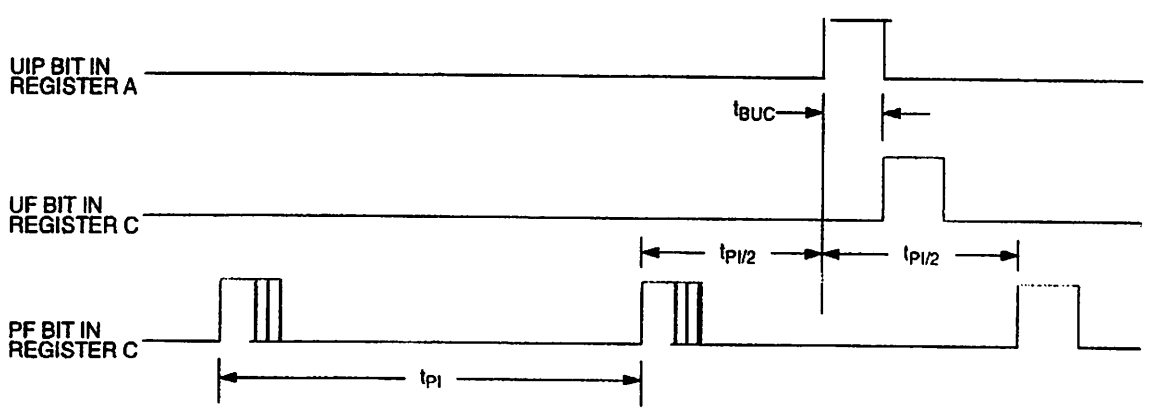
The DS12C887 executes an update cycle once per second regardless of the SET bit in Register B. When he SET bit in Register B is set to one, the user copy of the double buffered time, calendar, and alarm ytes is frozen and will not update as the time increments. How-ever, the time countdown chain continues o update the internal copy of the buffer. This feature allows time to maintain accuracy independent of eading or writing the time, calendar, and alarm buffers and also guarantees that time and calendar nformation is consistent. The update cycle also compares each alarm byte with the corresponding time yte and issues an alarm if a match or if a "don't care" code is present in all three positions.

here are three methods that can handle access of the real time clock that avoid any possibility of ccessing inconsistent time and calendar data. The first method uses the update-ended interrupt. If nabled, an interrupt occurs after every up date cycle that indicates that over 999 ms are available to read alid time and date information. If this interrupt is used, the IRQF bit in Register C should be cleared efore leaving the interrupt routine.

second method uses the update-in-progress bit (UIP) in Register A to determine if the update cycle is in rogress. The UIP bit will pulse once per second. After the UIP bit goes high, the update transfer occurs 44µs later. If a low is read on the UIP bit, the user has at least 244µs before the time/calendar data will e changed. Therefore, the user should avoid interrupt service routines that would cause the time needed read valid time/calendar data to exceed 244µs.

he third method uses a periodic interrupt to determine if an update cycle is in progress. The UIP bit in egister A is set high between the setting of the PF bit in Register C (see Figure 3). Periodic interrupts at occur at a rate of greater than t_{BUC} allow valid time and date information to be reached at each ccurrence of the periodic interrupt. The reads should be complete within 1 ($t_{PI/2} + t_{BUC}$) to ensure that ata is not read during the update cycle.

IPDATE-ENDED AND PERIODIC INTERRUPT RELATIONSHIP Figure 3



t_{PI} = PERIODIC INTERRUPT TIME INTERNAL PER TABLE 1
t_{BUC} = DELAY TIME BEFORE UPDATE CYCLE = 244 µs

ABSOLUTE MAXIMUM RATINGS

Voltage on Any Pin Relative to Ground	-0.3V to +7.0V
Operating Temperature	0° to 70°C
Storage Temperature	-40°C to +70°C
Soldering Temperature	260°C for 10 seconds (See Note 7)

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

RECOMMENDED DC OPERATING CONDITIONS (0°C to 70°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Power Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
Input Logic 1	V_{IH}	2.3		V_{CC}+0.3	V	1
Input Logic 0	V_{IL}	-0.3		0.8	V	1

DC ELECTRICAL CHARACTERISTICS (0°C to 70°C; VCC = 5.0V ± 10%)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Average V_{CC} Power Supply Current	I_{CC1}		7	15	mA	2
Input Leakage	I_{IL}	-1.0		+1.0	µA	3
I/O Leakage	I_{OL}	-1.0		+1.0	µA	4
Input Current	I_{MOT}	-1.0		+500	µA	3
Output @ 2.4V	I_{OH}	-1.0			mA	1,5
Output @ 0.4V	I_{OL}			4.0	mA	1
Write Protect Voltage	V_{TP}	4.0	4.25	4.5	V	

CAPACITANCE (tA = 25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Capacitance	C_{IN}			5	pF	
Output Capacitance	C_{OUT}			7	pF	

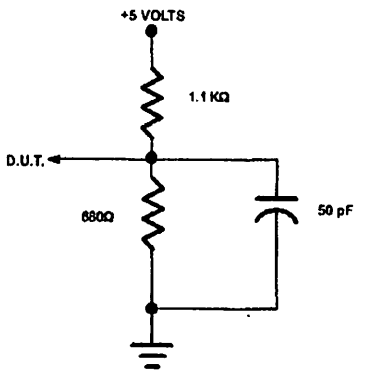
AC ELECTRICAL CHARACTERISTICS (0°C to 70°C; VCC = 5.0V ± 10%)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Cycle Time	t_{CYC}	385		DC	ns	
Pulse Width, DS/E Low or RD/$\overline{WR}$ High	PW_{EL}	150			ns	
Pulse Width, DS/E High or RD/$\overline{WR}$ Low	PW_{RH}	125			ns	
Input Rise and Fall	t_R, t_F			30	ns	
R/$\overline{W}$ Hold Time	t_{RWH}	10			ns	
R/$\overline{W}$ Setup Time Before DS/E	t_{RWS}	50			ns	
Chip Select Setup Time Before DS, $\overline{WR}$, or $\overline{RD}$	t_{CS}	20			ns	
Chip Select Hold Time	t_{CH}	0			ns	
Read Data Hold Time	t_{DHR}	10		80	ns	
Write Data Hold Time	t_{DHW}	0			ns	
Mux'ed Address Valid Time to ALE Fall	t_{ASL}	30			ns	
Mused Address Hold Time to ALE Fall	t_{AHL}	10			ns	
Delay Time DS/E to AS/ALE Rise	t_{ASD}	20			ns	
Pulse Width AS/ALE High	PW_{ASH}	60			ns	
Delay Time, AS/ALE to DS/E Rise	t_{ASED}	40			ns	
Output Data Delay Time from DS/E or $\overline{RD}$	t_{DDR}	20		120	ns	6
Data Setup Time	t_{DSW}	100			ns	
Reset Pulse Width	t_{RWL}	5			µs	
$\overline{IRQ}$ Release from DS	t_{IRDS}			2	µs	
$\overline{IRQ}$ Release from $\overline{RESET}$	t_{IRR}			2	µs	

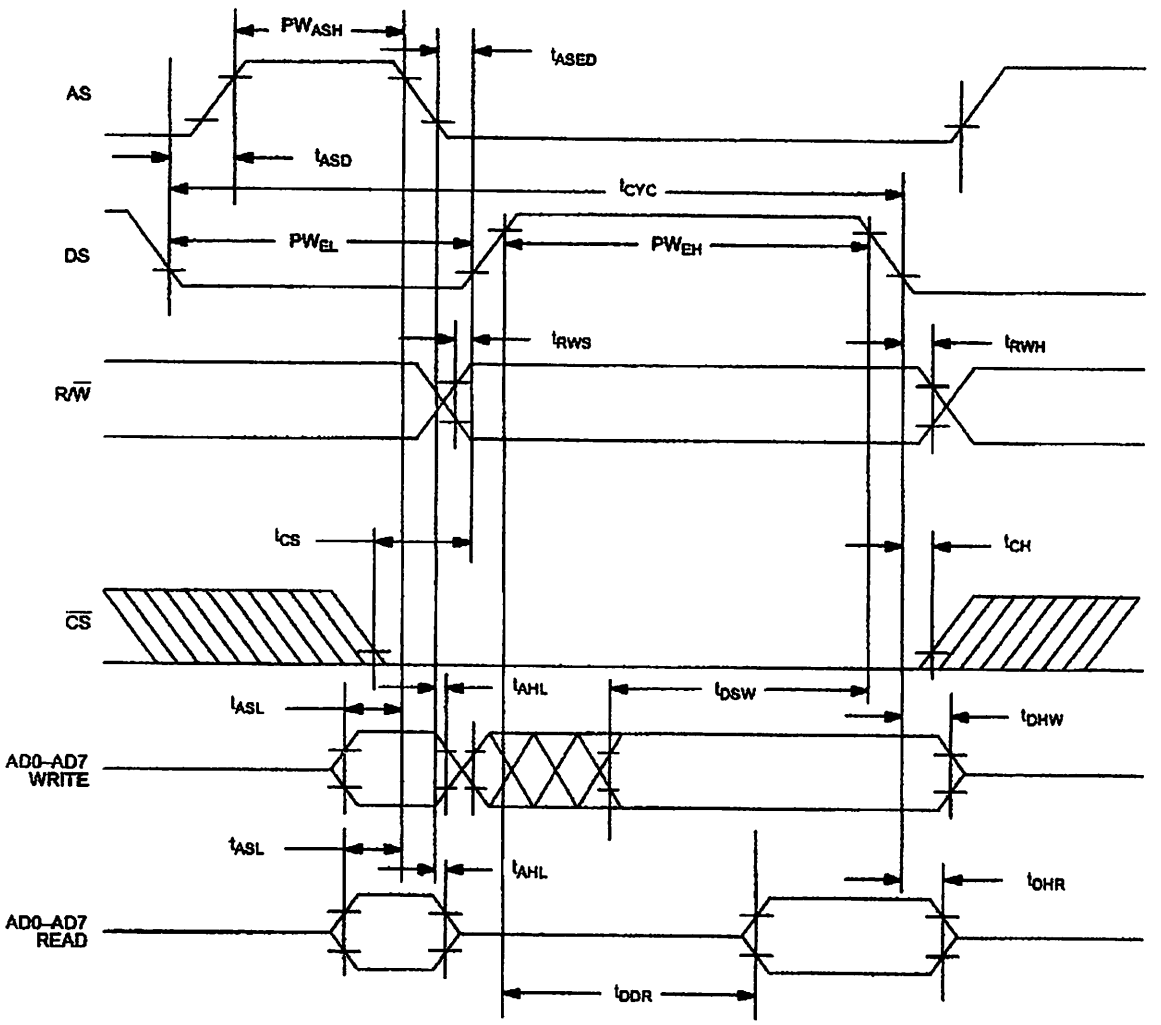
NOTES:

1. All voltages are referenced to ground.
2. All Outputs are open.
3. The MOT pin has an internal pull-down of 20KΩ.
4. Applies to the AD0-AD7 pins, the $\overline{IRQ}$ pin, and the SQW pin when each is in a high impedance state.
5. The $\overline{IRQ}$ pin is open drain.
6. Measured with a load as shown in Figure 4.
7. Real-Time Clock Modules can be successfully processed through conventional wave-soldering techniques as long as temperature exposure to the lithium energy source contained within does not exceed +85°C. Post solder cleaning with water washing techniques is acceptable, provided that ultrasonic vibration is not used. Such cleaning can damage the crystal.

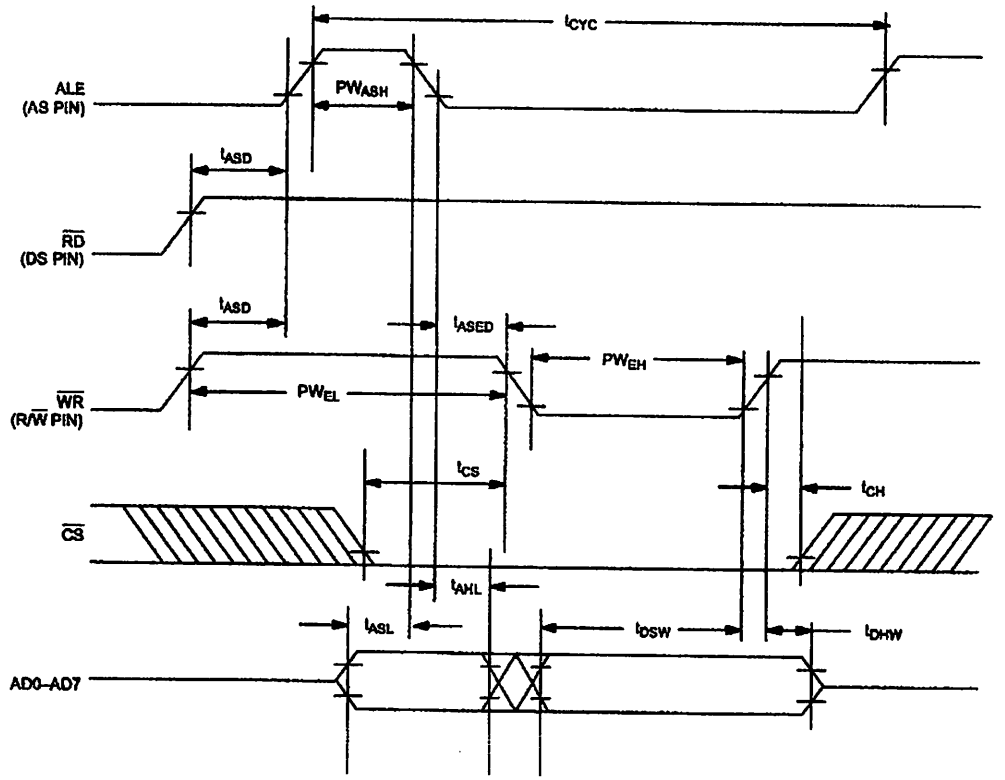
OUTPUT LOAD Figure 4



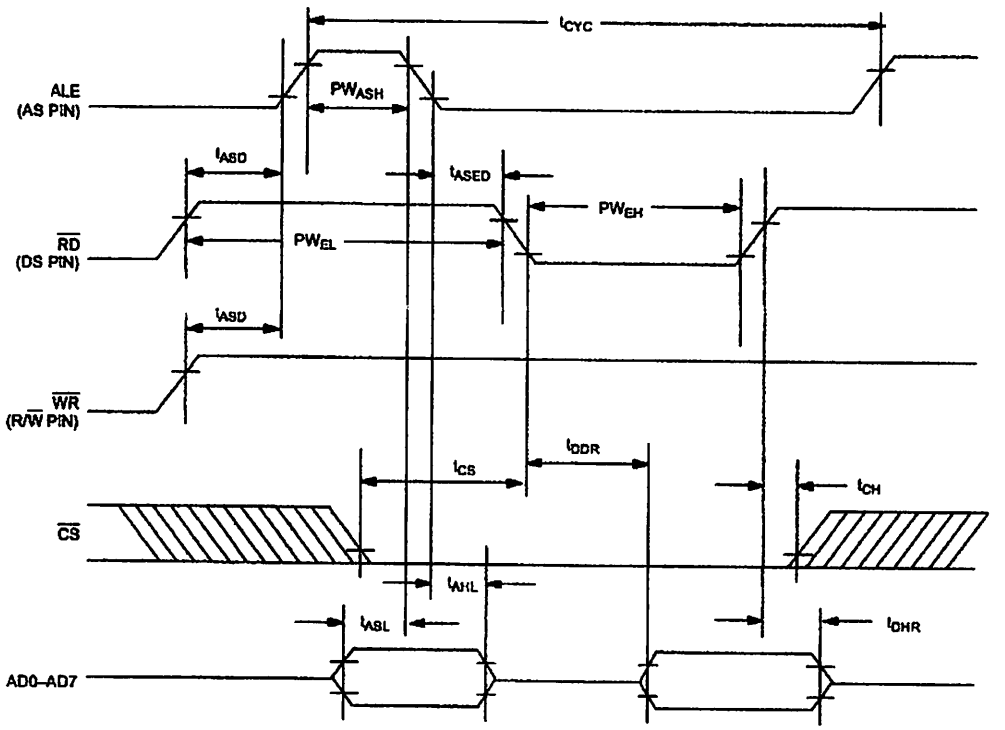
DS12C887 BUS TIMING FOR MOTOROLA INTERFACE



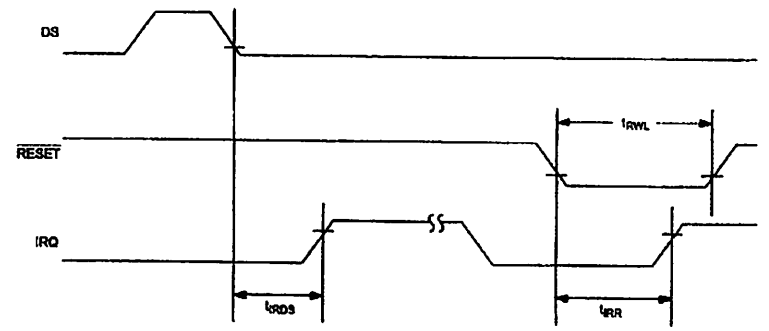
DS12C887 BUS TIMING FOR INTEL INTERFACE WRITE CYCLE



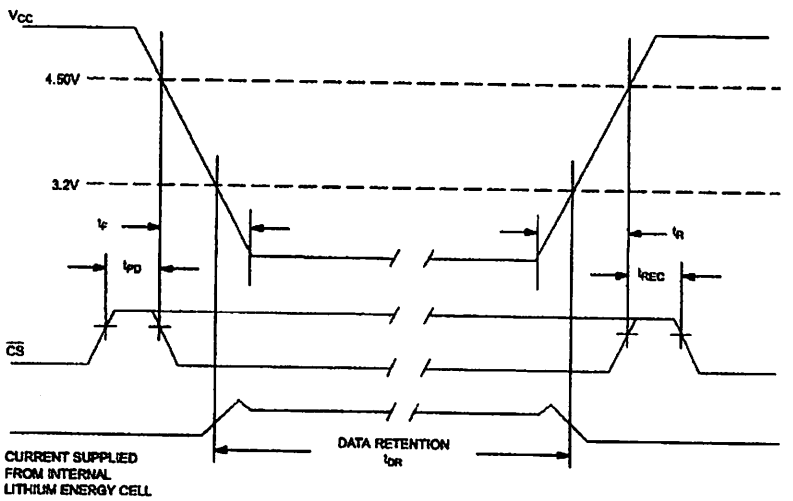
DS12C887 BUS TIMING FOR INTEL INTERFACE READ CYCLE



DS12C887 IRQ RELEASE DELAY TIMING



POWER DOWN / POWER UP TIMING



POWER DOWN / POWER UP TIMING

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
$\overline{CS}$ at V_{IH} before Power-Down	t_{PD}			0	µs	
V_{CC} slew from 4.5V to 0V ($\overline{CS}$ at V_{IH})	t_F $4.0 \le V_{CC} \le 4.5V$	300			µs	
V_{CC} slew from 0V to 4.5V ($\overline{CS}$ at V_{IH})	t_R	100			µs	
$\overline{CS}$ at V_{IH} after Power-Up	t_{REC}	20		200	ms	
Expected Data Retention	t_{DR}	10			years	10,11

(tA=25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Expected Data Retention	t_{DR}	10			years	10,11

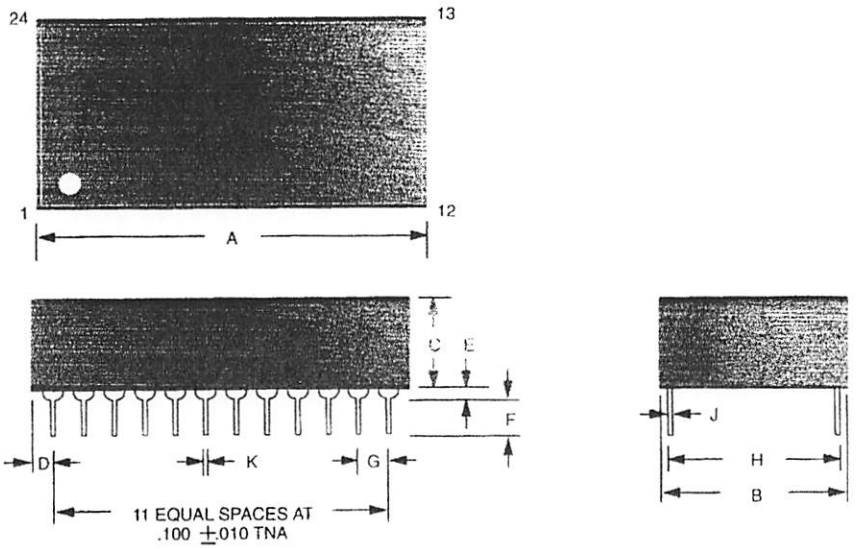
Note:

The real time clock will keep time to an accuracy of ±1 minute per month during data retention time for the period of t_{DR}.

Warning:

Under no circumstances are negative undershoots, of any amplitude, allowed when device is in battery back-up mode.

DS12C887 REAL TIME CLOCK PLUS RAM



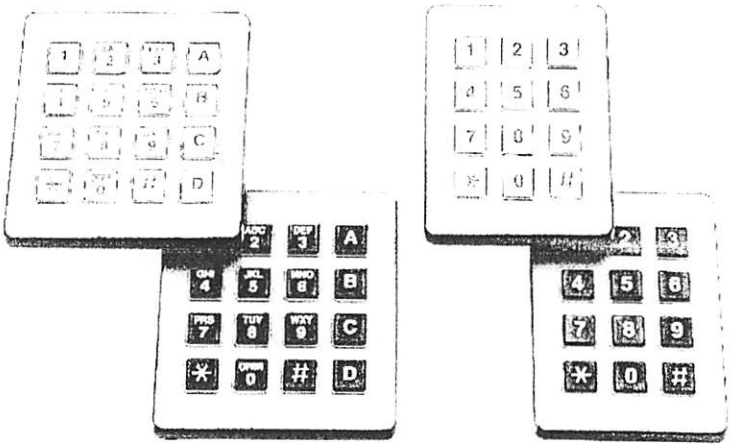
NOTE:

Pins 2, 3, 16, 20, 21 and 22 are missing by design

PKG	24-PIN	
DIM	MIN	MAX
A IN	1.320	1.335
MM	33.53	33.91
B IN	0.675	0.700
MM	17.15	17.78
C IN	0.345	0.370
MM	8.76	9.40
D IN	0.100	0.130
MM	2.54	3.30
E IN	0.015	0.030
MM	0.38	0.76
F IN	0.110	0.140
MM	2.79	3.56
G IN	0.090	0.110
MM	2.29	2.79
H IN	0.590	0.630
MM	14.99	16.00
J IN	0.008	0.012
MM	0.20	0.30
K IN	0.015	0.021
MM	0.38	0.53

SERIES 96
Conductive Rubber

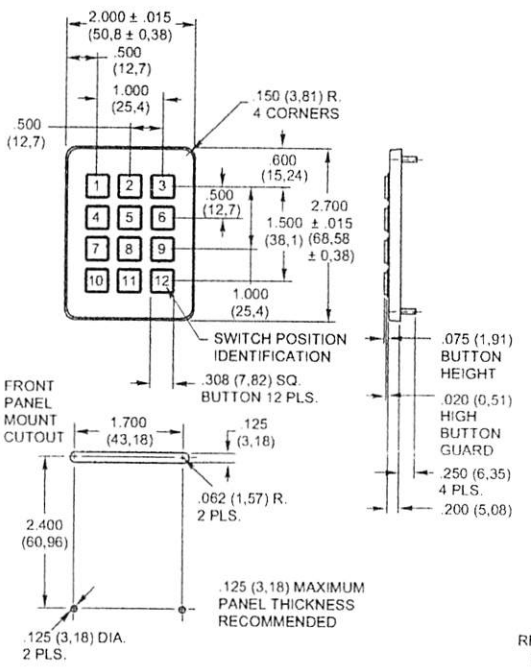
FEATURES
- Quality, Economical Keyboards
- Easily Customized Legends
- Matrix Circuitry
- Backlit and Shielded Options Available
- Termination Mates With Standard Connectors
- Tactile Feedback to Operator
- 1,000,000 Operations per Button
- Compatible With High Resistance Logic Inputs



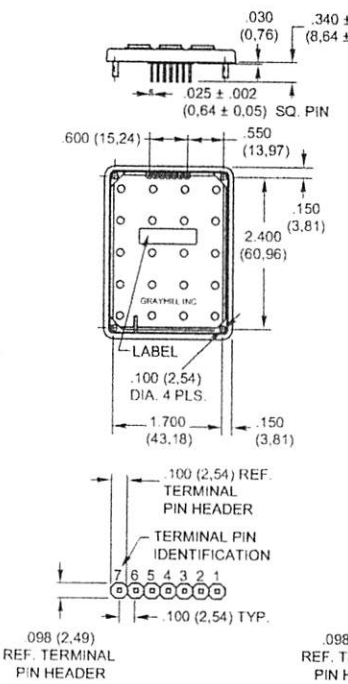
The Series 96 is Grayhill's most economical 3x4 and 4x4 keypad family. The contact system utilizes conductive rubber to mate the appropriate PC board traces. Offered in matrix circuitry, with shielded and backlit options. Built with quality component parts, the Series 96 is subjected to our rigid statistical process control to insure that it meets our reliability standards.

DIMENSIONS In inches (and millimeters)

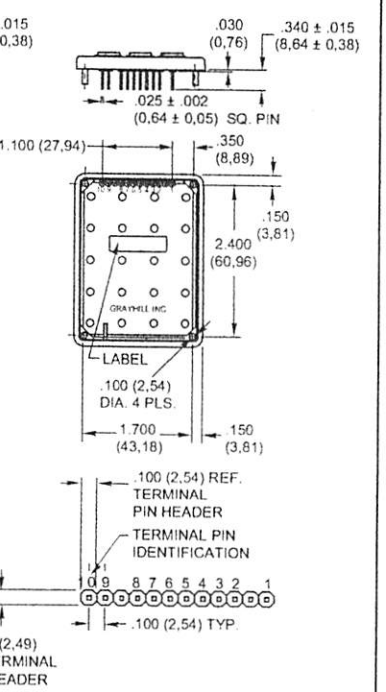
3x4 Front Mount Keyboard



Standard Versions

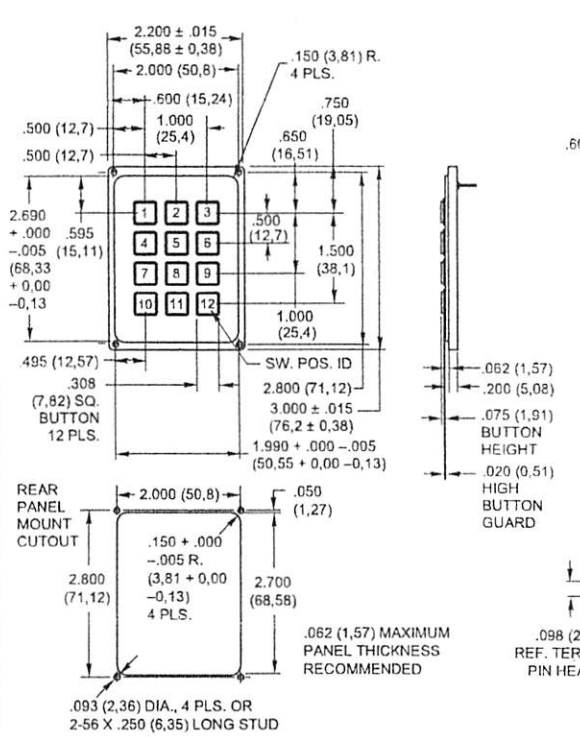


Shielded/Backlit Versions

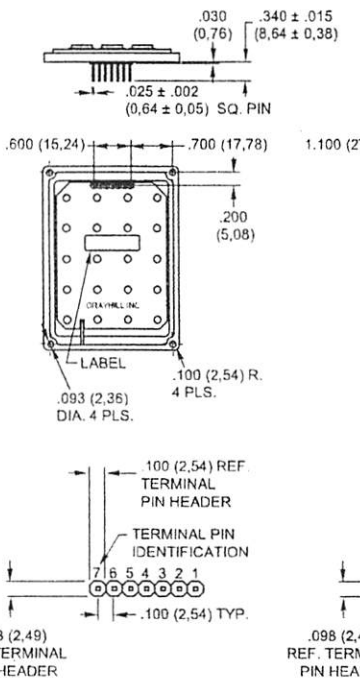


DIMENSIONS In inches (and millimeters)

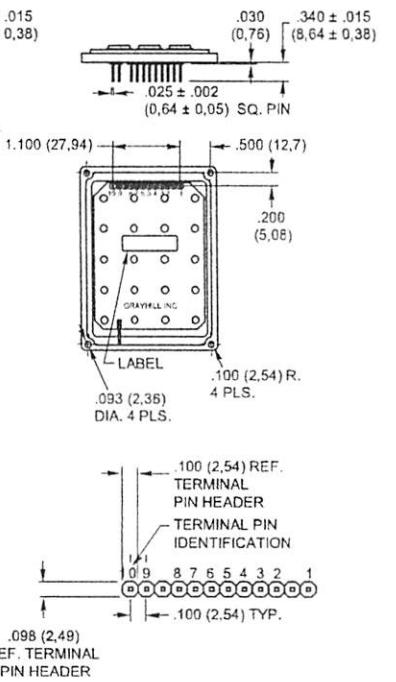
3x4 Rear Mount Keyboard



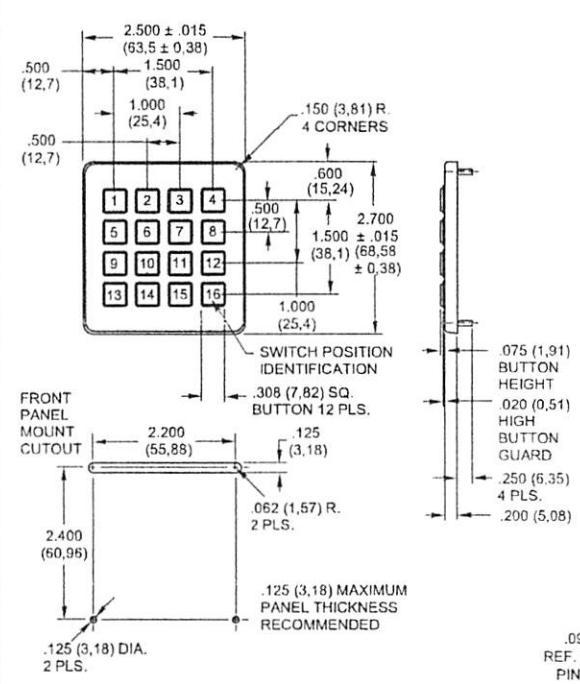
Standard Versions



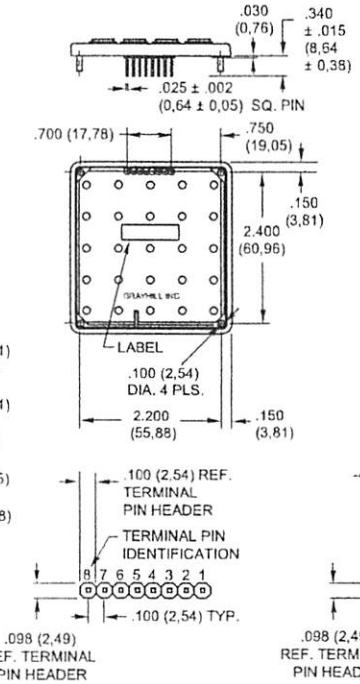
Shielded/Backlit Versions



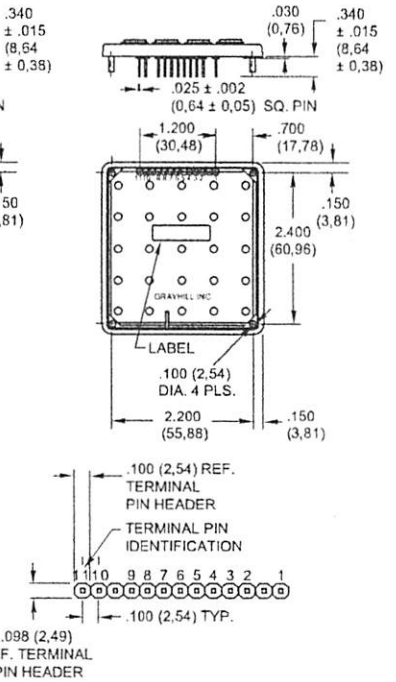
4x4 Front Mount Keyboard



Standard Versions



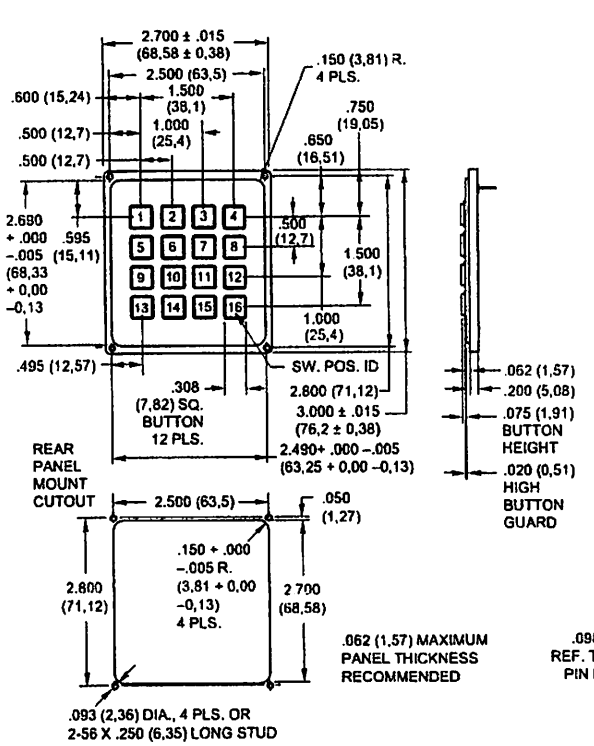
Shielded/Backlit Versions



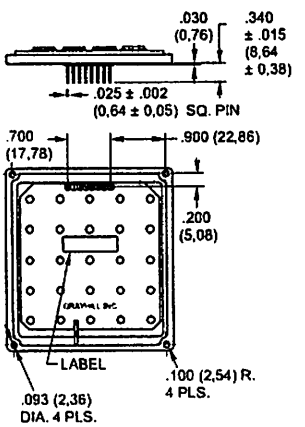
Grayhill, Inc. • 561 Hillgrove Avenue • LaGrange, Illinois 60525-5997 • USA • Phone: 708-354-1040 • Fax: 708-354-2820 • www.grayhill.com

IMENSIONS In inches (and millimeters)

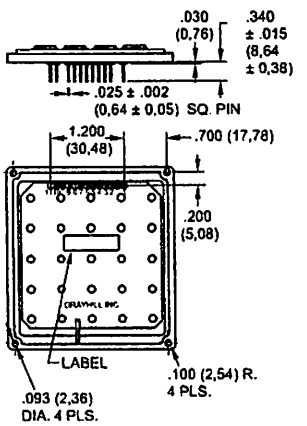
4x4 Rear Mount Keyboard

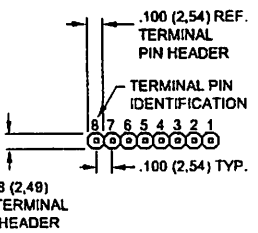


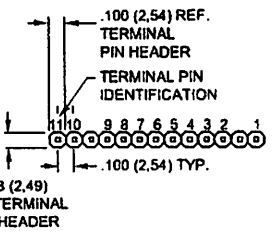
Standard Versions



Shielded/Backlit Versions



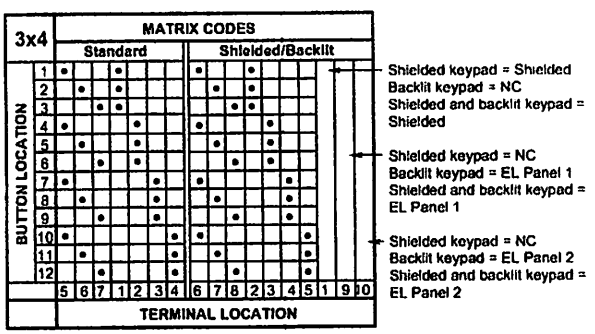




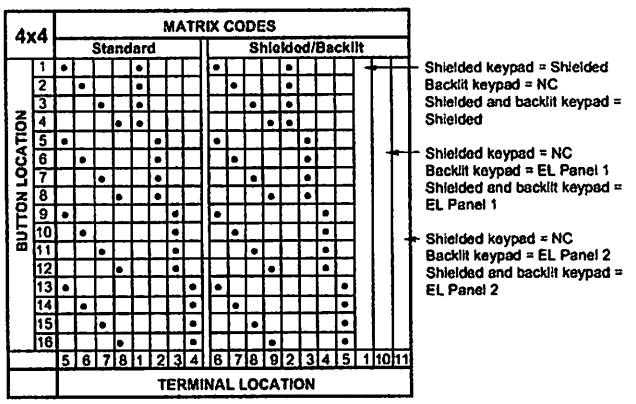
CODE AND TRUTH TABLES

Dots in the chart indicate connected terminals when switch is closed.
Terminals are identified on the keyboard.

12 Button Keypads



16 Button Keypads



SPECIFICATIONS

Rating Criteria

Rating at 12 Vdc: 5 milliamps for .5 seconds
Contact Bounce: < 12 milliseconds
Contact Resistance: < 100 ohms (at stated operating force)
Voltage Breakdown: 250 Vac between components
Mechanical Operation Life: 1,000,000 operations per key
Insulation Resistance: > 10^{12} ohms @ 500 Vdc
Push Out Force Per Pin: 5 lbs.

Operating Features

Travel: .040 minimum
Operating Force: 175 ± 40 grams
Operating Temperature: -30°C to +80°C

Material and Finishes

Terminal Pin: Phosphor bronze, solder-plated
PC Board: FR-4 glass cloth epoxy
Keypad: Silicone rubber, durometer 50 ± 5
Housing: ABS, cycolac "KJW"
Housing Color: Black

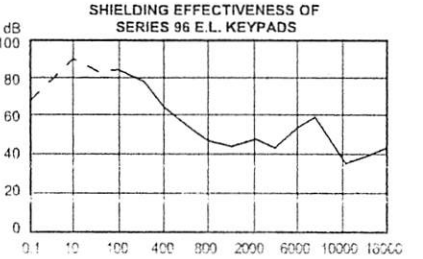
Shielding Effectiveness

Results shown are typical for a standard Grayhill Series 84S keyboard. A conductive gasket will generally increase the shielding, depending on the size and shape of the gasket and its material. Data derived for E-Field Radiation.

Test Method:
Measurements were made with the keyboard mounted to a brass plate, which in turn was mounted to a shielded enclosure containing the receiving equipment. A signal generator provided the frequency source that was radiated from the transmitting antenna to the enclosed receiving antenna. The spacing between antennas was maintained constant throughout the frequency range. The effectiveness rating is determined by establishing a reference reading without obstruction between the two antennas and determining the difference between that reading and the test setup reading.

Note:
When measured in actual equipment, shielding effectiveness is determined by many factors. This method accurately represents the shielding effectiveness of the Grayhill Series 84S under ideal test conditions.

SHIELDING EFFECTIVENESS OF SERIES 96 E.L. KEYPADS



— – Represents shielding effectiveness greater than or equal to line.

Frequency M Hz	Rating in dB
0.1	≥ 66.2
10	> 94.8
100	90.5
400	64.2
800	42.3
2,000	40.5
6,000	33.1
10,000	34.4
18,000	37.0

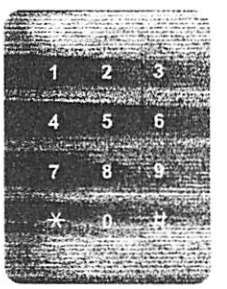
STANDARD LEGENDS

Available through Grayhill Distributors

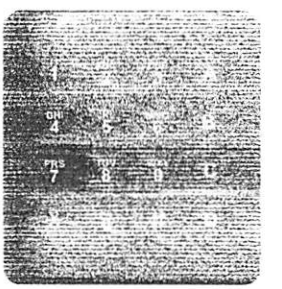
To order one of the configurations below, use the dash number shown here, select the keypad size and code, and order the part number with the appropriate legend dash number.

-102

-006



-152



-056

ORDERING INFORMATION

96AB2-102-FS-EL

Grayhill Series Number
Keyboard Size: A = 3x4, B = 4x4
Circuitry: B2 = Matrix (terminal pin header)

E.L. Panel Backlighting Option
EL = Backlit, Blank = Non-backlit
EMI/RFI Shielding Option
S = Shielded, Blank = Non-shielded
Mounting Option: F = Front panel mount, R = Rear panel mount
Standard Legend Choices
12 Position legends
102 = Black legends on a white button
152 = White legends on a black button
16 Position legends
006 = Black legends on a white button
056 = White legends on a black button

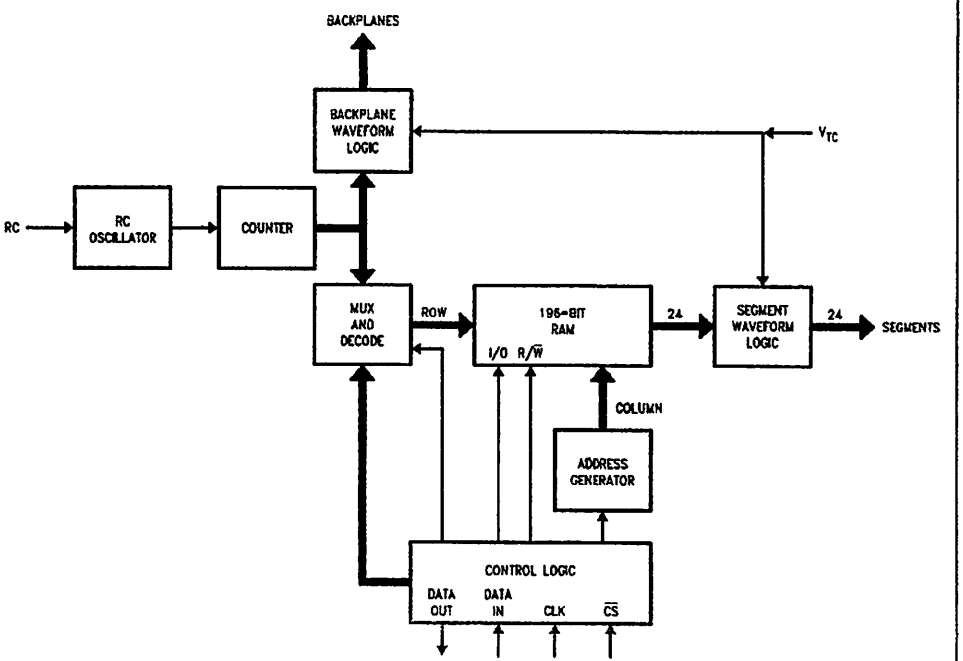
Available from your local Grayhill Distributor.
For prices and discounts, contact a local Sales Office, an authorized local Distributor or Grayhill.

Grayhill, Inc. • 561 Hillgrove Avenue • LaGrange, Illinois 60525-5997 • USA • Phone: 708-354-1040 • Fax: 708-354-2820 • www.grayhill.com

Designing an LCD Dot Matrix Display Interface

National Semiconductor
Application Note 350
Bob Lutz
May 1988

The MM58201 is a CMOS LCD driver capable of driving a multiplexed display of up to 192 segments (24 segment columns by 8 backplanes). The number of backplanes being driven is programmable from one to eight. Data to be displayed is sent to the chip serially and stored in an internal RAM. An external resistor and capacitor control the frequency of the driving signals to the LCD. The MM58201 can also be programmed to accept the oscillator output and backplane signals of another MM58201 for cascading purposes. The displayed data may also be read serially from the on-chip RAM. A simplified functional block diagram of the MM58201 is shown in *Figure 1*.



TL/B/5606-1

FIGURE 1. MM58201 Functional Diagram

RRD-B30M105/Printed in U. S. A.

BACKGROUND

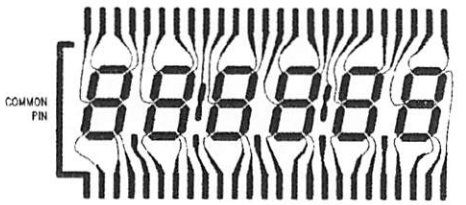
LCD displays have become very popular because of their ultra-low power consumption and high contrast ratio under high ambient light levels. Typically an LCD has a backplane that overlaps the entire display area and multiple segment lines that each overlap just one segment or descriptor. This means that a separate external connection is needed for every segment or descriptor as shown in *Figure 2*. For a display with many segments such as a dot matrix display, the number of external connections could easily grow to be very large.

Unlike other display technologies that respond to peak or average voltage and current, LCDs are sensitive to the rms voltage between the backplane and given segment location. Also, any DC bias across this junction would cause an irreversible electrochemical action that would shorten the life of the display. A typical LCD driving signal is shown in *Figure 3*. The backplane signal is simply a symmetrical square wave. The individual segment outputs are also square waves, either in phase with the backplane for an "off" segment or out of phase for an "on" segment. This causes a Vrms of zero for an "off" segment and a Vrms of +V for an "on" segment.

One way to reduce the number of external connections is to multiplex the display. An example of this could be an LCD with its segments arranged as intersections of an X-Y grid. A driver to control a matrix like this would be fairly straightforward for an LED display. However, it is more complex for an LCD because of the DC bias restriction.

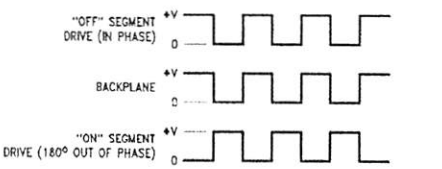
A multiplexed LCD driver must generate a complex set of output signals to insure that an "on" segment sees an rms voltage greater than the display's turn-on voltage and that an "off" segment sees an rms voltage less than the display's turn-off voltage. The driver must also insure that there is no DC bias.

One pattern that can accomplish this is shown as an example in *Figure 4*. This is the pattern that the MM58201 uses. The actual Vrms of an "on" segment and an "off" segment is shown in *Figure 5*. If there are eight backplanes, the Vrms (ON) = $0.2935 \times V_{TC}$ and the Vrms (OFF) = $0.2029 \times V_{TC}$. It can be seen in *Figure 6* that as the number of backplanes increases, the difference between Vrms (ON) and Vrms (OFF) becomes less. Refer to the specifications of the LCD to determine exactly what Vrms is required.



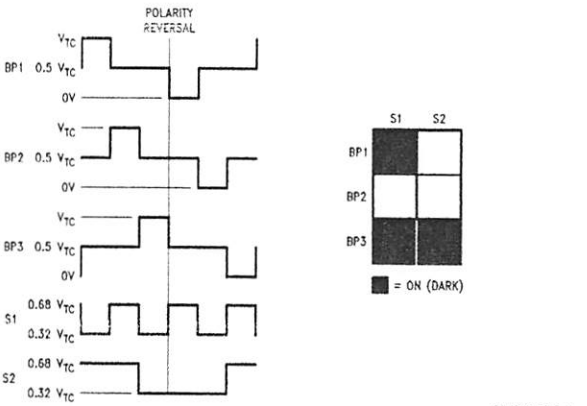
TL/B/5606-2

FIGURE 2. Typical LCD Pin Connections



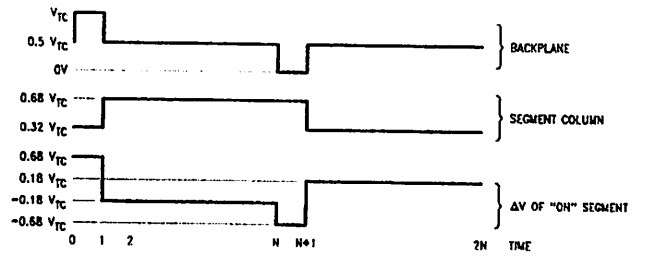
TL/B/5606-3

FIGURE 3. Drive Signals from a Direct Connect LCD Driver



TL/B/5606-4

FIGURE 4. Example of Backplane and Segment Patterns



TL/B/5606-5

$$\text{Vrms (ON)} = \left(\frac{1}{T}\int_{t0}^{t0+T} v^2(t)dt\right)^{1/2}$$

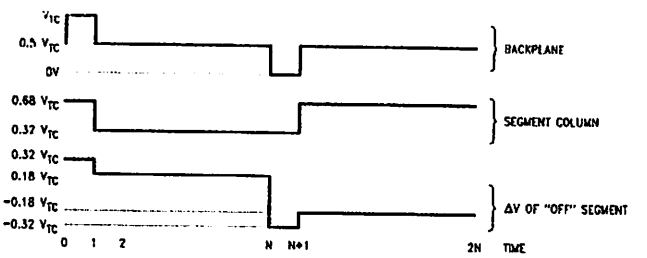
$$= \left(\frac{1}{N}\left[\int_0^1 (0.68\,V_{TC})^2 dt + \int_1^N (-0.18\,V_{TC})^2 dt\right]\right)^{1/2}$$

$$= \left(\frac{1}{N}V_{TC}^2[0.4624 + 0.0324(N-1)]\right)^{1/2}$$

$$= V_{TC}\left[\frac{0.4624 + 0.0324(N-1)}{N}\right]^{1/2}$$

N = number of backplanes

a. Analysis of Vrms (ON)



TL/B/5606-6

$$\text{Vrms (OFF)} = \left(\frac{1}{T}\int_{t0}^{t0+T} v^2(t)dt\right)^{1/2}$$

$$= \left(\frac{1}{N}\left[\int_0^1 (0.32\,V_{TC})^2 dt + \int_1^N (-0.18\,V_{TC})^2 dt\right]\right)^{1/2}$$

$$= \left(\frac{1}{N}V_{TC}^2[0.1024 + 0.0324(N-1)]\right)^{1/2}$$

$$V_{TC}\left[\frac{0.1024 + 0.0324(N-1)}{N}\right]^{1/2}$$

N = number of backplanes

b. Analysis of Vrms (OFF)

$$V_{TC} = 1/2\left[\frac{V_{rms}\ (OFF)}{\left(\frac{0.1024 + 0.0324(N-1)}{N}\right)} + \frac{V_{rms}\ (ON)}{\left(\frac{0.4624 + 0.0324(N-1)}{N}\right)}\right]$$

MUST BE GREATER THAN

TL/B/5606-7

Example: If N = 8
and Vrms (OFF) = 1.8V
and Vrms (ON) = 2.2V
then V_{TC} = 7.5V

FIGURE 5

FUNCTIONAL DESCRIPTION

Connecting an MM58201 to an LCD

The backplane and segment outputs of the MM58201 connect directly to the backplane and segment lines of the LCD. These outputs are designed to drive a display with a total "on" capacitance of up to 2000 pF. This is especially important for the backplane outputs, as it is usually the backplanes that have the most capacitance. As the capacitance of the output lines increases, the DC offset between a backplane and segment signal may increase. Most LCD displays specify that a maximum offset of 50 mV is acceptable. For backplane capacitance under 2000 pF the MM58201 guarantees an offset of less than 10 mV.

If the LCD display to be used has 24 segments per backplane or less, then each MM58201 should be configured as a "master" so that each one will generate its own set of backplane signals. However, if the LCD display has more than 24 segments per backplane, more than one MM58201 will be needed for each backplane. To synchronize the driving signals there must be one "master" chip and then an additional "slave" chip for every 24 segments after the first 24. When a chip is configured as a "slave" it does not generate its own backplane signals. It simply synchonizes itself to the backplane signals generated by a "master" chip by sensing the BP1 signal. An example of both an all "master" configuration and a "master-slave" configuration will be shown later.

Voltage Control Pin and Circuitry

The voltage presented at the V_{TC} pin determines the actual voltage that is output on the backplane and segment lines. These voltages are shown in *Figure 7*. V_{TC} should be set with respect to Vrms (ON) and Vrms (OFF) and can be calculated as shown in *Figure 5*.

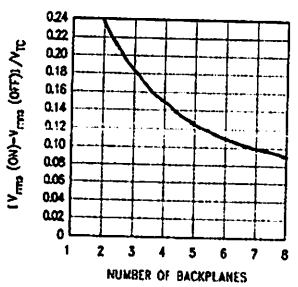
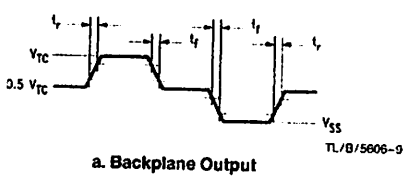
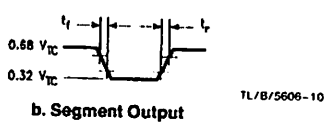


FIGURE 6. ΔVrms/V_{TC}



a. Backplane Output



b. Segment Output

FIGURE 7. Output Voltages

Since the input impedance of V_{TC} may vary between 10 kΩ and 30 kΩ, the output impedance of the voltage reference at V_{TC} should be relatively low. One example of a V_{TC} driver is shown in *Figure 8*. To put the MM58201 in a standby mode, bring V_{TC} to V_{SS} (ground). This will blank out the display and reduce the supply current to less than 300 μA.

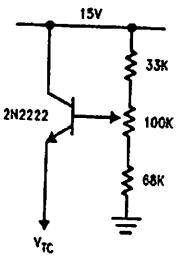


FIGURE 8. Example of V_{TC} Driver

RC Oscillator

This oscillator works with an external resistor tied to V_{DD} and an external capacitor tied to V_{SS}. The frequency of oscillation is related to the external R and C by:

$$f_{OSC} = 1/1.25\,RC \pm 30\%$$

The value of the external resistor should be in the range from 10 kΩ to 1 MΩ. The value of the external capacitor should be less than 0.005 μF.

The oscillator generates the timing required for multiplexing the LCD. The frequency of the oscillator is 4N times the refresh rate of the display, where N is the number of backplanes programmed. Since the refresh rate should be in the range from 32 Hz to 100 Hz, the oscillator frequency should be:

$$128N < f_{OSC} < 400N$$

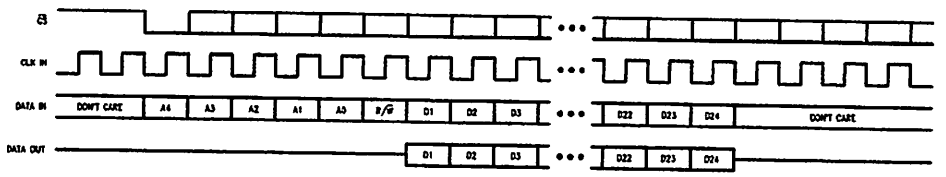
If the frequency is too slow, there will be a noticeable flicker in the display. If the frequency is too fast, there will be a loss of contrast between segments and an increase in power consumption.

Serial Input and Output

Data is sent to the MM58201 serially through the DATA IN pin. Each transmission must consist of 30 bits of information, as shown in *Figure 9*. The first five bits are the address, MSB first, of the first column of LCD segments that are to be changed. The next bit is a read or write flag. The following 24 bits are the actual data to be displayed.

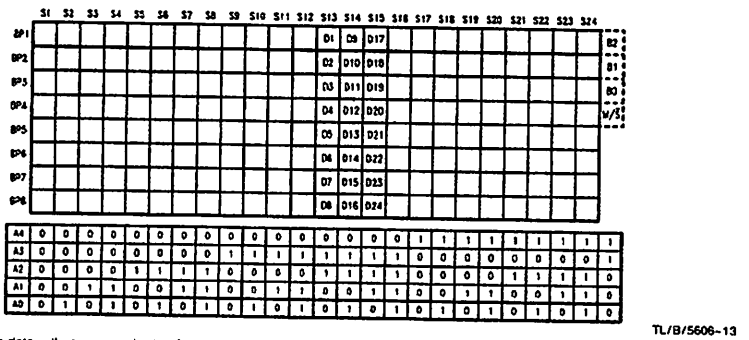
The address specifies the first LCD column that is going to be affected. The columns are numbered as shown in *Figure 10*. Data is always written in three column chunks. Twenty-four bits of data must always be sent, even if some of the backplanes are not in use. The starting column can be any number between one (00000) and twenty-four (10111). If column 23 or 24 is specified the displayed data will wrap around to column 1.

If the R/$\overline{W}$ bit is a "0" then the specified columns of the LCD will be overwritten with the new data. If the bit is a "1" then the data displayed in the specified columns will be available serially at the DATA OUT pin and the display will not be changed.



TL/B/5606-12

FIGURE 9. Transmission of Data



TL/B/5606-13

Diagram above shows where data will appear on display if starting address 01100 is specified in data format.

FIGURE 10. Address of Particular Segment Columns

The data is formatted as shown in *Figure 10*. The first bit in the data stream corresponds to backplane 1 in the first specified column. The second bit corresponds to backplane 2 in the first specified column and so on.

During initialization each MM58201 must be programmed to select how many backplanes are to be used, and whether the chip is to be a "master" or a "slave". The format of this transmission is just like a regular data transmission except for the following: the address must be 11000; the R/$\overline{W}$ must be a write (0); the first three data bits must be selected from the list in Table I. The next bit should be a "1" for the chip to be a master or a "0" for the chip to be a slave. The following 20 bits are necessary to complete the transmission but they will be ignored. The mode cannot be read back from the chip.

TABLE I. Backplane Select

Number of Backplanes	B2	B1	B0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1

The timing of the CLK, $\overline{CS}$, DATA IN, and DATA OUT are illustrated in *Figure 11*. The frequency of the clock can be between DC and 100 kHz with the shortest half-period being 5.0 μs. A transmission is initiated by $\overline{CS}$ going low. $\overline{CS}$ can then be raised anytime after the rising edge of the first clock pulse and before the rising edge of the last clock pulse (the clock edge that reads in D24). 30 bits of information must always be sent.

The data at DATA IN is latched on each rising edge of the clock pulse. The data at DATA OUT is valid after each falling edge of the last 24 clock pulses.

It is important to note that during a read or write transmission the LCD will display random bits. Thus the transmissions should be kept as short as possible to avoid disrupting the pattern viewed on the display. A recommended frequency is:

$f_{OSC} = 30/(t_{LCD} \quad 7\ t_S)$

t_{LCD} = turn on/off time of LCD

t_S = time between each successive transmission.

This should produce a flicker-free display.

The DATA OUT pin is an open drain N-channel device to V_{SS}. This output must be tied to V_{DD} through a resistor if it is to be used. It could also be tied to a lower voltage if this output is to be interfaced to logic running at a lower voltage. The value of the resistor is calculated by:

$R = (+V \cdot 0.4)/0.0006$

$+V$ = voltage of lower voltage logic

Power Supply

V_{DD} can range between 7V and 18V. A voltage should be used that is greater than or equal to the voltage that you calculate for V_{TC} as shown in *Figure 5*.

TYPICAL APPLICATIONS

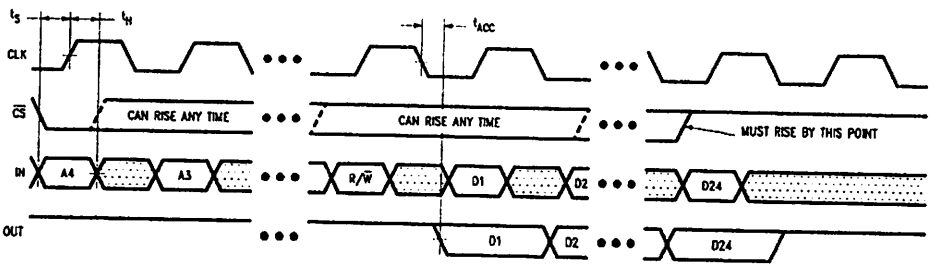
One application of the MM58201 is a general purpose display to show graphic symbols and text. This type of display could be used in an electronic toy or a small portable computer or calculator. One such display is shown in *Figure 12*. This display consists of four separate LCD displays that are built into one housing. Each separate LCD display has 8 backplanes and 24 segment lines. The entire display will require four MM58201s to control it.

The circuit diagram of this application is shown in *Figure 14*. Each separate LCD display is driven by one MM58201. The backplanes are driven by the separate MM58201s and are not paralleled together. There are three common lines: CLK, DATA IN, and DATA OUT. The CLK and DATA IN are generated from an output port such as an INS8255. Four other bits of the output port generate a linear select with a different bit going to each MM58201 chip select as shown in *Figure 13*. DATA OUT is sent to one bit of an input port.

The V_{TC} driver is as described beforehand. The MM74C906 is an open drain CMOS buffer that has near regular TTL compatible inputs. This is to provide level translation from the 5V supply of the computer system to the 12V supply of the MM58201.

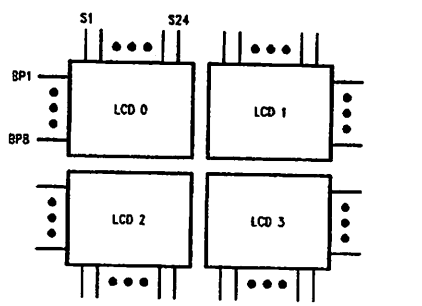
If I/O ports are not available, the circuit in *Figure 15* could be used as an interface between the MM58201s and a microprocessor bus.

To reduce the number of connections between the circuit and the LCD, all of the backplanes could have been driven by one MM58201 as shown in *Figure 16*. The other MM58201s would be configured as "slaves" synchronized to the one "master" MM58201. This would save 24 connections to the LCD but would increase the capacitance of the backplanes. In this application the capacitance is not a problem with either setup.



TL/B/5606-14

FIGURE 11. Timing of One Transmission



TL/B/5606-15

FIGURE 12. Four Separate LCD Displays Positioned to Look Like One Display

7	6	5	4	3	2	1	0
DATA IN	CLK	X	X	$\overline{CS4}$	$\overline{CS3}$	$\overline{CS2}$	$\overline{CS1}$

$\overline{CS4}$	$\overline{CS3}$	$\overline{CS2}$	$\overline{CS1}$	
1	1	1	0	Chip 1 Selected
1	1	0	1	Chip 2 Selected
1	0	1	1	Chip 3 Selected
0	1	1	1	Chip 4 Selected
1	1	1	1	No Chip Selected

FIGURE 13. Chip Select Scheme

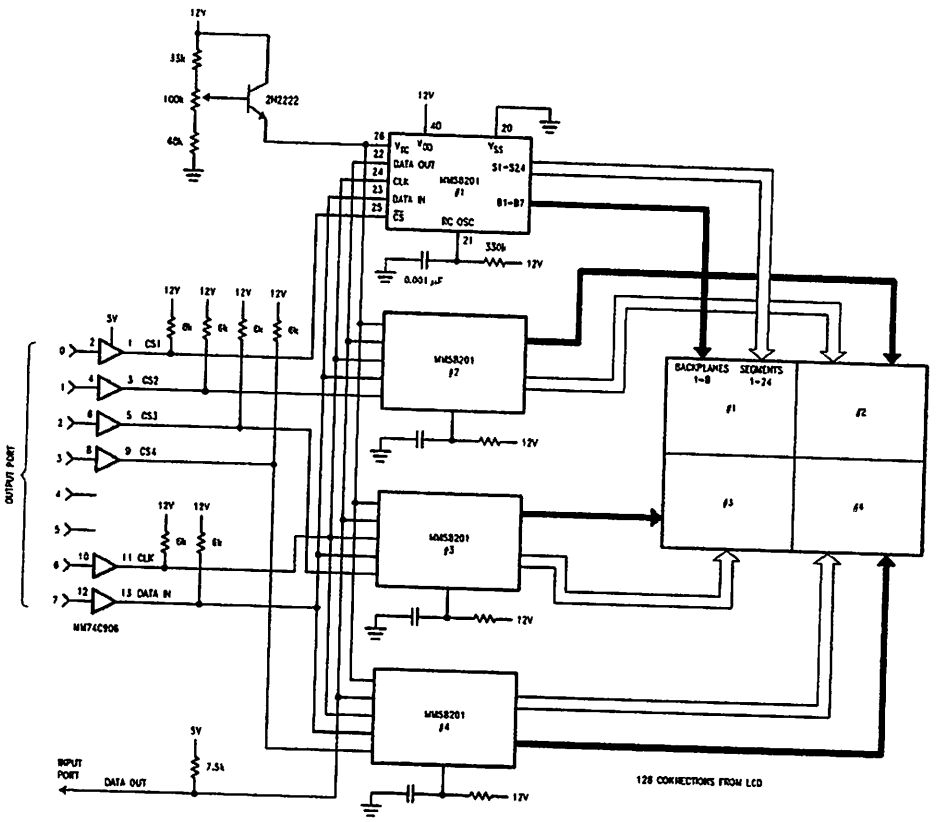
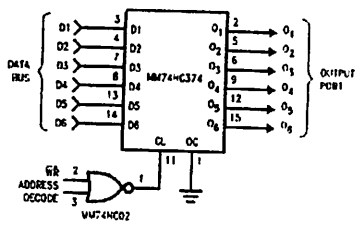


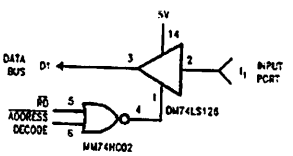
FIGURE 14. Diagram of Application

TL/B/5606–16



TL/B/5606–17

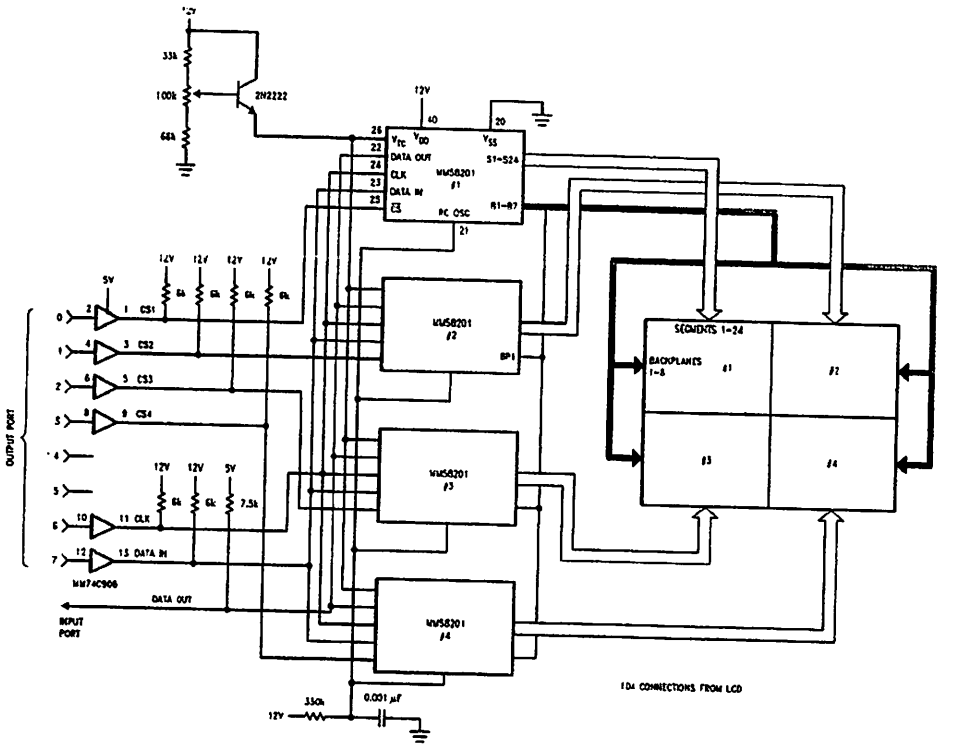
a. Output Port



TL/B/5606–18

b. Input Port

FIGURE 15. Input and Output Ports for Interface



TL/B/5606-19

FIGURE 16. Diagram of a Master-Slave Set-Up Not Used for This Application

SOFTWARE

The real heart of this system is the software which consists of four parts. Part one is the initialization portion. This sets up the MM58201s as "masters" and programs them for 8 backplanes. It then sets up the needed pointers for the other subroutines which consist of:

1) GRAPH: displays pattern on LCD.

2) TEXT: prints ASCII characters on display.

3) SCROLL: scrolls whatever pattern is displayed to the right until LCD is cleared.

This application used an NSC800™ with 8080 mnemonics. It could easily be adapted for other microprocessors.

MAIN

This program initializes the MM58201s. It controls the sequence of display output by calling other programs.

It first sends out a "dummy" transmission to make sure that the chips are ready to respond to a valid transmission. It then programs the chips to be "masters" and to use eight backplanes.

After initialization, this program sets up the correct pointers to display a graphic symbol. First it displays the upper eight bits of it, then it displays the lower eight bits.

The words "TESTING MM58201" are then displayed. A call to scroll then causes this to scroll to the right until the screen is blank. Finally the words "END OF TEST" appear and the program ends.

The method to create a custom graphic symbol will be demonstrated in the next section.

```
        NS030
        EXTRN GRAPH,WRITE,MODE,TEXT,CURSOR,SCROLL

;INITIALIZE THE STACK POINTER
        LXI SP,1FFFH

;INITIALIZE THE 810
;SET MODE 0 FOR PORT A
;INIT:  MVI A,00H
        OUT 27H
;SET PORT A AS OUTPUT AND PORT C AS INPUT
        MVI A,0FFH
        OUT 24H                 ;PORT A DDR
        MVI A,00H
        OUT 26H                 ;PORT B DDR

;INITIALIZE THE FOUR 58201'S
        MVI A,0                 ;SET FOR WRITE MODE
        STA MODE
        LXI H,MASTER,           ;SEND A COMPLETE TRANSMISSION TO CLEAR OUT
        MVI E,11000B            ;   ANY OLD CHIP SELECT.
        MVI D,00001110B
        CALL WRITE
        LXI H,MASTER            ;CONFIGURE CHIPS 0, 1, 2, AND 3 AS MASTERS
        MVI D,00001110B
        CALL WRITE
        LXI H,MASTER
        MVI D,00001101B
        CALL WRITE
        LXI H,MASTER
        MVI D,00001011B
        CALL WRITE
        LXI H,MASTER
        MVI D,00000111B
        CALL WRITE

;SET UP POINTER AND COUNTERS TO DISPLAY NATIONAL SEMI SYMBOL
        MVI B,21                ;B HOLDS # OF COLUMNS TO CHANGE
RESTRT: MVI D,0                 ;D HOLDS THE STARTING COLUMN NUMBER FOR UPPER HALF
        MVI E,48                ;E HOLDS STARTING COLUMN NUMBER FOR LOWER HALF
DSLOOP: MOV C,D
        LXI H,NATSM1            ;DISPLAY UPPER HALF OF GRAPHIC
        CALL GRAPH
        LXI H,NATSM2            ;DISPLAY LOWER HALF OF GRAPHIC
        MOV C,E
        CALL GRAPH

        LXI H,0FFFFH            ;PAUSE
PAUSE:  DCX H
        MOV A,H
        ORA L
        JNZ PAUSE

        INR D                   ;INCREMENT STARTING COLUMN NUMBERS
        INR D
        INR D
        INR E
        INR E
        INR E
        MVI A,30                ;DISPLAY IT UNTIL COLUMN COUNT IS 30
        CMP D
        JNZ DSLOOP

        LXI H,TEXT1             ;PRINT FIRST TEXT
        MVI A,0                 ;ZERO THE CURSOR
        STA CURSOR
        CALL TEXT

        CALL SCROLL             ;SCROLL THE TEXT

        LXI H,TEXT2             ;PRINT SECOND TEXT
        MVI A,0                 ;ZERO THE CURSOR
        STA CURSOR
        CALL TEXT

        LXI H,0FFFFH            ;PAUSE
PAUSE1: DCX H
        MVI A,2
PAUSE2: DCR A
        JNZ PAUSE2
        MOV A,H
        ORA L
        JNZ PAUSE1
```

```
        LXI H,TEXT3             :PRINT THIRD TEXT
        MVI A,0
        STA CURSOR
        CALL TEXT

        RST 6                   :END

TEXT1:  DB 'TESTING MM58201 ', 0
TEXT2:  DB 'THIS IS THE END ', 0
TEXT3:  DB ' OF THE   TEST  ', 0

MASTER: DB 1111B                :ADDRESS FOR MASTER
SLAVE:  DB 0111B                :ADDRESS FOR SLAVE

NATSM1: DB 0FFH, 0FFH, 0FFH, 7FH, 3FH, 9FH, 0CFH, 67H, 33H, 01H, 7FH
        DB 3FH, 9FH, 0CFH, 67H, 33H
        DB 99H, 0FFH, 0FFH, 00H, 00H
NATSM2: DB 0FFH, 0FFH, 0FFH, 0E6H, 0F3H, 0F9H, 0FCH, 0FEH, 0FFH
        DB 0E0H, 0E6H, 0F3H, 0F9H, 0FCH
        DB 0FEH, 0FFH, 0FFH, 0FFH, 0FFH, 00H, 00H
        END
```

GRAPH

This subroutine is the center of the software. It is the interface between the calling programs and the hardware. All I/O is generated by this subroutine.

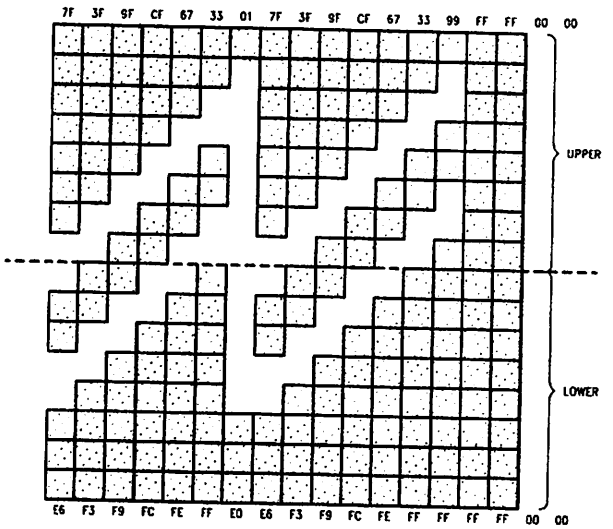
There are two entrances to this subroutine: graph and read. Graph is the entrance used to display new data. Read is the entrance used to read data from the display.

The HL register should point to the beginning of the data to be displayed. The B register should hold the number of columns to change. This must be a multiple of three. The C register should hold the column number to start with. This must also be a multiple of three. These restrictions are to simplify the software.

The first operation is the calculation of the correct chip to enable and the column number to start within that chip. The first bit of the column address is output with the correct chip select going low. The rest of the column address is then output with all the chip selects high. If the operation is a write, the data is sent to the display bit by bit. If the operation is a read, the data is read in bit by bit.

To create a custom graphic symbol, draw it on a grid as shown in *Figure 17*. Group the upper eight squares as a byte with the least significant bit at the top, counting a dark square as a one. Group the lower eight squares as a byte with the most significant bit at the bottom. Use this generated data as input lists to the graph subroutine. A good example of this is shown in the listing of main when it calls graph.

Pad the data at the end with zeros as shown to keep the number of data values a multiple of three. Remember, this is only a software restriction. A different routine could be used that would allow any number of columns to be displayed.



TL/B/5606-20

FIGURE 17. Example Graphic Symbol

```
        N8080
        PUBLIC GRAPH, READ, WRITE, MODE
:GRAPHIC DISPLAY DRIVER
:       INPUT:  HL - POINTS TO START OF DATA
:               B- # OF 8 BIT COLUMNS TO CHANGE (MUST BE MULT. OF 3)
:               C- COLUMN # TO START WITH (MUST BE MULT. OF 3)
:       OUTPUT: NO REGISTERS DISTURBED
:               DATA POINTED TO IS DISPLAYED ON LCD DISPLAY.
:               COLUMNS NOT SPECIFIED ARE NOT AFFECTED.
READ:
:SAVE ALL STATES
        PUSH PSW
        PUSH B
        PUSH D
        PUSH H
;FLAG FOR A READ OPERATION
        MVI A,10000000B
        STA MODE
        JMP GRAPH1
GRAPH:
:SAVE ALL STATES
        PUSH PSW
        PUSH B
        PUSH D
        PUSH H
;FLAG FOR A WRITE OPERATION
        MVI A,0
        STA MODE
;CALCULATE WHICH 58201 TO ACCESS
GRAPH1: MVI D,0EEH              ;START WITH CS1
ACC:    MOV A,C
        SUI 24                  ;SUBTRACT 24 FROM COLUMN COUNT
        JC GO                   ;IF CARRY IS SET THE CORRECT CHIP IS SELECTED
        MOV C,A                 ;REG C GETS NEW COLUMN NUMBER
        MOV A,D
        RLC                     ;INCREMENT THE CS TO NEXT CHIP
        MOV D,A
        JMP ACC
:MAIN LOOP
GO:     MOV E,C                 ;GET COLUMN NUMBER
M.LOOP: CALL WRITE              ;DRAW 3 COLUMNS
        DCR B                   ;SUBTRACT 3 FROM COLUMN COUNT
        DCR B
        DCR B
        JZ END.G                ;IF DONE, JUMP.
        MOV A,E                 ;ADD 3 TO ADDRESS
        ADI 3
        CPI 11000B              ;IF ADDRESS NOT MAX THEN SKIP THIS
        JNZ SKIP1
        MOV A,D
        RLC                     ;SELECT NEXT 58201 CS
        MOV D,A
        MVI A,0
SKIP1:  MOV E,A                 ;SAVE NEXT ADDRESS
        JMP M.LOOP              ;LOOP UNTIL DONE
END.G:  POP H                   ;RESTORE ALL STATES
        POP D
        POP B
        POP PSW
        RET
WRITE:
:       DISPLAY 3 COLUMNS OF DATA
:       INPUT:  HL- POINTS TO START OF DATA
:               E - ADDRESS
:               D - OUTPUT CS
:       OUTPUT: HL <- HL + 3
:SAVE ALL STATES
        PUSH PSW
        PUSH B
        PUSH D
START:  MVI A,00001111B ;ISOLATE CS IN REG D
        ANA D
        MOV D,A
        MOV A,E                 ;GET ADDRESS BITS AT HIGH END OF BYTE
        RLC
        RLC
        MOV E,A
```

```
  ;OUTPUT FIVE ADDRESS BITS WITH CHIP SELECT
          MVI C,5
  W.LOOP: MOV A,E
          RLC                         ;ROTATE ADDRESS
          MOV E,A
          MVI A,10000000B
          ANA E                       ;GET MSB
          ORA D                       ;MERGE WITH CHIP SELECT
          CALL DISPLY
          DCR C                       ;DEC ADDRESS BIT COUNTER
          JNZ W.LOOP                  ;LOOP UNTIL ADDRESS IS OUT
  ;SIGNAL FOR A READ OR WRITE
          LDA MODE
          ORI 00001111B
          CALL DISPLY
          JP DIS0                     ;JUMP IF THIS IS A WRITE
  ;READ THE DATA
          MVI B,3                     ;3 BYTES OF DATA
  READ1:  MVI C,8                     ;8 BITS PER BYTE
          MVI D,0                     ;CLEAR DATA BYTE
  READ2:  IN 22H                      ;GET A BIT OF DATA
          ANI 00000001B               ;MASK OFF UNWANTED BITS
          ORA D                       ;MERGE WITH DATA BYTE
          RRC                         ;ROTATE DATA
          MOV D,A
          MVI A,00001111B             ;SET UP 58201 TO READ NEXT BIT
          CALL DISPLY
          DCR C                       ;LOOP UNTIL DONE WITH BYTE
          JNZ READ2
          MOV M,D
          INX H                       ;INCREMENT BYTE POINTER
          DCR B                       ;LOOP UNTIL DONE WITH ALL BYTES
          JNZ READ1
  ;RESTORE STATES
          POP D
          POP B
          POP PSW
          RET
  ;DISPLAY THE DATA
  DIS0:   MVI B,3                     ;3 BYTES OF DATA
  DIS1:   MVI C,8                     ;8 BITS PER BYTE
          MOV D,M
  DIS2:   MOV A,D                     ;ROTATE DATA
          RRC
          MOV D,A
          ANI 10000000B               ;GET NEXT BIT
          ORI 00001111B               ;SET CS
          CALL DISPLY                 ;OUTPUT A BIT OF DATA
          DRC C
          JNZ DIS2                    ;LOOP UNTIL DONE WITH BYTE
          INX H
          DCR B
          JNZ DIS1                    ;LOOP UNTIL DONE WITH 3 BYTES
  ;RESTORE STATES
          POP D
          POP B
          POP PSW
          RET
DISPLY:
;DISPLAY ROUTINE
;         INPUT:  A - DATA AND CHIP SELECT
;                           BIT 7 - DATA
;                           BITS 0-3 - CHIP SELECT
;         OUTPUT: NO REGISTERS DISTURBED
;                 OUTPUT ONE BIT TO 58201
          PUSH PSW                    ;SAVE STATES
          ANI 10001111B               ;MASK OFF UNWANTED BITS
          OUT 20H                     ;SET UP DATA AND CHIP SELECT
          ORI 01000000B               ;CLOCK HIGH
          OUT 20H
          ANI 10111111B               ;CLOCK LOW
          POP PSW                     ;RESTORE STATES
          RET
MODE:     DS 1
          END
```

TEXT

This subroutine will take the ASCII text pointed to by HL and display it on the LCD starting at the column pointed to by the memory location CURSOR. The data should end with a zero. CURSOR should be in the range of 0–15 as this is the extent of this LCD display. The first operation is the calculation of the offset into the ASCII table of the first character. Thirty-two is subtracted from the ASCII number because the table starts with a space character. This result is then multiplied by six because the data to be displayed is six bytes long. We now have the offset into the table. The character is displayed on the LCD. This operation is repeated until all the characters have been displayed.

A custom font can be generated using the same technique as that used to create a custom graphic symbol.

```
        NS080
        EXTRN GRAPH
        PUBLIC TEXT, LET          TR, CURSOR

TEXT:
;DISPLAY A CHARACTER STRING ON LCD DISPLAY
;       INPUT:  HL-POINTS TO BEGINNING OF STRING
;               CURSOR-CURRENT CURSOR POSITION
;       OUTPUT: CURSOR <= CURSOR + LENGTH OF STRING
;               NO REGISTERS DISTURBED

        PUSH PSW                  ;SAVE STATES
        PUSH H
T.LOOP: MOV A,M                   ;CHECK FOR END OF STRING
        CPI 0
        JZ T.FIN
        CALL LETTR                ;PRINT LETTER
        INX H
        JMP T.LOOP                ;LOOP UNTIL DONE
T.FIN:  POP H                     ;RESTORE STATES
        POP PSW
        RET


LETTR:
;DISPLAY AN ASCII CHARACTER ON LCD DISPLAY
;       INPUT:  A-CHARACTER TO DISPLAY
;               CURSOR-CURRENT CURSOR LOCATION (0 - 95)
;       OUTPUT: CURSOR <= CURSOR + 1
;               NO REGISTERS DISTURBED

;SAVE STATES
        PUSH PSW
        PUSH B
        PUSH D
        PUSH H

;SET UP HL TO POINT TO CORRECT DATA
        LXI H,ASCII               ;HL POINTS TO BASE ADDRESS
        MVI B,0                   ;BC GETS ASCII OFFSET MINUS A CONSTANT
        SUI 20H
        MOV C,A
        CALL MULT                 ;MULTIPLY OFFSET BY 6 (DOUBLE PRECISION)
        DAD B                     ;HL POINTS TO CORRECT CHARACTER DATA
        LDA CURSOR                ;MULTIPLY CURSOR BY 6 TO GET COLUMN NUMBER
        MOV B,A
        ADD B
        ADD B
        ADD B
        ADD B
        ADD B
        MOV C,A
        MVI B,6                   ;EACH CHARACTER IS SIX COLUMNS WIDE
        CALL GRAPH                ;DISPLAY THE CHARACTER
        LDA CURSOR                ;INCREMENT CURSOR
        INR A
        CPI 16                    ;CHECK FOR END OF LCD DISPLAY
        JNZ T.END
        MVI A,0                   ;IF SO, RESET TO ZERO
T.END:  STA CURSOR

;RESTORE STATES
        POP H
        POP D
        POP B
        POP PSW
        RET
```

```
MULT:
;MULTIPLY BC REG BY SIX
;       INPUT:  BC - MULTIPLICAND
;       OUTPUT: BC <= BC * 6
;               NO REGISTERS DISTURBED
        PUSH PSW
        PUSH H
        MOV H,B
        MOV L,C
        DAD B
        DAD B
        DAD B
        DAD B
        DAD B
        MOV B,H
        MOV C,L
        POP H
        POP PSW
        RET
CURSOR: DS 1
ASCII:  DB 0,0,0,0,0,0          ;SPACE
        DB 0,95,95,0,0,0        ;!
        DB 0,7,0,7,0,0          ;"
        DB 20,127,20,127,20,0   ;#
        DB 36,42,127,42,18,0    ;$
        DB 35,19,8,100,98,0     ;%
        DB 54,73,102,32,80,0    ;&
        DB 0,0,7,0,0,0          ;'
        DB 0,28,34,65,0,0       ;(
        DB 0,65,34,28,0,0       ;)
        DB 34,20,127,20,34,0    ;*
        DB 8,8,62,8,8,0         ;+
        DB 0,64,48,0,0,0        ;'
        DB 8,8,8,8,8,0          ;-
        DB 0,96,96,0,0,0        ;.
        DB 32,16,8,4,2,0        ;/

        DB 62,81,73,69,62,0     ;0
        DB 0,66,127,64,0,0      ;1
        DB 122,73,73,73,70,0    ;2
        DB 34,65,73,73,54,0     ;3
        DB 15,8,8,126,8,0       ;4
        DB 39,69,69,69,57,0     ;5
        DB 62,73,73,73,49,0     ;6
        DB 1,97,17,9,7,0        ;7
        DB 54,73,73,73,54,0     ;8
        DB 6,9,9,9,126,0        ;9
        DB 0,54,54,0,0,0        ;:
        DB 96,54,54,0,0,0       ;;
        DB 8,20,34,65,0,0       ;<
        DB 20,20,20,20,20,0     ;=
        DB 0,65,34,20,8,0       ;>
        DB 2,1,88,5,2,0         ;?
        DB 62,65,93,89,78,0     ;@

        DB 124,18,17,18,124,0   ;A
        DB 127,73,73,73,54,0    ;B
        DB 62,65,65,65,34,0     ;C
        DB 127,65,65,65,62,0    ;D
        DB 127,73,73,65,65,0    ;E
        DB 127,9,9,1,1,0        ;F
        DB 62,65,65,81,114,0    ;G
        DB 127,8,8,8,127,0      ;H
        DB 0,65,127,65,0,0      ;I
        DB 32,64,64,64,63,0     ;J
        DB 127,8,20,34,65,0     ;K
        DB 127,64,64,64,64,0    ;L
        DB 127,2,12,2,127,0     ;M
        DB 127,4,8,16,127,0     ;N
        DB 62,65,65,65,62,0     ;O
        DB 127,9,9,9,6,0        ;P
        DB 62,65,81,33,94,0     ;Q
        DB 127,9,25,41,70,0     ;R
        DB 34,69,73,81,34,0     ;S
        DB 1,1,127,1,1,0        ;T
        DB 63,64,64,64,63,0     ;U
        DB 31,32,64,32,31,0     ;V
        DB 127,32,24,32,127,0   ;W
        DB 99,20,8,20,99,0      ;X
        DB 3,4,120,4,3,0        ;Y
        DB 97,81,73,69,67,0     ;Z
END
```

SCROLL

This subroutine will scroll whatever is displayed on the LCD to the right until the screen is clear. It first reads in three columns of data. It then writes three columns of data with the HL pointer shifted by one byte. This will shift the displayed data by one column. This is repeated until the entire LCD has been shifted by one column. Then the entire operation is repeated until all the displayed data is shifted off the screen.

This subroutine could easily be adapted to scroll the display to the left if desired.

```
        N8080
        PUBLIC SCROLL
        EXTRN READ,GRAPH

SCROLL:
;SCROLLS DISPLAY TO THE RIGHT UNTIL CLEAR
;       INPUT:  NONE
;       OUTPUT: NO REGISTERS ARE CHANGED
;               SCREEN IS SCROLLED UNTIL CLEAR

;SAVE ALL STATES
        PUSH PSW
        PUSH B
        PUSH D
        PUSH H

;SET UP ALL THE POINTERS
        MVI D,96                ;LOOP UNTIL SCREEN IS CLEAR (96 CYCLES)
REPEAT: MVI A,0                 ;CLEAR FIRST BYTE IN BUFFER
        STA BUFFER
        MVI B,3                 ;READ 3 COLUMNS ALWAYS
        MVI C,0                 ;START WITH COLUMN ZERO

;READ THE DATA
L.READ: LXI H,BUFFER+1          ;SET HL TO POINT TO BUFFER+1
        CALL READ
        LXI H,BUFFER            ;SET HL TO SHIFT THE DATA
        CALL GRAPH              ;REDRAW THE SHIFTED DATA

;MOVE LAST COLUMN OF LAST READ INTO FIRST COLUMN OF NEXT WRITE
        LDA BUFFER+3
        STA BUFFER

;UPDATE COUNTERS
        MOV A,C                 ;INCREMENT COLUMN NUMBER
        ADI 3
        MOV C,A
        CPI 96                  ;CHECK IF DONE WITH ONE CYCLE
        JNZ L.READ
        DCR D                   ;DECREMENT LOOP COUNT
        JNZ REPEAT              ;LOOP UNTIL DONE WITH ALL CYCLES

;RESTORE STATES
        POP H
        POP D
        POP B
        POP PSW
        RET

BUFFER: DS 4

        END
```

OTHER APPLICATIONS

There are many different types of LCDshat can be controlled by the MM58201. Some of these ar shown in *Figure 18*.

Up to 24 seven-segment digits can be ontrolled by one MM58201. The software to control a multipxed seven-segment display is not too much different from iat of the previous application. The software is simpler beause only one MM58201 is needed instead of four. A logi diagram for a six-digit multiplexed seven-segment LCD disay is shown in *Figure 19* and the software to control it is inisting 5.

Given a string of numbers to display, this subroutine simply looks up the data it needs from a look-up table and stores this data in a buffer. After every three digits, the subroutine sends this data to the MM58201 to be displayed. The digit backplanes are wired backward in groups of three to simplify the software. The subroutines that this subroutine uses are very similar to the equivalent subroutines in the LCD dot matrix application. Since there is only one MM58201, the software is simpler. There is no need to calculate which MM58201 chip select to enable.

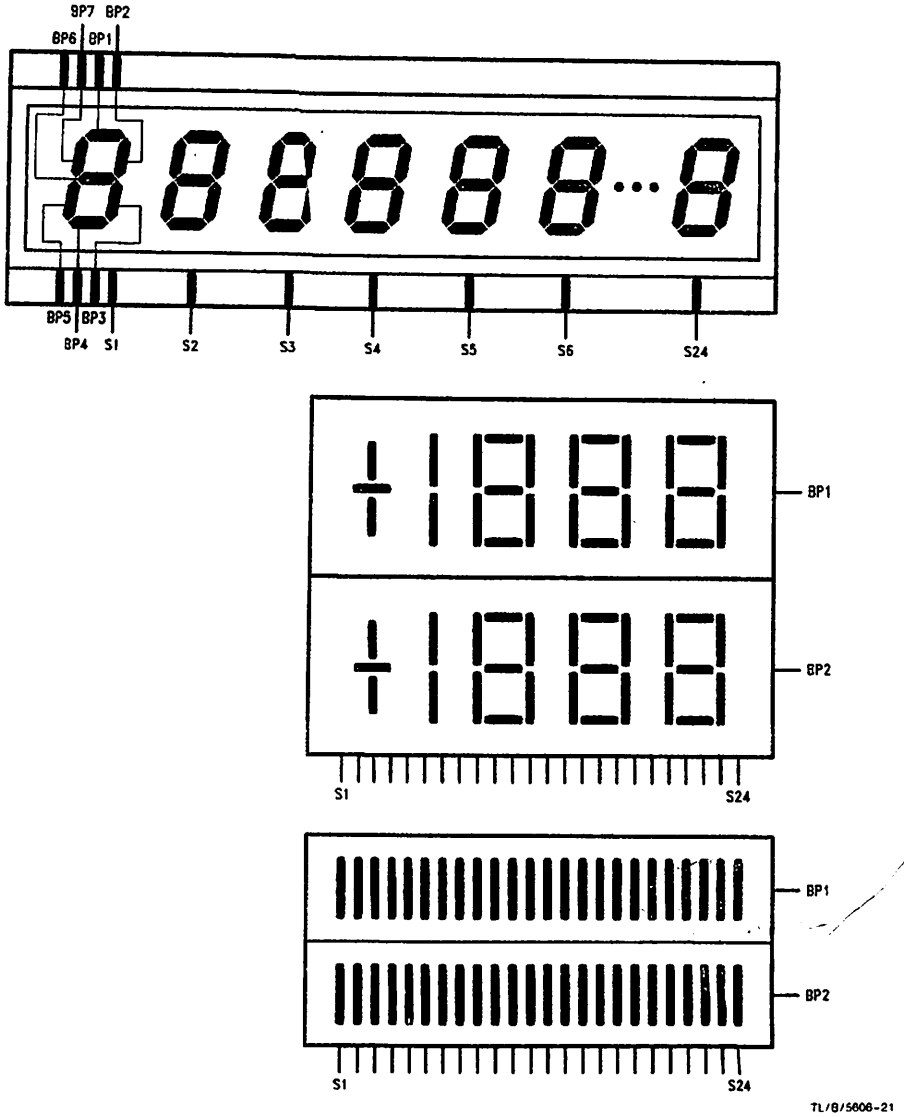


FIGURE 18. Typical LCD Connections to the MM58201

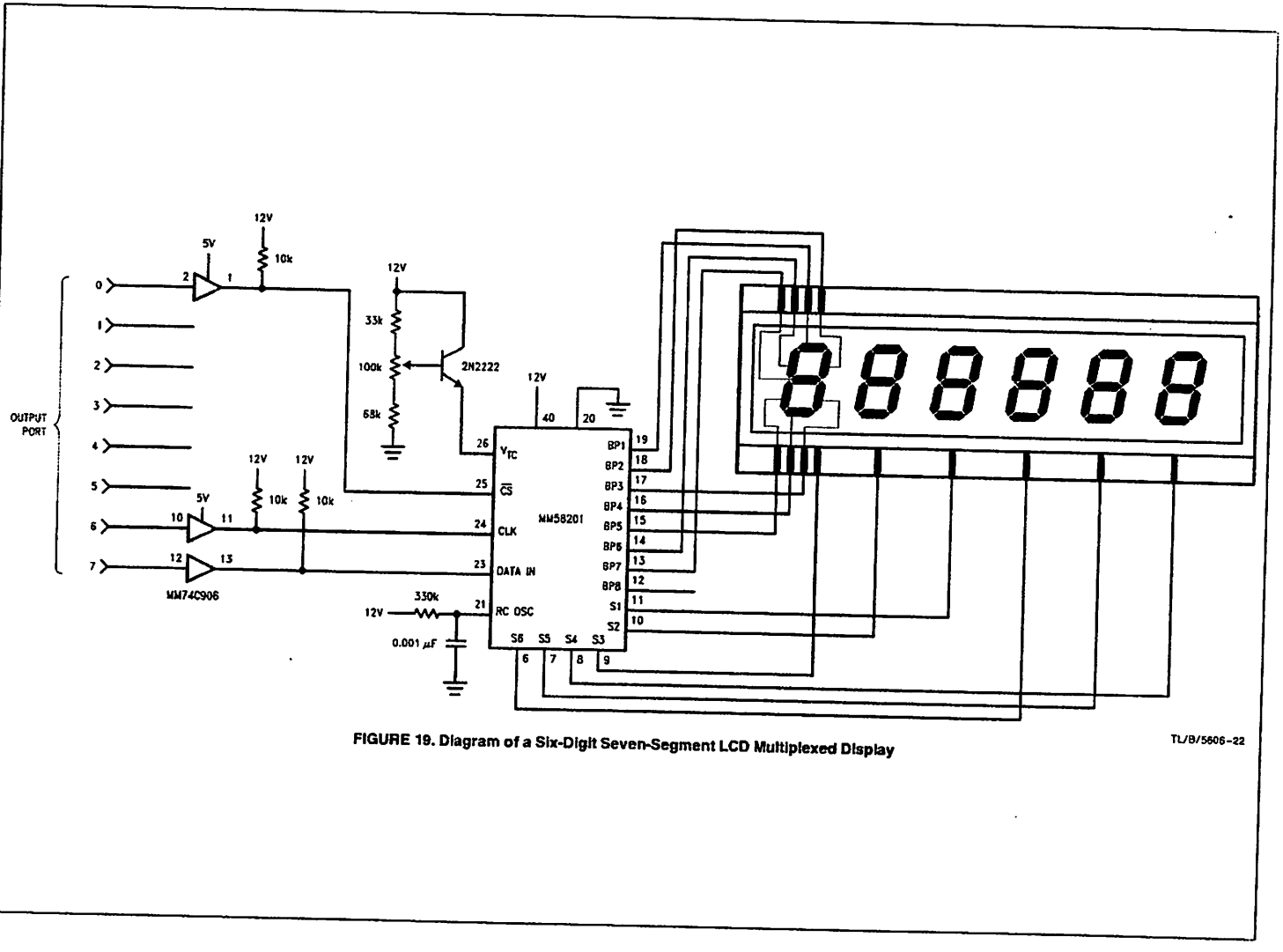


FIGURE 19. Diagram of a Six-Digit Seven-Segment LCD Multiplexed Display

TL/B/5606-22

```
          7800H
  ;INITIALIZE THE 810
          MV A,0
          OUT 27H
          MVI A,0FFH
          OUT 24H

          LXI BC,TEST
          MVI E,6
          CALL NUMBER
          RST 6

  TEST:   DB 1,2,3,4,5,6

  ;SUBROUTINE TO DISPLAY NUMERALS ON LCD DISPLAY
  ;       INPUT   BC-POINTS TO BCD DATA STRING
  ;               E -LENGTH OF DATA STRING (MULTIPLE OF 3)
  ;       OUTPUT    -NO REGISTERS DISTURBED
  ;                 -DATA STRING IS DISPLAYED
  ;
  NUMBER: PUSH PSW                  ;SAVE STATES
          PUSH B
          PUSH D
          PUSH H

  DIG3:   MVI D,3                   ;LOOP FOR 3 DIGITS
  LOOP:   LDAX B
          LXI H,TABLE               ;CALCULATE ADDRESS INTO TABLE
          ADD L
          MOV L,A
          MVI A,00H
          ADC H
          MOV H,A

          MOV A,M                   ;GET OUTPUT DATA FROM TABLE
          PUSH PSW

          LXI H,DATA                ;STORE INTO DATA BUFFER
          MOV A,L
          ADD D
          MOV L,A
          DCR L
          POP PSW
          MOV M,A

          INX B                     ;INCREMENT POINTER TO DATA STRING
          DCR E                     ;DECREMENT # OF DIGITS
          DCR D                     ;DECREMENT 3 DIGIT COUNT
          JNZ LOOP                  ;IF NOT THIRD DIGIT THEN LOOP BACK

          LXI H,DATA
          CALL WRITE                ;DISPLAY THESE THREE DIGITS

          MOV A,E                   ;CHECK FOR LAST DIGIT OF DATA STRING
          ANA A
          JNZ DIG3

          POP H                     ;RESTORE STATES
          POP D
          POP B
          POP PSW
          RET

  WRITE:
  ; DISPLAY 3 DIGITS
  ;       INPUT   HL-POINTS TO START OF DATA
  ;               E -COLUMN ADDRESS
  ;       OUTPUT    -NO REGISTERS DISTURBED
  ;
          PUSH PSW                  ;SAVE STATES
          PUSH B
          PUSH D
          PUSH H

          MOV A,E                   ;GET ADDRESS BITS AT HIGH END OF BYTE
          RLC
          RLC
          MOV E,A
```

```
;OUTPUT FIVE ADDRESS BITS
        MVI C,5
W.LOOP: MOV A,E
        RLC                         ;ROTATE ADDRESS
        MOV E,A
        MVI A,10000000B             ;GET MSB & ENABLE CHIP SELECT BIT
        ANA E
        CALL OUT                    ;OUTPUT BIT WITH CHIP SELECT
        DCR C
        JNZ W.LOOP                  ;LOOP UNTIL ADDRESS IS OUT

;SIGNAL FOR A WRITE
        MVI A,00H
        CALL OUT                    ;OUTPUT A ZERO BIT

;OUTPUT THE DATA
        MVI B,3                     ;3 BYTES OF DATA
DIS1:   MVI C,8                     ;8 BITS PER BYTE
        MOV D,M
DIS2:   MOV A,D                     ;ROTATE DATA
        RRC
        MOV D,A
        ANI 10000000B               ;GET NEXT BIT
        ORI 00000001B               ;DISABLE CHIP SELECT
        CALL OUT
        DCR C
        JNZ DIS2                    ;LOOP UNTIL DONE WITH BYTE
        INX H
        DCR B
        JNZ DIS1                    ;LOOP UNTIL DONE WITH 3 BYTES

        POP H                       ;RESTORE STATES
        POP D
        POP B
        POP PSW
        RET

OUT:
;SUBROUTINE TO OUTPUT ONE BIT TO THE MM58201
;       INPUT   A -DATA BIT IN MSB POSITION
;       OUTPUT    -NO REGISTERS DISTURBED
;                 -OUTPUT ONE BIT TO 58201

        PUSH PSW
        OUT 20H
        ORI 01000000B               ;CLOCK HIGH
        OUT 20H
        ANI 10111111B               ;CLOCK LOW
        OUT 20H
        POP PSW
        RET

DATA:   DS 3
TABLE:  DB 00111111B, 00000110B, 01011011B, 01001111B
        DB 01100110B, 01101101B, 01111101B, 00000111B
        DB 01111111B, 01101111B

        END
```

SUMMARY

The MM58201 makes it easy to interface a multiplexed LCD display to a microprocessor. It is simply a matter of connecting the display and the microprocessor to the chip, choosing a value for V_{CT}, then interfacing your program to use the subroutines listed here or similar ones. Multiplexed LCDs are the perfect way to cut down on display interconnections while still taking advantage of the LCD's low power consumption and high contrast ratio—and the MM58201 makes them easy to use.

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408

National Semiconductor

December 1995

LM124/LM224/LM324/LM2902 Low Power Quad Operational Amplifiers

General Description

The LM124 series consists of four independent, high gain, internally frequency compensated operational amplifiers which were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

Application areas include transducer amplifiers, dc gain blocks and all the conventional op amp circuits which now can be more easily implemented in single power supply systems. For example, the LM124 series can be directly operated off of the standard +5V power supply voltage which is used in digital systems and will easily provide the required interface electronics without requiring the additional ±15V power supplies.

Unique Characteristics

- In the linear mode the input common-mode voltage range includes ground and the output voltage can also swing to ground, even though operated from only a single power supply voltage
- The unity gain cross frequency is temperature compensated
- The input bias current is also temperature compensated

Advantages

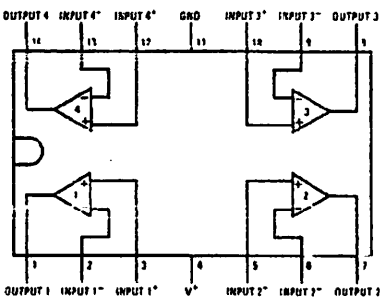
- Eliminates need for dual supplies
- Four internally compensated op amps in a single package
- Allows directly sensing near GND and VOUT also goes to GND
- Compatible with all forms of logic
- Power drain suitable for battery operation

Features

- In the linear mode the input common-mode voltage range includes ground and the output voltage can also swing to ground, even though operated from only a single power supply voltage
- Unity gain cross frequency temperature compensated
- Input bias current temperature compensated
- Large DC voltage gain 100 dB
- Wide bandwidth (unity gain) 1 MHz (temperature compensated)
- Wide power supply range
  - Single supply 3V to 32V
  - or dual supplies ±1.5V to ±16V
- Very low supply current drain (700 µA) — essentially independent of supply voltage
- Low input biasing current 45 nA (temperature compensated)
- Low input offset voltage 2 mV and offset current 5 nA
- Input common-mode voltage range includes ground
- Differential input voltage range equal to the power supply voltage
- Large output voltage swing 0V to V+ − 1.5V

Connection Diagrams

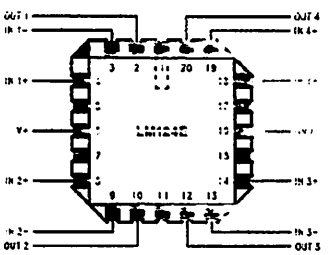
Dual-In-Line Package



TL/H/9299–1

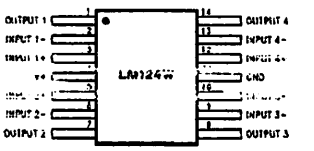
Top View

Order Number LM124J, LM124AJ, LM124J/883, LM124AJ/883*, LM224J, LM224AJ, LM224AM, LM324M, LM324AM, LM2902M, LM324N, LM324AN, LM2902N, LM2902V and LM2902VAN**
See NS Package Number J14A, M14A or N14A



TL/H/9299–32

Order Number LM124AE/883, LM224AE/883 and LM324AE/883
See NS Package Number E20A



TL/H/9299–33

Order Number LM124AW/883 or LM224AW/883
See NS Package Number W14B

*LM124A available per JM38510/11006
**LM124 available per JM38510/11005

 TL/H/9299

RRD-B30M115/Printed in U. S. A.

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.
(Note 9)

	LM124/LM224/LM324 LM124A/LM224A/LM324A	LM2902
Supply Voltage, V^+	32V	26V
Differential Input Voltage	32V	26V
Input Voltage	−0.3V to +32V	−0.3V to +26V
Input Current (V_{IN} < 0.3V) (Note 3)	50 mA	50 mA
Power Dissipation (Note 1) Molded DIP Cavity DIP Small Outline Package	1130 mW 1260 mW 800 mW	1130 mW 1260 mW 800 mW
Output Short-Circuit to GND (One Amplifier) (V^+ = 15V and T_A = 25°C)	Continuous	Continuous
Operating Temperature Range LM2902 LM324 LM224 LM124		−40°C to +85°C 0°C to +70°C −25°C to +85°C −55°C to +125°C

	LM124/LM224/LM324 LM124A/LM224A/LM324A	LM2902
Storage Temperature Range	−65°C to +150°C	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	260°C	260°C
Soldering Information Dual-In-Line Package Soldering (10 seconds) Small Outline Package Vapor Phase (60 seconds) Infrared (15 seconds)	260°C 215°C 220°C	260°C 215°C 220°C
See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.		
ESD tolerance (Note 10)	250V	250V

Electrical Characteristics V^+ = 5.0V (Note 4), unless otherwise stated

Parameter	Conditions	LM124A			LM224A			LM324A			LM124/LM224			LM324			LM2902			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage	(Note 5) T_A = 25°C		1	3		2	5		2	5		2	7		2	7		2	7	mV
Input Bias Current	($I_{IN(+)}$ or $I_{IN(-)}$) V_{CM} = 0V, T_A = 25°C		20	50		40	80		40	100		45	150		45	250		45	250	nA
Input Offset Current	($I_{IN(+)}$ − $I_{IN(-)}$) V_{CM} = 0V, T_A = 25°C		2	15		2	15		3	30		3	30		5	50		5	50	nA
Input Common-Mode Voltage Range	V^+ = 30V (LM2902, V^+ = 26V), T_A = 25°C	0		V^+ − 1.5	0		V^+ − 1.5	0		V^+ − 1.5	0		V^+ − 1.5	0		V^+ − 1.5	0		V^+ − 1.5	V
Supply Current	Over full temperature range V^+ = 5V, V_O = 0V V^+ = 30V (LM2902, V^+ = 26V) V^+ = 5V		1.5 0.7	3 1.2		1.5 0.7	3 1.2		1.5 0.7	3 1.2		1.5 0.7	3 1.2		1.5 0.7	3 1.2		1.5 0.7	3 1.2	mA
Voltage Gain	R_L ≥ 2 kΩ	50	100		50	100		50	100		50	100		50	100		25	100		V/mV
Large Signal Voltage Gain	V^+ = 15V, V_O = 1V to 11V, T_A = 25°C	25			15			15			15			15			10			V/mV
Common-Mode Rejection Ratio	DC, V_{CM} = 0V to V^+ − 1.5V	70	85		70	85		65	85		70	85		65	85		50	75		dB
Power Supply Rejection Ratio	V^+ = 5V to 30V (LM2902, V^+ = 5V to 26V)	65	100		65	100		65	100		65	100		65	100		50	100		dB

Electrical Characteristics V^+ = +5.0V (Note 4) unless otherwise stated (Continued)

Parameter		Conditions	LM124A			LM224A			LM324A			LM124/LM224			LM324			LM2902			Units
			Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Amplifier-to-Amplifier Coupling (Note 8)		f = 1 kHz to 20 kHz, T_A = 25°C (Input Referred)		−120			−120			−120			−120			−120			−120		dB
Output Current	Source	V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = +5V, V_O = 2V, T_A = 25°C	20	40		20	40		20	40		20	40		20	40		20	40		mA
	Sink	V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = +5V, V_O = 2V, T_A = 25°C	10	20		10	20		10	20		10	20		10	20		10	20		mA
		V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = +5V, V_O = 200 mV, T_A = 25°C	12	50		12	50		12	50		12	50		12	50		12	50		µA
Short Circuit to Ground		V^+ = +15V, T_A = 25°C		40	60		40	60		40	60		40	60		40	60		40	60	mA
Input Offset Voltage					4			4			5			7			9			10	mV
Input Offset Voltage Drift				7	20		7	20		7	30		7			7					µV/°C
Input Offset Current		$I_{IN(+)} - I_{IN(-)}$, V_{CM} = 0V			30			30			75			100			150			200	nA
Input Bias Current				10	200		10	200		10	300		10			10			1		µA
Input Common-Mode Voltage Range (Note 9)		V^+ = 30V (LM2902, V^+ = 26V)	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 2$	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 2$	V
Large Signal Voltage Gain		V^+ = 15V, $R_L \geq 2$ kΩ, V_O = 1V to 11V	50			25			15			50			25			15			V/mV
Common-Mode Rejection Ratio		DC, V_{CM} = 0V to $V^+ - 1.5$V	70			70			65			70			65			65			dB
Power Supply Rejection Ratio		V^+ = 5V to 30V (LM2902, V^+ = 5V to 26V)	65			65			65			65			65			65			dB
Amplifier-to-Amplifier Coupling (Note 8)		f = 1 kHz to 20 kHz, T_A = 25°C (Input Referred)		−120			−120			−120			−120			−120			−120		dB
Output Current	Source	V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = +5V, V_O = 2V, T_A = 25°C	20	40		20	40		20	40		20	40		20	40		20	40		mA
	Sink	V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = +5V, V_O = 2V, T_A = 25°C	10	20		10	20		10	20		10	20		10	20		10	20		mA
		V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = +5V, V_O = 200 mV, T_A = 25°C	12	50		12	50		12	50		12	50		12	50		12	50		µA
Short Circuit to Ground		V^+ = +15V, T_A = 25°C		40	60		40	60		40	60		40	60		40	60		40	60	mA
Input Offset Voltage					4			4			5			7			9			10	mV
Input Offset Voltage Drift				7	20		7	20		7	30		7			7					µV/°C
Input Offset Current		$I_{IN(+)} - I_{IN(-)}$, V_{CM} = 0V			30			30			75			100			150			200	nA
Input Bias Current				10	200		10	200		10	300		10			10			1		µA
Input Common-Mode Voltage Range (Note 9)		V^+ = 30V (LM2902, V^+ = 26V)	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 2$	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 2$	V
Large Signal Voltage Gain		V^+ = 15V, $R_L \geq 2$ kΩ, V_O = 1V to 11V	50			25			15			50			25			15			V/mV
Common-Mode Rejection Ratio		DC, V_{CM} = 0V to $V^+ - 1.5$V	70			70			65			70			65			65			dB
Power Supply Rejection Ratio		V^+ = 5V to 30V (LM2902, V^+ = 5V to 26V)	65			65			65			65			65			65			dB
Output Voltage Swing		V^+ = 30V, R_L = 2 kΩ	26			26			22			26			22						V
		V^+ = 30V, R_L = 10 kΩ	27	28		27	28		23	28		27	28		23	28					V
		V^+ = 30V, R_L = 10 kΩ			20			20			20	5	20			20				100	mV

Electrical Characteristics $V^+ = +5.0V$ (Note 4) unless otherwise stated (Continued)

Parameter		Conditions		LM124A			LM224A			LM324A			LM124/LM224			LM324			LM2902			Units
				Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Output Current	Source	$V_O = 2V$	$V_{IN}^- = +1V$, $V_{IN}^+ = 0V$, $V^+ = 15V$	10	20		10	20		10	20		10	20		10	20		10	20		mA
	Sink		$V_{IN}^- = +1V$, $V_{IN}^+ = 0V$, $V^+ = 15V$	10	15		5	8		5	8		5	8		5	8		5	8		

Note 1: For operating at high temperatures the LM324/LM324A/LM2902 must be derated based on a +125°C maximum junction temperature and a thermal resistance of 88°C/W which applies for the device soldered in a printed circuit board, operating in a still air ambient. The LM224/LM224A and LM124/LM124A can be derated based on a +150°C maximum junction temperature. The dissipation is the total of all four amplifiers—use external resistors, where possible, to allow the amplifier to saturate or to reduce the power which is dissipated in the integrated circuit.

Note 2: Short circuits from the output to V^+ can cause excessive heating and eventual destruction. When considering short circuits to ground, the maximum output current is approximately 40 mA independent of the magnitude of V^+. At values of supply voltage in excess of +15V, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.

Note 3: This input current will only exist when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistors becoming forward biased and thereby acting as input diode clamps. In addition to this diode action, there is also lateral NPN parasitic transistor action on the IC chip. This transistor action can cause the output voltages of the op amps to go to the V^+ voltage level (or to ground for a large overdrive) for the time duration that an input is driven negative. This is not destructive and normal output states will re-establish when the input voltage, which was negative, again returns to a value greater than −0.3V (at 25°C).

Note 4: These specifications are limited to −55°C ≤ T_A ≤ +125°C for the LM124/LM124A. With the LM224/LM224A, all temperature specifications are limited to −25°C ≤ T_A ≤ +85°C, the LM324/LM324A temperature specifications are limited to 0°C ≤ T_A ≤ +70°C and the LM2902 specifications are limited to −40°C ≤ T_A ≤ +85°C.

Note 5: Voltage gain, R_L = 2 kΩ with V^+ from 5V to 30V and over the full input common-mode range (0V to V^+ − 1.5V) for LM2902, V^+ from 5V to 26V.

Note 6: The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.

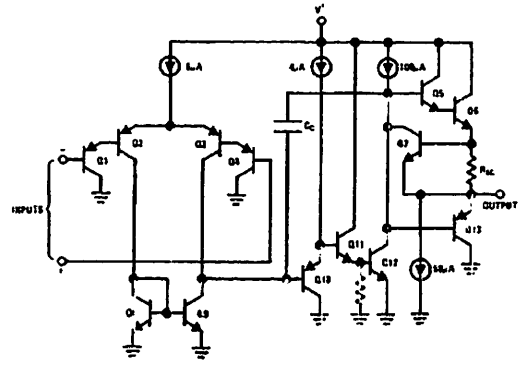
Note 7: The input common-mode voltage or either input signal voltage should not be allowed to go negative by more than 0.3V (at 25°C). The upper end of the common-mode voltage range is V^+ − 1.5V (at 25°C), but either or both inputs can go to +32V without damage (at 25°C), independent of the magnitude of V^+.

Note 8: Due to proximity of external components insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitance increases at higher frequencies.

Note 9: For LM124 military specifications refer to RETS124X for LM124 military specifications.

Note 10: Human body model, 1 kΩ in series with 100 pF.

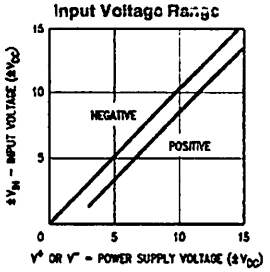
Schematic Diagram (one amplifier)



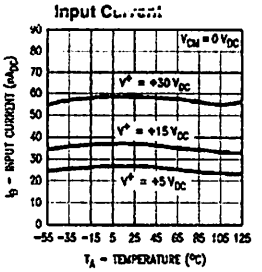
TL/H/9225-2

Typical Performance Characteristics

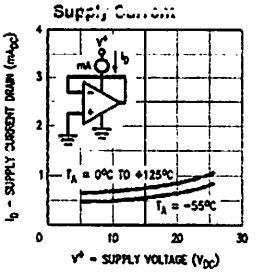
Input Voltage Range



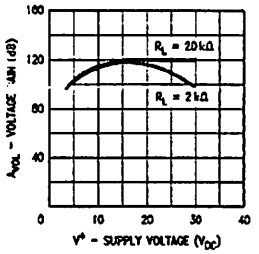
Input Current



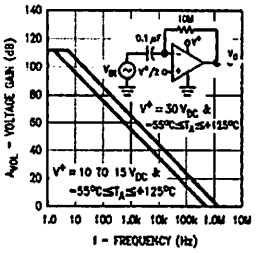
Supply Current



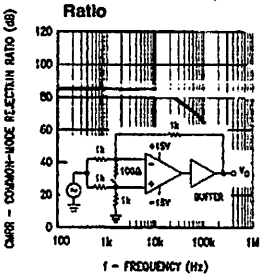
Voltage Gain



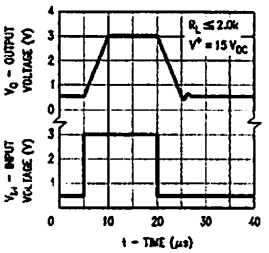
Open Loop Frequency Response



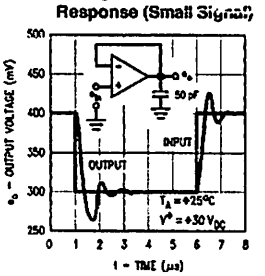
Common Mode Rejection Ratio



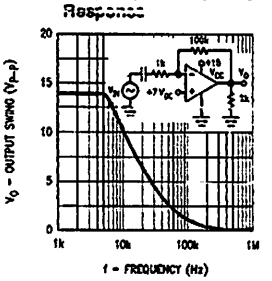
Voltage Follower Pulse Response



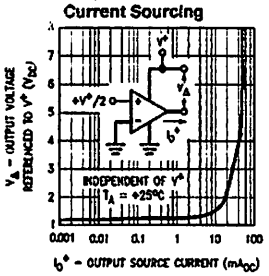
Voltage Follower Pulse Response (Small Signal)



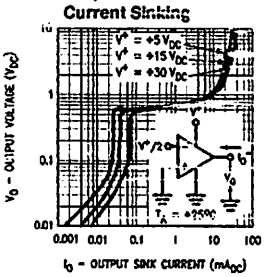
Large Signal Frequency Response



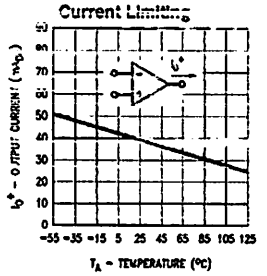
Output Characteristics Current Sourcing



Output Characteristics Current Sinking

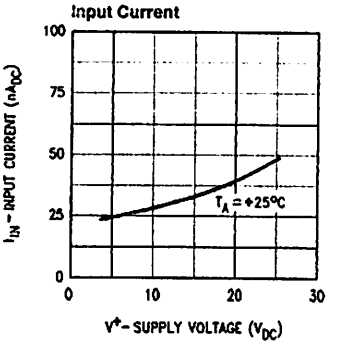


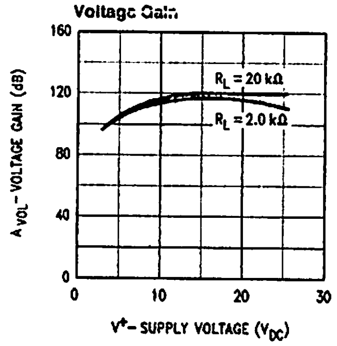
Current Limiting



TL/H/0299–3

Typical Performance Characteristics (Continued)





TL/H/9299–4

Application Hints

The LM124 series are op amps which [illegible] single power supply voltage, have true differential inputs, and remain in the linear mode with an input common mode voltage of 0 V_{DC}. These amplifiers [illegible] range of power supply voltage with little change in performance characteristics. At 25°C amplifier operation is possible down to a minimum supply voltage [illegible]

The pinouts of the package have been designed to simplify PC board layouts. Inverting inputs are adjacent to outputs for all of the amplifiers [illegible] placed at the corners of the package [illegible]

Precautions should be taken to insure that the power supply for the integrated circuit [illegible] or that the unit is not inadvertently installed [illegible] test socket as an unlimited current surge [illegible] ing forward diode within the IC could cause fusing of the internal conductors and result in a destroyed unit.

Large differential input voltages can be [illegible] dated and, as input differential voltage protection diodes are not needed, no large input currents result from large differential input voltages. The [illegible] larger than V+ without damaging the device. [illegible] should be provided to prevent the input [illegible] negative mo[illegible] diode with a resistor to the IC input [illegible]

To reduce the power supply drain, the amplifiers have a class A output [illegible] to class B in a large signal mode. [illegible] to both source and sink large [illegible] both NPN and PNP external current boost [illegible] be used to extend the power capability of the basic amplifiers. The output voltage [illegible] ode drop above ground to bias the [illegible] transistor for output current sinking applications.

For ac applications, where [illegible] to the output of the amplifier, a resistor [illegible] the output of the amplifier to ground to increase the class A bias current and prevent crossover distortion.

[illegible] these is an [illegible]

Capacitive loads which are applied directly to the output of [illegible] 50 pF can be accommodated using the worst-case non-inverting unity gain connection. Large closed loop gains or [illegible] must be [illegible]

The bias network of the LM124 establishes a drain current [illegible]

Output short circuits either to ground or to the positive power [illegible]

[illegible]

sive [illegible]

[illegible]

The [illegible]

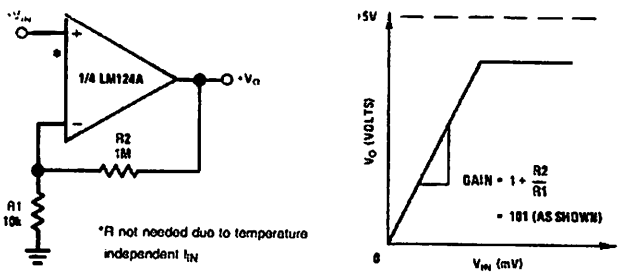
[illegible]

range [illegible]

[illegible]

Typical Single-Supply Applications (5.0 V_{DC})

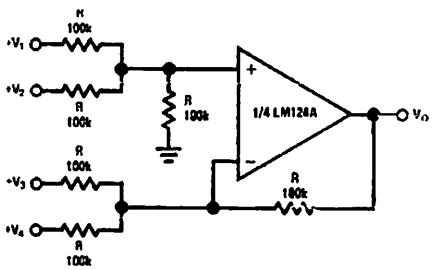
Non-Inverting DC Gain (0 Input = 0 Output)



TL/H/9299-5

DC Summing Amplifier

(V_{IN}'S ≥ 0 V_{DC} and V_O ≥ V_{DC})

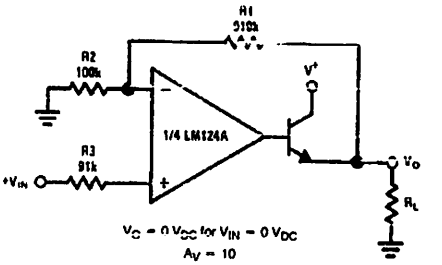


TL/H/9299-6

Where: $V_O = V_1 + V_2 - V_3 - V_4$

$(V_1 + V_2) \geq (V_3 + V_4)$ to keep $V_O > 0$ V_{DC}

Power Amplifier

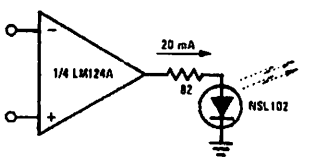


$V_O = 0$ V_{DC} for $V_{IN} = 0$ V_{DC}

$A_V = 10$

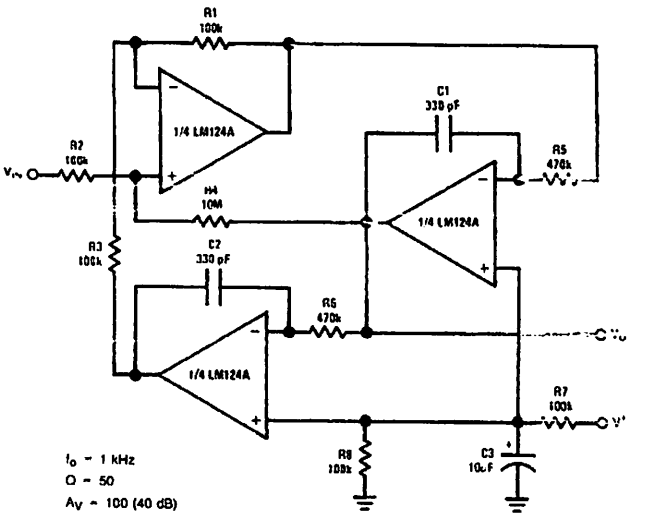
TL/H/9299-7

LED Driver



TL/H/9299-8

"BI-QUAD" RC Active Bandpass Filter



f_o = 1 kHz

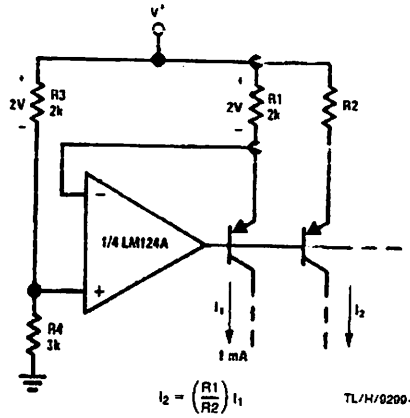
Q = 50

A_V = 100 (40 dB)

TL/H/9299-9

Typical Single-Supply Applications ($5.0\ V_{DC}$ Operation)

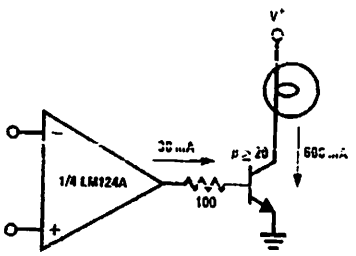
Fixed Current Sources



$$I_2 = \left(\frac{R1}{R2}\right) I_1$$

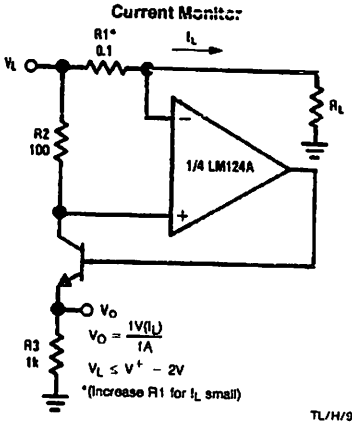
TL/H/9299–10

Lamp Driver



TL/H/9299–11

Current Monitor



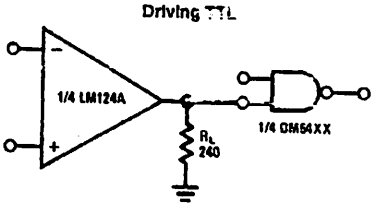
$$V_O = \frac{1V(I_L)}{1A}$$

$V_L \leq V^+ - 2V$

*(Increase R1 for I_L small)

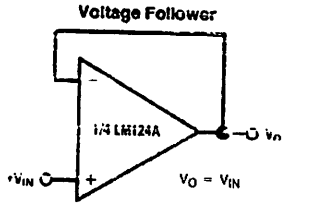
TL/H/9299–12

Driving TTL



TL/H/9299–13

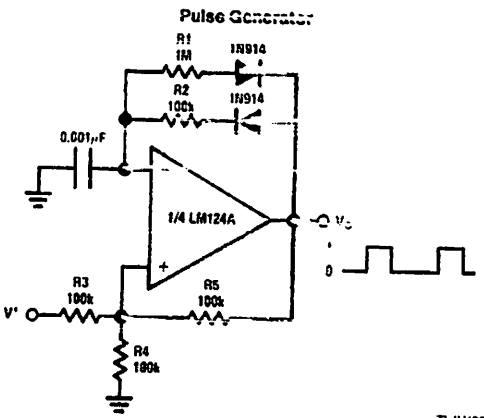
Voltage Follower



$V_O = V_{IN}$

TL/H/9299–14

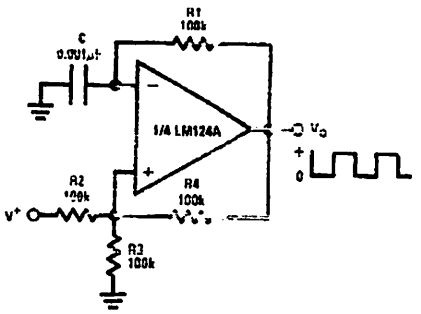
Pulse Generator



TL/H/9299–15

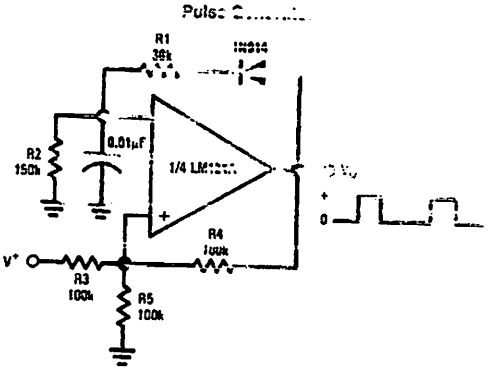
Typical Single-Supply Applications (5.0 V_{DC} supplied)

Square-wave Oscillator



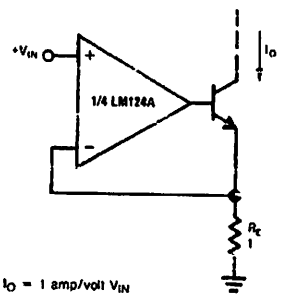
TL/H/9299–16

Pulse Generator



TL/H/9299–17

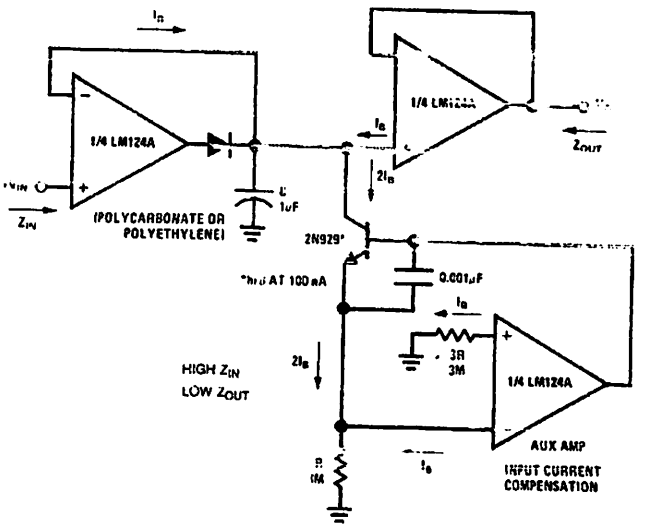
High Compliance Current Sink



I_O = 1 amp/volt V_{IN}
(Increase R_E for I_O small)

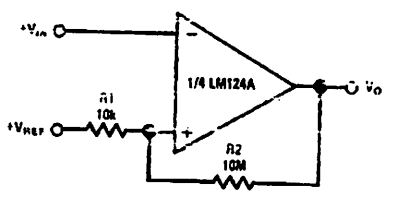
TL/H/9299–18

Low Drift Peak Detector



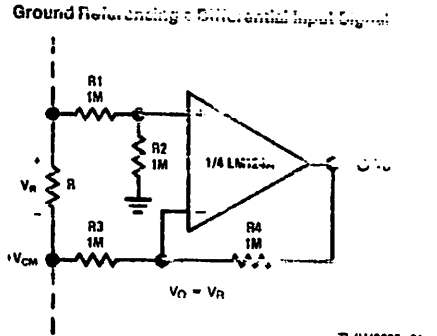
TL/H/9299–19

Comparator with Hysteresis



TL/H/9299–20

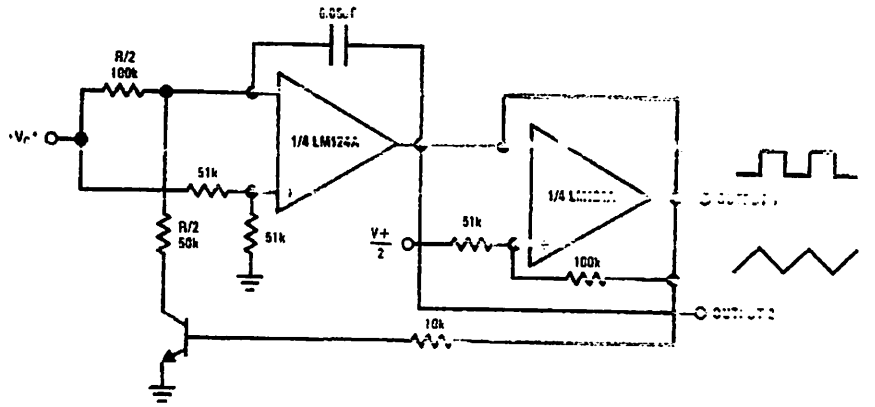
Ground Referencing a Differential Input Signal



TL/H/9299–21

Typical Single-Supply Applications (5.0 V_{DC})

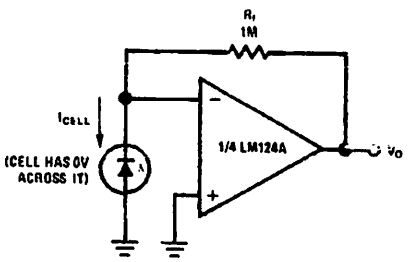
Voltage Controlled Oscillator



*Wide control voltage range: 0 $V_{DC} \leq V_C \leq 2\ (V^+ - 1.5\ V_{DC})$

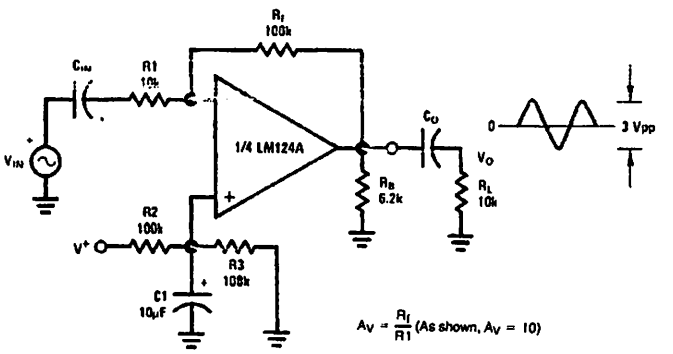
TL/H/9299–22

Photo Voltaic-Cell Amplifier



TL/H/9299–23

AC Coupled Inverting Amplifier

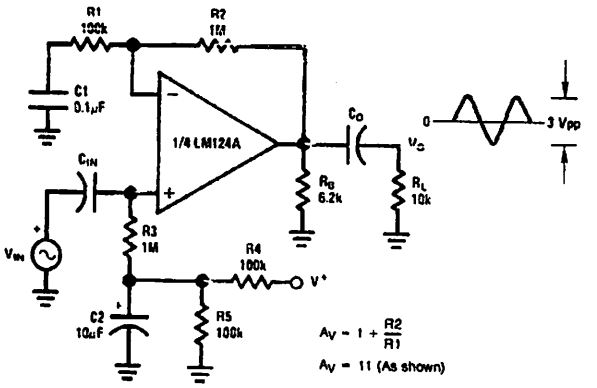


$A_V = \frac{R_f}{R1}$ (As shown, $A_V = 10$)

TL/H/9299–24

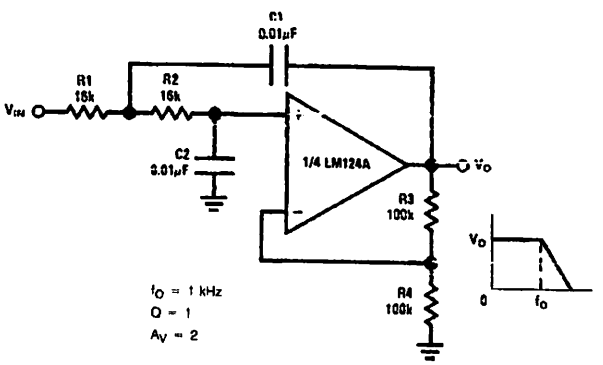
Typical Single-Supply Applications (V+ = 5.0 VDC) (Continued)

AC Coupled Non-Inverting Amplifier



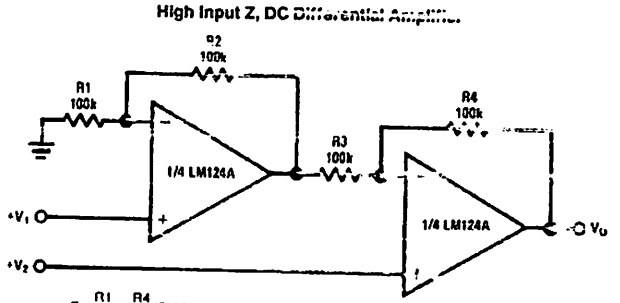
TL/H/9299–25

DC Coupled Low-Pass RC Active Filter



TL/H/9299–26

High Input Z, DC Differential Amplifier



For $\frac{R1}{R2} = \frac{R4}{R3}$ (CMRR depends on this resistor ratio match)

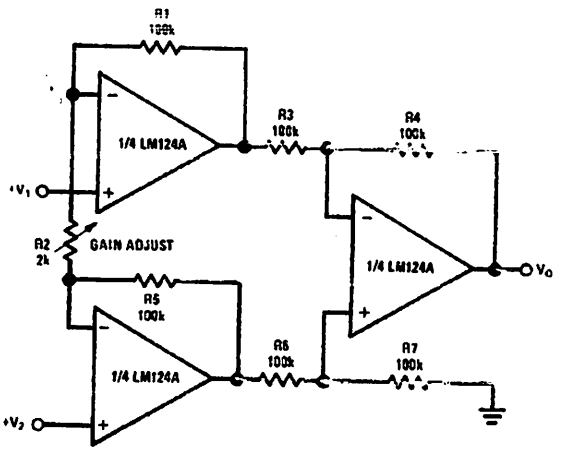
TL/H/9299–27

$V_O = 1 + \frac{R4}{R3}(V_2 - V_1)$

As shown: $V_O = 2(V_2 - V_1)$

Typical Single Supply Applications (5.0 VDC Supply Guaranteed)

High Input Z Adjustable Gain DC Instrumentation Amplifier

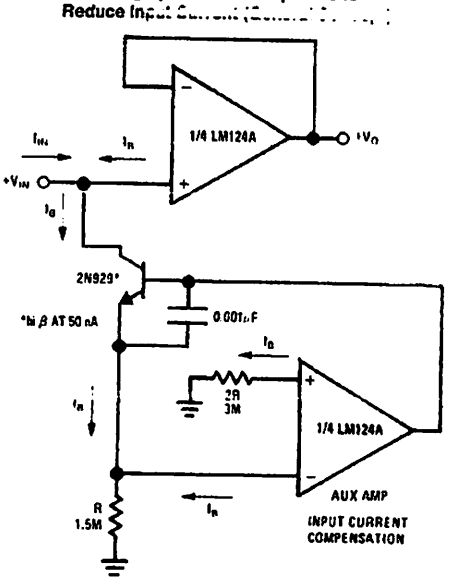


If R1 = R5 & R3 = R4 = R6 = R7 (CMRR depends on match) TL/H/9299–28

$$V_O = 1 + \frac{2R1}{R2}(V_2 - V_1)$$

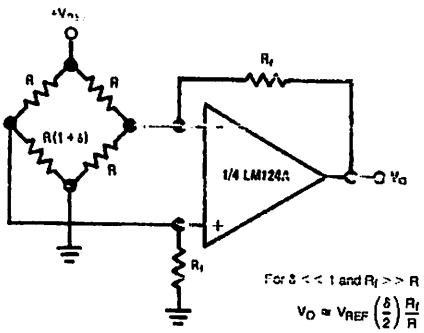
As shown $V_O = 101\,(V_2 - V_1)$

Using Symmetrical Amplifiers to Reduce Input Current (General Concept Only)



TL/H/9299–29

Bridge Current Amplifier



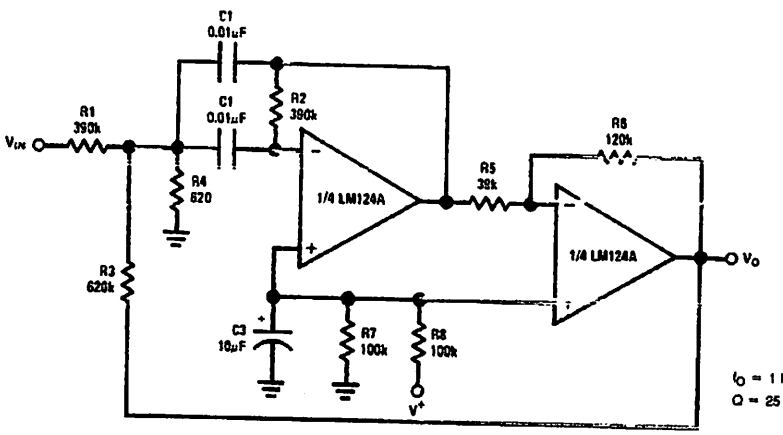
For $\delta << 1$ and $R_f >> R$

$$V_O \approx V_{REF}\left(\frac{\delta}{2}\right)\frac{R_f}{R}$$

TL/H/9299–30

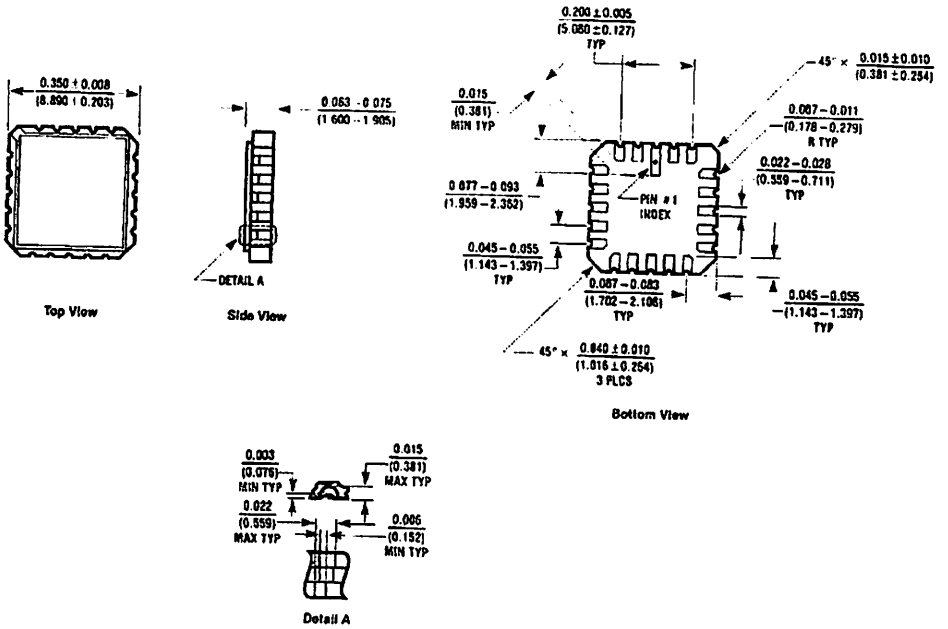
Typical Single Supply Applications (V+ = 5.0 VDC) (Continued)

Bandpass Active Filter

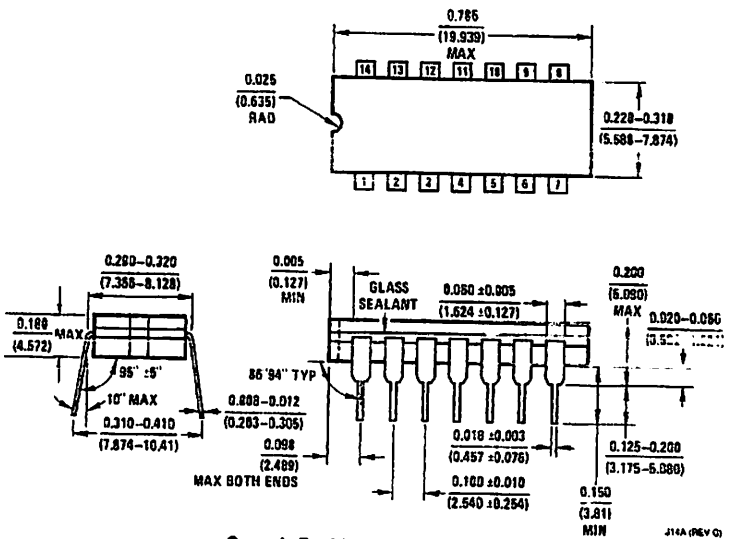


TL/H/0200–31

Physical Dimensions

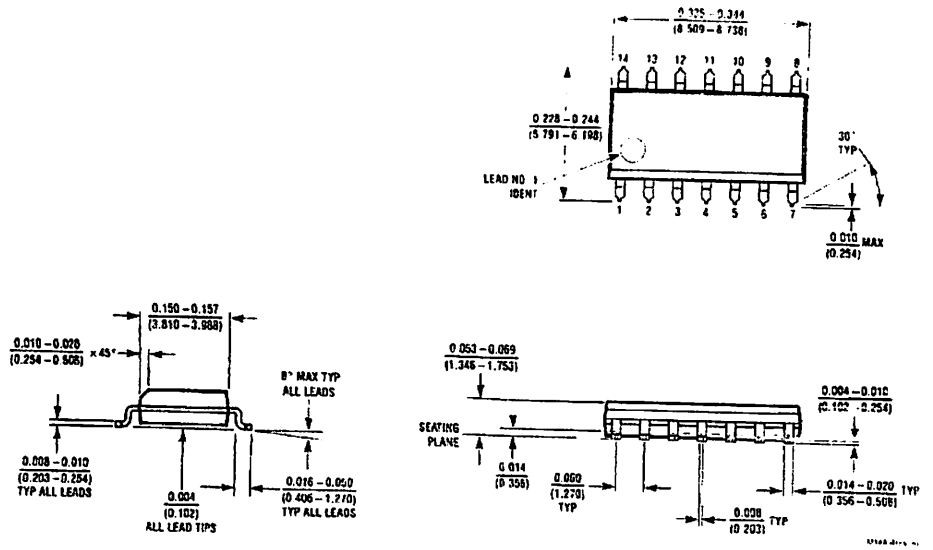


Leadless Chip Carrier Package
Order Number LM124AE/883 or LM124E/883
NS Package Number E20A

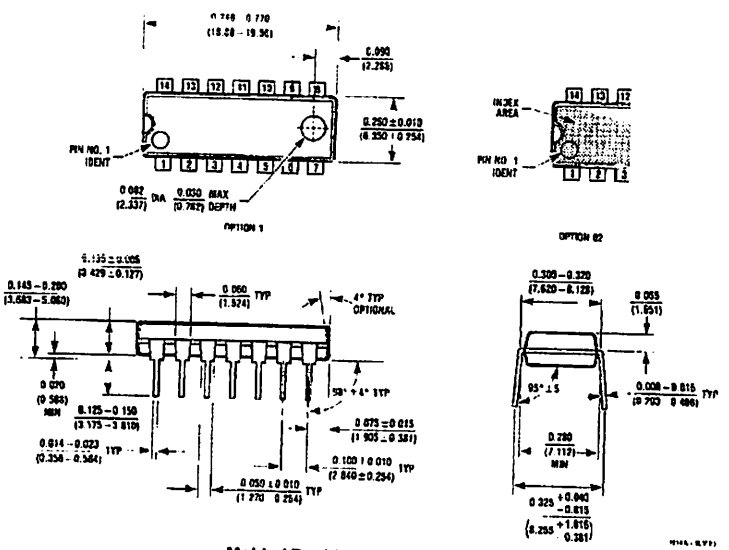


Ceramic Dual-In-Line Package (J)
Order Number LM124J, LM124AJ, LM124AJ/883, LM124J/883, LM124J, LM124AJ ... LM124J
NS Package Number J14A

Physical Dimensions

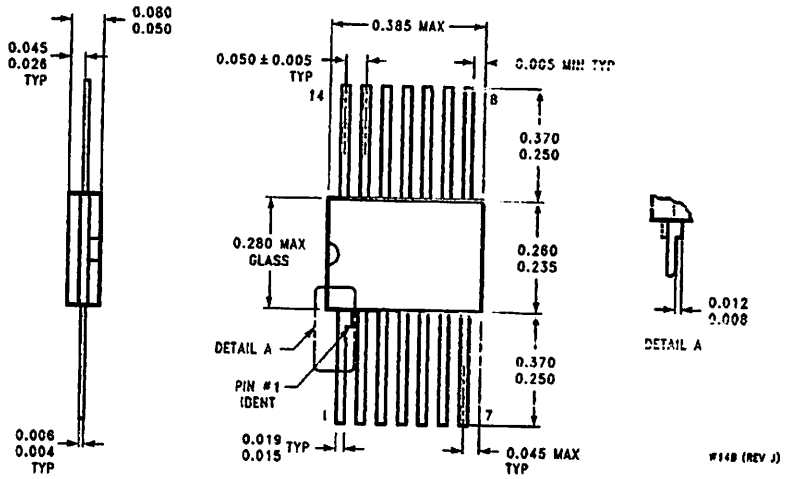


S.O. Package (M)
Order Number LM324M, LM324AM or LM2902M
NS Package Number M14A



Molded Dual-In-Line Package (N)
Order Number LM324N, LM324AN or LM2902N
NS Package Number N14A

Physical Dimensions



Ceramic Flatpak Package
Order Number LM124AW/883 or LM124W/883
NS Package Number W14B

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-18[illegible]
Email: cnwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180[illegible]
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
[illegible]
Tel: (852) 2737-1600
Fax: (852) [illegible]

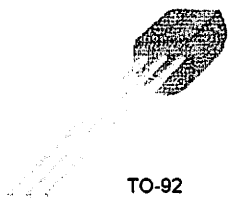
National Semiconductor Japan Ltd.
[illegible]
Tel: 81-043-299-2309

FAIRCHILD
SEMICONDUCTOR®

SS9013

1W Output Amplifier of Potable Radios in Class B Push-pull Operation.

- High total power dissipation. (P_T=625mW)
- High Collector Current. (I_C=500mA)
- Complementary to SS9012
- Excellent h_{FE} linearity.



TO-92

1. Emitter 2. Base 3. Collector

NPN Epitaxial Silicon Transistor

Absolute Maximum Ratings T_a=25°C unless otherwise noted

Symbol	Parameter	Ratings	Units
V_{CBO}	Collector-Base Voltage	40	V
V_{CEO}	Collector-Emitter Voltage	20	V
V_{EBO}	Emitter-Base Voltage	5	V
I_C	Collector Current	500	A
P_C	Collector Power Dissipation	625	W
T_J	Junction Temperature	150	°C
T_{STG}	Storage Temperature	-55 ~ 150	°C

Electrical Characteristics T_a=25°C unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units
BV_{CBO}	Collector-Base Breakdown Voltage	I_C =100μA, I_E =0	40			V
BV_{CEO}	Collector-Emitter Breakdown Voltage	I_C =1mA, I_B =0	20			V
BV_{EBO}	Emitter-Base Breakdown Voltage	I_E =100μA, I_C =0	5			V
I_{CBO}	Collector Cut-off Current	V_{CB} =25V, I_E =0			100	nA
I_{EBO}	Emitter Cut-off Current	V_{EB} =3V, I_C =0			100	nA
h_{FE1}	DC Current Gain	V_{CE} =1V, I_C =50mA	64	120	202	
h_{FE2}		V_{CE} =1V, I_C =500mA	40	120		
V_{CE} (sat)	Collector-Emitter Saturation Voltage	I_C =500mA, I_B =50mA		0.16	0.6	V
V_{BE} (sat)	Base-Emitter Saturation Voltage	I_C =500mA, I_B =50mA		0.91	1.2	V
V_{BE} (on)	Base-Emitter On Voltage	V_{CE} =1V, I_C =10mA	0.6	0.67	0.7	V

h_{FE} Classification

Classification	D	E	F	G	H
h_{FE1}	64 ~ 91	78 ~ 112	96 ~ 135	112 ~ 166	144 ~ 202

Typical Characteristics

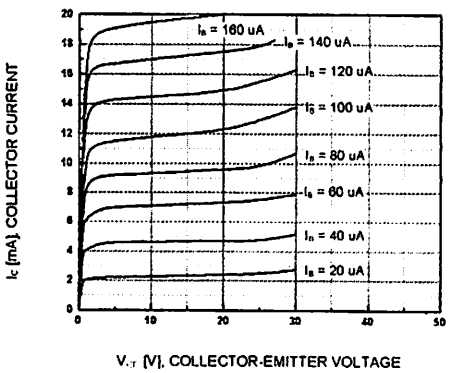


Figure 1. Static Characteristic

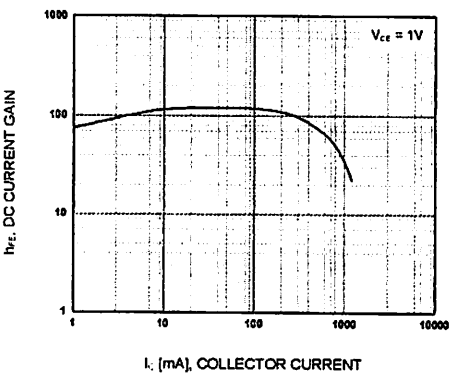


Figure 2. DC current Gain

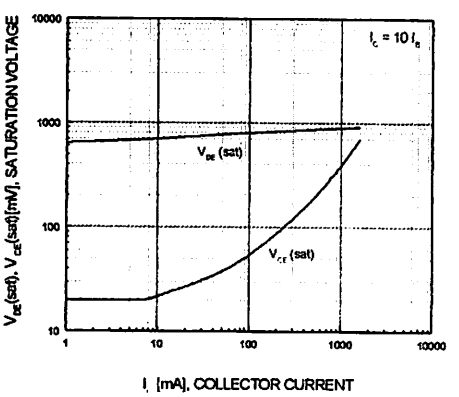


Figure 3. Base-Emitter Saturation Voltage Collector-Emitter Saturation Voltage

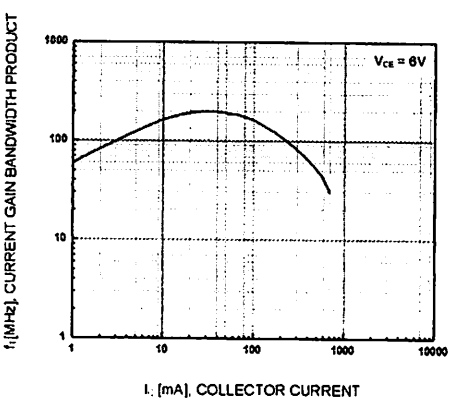
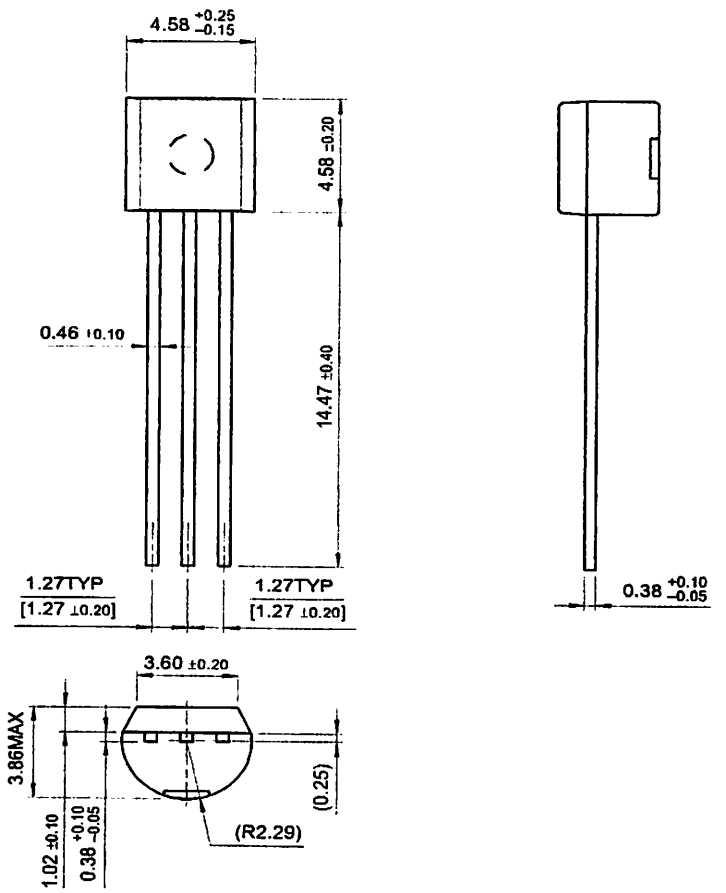


Figure 4. Current Gain Bandwidth Product

Package Demensions

TO-92



Dimensions in Millimeters

TRADEMARKS

The following are registered and unregistered trademarks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.

ACEx™	FAST®	OPTOPLANAR™	STAR*POWER™
Bottomless™	FASTr™	PACMAN™	Stealth™
CoolFET™	FRFET™	POP™	SuperSOT™-3
CROSSVOLT™	GlobalOptoisolator™	Power247™	SuperSOT™-6
DenseTrench™	GTO™	PowerTrench®	SuperSOT™-8
DOME™	HiSeC™	QFET™	SyncFET™
EcoSPARK™	ISOPLANAR™	QS™	TruTranslation™
E^2CMOS™	LittleFET™	QT Optoelectronics™	TinyLogic™
EnSigna™	MicroFET™	Quiet Series™	UHC™
FACT™	MICROWIRE™	SLIENT SWITCHER®	UltraFET®
FACT Quiet Series™	OPTOLOGIC™	SMART START™	VCX™

STAR*POWER is used under license

DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, or (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

PRODUCT STATUS DEFINITIONS

Definition of Terms

Datasheet Identification	Product Status	Definition
Advance Information	Formative or In Design	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	This datasheet contains preliminary data, and supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
No Identification Needed	Full Production	This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
Obsolete	Not In Production	This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only.

1 GHz, 5,500 V/µs Low Distortion Amplifier

AD8009

FEATURES

Ultrahigh Speed
- 5,500 V/µs Slew Rate, 4 V Step, G = +2
- 545 ps Rise Time, 2 V Step, G = +2
- Large Signal Bandwidth
  - 440 MHz, G = +2
  - 320 MHz, G = +10
- Small Signal Bandwidth (–3 dB)
  - 1 GHz, G = +1
  - 700 MHz, G = +2
- Settling Time 10 ns to 0.1%, 2 V Step, G = +2

Low Distortion Over Wide Bandwidth
- SFDR
  - –44 dBc @ 150 MHz, G = +2, V_O = 2 V p-p
  - –41 dBc @ 150 MHz, G = +10, V_O = 2 V p-p
- 3rd Order Intercept (3IP)
  - 26 dBm @ 70 MHz, G = +10
  - 18 dBm @ 150 MHz, G = +10

Good Video Specifications
- Gain Flatness 0.1 dB to 75 MHz
- 0.01% Differential Gain Error, R_L = 150 Ω
- 0.01° Differential Phase Error, R_L = 150 Ω

High Output Drive
- 175 mA Output Load Drive
- 10 dBm with –38 dBc SFDR @ 70 MHz, G = +10

Supply Operation
- +5 V to ±5 V Voltage Supply
- 14 mA (Typ) Supply Current

APPLICATIONS

Pulse Amplifier
IF/RF Gain Stage/Amplifiers
High Resolution Video Graphics
High Speed Instrumentations
CCD Imaging Amplifier

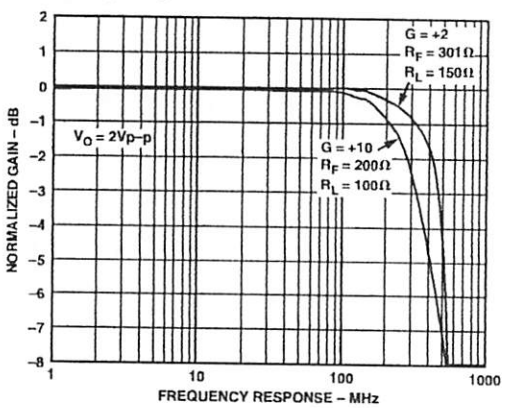
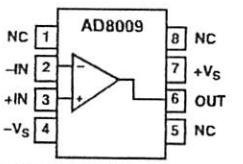


Figure 1. Large Signal Frequency Response; G = +2 and +10

REV. D

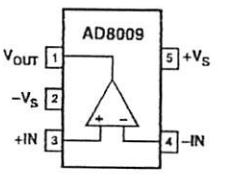
FUNCTIONAL BLOCK DIAGRAMS

8-Lead Plastic SOIC (SO-8)



NC = NO CONNECT

5-Lead SOT-23 (RT-5)



PRODUCT DESCRIPTION

The AD8009 is an ultrahigh speed current feedback amplifier with a phenomenal 5,500 V/µs slew rate that results in a rise time of 545 ps, making it ideal as a pulse amplifier.

The high slew rate reduces the effect of slew rate limiting and results in the large signal bandwidth of 440 MHz required for high resolution video graphic systems. Signal quality is maintained over a wide bandwidth with worst case distortion of –40 dBc @ 250 MHz (G = +10, 1 V p-p). For applications with multitone signals such as IF signal chains, the third order Intercept (3IP) of 12 dBm is achieved at the same frequency. This distortion performance coupled with the current feedback architecture make the AD8009 a flexible component for a gain stage amplifier in IF/RF signal chains.

The AD8009 is capable of delivering over 175 mA of load current and will drive four back terminated video loads while maintaining low differential gain and phase error of 0.02% and 0.04° respectively. The high drive capability is also reflected in the ability to deliver 10 dBm of output power @ 70 MHz with –38 dBc SFDR.

The AD8009 is available in a small SOIC package and will operate over the industrial temperature range –40°C to +85°C. The AD8009 is also available in an SOT-23-5 and will operate over the commercial temperature range 0°C to 70°C.

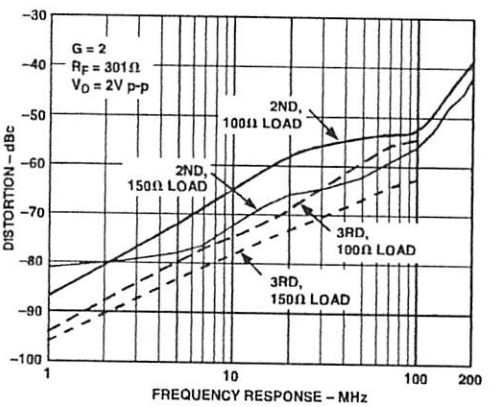


Figure 2. Distortion vs. Frequency; G = +2

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781/329-4700 World Wide Web Site: http://www.analog.com
Fax: 781/326-8703

AD8009—SPECIFICATIONS (@ T_A = 25°C, V_S = ±5 V, R_L = 100 Ω, for R Package: R_F = 301 Ω for G = +1, +2, R_F = 200 Ω for G = +10, for RT Package: R_F = 332 Ω for G = +1, R_F = 226 Ω for G = +2 and R_F = 191 for G = +10, unless otherwise noted.)

Model	Conditions	AD8009AR/JRT Min	Typ	Max	Unit
DYNAMIC PERFORMANCE					
-3 dB Small Signal Bandwidth, V_O = 0.2 V p-p					
R Package	G = +1, R_F = 301 Ω		1000		MHz
RT Package	G = +1, R_F = 332 Ω		845		MHz
	G = +2	480	700		MHz
	G = +10	300	350		MHz
Large Signal Bandwidth, V_O = 2 V p-p	G = +2	390	440		MHz
	G = +10	235	320		MHz
Gain Flatness 0.1 dB, V_O = 0.2 V p-p	G = +2, R_L = 150 Ω	45	75		MHz
Slew Rate	G = +2, R_L = 150 Ω, 4 V Step	4500	5500		V/µs
Settling Time to 0.1%	G = +2, R_L = 150 Ω, 2 V Step		10		ns
	G = +10, 2 V Step		25		ns
Rise and Fall Time	G = +2, R_L = 150 Ω, 4 V Step		0.725		ns
HARMONIC/NOISE PERFORMANCE					
SFDR G = +2, V_O = 2 V p-p	5 MHz		-74		dBc
	70 MHz		-53		dBc
	150 MHz		-44		dBc
G = +10, V_O = 2 V p-p	5 MHz		-58		dBc
	70 MHz		-41		dBc
	150 MHz		-41		dBc
Third Order Intercept (3IP)	70 MHz		26		dBm
W.R.T. Output, G = +10	150 MHz		18		dBm
	250 MHz		12		dBm
Input Voltage Noise	f = 10 MHz		1.9		nV/√Hz
Input Current Noise	f = 10 MHz, +In		46		pA/√Hz
	-In		41		pA/√Hz
Differential Gain Error	NTSC, G = +2, R_L = 150 Ω		0.01	0.03	%
	R_L = 37.5 Ω		0.02	0.05	%
Differential Phase Error	NTSC, G = +2, R_L = 150 Ω		0.01	0.03	Degrees
	R_L = 37.5 Ω		0.04	0.08	Degrees
DC PERFORMANCE					
Input Offset Voltage			2	5	mV
	T_{MIN}–T_{MAX}			7	mV
Offset Voltage Drift			4		µV/°C
-Input Bias Current			50	150	±µA
	T_{MIN}–T_{MAX}		75		±µA
+Input Bias Voltage			50	150	±µA
	T_{MIN}–T_{MAX}		75		±µA
Open Loop Transresistance		90	250		kΩ
	T_{MIN}–T_{MAX}		170		kΩ
INPUT CHARACTERISTICS					
Input Resistance	+Input		110		kΩ
	-Input		8		Ω
Input Capacitance	+Input		2.6		pF
Input Common-Mode Voltage Range			3.8		±V
Common-Mode Rejection Ratio	V_{CM} = ±2.5	50	52		dB
OUTPUT CHARACTERISTICS					
Output Voltage Swing		±3.7	±3.8		V
Output Current	R_L = 10 Ω, P_D Package = 0.7 W	150	175		mA
Short Circuit Current			330		mA
POWER SUPPLY					
Operating Range		+5		±6	V
Quiescent Current			14	16	mA
	T_{MIN}–T_{MAX}			18	mA
Power Supply Rejection Ratio	V_S = ±4 V to ±6 V	64	70		dB

Specifications subject to change without notice.

ABSOLUTE MAXIMUM RATINGS[1]

Supply Voltage ..... 12.6 V
Internal Power Dissipation[2]
Small Outline Package (R) ..... 0.75 Watts
Input Voltage (Common Mode) ..... $\pm V_S$
Differential Input Voltage ..... ±3.5 V
Output Short Circuit Duration
..... Observe Power Derating Curves
Storage Temperature Range R Package .... –65°C to +125°C
Operating Temperature Range (A Grade) ... –40°C to +85°C
Operating Temperature Range (J Grade) ..... 0°C to 70°C
Lead Temperature Range (Soldering 10 sec) ..... 300°C

NOTES
[1]Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.
[2]Specification is for device in free air:
8-Lead SOIC Package: θ_{JA} = 155°C/W.
5-Lead SOT-23 Package: θ_{JA} = 240°C/W.

MAXIMUM POWER DISSIPATION

The maximum power that can be safely dissipated by the AD8009 is limited by the associated rise in junction temperature. The maximum safe junction temperature for plastic encapsulated devices is determined by the glass transition temperature of the plastic, approximately 150°C. Exceeding this limit temporarily may cause a shift in parametric performance due to a change in the stresses exerted on the die by the package. Exceeding a junction temperature of 175°C for an extended period can result in device failure.

While the AD8009 is internally short circuit protected, this may not be sufficient to guarantee that the maximum junction temperature (150°C) is not exceeded under all conditions. To ensure proper operation, it is necessary to observe the maximum power derating curves.

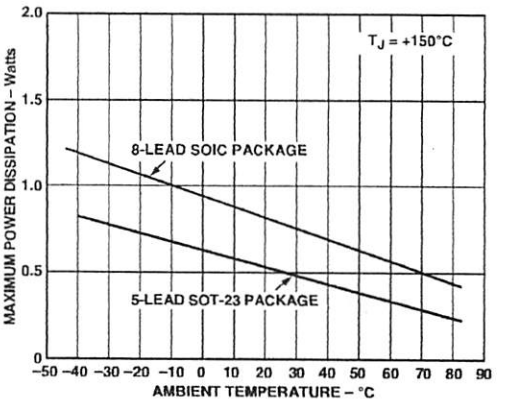


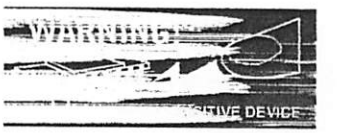
Figure 3. Plot of Maximum Power Dissipation vs. Temperature

ORDERING GUIDE

Model	Temperature Range	Package Description	Package Option	Branding Information
AD8009ACHIPS	–40°C to +85°C	Die		
AD8009AR	–40°C to +85°C	8-Lead SOIC	SO-8	
AD8009AR-REEL	–40°C to +85°C	8-Lead SOIC	13" Tape and Reel	
AD8009AR-REEL7	–40°C to +85°C	8-Lead SOIC	7" Tape and Reel	
AD8009JRT-REEL	0°C to 70°C	5-Lead SOT-23	13" Tape and Reel	HKJ
AD8009JRT-REEL7	0°C to 70°C	5-Lead SOT-23	7" Tape and Reel	HKJ
AD8009-EB		Evaluation Board	SO-8	

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the AD8009 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high-energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



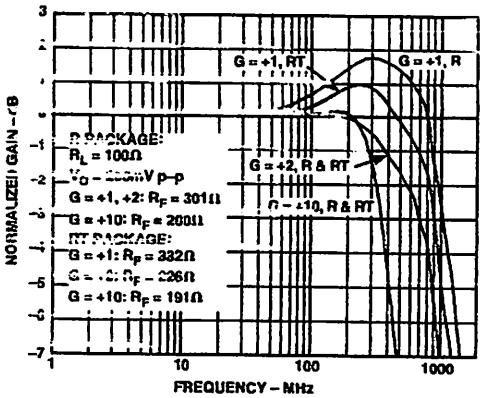


Figure 4. Frequency Response; G = +1, +2, +10, R and RT Packages

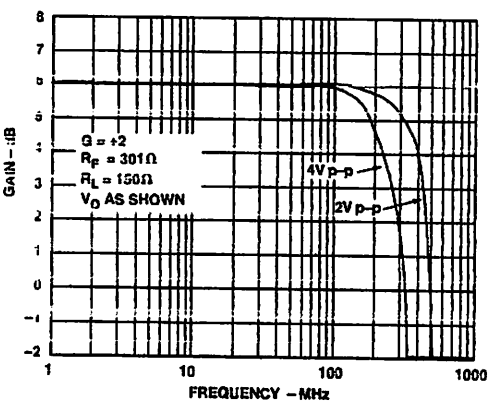


Figure 5. Large Signal Frequency Response; G = +2

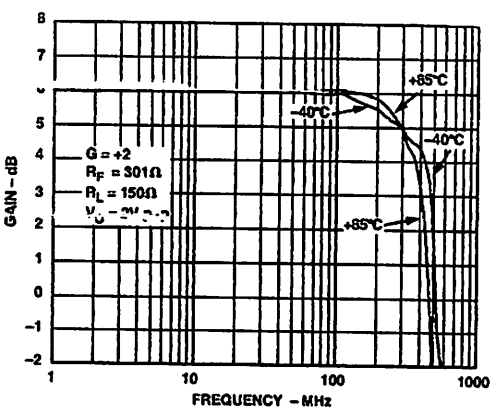


Figure 6. Large Signal Frequency Response vs. Temperature; G = +2

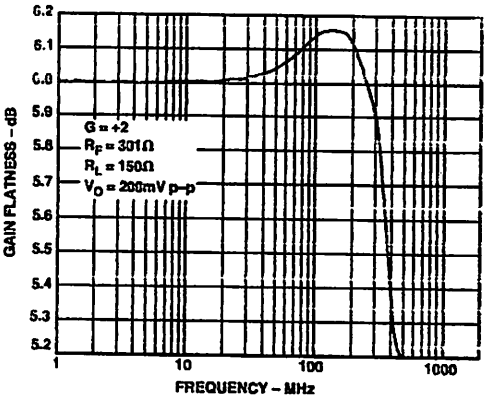


Figure 7. Gain Flatness; G = +2

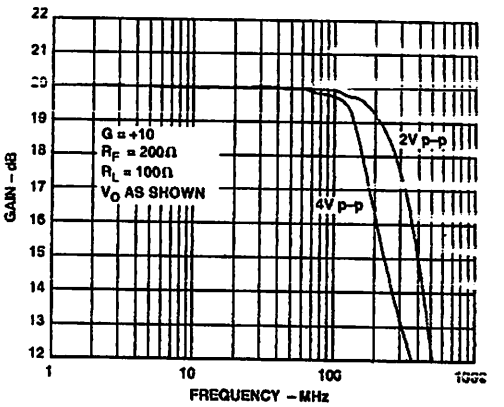


Figure 8. Large Signal Frequency Response, G = +10

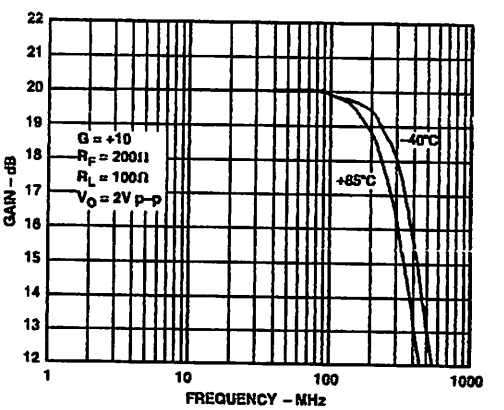


Figure 9. Large Signal Frequency Response vs. Temperature; G = +10

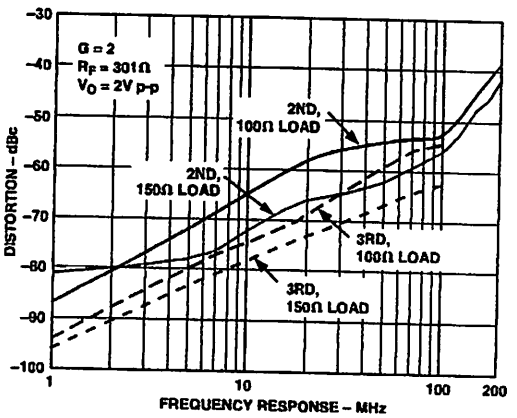


Figure 10. Distortion vs. Frequency; G = +2

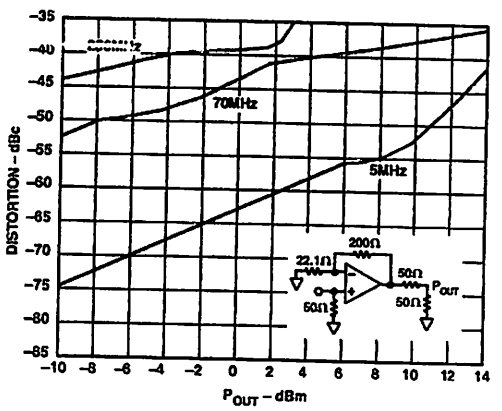


Figure 11. 2nd Harmonic Distortion vs. P_{OUT}; (G = +10)

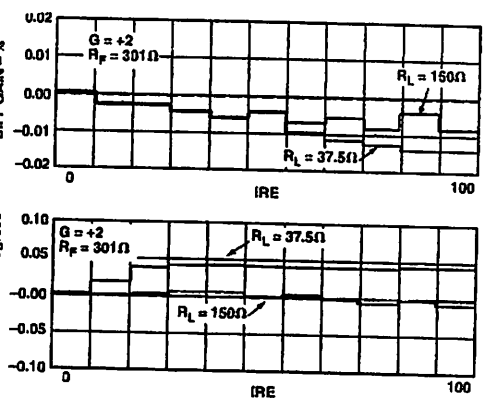


Figure 12. Differential Gain and Phase

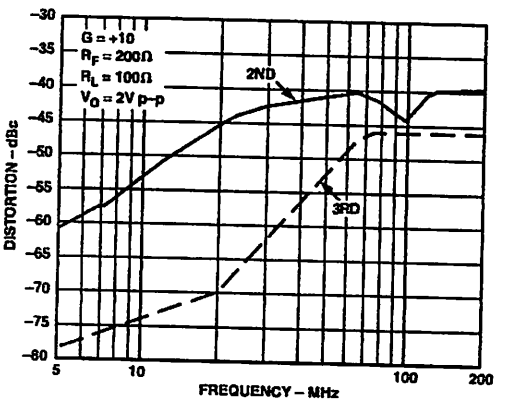


Figure 13. Distortion vs. Frequency; G = +10

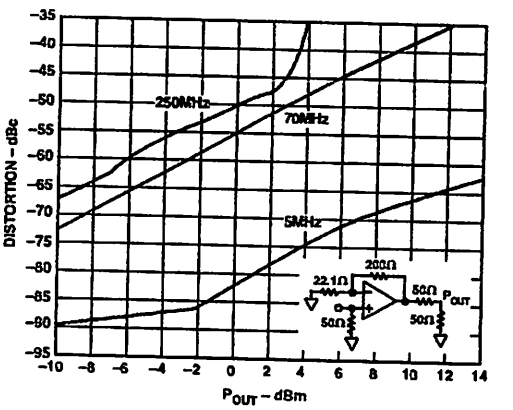


Figure 14. 3rd Harmonic Distortion vs. P_{OUT}; (G = +10)

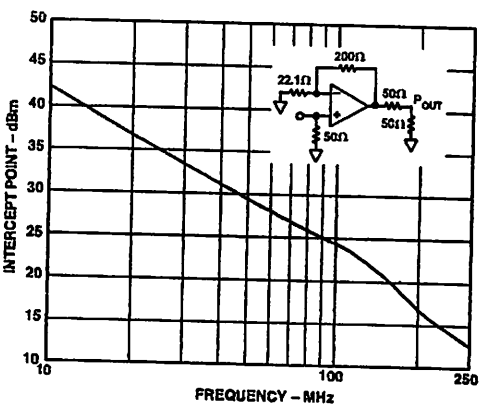


Figure 15. Two Tone, 3rd Order IMD Intercept vs. Frequency; G = +10

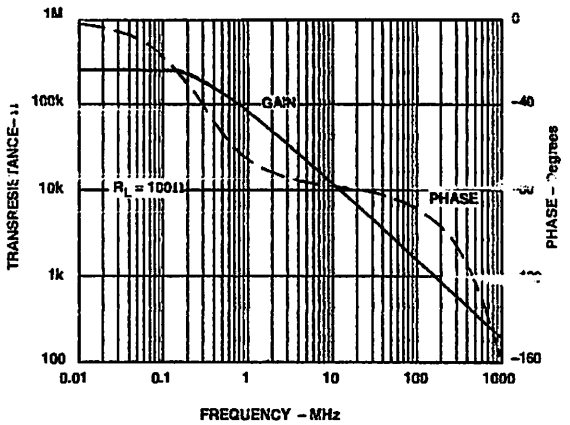


Figure 16. Transresistance and Phase vs. Frequency

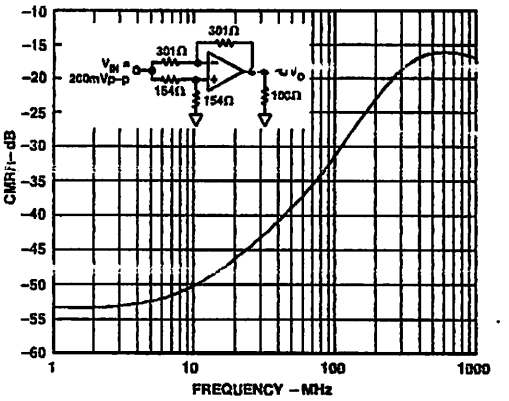


Figure 19. CMRR vs. Frequency

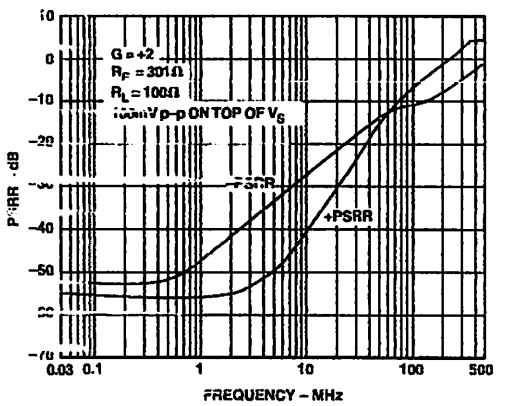


Figure 17. PSRR vs. Frequency

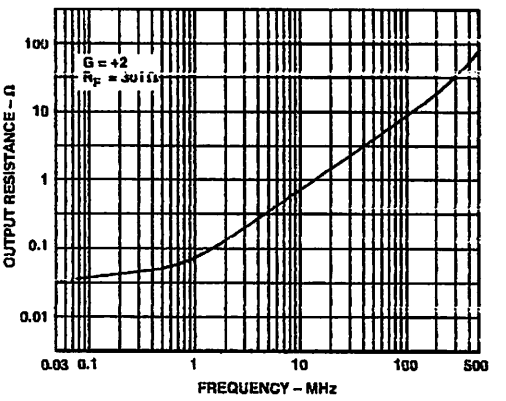


Figure 20. Output Resistance vs. Frequency

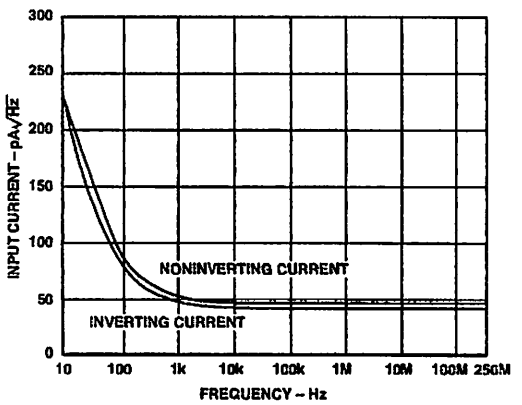


Figure 18. Current Noise vs. Frequency

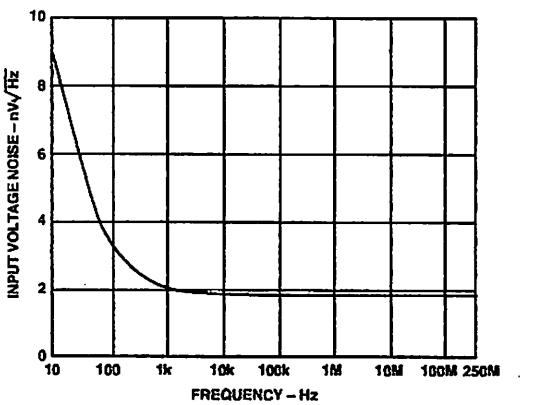


Figure 21. Voltage Noise vs. Frequency

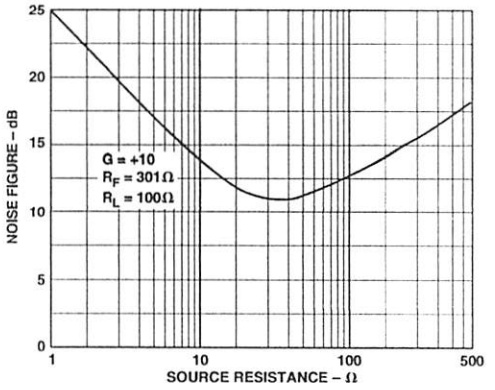


Figure 22. Noise Figure

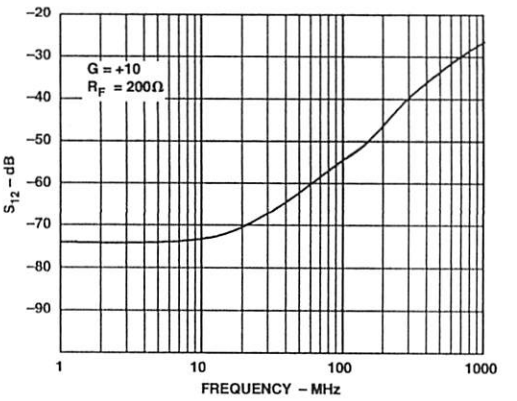


Figure 25. Reverse Isolation (S_{12}); G = +10

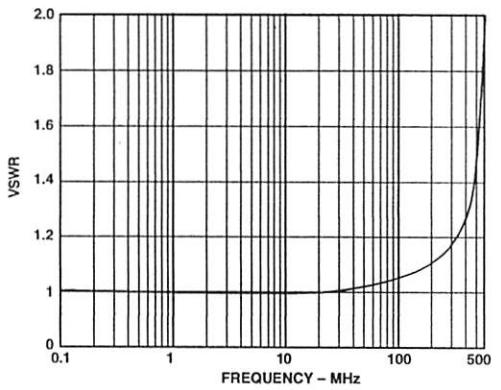


Figure 23. Input VSWR; G = +10

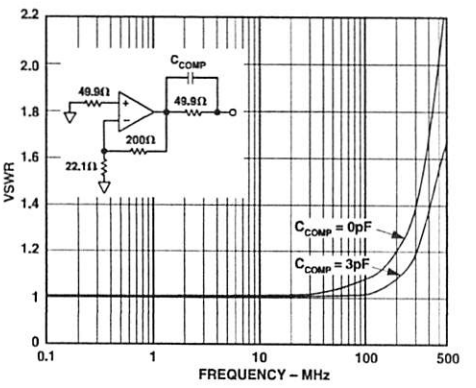


Figure 26. Output VSWR; G = +10

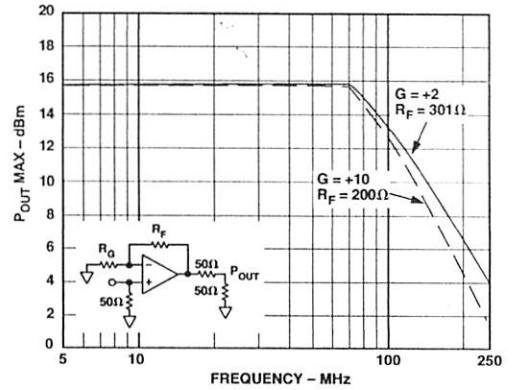


Figure 24. Maximum Output Power vs. Frequency

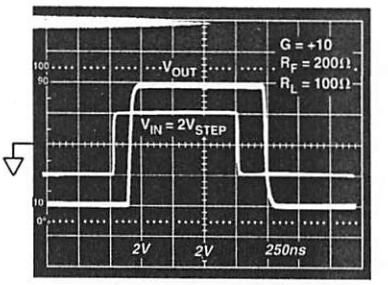


Figure 27. Overdrive Recovery; G = +10

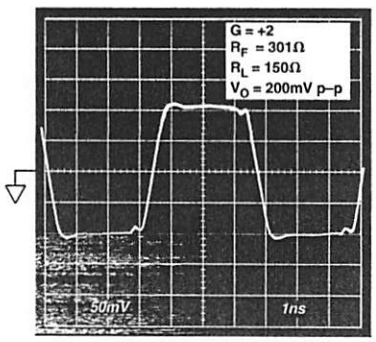


Figure 28. Small Signal Transient Response; G = +2

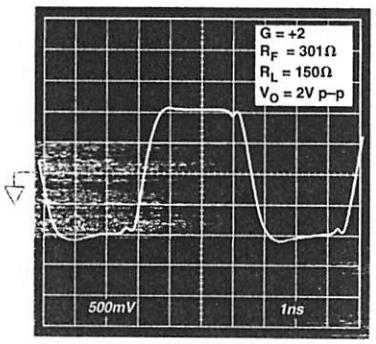


Figure 29. 2 V Transient Response; G = +2

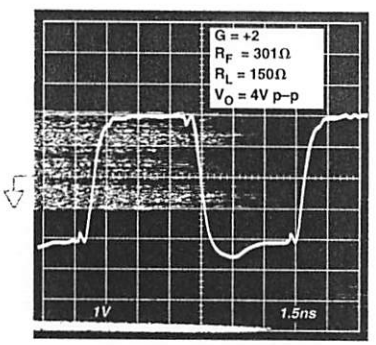


Figure 30. 4 V Transient Response; G = +2

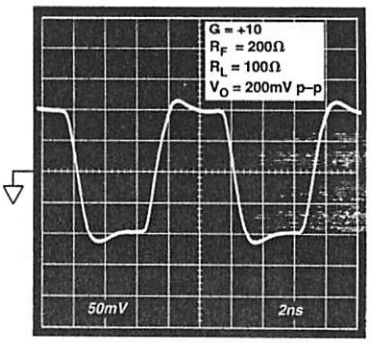


Figure 31. Small Signal Transient Response; G = +10

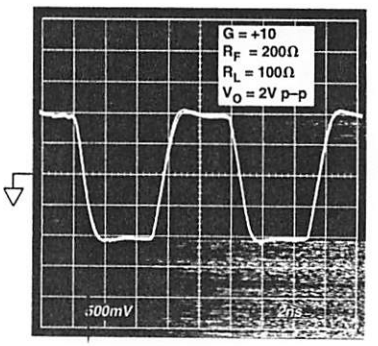


Figure 32. 2 V Transient Response; G = +10

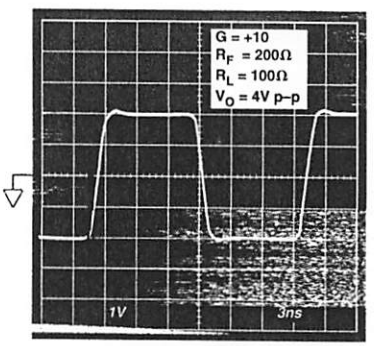


Figure 33. 4 V Transient Response; G = +10

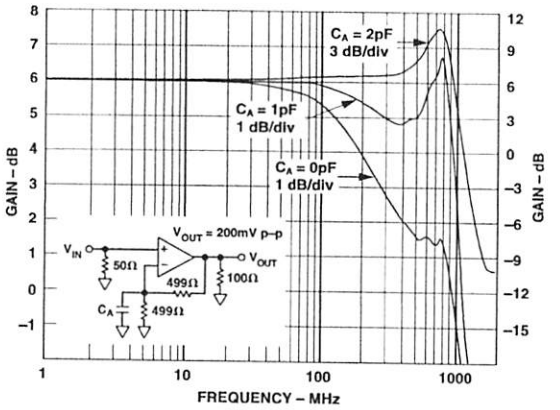


Figure 34. Small Signal Frequency Response vs. Parasitic Capacitance

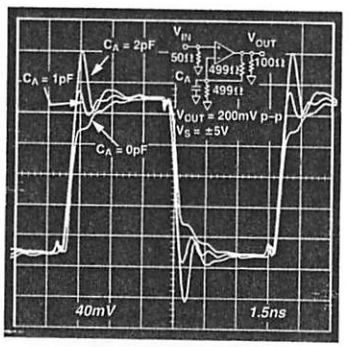


Figure 35. Small Signal Pulse Response vs. Parasitic Capacitance

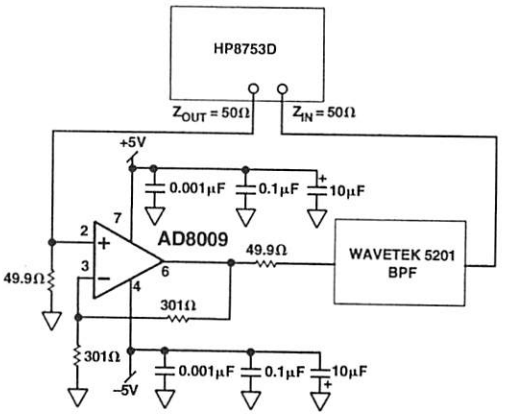


Figure 36. AD8009 Driving a Bandpass RF Filter

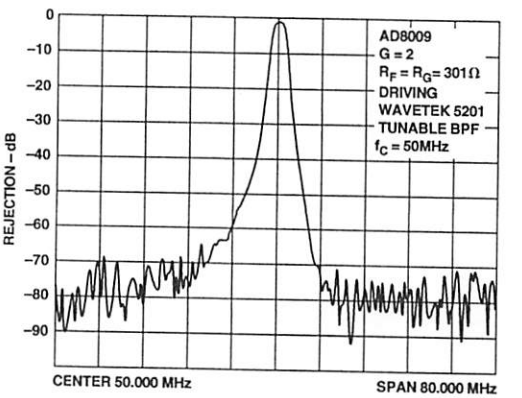


Figure 37. Frequency Response of Bandpass Filter Circuit

APPLICATIONS

All current feedback op amps are affected by stray capacitance on their –INPUT. Figures 34 and 35 illustrate the AD8009's response to such capacitance.

Figure 34 shows the bandwidth can be extended by placing a capacitor in parallel with the gain resistor. The small signal pulse response corresponding to such an increase in capacitance/bandwidth is shown in Figure 35.

As a practical consideration, the higher the capacitance on the –INPUT to GND, the higher R_F needs to be to minimize peaking/ringing.

RF Filter Driver

The output drive capability, wide bandwidth and low distortion of the AD8009 are well suited for creating gain blocks that can drive RF filters. Many of these filters require that the input be driven by a 50 Ω source, while the output must be terminated in 50 Ω for the filters to exhibit their specified frequency response.

Figure 36 shows a circuit for driving and measuring the frequency response of a filter, a Wavetek 5201 Tunable Band Pass Filter that is tuned to a 50 MHz center frequency. The HP8753D network provides a stimulus signal for the measurement. The analyzer has a 50 Ω source impedance that drives a cable that is terminated in 50 Ω at the high impedance noninverting input of the AD8009.

The AD8009 is set at a gain of two. The series 50 Ω resistor at the output, along with the 50 Ω termination provided by the filter and its termination, yield an overall unity gain for the measured path. The frequency response plot of Figure 37 shows the circuit to have an insertion loss of 1.3 dB in the pass band and about 75 dB rejection in the stop band.

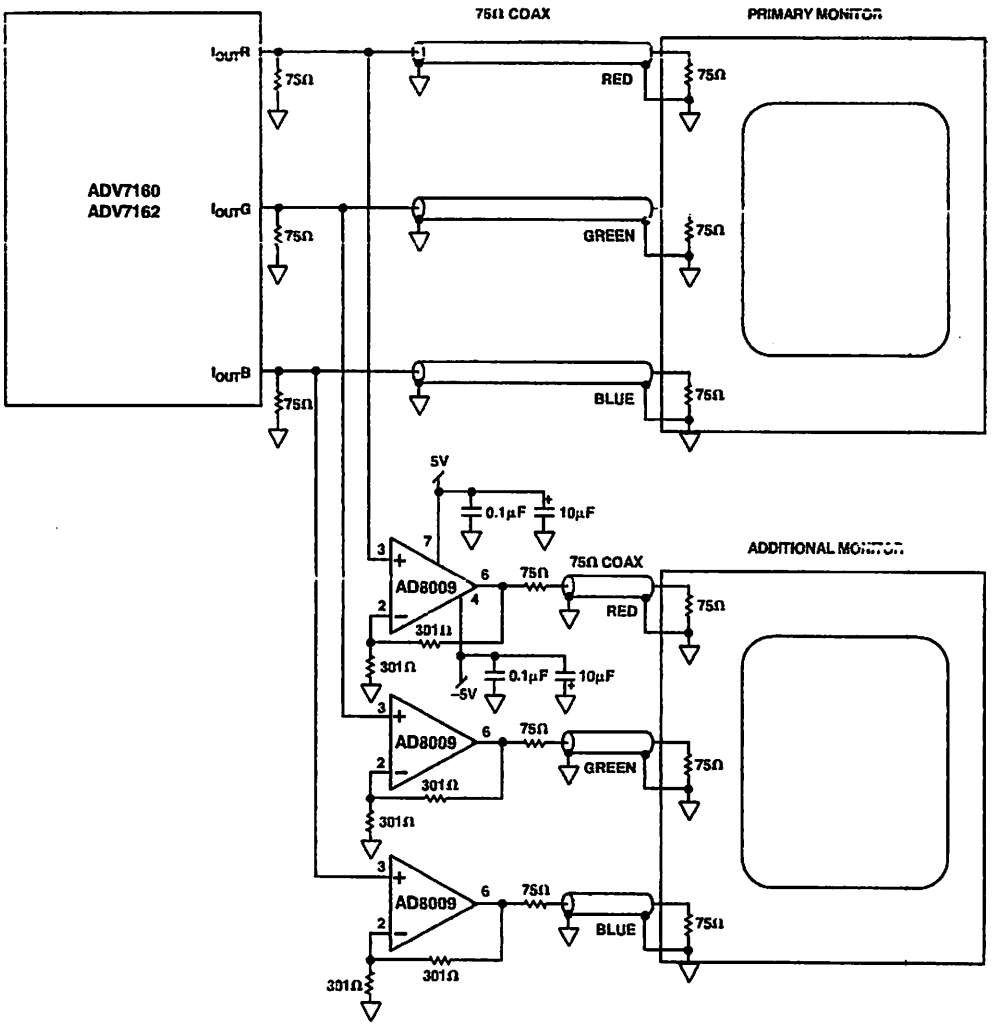


Figure 38. Driving an Additional High Resolution Monitor Using Three AD8009s

RGB Monitor Driver

High resolution computer monitors require very high full power bandwidth signals to maximize their display resolution. The RGB signals that drive these monitors are generally provided by a current-out RAMDAC that can directly drive a 75 Ω doubly terminated line.

There are times when the same output wants to be delivered to additional monitors. The termination provided internally by each monitor prohibits the ability to simply connect a second monitor in parallel with the first. Additional buffering must be provided.

Figure 38 shows a connection diagram for two high resolution monitors being driven by an ADV7160 or ADV7162, a 220 MHz (mega-pixel per second) triple RAMDAC. This pixel rate requires a driver whose full power bandwidth is at least half the pixel rate or 110 MHz. This is to provide good resolution for a worst case signal that swings between zero scale and full scale on adjacent pixels.

The primary monitor is connected in the conventional fashion with a 75 Ω termination to ground at each end of the 75 Ω cable. Sometimes this configuration is called "doubly terminated" and is used when the driver is a high output impedance current source.

For the additional monitor, each of the RGB signals close to the RAMDAC output is applied to a high input impedance, noninverting input of an AD8009 that is configured for a gain of +2. The outputs each drive a series 75 Ω resistor, cable and termination resistor in the monitor that divides the output signal by two, thus providing an overall unity gain. This scheme is referred to as "back termination" and is used when the driver is a low output impedance voltage source. Back termination requires that the voltage of the signal be double the value that the monitor sees. Double termination requires that the output current be double the value that flows in the monitor termination.

Driving a Capacitive Load

A capacitive load, like that presented by some A/D converters, can sometimes be a challenge for an op amp to drive depending on the architecture of the op amp. Most of the problem is caused by the pole created by the output impedance of the op amp and the capacitor that is driven. This creates extra phase shift that can eventually cause the op amp to become unstable.

One way to prevent instability and improve settling time when driving a capacitor is to insert a resistor in series between the op amp output and the capacitor. The feedback resistor is still connected directly to the output of the op amp, while the series resistor provides some isolation of the capacitive load from the op amp output.

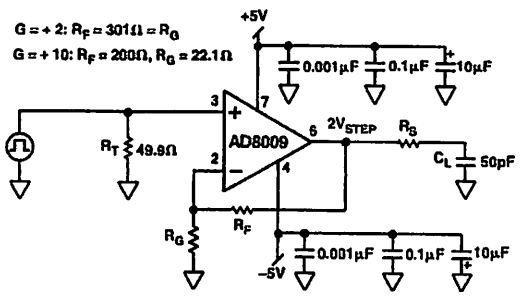


Figure 39. Capacitive Load Drive Circuit

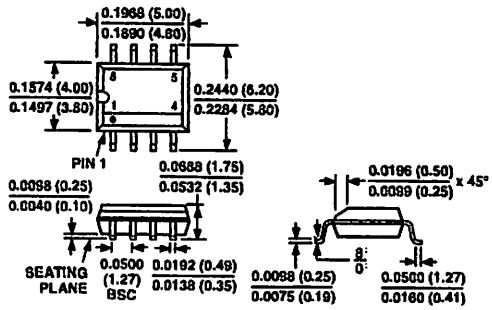
Figure 39 shows such a circuit with an AD8009 driving a 50 pF load. With R_S = 0, the AD8009 circuit will be unstable. For a gain of +2 and +10, it was found experimentally that setting R_S to 42.2 Ω will minimize the 0.1% settling time with a 2 V step at the output. The 0.1% settling time was measured to be 40 ns with this circuit.

For smaller capacitive loads, a smaller R_S will yield optimal settling time, while a larger R_S will be required for larger capacitive loads. Of course, a larger capacitance will always require more time for settling to a given accuracy than a smaller one, and this will be lengthened by the increase in R_S required. At best, a given RC combination will require about 7 time constants by itself to settle to 0.1%, so a limit will be reached where too large a capacitance cannot be driven by a given op amp and still meet the system's required settling time specification.

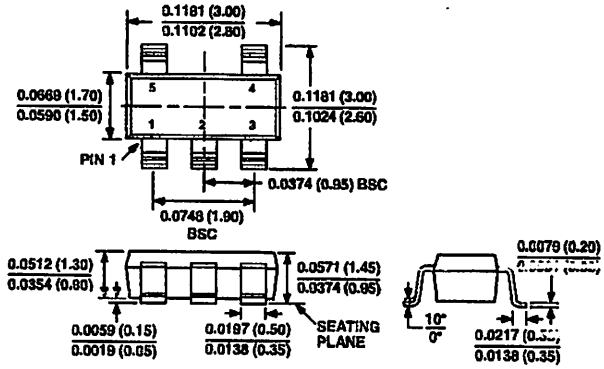
OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).

8-Lead SOIC
(SO-8)



5-Lead Plastic Surface Mount (SOT-23)
(RT-5)



C01011a-0-10/00 (rev. D)

PRINTED IN U.S.A.