SKRIPSI

PERANCANGAN SISTEM REMINDER DILENGKAPI DENGAN OUTPUT SUARA

Disusun Oleh :

NAMA : Andi Fauzi Firdaus

NIM : 95.17.038

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK ELEKTRONIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2009

LEMBAR PERSETUJUAN

PERENCANAAN SISTEM REMINDER DILENGKAPI DENGAN OUTPUT SUARA

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik

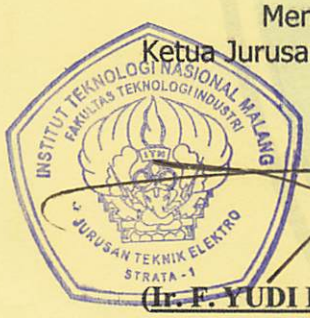
Program Studi Teknik Elektronika

Disusun Oleh :

Nama : Andi Fauzi Firdaus

NIM : 95.17.038

Mengetahui,
Ketua Jurusan Teknik Elektro S-1



(Ir. F. YUDI LIMPRAPTONO, MT)
NIP. Y.1039500274

Diperiksa/Disetujui
Dosen Pembimbing

(Ir. F. YUDI LIMPRAPTONO, MT)
NIP. Y.1039500274

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK ELEKTRONIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2009

LEMBAR PERSETUJUAN

PERENCANAAN SISTEM REMINDER DILENGKAPI DENGAN OUTPUT SUARA

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik
Program Studi Teknik Elektronika

Disusun Oleh :

Nama : **Andi Fauzi Firdaus**
NIM : **95.17.038**

Mengetahui,
Ketua Jurusan Teknik Elektro S-1

(Ir. E. YUDI LIMPRAPTONO, MT)
NIP. Y.1039500274

Diperiksa/Disetujui
Dosen Pembimbing

(Ir. E. YUDI LIMPRAPTONO, MT)
NIP. Y.1039500274

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK ELEKTRONIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2009

INSTITUT TEKNOLOGI NASIONAL
JL. RAYA KARANGLO, KM2
MALANG

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

1. Nama Mahasiswa : Andi Fauzi Firdaus
2. N . I . M : 95.17.038
3. N . I . R . M : 95.7.061.32022.18088
4. Jurusan : TEKNIK ELEKTRO S-1
5. Konsentrasi : T. Elektronika
6. Judul Skripsi : **Perancangan Sistem Reminder Dilengkapi Dengan Output Suara**

Dipertahankan dihadapan Tim Penguji Skripsi Jenjang program Strata-1 (S-1), pada :

Hari : Senin
Tanggal : 23 Maret 2009
Dengan Nilai : 76,25 (B+)

Panitia Ujian Tugas Akhir,

Ketua Majelis Penguji,

INSTITUT TEKNOLOGI NASIONAL MALANG

Sekretaris Majelis Penguji,

(Ir. H. Sidik Noertjahjono, MT)
NIP.Y.1028700163

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

Penguji I

(Ir. TH. Mimien Mustikawati, MT)
NIP.P.1030000352

Penguji II

(I Komang Somawirata, ST)
NIP.Y.1030100361

KATA PENGANTAR

Segala puji dan syukur kami panjatkan kehadirat Allah SWT atas limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan skripsi ini dengan judul, *Perancangan Sistem Reminder Dilengkapi Dengan Output Suara.*

Skripsi ini merupakan salah satu syarat yang harus ditempuh dalam menyelesaikan program studi (S-1) Jurusan Teknik Elektro Konsentrasi Energi listrik di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis tidak lepas dari bantuan dari berbagai banyak pihak, oleh karena itu pada kesempatan ini penulis ingin mengucapkan banyak terima kasih kepada:

1. Sang Penciptaku Allah SWT.
2. Nabi dan Rasulku Muhammad SAW.
3. Bapak Dr.Ir.Abraham Lomi, MSEE selaku Rektor ITN Malang.
4. Bapak Ir.H.Sidik Noertjahjono,MT, selaku Dekan Fakultas Teknologi Industri ITN Malang.
5. Bapak Ir. F. Yudi Limpraptono, MT selaku Dosen Pembimbing dan Ketua Jurusan Teknik Elektro (S1) di Institut Teknologi Nasional Malang.
6. Bapak dan Ibu Dosen Teknik Elektro yang telah memberi saran dan kritik.
7. Rekan – rekan mahasiswa dan semuanya yang telah membantu sampai terselesaikannya skripsi ini.

8. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang telah memberi motivasi dan dorongan serta sarana dan prasarana bagi penulis dalam mengerjakan Skripsi ini.

Penyusun menyadari bahwa skripsi ini masih terdapat kekurangan. Oleh karena itu, penyusun mengharapkan saran dan kritik yang bersifat membangun dari pembaca. Dan semoga skripsi ini bermanfaat bagi mahasiswa Institut Teknologi Nasional Malang.

Malang, Maret 2009

Penyusun

ABSTRAK

Perancangan Sistem Reminder dilengkapi dengan
output suara. Andi Fauzi Firdaus,
NIM 95.17.038, Teknik Elektro S1
Dosen Pembimbing : IR. F. YUDI LIMPRAPTONO, MT

Kata Kunci : AT89S51, RTC DS1307, ISD 1420.

Alat ini dirancang dengan memanfaatkan RTC DS1307 sebagai pewaktu yang dikontrol oleh Mikrokontroler AT89S51 dan dilengkapi dengan LCD M1632 sebagai display dan menggunakan ISD 1420 *(Informasi Stroge Device)* untuk menghasilkan output suara. Alat ini dapat digunakan sebagai sistem reminder berdasarkan input seting *user* berdasarkan tahun, tanggal, jam, menit dan detik yang diinginkan dengan output suara.

Alat ini menggunakan AT89S51 sebagai pengontrol, *Real Time Clock* DS1307 sebagai pewaktu. Sebagai input adalah 4 tombol push-button untuk melakukan seting waktu alarm, dan untuk memasukkan input suara kedalam ISD *(record)*. Sedangkan output-nya adalah output waktu pada LCD, output buzzer sebagai alarm, dan output suara yang terekam dalam ISD.

Pertama-tama kita dapat melakukan seting waktu sistem reminder melalui tombol *push button menu, up,* dan *down*, dan kemudian memasukkan input suara agar nantinya dihasilkan output suara sebagai alarm. Pada saat waktu telah sesuai dengan input nilai waktu alarm, maka secara otomatis mikrokontroler akan mengeluarkan suara alarm dan output suara sesuai dengan seting alarm yang telah di inputkan.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1. Latar Belakang

Mikrokontroler sebagai suatu terobosan teknologi mikroposesor dan mikrokomputer hadir memenuhi kebutuhan pasar *(market need)* dan teknologi baru. Sebagai teknologi baru, yaitu teknologi semikonduktor dengan kandungan transistor yang lebih banyak namun hanya membutuhkan ruang yang kecil serta dapat digunakan sebagai pengendali utama dalam berbagai aplikasi-aplikasi pengontrolan, sistem kendali, otomatisasi, dan lain sebagainya

Tingginya aktivitas dalam memenuhi kebutuhan ekonomi seringkali membuat kita lupa pada kegiatan yang telah lama kita rencanakan jauh-jauh hari sebelumnya sehingga membuat kita rugi atau terkadang dapat membuat orang lain sakit hati karena janji kita yang tidak terpenuhi dikarenakan kita terlupa. Sudah menjadi sifat kita sebagai manusia yang sering pelupa sehingga berangkat dari permasalahan tersebut penulis mengambil judul untuk Skripsi yaitu **"Perancangan Sistem Reminder Dilengkapi dengan Output Suara"**.

Alat ini dibuat dengan menggunakan RTC DS1307 sebagai pewaktu yang dikontrol oleh Mikrokontroler AT89S51 dan dilengkapi dengan LCD M1632 sebagai display dan menggunakan ISD 1420 *(Informasi Stroge Device)* untuk menghasilkan *output* suara. Diharapkan alat ini dapat digunakan sebagai sistem reminder berdasarkan input oleh User berdasarkan tahun, tanggal, jam, menit dan detik yang diinginkan dengan output suara.

1.2. Rumusan Masalah

Dari uraian tersebut diatas maka timbul beberapa permasalahan diantaranya adalah:

1. Bagaimana Perancangan Perangkat Keras dari Sistem ?
2. Bagaimana Perancangan Perangkat Lunak dari Sistem ?

1.3. Tujuan

Perancangan dan pembuatan sistem reminder dilengkapi dengan output suara untuk membantu kita dalam mengingat kegiatan yang telah kita rencanakan jauh-jauh hari sebelumnya dengan output suara.

1.4. Batasan Masalah

Agar permasalahan tidak meluas maka penulis membatasi permasalahan sebagai berikut:

1. Menggunakan Mikrokontroler AT89S51 sebagai pengendali utama..
2. Menggunakan RTC DS1307 sebagai pewaktu.
3. Menggunakan ISD 1420 untuk Voice Record.

1.5. Metodologi Penulisan

Metodologi yang dilakukan dalam penulisan Skripsi perancangan sistem reminder dilengkapi dengan output suara adalah sebagai berikut:

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

1.6. Sistematika Penulisan

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

BAB I : PENDAHULUAN

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

BAB II : DASAR TEORI

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

BAB III : PERENCANAAN DAN PEMBUATAN ALAT

Perancangan sistem reminder dilengkapi dengan output suara.

BAB IV : PENGUJIAN ALAT

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

BAB V : PENUTUP

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

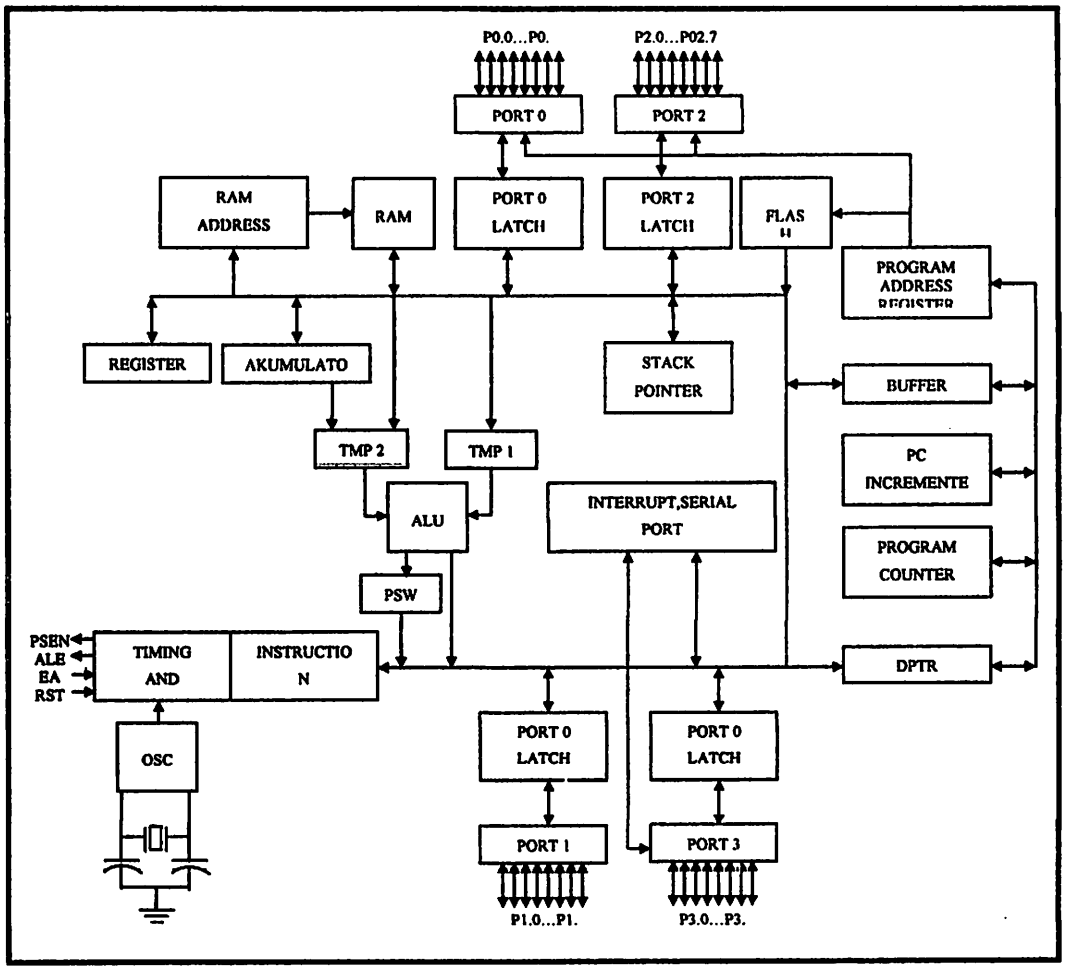
BAB II

TEORI DASAR

Bab ini akan menguraikan tentang dasar-dasar teori dasar yang menunjang dalam perancangan dan pembuatan alat. Uraian teori dalam bab ini meliputi teori IC AT89S51, ISD 1420, LCD M1632 dan perangkat-perangkat pendukung lainnya.

2.1. Mikrokontroler AT89S51

2.1.1. Teori umum.



Gambar 2.1. Diagram Blok AT89S51 [4]

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

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

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

2.1.2. Arsitektur AT89S51

Arsitektur Mikrokontroler AT89S51 adalah sebagai berikut:

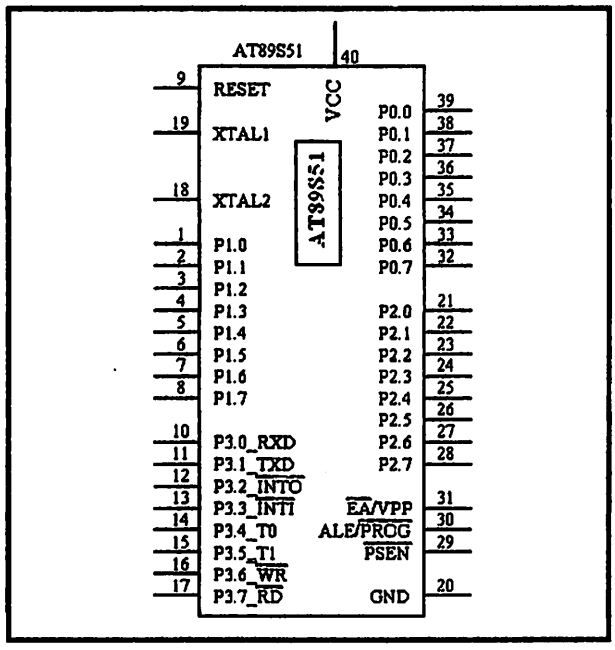
- CPU (Central Processing Unit) 8-bit dengan register A (Accumulator) dan B.
- 16-Bit Program Counter (PC) dan data pointer (DPTR).
- 8-Bit Program status word (PSW).
- 4-Bit Stack pointer (SP).
- 4 Kbyte internal EPROM
- 128 byte internal RAM.

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

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

2.1.3. Konfigurasi Pin pada Mikrokontroler AT89S51

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



Gambar 2.2. Konfigurasi Pin-Pin AT89S51 [4]

Fungsi dari tiap-tiap pin adalah sebagai berikut:

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

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

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

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

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

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

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

7. RST

   Input reset.

8. ALE\PROG

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

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

9. PSEN

   Merupakan strobe baca ke program memory external.

10. EA\VPP

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

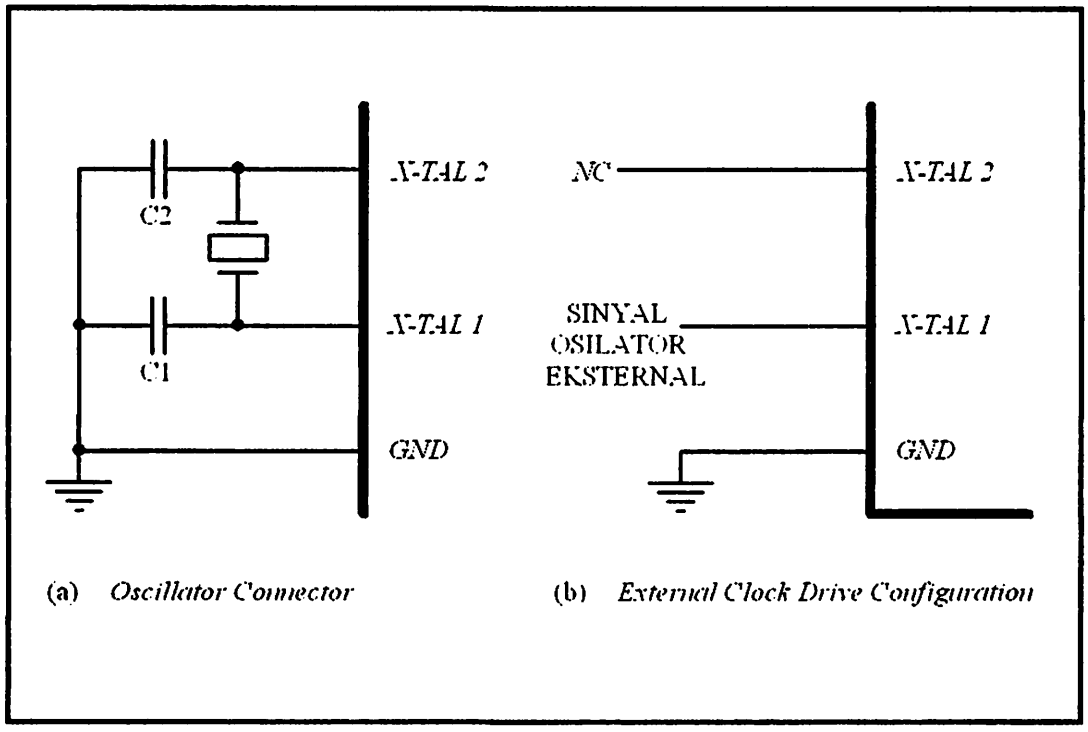
11. XTAL 1 dan XTAL 2.

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

2.1.4. Karakteristik Oscillator Inverting

X-TAL 1 dan X-TAL 2 secara berurutan merupakan input dan output dari sebuah inverting amplifier yang dapat dikonfigurasikan penggunaannya sebagai on chip oscillator seperti yang ditunjukkan pada gambar 2-3a. X-TAL 1 dan X-TAL 2 ini dapat menggunakan sebuah kristal quartz maupun resonator keramik.

Untuk memberikan AT89S51 dari sumber clock external. Maka pin X-TAL 2 dibiarkan tidak berhubungan dan X-TAL 1 dihubungkan dengan sumber clock external seperti pada gambar 2-3b. Rangkaian ini tidak melakukan duty cycle dari setiap sinyal clock internal, karena input bagi masukan rangkaian clock internal dihubungkan ke flip-flop pembagi dua, tetapi spesifikasi nilai tegangan pada saat tinggi dan rendah, maksimum dan minimumnya harus diberikan.



Gambar 2.3. Karakteristik Oscilator [4]

2.1.5. Organisasi Memory

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

2.1.5.1. Program Memory Internal

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

2.1.5.2. Data Memory (RAM) Internal

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

- 4 Bank Register

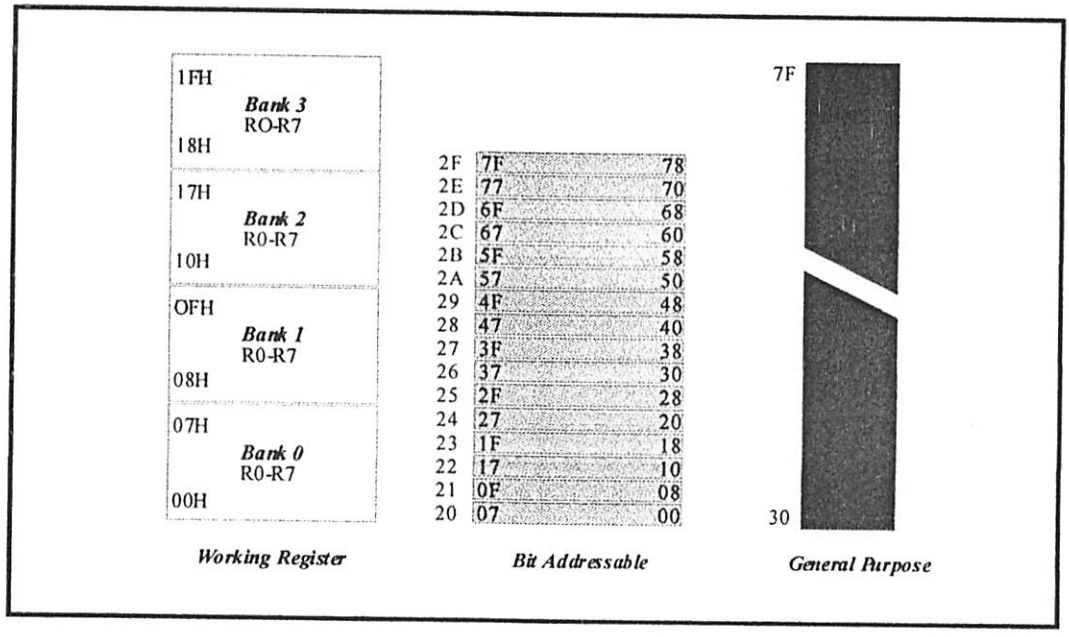
  Setiap bank terdiri dari 8 register (R0-R7). Sehingga jumlah register untuk keempat bank register (bank 0 - bank 3) menjadi 32 buah register yang menempati ruang alamat 00H-1FH. Mengaktifkan salah satu bank register dapat dilakukan dengan mengatur RSO-RSI pada PSW (Program Status Word).

- Bit Addressable.

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

- General Purpose.

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



Gambar 2.4. Organisasi RAM Internal [4]

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

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

2.1.5.3. SFR (Special Function Register).

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

Tabel 2-2. 128 Byte Special Function Register [4]

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

2.2. RTC DS1307

DS1307 adalah IC serial *Real Time Clock* (RTC) dimana alamat dan data ditransmisikan secara serial melalui sebuah jalur data dua arah I2C. Karena menggunakan jalur data I2C maka hanya memerlukan dua buah pin saja untuk komunikasi. Yaitu pin untuk data dan pin untuk sinyal clock. Sistem jalur data I2C adalah suatu standar protokol sistem komunikasi data serial yang dikembangkan oleh Philips dan cukup populer dikarenakan penggunaannya cukup mudah.

Pada dasarnya, pada sistem I2C terbagi atas dua bagian, yaitu suatu *device* yang bertindak sebagai pengontrol atau Master dan suatu *device* yang dikontrol atau *Slave*. Master dan *Slave* saling berkomunikasi melalui jalur data bus I2C. Alat yang mengendalikan komunikasi data disebut Master dan alat yang dikendalikan oleh Master dikenal sebagai *Slave*. Dimana pada perancangan ini, penulis menggunakan mikrokontroler AT89S51 sebagai Master.

Pada satu jalur data I2C yang sama dapat terdapat *slave* lebih dari satu oleh karena itu I2CBus harus dikendalikan oleh sebuah Master yang dapat membangkitkan serial clock (SCL), mengontrol sistem komunikasi data (SDA), dan juga dapat menghasilkan kondisi-kondisi "START" dan "STOP". Pada hal ini DS1307 beroperasi sebagai *slave* pada I2C bus. Contoh bagaimana data ditransfer pada jalur data I2C adalah seperti pada gambar 2.5. Pada jalur data bus I2C hanya terdapat 2 buah jalur yang digunakan yaitu Clock (SCL) dan Data (SDA). Terdapat beberapa macam jenis kondisi pada jalur data I2C, jenis kondisi tersebut adalah:

1. ***Bus not busy:*** Jalur data (SDA) dan clock (SCL) berlogika high
2. ***Start data transfer:*** Suatu perubahan kondisi pada jalur data, dari logika *high* ke logika *low*, ketika jalur data sedang berlogika *high*, menandakan kondisi START.
3. ***Stop data transfer:*** Suatu perubahan kondisi pada jalur data, dari logika *low* ke logika *high*, ketika jalur data sedang berlogika *high*, menandakan kondisi STOP
4. ***Data valid:*** Suatu kondisi ketika jalur data menandakan data valid, yaitu ketika setelah kondisi START, jalur data tetap stabil selama periode *high* sinyal clock. Data pada jalur data harus berubah selama periode *low* dari sinyal clock. Terdapat satu pulsa clock untuk setiap bit data. Setiap proses pengiriman data dimulai dengan kondisi START dan diakhiri dengan kondisi STOP. Banyaknya jumlah *byte* data yang ditransfer diantara kondisi START dan STOP tersebut tidak terbatas, dan diatur oleh Master.
5. ***Acknowledge:*** Setiap *device* yang dituju telah menerima data dengan benar akan membangkit kan kondisi *Acknowledge* setiap menerima *byte* data. *Device* yang membangkitkan *Acknowledge* harus membangkitkan logika *low* pada jalur SDA selama sebuah pulsa clock. Untuk mengakhiri suatu proses pengiriman data Master harus memberikan suatu tanda dengan tidak memberi kan tanda *acknowledge* melainkan memberikan tanda STOP pada *slave*.

Pada sistem jalur data I2C, terdapat dua tipe arah proses pengiriman data yaitu:

1. **Data transfer dari master menuju *slave*.** *Byte* pertama yang dikirimkan oleh master menuju *slave* adalah alamat *slave*. Lalu selanjutnya adalah *byte-byte*

Data Hasil Pengujian :

Tabel 4-1. Tabel Hasil Pengujian Mikrokontroler

Kondisi	Nyala LED
Tekan P3.0	1111 1111
Tekan P3.1	1111 1110
Tekan P3.2	1111 1101
Tekan P3.3	1111 1100
Tekan P3.4	1111 1011
Tekan P3.5	1111 1010
Tekan P3.6	1111 1001
Tekan P3.7	1111 1000

4.2. Pengujian Rangkaian Pemutar/Perekam Suara ISD1420

4.2.1 Tujuan

Pengujian Rangkaian Pemutar/Perekam Suara ISD1420 dilakukan untuk mengetahui apakah IC ISD1420 dapat dioperasikan untuk menyimpan suara dan memutar suara yang telah direkam.

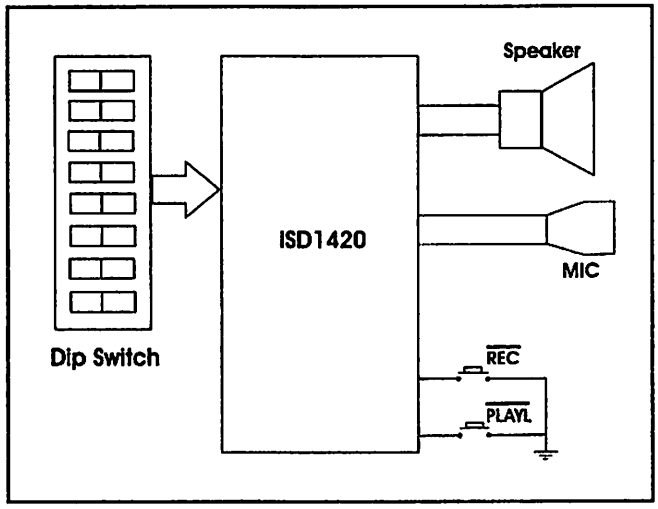
4.2.2 Peralatan Yang Digunakan

- Modul IC ISD1420
- Mikrokontroler AT89C51
- Catu daya 5 volt

4.2.3 Prosedur Pengujian

1. Merangkai rangkaian pengujian pemutar/perekam suara ISD1420 seperti ditunjukkan dalam Gambar 4.2.
2. Alamat penyimpanan suara yang akan direkam diset terlebih dahulu.

3. Untuk merekam suara, tekan tombol $\overline{REC}$. Setelah selesai merekam, tombol tersebut langsung dilepaskan.
4. Untuk memutar suara, tekan tombol $\overline{PLAYL}$ hingga akhir perekaman (ditandai dengan keluaran sinyal $\overline{\text{RECLED}}$ yang berlogika rendah) atau tombol $\overline{PLAYE}$.
5. Menguji memutar suara ISD1420 dengan program.



Gambar 4.2. Modul Pengujian IC ISD1420

Listing Program

```
Playe bit P2.5
;P0  Alamat ISD

mulai:      org 0h
      setb p2.5
      MOV A,#0
sini: mov p1,A
      mov p0,A
      JB P3.6,$
      JnB P3.6,$
      clr p2.5
      call delay
      setb p2.5
      mov r0,#10
      djnz r0,loncat
      jmp mulai
loncat:    add a,#10H
```

```
      jmp sini

delay:      mov   R7,#0
delay1: mov       R5,#0h
      djnz  R5,$
      djnz  R7,delay1
      ret
end
```

4.2.4 Hasil Pengujian

Hasil pengujian Rangkaian Pemutar/Perekam Suara ISD1420 dapat dilihat pada Tabel 4-2 dibawah ini:

Tabel 4-2. Pemilihan Alamat untuk Merekam Suara

Alamat Awal	Durasi (detik)	Kata	Kualitas Suara	Durasi Output
0000 0000	0,85	Nol	baik	0,85
0000 1010	0,85	Satu	baik	0,85
0001 0100	0,85	Dua	baik	0,85
0001 1110	0,85	Tiga	baik	0,85
0010 1000	0,85	Empat	baik	0,85
0011 0010	0,85	Lima	baik	0,85
0011 1100	0,85	Enam	baik	0,85
0100 0110	0,85	Tujuh	baik	0,85
0101 0000	0,85	Delapan	baik	0,85
0101 1010	0,85	Sembilan	baik	0,85
0110 0100	0,85	Puluh	baik	0,85
0110 1110	0,85	Belas	baik	0,85
0111 1000	0,85	Ratus	baik	0,85
Total durasi:	10,2			10,2

Tabel 4-2 menunjukkan pemilihan alamat untuk merekam suara serta durasi penyimpanannya. Untuk memutar suara yang telah direkam maka tombol $\overline{PLAYL}$ atau $\overline{PLAYE}$ diberi logika rendah dan pada waktu suara selesai diputar maka secara otomatis sinyal $\overline{RECLED}$ akan berlogika rendah dan ditandai menyalanya lampu LED.

4.2.5 Analisis hasil pengujian

Lamanya perekaman akan menentukan banyaknya alamat yang dipakai oleh ISD1420 semakin lama merekam maka alamat yang dipakai semakin banyak. Jika IC ISD1420 dipakai untuk merekam suara lebih dari satu maka lamanya perekaman tidak boleh melebihi atau masuk pada alamat suara berikutnya, karena suara yang telah direkam akan terdengar pada alamat suara berikutnya dan otomatis suara yang telah direkam akan terpotong.

Penggunaan program pada mikrokontroler akan memperdengarkan kembali suara yang terekam pada ISD1420 secara berurutan mulai dari alamat awal penyimpanan, dengan memberi logika 0 (rendah) pada port 3.6.

Berdasarkan hasil pengujian, rangkaian perekam/pemutar ulang suara menunjukkan bahwa rangkaian telah mampu memperdengarkan kembali suara yang telah direkam selama 10,2 detik.

4.3. Pengujian Rangkaian Tampilan LCD

- **Tujuan**

Untuk mengetahui kemampuan rangkaian tampilan yang sudah dibuat apakah dapat mendukung sistem yang direncanakan untuk memampilkan data pada LCD.

- **Peralatan yang dibutuhkan**

1. Power Supply 5 Volt
2. Sistem Mikrokontroler dan LCD M1632

- **Prosedur Pengujian**

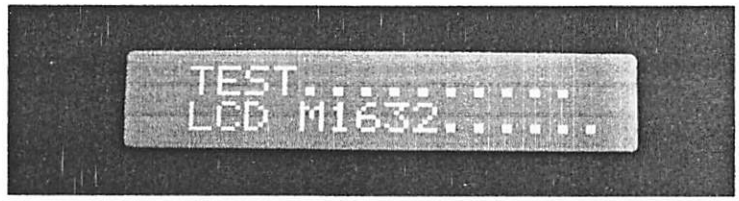
1. Menyusun rangkaian seperti dalam Gambar 4.3

2. Menjalankan program untuk menampilkan tulisan

   ' TEST . . . . . '

   'LCD M1632 . . . .'

3. Mengamati keluaran pada LCD

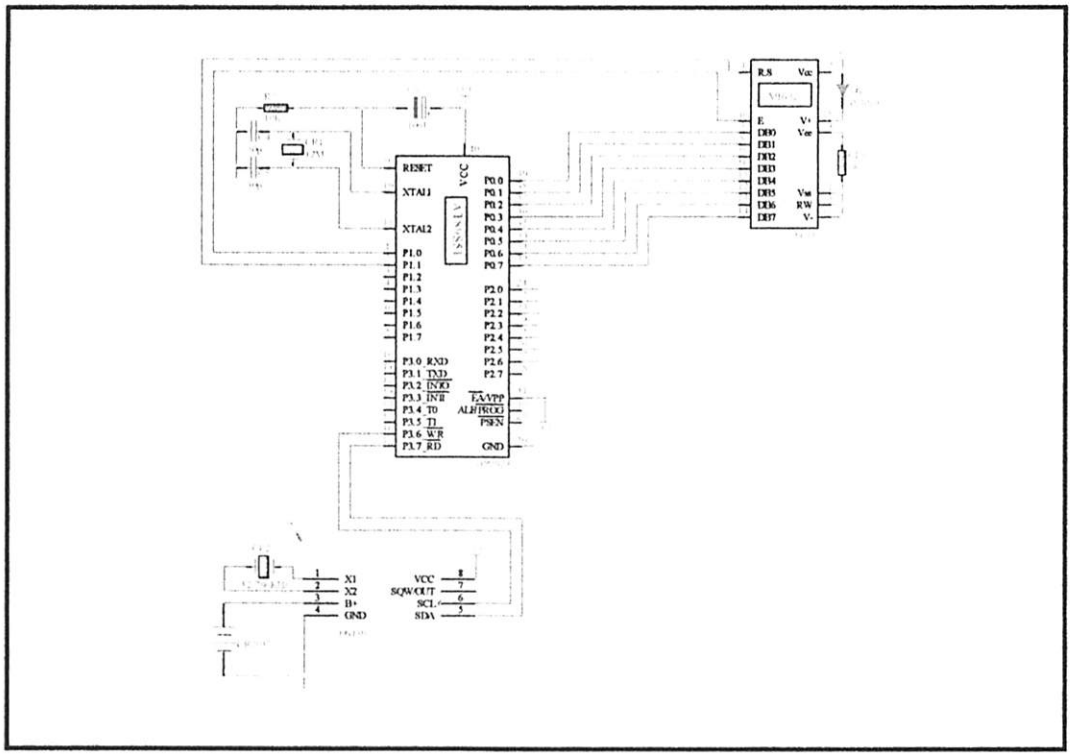


Gambar 4.3. Foto Tampilan LCD Hasil Percobaan

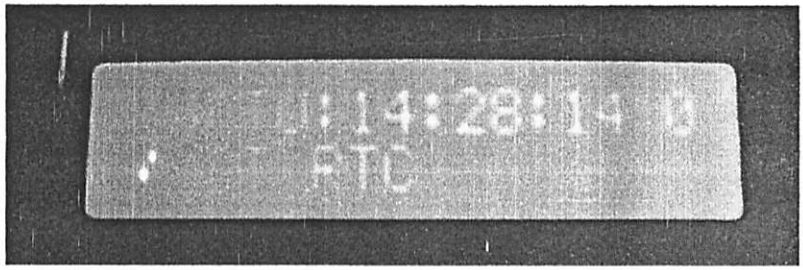
4.4. Pengujian Terhadap Rangkaian RTC DS1307

RTC DS1307 merupakan modul *Real Time Clock*, yaitu sebuah modul yang mempunyai sistem jam digital yang bekerja secara independent. Artinya, sistem jam digital pada modul RTC DS1307 ini bekerja mengaktifkan sistem jam digital di mana besaran jam, menit dan detik tersimpan dalam register-register tertentu dalam modul RTC DS1307. Seperti pembahasan pada bab sebelumnya bahwa RTC DS1307 berfungsi sebagai pewaktu maka pengujian dapat dilakukan dengan cara sebagai berikut:

1. Membuat rangkaian seperti gambar 4.4.
2. Memasukkan program untuk tes RTC pada mikrokontroler.
3. Menggunakan keypad untuk seting awal waktu.
4. Mengamati Waktu pada LCD.



Gambar 4.4. Rangkaian Pengujian RTC DS1307



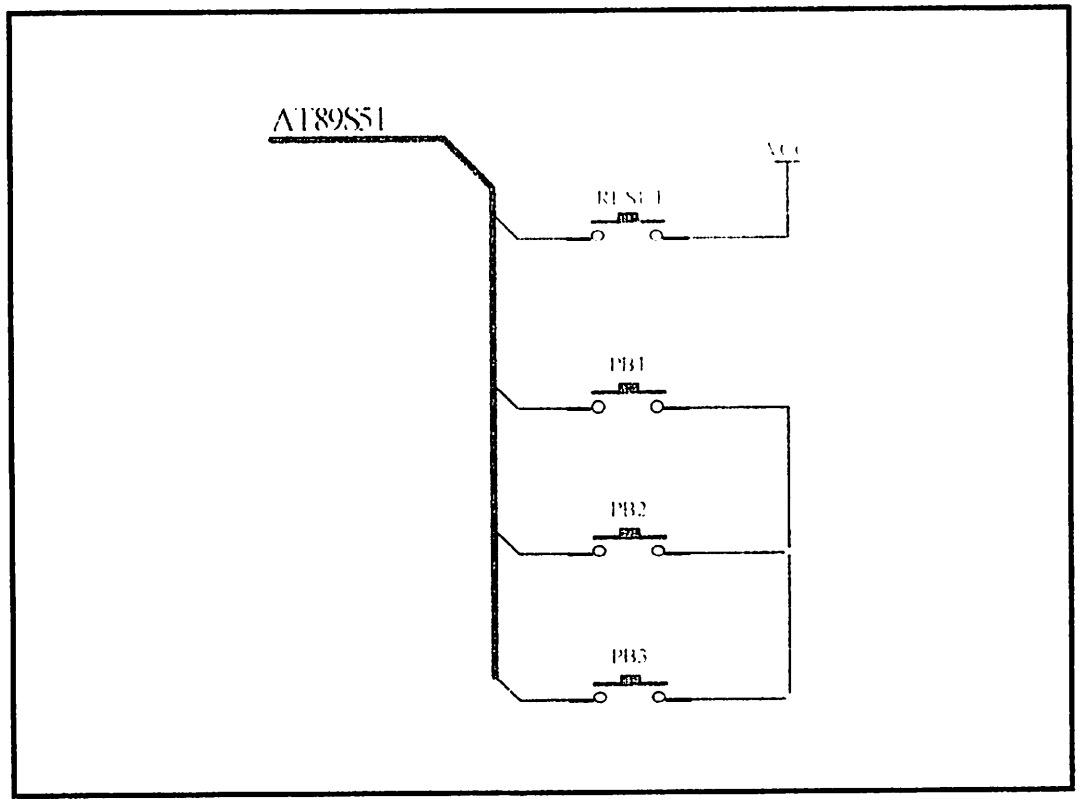
Gambar 4.5. Foto Hasil Pengujian Rangkaian RTC DS1307

4.5. Pengujian Terhadap Rangkaian *Push-button*

Rangkaian *push-button* berfungsi sebagai inputan mikrokontroler untuk seting sistem reminder. *Push-button* diberikan inputan tegangan sebesar 4,9 *Volt* dan dirangkai dengan menggunakan *common ground.* Sehingga apabila terjadi penekanan pada tombol *push-button*, maka akan terjadi arus hubung singkat yang

menyebabkan kondisi tegangan pada pin *push-button* yang terhubung dengan mikrokontroler berubah dari kondisi *"high"* 4,9 *Volt* menjadi kondisi *"low"* 0 *Volt.*

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



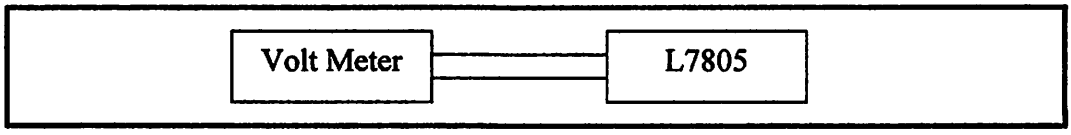
Gambar 4.6. Cara Pengukuran Tegangan Pada Pengujian Rangkaian *Push-Button*

Tabel 4-3. Hasil Pengukuran Pada Rangkaian *Push-Button*

Push-Button (Tertekan)	P1.5 (V)	P1.6 (V)	P1.7 (V)	RESET (V)
PB 1 *(Menu)*	0	4,9	4,9	0
PB 2 *(Up)*	4,9	0	4,9	0
PB 3 *(Down)*	4,9	4,9	0	0
RESET	4,9	4,9	4,9	4,9

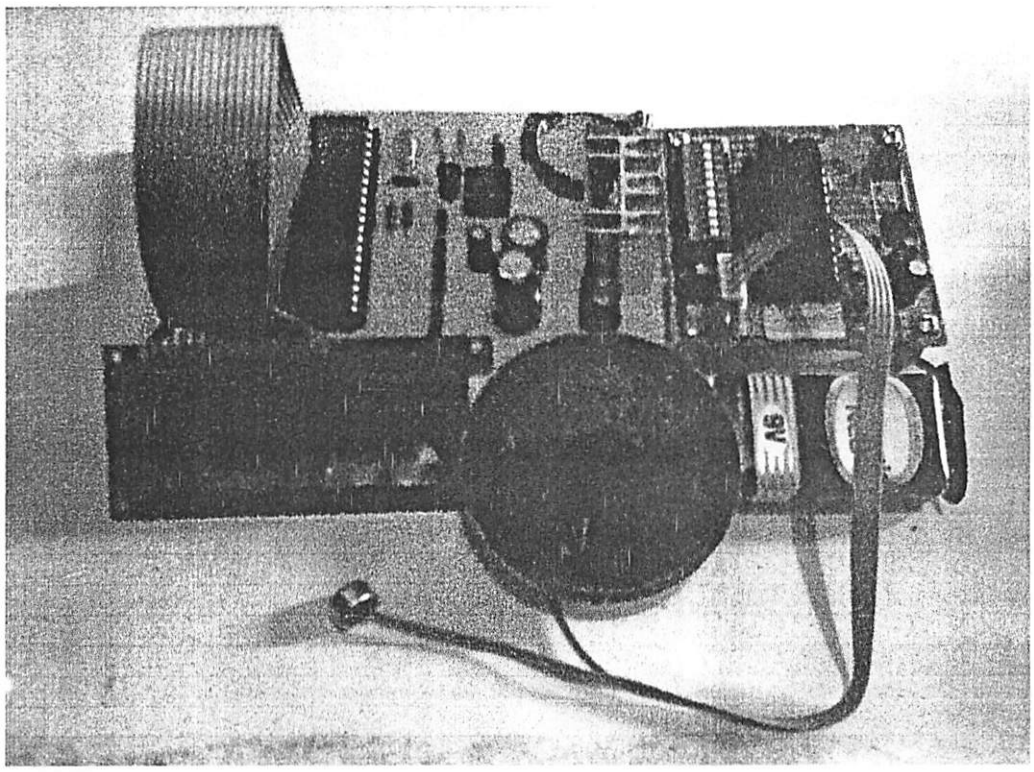
4.6. Pengujian Terhadap Rangkaian *Power Supply*

Pengujian ini bertujuan untuk mengetahui tegangan yang dikeluarkan oleh rangkaian *power supply* yang telah dibuat. Dengan begitu dapat diketahui apakah terjadi kesalahan terhadap rangkaian *power supply* atau tidak. Tegangan yang dibutuhkan untuk memberikan tegangan kerja untuk rangkaian keseluruhan adalah +5V. Untuk mengukur besarnya tegangan pada rangkaian *power supply* maka pada pengujian rangkaian *power supply* ini menggunakan Multimeter Digital. Berikut cara pengukuran untuk pengambilan data besarnya tegangan pada rangkaian *power supply*.



Gambar 4.7. Diagram Blok Pengujian Tegangan *Power Supply*

Untuk mengukur besarnya tegangan keluaran rangkaian *power supply* maka tegangan masukan pada rangkaian *power supply* juga harus tersambung dengan sumber tegangannya yaitu baterai 9 Volt. Multimeter digital di-seting pada *DC Volt* dengan batas maksimal pengukuran yaitu 20 *Volt DC*. Pengukuran tegangan dilakukan pada *output* keluaran dari *regulator* tegangan LM7805. Dari



Gambar 4.8. Foto Alat

data. *Slave* membalas dengan bit *acknowledge* setiap berhasil menerima 1 *byte* data.

2. **Data transfer dari *slave* menuju master.** *Byte* pertama (alamat *slave*) dikirimkan oleh master. Kemudian *slave* yang mempunyai alamat yang dituju oleh master membalas dengan bit *acknowledge*. Lalu diikuti dengan proses pengiriman *byte-byte* data dari *slave* menuju master. Master membalas dengan mengirimkan bit *acknowledge* setiap berhasil menerima 1 *Byte*. Untuk mengakhiri proses pengiriman data master membalas dengan mengirimkan bit *"not acknowledge"* kepada *slave*.

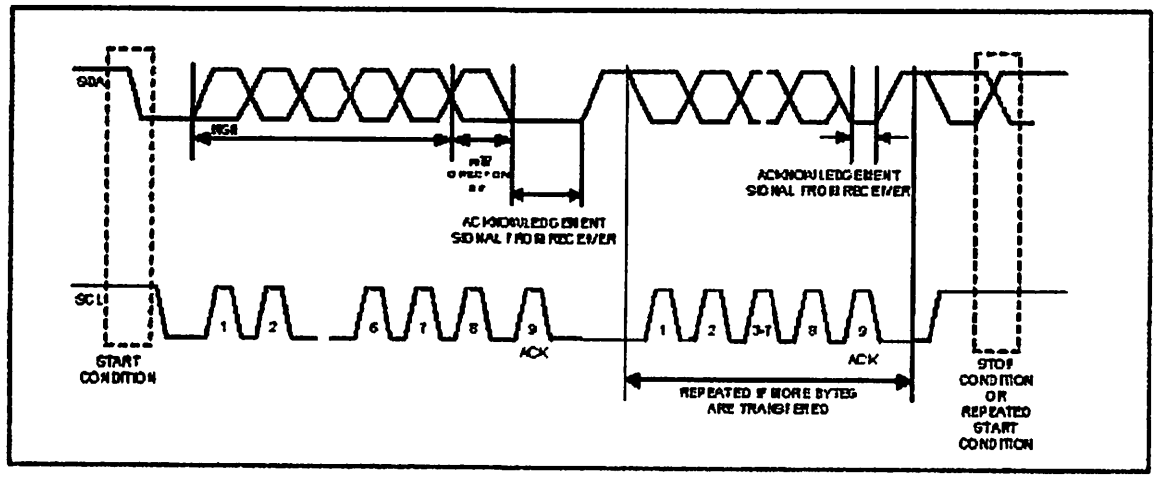
Pada aplikasi ini DS1307 bekerja dengan dua mode, yaitu:

1. **Mode *Slave* Penerima (Master Menulis Pada *Slave*):** Data serial dan clock diterima melalui SDA dan SCL. Setiap menerima *byte* data DS1307 akan merespon dengan membangkitkan bit *acknowledge*. Untuk mengawali proses pengiriman data dari master menuju *slave* diawali dengan kondisi START dan diakhiri dengan kondisi STOP. Setiap *slave* akan membaca alamat yang dituju oleh master dan memeriksa apakah alamat tersebut sama dengan alamat *Slave* tersebut. *Byte* alamat *slave* adalah *Byte* pertama yang diterima *slave* setelah master membangkitkan kondisi START. *Byte* alamat terdiri dari 7 bit data, untuk DS1307 *Byte* alamat tersebut adalah 1101000b, dan diikuti oleh bit arah (R/W), yang mana untuk penulisan data ke *slave* adalah 0. Setelah menerima dan menganalisa *Byte* alamat, DS1307 membangkitkan tanda *acknowledge* pada jalur SDA. Kemudian master akan mengirimkan sebuah data word alamat pada DS1307 untuk mengeset regiter pointer pada DS1307. Setelah itu Master dapat mengakhiri proses pengiriman data ataupun

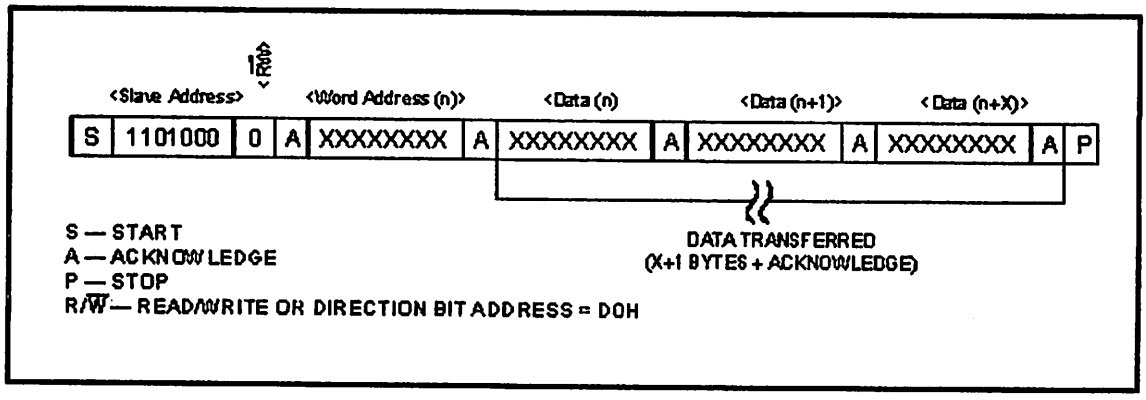
melanjutkannya dengan mengirimkan *Byte* data pada DS1307. Register pointer akan bertambah nilainya secara otomatis setiap terjadi proses penulisan data. Untuk mengakhiri proses pengiriman data master membangkitkan kondisi STOP.

2. **Mode Slave Pengirim (Master Membaca Dari Slave):** *Byte* pertama diterima dan diolah oleh *slave* seperti pada mode penerima, tetapi bit arah bernilai 1. DS1307 mengirimkan data serial pada SDA ketika menerima sinyal clock pada SCL. Untuk memulai prose pengiriman data diawali dengan kondis START dan diakhiri dengan kondisi STOP. *Byte* yang berisi data alamat diterima setelah master membangkitkan kondisi START. *Byte* alamat DS1307 terdiri dari 7-bit alamat dan 1 bit arah. 7-bit alamat tersebut adalah 1101000 dan bit arah tersebut (R/W) adalah 0 untuk *read.* Setelah menerima dan mengolah data alamat, DS1307 akan membalas dengan membangkitkan bit *acknowledge* pada pin SDA. Kemudian DS1307 mulai mengirimkan data dimulai dari data yang terdapat pada alamat yang ditunjuk oleh register pointer. Nilai register pointer secara otomatis akan bertambah setiap terjadi proses pembacaan 1 *Byte* data. Untuk mengakhiri proses pengiriman data maka master harus mengirimkan tadan "not *acknowledge*" kepada *slave*.

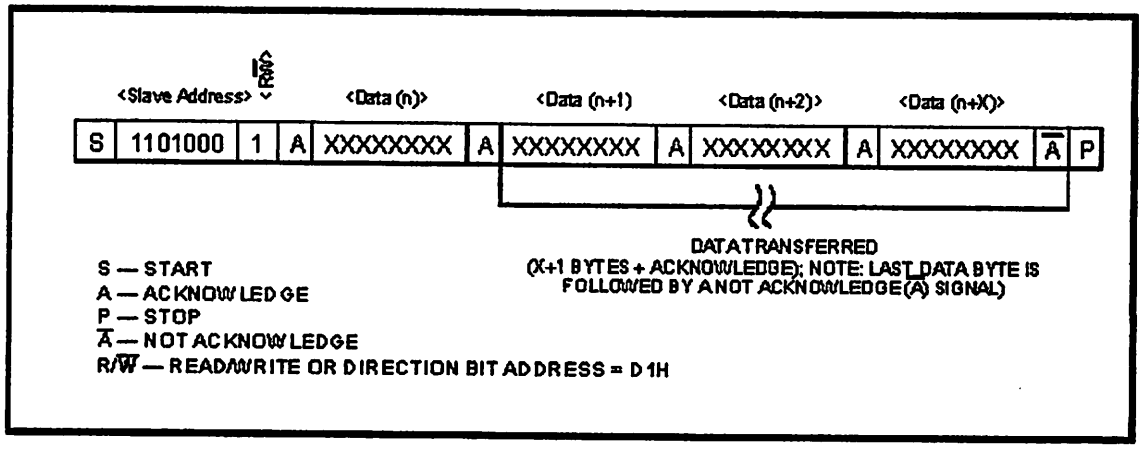
Untuk dapat mengambil nilai waktu dan tanggal maka master harus melakukan proses pembacaan data *(Read)* pada *slave* (DS1307), dengan alamat register sesuai dengan tabel 2-3. Setiap nilai-nilai waktu atau tanggal disimpan pada register yang mempunyai alamat yang berbeda-beda, misalnya untuk register detik yang menyimpan nilai detik, menempati alamat register 00h.



Gambar 2.5. Proses Transfer Data pada I2C [12]



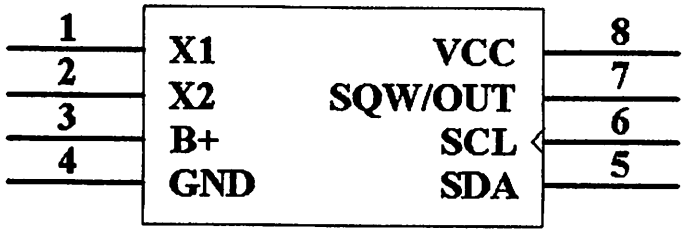
Gambar 2.6. Format Data [12]



Gambar 2.7. Format Data [12]

Tabel 2-3. Setting DS1307 [12]

ADDRESS	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	FUNCTION	RANGE
00H	CH	10 Seconds			Seconds				Seconds	00–59
01H	0	10 Minutes			Minutes				Minutes	00–59
02H	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23
		24	PM/AM							
03H	0	0	0	0	0	DAY			Day	01–07
04H	0	0	10 Date		Date				Date	01–31
05H	0	0	0	10 Month	Month				Month	01–12
06H	10 Year				Year				Year	00–99
07H	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08H-3FH									RAM 56 x 8	00H–FFH



Gambar 2.8. RTC DS1307 [12]

Konfigurasi Pin DS1307:

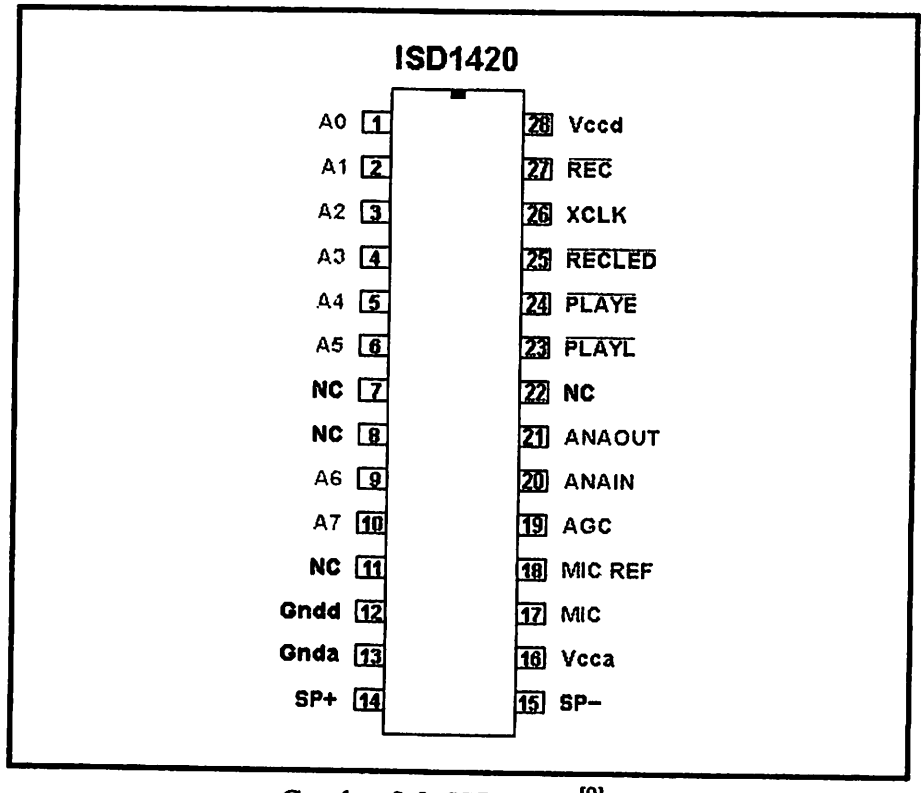
- **Vcc,** + 5 Volt
- **X1 dan X2,** merupakan jalur untuk koneksi ke kristal 32,768 kHz.
- **B+(V_{BAT}),** Input untuk baterai *back-up* tegangan sebesar 3Volt.
- **SQW/OUT,** *Square Wave/Output Driver*
- **SCL,** *Serial Clock*
- **SDA,** *Serial Data*
- **GND,** *Ground*

2.3. Speak Call ISD 1420 (IC Data Perekam Suara)

Pada perancangan alat ini selain menggunakan tampilan LCD kita juga menggunakan tampilan berupa suara yang telah kita rekam sebagai output sistem reminder. IC perekam suara yang kita gunakan adalah tipe ISD1420 (Information Storage Device) komponen rekaman/putar ulang suara berdurasi 20 detik dalam chip tunggal.

2.3.1 Gambaran umum ISD 1420

Komponen untuk menyimpan informasi ISD 1420, memberikan solusi rekaman atau putar ulang dalam chip tunggal untuk durasi penyimpanan 20 detik. Di dalam komponen CMOS-nya dilengkapi dengan chip osilator, pre-amplifier, microphone, kontrol gain otomatis, filter penghalus, dan amplifier speaker. Pada IC ini kompatibel penuh dengan mikrokontroler sehingga pesan-pesan alamatnya yang kompleks bisa dicapai. Perekaman disimpan dalam sel memori non-folatif sehingga menghasilkan penyimpanan pesan tanpa daya (Zero Power). Sinyal suara dan audio disimpan secara langsung dalam bentuk analog seperti aslinya di dalam memori. Komponen ini juga mempunyai penyetabil suhu sendiri dengan menggunakan osilator basis waktu. Komponen IC ISD 1420 beroperasi pada catu daya bila konsumsi daya minimum setelah mencapai tahap kritis. Komponen ini juga dapat dihubungkan dengan rangkaian antar muka dengan pengalamatan pesan pada alamat A0-A7 dan kontrol pesan untuk mode pengalamatan data rekam dan putar ulang..



Gambar 2.9. ISD 1420 [9]

Gambaran mengenai fungsi masing-masing pin pada IC ISD 1420 sebagai berikut :

- **Record ($\overline{\text{REC}}$)**

  Input REC mempunyai signal rekaman aktif rendah (LOW). Pin ini akan merekam pada saat REC aktif LOW selama durasi perekaman REC. harus didahulukan dari pada sinyal playback (baik PLAYE dan PLAYL). Jika REC ditarik LOW selama putaran palyback (tidak LOW), play back segera berhenti dan perekaman mulai. Putaran rekaman selesai daat REC ditarik HIGH. Penanda berakhirnyta pesan direkam di dalam, memungkinkan putaran playback berikutnya untuk mengakhiri dengan tepat. Secara otomatis power down menjadi mode standby.

- **Playback Edge Activated ($\overline{\text{PLAYE}}$)**

  Ketika transmisi ke LOW terdeteksi pada signal ini, mulailah putaran playback. Playback berlangsung secara terus menerus sampai penunda akhir pesan bertemu atau jarak akhir memori dicapai. Saat selesainya putaran playback, secara otomatis power down menjadi mode standby. Pengambilan PLAYE yang HIGH selama putaran playback tidak akan menghentikan putaran arus.

- **Playback Level Activated ($\overline{\text{PLAYL}}$)**

  Pada saat sinyal input bertransisi dari HIGH ke LOW inilah putaran playback berawal. Playback berlangsung terus sampai PLAYL ditarik HIGH, terdeteksinya penanda berakhirnya pesan atau jarak atau ruang dicapai secara otomatis power down menjadi mode standby tanda terselesainya putaran playback.

  ✓ Catatan : dalam playback, baik PLAYL atau PLAYE jika dipertahankan LOW selama EOM atau OVERFLOW. Alat akan tetap dalam keadaan stadby dan oscilator internal dan penghasil waktu akan berhenti. Bagaimanapun naiknya PLAYL dan PLAYE secara perlahan tidak lagi didebounce dan secara perlahan lahan yang ada pada pin input akan mengawali playback lainnya.

- **Record LED Output (RECLED)**

  Keluaran RECLED LOW selama putaran rekaman, bisa digunakan untuk membuat LED menhasilkan saklar arus bolak-balik, selama putaran rekaman dalam proses. Untuk catatan, RECLED kemudian berubah LOW saat penanda akhir pesan bertemu dalam putaran playback.

- **Microphone inpeet (MIC)**

  Microphone eksternal dikopling AC ke pin ini menggunakan kapasitor seri . harga input kapasitor seri dapat dipilih bersama dengan tahanan dalam sebesar 10 K ohm, menentukan besar frekuensi cut-off rendah untuk band frekuensi input IC ISD 1420.

- **Microphone Referensi Input (MICREF)**

  Dengan menggunakan pin ini ke V_{SSA} (ground analog) lewat sebuah kapasitor seri noise dapat dicegah pada pre-amplifier. Harga kapasitor sama dengan kapasitor kopling yang digunakan pendekatan ini akan mengurangi noise sebesar 10dB.

- **Output Analog (ANA OUT)**

  Pin ini menyediakan output pre-amplifier. Gain pre-amplifier ini ditentukan oleh tingkat tegangan pada pin AGC. ANA OUT inin memiliki gain maksimum sekitar 24 dB untuk tingkat sinyal yang kecil.

- **Input Analaog (ANA IN)**

  Pin input analog memindahkan sinyalnya ke chip untuk direkam. Untuk inpiut microphone, pin ANA OUT dihubungkan lewat kapasitor eksternal pada pin ANA IN. Harga kapsitor ini bersama-sama dengan impedansi input 3 KOhm dan ANA IN, dapat dipilh untuk memberikan cut off tambahan pada frekuensi rendah dari band frekuensi suara.

- **Automatic Gain Control (AGC)**

  AGC secara dinamis mengatur gain pre-amplifier untuk meghasilkan jangkauan yang lebar pada tingkat michrophone. AGC dapat merubah suara yang lemah menjadi lebih kuat untuk dapat direkam untuk distorsi minimal.

Waktu pengaktifan ditentukan oleh konstanta waktu tahanan dalam sebesar 5 KOhm dan kapasitorluar (C2) yang dihubungkan dari pin AGC ke ground analaog V_{SSA}. Harga nominal 470 KOhm dan 4,7 μF untuk memberikan hasil yang ideal. Untuk tegangan AGC 15 V dan dibawahnya. Pre-Amplifier akan berkurang apabila tegangannya sekitar 1,8 V.

- **Output Speaker (SP+/SP-)**

  Pin SP+ dan SP- memeberikan saluran pergerakan langsung ke loudspeaker dengan impedansi serendah 16 ohm. Akan tetapi satu output tunggal dapat digunakan untuk memberikan peningkatan daya empat kali lipat dinbandingkan dengan satu output saja. Untuk satu output saja diperlukan penghubng kapasitor kopling AC antara pin SP dan speaker. Bila hubungan kaki kai SP+ dan SP- digunakan kapasitor, maka kopling tidak diperlukan. Output speaker ditahan pada V_{SSA} selama perekaman dan pemadaman daya.

- **Operasional Eksternal Clock (XCLK)**

  Signal ini diikat ke dasar dalam rangkaian aplikasi, jika pemilihan ketelitian waktu yang lebih besar diinginkan (jam internal memiliki kurang lebih 25% toleransi terhadap temperatur dan jaringan voltase). Jika XCLK tidak digunakan input ini harus dihubungkan ke ground.

- **Input Tegangan (V_{CCA} dan V_{CCD})**

  Untuk meminimalkan noise, rangkaian analog dan digital dalam ISD 1420 menggunakan bus bus data terpisah. Bus-bus +5V dialirkan pada kaki-kaki yang terpisah dan hendaknya dibuat sedekat mungkin dengan suplainya.

- **Input-Input ground (V_{SSA} dan V_{SSD})**

  Komponen ISD 1420 ini menggunakan bus-bus ground analog dan digital yang terpisah.

- **Input-Input Alamat ($A_0 - A_7$)**

  Dalam IC ISD 1420 input-input / mode memeberikan fungsi alamat pesan . untuk menentukan lokasi memeori dari IC ISD 1420 cukup dengan menambahkan dipswitch yang berisi 8 pin yang dihubungkan dengan kaki-kaki addres A_0-A_7. Sehingga dengan cara seperti ini maka ada sebanyak 2^8 = 256 alamat yang berbeda.

2.4. LCD (Liquid Crystal Display) M1632

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

M1632 mempunyai seperangkat perintah untuk mengatur tata kerjanya, perangkat perintah tersebut meliputi perintah untuk menghapus tampilan, meletakkan kembali cursor pada baris huruf pertama baris pertama,

menghidup/matikan tampilan dan lain sebagainya, semua itu dibahas secara terperinci dalam lembar data M1632.

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

1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya trasfer data antar M1632 dan mikrokontroler dilakukan dengan mode 8 bit
3. Tunggu selama 4.1 mili-detik
4. Kirimkan sekali lagi perintah 30h
5. Tunggu lagi selama 100 mikro-detik

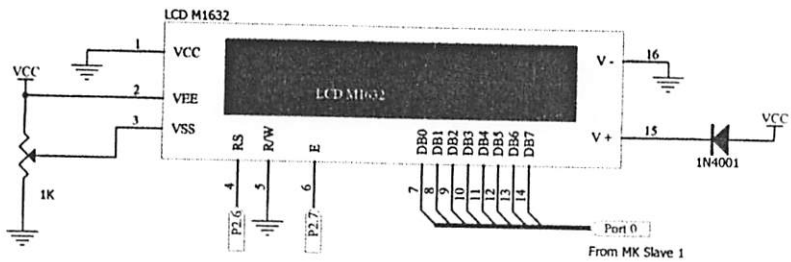
Setelah langkah-langkah tersebut di atas M1632 barulah bisa menerima data dan menampilkannya dengan baik. Pada awalnya tampilan akan nampak kacau, dengan demikian perlu segera dikirim perintah menghapus tampilan dan lain sebagainya, sesuai dengan petunjuk yang ada di Lembar Data.

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

M1632 mempunyai 8 jalur data dan memerlukan 3 jalur kontrol, dalam suatu rangkaian yang memakai banyak port dari MC-S51, bisa terjadi kekurangan

port untuk menghubungkan MCS51 ke M1632. Jika sampai terjadi hal semacam ini bisa ditempuh hal hal berikut :

1. M1632 dipakai dalam mode data 4 bit, yakni hanya memakai jalur data **D0..D3**
2. Dengan sedikit tambahan rangkaian sinyal **WR** dan **RD** diubah menjadi sinyal **E** dan **R/W** gaya Motorola, sehingga tidak perlu menyediakan port untuk men-simulasikan sinyal-sinyal tersebut. Berikut adalah gambar rangkaian LCD dengan komponen-komponen pendukung dengan pin-pin yang akan dihubungkan pada mikrokontroler MCS 51 :



Gambar 2.10. Rangkaian LCD M1632 [6]

LCD M1632 mempunyai spesifikasi sebagai berikut :

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

Tabel 2-4. Fungsi Pin-Pin LCD [6]

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

Pada LCD juga terdapat instruksi – instruksi sebagai berikut :

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

  S : 1/0 = menggeser layar.

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

- Display On/Off kontrol.

  D : 1 = layar on

  D : 0 = layar off

  C : 1 = kursor on

  C : 0 = kursor off

B : 1 = kursor berkedip-kedip

B : 0 = kursor tidak berkedip – kedip

- Cursor Display Shift

  S/C : 1 = LCD diidentifikasikan sebagai layar

  S/C : 0 = LCD diidentifikasikan sebagai kursor

  R/L : 1 = menggeser satu spasi ke kanan

  R/L : 0 = menggeser satu spasi ke kiri

- Fuction Set

  DL : 1 = panjang data LCD pada 8 bit

  DL : 0 = panjang data LCD pada 4 bit

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

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

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

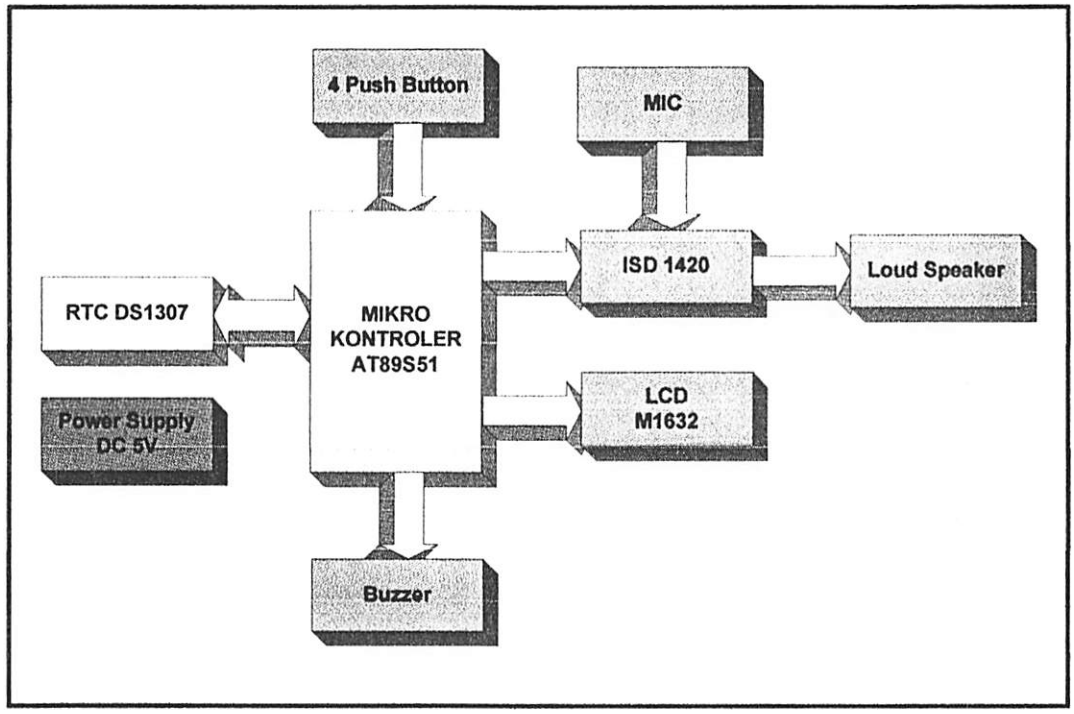
- CG RAM address set : menulis alamat RAM ke karakter
- DD RAM address set : menulis alamat RAM ke tampilan
- BF/address set : BF = 1/0, LCD dalam keadaan sibuk atau tidak sibuk.
- Data write to CG RAM or DD RAM : membaca byte dari alamat terakhir RAM yang dipilih.

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

3.1. Diagram Blok

Secara umum perancangan dan pembuatan Sistem Reminder Berbasis Mikrokontroler AT89S51 yang akan dibuat, digambarkan secara diagram blok sebagai berikut :



Gambar 3.1. Blok Diagram

- Mikrokontroler AT89S51 sebagai pengendali utama keseluruhan rangkaian.
- RTC DS1307 sebagai pewaktu yang menghitung berdasarkan tahun, tanggal, jam, menit dan detik.
- ISD 1420 digunakan untuk merekam suara sehingga dihasilkan output suara.
- Loud Speaker digunakan untuk mengeluarkan output suara.

- Mic digunakan untuk proses perekaman suara pada ISD.
- 4 Push-button digunakan sebagai input untuk seting sistem reminder, yaitu tombol menu, tombol up, tombol down, dan tombol reset.
- Buzzer digunakan sebagai alarm sistem reminder.
- LCD digunakan sebagai display.
- Power Supply digunakan untuk memberikan tegangan kerja yang dibutuhkan untuk keseluruhan rangkaian.

3.2. Prinsip Kerja Alat

1. User memasukkan input suara melalui mic.
2. User memasukkan seting aktif reminder berdasarkan tahun, tanggal, jam, menit dan detik yang diinginkan melalui push-button menu, up dan down.
3. Bila waktu sekarang telah menjadi sama dengan waktu yang diinputkan pada saat seting aktif reminder maka buzzer akan berbunyi dan LCD menampilkan ada pesan sistem reminder.
4. User menekan tombol push-button up untuk mematikan buzzer dan menampilkan pesan output suara melalui loud speaker.

3.3. Mikrokontroler AT89S51

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

1. *Data Bus*

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

2. *Address Bus*

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

3. *Control Bus*

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

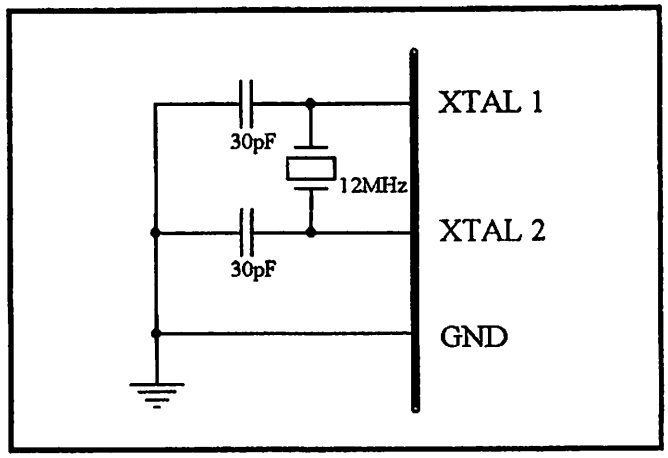
3.3.1. Pemetaan *Memori*

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

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

3.3.2. Rangkaian *Clock*

Mikrokontroler AT89S51 ini memiliki *internal clock*, yang berfungsi sebagai sumber *clock*, tapi masih diperlukan rangkaian tambahan untuk membangkitkan *clock* yang diperlukan. Rangkaian ini terdiri dari 2 buah kapasitor dan sebuah kristal dengan ketentuan seperti gambar berikut :



Gambar 3.2. Rangkaian *Clock* AT89S51

3.3.3. Rangkaian *Reset*

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

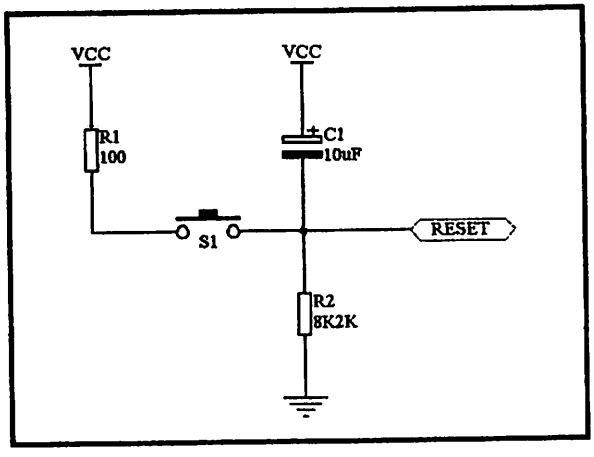
Jika saklar S1 ditekan , *reset* bekerja secara manual, aliran arus akan mengalir dari *VCC* melalui R1 menuju kaki RST. Tegangan di RST atau VR2 akan berubah menjadi :

$$VR2 = \frac{R2xVCC}{R1 + R2}$$

$$= \frac{8200x5}{100 + 8200}$$

$$= 4{,}94\,V$$

Saat saklar dilepas, aliran arus dari *VCC* melalui R1 akan berhenti dan tegangan pada kaki RST akan turun menuju ke nol dan proses *reset* selesai.



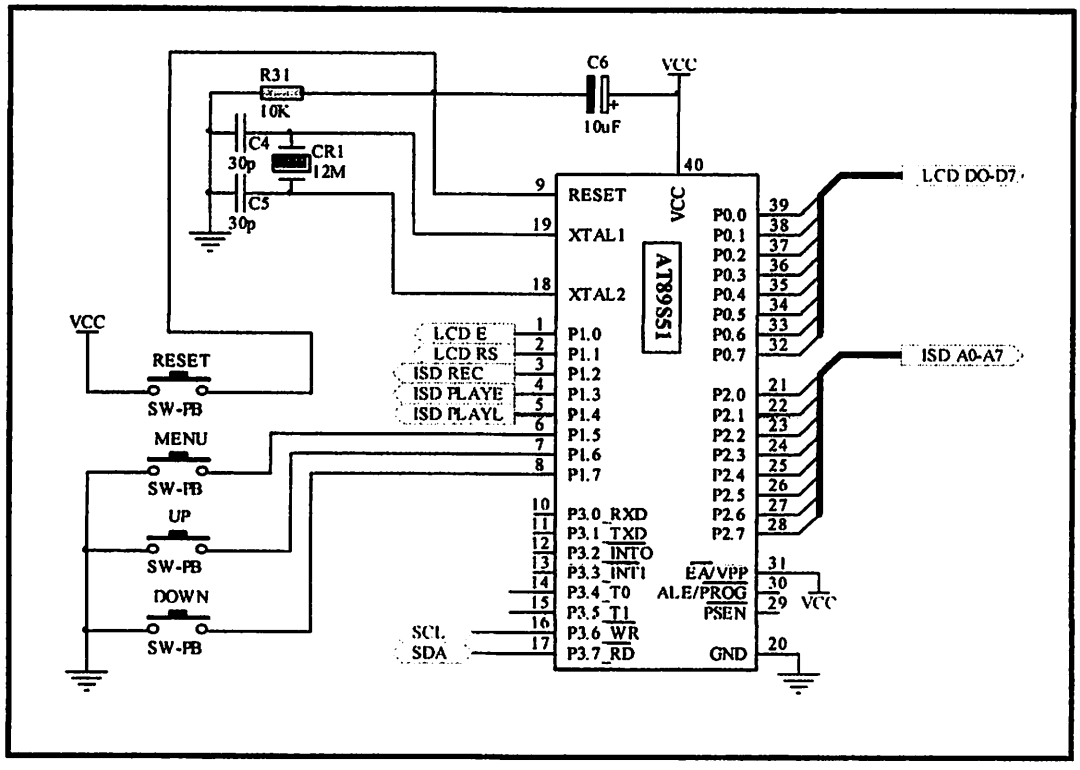
Gambar 3.3. Rangkaian *Reset*

3.3.4. Hubungan *Pin* Pada AT89S51

Diskripsi konfigurasi *Pin-Pin* AT89S51 pada rangkaian:

- *Pin* 1 dan 2 : dihubungkan ke *LCD* E dan RS
- *Pin* 3,4 dan 5 : dihubungkan ke rangkaian ISD REC, PLAYE, PLAYL
- *Pin* 6, 7 dan 8 : dihubungkan *Push-button Menu, Up,* dan *Down*
- *Pin* 9 : dihubungkan *Push-button reset*
- Pin 16 dan 17 : dihubungkan SCL dan SDA DS1307
- Pin 18 dan 19 : dihubungkan *X-tal* 12 MHz
- *Pin* 20 : dihubungkan ke GND

- *Pin* 32-39 : dihubungkan ke *LCD* D0-D7
- *Pin* 31 dan 40 : dihubungkan ke *VCC*



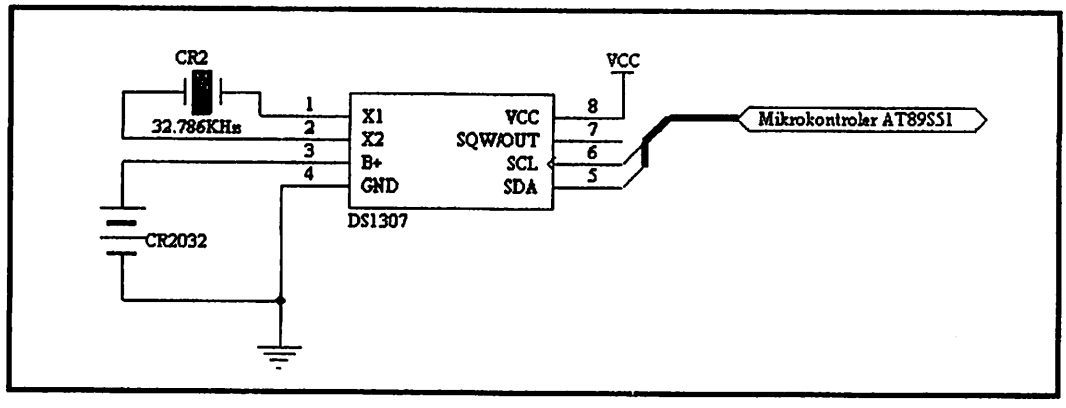
Gambar 3.4. Hubungan *Pin* Pada AT89S51

3.4. Rangkaian RTC DS1307

DS1307 adalah IC serial *Real Time Clock (RTC)* dimana alamat dan data ditransmisikan secara serial melalui sebuah jalur data dua arah. Karena menggunakan jalur data serial maka hanya memerlukan dua buah pin saja untuk komunikasi. Yaitu pin untuk data dan pin untuk sinyal clock. Dalam system yang kita rancang ini RTC DS1307 difungsikan sebagai pewaktu, yaitu peripheral yang menyediakan data detik, menit,jam, hari, tanggal, bulan dan tahun. Data waktu ini nantinya akan diolah oleh mikrokontroler dan ditampilkan pada LCD.

Pemilihan penggunaan serial RTC DS1307 pada perancangan ini adalah dikarenakan RTC DS1307 mempunyai beberapa kelebihan sebagai berikut::

- Harga yang relative lebih murah jika dibandingakan dengan RTC lainnya
- Karena akses data dilakukan secara serial, maka hanya butuh 2 pin saja, sehingga akan menghemat port mikrokontroler
- Tidak perlu lagi ada perbaikan penanggalan untuk penggunaan ditahun 2000 ke atas
- Sudah tersedia pin untuk baterai back-up, sehingga tidak perlu lagi dibuatkan rangkaian untuk baterai back-up, apabila kehilangan supply tegangan

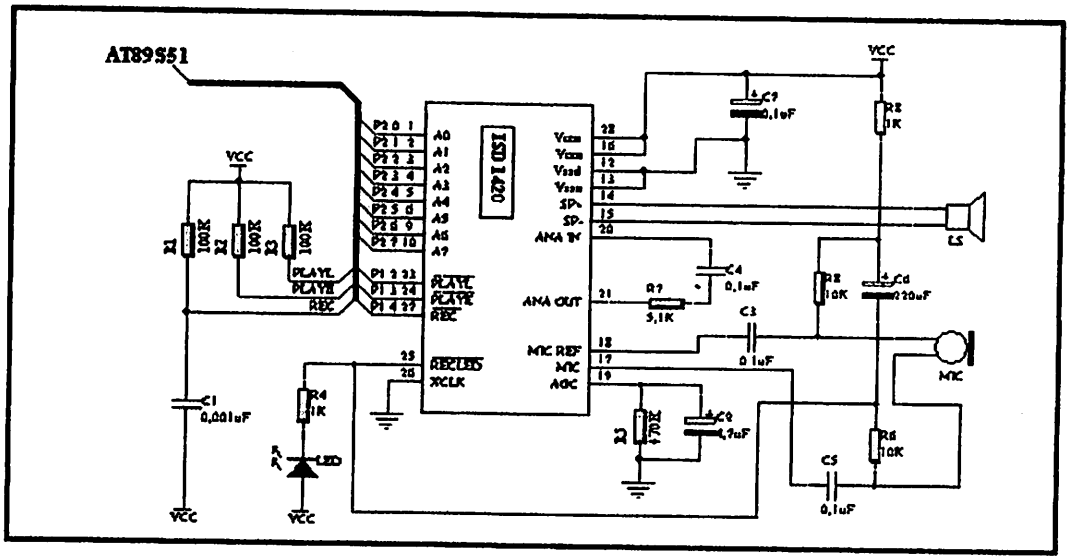


Gambar 3.5. Rangkaian Serial RTC DS1307

Baterai back-up yang digunakan adalah baterai back-up 3V CR2032, yang dapat bertahan untuk masa operasi 10 tahun, kondisi ini amat hemat biaya. Kristal yang digunakan adalah standart quartz kristal dengan dinilai 32,768KHz. Pin SDA dan SCL dari RTC serial DS1307 ini dihubungkan ke port mikrokontroler AT89S51 P3.6 dan P3.7.

3.5. Rangkaian Pemutar Suara ISD1420

Untuk menghasilkan output suara yang nantinya akan dijadikan sebagai output sistem reminder maka penulis menggunakan IC perekam/pemutar suara ISD 1420. Rangkaian antarmuka ISD1420 dengan mikrokontroler AT89S51 untuk memutar suara ditunjukkan dalam Gambar 3.6. Pin alamat A_0 sampai A_7 ISD1420 di hubungkan dengan port 2 mikrokontroler. Untuk memutar suara maka mikrokontroler mengirimkan data lokasi alamat tempat penyimpanan suara informasi tersebut ke ISD1420 dengan mengirimkan logika rendah pada pin $\overline{\text{PLAYE}}$ atau pin $\overline{\text{PLAYL}}$. Saat *playback cycle*, alamat input menetapkan mulainya alamat dan memutar suara secara kontinyu sampai alamat $\overline{\text{EOM}}$ (*End Of Message*) ditemukan ($\overline{\text{EOM}}$ berlogika rendah) dan ditandai dengan keluaran dari *Record Led Output* ($\overline{\text{RECLED}}$) yang akan berlogika rendah ketika operasi *playback cycle* berakhir atau adanya transisi tinggi dari $\overline{\text{PLAYL}}$ (apabila digunakan pin $\overline{\text{PLAYL}}$ untuk pemutaran suara).



Gambar 3.6 Rangkaian Pemutar Suara ISD1420

Sebagaimana diketahui durasi penyimpanan IC ini selama 20 detik, maka kata-kata yang akan disimpan tidak boleh melebihi dari durasi yang diijinkan, Tabel 3.1 memperlihatkan pemilihan alamat yang digunakan untuk merekam suara. ISD1420 memiliki 8 bit jalur alamat atau memiliki lokasi alamat sampai 256 bit. Jadi tiap detiknya membutuhkan jumlah alamat sebanyak

$$\text{Jumlah alamat perdetik} = \frac{256\,\text{bit}}{20\,\text{s}} = 12.8\,\text{bit}$$

Sehingga untuk lebih amannya dalam penyimpanan kata-kata, perdetiknya digunakan jumlah alamat yang dibutuhkan sebanyak 16 bit agar kata yang akan direkam tidak masuk ke lokasi alamat berikutnya.

3.6. Perancangan Rangkaian *LCD*

LCD Display Module M1632 buatan *Seiko Instrument Inc.* terdiri dari dua bagian, yang pertama merupakan panel *LCD* sebagai media penampil informasi dalam bentuk huruf/angka dua baris, masing-masing baris bisa menampung 16 huruf/angka. Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibalik pada panel *LCD*, berfungsi mengatur tampilan informasi serta berfungsi mengatur komunikasi M1632 dengan mikrokontroler yang memakai tampilan *LCD* itu..

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

Motorola. Sebaliknya sinyal-sinyal dari MCS51 merupakan sinyal khas *Intel* dengan kombinasi sinyal **WR** dan **RD.**

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

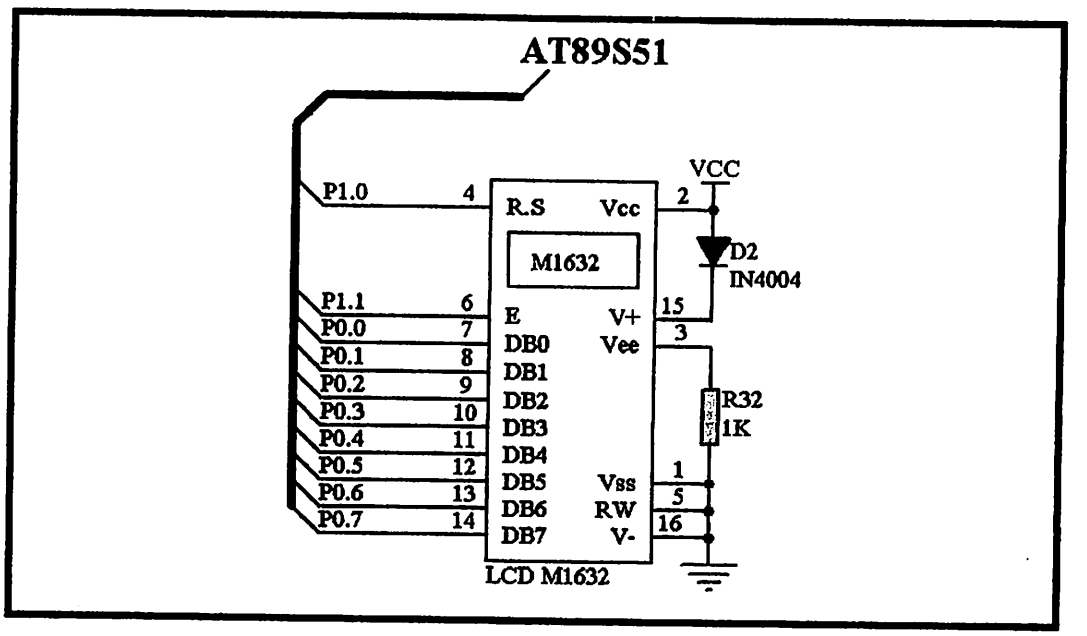
1. **RS** harus dipersiapkan dulu, untuk menentukan jenis data seperti yang telah dibicarakan di atas.
2. **R/W** di-nol-kan untuk menandakan akan diadakan pengiriman data ke M1632. Data yang akan dikirim disiapkan di **DB0..DB7**, sesaat kemudian sinyal **E** di-satu-kan dan di-nol-kan kembali. Sinyal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

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

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

1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya trasfer data antar M1632 dan mikrokontroler dilakukan dengan mode 8 *bit*
3. Tunggu selama 4.1 mili-detik
4. Kirimkan sekali lagi perintah 30h
5. Tunggu lagi selama 100 mikro-detik

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



Gambar 3.7. Rangkaian *LCD* Pada Mikrokontroler

Diskripsi konfigurasi *Pin-Pin* pada *LCD*:

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

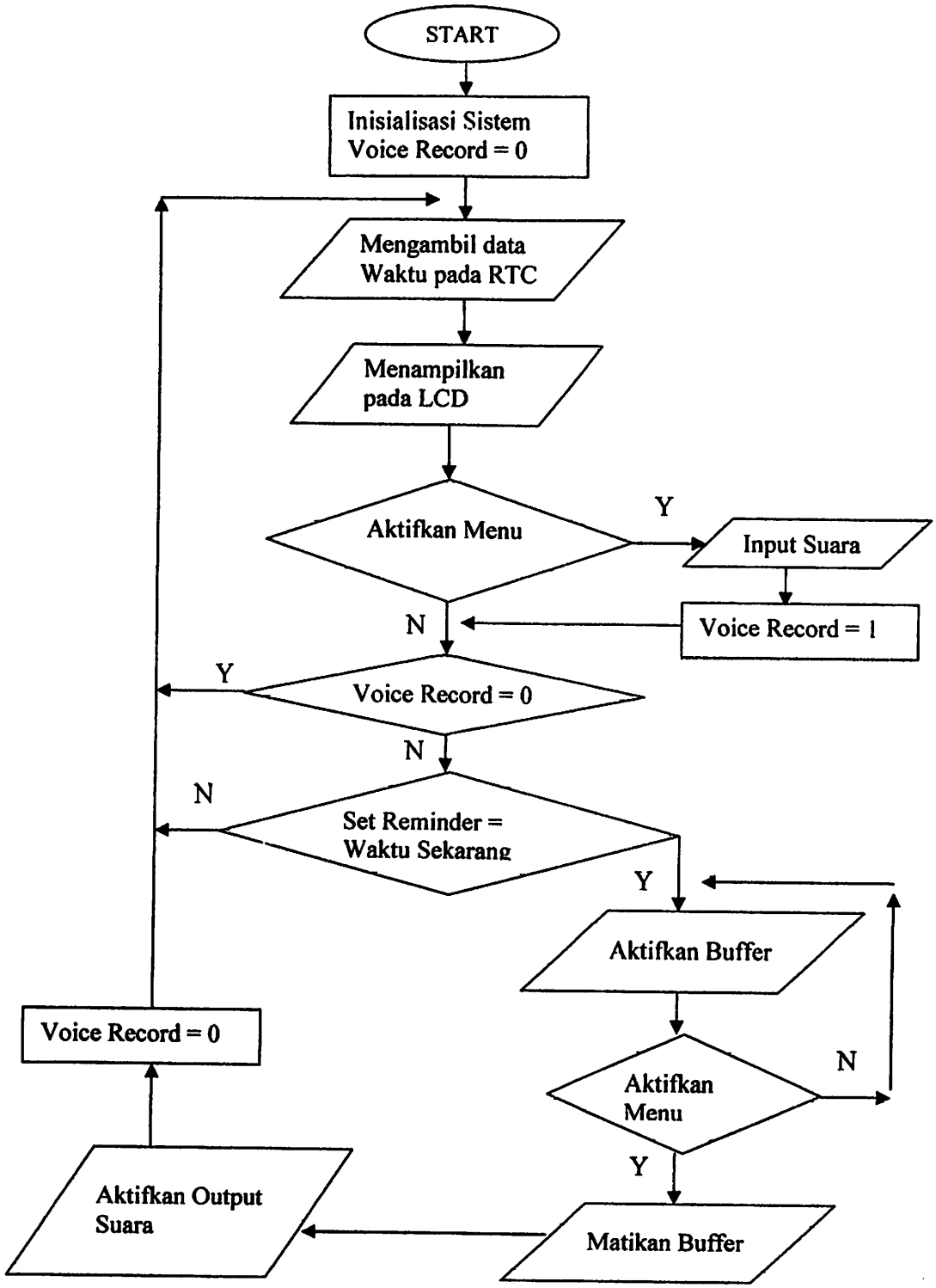
Untuk tampilan dipergunakan *LCD* Dot Matrik 2 x 16 karakter. Sinyal-sinyal yang diperlukan oleh *LCD* adalah RS dan *Enable*, sinyal RS dan *Enable* dipergunakan sebagai *input* yang outputnya dipakai untuk mengaktifkan *LCD*. *LCD* akan aktif apabila mikrokontroler memberikan instruksi tulis pada LCD. Saat kondisi RS *don't care* dan *Enable* 0 maka *LCD* tetap pada kondisi semula, pengiriman data ke *LCD* dilakukan saat RS berlogika 0 dan *enable* berlogika 1.

Instruksi dikirim pada *LCD* bila keadaan RS 1 dan *Enable* 1. Pin *LCD* ini untuk data terkoneksi pada *Port 0* mikrokontoler AT89S51. Kemudian untuk RS dihubungkan pada *Port* 1.0, tulis/baca (*Read/Write)* diberikan logika *low* karena disini LCD bersifat menulis data, dan yang terakhir *Enable* (E) dikendalikan dengan *Port 1.1*.

3.7. Perangkat Lunak *(software)*

Perencanaan perangkat lunak sangat diperlukan untuk menjalankan *sistem* seperti yang diinginkan. Mikrokontroler AT89S51 tidak akan bisa dijalankan tanpa adanya *software*. Didalam perencanaan perangkat lunak ini di buat perintah-perintah yang berfungsi untuk mengatur urutan dan tata kerja dari keseluruhan sistem.

Flowchart



Gambar 3.8. *Flowchart* Keseluruhan Alat

BAB IV

PENGUJIAN ALAT

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

1. Mengetahui sejauh mana Sistem Reminder Berbasis Mikrokontroler AT89S51 ini berfungsi sebagaimana yang kita harapkan.
2. Mengetahui kualitas suara serta ketepatan waktu yang di hasilkan oleh Sistem Reminder Berbasis Mikrokontroler AT89S51.
3. Mencari dan menemukan berbagai kendala yang mungkin timbul pada saat Sistem Reminder Berbasis Mikrokontroler AT89S51 beroperasi untuk kemudian diperbaiki sampai pada tingkat kesalahan sistem yang sekecil mungkin sehingga didapatkan hasil yang sebaik-baiknya.

4.1. Pengujian Mikrokontroler AT89S51

Pengujian ini bertujuan untuk mengetahui apakah minimum sistem dan port-port pada mikrokontroler yang digunakan dapat berjalan dengan baik.

❖ **Peralatan yang Digunakan**

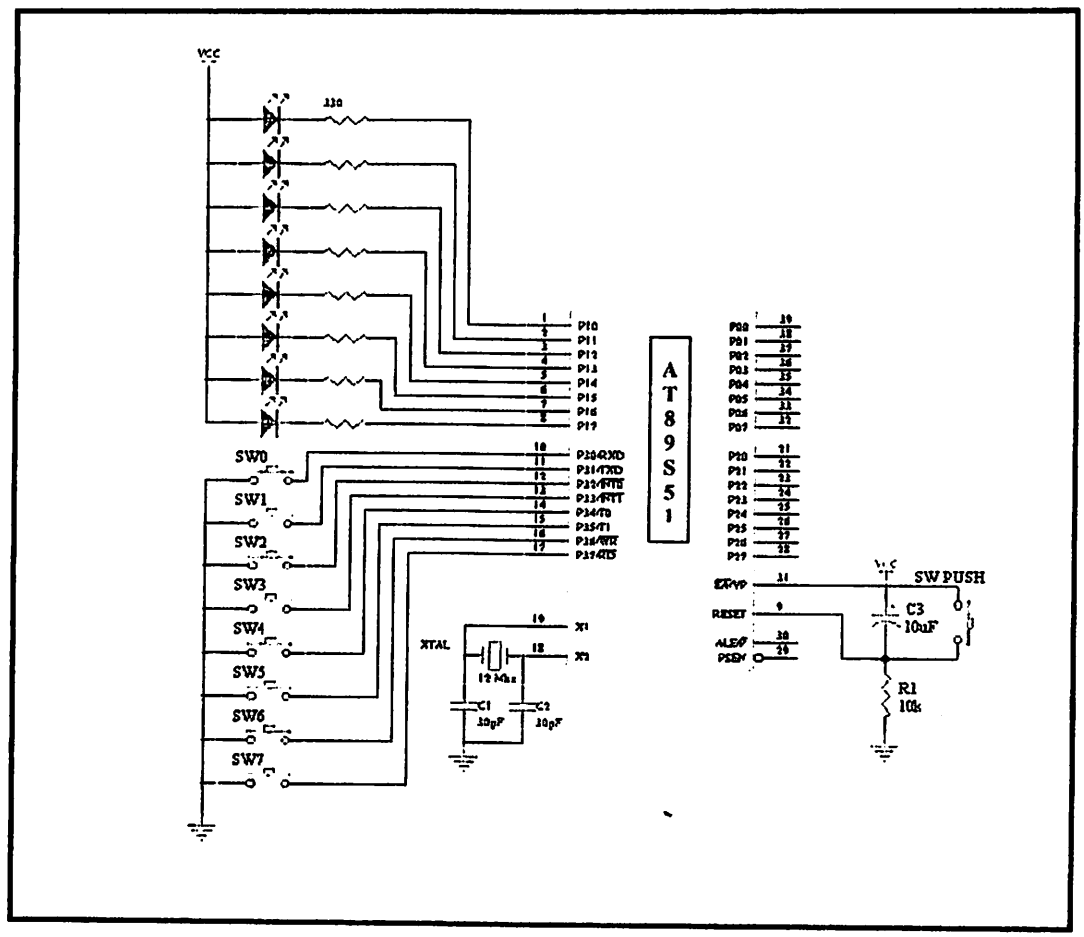
1. Mikrokontroler AT89S51
2. LED
3. Mikrokontroler *writer*

❖ **Prosedur Pengujian**

1. Merancang rangkaian seperti dalam Gambar 4.1

2. Menguji dengan program

3. Mencatat hasil keluaran pada tabel 4.1



Gambar 4.1. Rangkaian Pengujian Mikrokontroler dan Sistem Minimum

Listing Program :

```
        org 0h
mulai:  jnb p3.0,nol
        jnb p3.1,satu
        jnb p3.2,dua
        jnb p3.3,tiga
        jnb p3.4,empat
```

```
        jnb p3.5,lima
        jnb p3.6,enam
        jnb p3.7,tujuh
        jmp mulai
nol:    mov p1,#11111111b
        jmp mulai
satu:   mov p1,#11111110b
        jmp mulai
dua:    mov p1,#11111101b
        jmp mulai
tiga:   mov p1,#11111100b
        jmp mulai
empat:  mov p1,#11111011b
        jmp mulai
lima:   mov p1,#11111010b
        jmp mulai
enam:   mov p1,#11111001b
        jmp mulai
tujuh:  mov p1,#11111000b
        jmp mulai
end
```

BAB V

PENUTUP

5.1. Kesimpulan

Pada bab ini akan dibahas mengenai kesimpulan dari hasil laporan Tugas Akhir ini. Kesimpulan yang dibuat tentu saja berdasarkan dari hasil perencanaan dan pembuatan alat seperti yang dibahas pada bab-bab sebelumnya. Kesimpulan yang didapat adalah sebagai berikut:

1. Alat ini sangat mudah dalam pengoperasiannya dilengkapi dengan tombol menu untuk seting alat, LCD sebagai display, dengan output suara.
2. Dengan menggunakan RTC maka hanya membutuhkan 2 jalur untuk interface komunikasi data I2C yaitu SDA *(serial data)* dan SCA *(serial clock)* sehingga dapat menghemat port pada Mikrokontroler AT89S51.
3. Dikarenakan sistem reminder yang telah dibuat dilengkapi dengan output suara menggunakan IC perekam suara ISD 1420, sehingga batas durasi maksimum outputan suara yang dapat direkam pada sistem ini adalah 20 detik.
4. Output suara yang dihasilkan oleh sistem reminder mempunyai kualitas dan ketepatan waktu yang cukup baik.

5.2. Saran

Diharapkan alat ini dapat dikembangkan dengan menambahkan ISD yang memiliki kemampuan dapat menyimpan suara dalam durasi yang lebih panjang sehingga dapat menampung *output*-an suara yang lebih banyak.

DAFTAR PUSTAKA

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

Listing Program Assembler

```
    ORG   000H
    JMP   START

;===============
;LCD KONSTANTA
;===============
DISPCLR EQU 00000001B
FUNCSET EQU 00111000B
ENTRMOD EQU 00000110B
DISPON  EQU 00001100B
;==========
;PORT LCD
;==========
LCDE   BIT   P1.0
LCDRS  BIT   P1.1
PLCD   EQU   P0
BUZZ   BIT   P3.6
;=============
;RTC ADDRES
;=============
SCL   BIT   P3.5
SDA   BIT   P3.4
;==============
;==========
;PORT ISD
;==========
ISD_REC        BIT   P1.2
ISD_PLAYE      BIT   P1.3
ISD_PLAYL      BIT   P1.4
;====================
;PUSH BUTTON ADDRES
;====================
TBDOWN BIT   P1.7
TBUP   BIT   P1.6
TBMENU BIT   P1.5
;==============
;=====================================
;ALAMAT VARIABLE
;=====================================
PUTR       EQU   030H
RTC_DAT    EQU   031H
RTC_ADR    EQU   032H
SEC        EQU   033H
MIN        EQU   034H
HOUR       EQU   035H
DATE       EQU   036H
MONTH      EQU   037H
YEAR       EQU   038H
JMLHR      EQU   039H
```

```
TEMP        EQU   03AH
MIN_P       EQU   03BH
HOUR_P      EQU   03CH
DATE_P      EQU   03DH
MONTH_P     EQU   03EH
YEAR_P      EQU   03FH
M           EQU   040H
R           EQU   041H
;============
;MAIN PROGRAM
;============
START:
        SETB   BUZZ
    LCALL  INITLCD
        LCALL  IKLAN
        MOV    M,#0

START_A:
        CALL   DIS_BACK
        CALL   CEK_REMINDER
        JB     TBMENU,START_A
        JNB    TBMENU,$
        MOV    A,#DISPCLR
    CALL   LCDINS
        CALL   PILIH
        MOV    A,#DISPCLR
    CALL   LCDINS
        JMP    START_A
PILIH:
        CALL   DSP_PILIH
        JB     TBMENU,PILIH2
        JNB    TBMENU,$
        MOV    A,#DISPCLR
    CALL   LCDINS
        CALL   SET_REMINDER        ;SET REMINDER
        RET
PILIH2:
        JB     TBUP,PILIH
        JNB    TBUP,$
        MOV    A,#DISPCLR
    CALL   LCDINS
    CALL       DIS_BACK
        CALL   SET_WAKTU           ;SET WAKTU
        RET
;============================
CEK_REMINDER:
;============================
        MOV    A,M
        JZ     OUT_C
        MOV    A,YEAR
        CJNE   A,YEAR_P,OUT_C
```

```
        MOV   A,MONTH
        CJNE  A,MONTH_P,OUT_C
        MOV   A,DATE
        CJNE  A,DATE_P,OUT_C
        MOV   A,HOUR
        CJNE  A,HOUR_P,OUT_C
        MOV   A,MIN
        CJNE  A,MIN_P,OUT_C
        MOV   A,#DISPCLR
    CALL  LCDINS
    CALL      DSP_PESAN
BUNYI:
        CLR   BUZZ
        JB    TBMENU,BUNYI
        JNB   TBMENU,$
        SETB  BUZZ
        CALL  OUT_SUARA
        MOV   M,#0
        MOV   A,#DISPCLR
    CALL  LCDINS
        RET

OUT_C:
        RET
;==============================
IN_SUARA:
;==============================
        CALL  DSP_2
        JB    TBMENU,IN_SUARA
        JNB   TBMENU,$
        MOV   A,#DISPCLR
    CALL  LCDINS
        CALL  INPUT_SUARA
        RET
;==============================
INPUT_SUARA:
;==============================
        CALL  DSP_STOP_REC
        CLR   ISD_REC
        JB    TBMENU,INPUT_SUARA
        JNB   TBMENU,$
        SETB  ISD_REC
        RET
;==============================
OUT_SUARA:
;==============================
        CLR   ISD_PLAYE
        CALL  DELAY
        SETB  ISD_PLAYE
        CALL  DELAY
        RET
```

```
;============================
DSP_REMINDER:
;=============
        MOV   A,#DISPCLR
    CALL  LCDINS
        CALL    DIS_R
        RET
;====================================
SET_REMINDER:
        CALL    DSP_SET_R
        JB      TBMENU,HAPUS
        JNB     TBMENU,$
        CALL    MASUK
        CALL    IN_SUARA
        RET

HAPUS:
        JB      TBUP,SET_REMINDER
        JNB     TBUP,$
        MOV     M,#0
        CLR     ISD_REC
        CALL    DELAY
        SETB    ISD_REC
        RET
;==========================
MASUK:
        CALL    DSP_REMINDER
        MOV     A,#0C7H
        CALL    LCDDAT
        MOV     A,#00001101B
    CALL  LCDINS
    JMP   SJ4
            ;<-- SET TANGGAL
SJ0:
    MOV   A,MONTH
    CALL  C_JML_HR
    MOV   A,#087H
    CALL  LCDINS
    MOV   A,DATE
    MOV   B,#010
    DIV   AB
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,B
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,#088H
    CALL  LCDINS
SJ1:
    JB    TBUP,SJ2
    JNB   TBUP,$
```

```
    MOV   A,DATE
    CJNE  A,JMLHR,SJ1_1
    MOV   DATE,#01
    JMP   SJ0
SJ1_1:
    INC   DATE
    JMP   SJ0
SJ2:
    JB    TBDOWN,SJ3
    JNB   TBDOWN,$
    MOV   A,DATE
    CJNE  A,#01,SJ2_1
    MOV   DATE,JMLHR
    JMP   SJ0
SJ2_1:
    DEC   DATE
    JMP   SJ0
SJ3:
    JB    TBMENU,SJ1
    JNB   TBMENU,$
    MOV   DATE_P,DATE
    JMP   SJ8

          ;<-- SET BULAN
SJ4:
    MOV   A,#08AH
    CALL  LCDINS
    MOV   A,MONTH
    CALL  CARI_BULAN
    CALL  LCDSTRING
    MOV   A,#08CH
    CALL  LCDINS
    CALL      DELAY
    CALL      DELAY
    CALL      DELAY
    CALL      DELAY
    CALL      DELAY
    CALL      DELAY
SJ5:
    JB    TBUP,SJ6
    JNB   TBUP,$
    CALL      DELAY
    CALL      DELAY
    CALL      DELAY
    MOV   A,MONTH
    CJNE  A,#012,SJ5_1
    MOV   MONTH,#01
    JMP   SJ4
SJ5_1:
    INC   MONTH
    JMP   SJ4
```

```
SJ6:
    JB    TBDOWN,SJ7
    JNB   TBDOWN,$
    MOV   A,MONTH
    CJNE  A,#01,SJ6_1
    MOV   MONTH,#012
    JMP   SJ4
SJ6_1:
    DEC   MONTH
    JMP   SJ4
SJ7:
    JB    TBMENU,SJ5
    JNB   TBMENU,$
    MOV   MONTH_P,MONTH
    JMP   SJ0

          ;<-- SET TAHUN
SJ8:
    MOV   A,#08EH
    CALL  LCDINS
    MOV   A,YEAR
    MOV   B,#010
    DIV   AB
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,B
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,#08FH
    CALL  LCDINS
SJ9:
    JB    TBUP,SJ10
    JNB   TBUP,$
    MOV   A,YEAR
    CJNE  A,#099,SJ9_1
    MOV   YEAR,#0
    JMP   SJ8
SJ9_1:
    INC   YEAR
    JMP   SJ8
SJ10:
    JB    TBDOWN,SJ11
    JNB   TBDOWN,$
    MOV   A,YEAR
    JNZ   SJ10_1
    MOV   YEAR,#099
    JMP   SJ8
SJ10_1:
    DEC   YEAR
    JMP   SJ8
SJ11:
```

```
        JB    TBMENU,SJ9
        JNB   TBMENU,$
        MOV   YEAR_P,YEAR

                ;<-- SET JAM
SJ12:
        MOV   A,#0C8H
        CALL  LCDINS
        MOV   A,HOUR
        MOV   B,#010
        DIV   AB
        ADD   A,#030H
        CALL  LCDDAT
        MOV   A,B
        ADD   A,#030H
        CALL  LCDDAT
        MOV   A,#0C9H
        CALL  LCDINS
SJ13:
        JB    TBUP,SJ14
        JNB   TBUP,$
        MOV   A,HOUR
        CJNE  A,#023,SJ13_1
        MOV   HOUR,#0
        JMP   SJ12
SJ13_1:
        INC   HOUR
        JMP   SJ12
SJ14:
        JB    TBDOWN,SJ15
        JNB   TBDOWN,$
        MOV   A,HOUR
        JNZ   SJ14_1
        MOV   HOUR,#023
        JMP   SJ12
SJ14_1:
        DEC   HOUR
        JMP   SJ12
SJ15:
        JB    TBMENU,SJ13
        JNB   TBMENU,$
        MOV   HOUR_P,HOUR
                ;<-- SET MENIT
SJ16:
        MOV   A,#0CBH
        CALL  LCDINS
        MOV   A,MIN
        MOV   B,#010
        DIV   AB
        ADD   A,#030H
        CALL  LCDDAT
```

```
        MOV    A,B
        ADD    A,#030H
        CALL   LCDDAT
        MOV    A,#0CCH
        CALL   LCDINS
SJ17:
        JB     TBUP,SJ18
        JNB    TBUP,$
        MOV    A,MIN
        CJNE   A,#059,SJ17_1
        MOV    MIN,#0
        JMP    SJ16
SJ17_1:
        INC    MIN
        JMP    SJ16
SJ18:
        JB     TBDOWN,SJ19
        JNB    TBDOWN,$
        MOV    A,MIN
        JNZ    SJ18_1
        MOV    MIN,#059
        JMP    SJ16
SJ18_1:
        DEC    MIN
        JMP    SJ16
SJ19:
        JB     TBMENU,SJ17
        JNB    TBMENU,$
        MOV    MIN_P,MIN
        MOV        M,#1
        CALL       DISPLON
        RET

;=========================
DIS_BACK:
;=========
        MOV    A,#081H
        CALL   LCDINS
        MOV        A,#'T'
        CALL       LCDDAT
        MOV        A,#'G'
        CALL       LCDDAT
        MOV        A,#'L'
        CALL       LCDDAT
        MOV    A,#0C1H
        CALL   LCDINS
        MOV        A,#'J'
        CALL       LCDDAT
        MOV        A,#'A'
        CALL       LCDDAT
        MOV        A,#'M'
```

```
    CALL    LCDDAT
DIS_R:
    MOV     A,#89H
    CALL    LCDINS
    MOV     A,#'-'
    CALL    LCDDAT
    MOV     A,#8DH
    CALL    LCDINS
    MOV     A,#'-'
    CALL    LCDDAT
    MOV     A,#0CAH
    CALL    LCDINS
    MOV     A,#':'
    CALL    LCDDAT
    MOV     A,#0CDH
    CALL    LCDINS
    MOV     A,#':'
    CALL    LCDDAT
    MOV     A,#085H
    CALL    LCDINS
    MOV     A,#':'
    CALL    LCDDAT
    MOV     A,#0C5H
    CALL    LCDINS
    MOV     A,#':'
    CALL    LCDDAT

    MOV   RTC_ADR,#04
    CALL  GET_RTC
    CALL  PINDAH_DATA_RTC
    MOV   A,#087H
    CALL  LCDINS
    MOV   A,DATE
    MOV   B,#010
    DIV   AB
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,B
    ADD   A,#030H
    CALL      LCDDAT
    MOV   RTC_ADR,#05
    CALL      GET_RTC
    CALL  PINDAH_DATA_RTC
    MOV   A,#08AH
    CALL      LCDINS
    MOV   A,MONTH
    CALL  CARI_BULAN
    CALL  LCDSTRING
    MOV   RTC_ADR,#06
    CALL  GET_RTC
    CALL  PINDAH_DATA_RTC
```

```
MOV   A,#08EH
CALL  LCDINS
MOV   A,YEAR
MOV   B,#010
DIV   AB
ADD   A,#030H
CALL  LCDDAT
MOV   A,B
ADD   A,#030H
CALL  LCDDAT
MOV   RTC_ADR,#02
   CALL  GET_RTC
CALL  PINDAH_DATA_RTC
MOV   A,#0C8H
CALL  LCDINS
MOV   A,HOUR
MOV   B,#010
DIV   AB
ADD   A,#030H
CALL  LCDDAT
MOV   A,B
ADD   A,#030H
CALL  LCDDAT
MOV   RTC_ADR,#01
CALL        GET_RTC
CALL        PINDAH_DATA_RTC
MOV   A,#0CBH
CALL  LCDINS
MOV   A,MIN
MOV   B,#010
DIV   AB
ADD   A,#030H
CALL  LCDDAT
MOV   A,B
ADD   A,#030H
CALL  LCDDAT
MOV   RTC_ADR,#0
CALL  GET_RTC
CALL  PINDAH_DATA_RTC
MOV   A,#0CEH
CALL  LCDINS
MOV   A,SEC
MOV   B,#010
DIV   AB
ADD   A,#030H
CALL  LCDDAT
MOV   A,B
ADD   A,#030H
CALL  LCDDAT
RET
;==============
```

```
;============================
GET_RTC:
;========
    CLR   EA
    ACALL  SIGNAL_START
    JC    ERROR_GET
       MOV   A,#11010000B  ;==> ADRESS RTC DAN PROSES WRITE
       ACALL  OUT8BIT
    JC    ERROR_GET
    MOV   A,RTC_ADR
       ACALL  OUT8BIT
    JC    ERROR_GET
    ACALL  SIGNAL_START
       MOV   A,#11010001B  ;==> ADRESS RTC DAN PROSES READ
       ACALL  OUT8BIT
    JC    ERROR_GET
       ACALL  IN8BIT
    MOV   RTC_DAT,A
    JC    ERROR_GET
    ACALL  SIGNAL_STOP
    SETB  EA
    RET

ERROR_GET:
    MOV   A,#080H
    CALL  LCDINS
    MOV   DPTR,#TXT_RTC_ERROR_GET
    CALL  LCDSTRING
    RET

;=================================================================
SET_RTC:
;============================
    CLR   EA
    LCALL  SIGNAL_START
    JC    ERROR_SET
       MOV   A,#11010000B  ;==> ADRESS RTC DAN PROSES WRITE
       CALL   OUT8BIT
    JC    ERROR_SET
    MOV   A,RTC_ADR
       ACALL  OUT8BIT
    JC    ERROR_SET
    MOV   A,RTC_DAT
       ACALL  OUT8BIT
    JC    ERROR_SET
    ACALL  SIGNAL_STOP
    SETB  EA
       RET

ERROR_SET:
    MOV   A,#080H
```

```
    CALL  LCDINS
    MOV   DPTR,#TXT_RTC_ERROR_SET
    CALL  LCDSTRING
    RET

;=====================================================
OUT8BIT:
;=========================
      PUSH  B
      MOV   B,#8      ; akan digeser 8 kali (bit)
OUTLOOP:
      RLC   A         ; bit A.7 digeser ke C di PSW
    MOV   SDA,C       ; nilai C di SDA
      NOP             ; tunggu sebentar sebelum.. ..
    SETB  SCL         ; SCL dibuat = "1"
      NOP             ; tunggu lagi
      NOP
      NOP
      NOP
    CLR   SCL         ; SCL dibuat ="0", BYTE diambil "slave"
    DJNZ  B,OUTLOOP   ; ulangi terus sampai 8 kali
    SETB  SDA         ; SDA dibuat "1"
      NOP             ; agar 'slve' bisa mengirim ACK
      NOP
    SETB  SCL         ; clock ke 9 untuk menerima ACK
      NOP             ; tunggu dulu
      NOP
      NOP
      NOP
    MOV   C,SDA       ; ambil ACK yang dikirim "slave"
    CLR   SCL         ; SCK kembali ke "0"
      POP   B
      RET

;====================================
IN8BIT:
;=========================
      PUSH  B
    SETB  SDA         ; SDA="1" agar "slave" bisa kirim BYTE
      MOV   B,#8      ; akan digeser 8 kali (bit)
INLOOP:
    NOP             ; "slave" boleh mengubah BYTE selama SCK="0"
      NOP
      NOP
    SETB  SCL         ; SCK="1"
      NOP             ; tunggu sebentar
      NOP
    MOV   C,SDA       ; ambil kiriman bit dari "slave"
      RLC   A         ; ditampung di A
    CLR   SCL         ; "slave" boleh merubah BYTE selama SCK="0"
    DJNZ  B,INLOOP
```

```
        POP     B
        RET
;=================================================================
SIGNAL_START:
;============================
    SETB SDA
    SETB SCL        ; Memastikan I2C Bus bisa dipakai

    JNB  SDA,BUSBUSY  ; kalau SDA=0 I2C Bus tidak siap pakai
    JNB  SCL,BUSBUSY  ; kalau SCL=0 I2C Bus tidak siap pakai

        ; Membuat sinyal START
        NOP           ; tunggu sebentar
    CLR  SDA
        NOP           ; tunggu agar BYTE benar-benar stabil
        NOP
        NOP
        NOP
        NOP
    CLR  SCL
        CLR  C        ; C=0 berarti berhasil membuat START
        RET
BUSBUSY:
        SETB C        ; C=1 berarti gagal membuat START
        RET
;=================================================
SIGNAL_STOP:
;============================
    CLR  SDA       ; SDA=0 low beberapa saat
        NOP
        NOP
    SETB SCL       ; SCL=1
        NOP          ; tunggu sebentar
        NOP
        NOP
        NOP
        NOP
    SETB SDA       ; SDA=1
        RET
;==============================
CARI_BULAN:
;=========================
    CJNE  A,#01,CB2
    MOV   DPTR,#BLN_1
    SJMP  CBX
CB2:
    CJNE  A,#02,CB3
    MOV   DPTR,#BLN_2
    SJMP  CBX
CB3:
    CJNE  A,#03,CB4
```

```
        MOV   DPTR,#BLN_3
        SJMP  CBX
    CB4:
        CJNE  A,#04,CB5
        MOV   DPTR,#BLN_4
        SJMP  CBX
    CB5:
        CJNE  A,#05,CB6
        MOV   DPTR,#BLN_5
        SJMP  CBX
    CB6:
        CJNE  A,#06,CB7
        MOV   DPTR,#BLN_6
        SJMP  CBX
    CB7:
        CJNE  A,#07,CB8
        MOV   DPTR,#BLN_7
        SJMP  CBX
    CB8:
        CJNE  A,#08,CB9
        MOV   DPTR,#BLN_8
        SJMP  CBX
    CB9:
        CJNE  A,#09,CB10
        MOV   DPTR,#BLN_9
        SJMP  CBX
    CB10:
        CJNE  A,#010,CB11
        MOV   DPTR,#BLN_10
        SJMP  CBX
    CB11:
        CJNE  A,#011,CB12
        MOV   DPTR,#BLN_11
        SJMP  CBX
    CB12:
        MOV   DPTR,#BLN_12
    CBX:
        RET
    ;=====================================
    SET_WAKTU:
    ;=========================
            MOV   A,#00001101B
        CALL  LCDINS
        JMP   RSJ4
                ;<-- SET TANGGAL
    RSJ0:
        MOV   A,MONTH
        CALL  C_JML_HR
        MOV   A,#087H
        CALL  LCDINS
        MOV   A,DATE
```

```
    MOV   B,#010
    DIV   AB
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,B
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,#088H
    CALL  LCDINS
RSJ1:
    JB    TBUP,RSJ2
    JNB   TBUP,$
    MOV   A,DATE
    CJNE  A,JMLHR,RSJ1_1
    MOV   DATE,#01
    JMP   RSJ0
RSJ1_1:
    INC   DATE
    JMP   RSJ0
RSJ2:
    JB    TBDOWN,RSJ3
    JNB   TBDOWN,$
    MOV   A,DATE
    CJNE  A,#01,RSJ2_1
    MOV   DATE,JMLHR
    JMP   RSJ0
RSJ2_1:
    DEC   DATE
    JMP   RSJ0
RSJ3:
    JB    TBMENU,RSJ1
    JNB   TBMENU,$
    MOV   RTC_ADR,#04H
    MOV   RTC_DAT,DATE
    CALL  SET_RTC
    JMP   RSJ8

          ;<-- SET BULAN
RSJ4:
    MOV   A,#08AH
    CALL  LCDINS
    MOV   A,MONTH
    CALL  CARI_BULAN
    CALL  LCDSTRING
    MOV   A,#08CH
    CALL  LCDINS
    CALL      DELAY
    CALL      DELAY
RSJ5:
    JB    TBUP,RSJ6
    JNB   TBUP,$
```

```
        CALL    DELAY
        CALL    DELAY
        CALL    DELAY
        MOV     A,MONTH
        CJNE    A,#012,RSJ5_1
        MOV     MONTH,#01
        JMP     RSJ4
RSJ5_1:
        INC     MONTH
        JMP     RSJ4
RSJ6:
        JB      TBDOWN,RSJ7
        JNB     TBDOWN,$
        MOV     A,MONTH
        CJNE    A,#01,RSJ6_1
        MOV     MONTH,#012
        JMP     RSJ4
RSJ6_1:
        DEC     MONTH
        JMP     RSJ4
RSJ7:
        JB      TBMENU,RSJ5
        JNB     TBMENU,$
        MOV     RTC_ADR,#05H
        MOV     RTC_DAT,MONTH
        CALL    SET_RTC
        JMP     RSJ0
                ;<-- SET TAHUN
RSJ8:
        MOV     A,#08EH
        CALL    LCDINS
        MOV     A,YEAR
        MOV     B,#010
        DIV     AB
        ADD     A,#030H
        CALL    LCDDAT
        MOV     A,B
        ADD     A,#030H
        CALL    LCDDAT
        MOV     A,#08FH
        CALL    LCDINS
RSJ9:
        JB      TBUP,RSJ10
        JNB     TBUP,$
        MOV     A,YEAR
        CJNE    A,#099,RSJ9_1
        MOV     YEAR,#0
        JMP     RSJ8
RSJ9_1:
        INC     YEAR
        JMP     RSJ8
```

```
RSJ10:
    JB    TBDOWN,RSJ11
    JNB   TBDOWN,$
    MOV   A,YEAR
    JNZ   RSJ10_1
    MOV   YEAR,#099
    JMP   RSJ8
RSJ10_1:
    DEC   YEAR
    JMP   RSJ8
RSJ11:
    JB    TBMENU,RSJ9
    JNB   TBMENU,$
    MOV   RTC_ADR,#06H
    MOV   RTC_DAT,YEAR
    CALL  SET_RTC

          ;<-- SET JAM
RSJ12:
    MOV   A,#0C8H
    CALL  LCDINS
    MOV   A,HOUR
    MOV   B,#010
    DIV   AB
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,B
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,#0C9H
    CALL  LCDINS
RSJ13:
    JB    TBUP,RSJ14
    JNB   TBUP,$
    call delay
    MOV   A,HOUR
    CJNE  A,#023,RSJ13_1
    MOV   HOUR,#0
    JMP   RSJ12
RSJ13_1:
    INC   HOUR
    JMP   RSJ12
RSJ14:
    JB    TBDOWN,RSJ15
    JNB   TBDOWN,$
    call delay
    MOV   A,HOUR
    JNZ   RSJ14_1
    MOV   HOUR,#023
    JMP   RSJ12
RSJ14_1:
```

```
    DEC   HOUR
    JMP   RSJ12
RSJ15:
    JB    TBMENU,RSJ13
    JNB   TBMENU,$
    call delay
    MOV   RTC_ADR,#02H
    MOV   RTC_DAT,HOUR
    CALL  SET_RTC

              ;<-- SET MENIT
RSJ16:
    MOV   A,#0CBH
    CALL  LCDINS
    MOV   A,MIN
    MOV   B,#010
    DIV   AB
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,B
    ADD   A,#030H
    CALL  LCDDAT
    MOV   A,#0CCH
    CALL  LCDINS
RSJ17:
    JB    TBUP,RSJ18
    JNB   TBUP,$
    MOV   A,MIN
    CJNE  A,#059,RSJ17_1
    MOV   MIN,#0
    JMP   RSJ16
RSJ17_1:
    INC   MIN
    JMP   RSJ16
RSJ18:
    JB    TBDOWN,RSJ19
    JNB   TBDOWN,$
    MOV   A,MIN
    JNZ   RSJ18_1
    MOV   MIN,#059
    JMP   RSJ16
RSJ18_1:
    DEC   MIN
    JMP   RSJ16
RSJ19:
    JB    TBMENU,RSJ17
    JNB   TBMENU,$
    MOV   RTC_ADR,#01H
    MOV   RTC_DAT,MIN
    CALL  SET_RTC
    CALL      DELAY
```

```
    MOV   A,#DISPON
    CALL  LCDINS
    CALL  DELAY
    RET
;=============================
PINDAH_DATA_RTC:
;=========================
    MOV   A,RTC_ADR
    JNZ   PDR1
    MOV   A,RTC_DAT
    ANL   A,#0FH
    MOV   TEMP,A
    MOV   A,RTC_DAT
    SWAP  A
    ANL   A,#0FH
    MOV   B,#010
    MUL   AB
    ADD   A,TEMP
    MOV   SEC,A
    SJMP  PDRX
PDR1:
    CJNE  A,#01,PDR2
    MOV   A,RTC_DAT
    ANL   A,#0FH
    MOV   TEMP,A
    MOV   A,RTC_DAT
    SWAP  A
    ANL   A,#0FH
    MOV   B,#010
    MUL   AB
    ADD   A,TEMP
    MOV   MIN,A
    SJMP  PDRX
PDR2:
    CJNE  A,#02,PDR3
    MOV   A,RTC_DAT
    ANL   A,#0FH
    MOV   TEMP,A
    MOV   A,RTC_DAT
    SWAP  A
    ANL   A,#0FH
    MOV   B,#010
    MUL   AB
    ADD   A,TEMP
    MOV   HOUR,A
    SJMP  PDRX
PDR3:
    CJNE  A,#04,PDR4
    MOV   A,RTC_DAT
    ANL   A,#0FH
    MOV   TEMP,A
```

```
        MOV   A,RTC_DAT
        SWAP  A
        ANL   A,#0FH
        MOV   B,#010
        MUL   AB
        ADD   A,TEMP
        MOV   DATE,A
        SJMP  PDRX
PDR4:
        CJNE  A,#05,PDR5
        MOV   A,RTC_DAT
        ANL   A,#0FH
        MOV   TEMP,A
        MOV   A,RTC_DAT
        SWAP  A
        ANL   A,#0FH
        MOV   B,#010
        MUL   AB
        ADD   A,TEMP
        MOV   MONTH,A
        SJMP  PDRX
PDR5:
        CJNE  A,#06,PDRX
        MOV   A,RTC_DAT
        ANL   A,#0FH
        MOV   TEMP,A
        MOV   A,RTC_DAT
        SWAP  A
        ANL   A,#0FH
        MOV   B,#010
        MUL   AB
        ADD   A,TEMP
        MOV   YEAR,A
PDRX:
        MOV   TEMP,#0
        RET
;============================
C_JML_HR:
;========================
        CJNE  A,#01,CJH2
        MOV   JMLHR,#031
        SJMP  CJHX
CJH2:
        CJNE  A,#02,CJH3
        MOV   JMLHR,#028
        SJMP  CJHX
CJH3:
        CJNE  A,#03,CJH4
        MOV   JMLHR,#031
        SJMP  CJHX
CJH4:
```

```
    CJNE   A,#04,CJH5
    MOV    JMLHR,#030
    SJMP   CJHX
CJH5:
    CJNE   A,#05,CJH6
    MOV    JMLHR,#031
    SJMP   CJHX
CJH6:
    CJNE   A,#06,CJH7
    MOV    JMLHR,#030
    SJMP   CJHX
CJH7:
    CJNE   A,#07,CJH8
    MOV    JMLHR,#031
    SJMP   CJHX
CJH8:
    CJNE   A,#08,CJH9
    MOV    JMLHR,#031
    SJMP   CJHX
CJH9:
    CJNE   A,#09,CJH10
    MOV    JMLHR,#030
    SJMP   CJHX
CJH10:
    CJNE   A,#010,CJH11
    MOV    JMLHR,#031
    SJMP   CJHX
CJH11:
    CJNE   A,#011,CJH12
    MOV    JMLHR,#030
    SJMP   CJHX
CJH12:
    CJNE   A,#012,CJHX
    MOV    JMLHR,#031
CJHX:
    RET
;=========================
BLN_1  : DB 'JAN',0
BLN_2  : DB 'FEB',0
BLN_3  : DB 'MAR',0
BLN_4  : DB 'APR',0
BLN_5  : DB 'MEI',0
BLN_6  : DB 'JUN',0
BLN_7  : DB 'JUL',0
BLN_8  : DB 'AGT',0
BLN_9  : DB 'SEP',0
BLN_10 : DB 'OKT',0
BLN_11 : DB 'NOV',0
BLN_12 : DB 'DES',0
```

```
TXT_RTC_ERROR_GET    :    DB    ' ERROR GET RTC ',0
TXT_RTC_ERROR_SET    :    DB    ' ERROR SET RTC ',0
;=========================
PRINTSTRINGLOOP:
;=====================
    CALL  LCDDAT
    INC   DPTR
LCDSTRING:
    CLR   A
    MOVC  A,@A+DPTR
    JNZ   PRINTSTRINGLOOP
    RET
KURSOR:
       MOV   A,#00001101B
    CALL  LCDINS
    RET
DISPLON:
       MOV   A,#DISPON
    CALL  LCDINS
    MOV   A,#DISPCLR
    CALL  LCDINS
       RET
TAMPIL:
       ADD   A,#030H
    CALL  LCDDAT
    RET

;================================
LCDINS:
;=====================
    MOV   PLCD,A
    CLR   LCDRS
    SJMP  LCDOUT
LCDDAT:
    MOV   PLCD,A
    SETB  LCDRS
LCDOUT:
    SETB  LCDE
    CALL       DELAY
    CLR   LCDE
       CALL   DELAY
    RET
DSP_ON:
       MOV   A,#DISPON
    CALL  LCDINS
    CALL       DELAY
    RET
;================================
DELAY4:
    MOV   R7,#0250
DELAY4_1:
```

```
        DJNZ   R7,DELAY4_1
        RET
 DELAY3:
          MOV    PUTR,A
 MUTERZ:
        CALL   DELAY2
        DJNZ   PUTR,MUTERZ
        RET
 DELAY2:
        MOV    R5,#130
 MUTERX:
        MOV    R6,#250
        CALL   DELAY
        DJNZ   R6,$
        DJNZ   R5,MUTERX
        RET
 DELAY:
        MOV    R3,#08
 MUTER:
        MOV    R4,#0255
           DJNZ    R4,$
           DJNZ    R3,MUTER
           RET
;============================
;============================
INITLCD:
;============================
       MOV    A,#DISPCLR
       CALL   LCDINS
       CALL   DELAY
       MOV    A,#FUNCSET
       CALL   LCDINS
       CALL   DELAY
       MOV    A,#DISPON
       CALL   LCDINS
       CALL   DELAY
       MOV    A,#ENTRMOD
       CALL   LCDINS
       CALL   DELAY
       MOV    A,#DISPCLR
       CALL   LCDINS
       CALL   DELAY2
       RET
;==================================
DSP_PILIH:
          MOV    A,#080H
      CALL   LCDINS
      MOV    DPTR,#IDLE_1
      CALL   LCDSTRING
      MOV    A,#0C0H
      CALL   LCDINS
```

```
    MOV   DPTR,#IDLE_2
    CALL  LCDSTRING
      RET
DSP_2:
      MOV   A,#080H
    CALL  LCDINS
    MOV   DPTR,#IDLE_3
    CALL  LCDSTRING
    MOV   A,#0C0H
    CALL  LCDINS
    MOV   DPTR,#IDLE_4
    CALL  LCDSTRING
      RET

DSP_PESAN:
      MOV   A,#080H
    CALL  LCDINS
    MOV   DPTR,#IDLE_5
    CALL  LCDSTRING
    MOV   A,#0C0H
    CALL  LCDINS
    MOV   DPTR,#IDLE_6
    CALL  LCDSTRING
      RET
DSP_SET_R:
      MOV   A,#080H
    CALL  LCDINS
    MOV   DPTR,#IDLE_7
    CALL  LCDSTRING
    MOV   A,#0C0H
    CALL  LCDINS
    MOV   DPTR,#IDLE_8
    CALL  LCDSTRING
    RET

DSP_STOP_REC:
      MOV   A,#080H
    CALL  LCDINS
    MOV   DPTR,#IDLE_9
    CALL  LCDSTRING
    MOV   A,#0C0H
    CALL  LCDINS
    MOV   DPTR,#IDLE_A
    CALL  LCDSTRING
    RET

;=============
;TEXT SECTION
;=============
IDLE_1 : DB 'SET REMINDER >>M',0
IDLE_2 : DB 'SET WAKTU   >>U',0
```

```
IDLE_3 : DB '  Record Suara  ',0
IDLE_4 : DB '  Tekan PB Menu ',0
IDLE_5 : DB ' ANDA MENDAPAT  ',0
IDLE_6 : DB ' PESAN REMINDER ',0
IDLE_7 : DB 'SET REMINDER >>M',0
IDLE_8 : DB 'HPS REMINDER >>U',0
IDLE_9 : DB '  STOP RECORD   ',0
IDLE_A : DB ' Tekan PB Menu  ',0

ABOUT_1 : DB '    PERANCANGAN',0
ABOUT_2 : DB ' SISTEM REMINDER',0
ABOUT_3 : DB '     DILENGKAPI',0
ABOUT_4 : DB 'DGN OUTPUT SUARA',0
ABOUT_7 : DB '     ANDI FAUZI',0
ABOUT_8 : DB '  NIM:95.17.038',0
ABOUT_9 : DB 'DOSEN PEMBIMBING',0
ABOUT_A : DB 'IR. F.YUDI.L, MT',0

SETTIME:
        MOV   RTC_ADR,#0H
        MOV   RTC_DAT,#0H   ;DETIK
    CALL  SET_RTC
    MOV   RTC_ADR,#01H      ;MINUTES
    MOV   RTC_DAT,#30H
       CALL  SET_RTC
    CALL  INITLCD
    MOV   RTC_ADR,#02H      ;HOURS
    MOV   RTC_DAT,#30H
       CALL  SET_RTC
    CALL  INITLCD
    MOV   RTC_ADR,#04H      ;DATE
    MOV   RTC_DAT,#30H
       CALL  SET_RTC
    CALL  INITLCD
    MOV   RTC_ADR,#05H      ;MONTH
    MOV   RTC_DAT,#03H
       CALL  SET_RTC
    CALL  INITLCD
    MOV   RTC_ADR,#06H      ;YEAR
    MOV   RTC_DAT,#09H
       CALL  SET_RTC
       MOV   A,#DISPCLR
    CALL  LCDINS
       RET
       ;============================
       ; DAFTAR IKLAN
       ;============================
       IKLAN:
    MOV   A,#080H
    CALL  LCDINS
```

```
        MOV   DPTR,#ABOUT_1
        CALL  LCDSTRING
        MOV   A,#0C0H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_2
        CALL  LCDSTRING
        MOV   A,#03
        CALL  DELAY3
        MOV   A,#080H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_3
        CALL  LCDSTRING
        MOV   A,#0C0H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_4
        CALL  LCDSTRING
        MOV   A,#03
        CALL  DELAY3
        MOV   A,#080H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_7
        CALL  LCDSTRING
        MOV   A,#0C0H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_8
        CALL  LCDSTRING
        MOV   A,#03
        CALL  DELAY3
        MOV   A,#080H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_9
        CALL  LCDSTRING
        MOV   A,#0C0H
        CALL  LCDINS
        MOV   DPTR,#ABOUT_A
        CALL  LCDSTRING
        MOV   A,#03
        CALL  DELAY3
        MOV       A,#DISPCLR
        CALL  LCDINS
        RET
    END
```

Features

- Compatible with MCS-51® Products
- 4K Bytes of In-System Programmable (ISP) Flash Memory
  - Endurance: 1000 Write/Erase Cycles
- 4.0V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 128 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-bit Timer/Counters
- Six Interrupt Sources
- Full Duplex UART Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Fast Programming Time
- Flexible ISP Programming (Byte and Page Mode)

Description

The AT89S51 is a low-power, high-performance CMOS 8-bit microcontroller with 4K bytes of in-system programmable Flash memory. 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 Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S51 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.

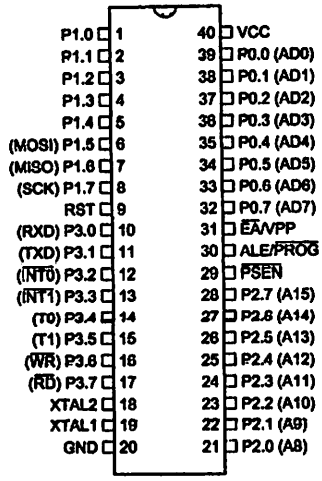
8-bit Microcontroller with 4K Bytes In-System Programmable Flash

AT89S51

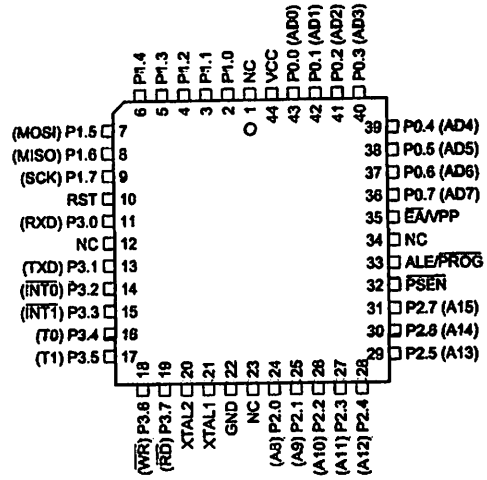
Rev. 2487A–10/01

Pin Configurations

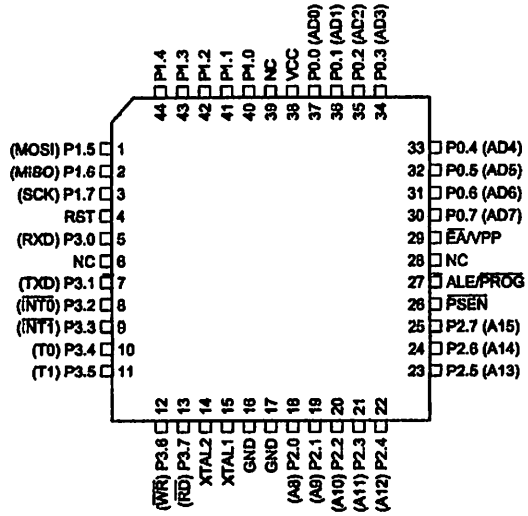
PDIP



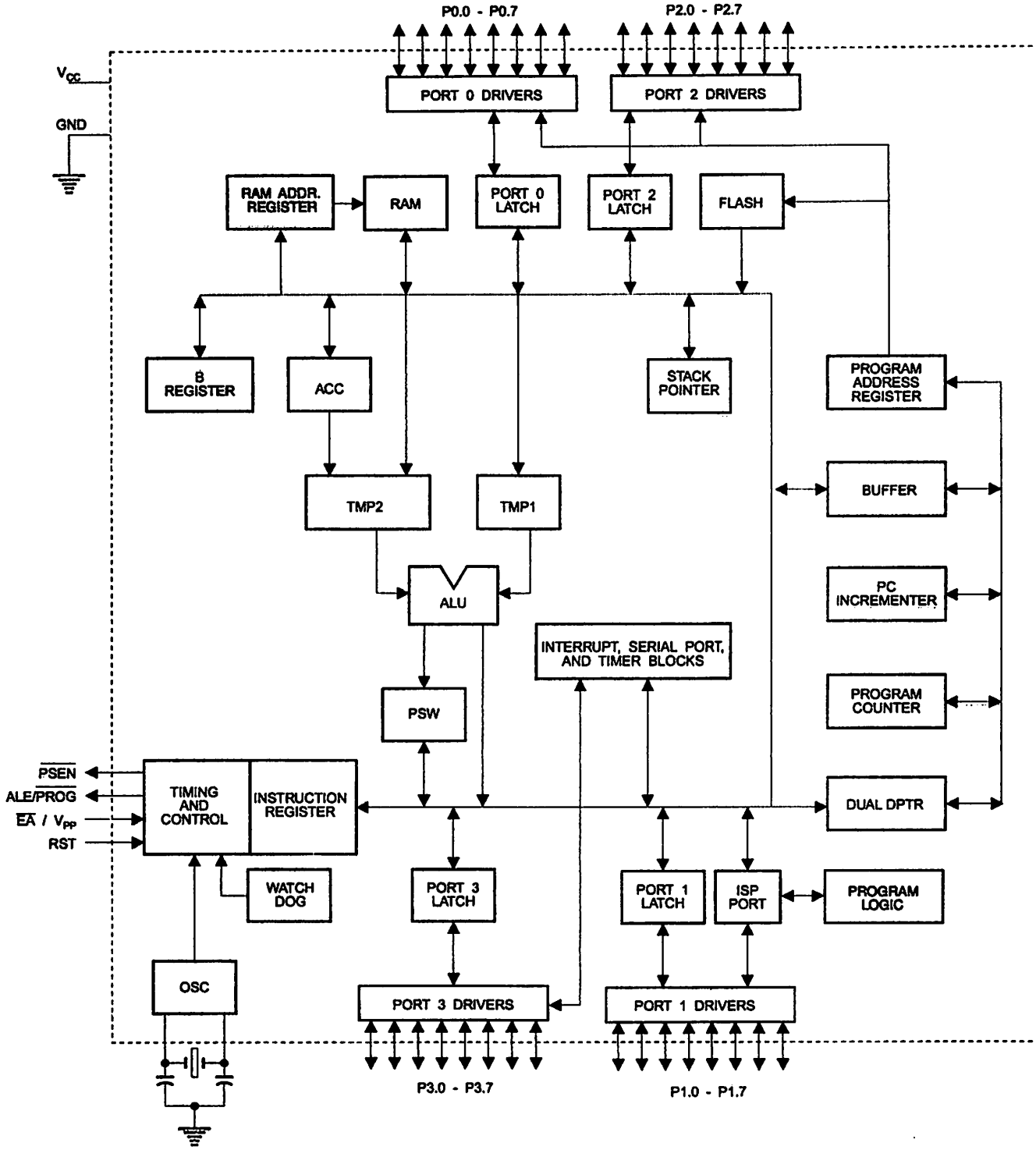
PLCC



TQFP



Block Diagram



Pin Description

VCC Supply voltage.

GND Ground.

Port 0 Port 0 is an 8-bit open drain bidirectional 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 bidirectional 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.

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

Port Pin	Alternate Functions
P1.5	MOSI (used for In-System Programming)
P1.6	MISO (used for In-System Programming)
P1.7	SCK (used for In-System Programming)

Port 2 Port 2 is an 8-bit bidirectional 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 bidirectional 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 AT89S51, 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. This pin drives High for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

ALE/$\overline{PROG}$

Address Latch Enable (ALE) 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 ($\overline{PSEN}$) is the read strobe to external program memory.

When the AT89S51 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.

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.

Table 1. AT89S51 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000								0D7H
0C8H									0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000								0AFH
0A0H	P2 11111111		AUXR1 XXXXXXX0				WDTRST XXXXXXXX		0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111								97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR XXX00XX0		8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000		PCON 0XXX0000	87H

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.

Interrupt Registers: The individual interrupt enable bits are in the IE register. Two priorities can be set for each of the five interrupt sources in the IP register.

Table 2. AUXR: Auxiliary Register

AUXR Address = 8EH Reset Value = XXX00XX0B

Not Bit Addressable

Bit	–	–	–	WDIDLE	DISRTO	–	–	DISALE
	7	6	5	4	3	2	1	0

– Reserved for future expansion

DISALE Disable/Enable ALE

DISALE Operating Mode

0 ALE is emitted at a constant rate of 1/6 the oscillator frequency

1 ALE is active only during a MOVX or MOVC instruction

DISRTO Disable/Enable Reset out

DISRTO

0 Reset pin is driven High after WDT times out

1 Reset pin is input only

WDIDLE Disable/Enable WDT in IDLE mode

WDIDLE

0 WDT continues to count in IDLE mode

1 WDT halts counting in IDLE mode

Dual Data Pointer Registers: To facilitate accessing both internal 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 AUXR1 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 rest under software control and is not affected by reset.

Table 3. AUXR1: Auxiliary Register 1

AUXR1

Address = A2H

Reset Value = XXXXXXX0B

Not Bit Addressable

	–	–	–	–	–	–	–	DPS
Bit	7	6	5	4	3	2	1	0

– Reserved for future expansion

DPS Data Pointer Register Select

DPS	
0	Selects DPTR Registers DP0L, DP0H
1	Selects DPTR Registers DP1L, DP1H

Memory Organization

MCS-51 devices have a separate address space for Program and Data Memory. Up to 64K bytes each of external Program and Data Memory can be addressed.

Program Memory

If the $\overline{EA}$ pin is connected to GND, all program fetches are directed to external memory.

On the AT89S51, if $\overline{EA}$ is connected to V_{CC}, program fetches to addresses 0000H through FFFH are directed to internal memory and fetches to addresses 1000H through FFFFH are directed to external memory.

Data Memory

The AT89S51 implements 128 bytes of on-chip RAM. The 128 bytes are accessible via direct and indirect addressing modes. Stack operations are examples of indirect addressing, so the 128 bytes of data RAM are available as stack space.

Watchdog Timer (One-time Enabled with Reset-out)

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upsets. The WDT consists of a 14-bit counter and the Watchdog Timer Reset (WDTRST) SFR. The WDT is defaulted to disable from exiting reset. To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDTRST register (SFR location 0A6H). When the WDT is enabled, it will increment every machine cycle while the oscillator is running. The WDT timeout period is dependent on the external clock frequency. There is no way to disable the WDT except through reset (either hardware reset or WDT overflow reset). When WDT overflows, it will drive an output RESET HIGH pulse at the RST pin.

Using the WDT

To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDTRST register (SFR location 0A6H). When the WDT is enabled, the user needs to service it by writing 01EH and 0E1H to WDTRST to avoid a WDT overflow. The 14-bit counter overflows when it reaches 16383 (3FFFH), and this will reset the device. When the WDT is enabled, it will increment every machine cycle while the oscillator is running. This means the user must reset the WDT at least every 16383 machine cycles. To reset the WDT the user must write 01EH and 0E1H to WDTRST. WDTRST is a write-only register. The WDT counter cannot be read or written. When WDT overflows, it will generate an output RESET pulse at the RST pin. The RESET pulse duration is 98xTOSC, where TOSC=1/FOSC. To make the best use of the WDT, it

should be serviced in those sections of code that will periodically be executed within the time required to prevent a WDT reset.

WDT During Power-down and Idle

In Power-down mode the oscillator stops, which means the WDT also stops. While in Power-down mode, the user does not need to service the WDT. There are two methods of exiting Power-down mode: by a hardware reset or via a level-activated external interrupt, which is enabled prior to entering Power-down mode. When Power-down is exited with hardware reset, servicing the WDT should occur as it normally does whenever the AT89S51 is reset. Exiting Power-down with an interrupt is significantly different. The interrupt is held low long enough for the oscillator to stabilize. When the interrupt is brought high, the interrupt is serviced. To prevent the WDT from resetting the device while the interrupt pin is held low, the WDT is not started until the interrupt is pulled high. It is suggested that the WDT be reset during the interrupt service for the interrupt used to exit Power-down mode.

To ensure that the WDT does not overflow within a few states of exiting Power-down, it is best to reset the WDT just before entering Power-down mode.

Before going into the IDLE mode, the WDIDLE bit in SFR AUXR is used to determine whether the WDT continues to count if enabled. The WDT keeps counting during IDLE (WDIDLE bit = 0) as the default state. To prevent the WDT from resetting the AT89S51 while in IDLE mode, the user should always set up a timer that will periodically exit IDLE, service the WDT, and reenter IDLE mode.

With WDIDLE bit enabled, the WDT will stop to count in IDLE mode and resumes the count upon exit from IDLE.

UART

The UART in the AT89S51 operates the same way as the UART in the AT89C51. For further information on the UART operation, refer to the ATMEL Web site (http://www.atmel.com). From the home page, select 'Products', then '8051-Architecture Flash Microcontroller', then 'Product Overview'.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S51 operate the same way as Timer 0 and Timer 1 in the AT89C51. For further information on the timers' operation, refer to the ATMEL Web site (http://www.atmel.com). From the home page, select 'Products', then '8051-Architecture Flash Microcontroller', then 'Product Overview'.

Interrupts

The AT89S51 has a total of five interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), two timer interrupts (Timers 0 and 1), and the serial port interrupt. These interrupts are all shown in Figure 1.

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 4 shows that bit position IE.6 is unimplemented. In the AT89S51, 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.

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

Table 4. Interrupt Enable (IE) Register

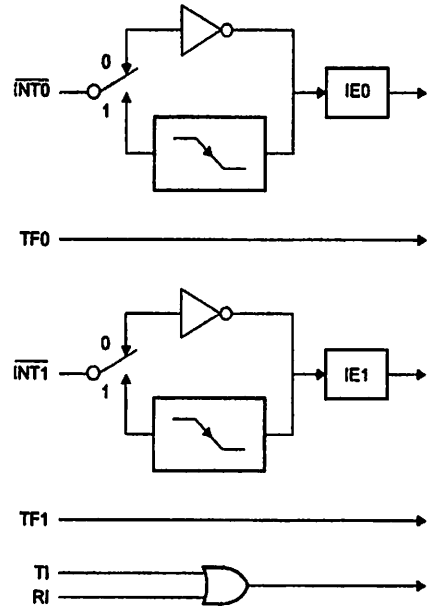
(MSB)							(LSB)
EA	–	–	ES	ET1	EX1	ET0	EX0

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
–	IE.5	Reserved
ES	IE.4	Serial Port 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 reserved bits, because they may be used in future AT89 products.		

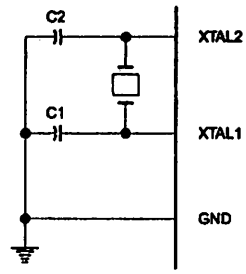
Figure 1. Interrupt Sources



Oscillator Characteristics

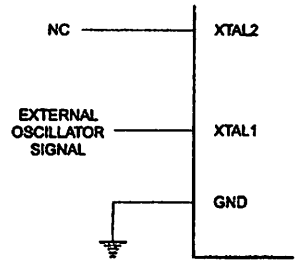
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 3. 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 2. Oscillator Connections



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

Figure 3. 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 function 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.

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 mode can be initiated either by a hardware reset or by activation of an enabled external interrupt into $\overline{INT0}$ or $\overline{INT1}$. 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.

Table 5. 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

Program Memory Lock Bits

The AT89S51 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.

Table 6. Lock Bit Protection Modes

Program Lock Bits				
	LB1	LB2	LB3	Protection Type
1	U	U	U	No program lock features
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 is disabled
3	P	P	U	Same as mode 2, but verify is also disabled
4	P	P	P	Same as mode 3, but external execution is also disabled

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.

Programming the Flash – Parallel Mode

The AT89S51 is shipped with the on-chip Flash memory array ready to be programmed. The programming interface needs a high-voltage (12-volt) program enable signal and is compatible with conventional third-party Flash or EPROM programmers.

The AT89S51 code memory array is programmed byte-by-byte.

Programming Algorithm: Before programming the AT89S51, the address, data, and control signals should be set up according to the Flash programming mode table and Figures 13 and 14. To program the AT89S51, take the following steps:

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise $\overline{EA}/V_{PP}$ to 12V.
5. Pulse ALE/$\overline{PROG}$ once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 50 µs. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

$\overline{Data}$ Polling: The AT89S51 features $\overline{Data}$ Polling to indicate the end of a byte write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on P0.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. $\overline{Data}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{\text{Busy}}$: The progress of byte programming can also be monitored by the RDY/$\overline{\text{BSY}}$ output signal. P3.0 is pulled low after ALE goes high during programming to indicate $\overline{\text{BUSY}}$. P3.0 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 data can be read back via the address and data lines for verification. The status of the individual lock bits can be verified directly by reading them back.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 000H, 100H, and 200H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

(000H) = 1EH indicates manufactured by Atmel
(100H) = 51H indicates 89S51
(200H) = 06H

Chip Erase: In the parallel programming mode, a chip erase operation is initiated by using the proper combination of control signals and by pulsing ALE/$\overline{\text{PROG}}$ low for a duration of 200 ns - 500 ns.

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 500 ms.

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

Programming the Flash – Serial Mode

The Code memory array can be programmed using the serial ISP interface 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 other operations can be executed. Before a reprogramming sequence can occur, a Chip Erase operation is required.

The Chip Erase operation turns the content of every memory location in the Code array into FFH.

Either an external system clock can be 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 1/16 of the crystal frequency. With a 33 MHz oscillator clock, the maximum SCK frequency is 2 MHz.

Serial Programming Algorithm

To program and verify the AT89S51 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 33 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 16.
3. The Code array is programmed one byte at a time in either the Byte or Page mode. The write cycle is self-timed and typically takes less than 0.5 ms at 5V.
4. Any memory location can be verified by using the Read instruction that 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 device operation.

Power-off sequence (if needed):

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

Set RST to "L".

Turn V_{CC} power off.

Data Polling: The Data Polling feature is also available in the serial mode. In this mode, during a write cycle an attempted read of the last byte written will result in the complement of the MSB of the serial output byte on MISO.

Serial Programming Instruction Set

The Instruction Set for Serial Programming follows a 4-byte protocol and is shown in Table 8 on page 18.

Programming Interface – Parallel Mode

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

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

Table 7. Flash Programming Modes

Mode	V_{CC}	RST	PSEN	ALE/PROG	EA/V_{PP}	P2.6	P2.7	P3.3	P3.6	P3.7	P0.7-0 Data	P2.3-0 Address	P1.7-0 Address
Write Code Data	5V	H	L	(2)	12V	L	H	H	H	H	D_{IN}	A11-8	A7-0
Read Code Data	5V	H	L	H	H	L	L	L	H	H	D_{OUT}	A11-8	A7-0
Write Lock Bit 1	5V	H	L	(3)	12V	H	H	H	H	H	X	X	X
Write Lock Bit 2	5V	H	L	(3)	12V	H	H	H	L	L	X	X	X
Write Lock Bit 3	5V	H	L	(3)	12V	H	L	H	H	L	X	X	X
Read Lock Bits 1, 2, 3	5V	H	L	H	H	H	H	L	H	L	P0.2, P0.3, P0.4	X	X
Chip Erase	5V	H	L	(1)	12V	H	L	H	L	L	X	X	X
Read Atmel ID	5V	H	L	H	H	L	L	L	L	L	1EH	0000	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	51H	0001	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	06H	0010	00H

Notes:
1. Each PROG pulse is 200 ns - 500 ns for Chip Erase.
2. Each PROG pulse is 200 ns - 500 ns for Write Code Data.
3. Each PROG pulse is 200 ns - 500 ns for Write Lock Bits.
4. RDY/BSY signal is output on P3.0 during programming.
5. X = don't care.

Figure 4. Programming the Flash Memory (Parallel Mode)

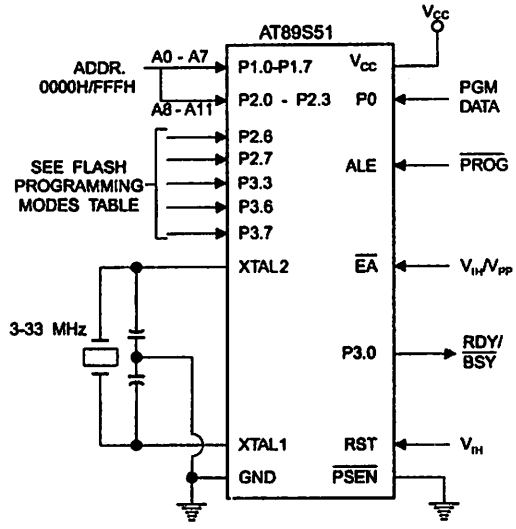
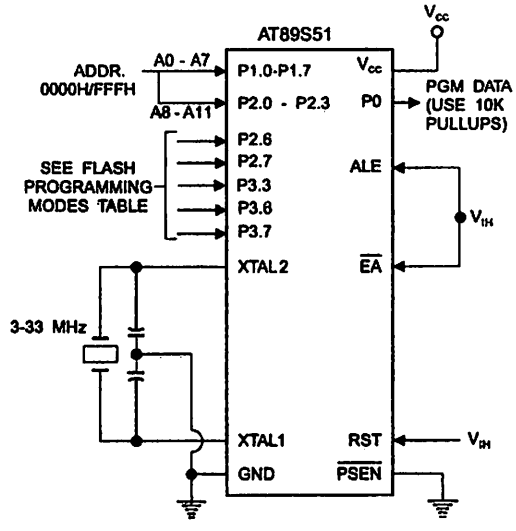


Figure 5. Verifying the Flash Memory (Parallel Mode)



Flash Programming and Verification Characteristics (Parallel Mode)

T_A = 20°C to 30°C, V_{CC} = 4.5 to 5.5V

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Supply Voltage	11.5	12.5	V
I_{PP}	Programming Supply Current		10	mA
I_{CC}	V_{CC} Supply Current		30	mA
$1/t_{CLCL}$	Oscillator Frequency	3	33	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_{GHSL}	V_{PP} Hold After $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	0.2	1	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		50	µs

Figure 6. Flash Programming and Verification Waveforms – Parallel Mode

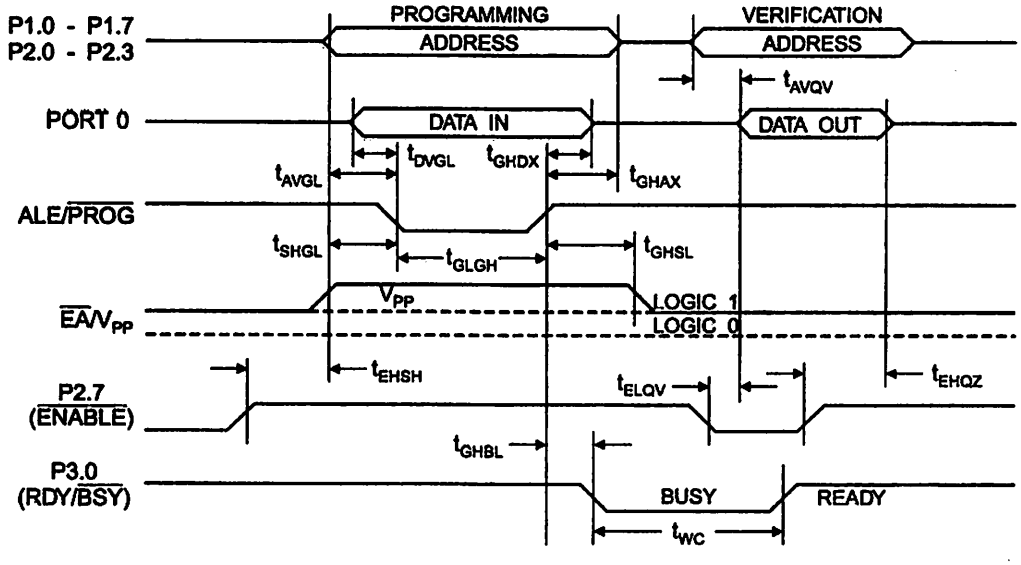
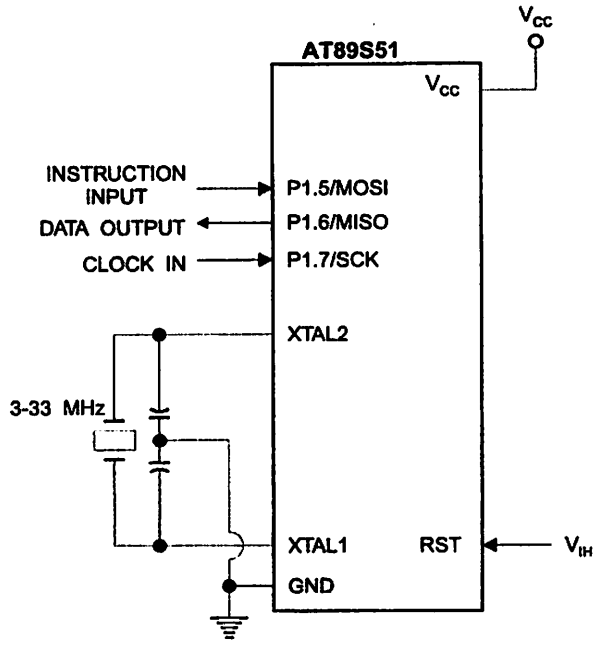


Figure 7. Flash Memory Serial Downloading



Flash Programming and Verification Waveforms – Serial Mode

Figure 8. Serial Programming Waveforms

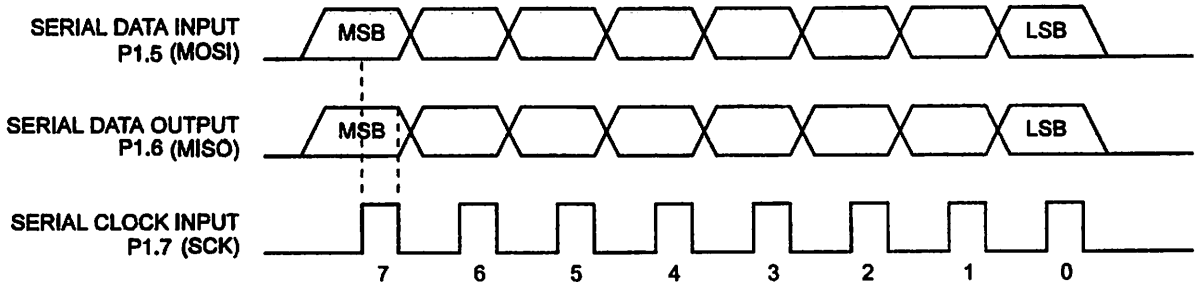


Table 8. Serial Programming Instruction Set

Instruction	Instruction Format Byte 1	Byte 2	Byte 3	Byte 4	Operation
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx 0110 1001 (Output)	Enable Serial Programming while RST is high
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash memory array
Read Program Memory (Byte Mode)	0010 0000	xxxx A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Read data from Program memory in the byte mode
Write Program Memory (Byte Mode)	0100 0000	xxxx A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Write data to Program memory in the byte mode
Write Lock Bits[2]	1010 1100	1110 00 B1 B2	xxxx xxxx	xxxx xxxx	Write Lock bits. See Note (2).
Read Lock Bits	0010 0100	xxxx xxxx	xxxx xxxx	xx LB3 LB2 LB1 xx	Read back current status of the lock bits (a programmed lock bit reads back as a "1")
Read Signature Bytes[1]	0010 1000	xxx A5 A4 A3 A2 A1	A0 xxx xxxx	Signature Byte	Read Signature Byte
Read Program Memory (Page Mode)	0011 0000	xxxx A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Read data from Program memory in the Page Mode (256 bytes)
Write Program Memory (Page Mode)	0101 0000	xxxx A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Write data to Program memory in the Page Mode (256 bytes)

Notes: 1. The signature bytes are not readable in Lock Bit Modes 3 and 4.

2. B1 = 0, B2 = 0 → Mode 1, no lock protection
   B1 = 0, B2 = 1 → Mode 2, lock bit 1 activated
   B1 = 1, B2 = 0 → Mode 3, lock bit 2 activated
   B1 = 1, B1 = 1 → Mode 4, lock bit 3 activated

   } Each of the lock bits needs to be activated sequentially before Mode 4 can be executed.

After Reset signal is high, SCK should be low for at least 64 system clocks before it goes high to clock in the enable data bytes. No pulsing of Reset signal is necessary. SCK should be no faster than 1/16 of the system clock at XTAL1.

For Page Read/Write, the data always starts from byte 0 to 255. After the command byte and upper address byte are latched, each byte thereafter is treated as data until all 256 bytes are shifted in/out. Then the next instruction will be ready to be decoded.

Serial Programming Characteristics

Figure 9. Serial Programming Timing

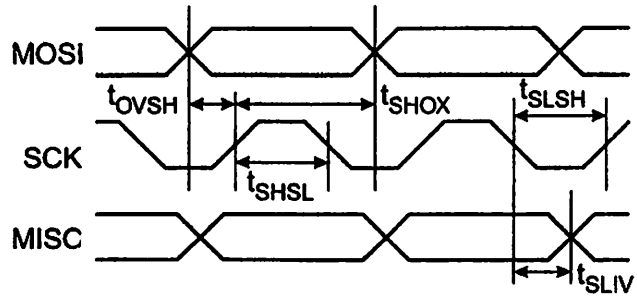


Table 9. Serial Programming Characteristics, T_A = -40° C to 85° C, V_{CC} = 4.0 - 5.5V (Unless Otherwise Noted)

Symbol	Parameter	Min	Typ	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0		33	MHz
t_{CLCL}	Oscillator Period	30			ns
t_{SHSL}	SCK Pulse Width High	8 t_{CLCL}			ns
t_{SLSH}	SCK Pulse Width Low	8 t_{CLCL}			ns
t_{OVSH}	MOSI Setup to SCK High	t_{CLCL}			ns
t_{SHOX}	MOSI Hold after SCK High	2 t_{CLCL}			ns
t_{SLIV}	SCK Low to MISO Valid	10	16	32	ns
t_{ERASE}	Chip Erase Instruction Cycle Time			500	ms
t_{SWC}	Serial Byte Write Cycle Time			64 t_{CLCL} + 400	µs

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} = 4.0V to 5.5V, 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		0.45	V
V_{OL1}	Output Low Voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	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 Pulldown 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} = 5.5V		50	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.

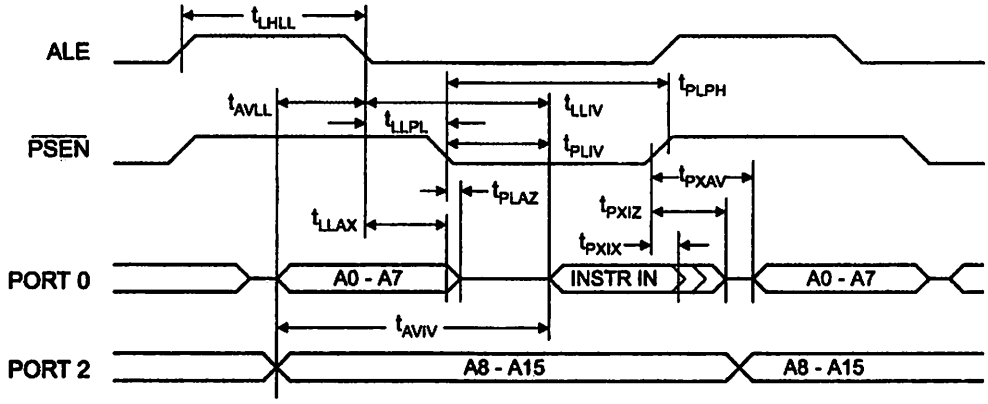
AC Characteristics

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

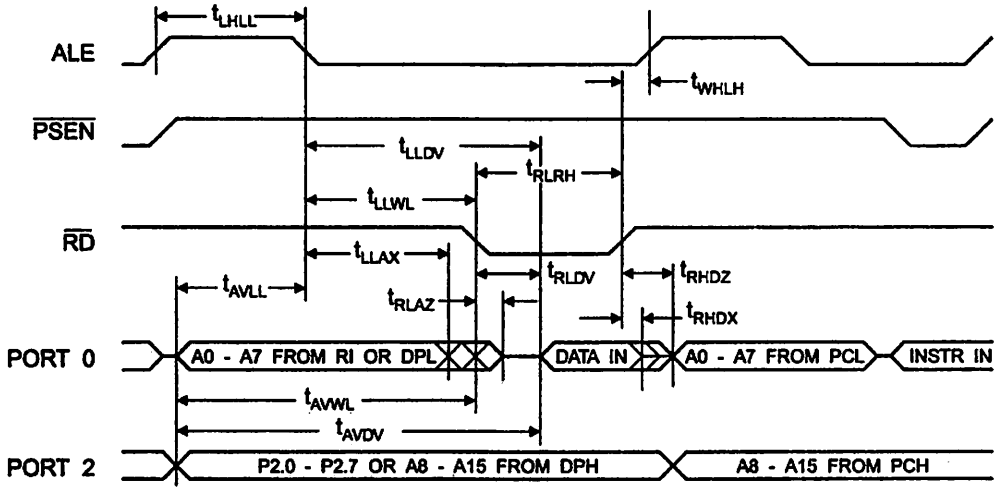
External Program and Data Memory Characteristics

Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	33	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-25		ns
t_{LLAX}	Address Hold After ALE Low	48		t_{CLCL}-25		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{CLCL}$-65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		t_{CLCL}-25		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}$-45		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}$-60	ns
t_{PXIX}	Input Instruction Hold After $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float After $\overline{PSEN}$		59		t_{CLCL}-25	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-80	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold After $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{RD}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8t_{CLCL}$-150	ns
t_{AVDV}	Address to Valid Data In		585		$9t_{CLCL}$-165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		t_{CLCL}-30		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}$-130		ns
t_{WHQX}	Data Hold After $\overline{WR}$	33		t_{CLCL}-25		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	t_{CLCL}-25	t_{CLCL}+25	ns

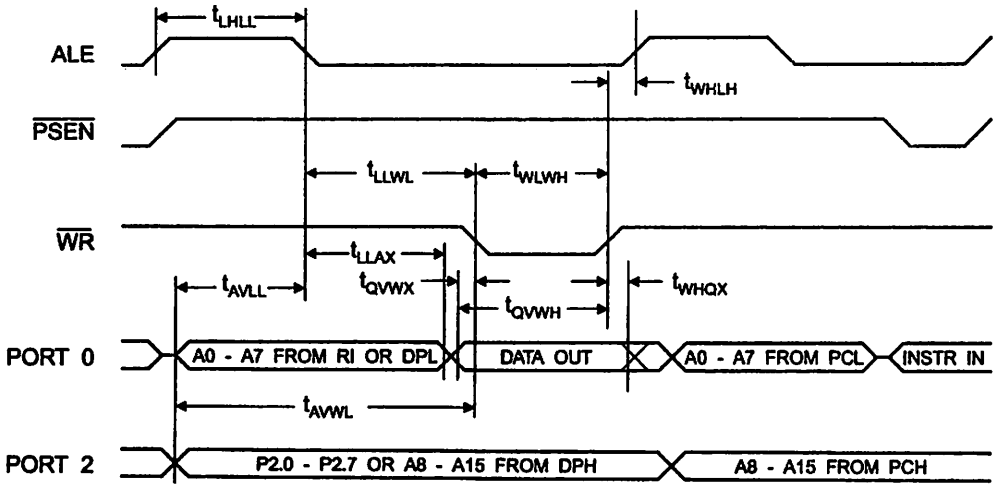
External Program Memory Read Cycle



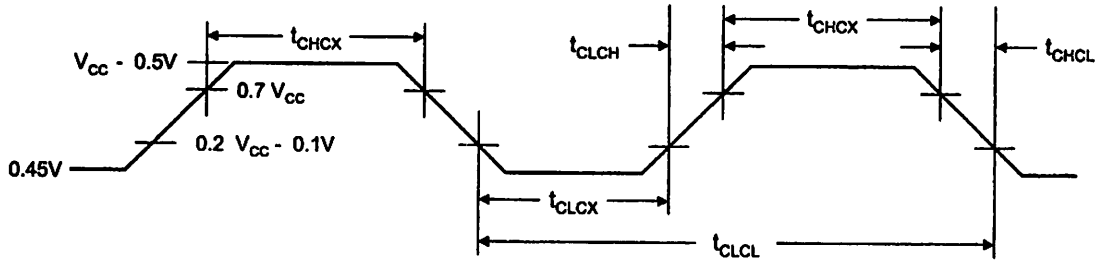
External Data Memory Read Cycle



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

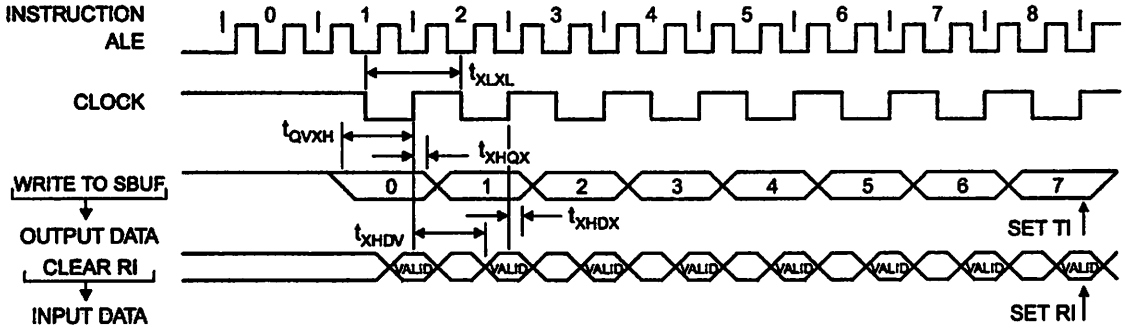
Symbol	Parameter	Min	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0	33	MHz
t_{CLCL}	Clock Period	30		ns
t_{CHCX}	High Time	12		ns
t_{CLCX}	Low Time	12		ns
t_{CLCH}	Rise Time		5	ns
t_{CHCL}	Fall Time		5	ns

Serial Port Timing: Shift Register Mode Test Conditions

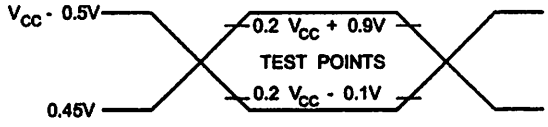
The values in this table are valid for V_{CC} = 4.0V to 5.5V and Load Capacitance = 80 pF.

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}$-133		ns
t_{XHQX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}$-80		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}$-133	ns

Shift Register Mode Timing Waveforms

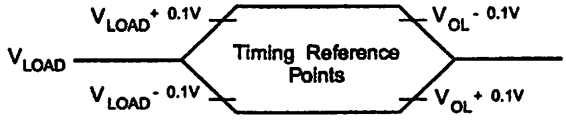


AC Testing Input/Output Waveforms(1)



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

Float Waveforms(1)



Note: 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.

Ordering Information

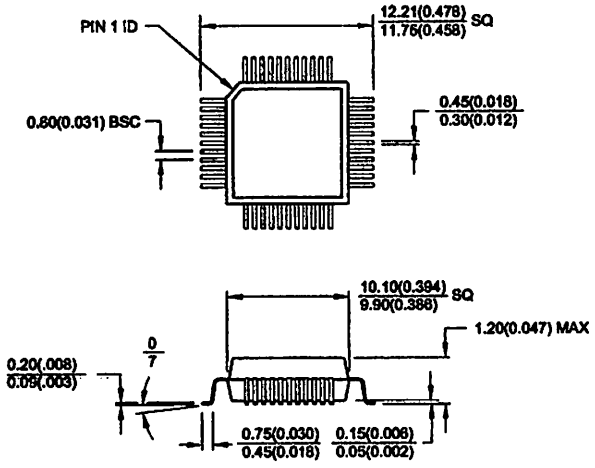
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 5.5V	AT89S51-24AC AT89S51-24JC AT89S51-24PC	44A 44J 40P6	Commercial (0° C to 70° C)
		AT89S51-24AI AT89S51-24JI AT89S51-24PI	44A 44J 40P6	Industrial (-40° C to 85° C)
33	4.5V to 5.5V	AT89S51-33AC AT89S51-33JC AT89S51-33PC	44A 44J 40P6	Commercial (0° C to 70° C)

= Preliminary Availability

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

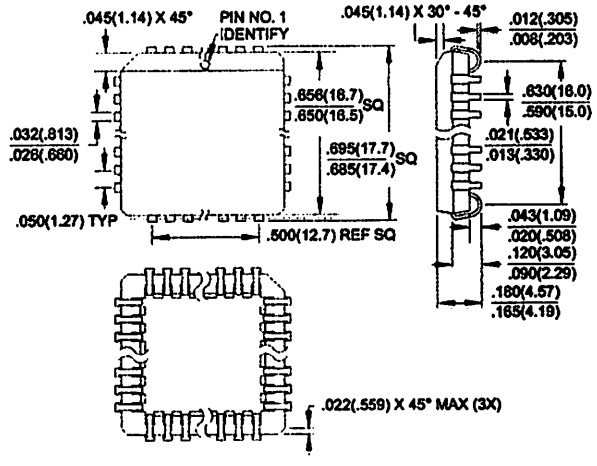
Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (TQFP)
Dimensions in Millimeters and (Inches)*

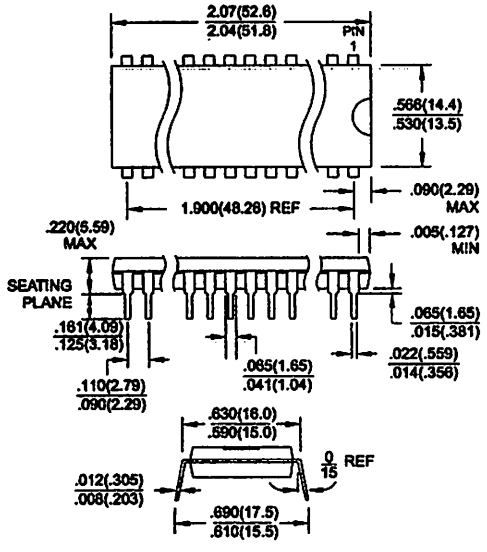


*Controlling dimension: millimeters

44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)



40P6, 40-pin, 0.600" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-011 AC



Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel SarL
Route des Arsenaux 41
Casa Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
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 Product Operations

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

Atmel Grenoble
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-7658-3000
FAX (33) 4-7658-3480

Atmel Heilbronn
Theresienstrasse 2
POB 3535
D-74025 Heilbronn, Germany
TEL (49) 71 31 67 25 94
FAX (49) 71 31 67 24 23

Atmel Nantes
La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
TEL (33) 0 2 40 18 18 18
FAX (33) 0 2 40 18 19 60

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

Atmel Smart Card ICs
Scottish Enterprise Technology Park
East Kilbride, Scotland G75 0QR
TEL (44) 1355-357-000
FAX (44) 1355-242-743

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

2487A–10/01/xM

DALLAS SEMICONDUCTOR

DS1307
64 x 8 Serial Real-Time Clock

www.maxim-ic.com

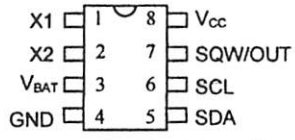
FEATURES

- Real-time clock (RTC) counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap-year compensation valid up to 2100
- 56-byte, battery-backed, nonvolatile (NV) RAM for data storage
- Two-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500nA in battery backup mode with oscillator running
- Optional industrial temperature range: -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Underwriters Laboratory (UL) recognized

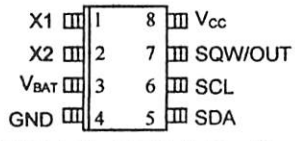
ORDERING INFORMATION

DS1307	8-Pin DIP (300-mil)
DS1307Z	8-Pin SOIC (150-mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

PIN ASSIGNMENT



DS1307 8-Pin DIP (300-mil)



DS1307 8-Pin SOIC (150-mil)

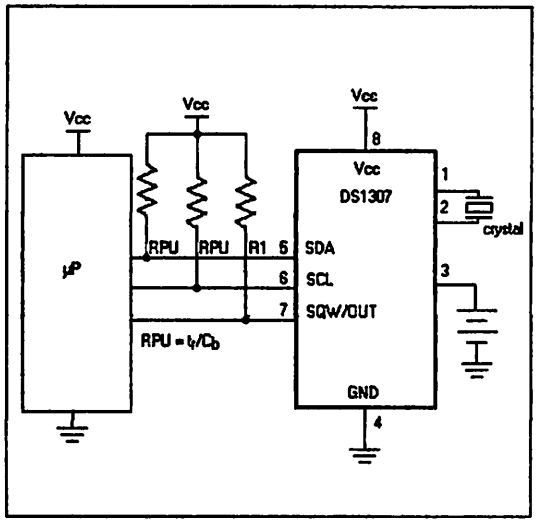
PIN DESCRIPTION

V_{CC}	- Primary Power Supply
X1, X2	- 32.768kHz Crystal Connection
V_{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square Wave/Output Driver

DESCRIPTION

The DS1307 Serial Real-Time Clock is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially via a 2-wire, bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit that detects power failures and automatically switches to the battery supply.

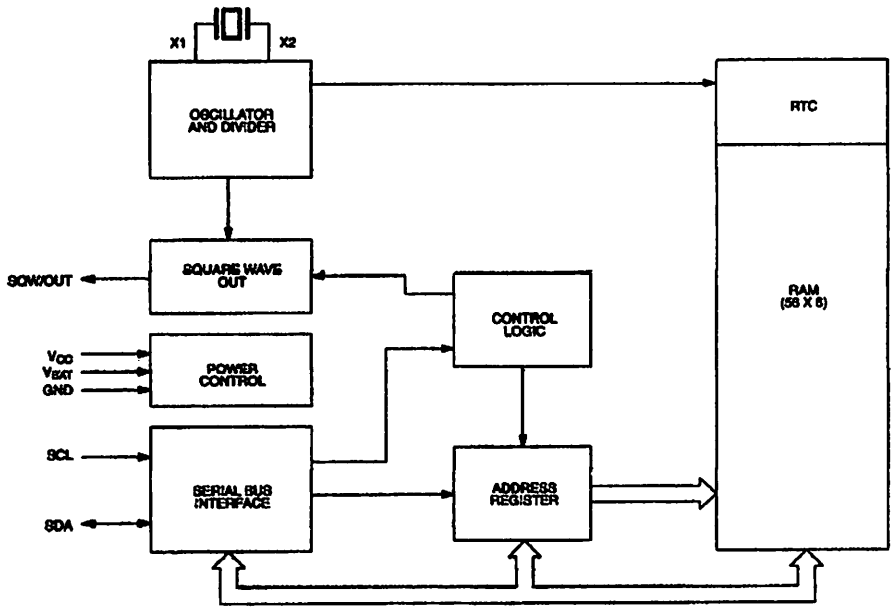
TYPICAL OPERATING CIRCUIT



OPERATION

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below 1.25 x V_{BAT} the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When V_{CC} falls below V_{BAT} the device switches into a low-current battery backup mode. Upon power-up, the device switches from battery to V_{CC} when V_{CC} is greater than V_{BAT} + 0.2V and recognizes inputs when V_{CC} is greater than 1.25 x V_{BAT}. The block diagram in Figure 1 shows the main elements of the serial RTC.

DS1307 BLOCK DIAGRAM Figure 1



SIGNAL DESCRIPTIONS

V_{CC}, GND – DC power is provided to the device on these pins. V_{CC} is the +5V input. When 5V is applied within normal limits, the device is fully accessible and data can be written and read. When a 3V battery is connected to the device and V_{CC} is below 1.25 x V_{BAT}, reads and writes are inhibited. However, the timekeeping function continues unaffected by the lower input voltage. As V_{CC} falls below V_{BAT} the RAM and timekeeper are switched over to the external power supply (nominal 3.0V DC) at V_{BAT}.

V_{BAT} – Battery input for any standard 3V lithium cell or other energy source. Battery voltage must be held between 2.0V and 3.5V for proper operation. The nominal write protect trip point voltage at which access to the RTC and user RAM is denied is set by the internal circuitry as 1.25 x V_{BAT} nominal. A lithium battery with 48mAhr or greater will back up the DS1307 for more than 10 years in the absence of power at 25°C. UL recognized to ensure against reverse charging current when used in conjunction with a lithium battery.

See "Conditions of Acceptability" at http://www.maxim-ic.com/TechSupport/QA/ntrl.htm.

SCL (Serial Clock Input) – SCL is used to synchronize data movement on the serial interface.

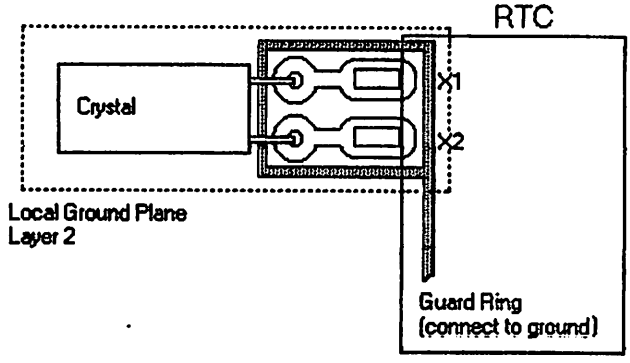
SDA (Serial Data Input/Output) – SDA is the input/output pin for the 2-wire serial interface. The SDA pin is open drain which requires an external pullup resistor.

SQW/OUT (Square Wave/Output Driver) – When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1Hz, 4kHz, 8kHz, 32kHz). The SQW/OUT pin is open drain and requires an external pull-up resistor. SQW/OUT will operate with either Vcc or Vbat applied.

X1, X2 – Connections for a standard 32.768kHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5pF.

For more information on crystal selection and crystal layout considerations, please consult Application Note 58, "Crystal Considerations with Dallas Real-Time Clocks." The DS1307 can also be driven by an external 32.768kHz oscillator. In this configuration, the X1 pin is connected to the external oscillator signal and the X2 pin is floated.

RECOMMENDED LAYOUT FOR CRYSTAL



CLOCK ACCURACY

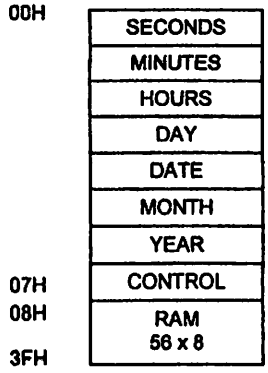
The accuracy of the clock is dependent upon the accuracy of the crystal and the accuracy of the match between the capacitive load of the oscillator circuit and the capacitive load for which the crystal was trimmed. Additional error will be added by crystal frequency drift caused by temperature shifts. External circuit noise coupled into the oscillator circuit may result in the clock running fast. See Application Note 58, "Crystal Considerations with Dallas Real-Time Clocks" for detailed information.

Please review Application Note 95, "Interfacing the DS1307 with a 8051-Compatible Microcontroller" for additional information.

RTC AND RAM ADDRESS MAP

The address map for the RTC and RAM registers of the DS1307 is shown in Figure 2. The RTC registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multi-byte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

DS1307 ADDRESS MAP Figure 2



CLOCK AND CALENDAR

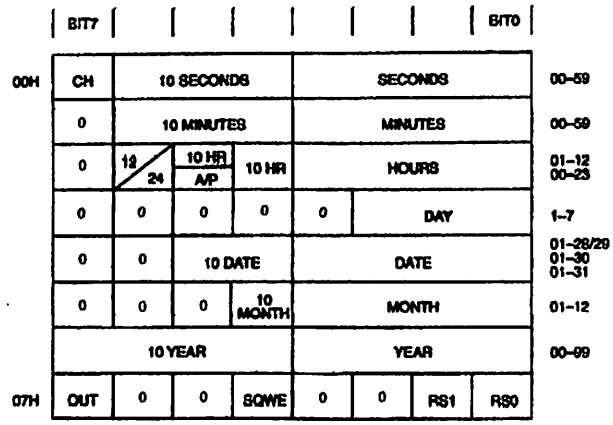
The time and calendar information is obtained by reading the appropriate register bytes. The RTC registers are illustrated in Figure 3. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the BCD format. Bit 7 of register 0 is the clock halt (CH) bit. When this bit is set to a 1, the oscillator is disabled. When cleared to a 0, the oscillator is enabled.

Please note that the initial power-on state of all registers is not defined. Therefore, it is important to enable the oscillator (CH bit = 0) during initial configuration.

The DS1307 can be run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12- or 24-hour mode select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit with logic high being PM. In the 24-hour mode, bit 5 is the second 10 hour bit (20-23 hours).

On a 2-wire START, the current time is transferred to a second set of registers. The time information is read from these secondary registers, while the clock may continue to run. This eliminates the need to re-read the registers in case of an update of the main registers during a read.

DS1307 TIMEKEEPER REGISTERS Figure 3



CONTROL REGISTER

The DS1307 control register is used to control the operation of the SQW/OUT pin.

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

OUT (Output control): This bit controls the output level of the SQW/OUT pin when the square wave output is disabled. If SQWE = 0, the logic level on the SQW/OUT pin is 1 if OUT = 1 and is 0 if OUT = 0.

SQWE (Square Wave Enable): This bit, when set to a logic 1, will enable the oscillator output. The frequency of the square wave output depends upon the value of the RS0 and RS1 bits. With the square wave output set to 1Hz, the clock registers update on the falling edge of the square wave.

RS (Rate Select): These bits control the frequency of the square wave output when the square wave output has been enabled. Table 1 lists the square wave frequencies that can be selected with the RS bits.

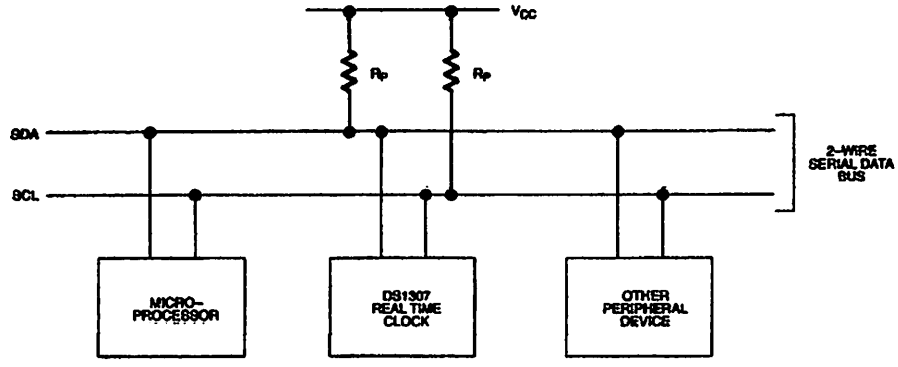
SQUAREWAVE OUTPUT FREQUENCY Table 1

RS1	RS0	SQW OUTPUT FREQUENCY
0	0	1Hz
0	1	4.096kHz
1	0	8.192kHz
1	1	32.768kHz

2-WIRE SERIAL DATA BUS

The DS1307 supports a bi-directional, 2-wire bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter and a device receiving data as a receiver. The device that controls the message is called a master. The devices that are controlled by the master are referred to as slaves. The bus must be controlled by a master device that generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1307 operates as a slave on the 2-wire bus. A typical bus configuration using this 2-wire protocol is show in Figure 4.

TYPICAL 2-WIRE BUS CONFIGURATION Figure 4



Figures 5, 6, and 7 detail how data is transferred on the 2-wire bus.

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is high will be interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus not busy: Both data and clock lines remain HIGH.

Start data transfer: A change in the state of the data line, from HIGH to LOW, while the clock is HIGH, defines a START condition.

Stop data transfer: A change in the state of the data line, from LOW to HIGH, while the clock line is HIGH, defines the STOP condition.

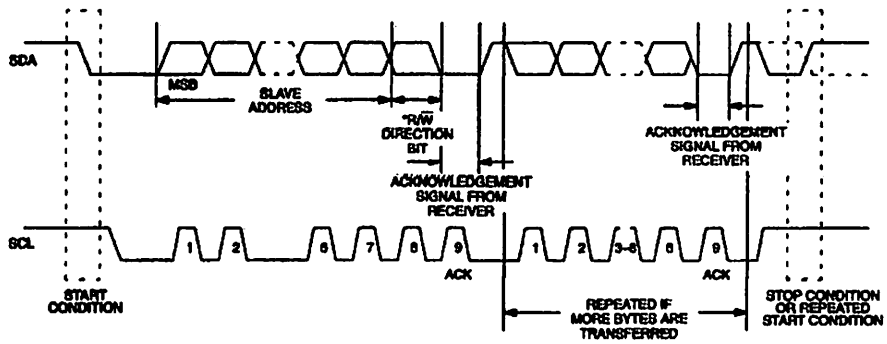
Data valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal. The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit. Within the 2-wire bus specifications a regular mode (100kHz clock rate) and a fast mode (400kHz clock rate) are defined. The DS1307 operates in the regular mode (100kHz) only.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

DATA TRANSFER ON 2-WIRE SERIAL BUS Figure 5



Depending upon the state of the R/$\overline{W}$ bit, two types of data transfer are possible:

1. **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.

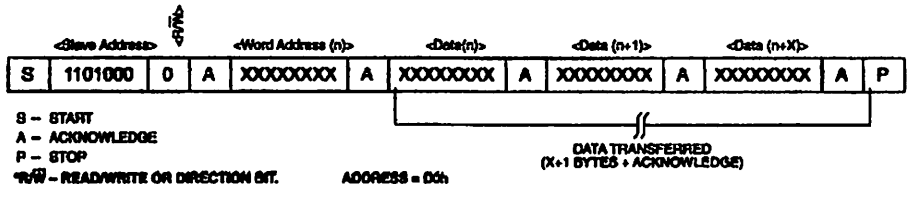
2. **Data transfer from a slave transmitter to a master receiver.** The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

The DS1307 may operate in the following two modes:

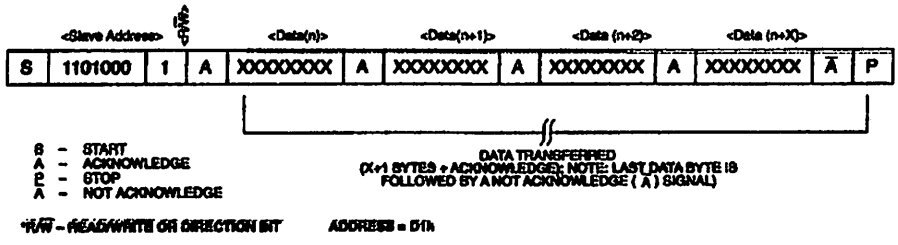
1. **Slave receiver mode (DS1307 write mode):** Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and *direction bit (See Figure 6). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7 bit DS1307 address, which is 1101000, followed by the *direction bit (R/$\overline{W}$) which, for a write, is a 0. After receiving and decoding the address byte the device outputs an acknowledge on the SDA line. After the DS1307 acknowledges the slave address + write bit, the master transmits a register address to the DS1307 This will set the register pointer on the DS1307. The master will then begin transmitting each byte of data with the DS1307 acknowledging each byte received. The master will generate a stop condition to terminate the data write.

DATA WRITE – SLAVE RECEIVER MODE Figure 6



2. **Slave transmitter mode (DS1307 read mode):** The first byte is received and handled as in the slave receiver mode. However, in this mode, the *direction bit will indicate that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1307 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer (See Figure 7). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7-bit DS1307 address, which is 1101000, followed by the *direction bit (R/$\overline{W}$) which, for a read, is a 1. After receiving and decoding the address byte the device inputs an acknowledge on the SDA line. The DS1307 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode the first address that is read is the last one stored in the register pointer. The DS1307 must receive a "not acknowledge" to end a read.

DATA READ – SLAVE TRANSMITTER MODE Figure 7



ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	-0.5V to +7.0V
Storage Temperature	-55°C to +125°C
Soldering Temperature	260°C for 10 seconds DIP See JPC/JEDEC Standard J-STD-020A for Surface Mount Devices

* 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.

Range	Temperature	V_{CC}
Commercial	0°C to +70°C	4.5V to 5.5V V_{CC1}
Industrial	-40°C to +85°C	4.5V to 5.5V V_{CC1}

RECOMMENDED DC OPERATING CONDITIONS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	
Logic 1	V_{IH}	2.2		V_{CC} + 0.3	V	
Logic 0	V_{IL}	-0.5		+0.8	V	
V_{BAT} Battery Voltage	V_{BAT}	2.0		3.5	V	

*Unless otherwise specified.

DC ELECTRICAL CHARACTERISTICS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage (SCL)	I_{LI}			1	µA	
I/O Leakage (SDA & SQW/OUT)	I_{LO}			1	µA	
Logic 0 Output (I_{OL} = 5mA)	V_{OL}			0.4	V	
Active Supply Current	I_{CCA}			1.5	mA	7
Standby Current	I_{CCS}			200	µA	1
Battery Current (OSC ON); SQW/OUT OFF	I_{BAT1}		300	500	nA	2
Battery Current (OSC ON); SQW/OUT ON (32kHz)	I_{BAT2}		480	800	nA	
Power-Fail Voltage	V_{PF}	1.216 x V_{BAT}	1.25 x V_{BAT}	1.284 x V_{BAT}	V	8

*Unless otherwise specified.

AC ELECTRICAL CHARACTERISTICS

(Over the operating range*)

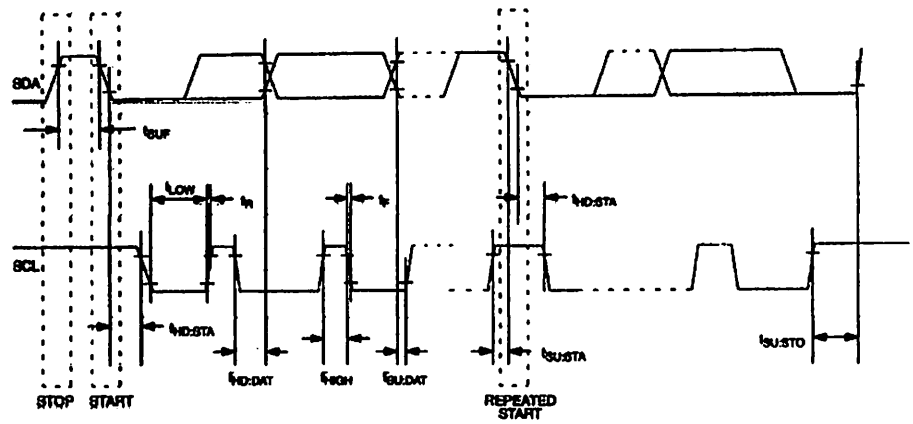
PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	kHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			µs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			µs	3
LOW Period of SCL Clock	t_{LOW}	4.7			µs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			µs	
Set-up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			µs	
Data Hold Time	$t_{HD:DAT}$	0			µs	4,5
Data Set-up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_R			1000	ns	
Fall Time of Both SDA and SCL Signals	t_F			300	ns	
Set-up Time for STOP Condition	$t_{SU:STO}$	4.7			µs	
Capacitive Load for each Bus Line	C_B			400	pF	6
I/O Capacitance (T_A = 25°C)	$C_{I/O}$		10		pF	
Crystal Specified Load Capacitance (T_A = 25°C)			12.5		pF	

*Unless otherwise specified.

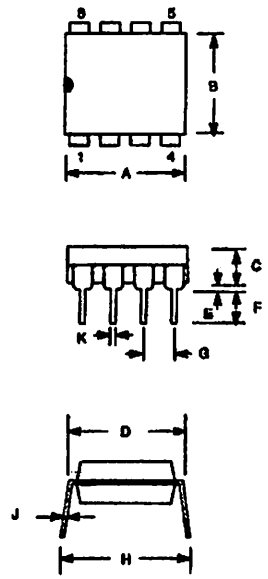
NOTES:

1. I_{CCS} specified with V_{CC} = 5.0V and SDA, SCL = 5.0V.
2. V_{CC} = 0V, V_{BAT} = 3V.
3. After this period, the first clock pulse is generated.
4. A device must internally provide a hold time of at least 300ns for the SDA signal (referred to the V_{IHMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
5. The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
6. C_B – Total capacitance of one bus line in pF.
7. I_{CCA} – SCL clocking at max frequency = 100kHz.
8. V_{PF} measured at V_{BAT} = 3.0V.

TIMING DIAGRAM Figure 8

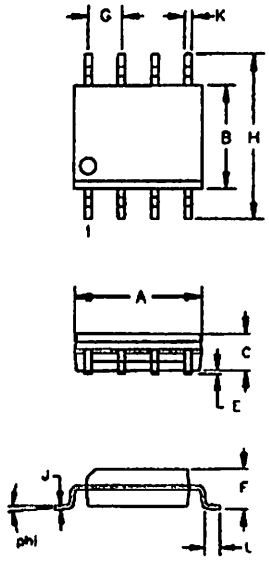


DS1307 64 X 8 SERIAL REAL-TIME CLOCK
8-PIN DIP MECHANICAL DIMENSIONS



PKG	8-PIN	
DIM	MIN	MAX
A IN.	0.360	0.400
MM	9.14	10.16
B IN.	0.240	0.260
MM	6.10	6.60
C IN.	0.120	0.140
MM	3.05	3.56
D IN.	0.300	0.325
MM	7.62	8.26
E IN.	0.015	0.040
MM	0.38	1.02
F IN.	0.120	0.140
MM	3.04	3.56
G IN.	0.090	0.110
MM	2.29	2.79
H IN.	0.320	0.370
MM	8.13	9.40
J IN.	0.008	0.012
MM	0.20	0.30
K IN.	0.015	0.021
MM	0.38	0.53

DS1307Z 64 X 8 SERIAL REAL-TIME CLOCK
8-PIN SOIC (150-MIL) MECHANICAL DIMENSIONS



PKG	8-PIN (150 MIL)	
DIM	MIN	MAX
A IN.	0.188	0.196
MM	4.78	4.98
B IN.	0.150	0.158
MM	3.81	4.01
C IN.	0.048	0.062
MM	1.22	1.57
E IN.	0.004	0.010
MM	0.10	0.25
F IN.	0.053	0.069
MM	1.35	1.75
G IN.	0.050 BSC	
MM	1.27 BSC	
H IN.	0.230	0.244
MM	5.84	6.20
J IN.	0.007	0.011
MM	0.18	0.28
K IN.	0.012	0.020
MM	0.30	0.51
L IN.	0.016	0.050
MM	0.41	1.27
phi	0°	8°

56-G2008-001

Winbond Electronics Corp.

ISD1400 SERIES

SINGLE-CHIP

VOICE RECORD/PLAYBACK DEVICES

16- AND 20-SECOND DURATION

Publication Release Date: March 2004
Revision 1.0

Winbond Electronics Corp.

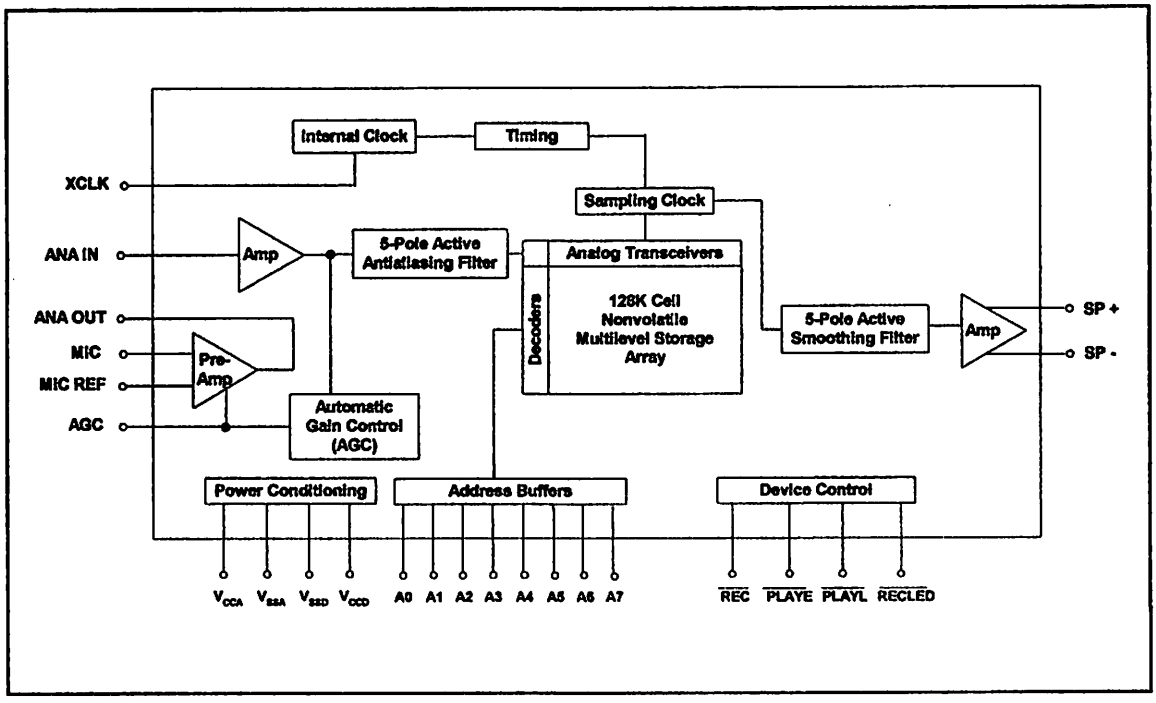
1. GENERAL DESCRIPTION

Winbond's ISD1400 ChipCorder® series provide high-quality, single-chip, Record/Playback solutions to short-duration messaging applications. The CMOS devices include an on-chip oscillator, microphone preamplifier, automatic gain control, anti-aliasing filter, smoothing filter, and speaker amplifier. A minimum Record/Playback subsystem can be configured with a microphone, a speaker, several passive components, two push buttons and a power source. Recordings are stored into on-chip non-volatile memory cells, providing zero-power message storage. This unique, single-chip solution is made possible through Winbond's patented Multi-Level Storage (MLS) technology. Voice and audio signals are stored directly into memory in their natural form, providing high-quality, solid-state voice reproduction.

2. FEATURES

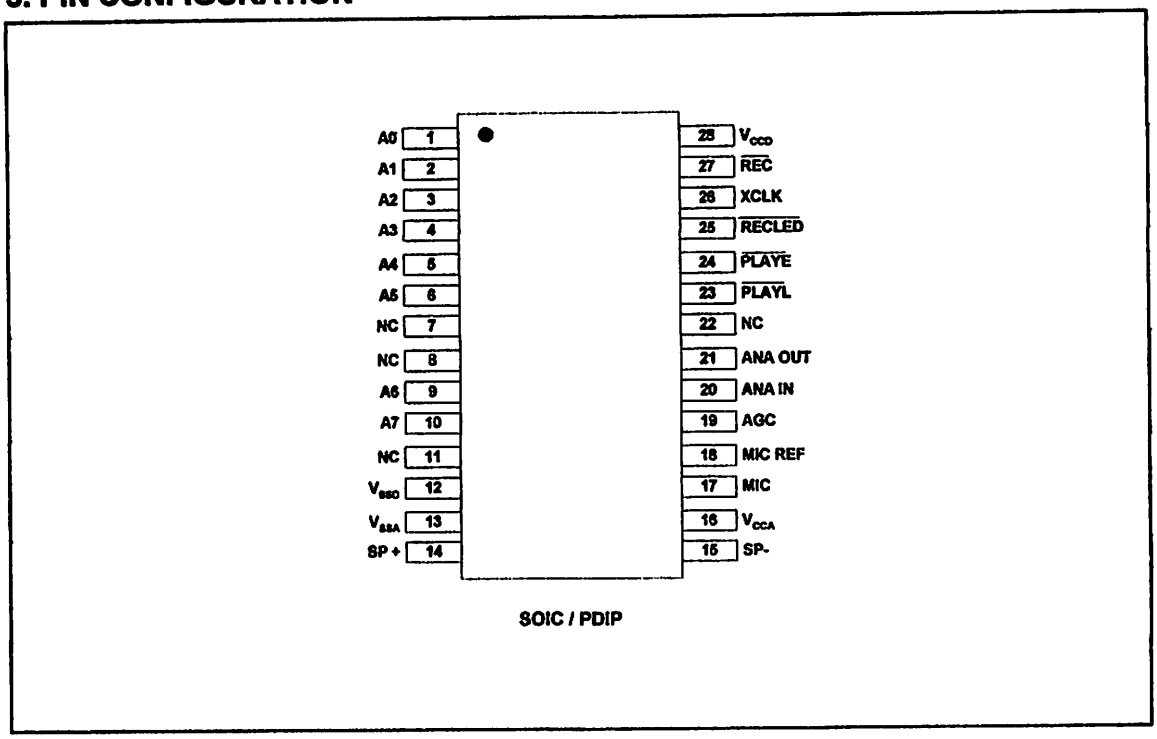
- Single +5 volt power supply
- Duration: 14 and 20 seconds.
- Easy-to-use single-chip, voice record/playback solution
- High-quality, natural voice/audio reproduction
- Manual switch or microcontroller compatible Playback can be edge- or level-activated
- Directly cascadable for longer durations
- Automatic power-down (push-button mode)
    - Standby current 1 µA (typical)
- Zero-power message storage
    - Eliminates battery backup circuits
- Fully addressable to handle multiple messages
- 100-year message retention (typical)
- 100,000 record cycles (typical)
- On-chip oscillator
- Programmer support for play-only applications
- Available in die, PDIP and SOIC
- Temperature:
    - Commercial - Packaged unit : 0°C to 70°C, Die : 0°C to 50°C
    - Industrial - Packaged unit : -40°C to 85°C

3. BLOCK DIAGRAM



4. TABLE OF CONTENTS

5. PIN CONFIGURATION



SOIC / PDIP

Note: NC means must be No connect

6. PIN DESCRIPTION

PIN NAME	PIN NO	FUNCTION
A0-A7	1-6, 9, 10	**Address Inputs**: The address inputs have two functions, depending on the level of the two Most Significant Bits (MSB) of the address. If either or both of the two MSBs are LOW, the inputs are all interpreted as address bits and are used as the start address for the current record or playback cycle. The address pins are inputs only and do not output internal address information as the operation progresses. Address inputs are latched by the falling edge of $\overline{\text{PLAYE}}$, $\overline{\text{PLAYL}}$, or $\overline{\text{REC}}$. If both A6 & A7 are HIGH, then the device is in special operational modes. Please refer to operational modes section for details.
NC	7, 8, 11, 22	**NC:** No Connect
V_{SSD}, V_{SSA}	12, 13	**Ground**: Similar to V_{CCA} and V_{CCD}, the analog and digital circuits internal to the ISD1400 series use separate ground buses to minimize noise. These pins should be tied together as close as possible to the device.
SP+, SP-	14, 15	**Speaker Outputs**: The SP+ and SP- pins provide direct drive for loudspeakers with impedances as low as 16 Ω. A single output may be used, but, for direct-drive loudspeakers, the two opposite-polarity outputs provide an improvement in output power of up to four times over a single-ended connection. Forthermore, when SP+ and SP- are used, a speakercoupling capacitor is not required. A single-ended connection will require an AC-coupling capacitor between the SP pin and the speaker. The speaker outputs are in a high-impedance state during a record cycle, and held at V_{SSA} during power down.
V_{CCA}, V_{CCD}	16, 28	**Supply Voltage**: Analog and digital circuits internal to the ISD1400 series use separate power buses to minimize noise on the chip. These voltage buses are brought out to separate pins on the package and should be tied together as close to the supply as possible. It is important that the power supply be decoupled as close to the package as possible.
MIC	17	**Microphone**: The microphone input transfers its signal to the on-chip preamplifier. An on-chip Automatic Gain Control (AGC) circuit controls the gain of this preamplifier from –15 to 24dB. An external microphone should be AC coupled to this pin via a series capacitor. The capacitor value, together with the internal 10 KΩ resistance on this pin, determines the low-frequency cutoff for the ISD1400 series passband. See Winbond's Application Information for additional information on low-frequency cutoff calculation.

PIN NAME	PIN NO	FUNCTION
MIC REF	18	**Microphone Reference**: The MIC REF input is the inverting input to the microphone preamplifier. This provides a noise-canceling or common-mode rejection input to the device when connected to a differential microphone.
AGC	19	**Automatic Gain Control (AGC)**: The AGC dynamically adjusts the gain of the preamplifier to compensate for the wide range of microphone input levels. The AGC allows the full range of sound, from whispers to loud sounds, to be recorded with minimal distortion. The "attack" time is determined by the time constant of a 5 KΩ internal resistance and an external capacitor (C6 on the schematic of section 11, Figure 5) connected from the AGC pin to V_{SSA} analog ground. The "release" time is determined by the time constant of an external resistor (R5) and an external capacitor (C6) connected in parallel between the AGC pin and V_{SSA} analog ground. Nominal values of 470 KΩ and 4.7 µF give satisfactory results in most cases.
ANA IN	20	**Analog Input**: The analog input pin transfers its signal to the chip for recording. For microphone inputs, the ANA OUT pin should be connected via an external capacitor to the ANA IN pin. This capacitor value, together with the 3.0 KΩ input impedance of ANA IN, is selected to give additional cutoff at the low-frequency end of the voice passband. If the desired input is derived from a source other than a microphone, the signal can be fed, capacitively coupled, into the ANA IN pin directly.
ANA OUT	21	**Analog Output**: This pin provides the preamplifier output to the user. The voltage gain of the preamplifier is determined by the voltage level at the AGC pin.
$\overline{\text{PLAYL}}$ [2]	23	**Playback, Level-Activated**: When this input signal is held LOW, a playback cycle is initiated, and playback continues until $\overline{\text{PLAYL}}$ is pulled HIGH, or an EOM marker is detected. The device automatically powers down and enters into standby mode upon completion of a playback cycle.
$\overline{\text{PLAYE}}$ [2]	24	**Playback, Edge-Activated**: When a LOW-going transition is input to this pin, a playback cycle begins. Taking $\overline{\text{PLAYE}}$ HIGH during a playback cycle will not terminate the current cycle. Playback continues until an EOM is encountered. Upon completion of a playback cycle, the device automatically powers down and enters into standby mode.

<table>
<tr><th>PIN NAME</th><th>PIN NO</th><th>FUNCTION</th></tr>
<tr><td>$\overline{\text{RECLED}}$</td><td>25</td><td>Record LED: The $\overline{\text{RECLED}}$ output is LOW during a record cycle. It can be used to drive an LED to indicate a record cycle is in progress. In addition, $\overline{\text{RECLED}}$ pulses LOW momentarily when an end-of-message is encountered in a playback operation.</td></tr>
<tr><td>XCLK</td><td>26</td><td>External Clock: The input has an internal pull-down device. The ISD1400 is configured at the factory with an internal sampling clock frequency that guarantees its minimum nominal record/playback time. For instance, an ISD1420 operating within specification will be observed to always have a minimum of 20 seconds of recording time. The sampling frequency is then maintained to a variation of ±2.25 percent over the commercial temperature and operating voltage ranges, while still maintaining the minimum specified recording duration. This will result in some devices having a few percent more than nominal recording time.

The Internal clock has a ±5 percent tolerance over the industrial temperature and voltage range. A regulated power supply is recommended for industrial temperature parts. If greater precision is required, the device can be clocked through the XCLK pin as follows:

EXTERNAL CLOCK SAMPLE RATES
<table><tr><th>Part Number</th><th>Sample Rate</th><th>Required Clock</th></tr><tr><td>ISD1416</td><td>8.0 kHz</td><td>1024 kHz</td></tr><tr><td>ISD1420</td><td>6.4 kHz</td><td>819.2 kHz</td></tr></table>
These recommended clock rates should not be varied because the antialiasing and smoothing filters are fixed, and aliasing problems can occur if the sample rate differs from the one recommended. The duty cycle on the input clock is not critical, as the clock is immediately divided by two. If the XCLK is not used, this input must be connected to ground.</td></tr>
</table>

PIN NAME	PIN NO	FUNCTION
$\overline{REC}$	27	**Record Input**: The $\overline{REC}$ input is an active-LOW record signal. The device records whenever $\overline{REC}$ is LOW. This signal must remain LOW for the duration of the recording. $\overline{REC}$ takes precedence over either playback ($\overline{PLAYE}$ or $\overline{PLAYL}$) signal. If $\overline{REC}$ is pulled LOW during a playback cycle, the playback immediately ceases and recording begins. A record cycle is completed when $\overline{REC}$ is pulled HIGH or the memory space is filled. And end-of-message marker (EOM) is internally recorded, enabling a subsequent playback cycle to terminate appropriately. The device automatically powers down to standby mode when $\overline{REC}$ goes HIGH.

Notes:

[1] The $\overline{REC}$ signal is debounced for 50 ms on the rising edge to prevent a false retriggering from a push-button switch.

[2] During playback, if either $\overline{PLAYE}$ or $\overline{PLAYL}$ is held LOW during EOM or OVF, the device will still enter into standby mode and the internal oscillator and timing generator will stop. However, the rising edge of $\overline{PLAYE}$ and $\overline{PLAYL}$ are not debounced and any subsequent falling edge (particularly switch bounce) present on the input pins will initiate another playback.

7. FUNCTIONAL DESCRIPTION

7.1. DETAILED DESCRIPTION

Speech/Sound Quality

The Winbond's ISD1400 series offer 6.4 and 8.0 kHz sampling frequencies, allowing the user a choice of speech quality options. The speech samples are stored directly into on-chip non-volatile memory without the digitization and compression associated with other solutions. Direct analog storage provides a very true, natural sounding reproduction of voice, music, tones, and sound effects not available with most solidstate digital solutions.

Duration

To meet end system requirements, the ISD1400 series offer single-chip solutions at 16 and 20 seconds.

TABLE 1: ISD1400 SERIES SUMMARY

Part Number	Duration (Seconds)	Input Sample Rate (kHz)	Typical Filter Pass Band* (kHz)
ISD1416	16	8.0	3.3
ISD1420	20	6.4	2.6

* 3dB roll-off-point

EEPROM Storage

One of the benefits of Winbond's ChipCorder® technology is the use of on-chip non-volatile memory, providing zero-power message storage. The message is retained for up to 100 years typically without power. In addition, the device can be re-recorded typically over 100,000 times.

Basic Operation

The ISD1400 ChipCorder® series are controlled by a single control signal, $\overline{\text{REC}}$, $\overline{\text{PLAYE}}$ (edge-activated playback) or $\overline{\text{PLAYL}}$ (level-activated playback). The ISD1400 parts are configured for simplicity of design in a single/multiple-message application. Using the address lines will allow multiple message applications.

Automatic Power-Down Mode

At the end of a playback or record cycle, the ISD1400 series automatically return to a low-power standby mode, consuming typically 0.5 µA. After a playback cycle, the device powers down automatically at the end of the message. After a record cycle, the device powers down immediately after $\overline{\text{REC}}$ is pulled to HIGH.

Addressing

In addition to providing single message application, the ISD1400 series provide a full addressing capability.

The ISD1400 series have 160 distinct addressable segments, providing the below resolutions. See Application Information for ISD1400 address tables.

TABLE 2: DEVICE PLAYBACK/RECORD DURATIONS

Part Number	Minimum Duration (Seconds)
ISD1416	100 ms
ISD1420	125 ms

7.2. OPERATIONAL MODES

The ISD1400 series have several built-in operational modes providing maximum functionality with a minimal additional components. The operational modes use the address pins, but are mapped to outside the normal address range. When the two Most Significant Bits (MSBs), A6 and A7, are HIGH, the remaining address signals are interpreted as mode bits and not as address bits. Therefore, operational modes and direct addressing are not compatible and cannot be used simultaneously.

There are two important considerations for using operational modes. Firstly, all operations begin initially at address 0, which is the beginning address. Later operations can begin at other address locations, depending on the operational mode(s) chosen. In addition, the address pointer is reset to 0 when the device is changed from record to playback but not from playback to record when A4 is HIGH in Operational Mode.

Secondly, an Operational Mode is executed when any of the control inputs, $\overline{\text{PLAYE}}$, $\overline{\text{PLAYL}}$ or $\overline{\text{REC}}$, goes LOW and the two MSBs are HIGH. This Operational Mode remains in effect until the next LOW-going control input signal, at which point the current address/mode levels are sampled and executed.

7.2.1. Operational Modes Description

The Operational Modes can be used in conjunction with a microcontroller, or they can be hardwired to provide the desired system operation.

A0 – Message Cueing

Message Cueing allows the user to skip through messages, without knowing the actual physical addresses of each message. Each LOW pulse causes the internal address pointer to skip to the next message. This mode is used for playback only and typically used with the A4 Operational Mode.

A1 – Delete $\overline{\text{EOM}}$ Markers

The A1 Operational Mode allows recording messages sequentially and playback as a single message with only one $\overline{\text{EOM}}$ set at the end of the final message.

A2 – Unused

A3 – Message Looping

The A3 Operational Mode allows repeating playback a message continuously from the beginning of the memory. A message can completely fill the ISD1400 device and will loop from beginning to end. Pulsing $\overline{\text{PLAYE}}$ will start the playback and pulsing $\overline{\text{PLAYL}}$ will end the playback.

A4 – Consecutive Addressing

During normal operation, the address pointer will reset when a message is played through to an $\overline{\text{EOM}}$ marker. The A4 Operational Mode inhibits the address pointer reset, allowing messages to be recorded or played back consecutively. When the device is in a static state; i.e., not recording or playback, momentarily taking this pin LOW will reset the address counter to zero.

A5 – Unsued

TABLE 3: OPERATIONAL MODES

Mode	Function	Typical Use	Jointly Compatible [1]
A0	Message cueing	Fast-forward through messages	A4
A1	Delete $\overline{\text{EOM}}$ markers	Position $\overline{\text{EOM}}$ marker at the end of the last message	A3, A4
A2	Unused		
A3	Looping	Continuous playback from Address 0	A1
A4	Consecutive addressing	Record/playback multiple consecutive messages	A0, A1
A5	Unused		

[1] Additional Operational Modes can be used simultaneously with the given mode.

8. TIMING DIAGRAMS

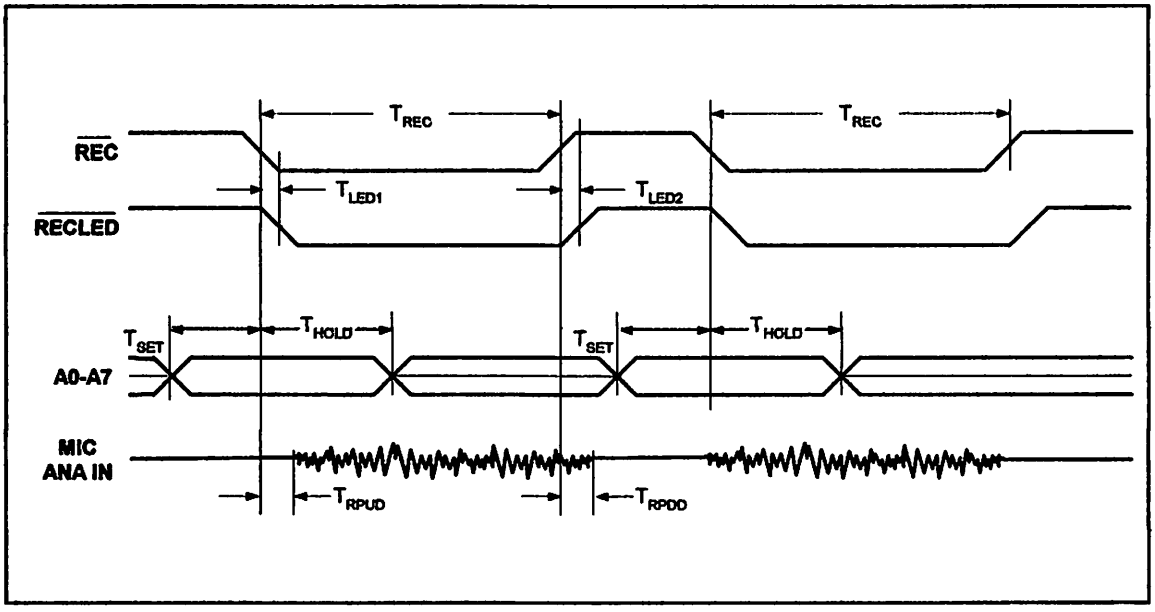


FIGURE 1: RECORD

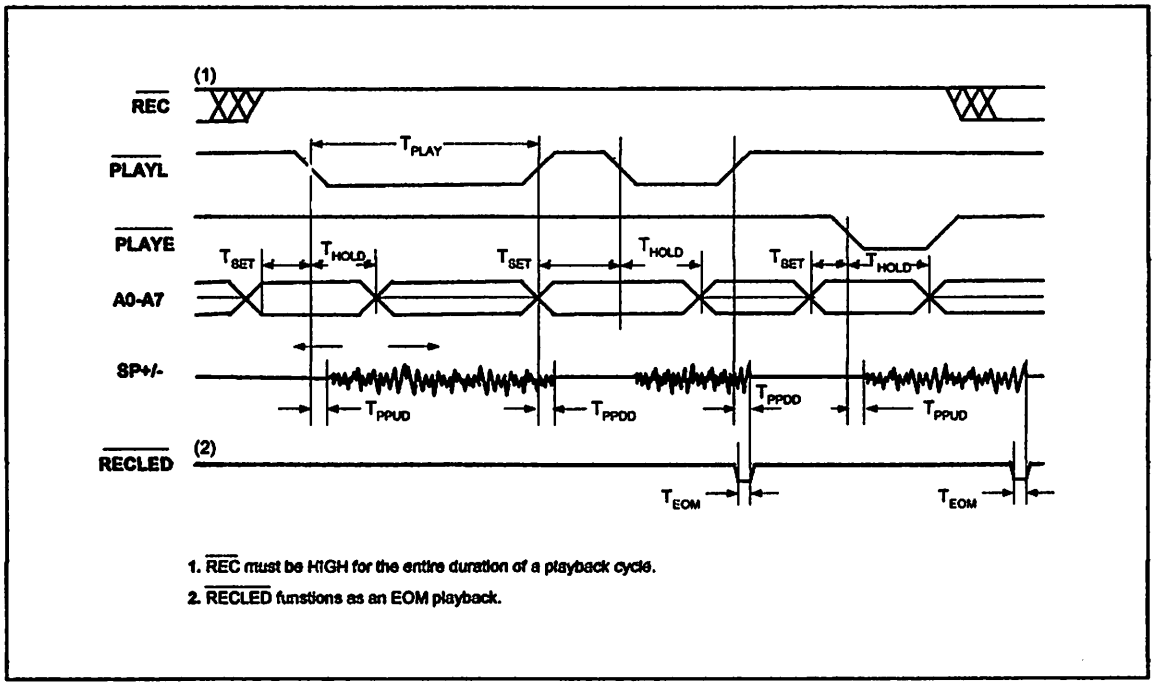


FIGURE 2: PLAYBACK

9. ABSOLUTE MAXIMUM RATINGS[2]

TABLE 4: ABSOLUTE MAXIMUM RATINGS (PACKAGED PARTS)

CONDITIONS	VALUES
Junction temperature	150°C
Storage temperature range	-65°C to +150°C
Voltage applied to any pin	(V_{SS} – 0.3V) to (V_{CC} + 0.3V)
Voltage applied to any pin (Input current limited to ±20 mA)	(V_{SS} – 1.0V) to (V_{CC} + 1.0V)
Lead temperature (Soldering – 10sec)	300°C
V_{CC} – V_{SS}	-0.3V to +7.0V

TABLE 5: ABSOLUTE MAXIMUM RATINGS (DIE)

CONDITIONS	VALUES
Junction temperature	150°C
Storage temperature range	-65°C to +150°C
Voltage applied to any pad	(V_{SS} – 0.3V) to (V_{CC} + 0.3V)
Voltage applied to any pad (Input current limited to ±20mA)	(V_{SS} – 1.0V) to (V_{CC} + 1.0V)
Lead Temperature (soldering 10 seconds)	330° C
V_{CC} – V_{SS}	-0.3V to +7.0V

[2] Stresses above those listed may cause permanent damage to the device. Exposure to the absolute maximum ratings may affect device reliability and performance. Functional operation is not implied at these conditions.

9.1 OPERATING CONDITIONS

TABLE 6: OPERATING CONDITIONS (PACKAGED PARTS)

CONDITIONS	VALUES
Commercial operating temperature range (Case temperature)	0°C to +70°C
Industrial operating temperature (Case temperature)	-40°C to +85°C
Supply voltage (V_{CC}) [1]	+4.5V to +5.5V
Ground voltage (V_{SS}) [2]	0V

TABLE 7: OPERATING CONDITIONS (DIE)

CONDITIONS	VALUES
Commercial operating temperature range	0°C to +50°C
Supply voltage (V_{CC}) [1]	+4.5V to +6.5V
Ground voltage (V_{SS}) [2]	0V

[1] $V_{CC} = V_{CCA} = V_{CCD}$

[2] $V_{SS} = V_{SSA} = V_{SSD}$

10. ELECTRICAL CHARACTERISTICS

10.1. PARAMETERS FOR PACKAGED PARTS

TABLE 8: DC PARAMETERS

PARAMETERS	SYMBOLS	MIN[2]	TYP[1]	MAX[2]	UNITS	CONDITIONS
Input Low Voltage	V_{IL}			0.8	V	
Input High Voltage	V_{IH}	2.4			V	
Output Low Voltage	V_{OL}			0.4	V	I_{OL} = 4.0 mA
Output High Voltage	V_{OH}	2.4			V	I_{OH} = -1.6 mA
V_{CC} Current (Operating)	I_{CC}		15	30	mA	V_{CC} = 5.5V[3], R_{EXT} = ∞
V_{CC} Current (Standby)	I_{SB}		0.5	10	µA	[3] [4]
Input Leakage Current	I_{IL}			±1	µA	
Input Current HIGH w/Pull Down	I_{ILPD}			130	µA	Force V_{CC} [5]
Output Load Impedance	R_{EXT}	16			Ω	Speaker Load
Preamp IN Input Resistance	R_{MIC}	4	9	17	KΩ	Pins 17, 18
ANA IN Input Resistance	$R_{ANA\ IN}$	2.5	3	5	KΩ	
Preamp Gain 1	A_{PRE1}	20	23	26	dB	AGC = 0.0V
Preamp Gain 2	A_{PRE2}		-45	-15	dB	AGC = 2.5V
ANA IN to SP+/- Gain	A_{ARP}	20	22	25	dB	
AGC Output Resistance	R_{AGC}	2.5	5	9.5	KΩ	
Preamp Out Source	I_{PREH}		-2		mA	@ V_{OUT} = 1.0V
Preamp In Sink	I_{PREL}		0.5		mA	@ V_{OUT} = 2.0V

[1] Typical values @ T_A = 25° and 5.0V.

[2] All Min/Max limits are guaranteed by Winbond via electronical testing or characterization. Not all specifications are 100 percent tested.

[3] V_{CCA} and V_{CCD} connected together.

[4] $\overline{REC}$, $\overline{PLAYL}$, and $\overline{PLAYE}$ must be at V_{CCD}.

[5] XCLK pin.

TABLE 9: AC PARAMETERS

CHARACTERISTICS	SYMBOLS	MIN[2]	TYP[1]	MAX[2]	UNITS	CONDITIONS
Sampling Frequency	F_S			8.0		
ISD1416				6.4	kHz	[5]
ISD1420					kHz	[5]
Filter Pass Band	F_{CF}					
ISD1416			3.3		kHz	3 dB Roll-Off Point[3][6]
ISD1420			2.6		kHz	3 dB Roll-Off Point[3][6]
Record Duration	T_{REC}					
ISD1416		16			sec	
ISD1420		20			sec	
Playback Duration	T_{PLAY}					
ISD1416		16			sec	[5]
ISD1420		20			sec	[5]
$\overline{\text{RECLED}}$ ON Delay	T_{LED1}		5		msec	
$\overline{\text{RECLED}}$ OFF Delay	T_{LED2}	30	38.9	95	msec	
ISD1416		40	48.6	110	msec	
ISD1420						
Address Setup Time	T_{SET}	300			nsec	
Address Hold Time	T_{HOLD}	0			nsec	
Record Power-Up Delay	T_{RPUD}					
ISD1416			26		msec	
ISD1420			32		msec	
Record Power-Down Delay	T_{RPDD}					
ISD1416			26		msec	
ISD1420			32		msec	
Play Power-Up Delay	T_{PPUD}					
ISD1416			26		msec	
ISD1420			32		msec	
Play Power-Down Delay	T_{PPDD}		6.5			
ISD1416			8.1		msec	
ISD1420					msec	

CHARACTERISTICS	SYMBOLS	MIN[2]	TYP[1]	MAX[2]	UNITS	CONDITIONS
$\overline{EOM}$ Pulse Width ISD1416 ISD1420	T_{EOM}		 12.5 15.625		 msec msec	
Total Harmonic Distortion	THD		1	3	%	@ 1 kHz
Speaker Output Power	P_{OUT}		12.2		mW	R_{EXT} = 16 Ω
Voltage Across Speaker Pins	V_{OUT}		1.25	2.5	V p-p	R_{EXT} = 600 Ω
MIC Input Voltage	V_{IN1}			20	mV	Peak-to-Peak[5]
ANA IN Input Voltage	V_{IN2}			50	mV	Peak-to-Peak

Notes:

[1] Typical values @ T_A = 25° and 5.0V.

[2] All Min/Max limits are guaranteed by Winbond via electronical testing or characterization. Not all specifications are 100 percent tested.

[3] Low-frequency cutoff depends upon the value of external capacitors (see Pin Descriptions)

[4] With 5.1 K Ω series resistor at ANA IN.

[5] Sampling Frequency and playback duration can vary as much as ±2.25 percent over the commercial temperature and voltage rangs. It may vary as much as ±5 percent over the industrial temperature and voltage ranges. All devices will meet the maximum sampling frequency and minimum playback duration parameters. For greater stability, an external clock can be utilized (see Pin Descriptions)

[6] Filter specification applies to the anti-aliasing filter and the smoothing filter. Typical Parameter Variation with Voltage and Temperature. This parameter is not checked during production testing and may vary due to process variations and other factors. Therefore, the customer should not rely upon this value for testing purposes.

10.1.1. Typical Parameter Variation with Voltage and Temperature

Chart 1: Record Mode Operating Current (I_{CC})

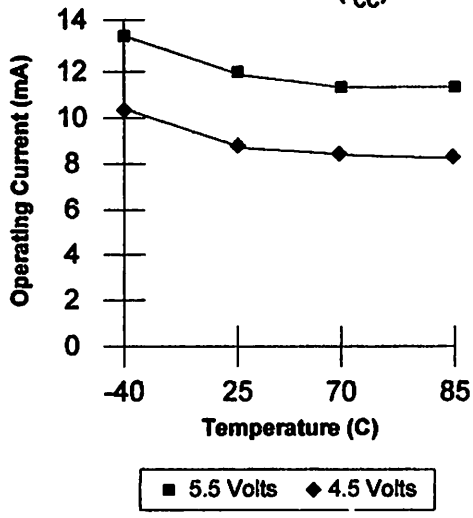


Chart 3: Standby Current (I_{SB})

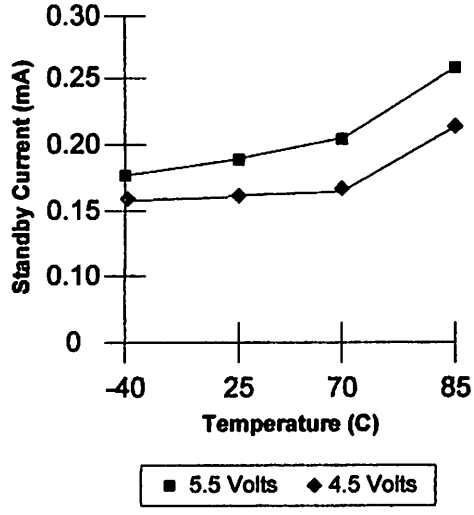


Chart 2: Total Harmonic Distortion

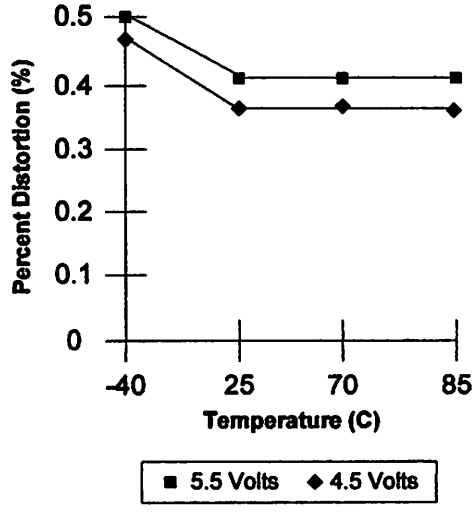
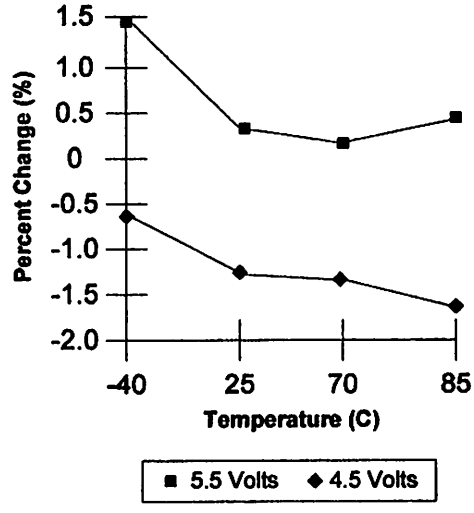


Chart 4: Oscillator Stability



10.2. PARAMETERS FOR DIE

TABLE 10: DC PARAMETERS

PARAMETERS	SYMBOLS	MIN[2]	TYP[1]	MAX[2]	UNITS	CONDITIONS
Input Low Voltage	V_{IL}			0.8	V	
Input High Voltage	V_{IH}	2.4			V	
Output Low Voltage	V_{OL}			0.4	V	I_{OL} = 4.0 mA
Output High Voltage	V_{OH}	2.4			V	I_{OH} = -1.6 mA
V_{CC} Current (Operating)	I_{CC}		15	30	mA	V_{CC} = 5.5V[3], R_{EXT} = ∞
V_{CC} Current (Standby)	I_{SB}		0.5	10	µA	[3] [4]
Input Leakage Current	I_{IL}			±1	µA	
Input Current HIGH w/Pull Down	I_{ILPD}			130	µA	Force V_{CC} [5]
Output Load Impedance	R_{EXT}	16			Ω	Speaker Load
Preamp IN Input Resistance	R_{MIC}	4	9	17	KΩ	Pads 17,18
ANA IN Input Resistance	$R_{ANA\ IN}$	2.5	3	5	KΩ	
Preamp Gain 1	A_{PRE1}	20	23	26	dB	AGC = 0.0V
Preamp Gain 2	A_{PRE2}		-45	-15	dB	AGC = 2.5V
ANA IN to SP+/- Gain	A_{ARP}	20	22	25	dB	
AGC Output Resistance	R_{AGC}	2.5	5	9.5	KΩ	
Preamp Out Source	I_{PREH}		-2		mA	@ V_{OUT} = 1.0V
Preamp In Sink	I_{PREL}		0.5		mA	@ V_{OUT} = 2.0V

[1] Typical values @ T_A = 25° and 5.0V.

[2] All Min/Max limits are guaranteed by Winbond via electronical testing or characterization. Not all specifications are 100 percent tested.

[3] V_{CCA} and V_{CCD} connected together.

[4] $\overline{REC}$, $\overline{PLAYL}$, and $\overline{PLAYE}$ must be at V_{CCD}.

[5] XCLK pin.

CHARACTERISTICS	SYMBOLS	MIN[2]	TYP[1]	MAX[2]	UNITS	CONDITIONS
$\overline{EOM}$ Pulse Width ISD1416 ISD1420	T_{EOM}		 12.5 15.625		 msec msec	
Total Harmonic Distortion	THD		1	3	%	@ 1 kHz
Speaker Output Power	P_{OUT}		12.2		mW	R_{EXT} = 16 Ω[4]
Voltage Across Speaker Pins	V_{OUT}		1.25	2.5	V p-p	R_{EXT} = 600 Ω
MIC Input Voltage	V_{IN1}			20	mV	Peak-to-Peak[4]
ANA IN Input Voltage	V_{IN2}			50	mV	Peak-to-Peak

Notes:

[1] Typical values @ T_A = 25° and 5.0V.

[2] All Min/Max limits are guaranteed by Winbond via electronical testing or characterization. Not all specifications are 100 percent tested.

[3] Low-frequency cutoff depends upon the value of external capacitors (see Pin Descriptions)

[4] With 5.1 K Ω series resistor at ANA IN.

[5] Sampling Frequency and playback duration can vary as much as ±2.25 percent over the commercial temperature and voltage rangs. It may vary as much as ±5 percent over the industrial temperature and voltage ranges. All devices will meet the maximum sampling frequency and minimum playback duration parameters. For greater stability, an external clock can be utilized (see Pin Descriptions)

[6] Filter specification applies to the anti-aliasing filter and the smoothing filter. Typical Parameter Variation with Voltage and Temperature. This parameter is not checked during production testing and may vary due to process variations and other factors. Therefore, the customer should not rely upon this value for testing purposes.

10.2.1. Typical Parameter Variation with Voltage and Temperature

Chart 5: Record Mode Operating Current (I_{CC})

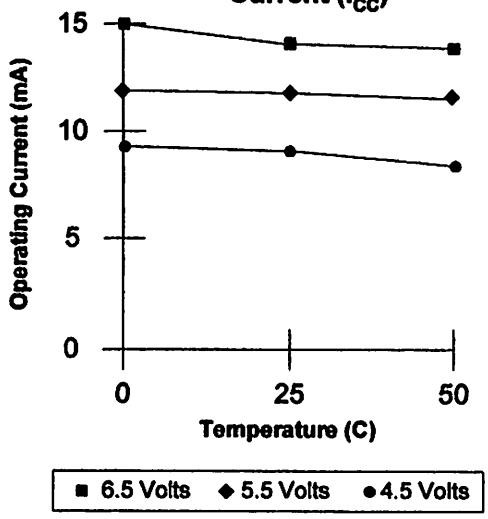


Chart 7: Standby Current (I_{SB})

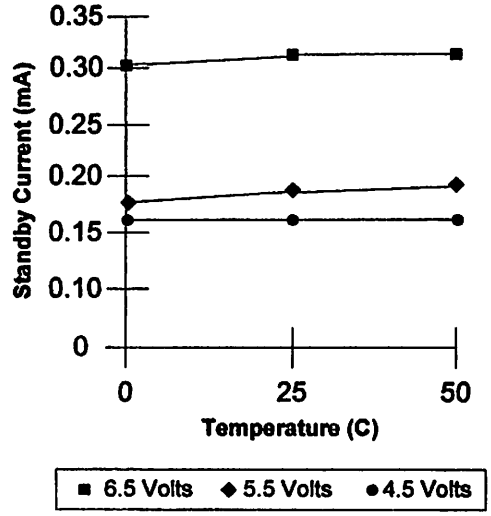


Chart 6: Total Harmonic Distortion

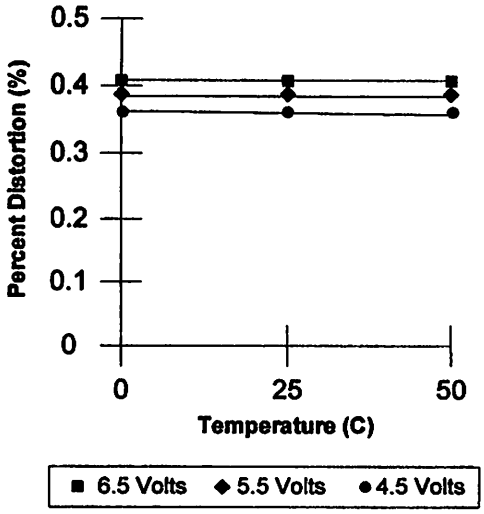
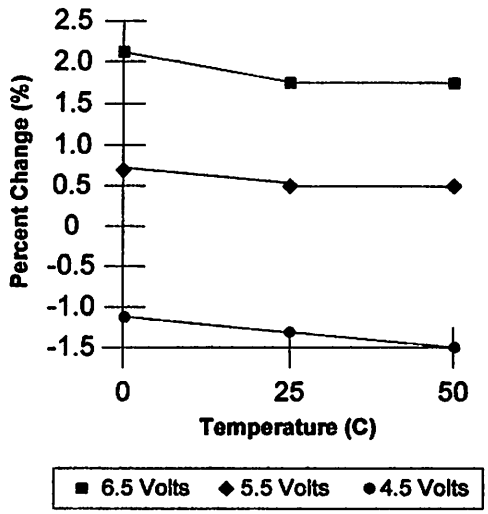


Chart 8: Oscillator Stability



11. TYPICAL APPLICATION CIRCUIT

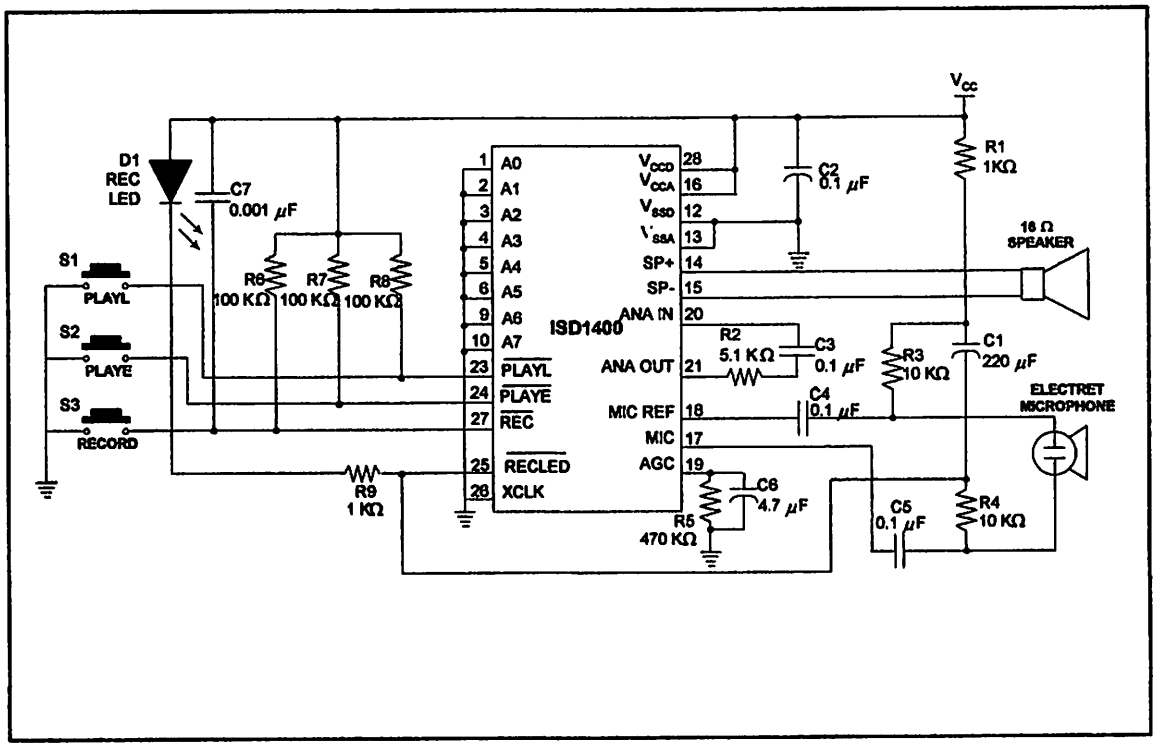


FIGURE 5: DESIGN SCHEMATIC

Functional Description Example

The following operating examples demonstrate the functionality of the ISD1400 series.

1. Record a message:

   Pulling the $\overline{\text{REC}}$ signal LOW initiates a record cycle from current location. When $\overline{\text{REC}}$ is held LOW, the recording continues. Until the memory array is filled up or when $\overline{\text{REC}}$ is pulled HIGH, recording ceases. An EOM marker is written at the end of message. Then the device will automatically power down.

2. Edge-activated playback:

   Pulling the $\overline{\text{PLAYE}}$ signal LOW initiates a playback cycle from the beginning of the message until the entire message is played. The rising edge of $\overline{\text{PLAYE}}$ has no effect on operation. When the EOM marker is encountered, the device automatically powers down. A subsequent falling edge on $\overline{\text{PLAYE}}$ initiates a new playback operation from the beginning of the message.

3. Level-activated playback:

   Holding the $\overline{\text{PLAYL}}$ signal LOW initiates a playback cycle from the beginning of the message, until $\overline{\text{PLAYL}}$ is pulled HIGH or when the EOM marker is encountered, playback operation stops and the device automatically powers down.

4. Record (interrupting playback).

   The $\overline{\text{REC}}$ signal takes precedence over playback operation. Holding $\overline{\text{REC}}$ LOW initiates a new record operation from current location, regardless of any current operation in progress.

5. $\overline{\text{RECLED}}$ operation.

   During record, the $\overline{\text{RECLED}}$ output pin provides an active-LOW signal, which can be used to drive an LED as a "record-in-progress" indicator. It returns to a HIGH state when the $\overline{\text{REC}}$ pin is pulled HIGH or when the recording is completed due to the memory being filled. However, during playback, this pin also pulses LOW to indicate an EOM at the end of a message.

Applications Note

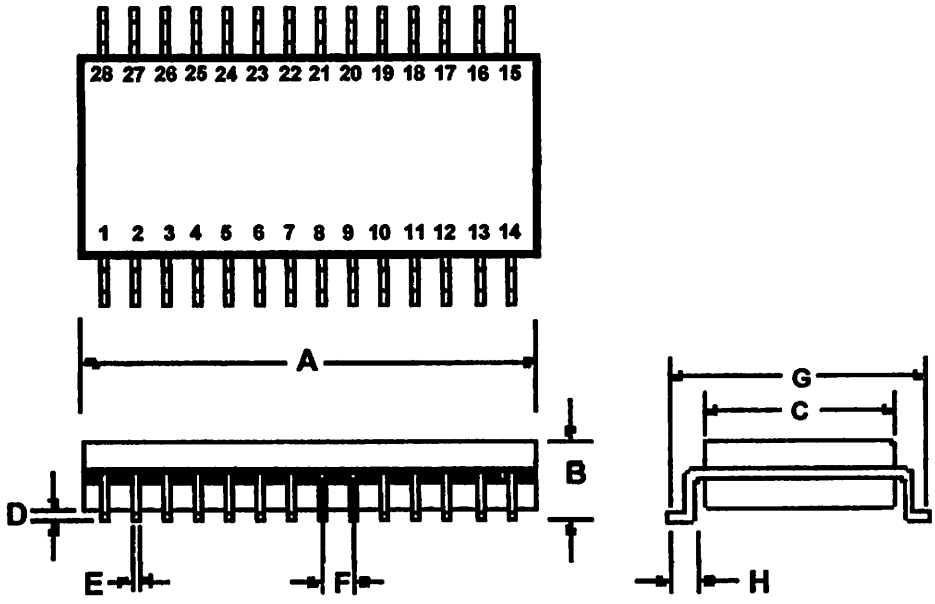
Some users may experience an unexpected recording taking place when their circuit is powered up, or the batteries are changed and V_{CC} rises faster than $\overline{REC}$. This undesired recording prevents playback of the previously recorded message. A spurious End Of Message (EOM) marker appears at the very beginning of the memory, preventing access to the original message, and nothing is played.

To prevent this occurrence, place a capacitor (approx. 0.001 µF) between the control pin ($\overline{REC}$) and V_{CC}. This pulls the control pin voltage up with V_{CC} as it rises. Once the voltage is HIGH, the pull-up device will keep the pin HIGH until intentionally pulled LOW, preventing the false EOM marker.

Since this anomaly depends on factors such as the capacitance of the user's printed circuit board, not all circuit designs will exhibit the spurious marker. However, it is recommended that the capacitor is included for design reliability. A more detailed explanation and resolution of this occurrence is described in Application Information.

12. PACKAGE DRAWING AND DIMENSIONS

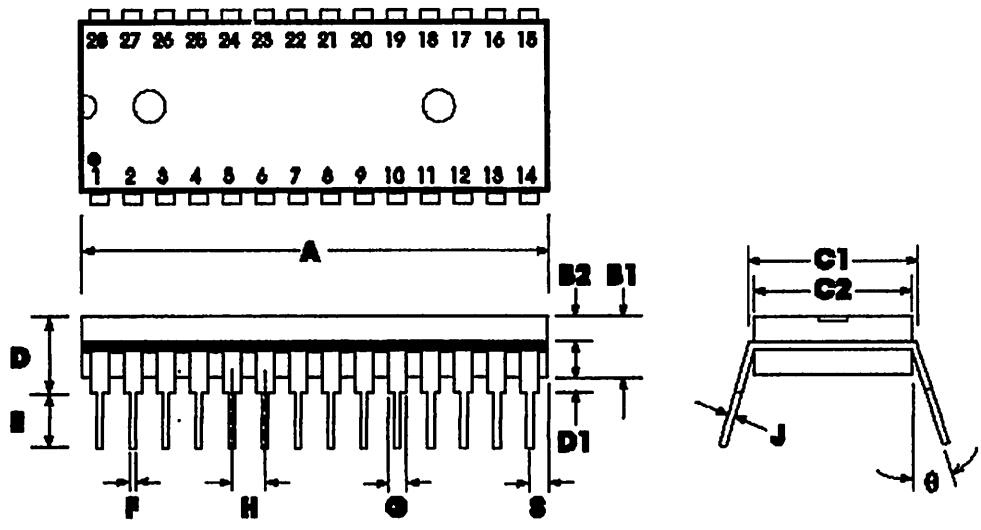
12.1. 28-LEAD 300 MIL PLASTIC SMALL OUTLINE IC (SOIC)



	INCHES			MILLIMETERS		
	Min	Nom	Max	Min	Nom	Max
A	0.701	0.706	0.711	17.81	17.93	18.06
B	0.097	0.101	0.104	2.46	2.56	2.64
C	0.292	0.296	0.299	7.42	7.52	7.59
D	0.005	0.009	0.0115	0.127	0.22	0.29
E	0.014	0.016	0.019	0.35	0.41	0.48
F		0.050			1.27	
G	0.400	0.406	0.410	10.16	10.31	10.41
H	0.024	0.032	0.040	0.61	0.81	1.02

Note: Lead coplanarity to be within 0.004 inches.

12.2. 28-LEAD 600 MIL PLASTIC DUAL INLINE PACKAGE (PDIP)

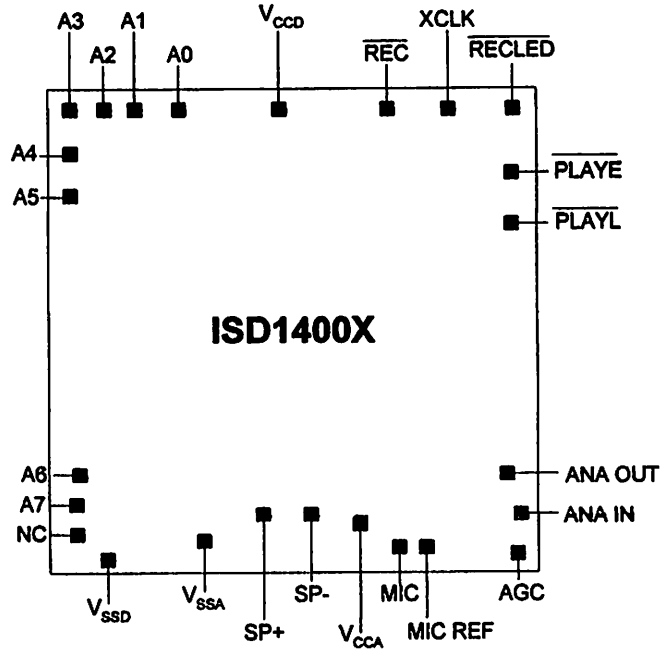


	INCHES			MILLIMETERS		
	Min	Nom	Max	Min	Nom	Max
A	1.445	1.450	1.455	36.70	36.83	36.96
B1		0.150			3.81	
B2	0.065	0.070	0.075	1.65	1.78	1.91
C1	0.600		0.625	15.24		15.88
C2	0.530	0.540	0.550	13.46	13.72	13.97
D			0.19			4.83
D1	0.015			0.38		
E	0.125		0.135	3.18		3.43
F	0.015	0.018	0.022	0.38	0.46	0.56
G	0.055	0.060	0.065	1.40	1.52	1.62
H		0.100			2.54	
J	0.008	0.010	0.012	0.20	0.25	0.30
S	0.070	0.075	0.080	1.78	1.91	2.03
q	0°		15°	0°		15°

12.3. DIE PHYSICAL LAYOUT [1]

ISD1400x

- Die Dimensions
  - X: 172.2 ± 1 mils
  - Y: 168.5 ± 1 mils
- Die Thickness[2]
  - 17.5 ± 1 mils
- Pad Opening
  - 100 x 112 microns
  - 3.9 x 4.4 mils



Notes:

[1] The backside of die is internally connected to V_{SS}. It **MUST NOT** be connected to any other potential or damage may occur.

[2] Die thickness is subject to change, please contact Winbond factory for status and availability.

ISD1400 SERIES PAD DESIGNATIONS

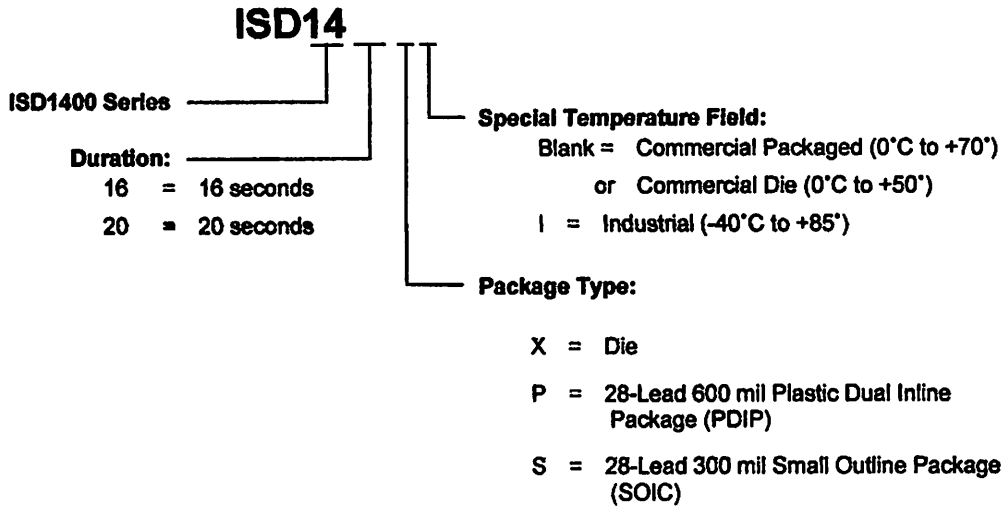
(with respect to die center)

Pad	Pad Name	X Axis (µm)	Y Axis (µm)
A0	Address 0	-1332.5	1973.8
A1	Address 1	-1628.9	1973.8
A2	Address 2	-1808.9	1973.8
A3	Address 3	-2014.1	1910.2
A4	Address 4	-2014.1	1722.6
A5	Address 5	-2014.1	1519.8
A6	Address 6	-2014.1	-1214.6
A7	Address 7	-2014.1	-1399.8
NC	No Connect	-2014.1	-1745.4
V_{SSD}	Digital Ground	-1894.1	-1971.8
V_{SSA}	Analog Ground	-358.1	-1971.8
SP+	Speaker Output +	-17.7	-1896.6
SP-	Speaker Output -	411.9	-1896.6
V_{CCA}	Analog Power Supply	779.5	-1936.2
MIC	Microphone Input	991.5	-1973.8
MIC REF	Microphone Reference	1168.7	-1973.8
AGC	Automatic Gain Control	1977.9	-1910.6
ANA IN	Analog Input	2005.1	-1580.2
ANA OUT	Analog Output	1990.7	-1379.0
$\overline{\text{PLAYL}}$	Level-Activated Playback	2013.9	1608.6
$\overline{\text{PLAYE}}$	Edge-Activated Playback	2013.9	1777.0
$\overline{\text{RECLED}}$	Record LED Output	2011.9	1971.8
XCLK	External Clock	1580.7	1973.8
$\overline{\text{REC}}$	Record	752.3	1973.8
V_{CCD}	Digital Power Supply	-48.5	1929.4

Note: Die dimensions and pad positions may be subjected to change. Please contact Winbond Sales Offices or Representatives to verify current or future specifications.

13. ORDERING INFORMATION

Product Number Descriptor Key



When ordering ISD1400 Series devices, please refer to the following valid part numbers.

Die / Package	16-Second		20-Second	
	Product P/N	Ordering P/N	Product P/N	Ordering P/N
Die	ISD1416X C5006	I1416X5006	ISD1416X C5006	I1420X5006
PDIP	ISD1416P C5006	I1416P5006	ISD1416P C5006	I1420P5006
	ISD1416PI C5006	I1416PI5006	ISD1416PI C5006	I1420PI5006
SOIC	ISD1416S C5006	I1416S5006	ISD1416S C5006	I1420S5006
	ISD1416SI C5006	I1416SI5006	ISD1416SI C5006	I1420SI5006

For the latest product information, access Winbond's worldwide website at
http://www.winbond-usa.com

14. VERSION HISTORY

VERSION	DATE	DESCRIPTION
0	Before 2004	Initial issue.
1.0	March 2004	Reformat the document. Revise footnote for Filter Passband in Tables 1, 9 & 11. Revise Functional Description Example section. Revise die picture. Revise ordering information.

Winbond Electronics Corp.

The contents of this document are provided only as a guide for the applications of Winbond products. Winbond makes no representation or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to discontinue or make changes to specifications and product descriptions at any time without notice. No license, whether express or implied, to any intellectual property or other right of Winbond or others is granted by this publication. Except as set forth in Winbond's Standard Terms and Conditions of Sale, Winbond assumes no liability whatsoever and disclaims any express or implied warranty of merchantability, fitness for a particular purpose or infringement of any Intellectual property.

Winbond products are not designed, intended, authorized or warranted for use as components in systems or equipments intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Further more, Winbond products are not intended for applications wherein failure of Winbond products could result or lead to a situation wherein personal injury, death or severe property or environmental injury could occur.

Application examples and alternative uses of any integrated circuit contained in this publication are for illustration only and Winbond makes no representation or warranty that such applications shall be suitable for the use specified.

ISD® and ChipCorder® are trademarks of Winbond Electronics Corporation. SuperFlash® is the trademark of Silicon Storage Technology, Inc.

The 100-year retention and 100K record cycle projections are based upon accelerated reliability tests, as published in the Winbond Reliability Report, and are neither warranted nor guaranteed by Winbond.

Information contained in this ISD® ChipCorder® data sheet supersedes all data for the ISD ChipCorder products published by ISD® prior to August, 1998.

This data sheet and any future addendum to this data sheet is(are) the complete and controlling ISD® ChipCorder® product specifications. In the event any inconsistencies exist between the information in this and other product documentation, or in the event that other product documentation contains information in addition to the information in this, the information contained herein supersedes and governs such other information in its entirety.

Winbond Electronics Corp.

Headquarters
No. 4, Creation Rd. III
Science-Based Industrial Park,
Hsinchu, Taiwan
TEL: 886-3-5770066
FAX: 886-3-5665577
http://www.winbond.com.tw/

Winbond Electronics Corporation America
2727 North First Street, San Jose,
CA 95134, U.S.A.
TEL: 1-408-9436666
FAX: 1-408-5441797
http://www.winbond-usa.com/

Winbond Electronics (Shanghai) Ltd.
27F, 299 Yan An W. Rd. Shanghai,
200336 China
TEL: 86-21-62365999
FAX: 86-21-62356998

Taipei Office
9F, No. 480, Pueiguang Rd.
Neihu District
Taipei, 114 Taiwan
TEL: 886-2-81777168
FAX: 886-2-87153579

Winbond Electronics Corporation Japan
7F Daini-ueno BLDG. 3-7-18
Shinyokohama Kohokuku,
Yokohama, 222-0033
TEL: 81-45-4781881
FAX: 81-45-4781800

Winbond Electronics (H.K.) Ltd.
Unit 9-15, 22F, Millennium City,
No. 378 Kwun Tong Rd.,
Kowloon, Hong Kong
TEL: 852-27513100
FAX: 852-27552064