

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



SKRIPSI

**PERANCANGAN DAN PEMBUATAN BEL -
KAMERA UNTUK RUMAH KOS BERBASIS
MIKROKONTROLER RENESAS R8C/13 TINY**

Disusun Oleh:
**KOMANG HADI SURYATA
NIM. 02.17.045**

SEPTEMBER 2007

LEMBAR PERSETUJUAN

**PERANCANGAN DAN PEMBUATAN BEL-KAMERA UNTUK RUMAH
KOS BERBASIS MIKROKONTROLER RENESAS R8C/13 TINY**

SKRIPSI


*Disusun dan diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana Teknik Elektronika Strata Satu (S-1)*

Disusun Oleh :

**KOMANG HADI SURYATA
02.17.045**

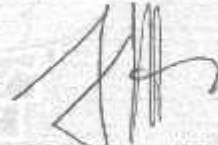
Diperiksa dan disetujui,

Dosen Pembimbing I



**Ir. Eko Nurcahyo.
NIP. Y. 1028700172**

Dosen Pembimbing II



**M. Ashar, ST, MT.
NIP. 1030500408**

Mengetahui,



Ketua Jurusan Teknik Elektro S-1

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

**KONSENTRASI TEKNIK ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2007**



BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : Komang Hadi Suryata
NIM : 02.17.045
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Masa Bimbingan : 22 Mei 2007 – 22 November 2007
Judul Skripsi : Perancangan dan Pembuatan Bel-Kamera Untuk Rumah
Kos Berbasis Mikrokontroler Renesas R8C/13 Tiny
Dipertahankan dihadapan Tim Penguji Skripsi Jenjang Strata Satu (S-1) Pada :
Hari : Kamis
Tanggal : 6 September 2007
Dengan Nilai : 81.5 (A) *81.5*



Panitia Ujian Skripsi

Ketua

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

Sekretaris

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

Anggota Penguji

Penguji Pertama

(Ir. Teguh Herbasuki, MT)
NIP.Y.1038900209

Penguji Kedua

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

ABSTRAKSI

**JUDUL : PERANCANGAN DAN PEMBUATAN BEL-KAMERA PADA
RUMAH KOS BERBASIS MIKROKONTROLER RENESAS
R8C/13 TINY**

Skripsi, Teknik Elektro S-1, Institut Teknologi Nasional Malang.

Disusun Oleh : Komang Hadi Suryata, 02.17.045, 2007

Dosen Pembimbing I : Ir. Eko Nurcahyo

Dosen Pembimbing II : M. Ashar, ST, MT

Kata Kunci : Keypad, ISD 2590, Mikrokontroler Renesas R8C/13 Tiny, Kamera CCTV.

Perkembangan ilmu pengetahuan dan teknologi dewasa ini sangatlah pesat. Seiring dengan perkembangan tersebut, manusia dituntut untuk melaksanakan segala sesuatu bukan hanya cepat, tetapi juga harus tepat. Untuk itu manusia senantiasa mencari cara untuk mempermudah pekerjaannya sehari-hari. Berkaitan dengan hal tersebut, manusia juga cenderung untuk mencari alternatif pemecahan masalah bagi pekerjaan yang dilakukan secara manual sehingga pekerjaan tersebut dapat dikerjakan secara otomatis.

Saat ini masih banyak kos-kos yang belum memakai bel pintu atau ada yang sudah memakainya tetapi masih sederhana, sehingga apabila ada yang datang berkunjung ke kos tersebut kadang-kadang sulit untuk bertemu dengan orang yang dicarinya. Hal ini banyak memakan waktu dan tentunya kurang efisien.

Skripsi ini memuat perancangan dan pembuatan bel –kamera kos yang menggunakan *keypad* dan memakai suara ke masing-masing kamar dengan suara yang berbeda-beda, serta ditambah dengan penggunaan *kamera CCTV*, sehingga apabila ada yang datang berkunjung untuk mencari salah seorang yang kos, maka orang tersebut dapat langsung mengetahui kalau sedang ada yang mencari.

Pertama, penghuni kos tersebut mengetikkan no kamar dan namanya ke *Keypad* dan menyimpannya, kemudian merekam suaranya ke *ISD 2590*. Apabila ada orang yang datang berkunjung ke kos tersebut kemudian mengetikkan no kamar pada *keypad*, kemudian data no kamar tersebut diproses oleh *Mikrokontroler Renesas R8C/13 Tiny* dan apabila data tersebut benar, maka *ISD 2590* akan bekerja dan memutar suara yang telah direkam ke kamar orang yang dicari. Penggunaan *kamera CCTV* adalah untuk melihat orang yang datang berkunjung ke kos tersebut dan akan ditampilkan ke TV di kamar orang yang dicari. *Kamera CCTV* terus terhubung ke TV di ruang tengah sehingga berfungsi juga sebagai kamera pengawas.

Apabila orang yang dicari tersebut tidak ada, maka orang yang datang berkunjung tersebut dapat meninggalkan pesan berupa suara singkat. Dan pesan tersebut dapat didengar apabila penghuni kos tersebut sudah pulang.

KATA PENGANTAR

Puja dan puji syukur kami haturkan kehadiran Hyang Widhi Wasa / Tuhan Yang Maha Esa karena atas berkat dan rahmat-Nya, penyusun dapat menyelesaikan skripsi dengan judul : **“Perencanaan Dan Pembuatan Bel-Kamera Untuk Rumah Kos Berbasis Mikrokontroler Renesas R8C/13 Tiny”**, sesuai dengan waktu yang telah ditentukan.

Tujuan dari penulisan Laporan Skripsi ini adalah sebagai salah satu syarat untuk memperoleh gelar Sarjana pada Jurusan Teknik Elektro S1, Konsentrasi Teknik Elektronika, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.

Usaha menyelesaikan Skripsi ini tidak akan berhasil dengan baik tanpa adanya bimbingan dan bantuan dari berbagai pihak. Oleh karenanya penyusun ingin menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME. selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang
3. Bapak Ir. F. Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro S-1 Institut Teknologi Nasional Malang.
4. Bapak Ir. Eko Nurcahyo, selaku dosen pembimbing I.
5. Bapak M. Ashar, ST, MT selaku dosen pembimbing II.
6. Bapak/Ibu Dosen Program Studi Teknik Elektro.
7. Orang tuaku tersayang, yang selalu menjadi spirit dan motivator bagiku.

8. Teman – teman jurusan Teknik Elektro yang telah banyak membantu dan memberikan semangat dalam menyelesaikan skripsi ini.

Tak ada gading yang tak retak. Oleh karena itu, penyusun mengharapkan kritik dan saran yang konstruktif dan positif demi kesempurnaan skripsi ini. Akhirnya penyusun berharap agar skripsi ini dapat bermanfaat bagi semua.

Malang, September 2007

Penyusun

DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN	i
LEMBAR BERITA ACARA	ii
ABSTRAKSI	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiv
BAB I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan.....	2
1.4. Batasan Masalah.....	2
1.5. Metodologi	3
1.6. Sistematika	3
BAB II. LANDASAN TEORI	5
2.1. Mikrokontroler Renesas R8C/13 Tiny (R5F21134FP).....	5
2.1.1. Spesifikasi R5F21134FP.....	6
2.1.2. Kelebihan Kunci R8C/Tiny.....	7
2.1.3. Konfigurasi Pin R8C R5F21134FP.....	8
2.1.4. Peripheral R8C R5F21134FP.....	11
2.2. Mikrokontroler AT89C51	14

2.2.1. Arsitektur AT89C51.....	15
2.2.2. Pin Deskripsi	17
2.2.3. Organisasi Memori.....	22
2.2.4. Program Status Word (PSW)	24
2.2.5. Stack Pointer	25
2.2.6. Serial Data Buffer.....	25
2.2.7. Register Kontrol	26
2.2.8. Sistem Interrupt.....	27
2.2.9. Timer/Counter	29
2.2.10. Mode Pengalamatan	30
2.3. LCD (Liquid Cristal Display)	32
2.3.1. Sinyal Interface M1632.....	33
2.3.2. Mengatur Tampilan M1632	34
2.4. Information Storage Device (ISD) 2590	39
2.5. Relay.....	45
2.6. IC ULN 2003 Sebagai Penggerak Relay	48
2.7. Kamera	49
2.7.1. Lensa	49
2.7.2. Bagian Tabung	50
2.8. Televisi	50
2.8.1. Proses Scanning.....	50
2.8.2. Proses Interlaced Scanning.....	52
2.8.3. Proses Sinkronisasi.....	52
2.9. Keypad (4 kolom x 7 baris).....	53

BAB III. PERENCANAAN DAN PEMBUATAN ALAT.....	55
3.1. Perancangan Perangkat Keras	55
3.1.1. Perancangan Minimum Sistem Mikrokontroler Master	
Renesas R8C/13 Tiny R5F21134FP	59
3.1.2. Perancangan Rangkaian LCD	60
3.1.3. Perencanaan dan Pembuatan Rangkaian Keypad.....	63
3.1.4. Perancangan Rangkaian ISD 2590.....	65
3.1.5. Rangkaian Mikrokontroler AT89C51	66
3.1.5.1. Rangkaian Clock	67
3.1.5.2. Rangkaian Reset.....	68
3.1.6. Perancangan Rangkaian Relay Dengan Menggunakan	
IC ULN 2003.....	69
3.2. Perancangan Perangkat Lunak	70
3.2.1. Flowchart.....	72
BAB IV. PENGUJIAN ALAT DAN ANALISA SISTEM.....	74
4.1. Pengujian Keypad 4 x 7	75
4.1.1. Tujuan Pengujian.....	75
4.1.2. Alat dan Bahan	75
4.1.3. Prosedur Pengujian.....	75
4.1.4. Hasil Pengujian	75
4.2. Pengujian LCD.....	78
4.2.1. Tujuan Pengujian.....	78
4.2.2. Alat dan Bahan	78
4.2.3. Prosedur Pengujian.....	78

4.2.4. Hasil Pengujian	79
4.3. Pengujian Rangkaian IC ULN 2003 dan Relay	79
4.3.1. Tujuan Pengujian.....	79
4.3.2. Alat dan Bahan	79
4.3.3. Prosedur Pengujian.....	79
4.3.4. Hasil Pengujian	80
4.4. Pengujian Rangkaian ISD 2590	83
4.4.1. Tujuan Pengujian.....	83
4.4.2. Alat dan Bahan	83
4.4.3. Prosedur Pengujian.....	83
4.4.4. Hasil Pengujian	84
4.5. Pengujian Rangkaian Mikrokontroler AT98C51	86
4.5.1. Tujuan Pengujian.....	86
4.5.2. Alat dan Bahan	86
4.5.3. Prosedur Pengujian.....	86
4.5.4. Hasil Pengujian	87
4.6. Pengujian Rangkaian Mikrokontroler Renesas	88
4.6.1. Tujuan Pengujian.....	88
4.6.2. Alat dan Bahan	88
4.6.3. Prosedur Pengujian.....	88
4.7. Pengujian Rangkaian Secara Keseluruhan	89
4.7.1. Tujuan Pengujian.....	89
4.7.2. Alat dan Bahan	89
4.7.3. Prosedur Pengujian.....	89

4.7.4. Hasil Pengujian	90
4.8. Spesifikasi Alat	92
BAB V. PENUTUP.....	93
5.1. Kesimpulan.....	93
5.2. Saran – Saran.....	93
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
Gambar 2-1. Blok Diagram R8C/11, 13 dan Peta Peripheralsnya.....	6
Gambar 2-2. Konfigurasi Pin R8C R5F21134FP.....	8
Gambar 2-3. Diagram Blok ADC	11
Gambar 2-4. Blok Diagram AT89C51.....	15
Gambar 2-5. Pin – Pin AT89C51	16
Gambar 2-6. Rangkaian Power On Reset.....	20
Gambar 2-7. Rangkaian Cristal.....	22
Gambar 2-8. Organisasi Memori Keluarga MCS51	23
Gambar 2-9. Memori Program Terendah Untuk Pelayanan Interupsi	23
Gambar 2-10. Skema Mendefinisikan PSW	24
Gambar 2-11. Kegunaan Interrupt Enable Register	28
Gambar 2-12. Konfigurasi TMOD.....	29
Gambar 2-13. Pin Out LCD M1632 Standard.....	32
Gambar 2-14. Mengirim/Mengambil Data Ke/Dari M1632	33
Gambar 2-15. Pin-Pin IC ISD 2590	40
Gambar 2-16. Cara Kerja Relay	45
Gambar 2-17. Relay SPST	46
Gambar 2-18. Relay SPDT.....	46
Gambar 2-19. Relay DPDT.....	46
Gambar 2-20. Konfigurasi Pin ULN 2003	48
Gambar 2-21. Konstruksi Dasar Kamera	49

Gambar 2-22. Garis Penyapuan (Scanning).....	51
Gambar 2-23. Pengulangan Berurutan	51
Gambar 2-24. Posisi Tombol Pada Keypad 4x7	53
Gambar 3-1. Blok Diagram Bel-Kamera Pada Rumah Kos.....	56
Gambar 3-2. Rangkaian Minimum System R8C/13 Tiny R5F21134FP	60
Gambar 3-3. Perancangan Rangkaian Liquid Crystal Display (LCD).....	62
Gambar 3-4. Keypad Matrik 4x7	63
Gambar 3-5. Rangkaian ISD 2590	66
Gambar 3-6. Rangkaian Mikrokontroler AT89C51	67
Gambar 3-7. Rangkaian Clock	67
Gambar 3-8. Rangkaian Reset.....	69
Gambar 3-9. Rangkaian Relay Dengan Menggunakan IC ULN 2003.....	69
Gambar 3-10. Flowchart Bel-Kamera Pada Rumah Kos	72
Gambar 3-11. Flowchart Proses Perckaman Suara Penghuni Kos.....	73
Gambar 4-1. Blok Diagram Pengujian Rangkaian Keypad 4x7	76
Gambar 4-2. Foto Rangkaian Keypad 4x7.....	76
Gambar 4-3. Blok Diagram Pengujian Rangkaian LCD.....	78
Gambar 4-4. Foto Rangkaian LCD	79
Gambar 4-5. Foto Hasil Pengukuran Relay	80
Gambar 4-6. Pengujian Rangkaian Relay	81
Gambar 4-7. Blok Diagram Pengujian Rangkaian ISD 2590	83
Gambar 4-8. Foto Rangkaian ISD 2590.....	84
Gambar 4-9. Blok Diagram Pengujian Mikrokontroler AT89C51	86
Gambar 4-10. Foto Pengujian Rangkaian Mikrokontroler AT89C51.....	87

Gambar 4-11. Blok Diagram Pengujian Mikrokontroler Renesas	88
Gambar 4-12. Foto Rangkaian Renesas	89
Gambar 4-13. Foto Rangkaian Keseluruhan	90
Gambar 4-14. Foto Tampilan Kamera Pada TV	91
Gambar 4-15. Foto Miniatur Posisi Speaker Dan Tombol Di Tiap Kamar ..	92

DAFTAR TABEL

	Halaman
Tabel 2-1. Konfigurasi Pin-Pin Dari R8C R5F21134FP.....	9
Tabel 2-2. Mode-Mode Timer.....	12
Tabel 2-3. Fungsi-Fungsi Alternative Port 1.....	18
Tabel 2-4. Fungsi-Fungsi Alternative Port 3.....	19
Tabel 2-5. Fungsi Dan Arti Dari Bit-Bit PSW.....	25
Tabel 2-6. Nilai Awal SFR Setelah Reset.....	26
Tabel 2-7. Mode Operasi Timer/Counter.....	30
Tabel 2-8. Fungsi Pin-Pin LCD.....	36
Tabel 2-9. Kombinasi Masukan Keypad 4x7.....	54
Tabel 3-1. Kombinasi Masukan Keypad 4x7.....	64
Tabel 4-1. Kombinasi Masukan Keypad 4x7.....	77
Tabel 4-2. Perbandingan Hasil Perhitungan Dan Pengukuran I relay.....	81
Tabel 4-3. Hasil Pengujian Untuk Mengetahui Respon Relay.....	82
Tabel 4-4. Hasil Pengujian Rangkaian ISD 2590.....	85
Tabel 4-5. Hasil Pengujian Mikrokontroler AT89C51.....	87
Tabel 4-6. Hasil Pengujian Penekanan Tombol.....	90
Tabel 4-7. Hasil Pengujian Perekaman Suara.....	91
Tabel 4-8. Hasil Pengujian Tampilan Kamera Pada TV.....	91

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi dewasa ini sangatlah pesat. Seiring dengan perkembangan tersebut, manusia dituntut untuk melaksanakan segala sesuatu bukan hanya cepat, tetapi juga harus tepat. Untuk itu manusia senantiasa mencari cara untuk mempermudah pekerjaannya sehari-hari. Berkaitan dengan hal tersebut, manusia juga cenderung untuk mencari alternatif pemecahan masalah bagi pekerjaan yang dilakukan secara manual sehingga pekerjaan tersebut dapat dikerjakan secara otomatis.

Saat ini masih banyak kos-kos yang belum memakai bel pintu atau ada yang sudah memakainya tetapi masih sederhana, sehingga apabila ada yang datang berkunjung ke kos tersebut kadang-kadang sulit untuk bertemu dengan orang yang dicarinya. Hal ini banyak memakan waktu dan tentunya kurang efisien.

Oleh karena hal tersebut diatas, maka penulis ingin membuat suatu alat bel –kamera kos yang menggunakan keypad dan memakai suara ke masing-masing kamar dengan suara yang berbeda-beda, serta ditambah dengan penggunaan kamera, sehingga apabila ada yang datang berkunjung untuk mencari salah seorang yang kos, maka orang tersebut dapat langsung mengetahui kalau sedang ada yang mencari.

1.2. Rumusan Masalah

Berdasarkan pada latar belakang masalah di atas, maka rumusan masalah dapat dijabarkan sebagai berikut :

1. Bagaimana merancang dan membuat alat bel - kamera berbasis Mikrokontroler Renesas R8C/13 Tiny.
2. Bagaimana merancang suatu bel yang efisien dan komunikatif untuk rumah kos.

1.3. Tujuan

Tujuan dari penulisan skripsi ini adalah untuk merancang dan membuat bel – kamera yang dapat mempermudah dalam pemanggilan penghuni rumah kos dengan berbasis Mikrokontroler Renesas R8C/13 Tiny.

1.4. Batasan Masalah

Mengingat begitu luasnya bahan kajian mengenai alat bel kos menggunakan keypad, maka pokok bahasan dibatasi sebagai berikut :

1. Sistem kontrol yang didesain adalah Mikrokontroler Renesas R8C/13 Tiny dengan bahasa pemrogramannya menggunakan bahasa “C”.
2. Alat yang dibuat hanya berupa miniatur untuk 2 kamar, 1 kamera, 3 speaker dan 2 TV.

1.5. Metodologi

Metodologi yang dipakai dalam pembuatan skripsi ini adalah:

1. Studi Literatur

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

2. *Field Research*

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

3. *Design* dan Pembuatan Alat

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

4. Pengujian Alat

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

5. Menyusun Laporan Skripsi.

1.6. Sistematika

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

BAB I PENDAHULUAN

Berisi latar belakang, rumusan masalah, tujuan, pembatasan masalah, metodologi penulisan, serta sistematika penyusunan dan pembuatan alat.

BAB II LANDASAN TEORI

Berisikan tentang teori-teori yang berhubungan dengan komponen-komponen yang digunakan dalam perancangan alat.

BAB III PERENCANAAN DAN PEMBUATAN ALAT

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

BAB IV PENGUJIAN ALAT

Membahas tentang pengujian peralatan secara keseluruhan dan analisa hasil pengujian.

BAB V PENUTUP

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

BAB II

LANDASAN TEORI

2.1. Mikrokontroler Renesas R8C/13 Tiny (R5F21134FP)

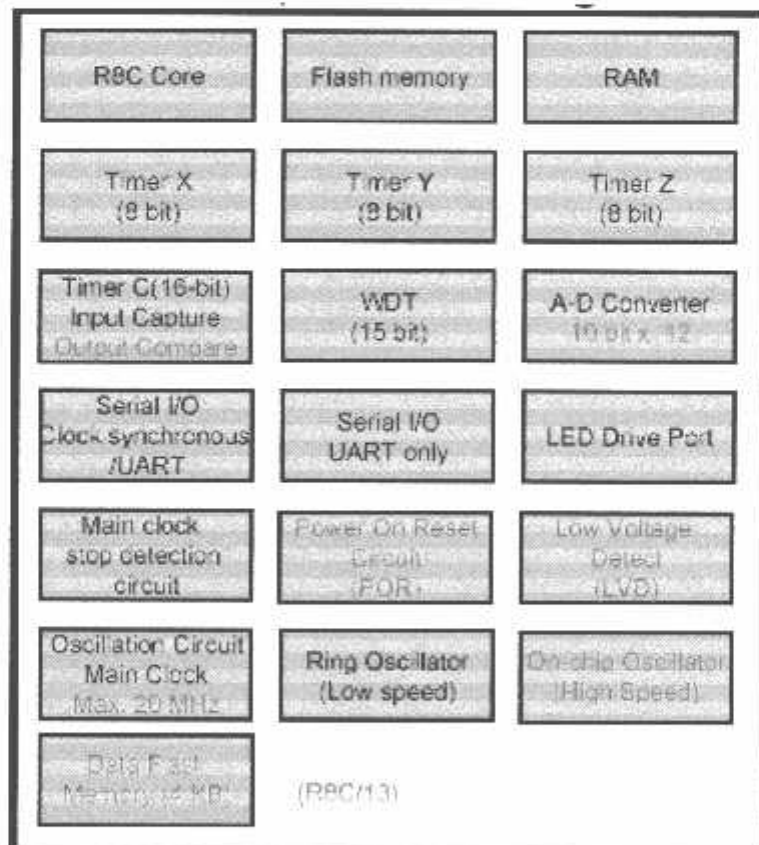
Renesas Technology adalah produsen semikonduktor tingkat internasional. Renesas terbangun dari gabungan dua produsen semikonduktor, yaitu Mitsubishi dan Hitachi. Sebagai produsen semikonduktor, renesas juga mengeluarkan berbagai jenis keluarga mikrokontroler (MK).

Renesas R8C adalah salah satu jenis seri dalam keluarga MK M16C. CPU R8C sama dengan CPU CISC 16-bit M16C, hanya saja lebar jalur data R8C adalah 8-bit. Karena menggunakan CPU yang sama maka R8C memiliki *instruction set* hampir sama dengan M16C. Perbedaannya hanya terletak pada 2 instruksi, yaitu R8C tidak memiliki instruksi JMPS (*Jump Special Page*) dan JSRS (*Jump Subroutine Special Page*). R8C/13 adalah salah satu tipe MK dalam seri R8C. MK ini memiliki kemasan 32-pin LQFP. Dalam perancangan pada skripsi ini menggunakan MK seri R5F21134, yaitu R8C/13 yang memiliki Flash ROM 16 KB (1000 E/W cycles) dan RAM sebesar 1 KB.

2.1.1. Spesifikasi R5F21134FP

Berikut ini adalah spesifikasi *R5F21134FP* dengan peta peripheral dan memori-memorinya.

- ❖ Mempunyai *CPU Core* (16-bit) 1 – 20 MHz, 3.0 – 5.5 Volt dan 1 – 10MHz, 2.7 – 5.5 Volt.
- ❖ Rangkaian Clock, kecepatan *Low/High On-Chip Oscillator*. Clock utama dengan *Xin/Xout*.
- ❖ Memory (ROM/SRAM) 16 Kbytes / 1 Kbytes, 2 x 2 K Bytes Data Flash pada R8C/12, 13.
- ❖ Kemasan 32 pin LQFP (7mm x 7mm)



Gambar 2-1 Blok Diagram R8C/11, 13 dan Peta *Peripheral*-nya

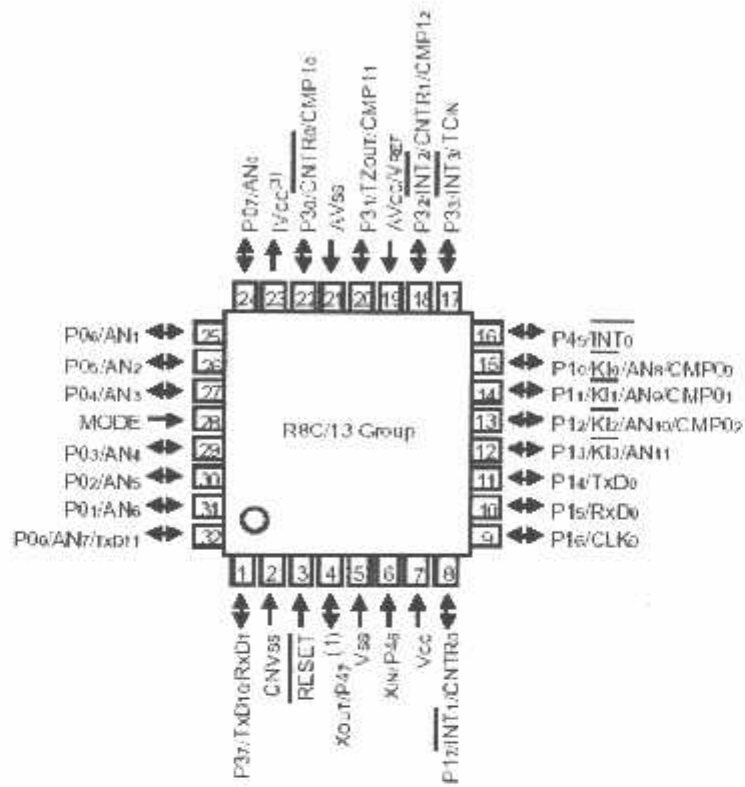
Sumber ; www.renesas.com

2.1.2. Kelebihan Kunci R8C/Tiny

Banyak kelebihan-kelebihan yang dimiliki R8C/Tiny diantaranya adalah :

- ❖ Kompatibel dengan M16C yaitu kompatibel dalam instruksi dan kode.
- ❖ *Peripheral* lebih terintegrasi jadi lebih hemat.
- ❖ *Electromagnetic Compatibility* (EMC) mempunyai EMI rendah, EMS tinggi.
- ❖ *Development Tool* (*Compiler* dan *Debugger*) didapat dengan murah dan difasilitasi *On-Chip Debugger*
- ❖ Mempunyai fitur *fail-safe* yaitu pengamanan terhadap kegagalan sistem.
- ❖ Konsumsi daya rendah.
- ❖ 16-bit CISC CPU dengan kecepatan maksimal 20 MHz (1:1).
- ❖ 89 instruksi CISC lebih hemat ROM kira-kira 20 %, RAM sampai 1 KB.
- ❖ Waktu konversi ADC hanya 3 μ S.

2.1.3. Konfigurasi Pin R8C R5F21134FP



Gambar 2-2. Konfigurasi Pin R8C R5F21134FP

Sumber : www.pmc5as.com

Gambar diatas adalah konfigurasi pin-pin dari *R8C R5F21134FP*, untuk lebih jelasnya dapat diamati pada tabel deskripsi pin-pin berikut ini :

Tabel 2-1 Kofigurasi pin-pin dari *R8C R5F21134FP*

Nama Sinyal	Nama Pin	Type I/O	Fungsi
Masukan Catu Daya	Vcc, Vss	I	Tegangan 2.7 V – 5.5 V pada pin Vcc. Tegangan 0 V pada Vss pin
I Vcc	Ivcc	O	Pin ini untuk men-stabilkan catu daya <i>internal</i> , pin ini dihubungkan pada Vss melalui kapasitor 100nF. Jangan dihubungkan pada Vcc.
Input Catu Daya Analog	Avcc, Avss	I	Ini adalah untuk catu daya pada ADC. Avcc dihubungkan pada Vcc, A Vss dihubungkan ke Vss. Dianjurkan untuk menghubungkan kapasitor diantara pin A Vcc dan A Vss.
Input Reset	RESET	I	“L” untuk masukan ini mereset MCU
CNVss	CNVss	I	Pin ini dihubungkan pada Vss melalui sebuah resistor.
MODE	MODE	I	Pin ini dihubungkan pada Vcc melalui sebuah resistor.
Input Clock Utama	Xin	I	Pin-pin ini disediakan untuk membangkitkan rangkaian I/O Clock Utama. Dihubungkan dengan sebuah keramik resonator atau kristal diantara pin Xin dan Xout. Jika digunakan clock internal maka pin Xin dan Xout dalam keadaan terbuka.
Output Clock Utama	Xout	O	
Input Interupsi	INT0 –INT3	I	Pin ini sebagai masukan interupsi.
Input Kunci Interupsi	KI0 – KI3	I	Pin ini sebagai masukan kunci interupsi.
Timer X	CNTR 0	I/O	Pin I/O ini adalah untuk Timer X .
	CNTR 0	O	Pin Ouput untuk Timer X.

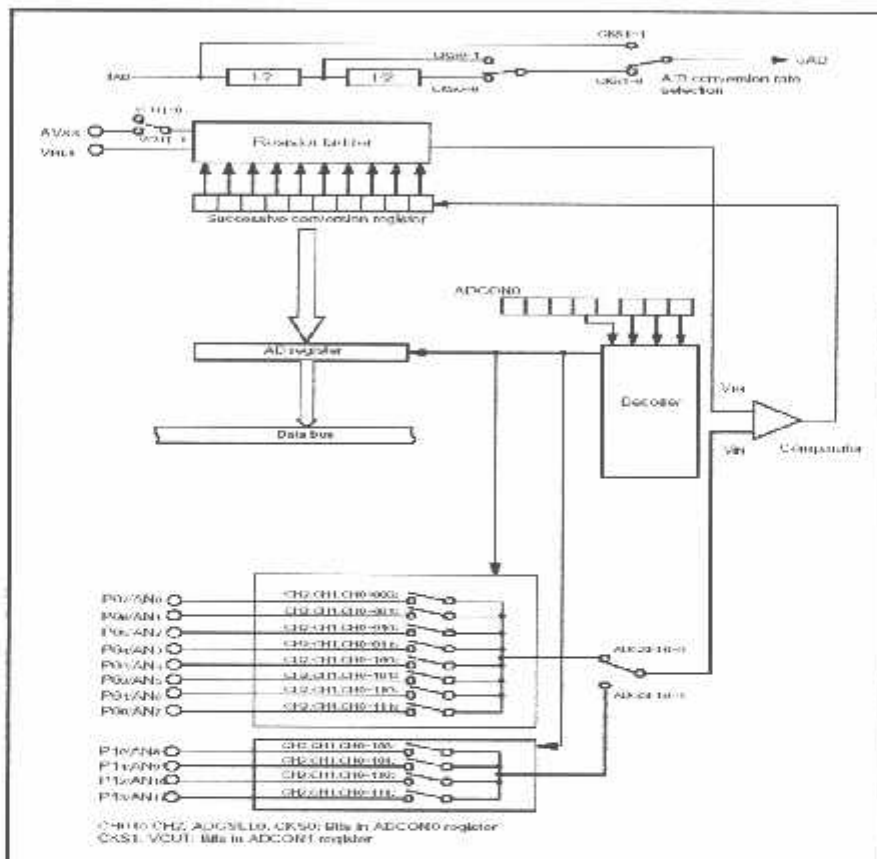
Timer Y	CNTR 1	I/O	Pin I/O untuk Timer Y.
Timer Z	TZout	O	Pin Output untuk Timer Z.
Timer C	TC in	I	Pin Input untuk Timer C.
	CMP00 – CMP03, CMP10 CMP13	O	Pin Output untuk Timer C.
Serial Interface	CLK 0	I/O	Pin I/O untuk memindahkan Clock.
	RXD0, RXD1	I	Pin input untuk data Serial.
	TXD0, TXD10, TXD11	O	Pin output untuk data Serial.
Input Tegangan Referensi	Vref	I	Tegangan referensi input ini untuk ADC. Vref pin dihubungkan ke Vcc.
ADC, pengubah dari analog ke digital	AN0–AN11	I	Pin analog input pada ADC.
Port I/O	P00-P07, P10-P17, P30-P33, P37, P45	I/O	Merupakan port I/O CMOS 8-bit . Setiap port mempunyai pilihan register pengarah sebagai input atau output. Tiap Port dapat dialamat per bit. Dapat di-set menggunakan pull up resistor dengan program. P10 – P17 mempunyai driver transistor.
Port Input	P46, P47	I	Pin ini hanya bisa digunakan sebagai input.

Sumber : www.pines25.com

2.1.4. Peripheral R8C R5F21134FP

Mikrokontroler *R8C R5F21134FP* mempunyai beberapa *peripheral-peripheral* yang banyak digunakan pada beberapa aplikasi-aplikasi penting, diantaranya adalah sebagai berikut :

❖ *Analog To Digital Converter (ADC)*



Gambar 2-3 Diagram blok ADC

Sumber : www.renesas.com

Dengan 12 SAR ADC S/H yang mempunyai resolusi 8-bit atau 10-bit. Mode Operasinya menggunakan *One-Shot dan Repeat* dengan waktu konversi 2.8 uS (pada clock 10 MHz).

❖ **Timer Mode**

Mempunyai timer sebanyak 4 yaitu timer X, Y, Z, C. Berikut adalah mode-mode timernya :

Tabel 2-2 Mode-mode Timer

Item		Timer X	Timer Y	Timer Z	Timer C
Configuration		8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	16-bit timer
Count		Down	Down	Down	Up
Count source		f1 f2 f8 f32	f1 f8 f8MG input from CNTR1 pin	f1 f2 f8 Timer Y underflow	f1 f8 f32
Function	Timer mode	provided	provided	provided	not provided
	Pulse output mode	provided	not provided	not provided	not provided
	Event counter mode	provided	provided ¹	not provided	not provided
	Pulse width measurement mode	provided	not provided	not provided	not provided
	Pulse period measurement mode	provided	not provided	not provided	not provided
	Programmable waveform generation mode	not provided	provided	provided	not provided
	Programmable one-shot generation mode	not provided	not provided	provided	not provided
	Programmable wait one-shot generation mode	not provided	not provided	provided	not provided
	Capture	not provided	not provided	not provided	provided
Input pin		CNTR0	CNTR1	INTC	TCIN
Output pin		CNTR0 CNTR0	CNTR1	TZOUT	not provided
Related interrupt		Timer X int INT1 int	Timer Y int INT2 int	Timer Z int INT0 int	Timer C int INT3 int
Timer stop		provided	provided	provided	provided

Sumber : www.tinexas.com

❖ **Low Voltage Detect (LVD)**

LVD adalah untuk mendeteksi Vcc kurang dari 3.8 V (± 0.5 V)

❖ **Watchdog Timer**

Watchdog berfungsi untuk mendeteksi ketika program diluar kontrol.

❖ *On Chip Debugger*

Fasilitas ini mempunyai fungsi untuk dapat di-*debug* pada waktu mikro sedang berjalan. Antara PC dan MK dapat berkomunikasi, PC akan mengetahui aktivitas MK saat itu. Syarat-syarat *On Chip Debugger* adalah:

- ❑ Vektor *Address Match interrupt* harus dihindari.
- ❑ *Single step interrupt* tidak dapat digunakan bersamaan interrupt lain.
- ❑ *UART1* tidak boleh dipakai.
- ❑ Instruksi BRK tidak boleh dipakai.
- ❑ Flash Address C000H – C7FFH.
- ❑ PD 3.7 harus “0”.
- ❑ B5 FMR 0 harus “1”
- ❑ Menyiapkan 8 Byte untuk Stack.
- ❑ *On Chip Debugger* berpengaruh pada *timing run*.

❖ **Rangkaian Osilator**

Pada osilator utama menggunakan kristal luar sampai dengan 20 MHz, dengan memiliki fitur *Clock Stop Detect*. Kemudian untuk *On Chip Osilator* disediakan kecepatan *Low* 125 KHz dan *High* 8 MHz. Saat setelah reset, default clock adalah kecepatan rendah *On Chip osilator* 125 KHz.

2.2. Mikrokontroler AT89C51

Mikrokontroler bisa dipandang sebagai sebuah mini komputer yang terintegrasi dalam sebuah chip. Didalam satu chip mikrokontroler sudah terdapat bagian-bagian seperti dalam sebuah komputer. Bagian-bagian itu antara lain : ALU (*Arithmetic Logic Unit*), PC (*Program Counter*), SP (*Stack Pointer*), Register, ROM (*Read Only Memory*), RAM (*Random Acces Memory*), Paralel I/O, Serial I/O, *Counter* dan sebuah rangkaian *Clock*.

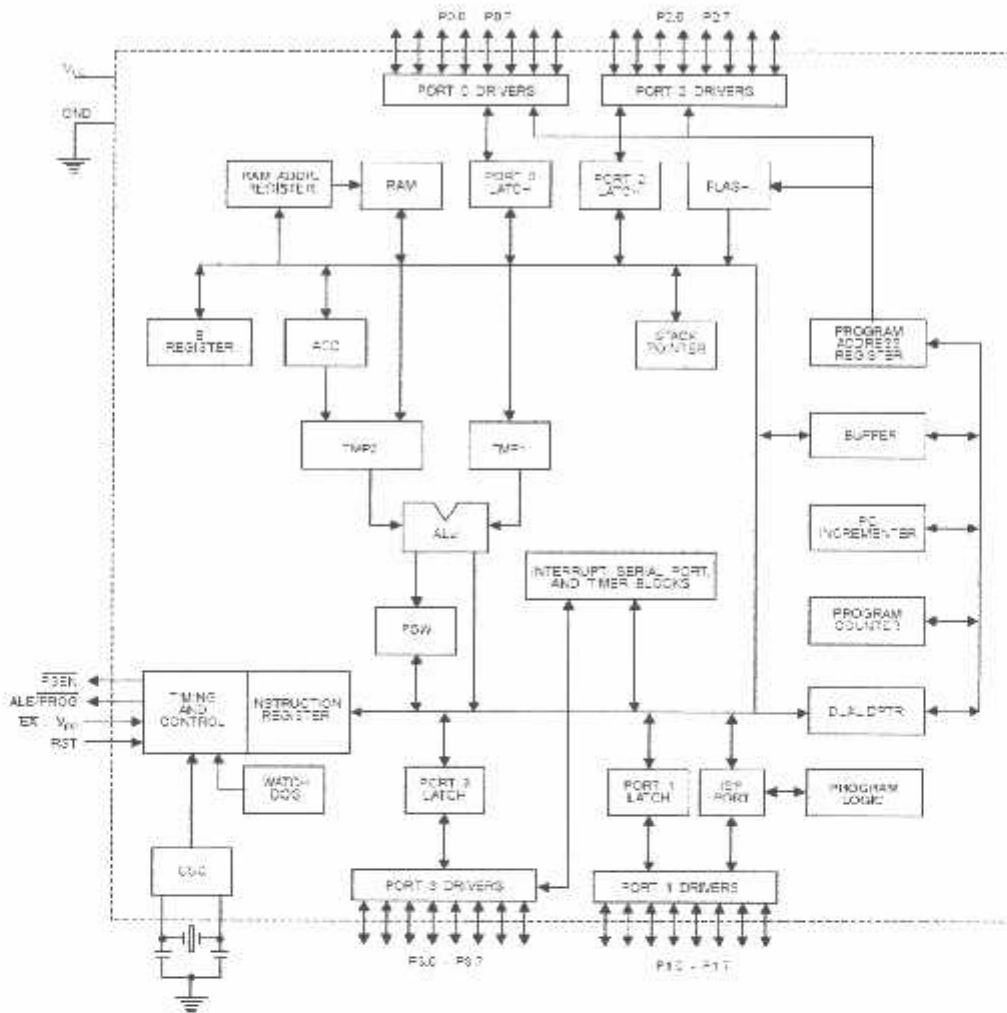
Seperti sebuah mikroprocessor, mikrokontroler adalah sebuah perangkat serbaguna, yang fungsi kerjanya dapat ditentukan melalui sebuah perangkat lunak yang mendeskripsikan sebuah sistem yang diinginkan.

Pada saat ini terdapat banyak keluarga mikrokontroler salah satunya adalah keluarga MCS51. Salah satu tipe mikrokontroler yang termasuk dalam keluarga MCS51 adalah AT89C51 buatan Atmel.

AT89C51 adalah mikrokontroler keluaran atmel dengan 4K byte Flash PEROM (*Programmable and Erasable Read Only Memory*), AT89C51 merupakan memori dengan teknologi nonvolatile memori, artinya isi memori tersebut dapat diisi ulang ataupun dihapus berulang kali.

Memori ini biasa digunakan untuk menyimpan instruksi (Perintah) berstandar MCS – 51 code sehingga memungkinkan mikrokontroler ini untuk bekerja dalam mode *Single Chip Operation* (Mode Operasi Keping Tunggal) yang tidak memerlukan *Eksternal Memori* (Memori luar) untuk menyimpan source code tersebut.

2.2.1. Arsitektur AT89C51



Gambar 2-4 Blok Diagram AT89C51

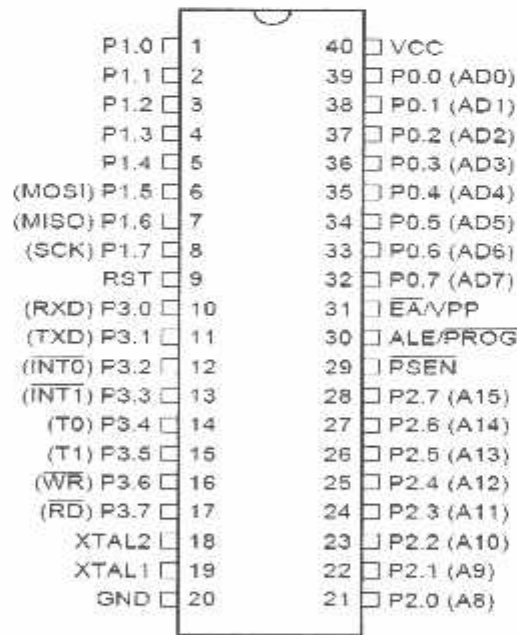
Sumber : www.atmel.com

IC ATMEL AT89C51 menyediakan standart berikut:

- ⇒ 4K Bytes memori yang dapat diprogram ulang
- ⇒ 128 Bytes internal RAM
- ⇒ 32 jalur I/O (Input dan Output) yang dapat diprogram
- ⇒ Sepasang 16 bit Timer dan Counter
- ⇒ Dual data Pointer (DPTR)

- ⇒ Watchdog Timer
- ⇒ ISP Port
- ⇒ Mendukung serial Port secara penuh
- ⇒ Waktu Pemrograman yang singkat

Sebagai tambahan AT89C51 dirancang menggunakan logika yang statis untuk mode pengoperasian yang menuju ke frekwensi dasar dan pendukungan terhadap dua Software, serta dapat memilih model Power Savingnya. Mode idle akan berhenti ketika CPU sedang menjalankan RAM, Timer/Counter, Serial Port dan Interrupt System untuk terus melanjutkan fungsinya. Model power down akan menyimpan isi dari RAM tapi akan memberhentikan ossilator dan akan menghentikan semua chip lain yang sedang berfungsi sampai terdapat adanya gangguan dari luar atau hardware di reset.



Gambar 2-5 Pin - Pin AT89C51

Sumber : www.atmel.com

2.2.2. Pin Deskripsi

VCC : Power Supply

GND : Ground

Port 0 : Port 0 berfungsi sebagai 8 bit I/O bertipe *open drain bi-directional*. Sebagai port keluaran masing – masing pin dapat menyerap arus sebesar 8 masukan TTL (sekitar 3,8 mA). Ketika diberikan logika ‘1’ pada pin port 0 ini maka pin – pin port 0 ini akan dapat digunakan sebagai inputan berimpedansi tinggi.

Port 0 juga dapat dikonfigurasi pada sebagai bus alamat/data selama proses pengaksesan data memori dan program eksternal.

Jika digunakan dalam mode ini port 0 memiliki internal Pull Up.

Port 0 juga menerima kode – kode data yang diberikan padanya selama proses pemrograman dan memberikan kode – kode selama proses verifikasi program yang telah tersimpan didalam memori. Dalam hal ini dibutuhkan eksternal Pull Up selama proses verifikasi program.

Port 1 : Port 1 berfungsi sebagai 8 bit I/O Bi-directional yang dilengkapi dengan internal Pull Up. Ketika diberikan logika ‘1’ pin ini akan di Pull Up secara internal sehingga dapat digunakan sebagai input. Sebagai inputan jika pin – pin ini dihubungkan ke ground maka masing – masing pin ini dapat menghantarkan arus karena di Pull High secara internal. Port 1 juga menerima *Low Order Address Bytes* selama melakukan verifikasi program.

Pada port 1 di AT89C51 pin ini mempunyai alternatif seperti pada tabel berikut ini:

Tabel 2-3 Fungsi - Fungsi Alternative Port 1

Port Pin	Alternate Funtions
P1.5	MOSI (Master Output Slave Input)*
P1.6	MISO (Master Input Slave Output)*
P1.7	SCK (Serial Clock)*

Sumber : www.atmel.com

Port 2 : Port 2 berfungsi sebagai 8 bit I/O Bi-directional yang dilengkapi dengan internal Pull Up Penyangga keluaran port 2 dapat memberikan atau menyerap arus empat masukan TTL (sekitar 1,6 mA)

Jika diberikan logika '1' pada pin – pin port 2, maka masing – masing pin akan di Pull High secara internal sehingga dapat digunakan sebagai inputan. Sebagai inputan jika pin – pin port 2 dihubungkan ke ground (di Pull Low), maka masing – masing pin dapat menghantarkan arus karena di Pull High secara internal.

Port 2 akan memberikan byte alamat bagian tinggi (High Byte) selama pengambilan instruksi dari memori program eksternal dan selama pengaksesan memori data eksternal yang menggunakan perintah dengan alamat 16 bit (misalkan **MOVX@DPTR**). Dalam aplikasi ini , jika ingin mengirimkan '1', maka digunakan Pull Up internal yang sudah disediakan. Selama pengaksesan memori data eksternal yang menggunakan perintah 8 bit (misalkan

MOVX@RI), port 2 akan mengirimkan isi dari SFR P2 (*Special Function Register Port 2*). Port 2 juga menerima alamat bagian tinggi (High Order Address) selama pemrograman dan verifikasi memori.

Port 3 : Port 3 sebagai 8 bit I/O Bi-directional yang dilengkapi dengan Pull Up Internal. Penyangga keluaran port 3 dapat memberikan atau menyerap arus empat masukan TTL (sekitar 1,6 mA).

Jika diberikan logika '1' pada pin-pin port 3, maka masing – masing pin akan di Pull High oleh Pull Up internal sehingga dapat digunakan sebagai inputan. Sebagai inputan, jika pin – pin port 3 dihubungkan ke ground, maka masing – masing kaki akan memberikan arus karena di Pull High secara internal.

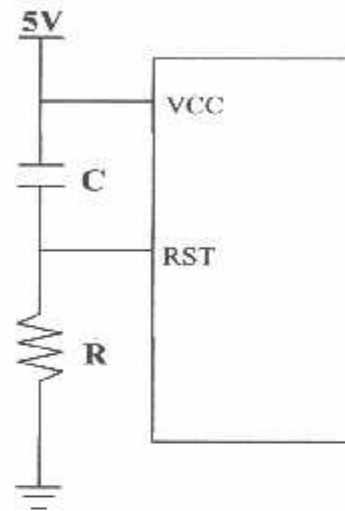
Seperti Port 1, port 3 juga mempunyai fungsi – fungsi alternatif yang diberikan oleh AT89C51 seperti pada tabel berikut ini:

Tabel 2-4 Fungsi - Fungsi Alternatif Port 3

Port Pin	Fungsi Alternatif
P3.0	RXD (Serial Input Port)
P3.1	TXD (Serial Output Port)
P3.2	INT0 (Eksternal Interrupt 0)
P3.3	INT1 (Eksternal Interrupt 1)
P3.4	T0 (Timer 0 Eksternal Input)
P3.5	T1 (Timer 1 Eksternal Input)
P3.6	WR (Eksternal Data Memory Write Strobe)
P3.7	RD (Eksternal Data Memory Read Strobe)

Sumber : www.atmel.com

Reset : Inputan Reset akan memberikan logika High '1' pada pin ini dengan jangka waktu yang ditentukan oleh lamanya pengosongan data muatan kapasitor. Jangka waktu minimal adalah 2 siklus mesin (24 periode frekwensi clock) ditambah waktu start on Osilator.



Gambar 2-6 Rangkaian Power On Reset

Sumber : Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51

ALE/ $\overline{\text{PROG}}$: Keluaran ALE (*Address Latch Enable*) menghasilkan pulsa – pulsa untuk menutup byte rendah (*Low Byte*) alamat selama mengakses memori eksternal. Pin ini juga berfungsi sebagai inputan pulsa program (*The Program Pulse Input*) atau PROG selama melakukan Flash Program. Pada operasi normal, ALE akan berpulsa dengan pewaktuan (*Timing*) atau pendetakan (*Clocking*) rangkaian eksternal. Sebagai catatan ada sebuah pulsa yang dilewati selama pengaksesan memori data eksternal. Jika dikehendaki operasi ALE dapat di nonaktifkan dengan cara mengatur bit 0 dari SFR (*Special Function Register*) lokasi 8Eh. Jika diberi logika '1' ALE hanya

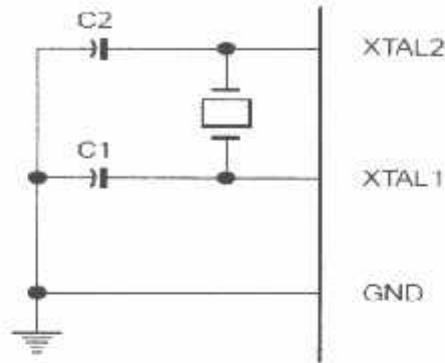
akan aktif selama menemui instruksi **MOVX** atau **MOVC**. Selain itu, pin ini secara perlahan akan di Pull High. Mematikan bit ALE tidak akan ada efeknya jika mikrokontroller mengeksekusi program secara eksternal.

$\overline{\text{PSEN}}$: **$\overline{\text{PSEN}}$** (*Program Store Enable*) merupakan sinyal baca untuk memori program eksternal. Ketika mikrokontroller AT89C51 menjalankan kode dari program eksternal, **PSEN** akan diaktifkan sebanyak 2 kali per siklusnya, kecuali dua aktivasi **PSEN** dilompati (Diabaikan) saat mengakses memori data eksternal.

$\overline{\text{EA/VPP}}$: **$\overline{\text{EA/VPP}}$** (*External Access Enable*). **$\overline{\text{EA}}$** harus selalu dihubungkan ke Ground karena digunakan untuk mengakses eksternal memori dengan lokasi 0000H sampai FFFFH. Catatan sekalipun bit '1' sudah terkunci dan terprogram, maka EA akan terkunci pada reset. **EA** juga harus dihubungkan ke **Vcc** untuk melakukan menjalankan program secara internal. Pada saat Flash Programming pin ini mendapatkan tegangan sebesar 12 Volt.

XTAL1 : Merupakan input ke penguat pembalik osilator dan ke rangkaian operasi Clock internal.

XTAL2 : Keluaran dari penguat pembalik osilator.



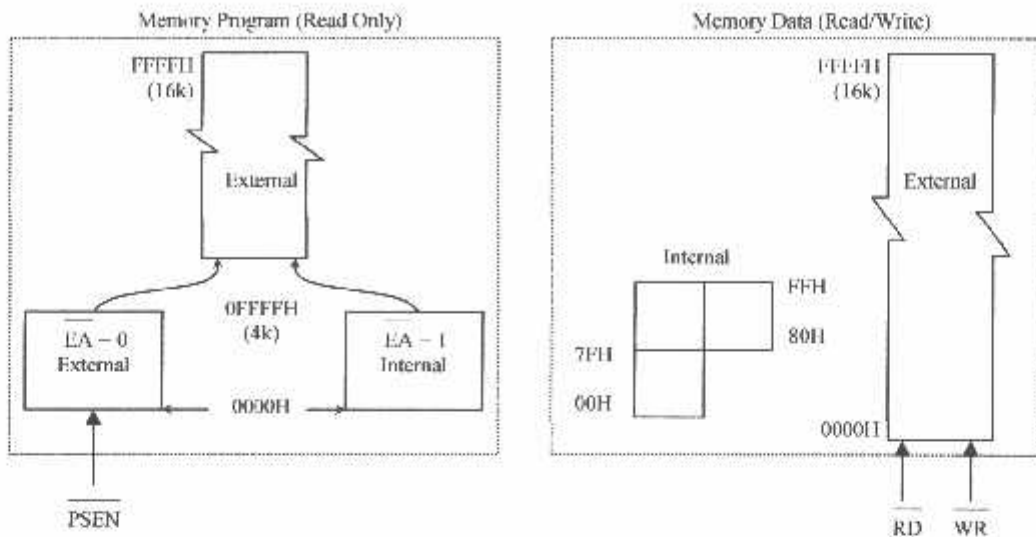
Gambar 2-7 Rangkaian Cristal

Sumber : Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51

Mikrokontroler AT89C51 memiliki rangkaian osilator internal dengan mengacu pada frekwensi referensi pada pin XTAL1 dan XTAL2.

2.2.3. Organisasi Memori

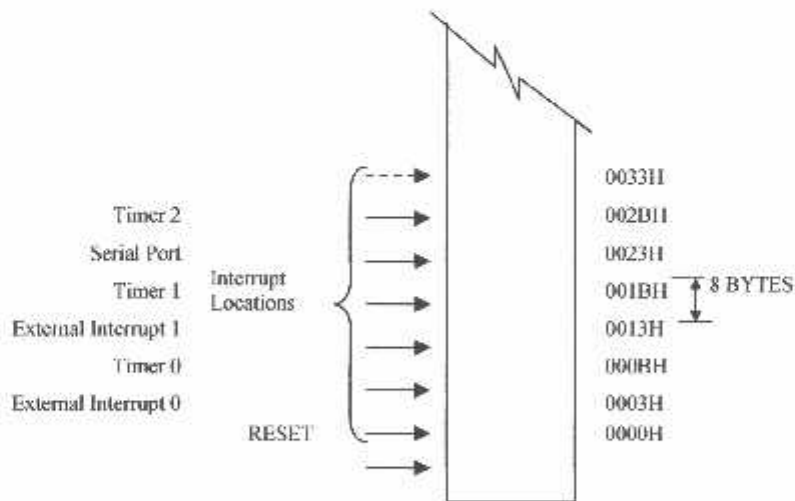
Ruang alamat memori program dan memori data di organisasikan secara terpisah, Memori program hanya dapat di baca dan tidak bisa ditulisi. Kapasitasnya dapat di kembangkan hingga 64k byte. Perhatikan Gambar 2-8, memori 4k internal atau eksternal terendah dapat dipilih dengan memberi logik 1 atau 0 pada pin \overline{EA} . Bila digunakan memori program eksternal, sinyal kontrol untuk membaca adalah \overline{PSE} (*Program Store Enable*).



Gambar 2-8. Organisasi Memori Keluarga MCS51

Sumber : Agfianto Eko Putra, 2002 : 3

Awal eksekusi program (setelah reset) adalah alamat 0000H. Beberapa alamat selanjutnya sudah di tentukan sebagai tempat untuk rutin pelayanan interupsi (ISR : *Interrupt Service Rutin*). Peta alamat terendah dari memori program di tunjukan dalam Gambar 2-9.

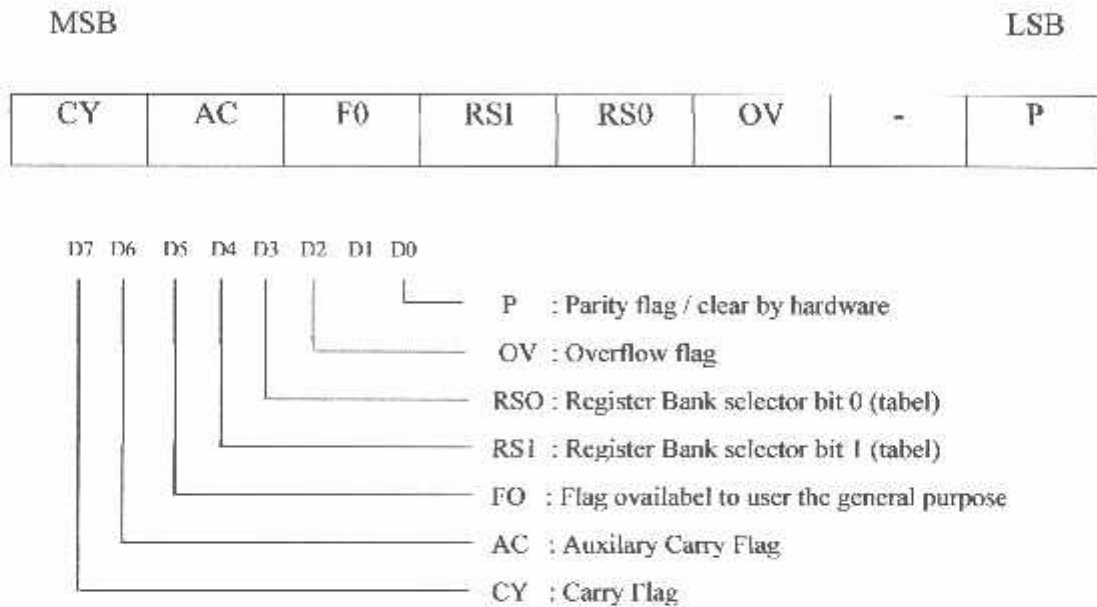


Gambar 2-9. Memori Program Terendah Untuk Pelayanan Interupsi

Sumber : Agfianto Eko Putra, 2002 : 4

2.2.4. Program Status Word (PSW)

Register PSW berisi informasi status dari hasil operasi baru saja dikerjakan CPU. Status hasil operasi yang dikerjakan oleh CPU yang tersimpan selengkapnya ditunjukkan Gambar 2.10.



Gambar 2-10 Skema Mendefinisikan PSW

Sumber : Agfianto Eko Putra, 2002 : 9

Tabel 2-5. Fungsi Dan Arti Dari Bit-Bit PSW

Simbol	Posisi	Arti
CY	PSW.7	Carry Flag
AC	PSW.6	Auxildary Carry Flag (untuk operasi BCD)
F0	PSW.5	Flag 0 (disediakan untuk user)
RS1	PSW.4	Register bank select.
RS0	PSW.3	} Bit 1 & 0 diset/clear oleh software untuk menentukan bank register yang dipekerjakan (lihat catatan)
OV	PSW.2	Over-flow flag
-	PSW.1	Cadangan
P	PSW.0	Parity flag. Diset/clear oleh hardware masing-masing siklus intruksi untuk menandakan jumlah bit 1 ganjil/genap (0/1)

Sumber : Agfianto Eko Putra,2002 : 9

2.2.5. Stack Pointer (SP : Penunjuk Tumpukan)

Nilainya dinaikkan 1 sebelum data disimpan dengan instruksi PUSH dan CALL. Stack pointer di beri nilai awal 07H setelah reset, ini berarti tumpukan di mulai pada lokasi 08H. Lokasi stack dapat dipindah dimanapun dalam RAM internal.

2.2.6. Serial Data Buffer (SBUF)

Serial data buffer register sebenarnya merupakan dua register yang terpisah, buffer tranmisi dan buffer penerima.

Bila data dimuatkan ke SBUF, data akan masuk buffer transmisi di mana data tersebut akan di transmisikan seri. Pemuatan sebuah byte ke SBUF adalah awal dari tranmisi.

Bila data dibaca dari SBUF, berarti berasal dari buffer penerima.

2.2.7. Register Kontrol

IP, IE, TMOD, TCON, SCON, dan PCON berisi bit-bit kontrol dan status untuk sistem interupsi, timer, dan port seri.

Nilai SFR setelah reset merupakan nilai awal yang sudah ditentukan secara *hardware*, sebagai berikut (Tabel 2.6) :

Tabel 2-6. Nilai Awal SFR Setelah Reset.

Register	Nilai (biner)	Register	Nilai (biner)
ACC	00000000	IE : 8051	0xx00000
B	00000000	8052	0x000000
PSW	00000000	TMOD	00000000
SP	00000111	TCON	00000000
DPTR : DPH	00000000	TH0	00000000
DPL	00000000	TLO	00000000
P0	11111111	TH1	00000000
P1	11111111	TL1	00000000

P2	11111111	SCON	00000000
P3	11111111	SBUF	indeterminate
IP : 8051	xxx00000	PCON : NMOS	0xxxxxxx
8052	xx000000	CMOS	0xxx0000

Sumber : Agfianto Eko Putra,2002 : 12

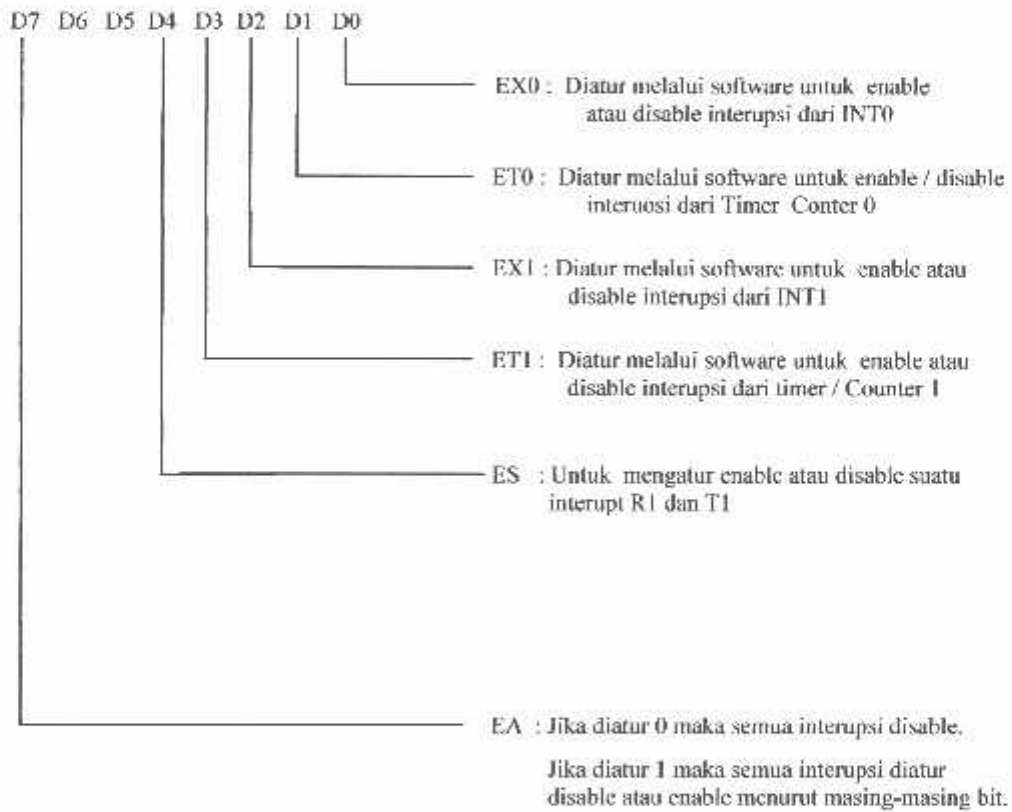
Selain Memori Data internal (256 byte) MCS51 dapat mengakses Memori Data eksternal dengan kapasitas maksimum 64K byte. CPU akan membangkitkan sinyal \overline{RD} dan \overline{WR} pada pin P3.7 dan P3.6 untuk akses ke memori data eksternal.

2.2.8. Sistem Interrupt

Mikrokontroler AT89C51 mempunyai 5 buah sumber interrupt yang dapat membangkitkan permintaan interrupt, yaitu INT0, INT1, T1, T2 dan port serial. Saat terjadinya interrupt Mikrokontroler secara otomatis akan menuju ke subrutin pada alamat tersebut. Setelah interrupt service selesai dikerjakan, Mikrokontroler akan mengerjakan program semula. Dua sumber eksternal adalah INT0 dan INT1, dimana ke dua interupsi eksternal akan aktif level atau aktif transisi tergantung isi IT0 dan IT1 pada register TCON. Interupsi T0 dan T1 aktif pada saat timer yang sesuai mengalami *roll over*. Interupsi serial dibangkitkan dengan melakukan operasi OR pada RI dan TI. Tiap-tiap sumber interupsi dapat *enable* atau *disable* secara *software*.

Tingkat prioritas semua sumber interupsi dapat diprogram sendiri-sendiri dengan *set* atau *clear* bit pada SFRS IP (*Interrupt Priority*). Interupt tingkat rendah dapat di interupsi oleh interupt yang mempunyai tingkat lebih tinggi, tetapi tidak sebaliknya. Walaupun demikian interupsi yang mempunyai tingkat lebih tinggi tidak bisa menginterupsi sumber interupt lain.

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



Gambar 2-11 Kegunaan Interrupt Enable Register

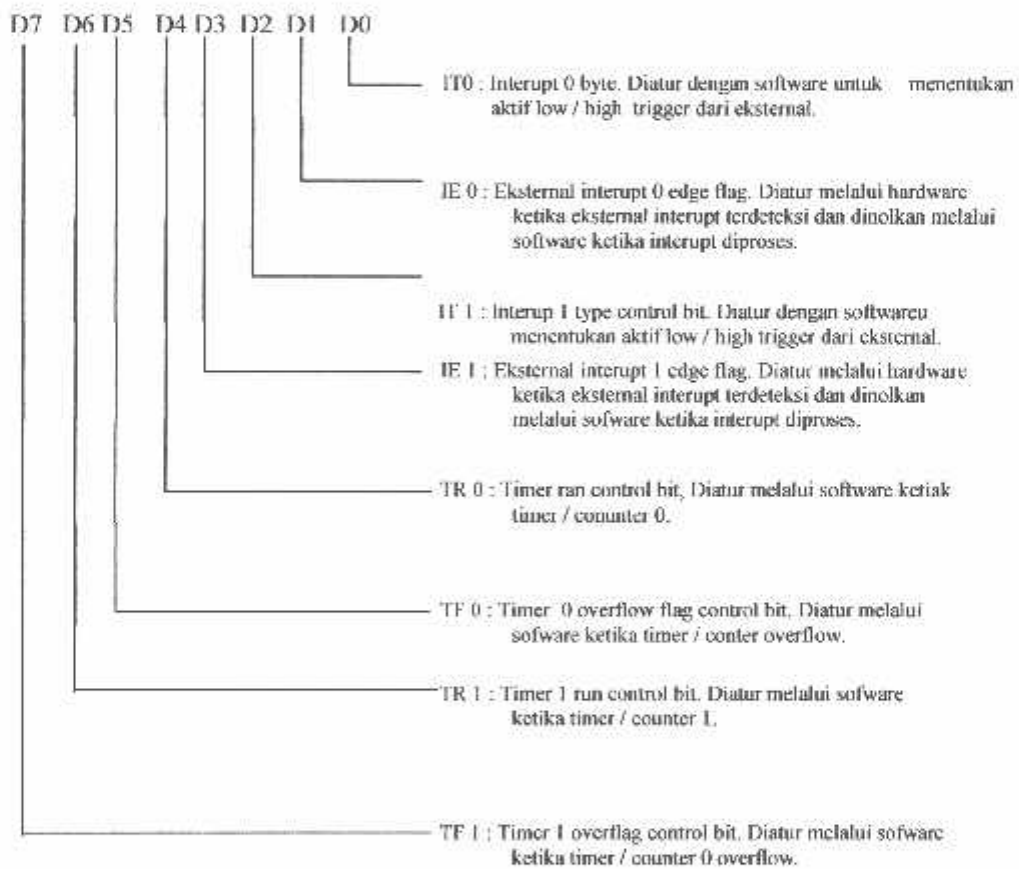
Sumber : Agfianto Eko Putra, 2002 : 25

2.2.9. Timer/Counter

Pengendalian kerja dari *timer / counter* dilakukan dengan pengaturan register yang berhubungan dengan kerja dari *timer / counter*, yaitu melalui sebuah *timer / Counter Mode Control*.

Untuk mengaktifkan *timer / counter* yang meliputi penentuan fungsi sebagai timer atau sebagai counter serta pemilihan mode operasi dapat diatur melalui TMOD yang beralamat pada 90H.

Konfigurasi yang dimaksud sebagai berikut :



Gambar 2-12. Konfigurasi TMOD

Sumber : Agfianto Eko Putra, 2002 : 114

Pengaturan *timer* atau *counter* dilakukan pertama kali dengan mengatur TMOD. Disini kita akan menentukan fungsi sebagai timer atau counter lengkap dengan spesifikasinya. Demikian juga dengan pengaktifan interupsi yang berhubungan dengan penggunaan mode ini.

Tabel 2-7 Mode Operasi Timer / Counter

MI	M0	OPERATING MODE
0	0	Timer 13 bit
0	1	Timer / Counter 16 bit
1	0	8 bit Auto Reload / Counter
1	1	TL 0 dari Timer adalah 8 bit Timer / Counter dikendalikan oleh control bit Timer 0. TH 0 adalah 8 bit yang dikendalikan oleh Timer 1 control.

Sumber : Agfianto Eko Putra,2002 : 115

2.2.10. Mode Pengalamatan

Bagaimana suatu instruksi prosesor mengambil *operand* (informasi yang dioperasikan) pada suatu lokasi alamat mendefinisikan ragam pengalamatan (*addressing mode*). Instruksi-instruksi MCS51 mempunyai ragam pengalamatan sebagai berikut:

1. Direct Addressing

Pada ragam ini operand ditentukan oleh medan alamat 8 bit suatu instruksi. Hanya Memori data internal yang dapat diakses dengan pengalamatan langsung.

2. Indirect Addressing

Instruksi menentukan register yang berisi alamat dari operand. Masing-masing RAM internal dan eksternal dapat dialamati secara tak langsung.

Alamat register untuk pengalamatan 8 bit adalah R0 atau R1 (Ri) dari bank register yang aktif atau *Stack Pointer* (SP). Alamat register untuk pengalamatan 16 bit adalah hanya DPTR (*Data Pointer*).

3. Register Instructions

Bank register, terdiri dari R0 hingga R1, dapat diakses oleh instruksi tertentu yang menyediakan 3 bit didalam kode operasinya untuk mengacu salah satu register. Instruksi yang mengakses register dengan cara ini menjadi efisien, karena membatasi jumlah byte alamat.

4. Register Specific Instructions

Beberapa instruksi dikhususkan untuk register tertentu. Contoh, beberapa instruksi yang selalu mengoperasikan *Accumulator*, atau *Data Pointer*, disini tidak diperlukan byte alamat. *Opcode* itu sendiri yang mengacunya. Penulisan instruksi yang mengacu *Accumulator* dengan A akan dirakit sebagai kode operasi khusus untuk akumulator.

5. Immediate Constants

Nilai tetap dapat disertakan dalam *opcode* dalam Memori Program.

Contoh : MOV A,#100 (muati akumulator dengan bilangan desimal 100)

6. Indexed Addressing

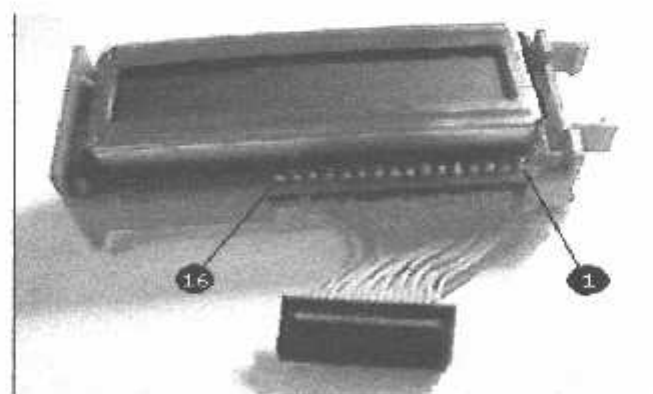
Hanya Memori Program yang dapat diakses dengan pengalamatan berindex dan hanya dapat dibaca.

2.3 LCD (Liquid Cristal Display)

LCD display model M1632 buatan Seiko Inc. terdiri dari bagian, yang pertama panel LCD sebagai media penampil informasi dalam bentuk huruf atau angka, dua baris masing-masing baris dapat menampung 16 huruf atau angka.

Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibalik pada panel LCD, berfungsi mengatur tampilan informasi serta berfungsi mengatur komunikasi dengan mikrokontroler yang memakai tampilan LCD itu. Dengan demikian pemakaian M1632 menjadi sederhana.

LCD diperlukan untuk menampilkan informasi dari berapa lama waktu yang diatur atau diseting pada mikrokontroler melalui keypad, LCD M1632 pada skripsi ini digunakan sebagai tampilan saja sehingga LCD hanya menerima inputan dan kemudian dikirimkan ke mikrokontroler. Hal ini dilakukan dengan memberi tegangan positif pada pin no.5 (R/W) untuk menyalakan back light maka pin no.15 (V + BL) harus diberi tegangan 4.6 Volt dan pin no.16 (V - BL) harus diground.



Gambar 2-13 Pin Out LCD M1632 Standard

Sumber : LCD Modul User Manual, Seiko Instrument Ing, 1987

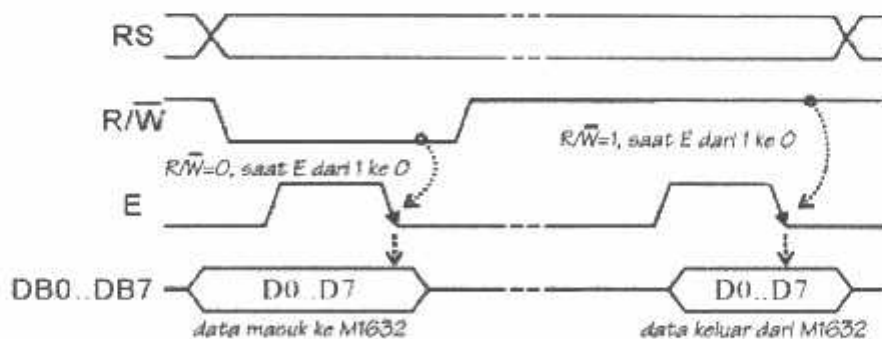
2.3.1. Sinyal interface M1632

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

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

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

Proses mengirim/mengambil data ke/dari M1632 digambarkan dalam gambar 2-14 bisa dijabarkan sebagai berikut :



Gambar 2-14 Mengirim/Mengambil Data Ke/Dari M1632

Sumber : LCD Modul User Manual, Seiko Instrument Ing, 1987

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 saat **E** menjadi 1, M1632 akan meletakkan datanya di **DB0 .. DB7**, data ini harus diambil sebelum sinyal **E** di-nol-kan kembali.

2.3.2 Mengatur tampilan M1632

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

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

1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya transfer 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 Renesas R8C/13Tiny 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 8 bit, yakni hanya memakai jalur data **D0..D7**
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.

LCD modul M1632 mempunyai 16 pin dengan fungsi sebagai berikut :

Tabel 2-8 Fungsi Pin - Pin LCD

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 mengaktifkan data tulis atau baca
7 - 14	DB0 - DB7	Merupakan saluran data berisi perintah data yang akan ditampilkan
15	V + BL	Back Light Supply 5 Volt (Volt)
16	V - BL	Back Ligth Supply 0 (Ground)

Sumber : LCD Modul User Manual, Seiko Instrument Ing, 1987

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.

Pada LCD juga terdapat instruksi – instruksi sebagai berikut :

- ❖ *Display clear* : membersihkan tampilan yang ada pada LCD serta menyimpan, sedangkan kursor kembali ke posisi semula.
- ❖ *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.

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

- ❖ *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 diidentifikasi sebagai layar

S/C : 0 = LCD diidentifikasi sebagai kursor

R/L : 1 = menggeser satu spasi ke kanan

R/L : 0 = menggeser satu spasi ke kiri

❖ *Fuction Set*

DL : 1 = panjang data LCD pada 8 bit

DL : 0 = panjang data LCD pada 4 bit

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

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

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

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

2.4. Information Storage Device (ISD) 2590

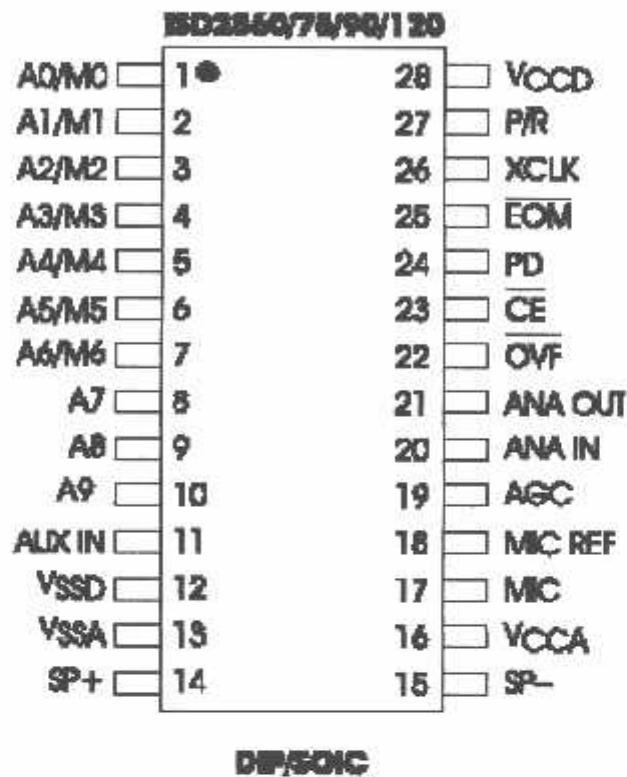
IC penyimpan suara yang digunakan merupakan jenis EEPROM (*Electrically Erasable Programmable Read Only Memory*) yaitu ROM yang dapat diprogram, dihapus dan diprogram ulang secara elektrik dengan arus listrik, bukan sinar *ultraviolet*. IC ISD (*Information Storage Device*), yang dipakai yaitu ISD 2590. IC ini dapat merekam pesan maksimal 90 detik dan dapat dikaskade sehingga pesan yang disimpan dapat diperpanjang sesuai dengan keinginan kita dengan 160 alamat yang berbeda.

Didalam ISD tersebut dilengkapi dengan internal *amplifier*, *internal automatic gain control (AGC)*, *filter antialiasing* (perata) dan *speaker amplifier* (penguat speaker). Secara keseluruhan seri ISD dapat melakukan sebuah perekaman atau pemutaran ulang pesan dengan komponen sederhana seperti mikropon, *speaker*, beberapa komponen penunjang, dua buah saklar dan sumber tegangan.

Rekaman akan disimpan dalam sel memori yang tidak mudah hilang (*non volatile*), memberikan tempat penyimpanan yang masih kosong. Cara unik ini yang membuat ISD disebut *Direct Analog Storage Technology (DAST)* atau teknik penyimpanan analog langsung, dengan jalan sinyal suara (*voice*) dan bunyi disimpan secara langsung dalam bentuk analog, kedalam memori EPROM. Penyimpanan analog langsung memungkinkan reproduksi suara secara alami dalam satu chip tunggal.

Susunan ISD 2590 DAST adalah dikelompokkan dalam 160 segmen dari alamat A0 sampai A7 yang menunjukkan akses tips segmen dalam kesatuan untuk

alamat pesan. Kemampuan pemberian atau penyediaan alamat yang berupa pesan yang disimpan dalam bentuk kalimat dan suara.



Gambar 2-15. Pin-pin IC ISD 2590

Sumber : www.national.com/semiconductor

Berikut adalah konfigurasi pin-pin ISD 2590 :

❖ **Address Input (A0-A7) Pin 1-6 Dan 9-10**

Input alamat ini mempunyai dua fungsi, tergantung dari level dari dua *Most Significant Bits* (MSB) dari alamat. Jika dua MSB ini keduanya low, maka semua input digunakan sebagai bit pengalamatan (*Address Bits*) dan digunakan sebagai alamat untuk memulai (*Start Address*)

dari perekaman atau pemutaran ulang (*Play Back*). Kaki-kaki dari pengalamatan hanya merupakan masukan dan bukan merupakan informasi keluaran pengalamatan internal (*Output Internal Address Information*). Ketika proses operasional sedang berjalan dan pada saat kedua MSB ini high, maka sinyal input pengalamatan digunakan sebagai bits mode (*Mode Bit*) yang membuat mode operasi normal dan pengalamatan secara tidak langsung (*Simultaneously*).

❖ **Vssd dan Vssa (Ground) Pin 12 dan 13**

Sama seperti Vccd dan Vcca, analog input dan digital sirkuit di dalam ISD menggunakan *bus ground* yang terpisah untuk meminimalisasi *noise*. Pin ini harus dihubungkan sedekat mungkin dengan *ground*.

❖ **Speaker Output (SP +, SP -) Pin 14 dan 15**

Pin SP + dan SP – digunakan untuk mengeluarkan suara yang telah direkam ke *speaker* atau ke *device* lainnya. Output ini mempunyai impedansi sebesar 16 Ohm.

❖ **Microphone Input (mic) Pin 17**

Kaki mikropon ini terhubung dengan Vcc melalui beberapa kapasitor yang terhubung secara seri, bersamaan dengan resistor 10 K Ω yang berada didalam chip (*internal*). Harga dari kapasitor dari dalam perancangan ini menggunakan harga kapasitor sesuai dengan yang tertera dalam rangkaian data sheet ISD.

❖ **Microphone Reference (MIC REF) Pin 18**

Ketika MIC REF menghubungkan antara Vcc dengan mikropon *ground*, maka tingkat noise selama perekaman dapat dikurangi. Noise itu disebabkan oleh *pre-amplifier* yang terdapat didalam *chip*. Bila pin ini tidak digunakan, maka tidak boleh dihubungkan dengan sinyal atau dengan tegangan apapun, harus dalam keadaan terbuka.

❖ **Automatic Gain Control (AGC) Pin 19**

Kegunaan dari AGC adalah untuk menambah atau mengurangi secara otomatis penguatan (*Gain*) dari *pre-amplifier* yang juga meluaskan batas dari sinyal input yang dapat digunakan oleh mikropon tanpa terjadi distorsi. AGC ini dapat secara dinamis meluaskan batas dari suara yang terekam baik itu suara bisikan sampai suara yang keras. Untuk menggunakan fasilitas AGC ini, resistor dan kapasitor luar (eksternal) harus dihubungkan secara *parallel* antara pin AGC dengan *ground*. Harga yang direkomendasikan adalah $R = 470\text{ K}\Omega$ dan $C = 4,7\ \mu\text{F}$ (Dalam perancangan ini juga dipakai harga seperti diatas sama dengan data sheet ISD).

❖ **Analog Input (ANA IN) Pin 20**

Kapasitor eksternal (luar) menghubungkan antara ANA IN ke ANA OUT pin harga-harga dari kapasitor luar bersama dengan $3\text{ K}\Omega$ input impedansi di ANA IN dapat dipilih sendiri untuk memberikan keadaan *cut off* (terputus) pada frekuensi rendah sampai pada *pass band* suara. ANA IN juga dapat digunakan pada input sumber *alternative* dari sinyal analog pada sinyal mikropon terus ke kapasitor kopling.

❖ **Analog Output (ANA OUT) Pin 21**

Sinyal dari mikropon dikuatkan dan dikeluarkan melalui ANA OUT pin. Penguatan tegangan dari *pre-amp* tergantung dari tingkat tegangan AGC (*Automatik Gain Control*) pin. *Pre-amplifier* ini mempunyai penguatan maksimum sekitar 24 dB untuk tingkat masukan kecil.

❖ **Playback, Level – Activated (PLAY L) Pin 23**

Ketika sinyal ini berpindah dari *high* ke *low*, maka PLAY L akan berjalan. *Playback* akan berjalan sampai input ini tertekan *high*, tanda akhir dari pesan tercapai atau ruang memori sudah habis. ISD akan kembali ke mode *stanby* setelah *playback* ini berhenti.

❖ **Playback Edge-Aktivated (PLAY E) Pin 24**

Ketika sinyal akan berpindah menuju *low* (*low-going transition*) terdeteksi di input ini, maka PLAY E akan berjalan. *Playback* berjalan sampai tanda akhir dari pesan tercapai (Akhir dari ruang memori tercapai). Setelah menyelesaikan *playback*, ISD secara otomatis akan kembali ke mode *stanby*, menekan PLAY E ke *high* pada waktu *playback* berjalan tidak akan menghentikan *playback*. Jadi *playback* akan berhenti bila mencapai akhir dari pesan atau ruang memori habis.

❖ **Record LED Output (RECLEL) Pin 25**

Selama proses perekaman output RECLEL akan *low*. Maka output ini bisa digunakan untuk menjalankan sebuah led yang berfungsi untuk mengetahui bahwa terjadi proses perekaman. Ketika tanda akhir dari pesan tercapai pada saat *playback*, maka RECLEL akan *low* sebentar.

❖ ***Optimal External Clock (XCLK) Pin 26***

Digunakan untuk penambahan kristal clock bila dibutuhkan pewaktuan yang lebih besar dan presisi. Bila input ini tidak digunakan, harus dihubungkan dengan ground.

❖ ***Record (REC) Pin 27***

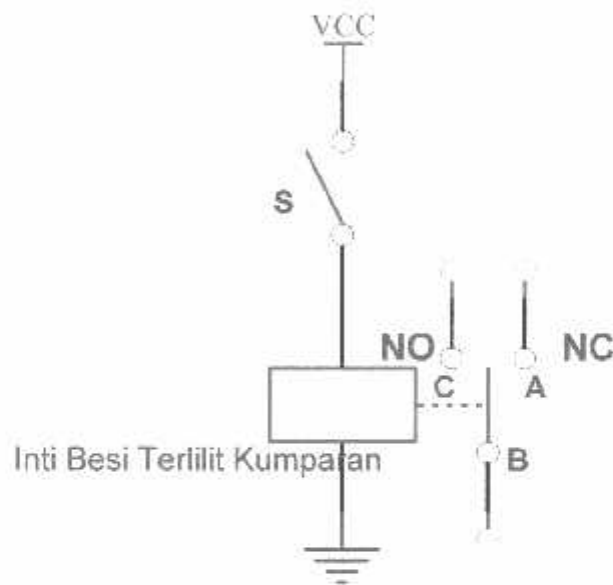
Input sinyal REC akan aktif dalam kondisi *low*. ISD akan merekam bila REC dalam keadaan *low*, dan sinyal ini harus terus dalam keadaan *low* bila ingin terus merekam. Jika input REC ini tertekan *low* dalam keadaan masih memutar ulang pesan (*playback*), maka *playback* akan berhenti dan ISD akan merekam.

❖ ***VCCA Dan VCCD Pin 16 Dan 28***

Analog dan digital sirkuit yang terdapat didalam chip ISD menggunakan bus power yang terpisah untuk meminimalisasi noise. Pin power ini harus dihubungkan sedekat mungkin dengan sumber tegangan.

2.5. Relay

Relay adalah komponen elektronika yang terdiri dari sebuah lilitan kawat (kumparan/koil) yang terlilit pada sebuah besi lunak. Pada Gambar 2-16 jika kumparan dialiri arus listrik maka inti besi akan menjadi magnet dan menarik pegas sehingga kotak AB terhubung dan BC terputus. *Relay* merupakan suatu alat untuk menghubungkan atau memerlukan kontak antara komponen yang satu dengan yang lain.



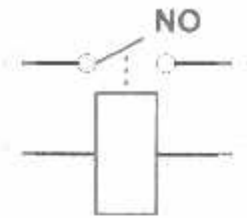
Gambar 2-16. Cara Kerja Relay

Sumber : Elektronika dalam Industri

Dalam memutuskan atau menghubungkan kontak digerakkan oleh fluksi yang ditimbulkan dari adanya medan magnet listrik yang dihasilkan oleh kumparan yang melilit pada besi lunak. Ada beberapa macam *relay*, antara lain:

1. SPST (Single Pin Single Terminal)

Simbol Relay SPST

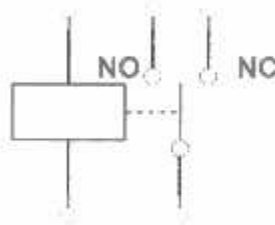


Gambar 2-17. Relay SPST

Sumber : Elektronika dalam Industri

2. SPDT (Single Pin Dual Terminal)

Simbol Relay SPDT

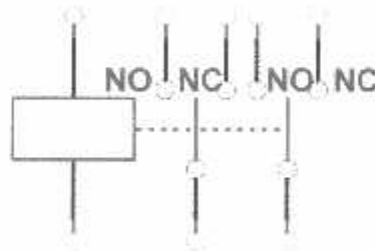


Gambar 2-18. Relay SPDT

Sumber : Elektronika dalam Industri

3. DPDT (Dual Pin Dual Terminal)

Simbol Relay DPDT



Gambar 2-19. Relay DPDT

Sumber : Elektronika dalam Industri

Adapun karakteristik dari sebuah relay adalah sebagai berikut :

1. Tegangan operasi, yaitu tegangan coil minimum yang diperlukan coil untuk dapat mengaktifkan kontak saklar relay dari posisi normal ke posisi operasi.
2. Tegangan lepas, adalah tegangan coil minimum yang diperlukan coil untuk dapat mengaktifkan kontak saklar relay dari posisi operasi ke posisi normal.
3. Tegangan maksimum, yaitu coil maksimum yang diperlukan coil untuk dapat bekerja tanpa merusak coilnya sendiri.
4. Tegangan normal, adalah tegangan kerja nominal sehingga coil dapat bekerja secara normal.

Adapun prinsip kerja dari relay yang kondisi normalnya Off (open) adalah:

- a. Apabila lilitan terisolasi dari relay tersebut kita beri tegangan, maka akan timbul induksi.
- b. Dengan adanya induksi maka besi (inti) yang diselubungi lilitan akan berubah sifatnya menjadi magnet yang bersifat sementara.
- c. Karena besi tersebut menjadi bersifat magnet, maka akan dapat menarik besi lain yang berfungsi sebagai kontak sehingga relay menjadi ON.
- d. Jika tegangan pada lilitan dihilangkan, maka besi akan kembali seperti sifat semula sehingga relay menjadi Off.

2.6. IC ULN 2003 Sebagai Penggerak Relay

IC ULN 2003 adalah IC yang dirancang untuk difungsikan sebagai saklar, IC ULN 2003 dapat terpicu dengan tegangan 5 Volt dengan tegangan hubung sebesar 100 Volt dan arus maksimum yang diperbolehkan sebesar 500mA dengan suhu kerja dari -20°C sampai 80°C dengan data yang ada diatas maka IC ULN 2003 mampu digunakan untuk menghidup-matikan relay yang hanya memiliki tegangan maksimal sebesar 12 volt dengan resistansi kumparan sebesar 400 ohm maka dapat diketahui arus relay sebesar:

$$\text{Dimana : } I_{\text{relay}} = \frac{V_{cc}}{R_{\text{relay}}} \dots\dots\dots (2-1)$$

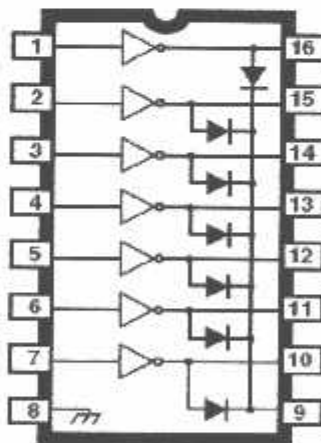
$$I_{\text{relay}} = \frac{12V}{400} = 0,03A$$

$$I_{\text{relay}} = 30mA \dots\dots\dots(2-2)$$

maka :

$$I_{\text{maxULN}} > I_{\text{RELAY}}$$

Karena pada arus maksimal IC ULN 2003 adalah sebesar 500mA maka dapat disimpulkan bahwa ULN 2003 akan dapat difungsikan sebagai driver relay.

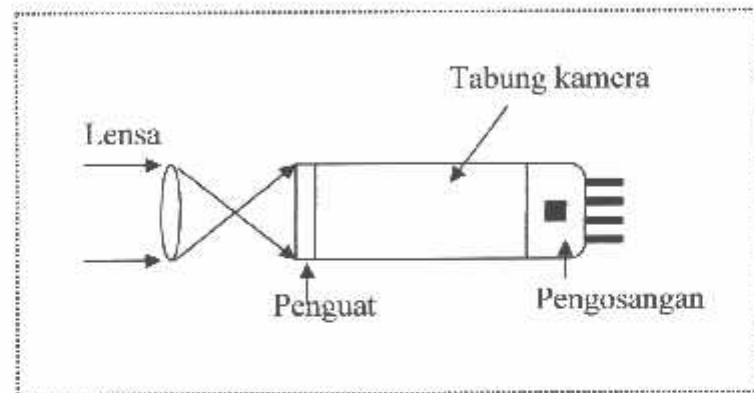


Gambar 2-20. Konfigurasi PIN ULN 2003

Sumber : www.national.com/semiconductor

2.7.Kamera

Kamera merupakan peralatan yang mampu mengubah sebuah gambar menjadi sinyal video. Komponen dasar dari kamera ini terdiri dari dua komponen pokok, yaitu lensa dan tabung kamera. Konstruksi dasar dari kamera ini ditunjukkan dalam Gambar 2.21.



Gambar 2-21 Konstruksi Dasar Kamera

Sumber : Teknik Televisi dan Kamera jilid 1 : 85

2.7.1. Lensa.

Lensa pada kamera berfungsi sebagai pembentuk bayangan optik yang difokuskan pada plat kaca depan, karena memiliki kaca transparan, cahaya akan menabrak plat bayangan foto listrik pada permukaan bagian dalam lensa-lensa optik menghasilkan suatu bayangan yang terbalik pada daerah persegi yang dipancarkan oleh berkas elektron.

Selain membentuk bayangan optik lensa juga mengatur cahaya dengan sebuah selaput pelangi mekanis yang terpasang pada rumah lensa, selaput ini menyesuaikan permukaan, untuk menentukan banyaknya cahaya yang menembus lensa.

2.1.2. Bagian Tabung.

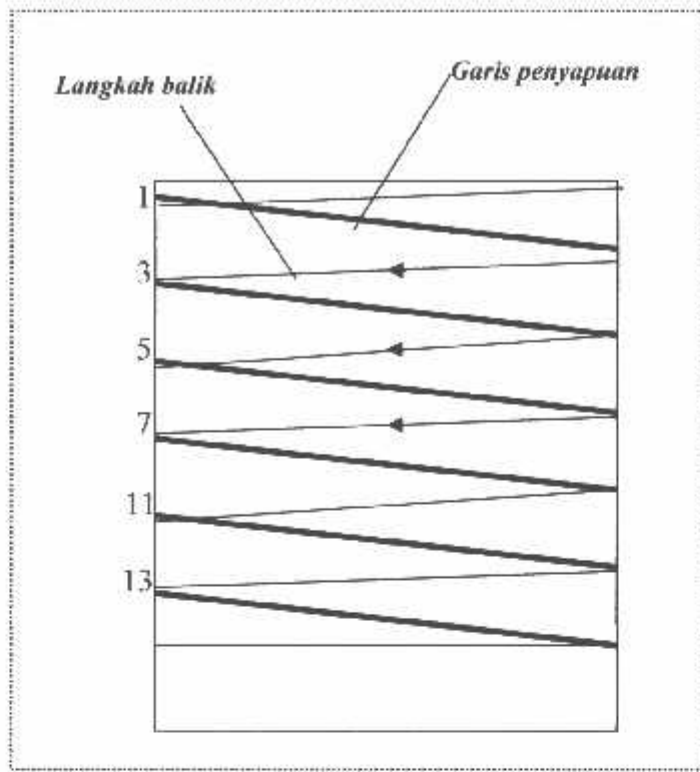
Pada bagian tabung kamera, bayangan cahaya dikonversi menjadi suatu pola muatan listrik, pola muatan ini dipancarkan secara berurutan menurut waktu oleh berkas elektron yang meyapu lewat plat bayangan. Fungsi berkas pemancaran elektron adalah mengosongkan muatan ini menghasilkan arus sinyal elektroda keluaran tabung kamera.

2.8. Televisi

Televisi digunakan untuk menampilkan gambar yang diterima oleh kamera. Pada sistem televisi, gambar yang dipancarkan itu dihasilkan oleh kamera elektron pada lembaran peka cahaya yang selanjutnya di baca cepat oleh berkas elektron. Pancaran hasil pembacaan tersebut umumnya meliputi gerakan, yang dibaca dalam waktu yang cukup singkat sehingga gambaran berurutan yang dihasilkan tidak begitu berbeda. Ada beberapa proses yang terjadi dalam sistem televisi, yaitu:

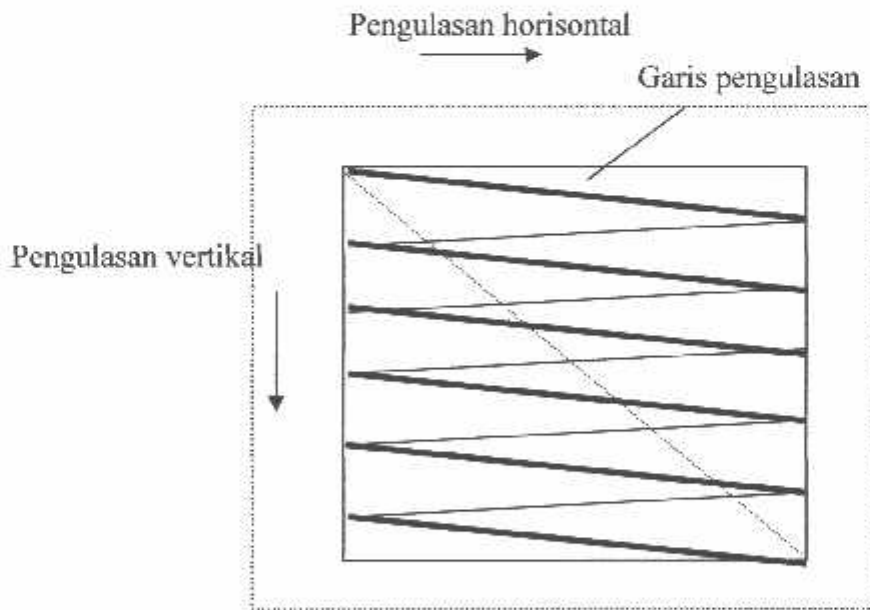
2.8.1. Proses Scanning

Suatu metode yang dilaksanakan dengan mengubah gambar-gambar yang berdimensi dua ke kebesaran-besaran listrik yang berubah dengan waktu, maka proses itu disebut *scanning*, dan garis-garis horizontal disebut garis-garis penyapuan (*scanning lines*).



Gambar 2-22 Garis Penyapuan (Scanning).

Sumber : Teknik Televisi dan Kamera jilid 1 : 74



Gambar 2-23 pengulangan berurutan.

Sumber : Teknik Televisi dan Kamera jilid 1 : 74

2.8.2. Proses Interlaced Scanning.

Proses interlaced scanning yang dipakai untuk televisi belum selesai dengan proses scanning pertama, yang dilakukan dari garis atas ke garis bawah pada permukaan gambar. Proses scanning pertama dilakukan secara kasar untuk garis-garis yang berbeda, kemudian scanning kedua dilakukan dari bagian atas, diantara garis perabaan yang tidak diraba, ke bagian bawah. Sistem perabaan secara berganda tersebut dinamakan perabaan antara atau *interlaced Scanning* .

Proses interlaced scanning ini mempunyai keuntungan mengurangi jalur frekuensi yang ditempati oleh signal video menjadi setengah tanpa mengurangi kualitas gambar. Sedangkan jalur frekuensi dari signal video tidak mengalami perubahan.

2.8.3. Proses Sinkronisasi

Untuk dapat memproduksi gambar pada permukaan fosfor tabung gambar yang sama dengan apa yang dikirimkan, maka diperlukan penyesuaian yang benar dengan ulasan yang telah terurai pada bagian pengirim dan pada bagian penerima ulasan harus tersusun kembali. Signal sinkronisasi dikirim bersama-sama signal video yang bertujuan untuk mengatur kecepatan dan fasa pengulasan (*scanning*), untuk melaksanakan pengulasan diantara yang mantap antara pengiriman dan penerimaan. Signal sinkronisasi horisontal diperlukan untuk pengulasan horizontal dan signal sinkronisasi vertikal untuk pengulasan vertikal. Kedua signal sinkronisasi tersebut disisipkan dalam periode penggelapan dan berbentuk segi empat dengan warna hitam yang lebih hitam dari nilai hitam.

2.9. Keypad (4 kolom x 7 baris)

Untuk mengatur konfigurasi dan mengendalikan sistem, maka diperlukan unit masukan ke sistem melalui keypad atau papan tombol. Karena perlu masukan data berupa angka dari 0 sampai 9 dan huruf dari A sampai Z, dan beberapa tombol lagi untuk keperluan lain maka ada 28 tombol masukan yang diperlukan.

Dengan X1, X2, X3, X4, X5, X6 dan X7 untuk baris dan Y1, Y2, Y3, dan Y4 untuk kolom, maka dapat dibuat keypad 4x7 sehingga keseluruhan ada 28 tombol. Posisi tombol pada keypad 4x7 seperti yang ditunjukkan pada Gambar 2-24 dibawah ini.

A/0	B/1	C/2	D/3
E/4	F/5	G/6	H/7
I/8	J/9	K	L
M	N	O	P
Q	R	S	T
U	V	W	X
Y	Z	ENTER	CANCEL

Gambar 2-24. Posisi Tombol Pada Keypad 4x7

Sumber : Perancangan

Tabel 2-9. Kombinasi Masukan Keypad 4x7

Switch	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	ENTER	CANCEL
Position	X1	X1	X1	X1	X2	X2	X2	X2	X3	X3	X3	X3	X4	X4	X4	X4	X5	X5	X5	X5	X6	X6	X6	X6	X7	X7	X7	X7
Output	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4
A	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
B	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
C	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
D	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

Sumber : Perancangan

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

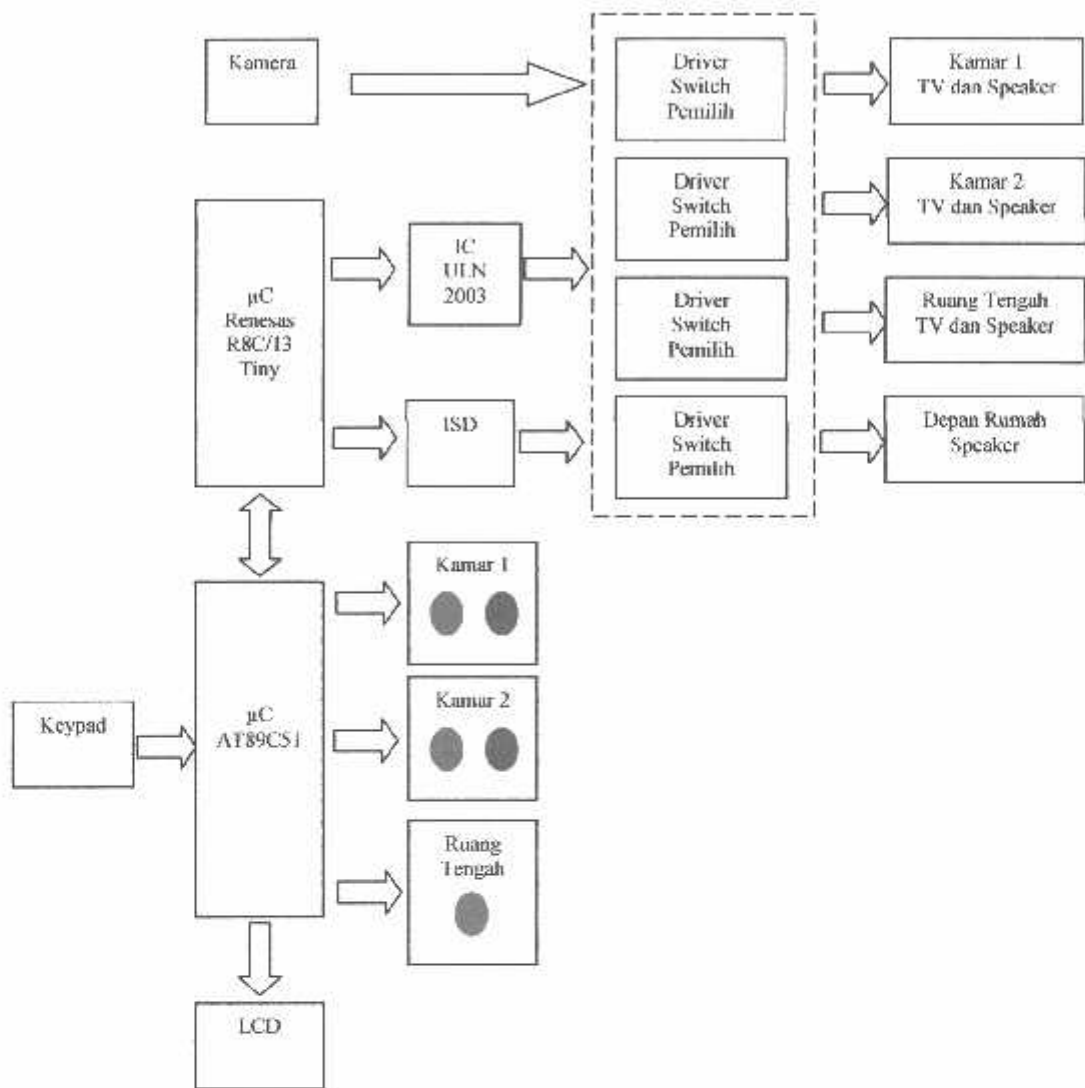
Dalam bab ini akan dibahas perancangan dan pembuatan alat. Pembahasan akan dilakukan pada setiap blok rangkaian, cara kerja masing-masing blok rangkaian, perhitungan dan fungsi masing-masing blok rangkaian tersebut. Secara garis besar terdapat dua bagian perangkat yang ada yaitu :

- Perancangan perangkat keras (*Hardware*).
- Perancangan perangkat lunak (*Software*).

Pada perancangan perangkat keras akan meliputi seluruh *peripheral* yang digunakan pada sistem ini. Pada perancangan perangkat lunak akan meliputi diagram alir dan *software* secara umum. Akan tetapi kedua perangkat ini dalam kerjanya akan saling menunjang satu sama lain.

3.1. Perancangan Perangkat Keras

Gambar 3-1. merupakan blok diagram keseluruhan sistem yang secara umum terdiri dari masukan-masukan dan keluaran-keluaran yang diproses oleh mikrokontroler. Dalam blok diagram terdapat 9 bagian yaitu *Mikrokontroler Renesas R8C/13 Tiny R5F21134FP* sebagai master, *Mikrokontroler AT89C51* sebagai slave, *Keypad*, *LCD*, *ISD*, Kamera *CCTV*, *TV*, *Speaker* dan Driver Switch Pemilih menggunakan *Relay*.



Gambar 3-1. Blok Diagram Bel-Kamera Pada Rumah Kos

Sumber : Perancangan

Fungsi Masing-Masing Blok Diagram

Fungsi dari tiap-tiap blok diagram dijelaskan sebagai berikut :

1. Mikrokontroler Renesas R8C/13 Tiny

Mikrokontroler sebagai pengolah data dan pengontrol keseluruhan kinerja sistem dan sebagai mikrokontroler utama atau master

2. Mikrokontroler AT89C51

Sebagai mikrokontroler slave yang berhubungan dengan mikrokontroler master.

3. Keypad

Keypad digunakan sebagai masukan nomor kode yang dikehendaki oleh pemakai alat.

4. LCD

LCD digunakan untuk menampilkan nomor kode, dan apakah nomor kode yang dimasukan ada atau tidak.

5. ISD

ISD berfungsi sebagai perekam suatu pesan berupa suara pendek dan dapat memutar kembali pesan tersebut melalui speaker.

6. Kamera

Kamera berfungsi untuk melihat dan mengetahui siapa yang datang berkunjung

7. Driver Switch Pemilih (Relay)

Driver Switch Pemilih berfungsi untuk menentukan kamar mana yang aktif speaker dan TVnya sesuai dengan masukan pada keypad

8. Speaker

Speaker sebagai output dari sistem ini dan berfungsi untuk mengeluarkan suara yang terpasang di masing-masing kamar, dan satu lagi yang terpasang di depan rumah.

9. TV

TV berfungsi untuk menampilkan gambar dari kamera yang terdapat di tiap-tiap kamar.

10. Lingkaran warna biru dan merah adalah tombol Push ON yang terdapat di tiap-tiap kamar.

Cara Kerja

Cara kerja alat ini diawali dengan orang yang tinggal di kos tersebut mengetikkan no kamar dan namanya ke keypad dan menyimpannya, kemudian merekam suaranya melalui mic. Kemudian apabila ada orang yang tinggal di kos tersebut pindah dan digantikan dengan orang lain, maka orang yang baru tersebut dapat mengganti no kamar dan nama yang lama dengan yang baru melalui keypad, demikian pula dengan suaranya dapat diganti pula melalui mic.

Apabila ada orang yang datang berkunjung ke kos tersebut kemudian mengetikkan no kamar pada keypad dan menampilkan no kamar tersebut ke LCD, kemudian data no kamar tersebut diproses oleh mikrokontroler dan apabila data tersebut benar, maka ISD akan bekerja dan memutar suara yang telah direkam ke kamar orang yang dicari. Penggunaan kamera adalah untuk melihat orang yang datang berkunjung ke kos tersebut dan akan ditampilkan ke TV di kamar orang

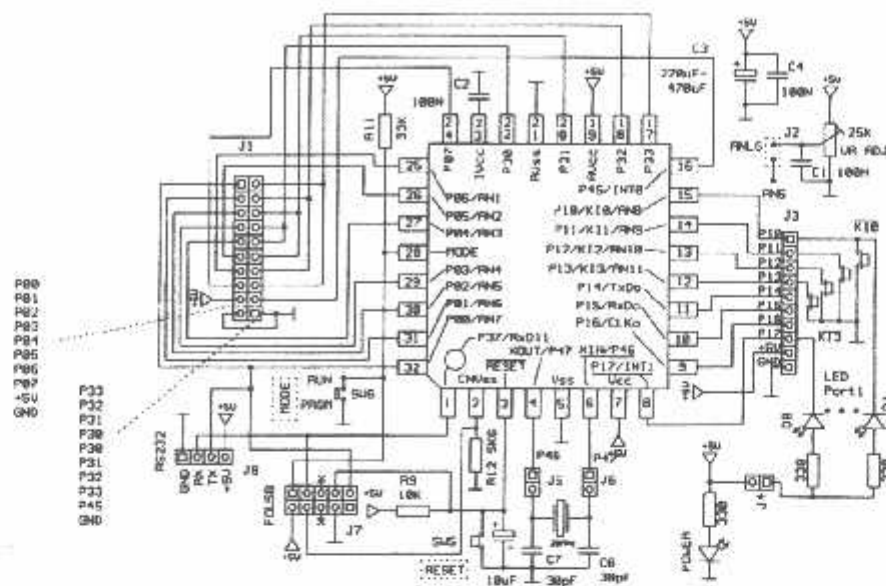
yang dicari. Apabila tidak ada yang memasukan no kamar ke keypad, maka kamera akan terhubung ke TV di ruang tengah.

Apabila orang yang dicari tersebut tidak ada, maka di LCD akan ditampilkan pesan bahwa orang yang dicari sedang keluar, dan orang yang datang berkunjung tersebut dapat meninggalkan pesan berupa suara singkat melalui mic. Dan pesan tersebut dapat didengar apabila penghuni kos tersebut sudah pulang.

3.1.1. Perancangan Minimum System Mikrokontroler Master Renesas R8C/13 Tiny R5F21134FP.

Mikrokontroler untuk *system minimum* dalam hal ini menggunakan mikrokontroler *Renesas R8C/13 Tiny R5F21134FP* karena mempertimbangkan keunggulan-keunggulannya dan fasilitas-fasilitas yang dimiliki IC ini. Pembahasan ini telah dijelaskan pada bab-bab sebelumnya. Mikrokontroler ini mempunyai I/O Port yaitu P0.0 – P0.7, P1.0 – P1.7, P3.0 – P3.3, P3.7 sedangkan P4.6, P4.7 hanya bisa digunakan sebagai *input* saja, bila konfigurasi kristal memakai kristal *internal*. Dalam hal ini yang digunakan Port I/O saja. Berikut adalah konfigurasi pin-pin mikrokontroler :

1. P0.0 – P0.3 = *Inputan* data dari AT89C51
2. P1.0 – P1.7 = *Output* data ISD 2590
P3.0 = Play ISD 2590
P3.1 = Rec ISD 2590
3. P3.2, P3.3, P3.7 = *Output* Relay



Gambar 3-2. Rangkaian *Minimum System R8C/13 Tiny R5F21134FP*

Sumber : www.renesas.com

3.1.2. 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. Dengan demikian pemakaian M1632 menjadi sederhana, sistem lain yang M1632 cukup mengirimkan kode-kode ASCII dari informasi yang ditampilkan seperti layaknya memakai sebuah printer.

Untuk berhubungan dengan mikrokontroler, M1632 dilengkapi dengan 4 jalur data (**DB0..DB7**) yang dipakai untuk menyalurkan kode ASCII maupun perintah pengatur kerjanya M1632. Selain itu dilengkapi pula dengan **E**, **R/W*** dan **RS** seperti layaknya komponen yang kompatibel dengan mikroprosesor.

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

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

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

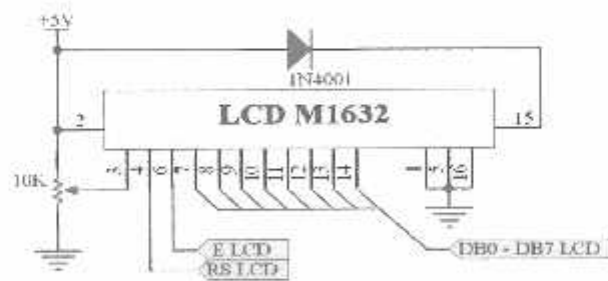
Proses mengirim/mengambil data ke/dari M1632 bisa dijabarkan sebagai berikut:

- **RS** harus dipersiapkan dulu, untuk menentukan jenis data seperti yang telah dibicarakan di atas.
- **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.
- Untuk mengambil data dari M1632 sinyal **R/W*** di-satu-kan, menyusul sinyal **E** di-satu-kan. Pada suatu **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 lain sebagainya, semua itu dibahas secara terperinci dalam Lembar Data M1632.

Untuk tampilan dipergunakan LCD Dot Matrik 2 x 16 karakter. Sinyal-sinyal yang diperlukan oleh LCD adalah RS dan Enable, sinyal RS dan Enable dipergunakan sebagai input yang outputnya dipakai untuk mengaktifkan LCD. LCD akan aktif apabila mikrokontroller memberikan instruksi tulis pada LCD. Saat kondisi RS don't care dan Enable 0 maka LCD tetap pada kondisi semula, pengiriman data ke LCD dilakukan saat RS berlogika 0 dan Enable berlogika 1. Instruksi dikirim pada LCD bila keadaan RS 1 dan Enable 1. Pin LCD ini untuk data terkoneksi pada Port 1 mikrokontoler Slave 1. Kemudian untuk RS dihubungkan pada Port 2.0, tulis/baca (*Read/Write*) diberikan logika low karena disini LCD bersifat menulis data, dan yang terakhir Enable (E) dikendalikan dengan Port 2.1.

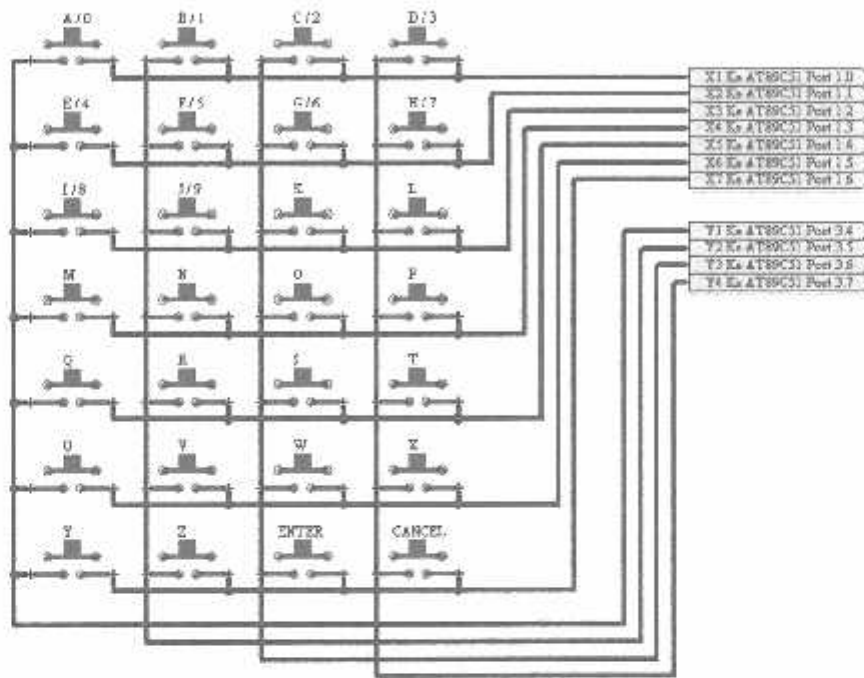


Gambar 3-3. Perancangan Rangkaian Liquid Crystal Display (LCD)

Sumber : Perancangan

3.1.3. Perencanaan dan Pembuatan Rangkaian Keypad

Untuk memasukan data pada mikrokontroler maka membutuhkan rangkaian keypad. Pada perencanaan dan pembuatan ini digunakan keypad berukuran 4x7. Gambar 3-4 di bawah ini memperlihatkan suatu keypad 4x7.



Gambar 3-4. Keypad Matrik 4x7

Sumber : Perancangan

Berdasarkan tabel, kita dapat dengan mudah menyeleksi tombol mana yang ditekan pada rangkaian keypad yang dibuat. Keluaran tombol yang dipilih dihubungkan langsung pada port Mikrokontroler AT89C51, yaitu : Port 1.0 untuk output X1, Port 1.1 untuk output X2, Port 1.2 untuk output X3, Port 1.3 untuk output X4, Port 1.4 untuk output X5, Port 1.5 untuk output X6, Port 1.6 untuk output X7, Port 3.4 untuk output Y1, Port 3.5 untuk output Y2, Port 3.6 untuk output Y3, dan Port 3.7 untuk output Y4.

Tabel 3-1. Kombinasi Masukan Keypad 4x7

Switch	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	ENTER	CANCEL	
Position	X1	X1	X1	X1	X2	X2	X2	X2	X3	X3	X3	X3	X4	X4	X4	X4	X5	X5	X5	X5	X6	X6	X6	X6	X6	X7	X7	X7	
Output	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	
A	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
B	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
C	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0
D	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1

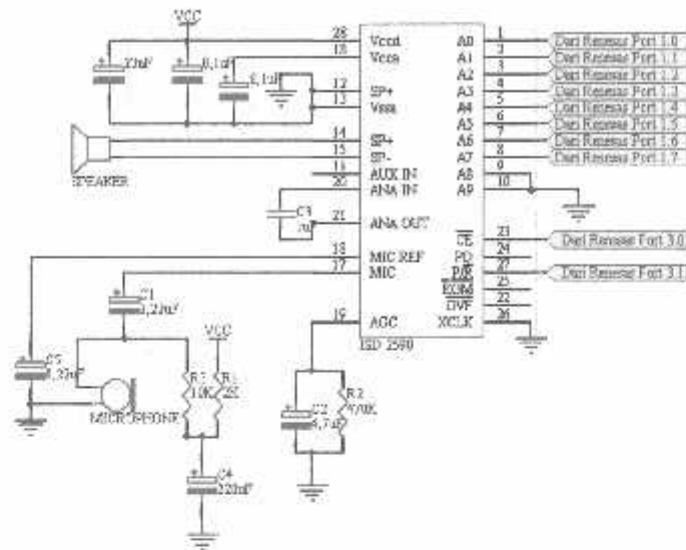
Sumber : Perancangan

3.1.4. Perancangan Rangkaian ISD 2590

Pada perancangan alat ini ISD (Information Storage Device) yang digunakan tipe ISD 2590, dimana ISD ini mempunyai kemampuan daya simpan suara selama 90 detik. Demikian juga untuk waktu putar ulangnya. ISD dengan tipe 2590 mempunyai rekaman suara akan disimpan dalam sel memori yang tidak mudah hilang (nonvolatile).rangkaiannya dapat dilihat pada Gambar 3.5.

Untuk merekam suara maka inputan dari Rec ini (pin27) harus dalam keadaan low, jika rec berlogika low dan play berlogika low (playback), maka perekaman akan mulai. Perekaman akan berhenti bila Pin REC dalam keadaan high, dan tanda untuk mengetahui bahwa pesan sudah berakhir (end of message marker) akan secara otomatis direkam.

End of message ini berguna supaya ISD tahu bahwa playback sudah berhenti dan IC ISD akan melanjutkan perintah berikutnya. Untuk playback ada dua cara pengontrolan yaitu dengan PLAYE dan PLAYL. Pada PLAYE (pin 24), pesan akan diputar pada saat sinyal input ini akan beralih ke low. Playback akan terus berjalan sampai akhir dari pesan. Pada PLAYL (pin 23), pesan akan diputar pada saat sinyal input berlogika high ke low, pesan akan terus berjalan sampai pesan berakhir.



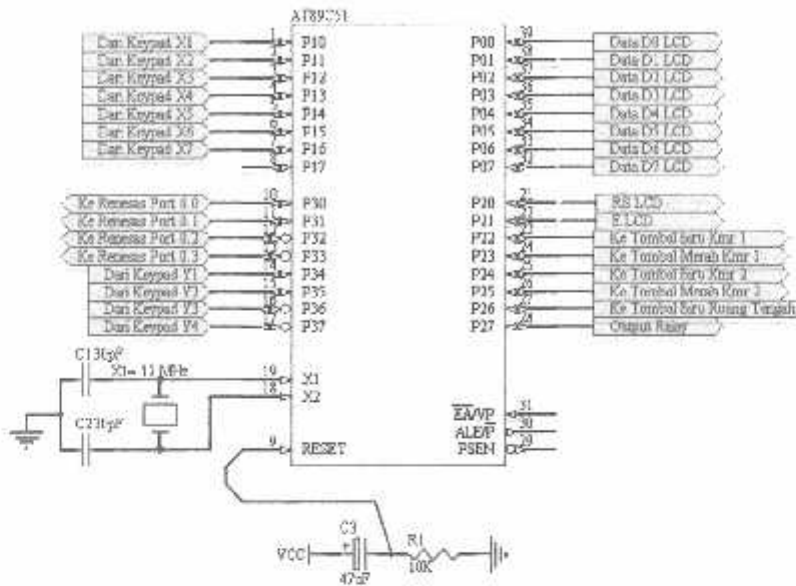
Gambar 3-5. Rangkaian ISD 2590

Sumber : perancangan

3.1.5. Rangkaian Mikrokontroler AT89C51

Mikrokontroler pada sistem berfungsi untuk menerima data masukan (X1....X7 dan Y1...Y4) dari keypad dan I.CD serta tombol. Berikut adalah konfigurasi pin-pin mikrokontroler :

1. P0.0 – P0.7 = Output Data LCD
 - P2.0 = RS
 - P2.1 = E
2. P1.0 – P1.6 = *Input* data (X1 sampai X7) dari Keypad
 - P3.4 – P3.7 = *Input* data (Y1 sampai Y4) dari Keypad
3. P3.0 – P3.3 = Data Ke Renesas
4. P2.2 – P2.6 = Tombol
5. P2.7 = Output Relay

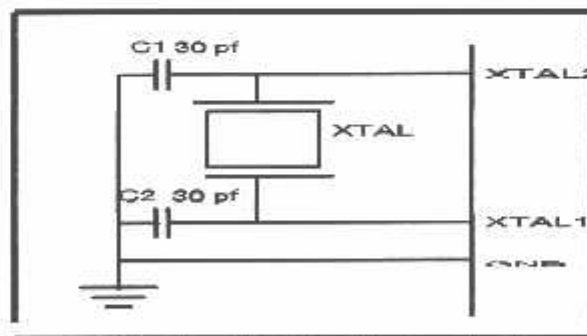


Gambar 3-6. Rangkaian Mikrokontroller

Sumber : perancangan

3.1.5.1. Rangkaian Clock (X-TAL 1 dan X-TAL 2, pin 19, 18)

Pin ini dihubungkan dengan kristal bila menggunakan *osilator internal*. X-TAL 1 merupakan masukan ke rangkaian *osilator internal* sedangkan X-TAL 2 keluaran dari rangkaian osilator internal. Untuk keperluan ini diperlukan kapasitor penstabil sebesar 30pF.



Gambar 3-7. Rangkaian Clock

Sumber : Perancangan

Untuk kapasitor C1 dan C2 dipilih harga 30pF, karena pada Mikrokontroler frekuensi maksimalnya 12 MHz, dimana 1 siklus mesin = 12 clock , sedangkan dalam rangkaian ini kristal yang dipakai mempunyai harga 12MHz. Maka langkah-langkah pemrosesan program yang dijalankan dengan frekuensi osilator sebesar 12 MHz adalah sebesar 1 μ s.

Siklus tersebut diambil berdasarkan ketentuan Mikrokontroler AT89C51 yaitu 12 clock = 1 siklus mesin, sedangkan frekuensi yang dipakai 12 MHz.

Jadi waktu yang dipakai dalam setiap siklus mesin, adalah:

$$T = \frac{1}{F_{(osc)}} \text{ dimana : } F_{(osc)} = 12 \text{ MHz}$$

Jadi siklus mesin mikrokontroler AT89C51 dengan kristal sebesar 12 MHz

adalah: $T = \frac{1}{12 \times 10^6}$

Satu siklus mesin = 12T

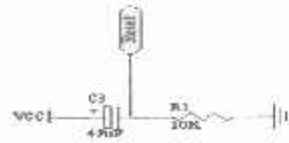
$$= 12 \frac{1}{12 \times 10^6}$$

$$= 1 \mu\text{s}$$

3.1.5.2. Rangkaian Reset (Pin 9)

Untuk melakukan *reset* sistem pada mikrokontroler yaitu untuk mengawali eksekusi program pada alamat paling rendah ,dapat dimanfaatkan pin *reset* yang ada pada mikrokontroler.

Pin 9 dihubungkan dengan rangkaian *reset* rangkaian ini diharapkan agar dapat mempunyai kemampuan *power ON Reset*, yaitu Reset terjadi saat *power* diaktifkan. Dibawah ini adalah adalah rangkaian *reset* :



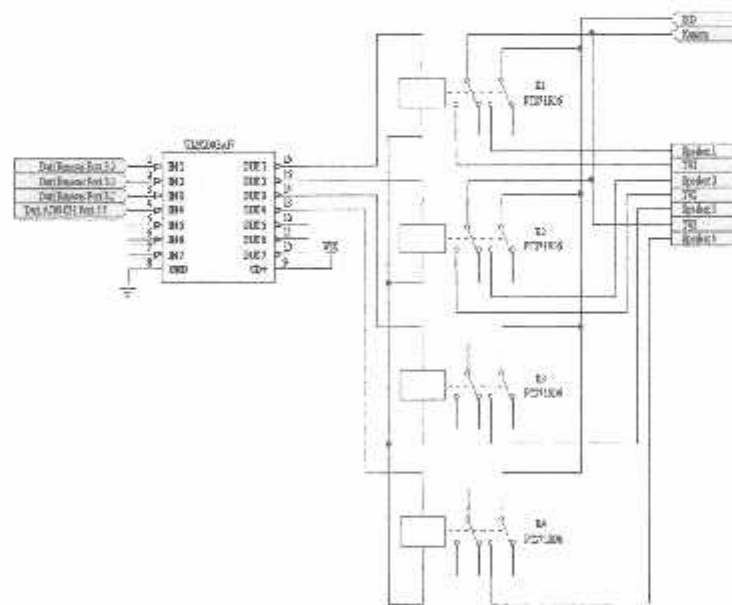
Gambar 3-8. Rangkaian Reset.

Sumber : Perancangan

Sehingga dengan komponen *resistor* dengan nilai 10 K serta kapasitor dengan nilai 47µF akan dihasilkan :

$$\begin{aligned}
 T &= R.C \ln 2 \\
 &= 10.10^3 \times 47.10^{-6} \times 0.693 \\
 &= 0.3257 \text{ s}
 \end{aligned}$$

3.1.6. Perancangan Rangkaian Relay Dengan Menggunakan IC ULN 2003



Gambar 3-9. Rangkaian Relay Dengan Menggunakan IC ULN 2003

Sumber : Perancangan

Untuk rangkaian ULN2003 sebagai saklar, dalam perancangan ini digunakan IC ULN2003 dan relay sebagai komponen utamanya, yang nantinya digunakan untuk mengaktifkan TV dan Speaker. Relay akan bekerja jika inputan pada IC ULN2003 diberi logika tinggi dimana data sheet menyebutkan bahwa IC ULN2003 dapat terpicu dengan tegangan 5 Volt dengan tegangan hubung sebesar 50 Volt dan arus maksimum yang diperbolehkan sebesar 500mA dengan suhu kerja dari -20°C sampai 80°C , dengan data yang ada diatas maka IC ULN2003 mampu digunakan untuk menghidup-matikan relay yang hanya memiliki tegangan maksimal sebesar 12 volt dengan resistansi kumparan sebesar 400 ohm jadi dapat diketahui arus relay sebesar:

$$\begin{aligned}\text{Dimana : } I_{\text{relay}} &= \frac{V_{cc}}{R_{\text{relay}}} \\ &= \frac{12}{400} \\ &= 30 \text{ mA.}\end{aligned}$$

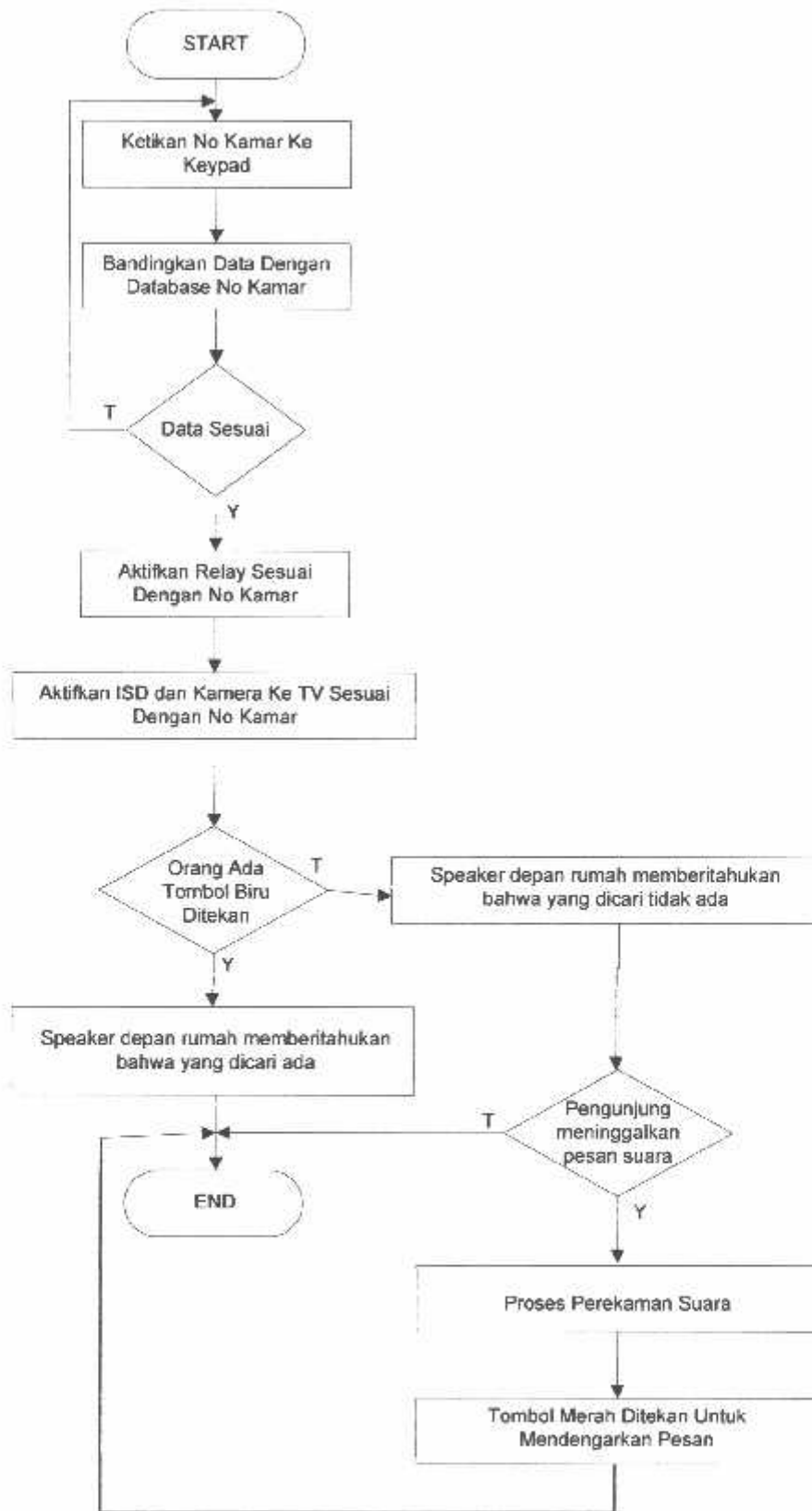
Dengan adanya arus *relay* sebesar 30 mA maka IC ULN2003A dapat menggerakkan *relay* tersebut karena ULN 2003AN memiliki arus maksimum sebesar 500 mA sesuai dengan data sheet.

3.2. Perancangan Perangkat Lunak (*Software*).

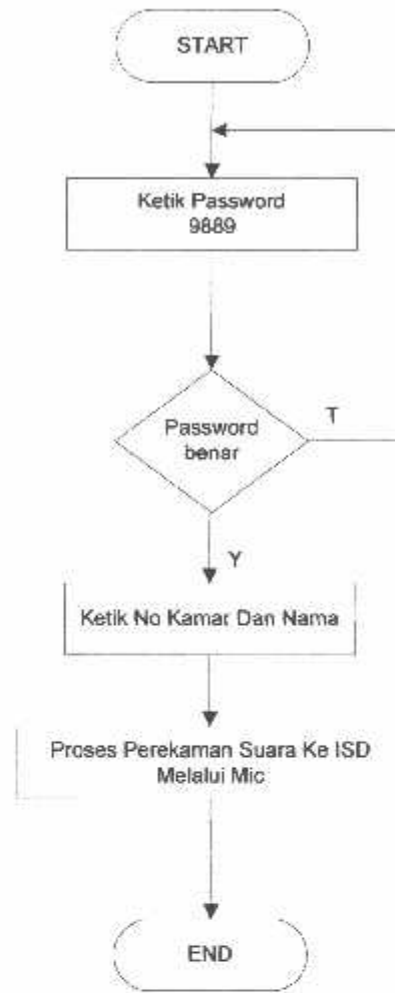
Pada perancangan perangkat lunak ini dipaparkan dalam diagram alir secara keseluruhan sistem namun pada sistem perangkat lunak *Dot Matrik* dijelaskan secara sub program pada *scan* datanya. Untuk bahasa pemrograman dalam hal ini dibagi menjadi 2 bagian karena disini menggunakan 2 software yang berbeda yakni :

- 1) Mikrokontroler Renesas R8C/Tiny R5F21134FP menggunakan pemrograman bahasa C dengan *Compiler* yang dipaket bersama pada suatu IDE yaitu HEW (*High-performance Embedded Workshop*).
- 2) Fasilitas lainnya yang dibawakan Renesas adalah *software emulator* KD30 dengan menggunakan fasilitas On-Chip Debugger R8C yang mempunyai kehandalan mengeliminasi kebutuhan akan *simulator software* dan dapat melakukan *debug* langsung pada *hardware*. *Development Tool Software* ini disediakan *freeware*-nya beserta *application-application note*-nya pada situs Renesas.

3.2.1. Flowchart



Gambar 3-10. Flowchart Bel - Kamera Pada Rumah Kos



Gambar 3-11. Flowchart Proses Perekaman Suara Penghuni Kos

BAB IV

PENGUJIAN ALAT DAN ANALISA SISTEM

Bab ini akan membahas pengujian alat yang telah dirancang, dirakit serta direalisasikan. Tujuan pengujian alat ini adalah mengetahui kerja dari masing-masing sistem yang dibuat secara per-blok. Dengan demikian dapat diketahui kepresisian kerja dari alat yang direncanakan dan dibuat. Secara umum tujuan dari pengujian alat tersebut adalah sebagai berikut :

1. Mengetahui proses kerja dari masing-masing rangkaian.
2. Mengetahui hasil dari suatu perencanaan yang telah dibuat.
3. Memudahkan perawatan dan perbaikan apabila sewaktu-waktu terjadi kerusakan.

Prosedur Pengujian :

1. Pengujian Keypad
2. Pengujian LCD
3. Pengujian Relay
4. Pengujian ISD
5. Pengujian AT89C51
6. Pengujian Renesas
7. Pengujian Keseluruhan

4.1. Pengujian Keypad 4 x 7

4.1.1. Tujuan Pengujian.

Untuk mengetahui apakah rangkaian keypad yang telah dibuat dapat bekerja sesuai dengan yang direncanakan.

4.1.2. Alat dan Bahan.

1. Rangkaian keypad yang diuji
2. Rangkaian Mikrokontroler AT89C51.
3. Programmer dan evaluator board
4. Catu daya

4.1.3. Prosedur Pengujian.

1. Menghubungkan rangkaian keypad kecatu daya.
2. Memprogram IC mikrokontroler kemudian menjalankannya.

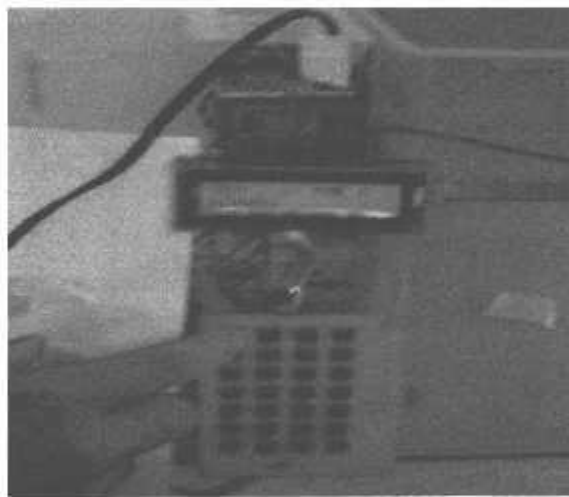
4.1.4. Hasil Pengujian.

Encoder keypad adalah berfungsi untuk mendeteksi adanya penekanan tombol pada keypad dan data penekanan tombol ini akan diterjemahkan kedalam bentuk kode-kode biner dalam jumlah 5 bit. Dalam perencanaan dan pembuatan alat ini menggunakan keypad 4 x 7 di mana outputnya (X1 sampai X7 dan Y1 sampai Y4) ke mikrokontroler AT89C51 untuk diproses, sehingga data-data yang dikeluarkan oleh keypad dapat diterjemahkan oleh mikrokontroler dengan software untuk menyesuaikan tombol yang telah ditekan. Dimana blok pengujian keypad dapat dilihat pada Gambar 4-1 dibawah ini.



Gambar 4-1. Blok Diagram Pengujian Rangkaian Keypad 4X7

Sumber : Pengujian



Gambar 4-2. Foto Rangkaian Keypad 4X7

Sumber : Pengujian

Adapun hasil dari pengujian keypad diperlihatkan pada Tabel 4.1 dibawah ini.

Tabel 4-1. Kombinasi Masukan Keypad 4x7

Switch	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	ENTER	CANCEL
Position	X1	X1	X1	X1	X2	X2	X2	X2	X3	X3	X3	X3	X4	X4	X4	X4	X5	X5	X5	X5	X6	X6	X6	X6	X7	X7	X7	X7
Output	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4
A	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
B	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
C	0	0	0	0	1	1	1	1	0	0	0	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
D	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

Sumber : Perancangan

4.2. Pengujian LCD

4.2.1. Tujuan Pengujian.

Untuk mengetahui apakah rangkaian LCD yang telah dibuat dapat bekerja sesuai dengan yang direncanakan.

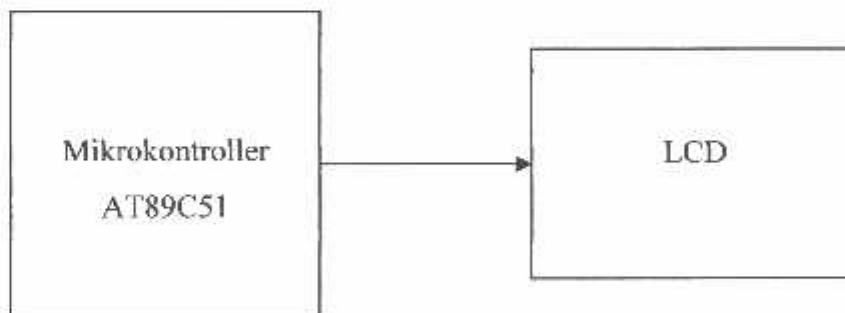
4.2.2. Alat dan Bahan.

1. Rangkaian LCD yang diuji
2. Rangkaian Mikrokontroler AT89C51.
3. Programmer dan evaluator board
4. Catu daya

4.2.3. Prosedur Pengujian.

1. Mengisi Mikrokontroler dengan software, dengan program yang dapat ditampilkan pada layar LCD berupa angka dan huruf.

Adapun cara pengujian dapat dilihat pada blok pengujian yang diperlihatkan pada Gambar 4.3 dibawah ini.

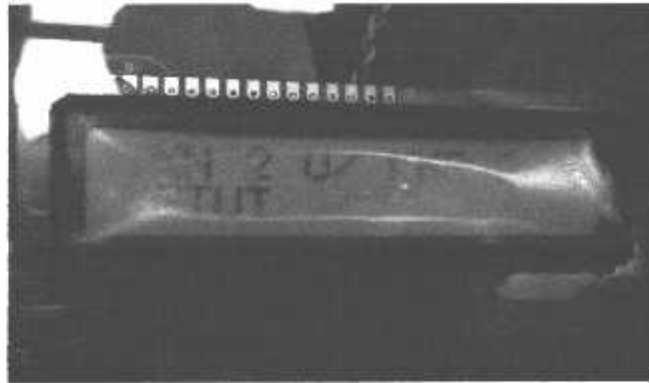


Gambar 4-3. Blok Diagram Pengujian Rangkaian LCD.

Sumber : Pengujian

4.2.4. Hasil Pengujian.

Foto rangkaian LCD dapat dilihat pada Gambar 4-4.



Gambar 4-4. Foto Rangkaian LCD

Sumber : Pengujian

4.3. Pengujian Rangkaian IC ULN 2003 dan Relay.

4.3.1. Tujuan Pengujian.

Untuk mengetahui apakah rangkaian IC ULN 2003 dan Relay yang telah dibuat dapat bekerja sesuai dengan yang direncanakan.

4.3.2. Alat dan Bahan.

1. Rangkaian IC ULN 2003 dan Relay yang diuji
2. Catu daya 5 Volt, untuk relay 12 volt

4.3.3. Prosedur Pengujian.

1. Menghubungkan rangkaian kecatu daya.
2. Mengukur arus yang mengalir pada relay menggunakan AVO Meter.

4.3.4. Hasil Pengujian

Untuk rangkaian ULN2003 sebagai saklar, dalam perancangan ini digunakan IC ULN2003 dan relay sebagai komponen utamanya, yang nantinya digunakan untuk mengaktifkan TV dan Speaker. Relay akan bekerja jika inputan pada IC ULN2003 diberi logika tinggi dimana data sheet menyebutkan bahwa IC ULN2003 dapat terpicu dengan tegangan 5 Volt dan arus maksimum yang diperbolehkan sebesar 500mA dengan suhu kerja dari -20°C sampai 80°C , dengan data yang ada diatas maka IC ULN2003 mampu digunakan untuk menghidup-matikan relay yang hanya memiliki tegangan maksimal sebesar 12 volt dengan resistansi sebesar 360 ohm jadi dapat diketahui arus relay.

Hasil Perhitungan :

$$I_{\text{relay}} = \frac{V_{cc}}{R_{\text{relay}}}$$
$$= \frac{12}{360}$$
$$= 0,033 \text{ A.}$$

Dengan adanya arus *relay* sebesar 33 mA maka IC ULN2003 dapat menggerakkan *relay* tersebut karena IC ULN 2003 memiliki arus maksimum sebesar 500 mA.

Hasil Pengukuran :



Gambar 4-5. Foto Hasil Pengukuran Relay

Sumber : Pengujian

Dari perhitungan dan pengukuran didapat hasil yang berbeda, maka dapat dicari % Error sebagai berikut :

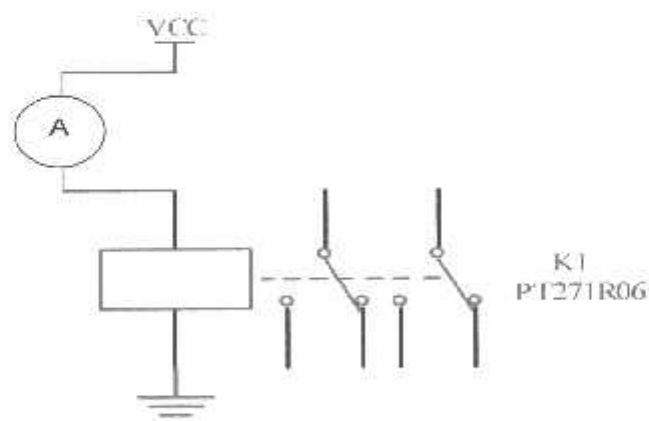
$$\begin{aligned} \% \text{ Error} &= \frac{I(\text{Perhitungan}) - I(\text{Pengukuran})}{I(\text{Perhitungan})} \times 100\% \\ &= \frac{0.033 - 0.03}{0.033} \times 100\% \\ &= 9.09\% \end{aligned}$$

Hasil perhitungan dan pengukuran serta %Error dapat dilihat dari Tabel 4-2. dibawah ini :

Tabel 4-2. Perbandingan Hasil Perhitungan Dan Pengukuran I relay

Pengukuran		Perhitungan		%Error
R (Ohm)	I relay (A)	R (Ohm)	I relay (A)	
360	0.03	360	0.033	9.09 %

Sumber : Pengujian



Gambar 4-6. Pengujian Rangkaian Relay

Sumber : Pengujian

Untuk mengetahui respon dari IC ULN2003 pada saat diberi inputan oleh mikrokontroller. Apabila inputan diberi logika *high*, maka IC ULN2003 akan menjadi aktif. Hal ini mengakibatkan *relay* dalam keadaan ON. Dan apabila inputan diberi logika *low* maka, IC ULN2003 akan menjadi tidak aktif. Hal ini mengakibatkan *relay* dalam keadaan OFF.

Tabel 4-3. Hasil Pengujian Untuk Mengetahui Respon Relay

Relay	Logika	Kondisi Speaker dan TV
Relay 1	<i>High</i>	Speaker dan TV Pada Kamar 1 Aktif
Relay 2	<i>High</i>	Speaker dan TV Pada Kamar 2 Aktif
Relay 3	<i>High</i>	Speaker Pada Ruang Tengah Aktif
Relay 4	<i>High</i>	Speaker Pada Depan Rumah Aktif

Sumber : Pengujian

Apabila tombol 1 pada keypad ditekan, maka Relay 1 dan Relay 3 akan aktif, sehingga speaker dan TV pada kamar 1 aktif dan speaker pada ruang tengah juga aktif.

Apabila tombol 2 pada keypad ditekan, maka Relay 2 dan Relay 3 akan aktif, sehingga speaker dan TV pada kamar 2 aktif dan speaker pada ruang tengah juga aktif.

Relay 4 berfungsi untuk mengaktifkan speaker di depan rumah yang digunakan untuk memberitahukan pengunjung bahwa penghuni kos yang dicari ada atau tidak ada di rumah.

4.4. Pengujian Rangkaian ISD 2590

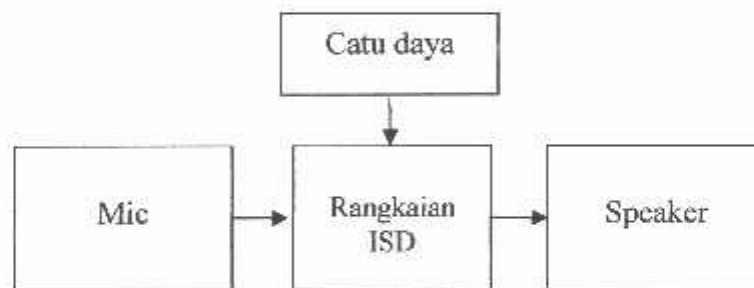
4.4.1. Tujuan Pengujian.

Untuk mengetahui apakah rangkaian ISD 2590 yang telah dibuat dapat bekerja sesuai dengan yang direncanakan

4.4.2. Alat dan Bahan.

1. Rangkaian ISD 2590
2. Catu daya 5 Volt
3. Mic
4. Speaker

4.4.3. Prosedur Pengujian



Gambar 4-7. Blok Diagram Pengujian Rangkaian ISD 2590

Sumber : Pengujian

Prosedur pengujian adalah sebagai berikut:

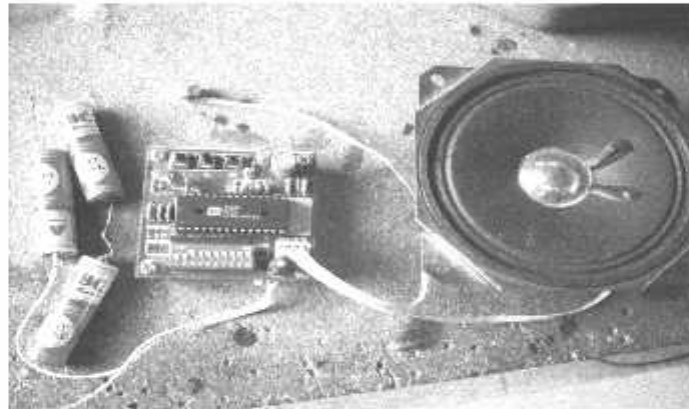
1. Rangkaian ISD diberi input tegangan 5 Volt Dc.
2. Pada pin 23 (CE) dan pin 27 (P/R) diberi logika rendah fungsinya untuk melakukan perekaman suara, setelah proses perekaman suara selesai pin-pin yang dibuat berlogika rendah dikembalikan lagi pada posisi tinggi (berlogika tinggi).

3. Untuk memutar ulang suara yang direkam maka posisi pin 23 (CE) diberi logika rendah sedangkan untuk pin 27 (P/R) tetap posisinya pada logika tinggi.

4.4.4. Hasil Pengujian.

Dalam perencanaan dan pembuatan alat ISD membutuhkan tegangan 5 Volt dan mendapat input control dari mikrokontroller. Dari hasil pengujian didapat data bahwa rangkaian ISD 2590 dapat merekam suara dan dapat pula memutar ulang suara tersebut.

Adapun Foto rangkaian ISD 2590 dapat dilihat pada Gambar 4-8. dibawah ini :



Gambar 4-8. Foto rangkaian ISD 2590

Sumber : Pengujian

Hasil pengujiannya dapat dilihat pada Tabel 4-4. dibawah ini :

Tabel 4-4. Hasil Pengujian Rangkaian ISD 2590.

Alamat (desimal)	Alamat (Heksadesimal)	Alamat (Biner)	Durasi (Detik)	Suara Yang Direkam
3	3H	0000 0011	5	Komang ada yang cari
18	12H	0001 0010	5	Komang ada di rumah
33	21H	0010 0001	5	Komang tidak ada di rumah
48	30H	0011 0000	5	Ketut ada yang cari
57	39H	0011 1001	5	Ketut ada di rumah
72	48H	0100 1000	5	Ketut tidak ada di rumah
87	57H	0101 0111	5	Untuk litip pesan tekan enter
102	66H	0110 0110	11	# Pesan 1 untuk kamar 1 #
147	93H	1001 0011	11	# Pesan 2 untuk kamar 1 #
192	C0H	1100 0000	11	# Pesan 1 untuk kamar 2 #
231	E7H	1110 0111	11	# Pesan 2 untuk kamar 2 #

Sumber : Pengujian

4.5. Pengujian Rangkaian Mikrokontroller AT89C51

4.5.1. Tujuan Pengujian.

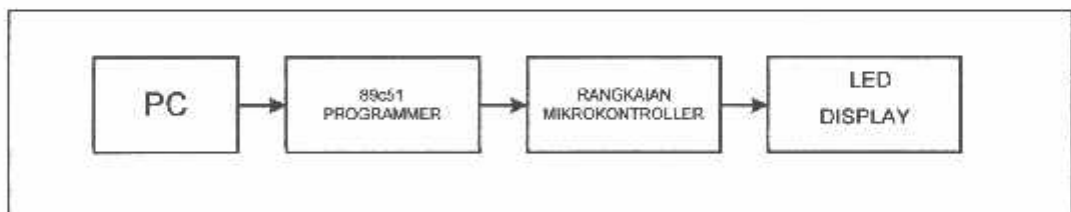
Tujuan dari pengujian mikrokontroller AT89C51 adalah untuk mengetahui apakah mikrokontroller yang dirancang telah dapat bekerja sesuai yang telah direncanakan.

4.5.2. Alat dan Bahan.

1. Rangkaian mikrokontroller yang diuji
2. Catu daya
3. LED peraga output mikrokontroller
4. Programmer dan evaluator board

4.5.3. Prosedur Pengujian.

1. Merancang software pengujian
2. Menghubungkan konektor led kepada setiap port secara bergantian dan mengamati hasil nyala LED.



Gambar 4-9. Blok Diagram Pengujian Mikrokontroller AT89C51

Sumber : Pengujian

4.5.4. Hasil Pengujian.

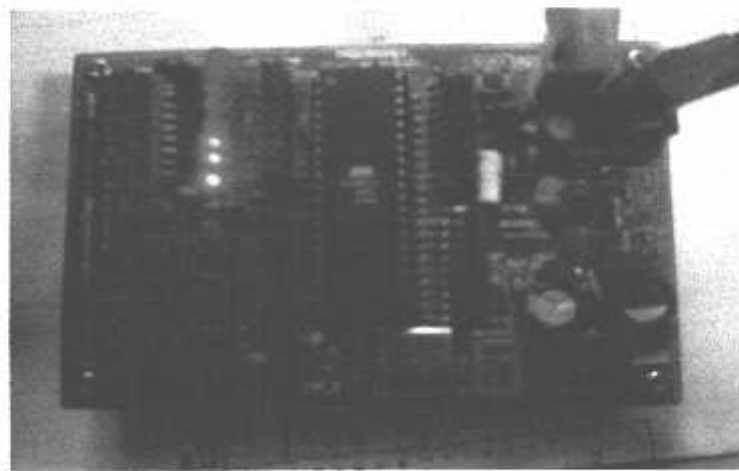
Setelah melakukan langkah langkah pengujian maka di dapatkan hasil pengujian yang ditunjukkan pada Tabel 4-5. dibawah ini.

Tabel 4-5. Hasil Pengujian Mikrokontroller AT89C51

Port Ke	Output Led							
	7	6	5	4	3	2	1	0
0	On	Off	On	Off	On	Off	On	Off
1	On	On	On	On	Off	Off	Off	Off
2	Off	Off	Off	Off	On	On	On	On
3	Off	Off	Off	Off	Off	Off	Off	Off

Sumber : Pengujian

Berdasarkan nyala LED yang diperoleh saat hasil pengujian alat menunjukkan nilai P0 = AAH, P1 = F0H, P2 = 0FH, P3 = 00H. dengan demikian maka dapat disimpulkan bahwa rangkaian mikrokontroller dapat berfungsi sebagaimana mestinya. Di bawah ini adalah foto pengujian mikrokontroller AT98C51 yang ditunjukan oleh Gambar 4-10.



Gambar 4-10. Foto Pengujian Rangkaian Mikrokontroller AT89C51

Sumber : Pengujian

4.6. Pengujian Rangkaian Mikrokontroller Renesas

4.6.1. Tujuan Pengujian

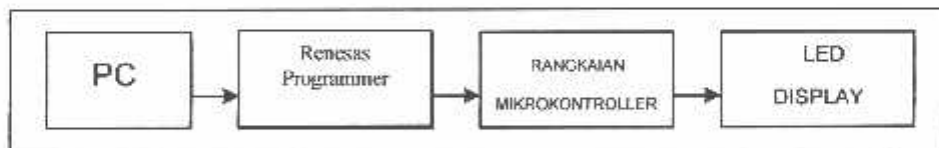
Tujuan dari pengujian mikrokontroller Renesas adalah untuk mengetahui apakah mikrokontroller yang dirancang telah dapat bekerja sesuai yang telah direncanakan.

4.6.2. Alat dan Bahan

1. Rangkaian mikrokontroller Renesas yang diuji
2. Catu daya 5 Volt
3. LED peraga output mikrokontroller
4. Programmer dan evaluator board

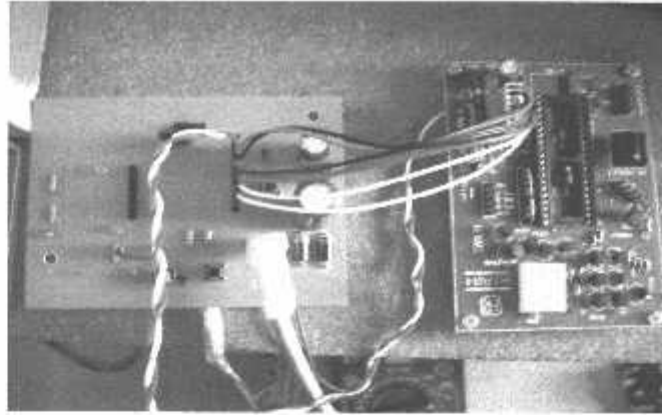
4.6.3. Prosedur Pengujian.

1. Merancang software untuk Mikrokontroller Renesas
2. Menghubungkan konektor led kepada setiap port secara bergantian dan mengamati hasil nyala LED.



Gambar 4-11. Blok Diagram Pengujian Mikrokontroller Renesas

Sumber : Pengujian



Gambar 4-12. Foto rangkaian Renesas

Sumber : Pengujian

4.7. Pengujian Rangkaian Secara Keseluruhan

4.7.1. Tujuan Pengujian

Tujuan dari pengujian rangkaian secara keseluruhan adalah untuk mengetahui apakah sistem yang telah dibuat dapat bekerja sesuai yang telah direncanakan.

4.7.2. Alat dan Bahan

1. Rangkaian alat keseluruhan
2. Catu daya 5 Volt, khusus untuk relay tegangannya 12 volt.

4.7.3. Prosedur Pengujian

1. Pengujian rangkaian secara keseluruhan dilakukan dengan menghubungkan masing – masing rangkaian perblok dan menjalankan perangkat lunak (software) yang telah dibuat.
2. Lakukan pengujian pada alat dengan menekan tombol 1 atau tombol 2 pada keypad.
3. Mengamati proses kerjanya dan mencatat hasilnya pada tabel.



Gambar 4-13. Foto Rangkaian Keseluruhan

Sumber : Pengujian

4.7.4. Hasil Pengujian

Tabel 4-6. Hasil Pengujian Penekanan Tombol

No.	Tombol yang ditekan oleh tamu	Tombol yang ditekan oleh penghuni kos	Speaker yang aktif	Suara pada speaker
1	Tombol 1		Speaker kamar 1 dan ruang tengah	Komang ada yang cari
2		Tombol biru ditekan	Speaker depan rumah	Komang ada
3		Tombol biru tidak ditekan	Speaker depan rumah	Komang tidak ada, untuk titip pesan tekan enter
4	Tombol Enter (Proses perekaman pesan suara)	Tombol merah	Speaker kamar 1	# Pesan #
5	Tombol 2		Speaker kamar 2 dan ruang tengah	Ketut ada yang cari
6		Tombol biru ditekan	Speaker depan rumah	Ketut ada
7		Tombol biru tidak ditekan	Speaker depan rumah	Ketut tidak ada, untuk titip pesan tekan enter
8	Tombol Enter (Proses perekaman pesan suara)	Tombol merah	Speaker kamar 2	# Pesan #

Sumber : Pengujian

Tabel 4-7. Hasil Pengujian Perekaman Suara

No.	Kamar yang dipilih	Masukan nama	Suara yang disimpan	Durasi perekaman suara
1	1	Komang	Komang ada yang cari	3 detik
			Komang ada	3 detik
			Komang tidak ada	3 detik
2	2	Ketut	Ketut ada yang cari	3 detik
			Ketut ada	3 detik
			Ketut tidak ada	3 detik

Sumber : Pengujian

Tabel 4-8. Hasil Pengujian Tampilan Kamera Pada TV

No.	Tombol yang ditekan	Kondisi TV
1	Tombol 1	TV kamar 1 aktif
2	Tombol 2	TV kamar 2 aktif
3		TV ruang tengah aktif terus

Sumber : Pengujian



Gambar 4-14. Foto Tampilan Kamera Pada TV

Sumber : Pengujian

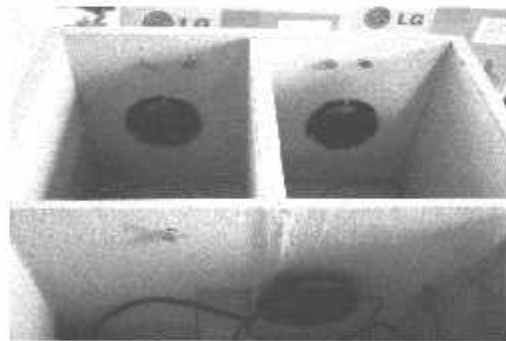
Dari pengujian secara keseluruhan terlihat bahwa alat bel-kamera pada rumah kos yang menggunakan mikrokontroler Renesas R8C/13 Tiny tersebut dapat bekerja sesuai dengan yang telah direncanakan.

4.8. Spesifikasi Alat.

Dalam pembuatan alat yang dibuat dimana panjangnya 50 cm dan lebar 50 cm untuk miniatur dengan 2 kamar dan 1 ruang tengah. Dimana memakai tegangan kerja 220 volt, input untuk rangkaian mikrokontroller, LCD, ISD, IC ULN 2003 sebesar 5 volt DC sedangkan untuk relay membutuhkan tegangan input sebesar 12 volt DC.

Spesifikasi yang digunakan :

1. Mikrokontroller Renesas R8C/13 Tiny sebagai master
2. Mikrokontroller AT89C51 sebagai slave
3. LCD M1632
4. Keypad 4 X 7
5. ISD 2590
6. IC ULN 2003AN
7. Kamera CCTV
8. Speaker 4 buah
9. Tombol Biru 3 buah
10. Tombol Merah 2 buah



Gambar 4-15. Foto Miniatur Posisi Speaker dan Tombol di tiap-tiap kamar

Sumber : Pengujian

BAB V

PENUTUP

5.1 Kesimpulan

Dari perancangan dan pembuatan bel-kamera untuk rumah kos berbasis mikrokontroller Renesas R8C/13 Tiny, maka dapat diambil kesimpulan sebagai berikut :

1. Untuk komunikasi dua mikro menggunakan komunikasi paralel dengan alamat data dari 0xf0 sampai 0xfe
2. ISD yang digunakan adalah ISD 2590 yang dapat menyimpan suara dengan durasi 90 detik, dimana suara penghuni kos yang direkam untuk masing-masing alamat 5 detik dan untuk menitipkan pesan masing-masing alamat 11 detik.
3. Waktu yang diperlukan antara penekanan tombol kamar yang pertama dan penekanan tombol kamar berikutnya adalah 10 sampai 15 detik.
4. Pada proses perekaman suara, jarak ke mic ± 3 cm agar kualitas suara yang direkam lebih baik.
5. Tampilan kamera pada TV kamar 1 atau kamar 2 aktif antara 10 sampai 15 detik tergantung penekanan tombol biru, sedangkan TV ruang tengah aktif terus.

5.2 Saran

1. Dalam pembuatan alat ini sebaiknya menggunakan komponen dengan kualitas yang lebih baik agar didapatkan hasil yang lebih baik.

DAFTAR PUSTAKA

- [1] Agfianto Eka Putra, 2002, "*Belajar Mikrokontroller AT 89S51/52/55*", Gava Media Yogyakarta.
 - [2] Budioko, Totok, 2005, "*Belajar Dengan Mudah dan Cepat Pemrograman Bahasa C Dengan SDCC (Small Device C Compiler) Pada Mikrokontroler AT89X05 / AT89C51/52 Teori, Simulasi dan Aplikasi*", Gava Media, Yogyakarta.
 - [3] *LCD Modul User Manual*, Seiko Instrument Ing, 1987
 - [4] Sutiono, Budi, 2002, "*Teknik Antarmuka dan Pemrograman Mikrokontroller AT89C51*", Balai Pustaka, Jakarta.
 - [5] Suarka, Agus, 2000, "*Elektronika Dalam Industri*", PT. Gramedia Pustaka Utama, Jakarta.
 - [6] Setiawan, Sulhan, 2004, "*Teknik Televisi dan Kamera jilid 1*", Penerbit ANDI Yogyakarta.
 - [7] www.renesas.com
 - [8] www.alldatasheet.com
 - [9] www.atmel.com
 - [10] www.national.com/semiconductor
-

LAMPIRAN



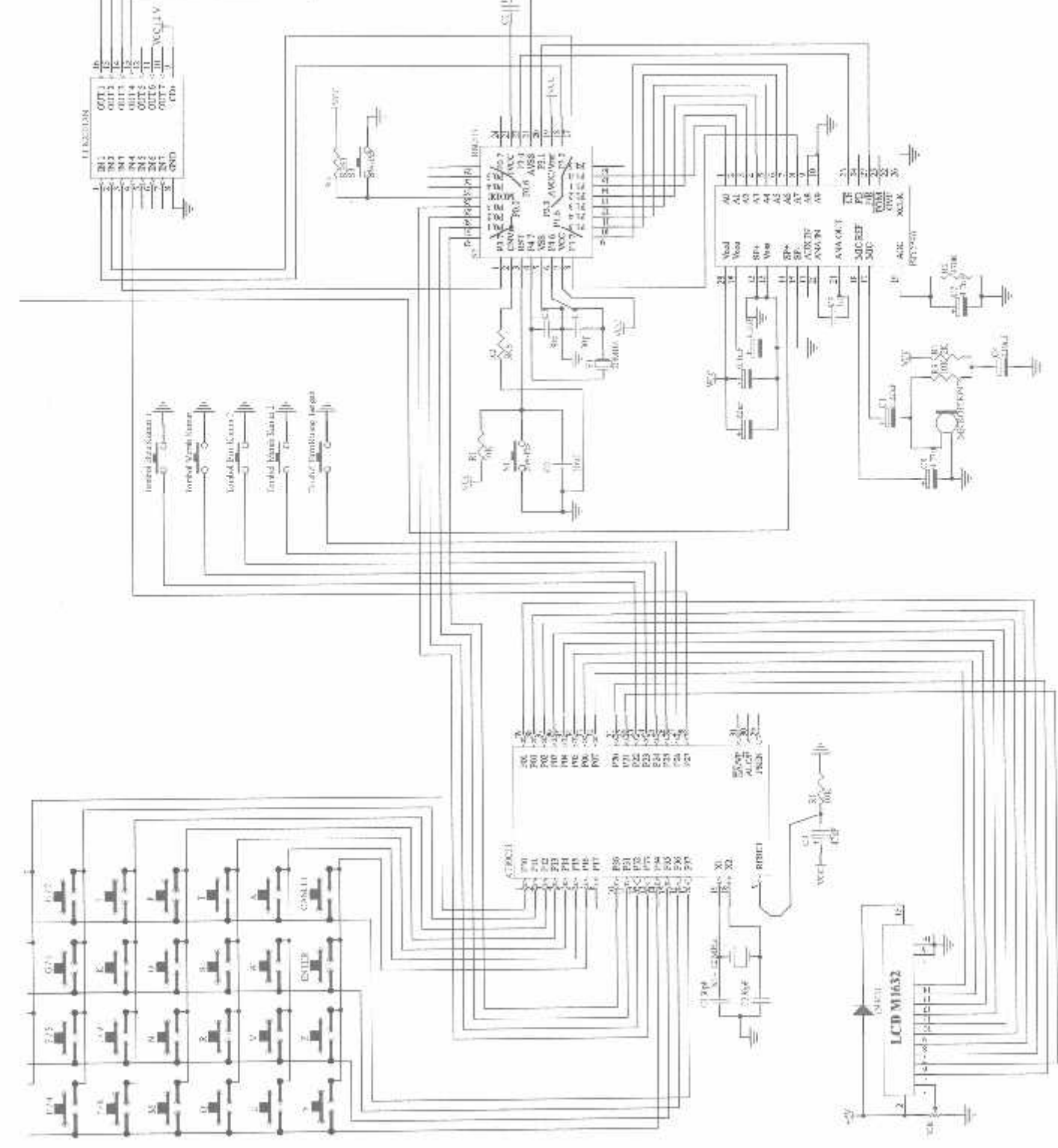
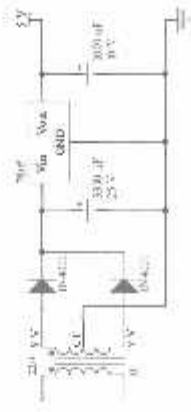
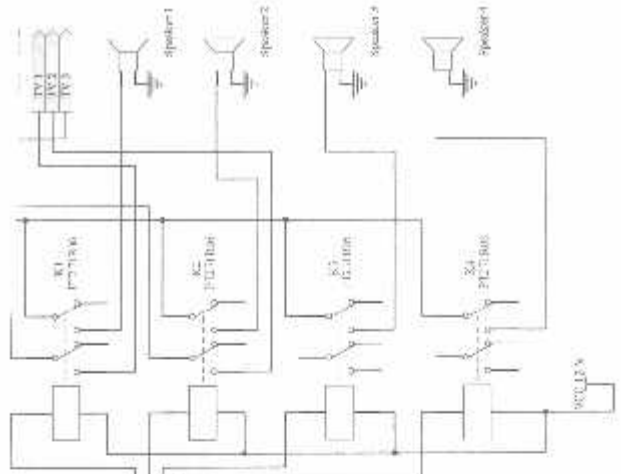


Table with 3 columns: No, Nomor, and Revisi.

No	Nomor	Revisi
1	1	1

Sheet 1 of 1
 Date: 2023/07/15
 File: C:\Users\user\Documents\Spring 2023\2202232312007.rtf



FORMULIR BIMBINGAN SKRIPSI

Nama : Komang Hadi Suryata
NIM : 02.17.045
Masa Bimbingan : 22 Mei 2007 – 22 November 2007
Judul Skripsi : Perancangan Dan Pembuatan Bel - Kamera Untuk Rumah Kos Berbasis Mikrokontroler Renesas R8C/13 Tiny

NO	Tanggal	Uraian	Paraf Pembimbing
1	1/8 ⁰⁷	Revisi bab I	
2	4/8 ⁰⁷	Revisi bab II & bab III	
3	7/8 ⁰⁷	Acc bab I, II, III	
4	9/8 ⁰⁷	Revisi bab III	
5	15/8 ⁰⁷	Acc bab IV, revisi bab V	
6	18/8 ⁰⁷	Acc bab V	
7	20/8 ⁰⁷	Acc Seminar hasil	
8	29/8 ⁰⁷	Revisi daftar isi	
9	31/8 ⁰⁷	Acc daftar isi	
10	2/9 ⁰⁷	Acc Ujian Skripsi	

Malang,
Dosen Pembimbing I

(Ir. Eko Nurcahyo)
NIP. 1028700172



FORMULIR BIMBINGAN SKRIPSI

Nama : Komang Hadi Suryata
NIM : 02.17.045
Masa Bimbingan : 22 Mei 2007 – 22 November 2007
Judul Skripsi : Perancangan Dan Pembuatan Bel - Kamera Untuk Rumah Kos Berbasis Mikrokontroler Renesas R8C/13 Tiny

NO	Tanggal	Uraian	Paraf Pembimbing
1	2 April 07	- Bab I, II, III	
2		- Pengujian system	
3		- Lembaran & Saran	
4		- Makalah Seminar	
5		- Bab IV	
6		- Pengujian system	
7		- Pengujian Hardware	
8		- Pengujian Software	
9		- Att Kompre!	
10			

Malang,
Dosen Pembimbing II

(M. Ashar, ST, MT)



FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian komprehensif jenjang Strata Satu (S-1) Jurusan Teknik Elektro, Konsentrasi Elektronika yang diselenggarakan pada:

Hari : Kamis
Tanggal : 6 September 2007

Telah dilakukan perbaikan skripsi oleh :

Nama : Komang Hadi Suryata
N.I.M : 02.17.045
Masa Bimbingan : 22 Mei 2007 – 22 November 2007
Judul Skripsi : Perancangan Dan Pembuatan Bel-Kamera Untuk Rumah Kos Berbasis Mikrokontroler Renesas R8C/13 Tiny

Perbaikan Meliputi :

No	Materi Perbaikan	Paraf Dosen Penguji
1.	Kata-kata "Tugas Akhir" harap dirubah "Skripsi"	
2.	Gambar rangkaian power supply	

Diperiksa dan Disetujui,
Dosen Penguji Pertama

(Ir. Teguh Herbasuki, MT)
NIP.Y.1038900209

Mengetahui,

Dosen Pembimbing I

(Ir. Eko Nurcahyo)
NIP.Y.1028700172

Dosen Pembimbing II

(M. Ashar, ST, MT)
NIP.1030500408



FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian komprehensif jenjang Strata Satu (S-1) Jurusan Teknik Elektro, Konsentrasi Elektronika yang diselenggarakan pada:

Hari : Kamis
Tanggal : 6 September 2007

Telah dilakukan perbaikan skripsi oleh :

Nama : Komang Hadi Suryata
N.I.M : 02.17.045
Masa Bimbingan : 22 Mei 2007 – 22 November 2007
Judul Skripsi : Perancangan Dan Pembuatan Bel-Kamera Untuk Rumah Kos Berbasis Mikrokontroler Renesas R8C/13 Tiny

Perbaikan Meliputi :

No	Materi Perbaikan	Paraf Dosen Penguji
1.	Kesimpulan ambil dari beberapa pengujian	

Diperiksa dan Disetujui,
Dosen Penguji Kedua

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

Mengetahui,

Dosen Pembimbing I

(Ir. Eko Nurcahyo)
NIP.Y.1028700172

Dosen Pembimbing II

(M. Ashar, ST, MT)
NIP. 1030500408

```

#include <stdio.h>
#include "sfr_r813.h" /* Definition of the R8C/13 SFR */
#include "TombolLED.h"
int cout = 0;
int cout1 = 0;
void Delay_ms (int n)
{
    prex = 99;
    tx = 199;
    txs = 1;
    while(n>0)
    {
        while(txundl - 1);
        txund = 0;
        n--;
    }
    txs = 0;
}

/*****
* Function : main()
* program section
*****/
void main()
{
    int batas; // definisi variable batas
    asm("FCLR I"); /* Interrupt disable */
    prcr = 1; /* Protect off */
    cm13 = 1; /* X-in X-out */
    cm15 = 1; /* XCIN-XCOUT drive capacity select bit : HIGH */
    cm05 = 0; /* X-in on */
    cm16 = 0; /* Main clock = No division mode */
    cm17 = 0;
    cm06 = 0; /* CM16 and CM17 enable */
    asm("nop");
    asm("nop");
    asm("nop");
    asm("nop");
    ocd2 = 0; /* Main clock change */
    prcr = 0; /* Protect on */

    p1 = 0xFF;
    pd1 = 0xFF;
    p0 = 0x00; /* Port 0 : Matikan seven segment */
    prc2 = 1; /* Proteksi untuk PD0 off */
    pd0 = 0x00; /* Port 0 : Semua pin pd Port 0 digunakan untuk output */
    p3 = 0xFF;
    pd3 = 0xFF;
    //pd4 = pd4 & 0xdf; /* P45 port direction = input */
    //pu11 = 1; /* P4_5 pull-up diaktifkan

    while(1)
    {

        RELAYA = 0;
        RELAYB = 0;
        RELAYC = 0;
    }
}

```

```
if (p0 == 0xf1)
{
    RELAYA = 1;
    Delay_ms (1000);
    RELAYC = 1;
    p1 = 0x03;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
    RELAYA = 0;
    RELAYC = 0;
}
```

```
if (p0 == 0xf2)
{
    p1 = 0x12;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
}
```

```
if (p0 == 0xf3)
{
    p1 = 0x21;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);

    p1 = 0x57;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
}
```

```
if (p0 == 0xf4)
{
    RELAYB = 1;
    Delay_ms (1000);
    RELAYC = 1;
    p1 = 0x30;
    Delay_ms (1000);
}
```

```
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
    RELAYB = 0;
    RELAYC = 0;
}

if (p0 == 0xf5)
{
    p1 = 0x39;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
}

if (p0 == 0xf6)
{
    p1 = 0x48;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);

    p1 = 0x57;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
}

if (p0 == 0xf7)
{
    p1 = 0x57;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    Delay_ms (1000);
    p1 = 0xff;
    Delay_ms (4000);
}
```

```

if (p0 == 0xf8)
{
    if (cout == 0)
    {
        cout = 1;
        p1 = 0x66;
        Delay_ms (1000);
        rec = 0;
        Delay_ms (1000);
        play = 0;
        Delay_ms (5000);
        play = 1;
        rec = 1;
        p1 = 0xff;
        Delay_ms (4000);
    }
    else
    {
        cout = 0;
        p1 = 0x93;
        Delay_ms (1000);
        rec = 0;
        Delay_ms (1000);
        play = 0;
        Delay_ms (5000);
        play = 1;
        rec = 1;
        p1 = 0xff;
        Delay_ms (4000);
    }
}

if (p0 == 0xfa)
{
    if (cout1 == 0)
    {
        cout1 = 1;
        p1 = 0xc0;
        Delay_ms (1000);
        rec = 0;
        Delay_ms (1000);
        play = 0;
        Delay_ms (5000);
        play = 1;
        rec = 1;
        p1 = 0xff;
        Delay_ms (4000);
    }
    else
    {
        cout1 = 0;
        p1 = 0xe7;
        Delay_ms (1000);
        rec = 0;
        Delay_ms (1000);
        play = 0;
        Delay_ms (5000);
        play = 1;
        rec = 1;
        p1 = 0xff;
    }
}

```

```

Delay_ms (4000);
}

if (p0 == 0xfc)
{
    RELAYA = 1;
    p1 = 0x66;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    p1 = 0xff;
    Delay_ms (5000);
    p1 = 0x93;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    p1 = 0xff;
    Delay_ms (5000);
    RELAYA = 0;
}

if (p0 == 0xfd)
{
    RELAYB = 1;
    p1 = 0xc0;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    p1 = 0xff;
    Delay_ms (5000);

    p1 = 0xe7;
    Delay_ms (1000);
    play = 0;
    Delay_ms (1000);
    play = 1;
    p1 = 0xff;
    Delay_ms (5000);
    RELAYB = 0;
}

if (p0 == 0xfe)
{
    p1 = 0x03;
    Delay_ms (1000);
    rec = 0;
    Delay_ms (1000);
    play = 0;
    Delay_ms (4000);
    play = 1;
    rec = 1;
    p1 = 0xff;
    Delay_ms (4000);

    p1 = 0x12;
    Delay_ms (1000);

```

```

rec = 0;
Delay_ms (1000);
play = 0;
Delay_ms (4000);
play = 1;
rec = 1;
p1 = 0xff;
Delay_ms (4000);

p1 = 0x21;
Delay_ms (1000);
rec = 0;
Delay_ms (1000);
play = 0;
Delay_ms (4000);
play = 1;
rec = 1;
p1 = 0xff;
Delay_ms (4000);
}

if (p0 == 0xf0)
{
p1 = 0x30;
Delay_ms (1000);
rec = 0;
Delay_ms (1000);
play = 0;
Delay_ms (4000);
play = 1;
rec = 1;
p1 = 0xff;
Delay_ms (4000);

p1 = 0x39;
Delay_ms (1000);
rec = 0;
Delay_ms (1000);
play = 0;
Delay_ms (4000);
play = 1;
rec = 1;
p1 = 0xff;
Delay_ms (4000);

p1 = 0x48;
Delay_ms (1000);
rec = 0;
Delay_ms (1000);
play = 0;
Delay_ms (4000);
play = 1;
rec = 1;
p1 = 0xff;
Delay_ms (4000);
}
}
}

```

```
; LCD CONSTANTA
DISPCLR EQU 00000001B
BLINK EQU 00001101B
ENTRMOD EQU 00000110B
DISPON EQU 00001100B
CURSOR EQU 00011100B
FUNCSET EQU 00111000B
```

```
;
;DEVICE ADDRESS [LCD]
RS BIT P2.0 ;LCD
E BIT P2.1 ;LCD
```

```
COD1 EQU 27H
COD2 EQU 28H
COD3 EQU 29H
DATAKEY0 EQU 30H
DATAKEY1 EQU 31H
DATAKEY2 EQU 32H
DATAKEY3 EQU 33H
DATAKEY4 EQU 34H
DATAKEY5 EQU 35H
DATAKEY6 EQU 36H
DATAKEY7 EQU 37H
DATAKEY8 EQU 38H
```

```
DATAKEY9 EQU 3AH
DATAKEY10 EQU 3BH
DATAKEY11 EQU 3CH
DATAKEY12 EQU 3DH
DATAKEY13 EQU 3EH
DATAKEY14 EQU 3FH
DATAKEY15 EQU 4AH
```

```
DATAKEY16 EQU 4BH
DATAKEY17 EQU 4CH
DATAKEY18 EQU 4DH
DATAKEY19 EQU 4EH
DATAKEY20 EQU 4FH
DATAKEY21 EQU 5AH
DATAKEY22 EQU 5BH
```

```
DATAKEY23 EQU 5CH
DATAKEY24 EQU 5DH
DATAKEY25 EQU 5EH
DATAKEY26 EQU 5FH
DATAKEY27 EQU 6AH
DATAKEY28 EQU 6BH
DATAKEY29 EQU 6CH
```

```
LOKASHI EQU 6DH
no2D EQU 6EH
DATAKEY0D EQU 6FH
DATAKEY1D EQU 7AH
DATAKEY2D EQU 7BH
```

DATAKEY3D	EQU	7CH
DATAKEY4D	EQU	7DII
DATAP3	EQU	7EH
NILAI2	EQU	7FH
DATAKEY30	EQU	39H
DATAKEY31	EQU	40H
DATAKEY32	EQU	41H
DATAKEY33	EQU	42H
DATAKEY34	EQU	43H
DATAKEY35	EQU	44H
DATAKEY36	EQU	45H
DATAKEY37	EQU	46H
LOKASI	EQU	47H
DATAKEY38	EQU	48H
DATAKEY39	EQU	49H
DATAKEY40	EQU	50H
DATAKEY41	EQU	51H
DATAKEY42	EQU	52H
DATAKEY43	EQU	53H
DATAKEY44	EQU	54H
DATAKEY45	EQU	55H
DATAKEY46	EQU	56H
DATAKEY47	EQU	57H
DATAKEY48	EQU	58H
DATAKEY49	EQU	59H
DATAKEY50	EQU	60H
DATAKEY51	EQU	61H
keydata	EQU	62h
keybounc	EQU	63H
COUNTER	EQU	64h
NILAIAB	EQU	65H
NIALIBA	EQU	66h
NILAIBB	EQU	67H
NILAICA	EQU	68h
NILAICB	EQU	69H
NILAIDA	EQU	70h
NILAIDB	EQU	71H
TAMPUNGA	EQU	72h
TAMPUNGB	EQU	73h
TAMPUNGD	EQU	75h
TAMPUNGF	EQU	76H
kolom1	BIT	P3.4
kolom2	BIT	P3.5
kolom3	BIT	P3.6
kolom4	BIT	P3.7
:		
baris1	BIT	P1.0
baris2	BIT	P1.1
baris3	BIT	P1.2
baris4	BIT	P1.3
baris5	BIT	P1.4

```

baris6      BIT      P1.5
baris7      BIT      P1.6
;
keyport     EQU      P3

```

```

        JMP  MULAI

```

```

MULAI:

```

```

        MOV DATAKEY0,#4BH
        MOV DATAKEY1,#4FH
        MOV DATAKEY2,#4DH
        MOV DATAKEY3,#41H
        MOV DATAKEY4,#4EH
        MOV DATAKEY5,#47H
        MOV DATAKEY6,#' '
        MOV DATAKEY7,#4BH
        MOV DATAKEY8,#45H
        MOV DATAKEY9,#54H
        MOV DATAKEY10,#55H
        MOV DATAKEY11,#54H
        MOV DATAKEY12,#' '

```

```

        MOV DATAKEY13,#0
        MOV DATAKEY14,#0
        MOV DATAKEY15,#0
        MOV DATAKEY16,#0
        MOV DATAKEY17,#0
        MOV DATAKEY18,#0
        MOV DATAKEY19,#0
        MOV DATAKEY20,#0
        MOV DATAKEY21,#0
        MOV DATAKEY22,#0
        MOV DATAKEY23,#0
        MOV DATAKEY24,#0
        MOV DATAKEY25,#0
        MOV LOKASI,#0
        MOV COUNTER,#0
        CLR  P2.7
        CLR  P2.7
        CALL INIT_LCD
        MOV LOKASI,#0

```

```

INISIALISASI:

```

```

;*****
;* INISIALISASI LCD *
;*****
DELAY_INIT_LCD:
        MOV R6,#20H

```

```

DLY_LCD_LP:

```

```
MOV R7,#0
DJNZ R7,$
DJNZ R6,DLY_LCD_LP
RET
```

INIT_LCD:

```
SETB RS
CLR E

MOV A,#DISPCLR
CALL CONTROLOUT
CALL DELAY_INIT_LCD
MOV A,#FUNCSET
CALL CONTROLOUT
MOV A,#DISPON
CALL CONTROLOUT
MOV A,#ENTRMOD
CALL CONTROLOUT
MOV DPTR,#NAMA
CALL PRINTSTRING1
MOV DPTR,#NIM
CALL PRINTSTRING2
CALL TUNDATAM
MOV DPTR,#JUDUL
CALL PRINTSTRING1
MOV DPTR,#JUDUL1
CALL PRINTSTRING2
CALL TUNDATAM
```

OKE1:

NAIK:

```
MOV DPTR,#DATA1
CALL PRINTSTRING1
MOV DPTR,#PAS3
CALL PRINTSTRING2
CALL TOMBOL
MOV A,#1
LCALL POSISI2
MOV A,DATAKEY0
LCALL DATAOUT
MOV A,#2
LCALL POSISI2
MOV A,DATAKEY1
LCALL DATAOUT
MOV A,#3
LCALL POSISI2
MOV A,DATAKEY2
LCALL DATAOUT
MOV A,#4
LCALL POSISI2
MOV A,DATAKEY3
LCALL DATAOUT
MOV A,#5
LCALL POSISI2
```

```

MOV A,DATAKEY4
LCALL DATAOUT
MOV A,#6
LCALL POSISI2
MOV A,DATAKEY5
LCALL DATAOUT
MOV A,#7
LCALL POSISI2
MOV A,DATAKEY6
LCALL DATAOUT
CALL TUNDATAM
CALL TOMBOL
MOV keybounc,#50
mov keyport,#0FFh
SETB KOLOM2
SETB KOLOM3
SETB KOLOM4
clr kolom2
ul1Q:  jb baris1,key1Q
      djnz keybounc,ul1Q
      JMP KAMAR1

```

```

TUNDA_ISD:
MOV R7,#100
L82A:  MOV R6,#100
L72A:  MOV R5,#100
      DJNZ R5,$
      DJNZ R6,L72A
      DJNZ R7,L82A
      RET

```

```

key1Q:  CALL TOMBOL
      MOV DPTR,#DATA3
      CALL PRINTSTRING1
      MOV DPTR,#PAS3
      CALL PRINTSTRING2
      MOV A,#1
      LCALL POSISI2
      MOV A,DATAKEY7
      LCALL DATAOUT
      MOV A,#2
      LCALL POSISI2
      MOV A,DATAKEY8
      LCALL DATAOUT
      MOV A,#3
      LCALL POSISI2
      MOV A,DATAKEY9
      LCALL DATAOUT
      MOV A,#4
      LCALL POSISI2
      MOV A,DATAKEY10
      LCALL DATAOUT
      MOV A,#5
      LCALL POSISI2

```

```

MOV A,DATAKEY11
LCALL DATAOUT
MOV A,#6
LCALL POSISI2
MOV A,DATAKEY12
LCALL DATAOUT
CALL TUNDATAM
CALL TOMBOL
SETB KOLOM1
SETB KOLOM2
SETB KOLOM4
clr kolom3
ul2Q:  jb baris1,key2Q
      djnz keybounc,ul2Q
      JMP KAMAR2
key2Q:

      SETB KOLOM2
      SETB KOLOM4
      SETB KOLOM1
      clr kolom3
ul3Q:  jb baris7,key3Q
      djnz keybounc,ul3Q
      JMP PASSWORD
key3Q:

      JMP NAIK

TOMBOL:
JNB P2.3,SIMPAN1
JNB P2.5,SIMPAN2
JNB P2.3,SIMPAN1
JNB P2.5,SIMPAN2
JNB P2.3,SIMPAN1
JNB P2.5,SIMPAN2
RET

SIMPAN1:
MOV P3,#0FCH
CALL TUNDATAM
MOV P3,#0FFH
RET

SIMPAN2:
MOV P3,#0FDH
CALL TUNDATAM
MOV P3,#0FFH
RET

KAMARI:
MOV P3,#0F1H
MOV DPTR,#DATA5
CALL PRINTSTRING1
MOV DPTR,#DATA6
CALL PRINTSTRING2
CALL TUNDATAM

```

```

MOV P3,#0FFH

MOV counter,#0
coun1:
MOV A,counter
INC A
MOV counter,A
CALL TUNDA_ISD
JNB P2.2,TIDAK1
JNB P2.6,TIDAK1
CJNE A,#0fh,coun1
CALL TUNDA_ISD

SETB P2.7
MOV P3,#0F3H
CALL TUNDATAM
MOV P3,#0FFH
CLR P2.7

MOV DPTR,#DATA14
CALL PRINTSTRING1
MOV DPTR,#DATA15
CALL PRINTSTRING2
MOV counter,#0
coun2:
MOV A,counter
INC A
MOV counter,A
MOV keybounc,#50
mov keyport,#0FFh
SETB KOLOM2
SETB KOLOM1
SETB KOLOM4
clr kolom3
ul1Qp0: jb baris7,key1Qp1
        djnz keybounc,ul1Qp0
        JMP titip1
key1Qp1: CALL TUNDATAM
        CJNE A,#0fh,coun2
        MOV counter,#0
        JMP NAIK
titip1:  MOV P3,#0F8H
        MOV DPTR,#DATA7
        CALL PRINTSTRING1
        MOV DPTR,#DATA13
        CALL PRINTSTRING2
        CALL TUNDATAM
        MOV P3,#0FFH
        JMP NAIK
TIDAK1: SETB P2.7
        MOV P3,#0F2H

```

```
CALL TUNDATAM
MOV P3,#0FFH
CLR P2.7
JMP NAIK
```

KAMAR2:

```
MOV P3,#0F4H
MOV DPTR,#DATA7
CALL PRINTSTRING1
MOV DPTR,#DATA8
CALL PRINTSTRING2
CALL TUNDATAM
MOV P3,#0FFH
```

```
MOV counter,#0
```

coun3:

```
MOV A,counter
INC A
MOV counter,A
CALL TUNDA_ISD
JNB P2.4,TIDAK2
JNB P2.6,TIDAK2
CJNE A,#0fh,coun3
CALL TUNDA_ISD
```

```
SETB P2.7
MOV P3,#0F6H
CALL TUNDATAM
MOV P3,#0FFH
CALL TUNDATAM
CLR P2.7
```

```
MOV DPTR,#DATA14
CALL PRINTSTRING1
MOV DPTR,#DATA15
CALL PRINTSTRING2
```

```
MOV counter,#0
```

coun4:

```
MOV A,counter
INC A
MOV counter,A
MOV keybounc,#50
MOV keyport,#0FFh
SETB KOLOM2
SETB KOLOM1
SETB KOLOM4
clr kolom3
```

ul1Qp2: jb baris7,key1Qp3
djnz keybounc,ul1Qp2
JMP titip2

key1Qp3:

```
CALL TUNDATAM
CJNE A,#0fh,coun4
```

```

        MOV counter,#0
        JMP  NAIK
titip2:
        MOV P3,#0FAH
        MOV DPTR,#DATA7
        CALL PRINTSTRING1
        MOV DPTR,#DATA13
        CALL PRINTSTRING2
        CALL TUNDATAM
        JMP  NAIK
TIDAK2:
        SETB P2.7
        MOV P3,#0F5H
        CALL TUNDATAM
        MOV P3,#0FFH
        CLR  P2.7
        JMP  NAIK

```

```

PASSWORD:
        MOV DPTR,#PRO4
        CALL PRINTSTRING1
        MOV DPTR,#PRO5
        CALL PRINTSTRING2
        CALL TUNDATAM

```

```

CENSEL:
        MOV DPTR,#PRO6
        CALL PRINTSTRING2
        CALL TUNDA_ISD

```

```

,*****
,

```

```

        MOV DATAKEY20,#0
        MOV DATAKEY21,#0
        MOV DATAKEY22,#0
        MOV DATAKEY23,#0
        MOV DATAKEY24,#0
        MOV COD1,#98H
        MOV COD2,#89H
        MOV LOKASI1,#0

```

```

KEY:

```

```

TIDAK_ADA:
        call  PASS
        mov  A,keydata
        cjne A,#0FFH,LAYARI
        jmp  TIDAK_ADA

```

```

PASS:
        mov  keybounc,#50
        mov  keyport,#0FFh

```

```

SETB KOLOM2
SETB KOLOM3
SETB KOLOM4
clr kolom1
ul1A:  jb baris1,key1A
      djnz keybounc,ul1A
      mov  keydata,#0
      RET
key1A:  jb baris2,key2A
      djnz keybounc,key1A
      mov  keydata,#4
      RET
key2A:  jb baris3,key3A
      djnz keybounc,key2A
      mov  keydata,#8
      RET
LAYAR1:
      AJMP LAYAR

key3A:  setb kolom1
      clr kolom2
      jb  baris1,key5A
      djnz keybounc,key3A
      mov  keydata,#1
      RET
key5A:  jb baris2,key6A
      djnz keybounc,key5A
      mov  keydata,#5
      RET
key6A:  jb baris3,key7B
      djnz keybounc,key6A
      mov  keydata,#9
      RET

key7B:  setb kolom2
      clr kolom3
      jb  baris1,key9A
      djnz keybounc,key7B
      mov  keydata,#2
      RET
key9A:  jb baris2,key10A
      djnz keybounc,key9A
      mov  keydata,#6
      RET

key10A: JB  baris7,key13A
      DJNZ keybounc,key10A
      mov  keydata,#0FH
      RET

key13A: setb kolom3
      clr kolom4
      jb  baris1,key14A

```

```

        MOV A,DATAKEY23
        JMP TAMPILQ
PIL4Q:  LJMP PIL0Q
        RET

PAS_SALAH:
        MOV DPTR,#PRO9
        CALL PRINTSTRING1
        MOV DPTR,#PRO10
        CALL PRINTSTRING2
        CALL TUNDATAM
        AJMPOKEI
LOOPKEY:
        MOV DPTR,#PRO7
        CALL PRINTSTRING1
        MOV DPTR,#PRO8
        CALL PRINTSTRING2
        CALL TUNDATAM

ENTER1:

        MOV A,DATAKEY20
        SWAP A
        ORL A,DATAKEY21
        MOV DATAKEY20,A
        MOV A,COD1
        XRL A,DATAKEY20
        MOV R0,A
        CLR A
        MOV A,DATAKEY22
        SWAP A
        ORL A,DATAKEY23
        MOV DATAKEY22,A
        MOV A,COD2
        XRL A,DATAKEY22
        ORI A,R0
        CJNE A,#00H,PAS_SALAH
QQ:     MOV DPTR,#PRO11
        LCALL PRINTSTRING1
        MOV DPTR,#PRO12
        LCALL PRINTSTRING2

        MOV keybounc,#50
        mov keyport,#0FFh
        SETB KOLOM1
        SETB KOLOM3
        SETB KOLOM4
        clr kolom2
ul1QA: jb baris1,key1QA

```

```

        djnz keybounc,ul1QA
        JMP KAMAR1A
key1QA:

        SETB KOLOM1
        SETB KOLOM2
        SETB KOLOM4
        clr kolom3
ul2QA:  jb baris1,key2QA
        djnz keybounc,ul2QA
        JMP KAMAR2A
key2QA:
        JMP QQ

```

TAMPILQ:

```

        ORL A,#30H
        LCALL DATAOUT
        CALLDelay1
        AJMPKEY
        RET

```

KAMAR1A:

```

        MOV DPTR,#SIAP
        CALLPRINTSTRING1
        MOV DPTR,#SIAP1
        CALLPRINTSTRING2
        CALLTUNDATAM
        MOV DPTR,#SIAP2
        CALLPRINTSTRING1
        MOV DPTR,#SIAP3
        CALLPRINTSTRING2
        MOV DATAKEY0,#20H
        MOV DATAKEY1,#20H
        MOV DATAKEY2,#20H
        MOV DATAKEY3,#20H
        MOV DATAKEY4,#20H
        MOV DATAKEY5,#20H
        MOV DATAKEY6,#20H
        MOV NO2D,#01H
        MOV LOKASI,#0
        JMP KEYLOOP

```

KAMAR2A:

```

        MOV DPTR,#SIAP
        CALLPRINTSTRING1
        MOV DPTR,#SIAP1

```

```

CALLPRINTSTRING2
CALLTUNDATAM
MOV DPTR,#SIAP4
CALLPRINTSTRING1
MOV DPTR,#SIAP3
CALLPRINTSTRING2
MOV DATAKEY7,#20H
MOV DATAKEY8,#20H
MOV DATAKEY9,#20H
MOV DATAKEY10,#20H
MOV DATAKEY11,#20H
MOV DATAKEY12,#20H
MOV NO2D,#02H
MOV LOKASI,#7

```

KEYLOOP:

```

ulangT:  call  Keypad4x7
        mov  A,keydata
        cjne A,#0FFH,tampil
        jmp  ulangT

```

Keypad4x7:

```

        mov  keybounc,#50
        mov  keyport,#0FFh
        SETB KOLOM2
        SETB KOLOM3
        SETB KOLOM4
        clr  kolom1
ulang1:  jb  baris1,key1
        djnz keybounc,ulang1
        mov  keydata,#41H
        RET
key1:    jb  baris2,key2
        djnz keybounc,key1
        mov  keydata,#45H
        RET
key2:    jb  baris3,key3
        djnz keybounc,key2
        mov  keydata,#49H
        RET
key3:    jb  baris4,keyz
        djnz keybounc,key3
        mov  keydata,#4DH
        RET
tampil:  ljmp  tampil2
keyz:    jb  baris5,keyzA
        djnz keybounc,keyz

```

```

mov keydata,#51H
RET
keyzA: jb baris6,keyzB
      djnz keybounc,keyzA
      mov keydata,#55H
      RET
keyzB: jb baris7,key4
      djnz keybounc,keyzB
      mov keydata,#59H
      RET

key4:  setb kolom1
      clr kolom2
      jb baris1,key5
      djnz keybounc,key4
      mov keydata,#42H
      RET
key5:  jb baris2,key6
      djnz keybounc,key5
      mov keydata,#46H
      RET
key6:  jb baris3,key7
      djnz keybounc,key6
      mov keydata,#4AH
      RET
key7:  jb baris4,key8a
      djnz keybounc,key7
      mov keydata,#4EH
      RET
key8a: jb baris5,key7A
      djnz keybounc,key8a
      mov keydata,#52H
      RET
key7A: jb baris6,key8aA
      djnz keybounc,key7A
      mov keydata,#56H
      RET
key8aA: jb baris7,key8
      djnz keybounc,key8aA
      mov keydata,#5AH
      RET

key8:  setb kolom2
      clr kolom3
      jb baris1,key9
      djnz keybounc,key8
      mov keydata,#43H
      RET
key9:  jb baris2,key10
      djnz keybounc,key9
      mov keydata,#47H
      RET
key10: jb baris3,key11
      djnz keybounc,key10

```

```

        mov keydata,#4BH
        RET
key11:  jb baris4,key12a
        djnz keybounc,key11
        mov keydata,#4FH
        RET
key12a: jb baris5,key11A
        djnz keybounc,key12a
        mov keydata,#53H
        RET
key11A: jb baris6,key12aA
        djnz keybounc,key11A
        mov keydata,#57H
        RET
key12aA: JB baris7,key13
        DJNZ keybounc,key12aA
        mov keydata,#0FEH
        RET

key13:  setb kolom3
        clr kolom4
        jb baris1,key14
        djnz keybounc,key13
        MOV keydata,#44H
        RET
key14:  jb baris2,key15
        djnz keybounc,key14
        mov keydata,#48H
        RET
key15:  jb baris3,key16
        djnz keybounc,key15
        mov keydata,#4CH
        RET

key16:  jb baris4,key17
        djnz keybounc,key16
        mov keydata,#50H
        RET

key17:  jb baris5,key16A
        djnz keybounc,key17
        mov keydata,#54H
        RET
key16A: jb baris6,key17A
        djnz keybounc,key16A
        mov keydata,#58H
        RET

key17A: jb baris7,key18B
        djnz keybounc,key17A
        RET
key18B: mov keydata,#0FFh

```

RET

LOOPKEYA:

LJMP LOOPKEY1A

TAMPIL2:

MOV A,KEYDATA

MOV B,A

CLR C

SUBBA,#0EFH

JNC LOOPKEYA

MOV A,LOKASI

INC A

MOV LOKASI,A

PIL0:

CJNE A,#1,PIL1

MOV DATAKEY0,B

MOV A,#1

LCALL POSISI2

MOV A,DATAKEY0

JMP TAMPILL

PIL1:

CJNE A,#2,PIL2

MOV DATAKEY1,B

MOV A,#2

LCALL POSISI2

MOV A,DATAKEY1

JMP TAMPILL

PIL2:

CJNE A,#3,PIL3

MOV DATAKEY2,B

MOV A,#3

LCALL POSISI2

MOV A,DATAKEY2

JMP TAMPILL

PIL3:

CJNE A,#4,PIL4

MOV DATAKEY3,B

MOV A,#4

LCALL POSISI2

MOV A,DATAKEY3

JMP TAMPILL

PIL4:

CJNE A,#5,PIL5

MOV DATAKEY4,B

MOV A,#5

LCALL POSISI2

MOV A,DATAKEY4

JMP TAMPILL

PIL5:

CJNE A,#6,PIL6

MOV DATAKEY5,B

MOV A,#6

LCALL POSISI2

```

MOV A,DATAKEY5
JMP TAMPILL
PIL6:
CJNE A,#7,PIL7
MOV DATAKEY6,B
MOV A,#7
LCALL POSISI2
MOV A,DATAKEY6
MOV LOKASI,#0
JMP TAMPILL
PIL7:
CJNE A,#8,PIL8
MOV DATAKEY7,B
MOV A,#10
LCALL POSISI2
MOV A,DATAKEY7
JMP TAMPILL
PIL8:
CJNE A,#9,PIL9
MOV DATAKEY8,B
MOV A,#11
LCALL POSISI2
MOV A,DATAKEY8
JMP TAMPILL
PIL9:
CJNE A,#10,PIL10
MOV DATAKEY9,B
MOV A,#12
LCALL POSISI2
MOV A,DATAKEY9
JMP TAMPILL
PIL10:
CJNE A,#11,PIL11
MOV DATAKEY10,B
MOV A,#13
LCALL POSISI2
MOV A,DATAKEY10
JMP TAMPILL
PIL11:
CJNE A,#12,PIL12
MOV DATAKEY11,B
MOV A,#14
LCALL POSISI2
MOV A,DATAKEY11
JMP TAMPILL
PIL12:
CJNE A,#12,PIL13
MOV DATAKEY12,B
MOV A,#15
LCALL POSISI2
MOV A,DATAKEY12
MOV LOKASI,#7
JMP TAMPILL
PIL13:

```

```

        LJMP PILO
        RET
TAMPILL:
        LCALL    DATAOUT
        CALLroll
        JMP KEYLOOP
        RET
roll:
        MOV R7,#50
LOOPAC: MOV R6,#50
LOOPAD: MOV R5,#100
        DJNZ R5,$
        DJNZ R6,LOOPAD
        DJNZ R7,LOOPAC
        RET
LOOPKEY1A:
        MOV A,NO2D
        CJNE A,#01H,BUKAN_KAMARI
        MOV DPTR,#SIAP5
        CALLPRINTSTRING1
        MOV DPTR,#SIAP10
        CALLPRINTSTRING2
        CALLTUNDATAM
        MOV DPTR,#SIAP5
        CALLPRINTSTRING1
        MOV DPTR,#SIAP6
        CALLPRINTSTRING2
        CALLTUNDATAM
        MOV DPTR,#SIAP7
        CALLPRINTSTRING2
        CALLTUNDA1
        MOV DPTR,#SIAP8
        CALLPRINTSTRING2
        CALLTUNDA1
        MOV DPTR,#SIAP9
        CALLPRINTSTRING2
        MOV P3,#0FEH
        CALLTUNDATAM
        MOV P3,#0PPH
        CALLTUNDATAM
        AJMPNAIK
BUKAN_KAMARI:
        MOV DPTR,#SIAP5
        CALLPRINTSTRING1
        MOV DPTR,#SIAP11
        CALLPRINTSTRING2
        CALLTUNDATAM
        MOV DPTR,#SIAP5
        CALLPRINTSTRING1
        MOV DPTR,#SIAP6
        CALLPRINTSTRING2
        CALLTUNDATAM

```

```

MOV DPTR,#SIAP7
CALL PRINTSTRING2
CALL TUNDA1
MOV DPTR,#SIAP8
CALL PRINTSTRING2
CALL TUNDA1
MOV DPTR,#SIAP9
CALL PRINTSTRING2
MOV P3,#0F0H
CALL TUNDATAM
MOV P3,#0FFH
CALL TUNDATAM
AJMP NAIK

```

```

;*****
;* KUMPULAN RUTIN PELAYANAN LCD *
;*****

```

```

POSISI2_1:
MOV A,#1
POSISI2:
ADD A,#11000000B
SJMP POSISI_SUB

POSISI1_1:
MOV A,#1
POSISI1:
ADD A,#10000000B
POSISI_SUB:
DEC A
LCALL CONTROLOUT
RET

PRINTSTRING2:
LCALL POSISI2_1
SJMP PRINTSTRING

PRINTSTRING1:
LCALL POSISI1_1

PRINTSTRING:
SJMP OUTSTRING
PRINTSTRINGLOOP:
LCALL DATAOUT
INC DPTR

OUTSTRING:
CLR A
MOVC A,@A+DPTR
JNZ PRINTSTRINGLOOP
RET

CONTROLOUT:

```

```

DATA2: DB 'KOMANG      ',0
DATA3: DB 'TKN 2 U/ KMR 2 ',0
DATA4: DB 'KETUT      ',0

DATA5: DB 'TUNGGU PROSES ',0
DATA6: DB 'PANGGILAN KMR 1 ',0

DATA7: DB 'TUNGGU PROSES ',0
DATA8: DB 'PANGGILAN KMR 2 ',0
DATA13: DB ' PEREKAMAN ',0

PRO4:  DB 'MASUKAN 4 DIGIT ',0
PRO5:  DB 'NOMER ID ANDA ',0
PRO6:  DB 'NO ID :      ',0
PRO7:  DB 'ANALISA NO ID ',0
PRO8:  DB ' WAIT A MINUTE ',0
PRO9:  DB '  MAAF      ',0
PRO10: DB ' NO ID SALAH ',0
PRO11: DB 'TEKAN TOMBOL 1: ',0
PRO12: DB 'KAMAR1 2:KAMAR2 ',0
DATA14: DB 'TEKAN ENTER U/ ',0
DATA15: DB 'TITIP PESANN ',0

```

TUNDATAM:

```

      MOV R7,#100
LOOP2: MOV R6,#100
LOOP1: MOV R5,#100
      DJNZ R5,$
      DJNZ R6,LOOP1
      DJNZ R7,LOOP2

```

TUNDA1:

```

      MOV R7,#100
LOOP4: MOV R6,#100
LOOP3: MOV R5,#100
      DJNZ R5,$
      DJNZ R6,LOOP3
      DJNZ R7,LOOP4
      RET

```

delay1:

```

      MOV R3,#50

```

del111:

```

      MOV R1,#50

```

del11:

```

      MOV R2,#50
      DJNZ R2,$
      DJNZ R1,del11
      DJNZ R3,del111

```

```

      ret
      END

```

1.2 Performance Outline

Table 1.1. lists the performance outline of this MCU.

Table 1.1 Performance outline

Item	Performance	
CPU	Number of basic instructions	89 instructions
	Shortest instruction execution time	50 ns ($f(XIN) = 20$ MHz, $V_{CC} = 3.0$ to 5.5 V) 100 ns ($f(XIN) = 10$ MHz, $V_{CC} = 2.7$ to 5.5 V)
	Operating mode	Single-chip
	Address space	1M bytes
	Memory capacity	See Table 1.2.
Peripheral function	Interrupt	Internal: 11 factors, External: 5 factors, Software: 4 factors, Priority level: 7 levels
	Watchdog timer	15 bits x 1 (with prescaler) Reset start function selectable
	Timer	Timer X: 8 bits x 1 channel, Timer Y: 8 bits x 1 channel, Timer Z: 8 bits x 1 channel (Each timer equipped with 8-bit prescaler) Timer C: 16 bits x 1 channel Circuits of input capture and output compare.
	Serial interface	•1 channel Clock synchronous, UART •1 channel UART
	A/D converter	10-bit A/D converter: 1 circuit, 12 channels
	Clock generation circuit	2 circuits •Main clock generation circuit (Equipped with a built-in feedback resistor) •On-chip oscillator (high-speed, low-speed) On high-speed on-chip oscillator the frequency adjustment function is usable.
	Oscillation stop detection function	Stop detection of main clock oscillation
	Voltage detection circuit	Included
	Power on reset circuit	Included
	Port	Input/Output: 22 (including LED drive port), Input: 2 (LED drive I/O port: 8)
Electrical characteristics	Power supply voltage	$V_{CC} = 3.0$ to 5.5V ($f(XIN) = 20$ MHz) $V_{CC} = 2.7$ to 5.5V ($f(XIN) = 10$ MHz)
	Power consumption	Typ.9 mA ($V_{CC} = 5.0$ V, ($f(XIN) = 20$ MHz, High-speed mode) Typ.5 mA ($V_{CC} = 3.0$ V, ($f(XIN) = 10$ MHz, High-speed mode) Typ.35 μ A ($V_{CC} = 3.0$ V, Wait mode, Peripheral clock stops) Typ.0.7 μ A ($V_{CC} = 3.0$ V, Stop mode)
Flash memory	Program/erase voltage	$V_{CC} = 2.7$ to 5.5 V
	Number of program/erase	10,000 times (Data area) 1,000 times (Program area)
Operating ambient temperature	-20 to 85°C -40 to 85°C (D-version)	
Package	32-pin plastic mold LQFP	

1.3 Block Diagram

Figure 1.1 shows this MCU block diagram.

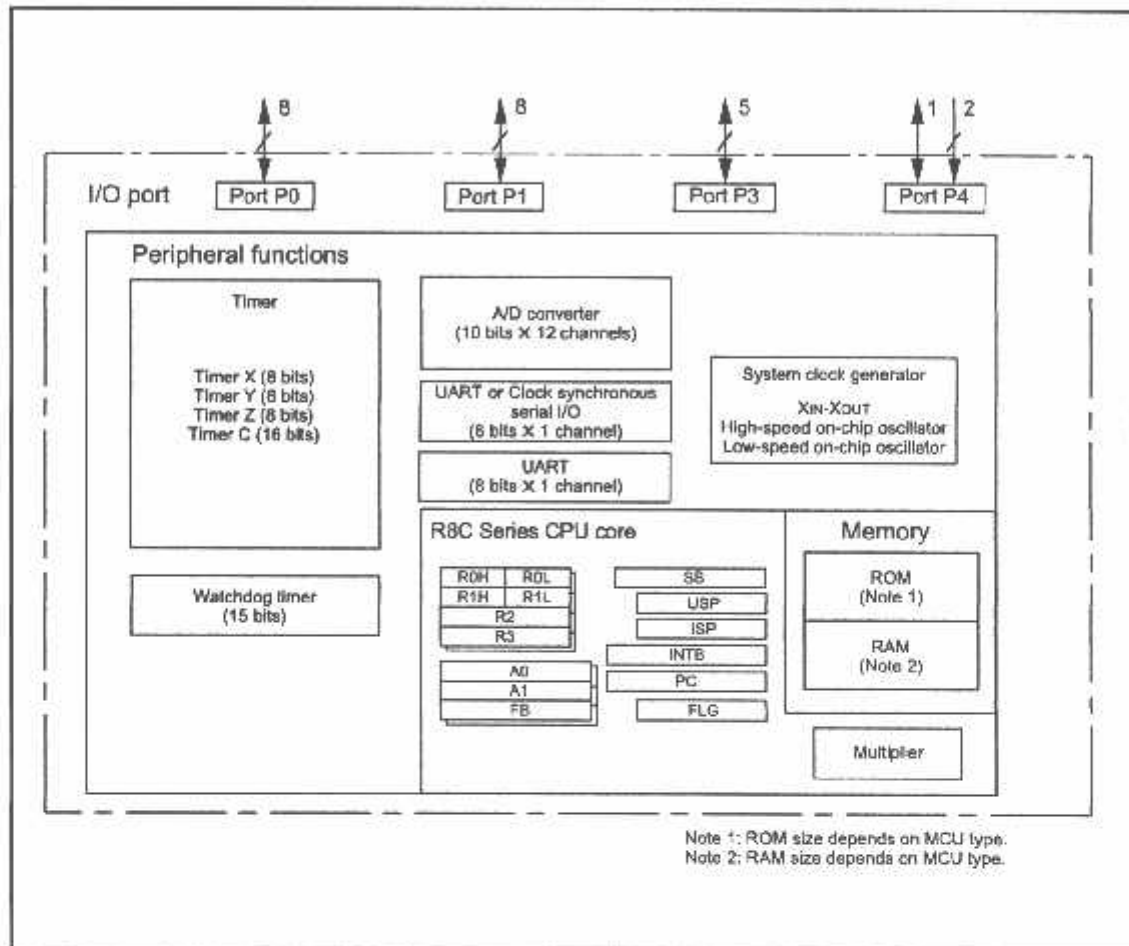


Figure 1.1 Block Diagram

1.4 Product Information

Table 1.2 lists the products.

Table 1.2 Product List

As of April 2005

Type No.	ROM capacity		RAM capacity	Package type	Remarks
	Program area	Data area			
R5F21132FP	8K bytes	2K bytes x 2	512 bytes	PLQP0032GB-A	Flash memory version
R5F21133FP	12K bytes	2K bytes x 2	768 bytes	PLQP0032GB-A	
R5F21134FP	16K bytes	2K bytes x 2	1K bytes	PLQP0032GB-A	
R5F21132DFP	8K bytes	2K bytes x 2	512 bytes	PLQP0032GB-A	D version
R5F21133DFP	12K bytes	2K bytes x 2	768 bytes	PLQP0032GB-A	
R5F21134DFP	16K bytes	2K bytes x 2	1K bytes	PLQP0032GB-A	

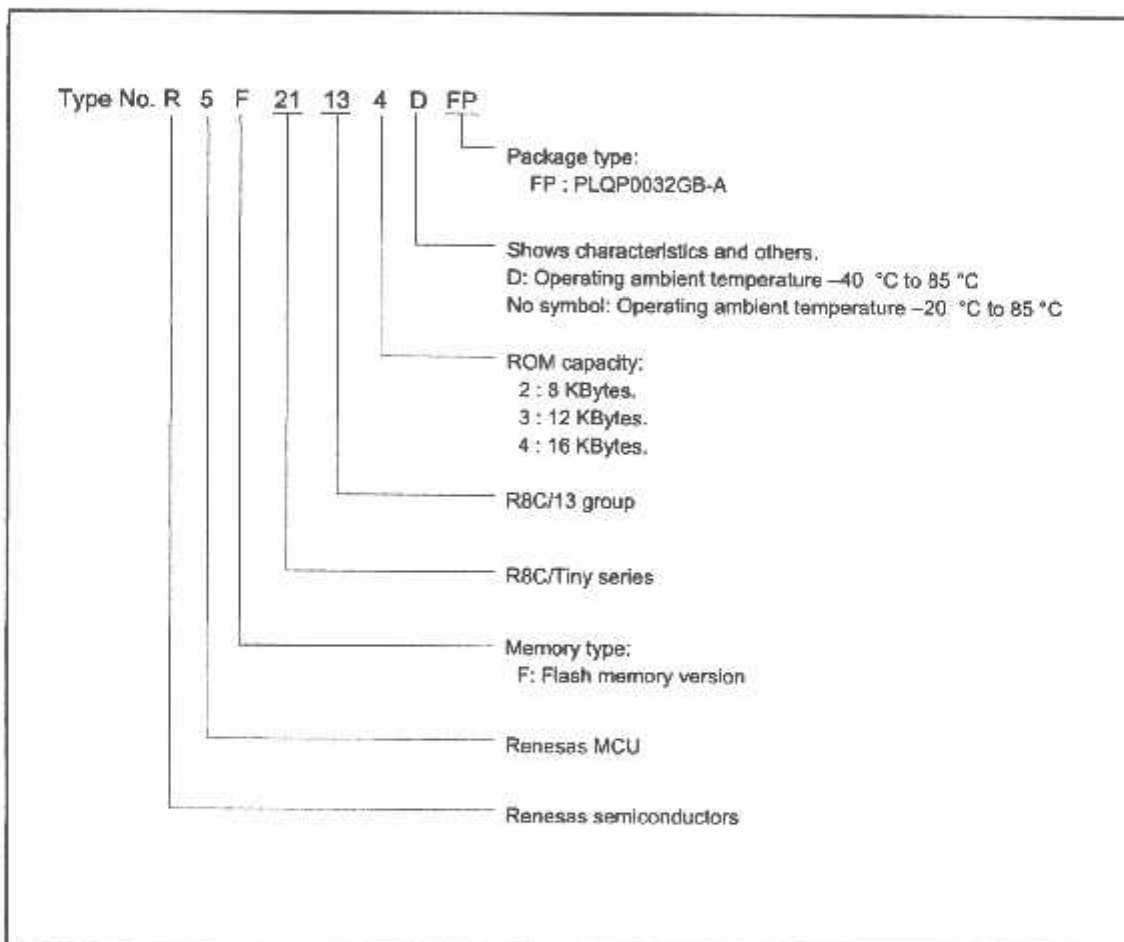


Figure 1.2 Type No., Memory Size, and Package

1.5 Pin Assignments

Figure 1.3 shows the pin configuration (top view).

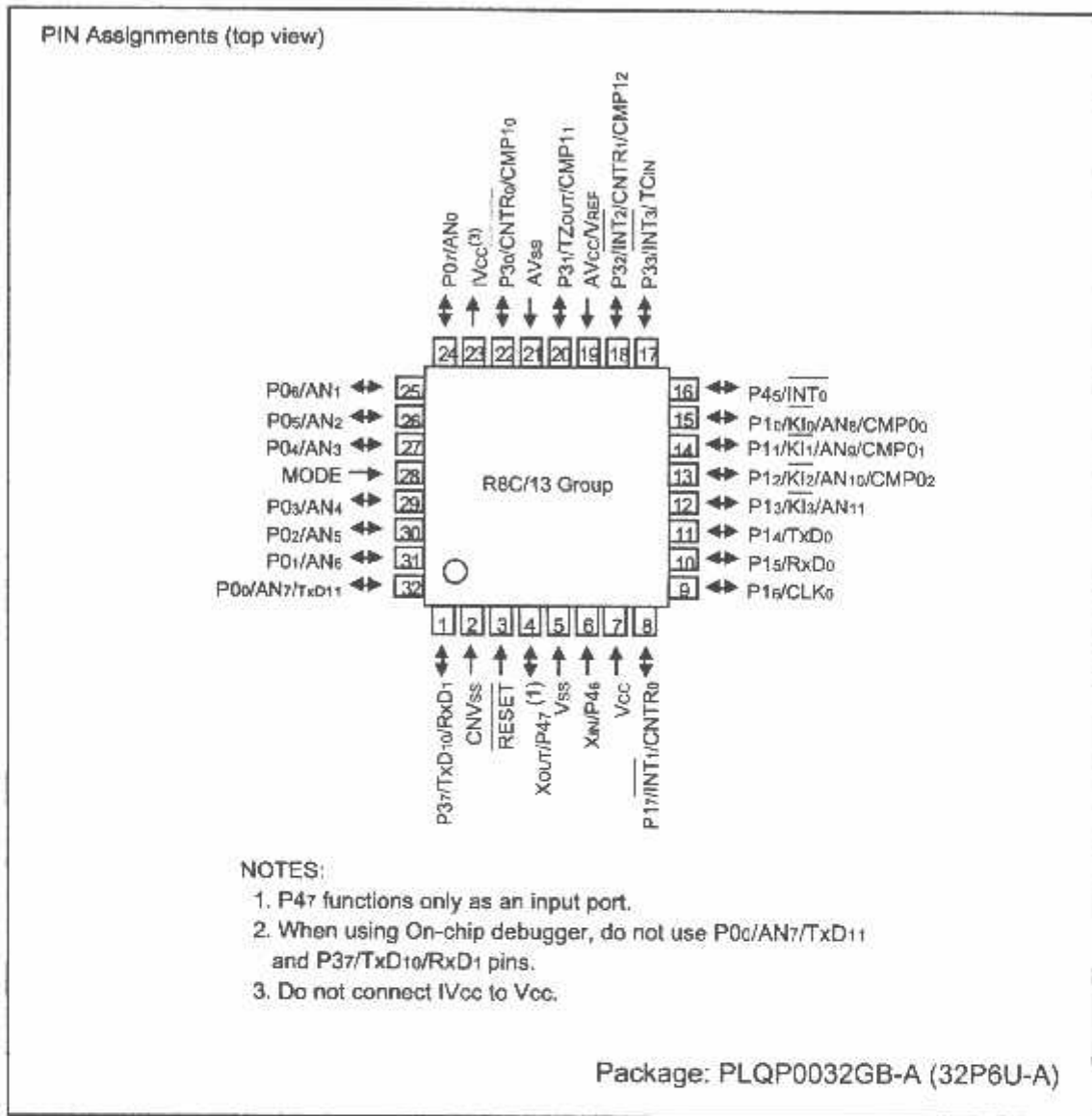


Figure 1.3 Pin Assignments (Top View)

1.6 Pin Description

Table 1.3 shows the pin description

Table 1.3 Pin description

Signal name	Pin name	I/O type	Function
Power supply input	Vcc, Vss	I	Apply 2.7 V to 5.5 V to the Vcc pin. Apply 0 V to the Vss pin.
IVcc	IVcc	O	This pin is to stabilize internal power supply. Connect this pin to Vss via a capacitor (0.1 μ F). Do not connect to Vcc.
Analog power supply input	AVcc, AVss	I	These are power supply input pins for A/D converter. Connect the AVcc pin to Vcc. Connect the AVss pin to Vss. Connect a capacitor between pins AVcc and AVss.
Reset input	RESET	I	"L" on this input resets the MCU.
CNVss	CNVss	I	Connect this pin to Vss via a resistor ⁽¹⁾
MODE	MODE	I	Connect this pin to Vcc via a resistor
Main clock input	XIN	I	These pins are provided for the main clock generating circuit I/O. Connect a ceramic resonator or a crystal oscillator between the XIN and XOUT pins. To use an externally derived clock, input it to the XIN pin and leave the XOUT pin open.
Main clock output	XOUT	O	
INT interrupt input	INT0 to INT3	I	These are INT interrupt input pins.
Key input interrupt input	KI0 to KI3	I	These are key input interrupt pins.
Timer X	CNTR0	I/O	This is the timer X I/O pin.
	CNTR0	O	This is the timer X output pin.
Timer Y	CNTR1	I/O	This is the timer Y I/O pin.
Timer Z	TZOUT	O	This is the timer Z output pin.
Timer C	TCIN	I	This is the timer C input pin.
	CMP00 to CMP03, CMP10 to CMP13	O	These are the timer C output pins.
Serial interface	CLK0	I/O	This is a transfer clock I/O pin.
	RxD0, RxD1	I	These are serial data input pins.
	TxD0, TxD10, TxD11	O	These are serial data output pins.
Reference voltage input	VREF	I	This is a reference voltage input pin for A/D converter. Connect the VREF pin to Vcc.
A/D converter	AN0 to AN11	I	These are analog input pins for A/D converter.
I/O port	P00 to P07, P10 to P17, P30 to P33, P37, P45	I/O	These are 8-bit CMOS I/O ports. Each port has an I/O select direction register, allowing each pin in that port to be directed for input or output individually. Any port set to input can select whether to use a pull-up resistor or not by program. P10 to P17 also function as LED drive ports.
Input port	P46, P47	I	These are input only pins.

2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers. The CPU has 13 registers. Of these, R0, R1, R2, R3, A0, A1 and FB comprise a register bank. There are two register banks.

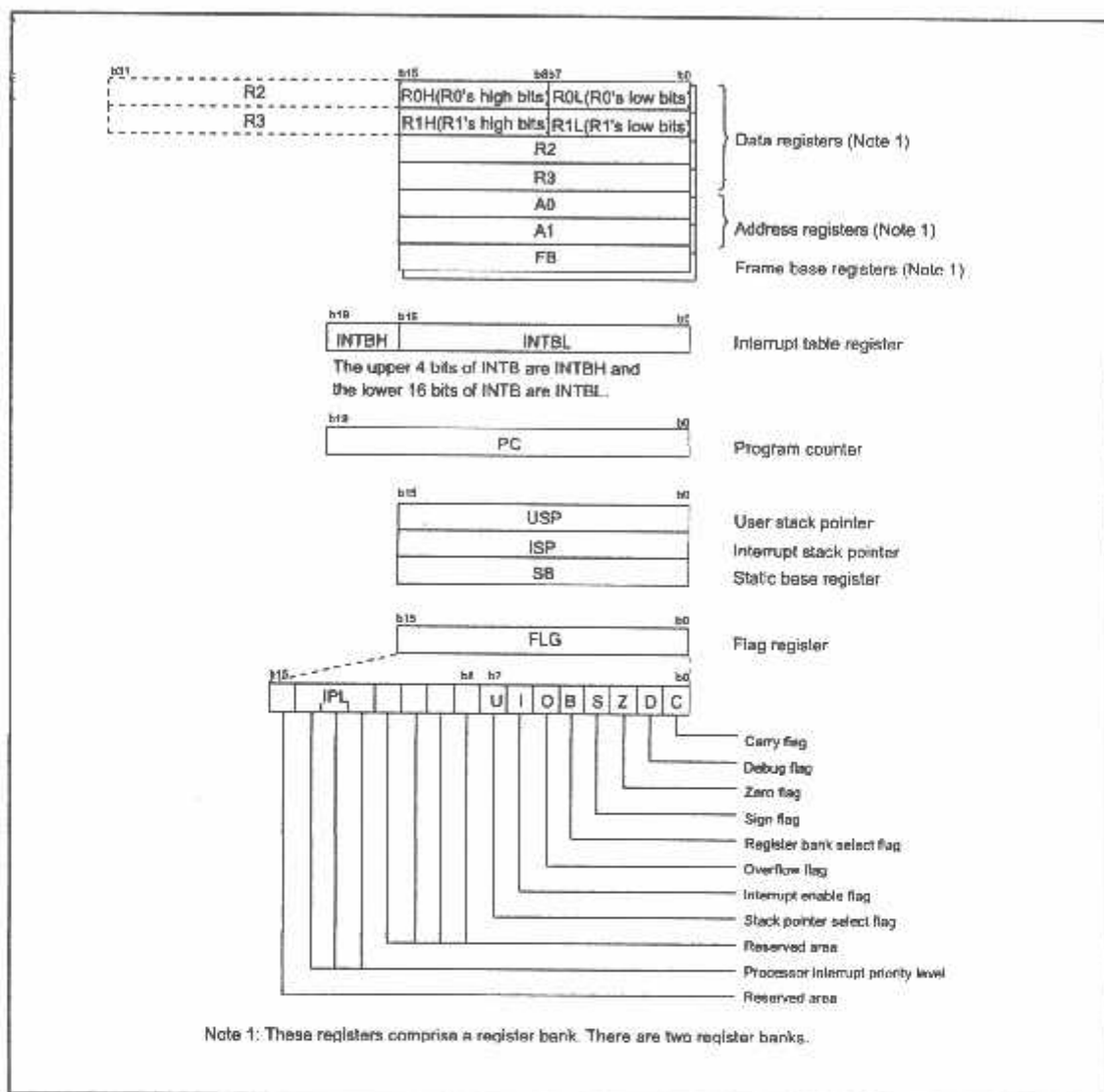


Figure 2.1 Central Processing Unit Register

2.1 Data Registers (R0, R1, R2 and R3)

The R0 register consists of 16 bits, and is used mainly for transfers and arithmetic/logic operations. R1 to R3 are the same as R0.

The R0 register can be separated between high (R0H) and low (R0L) for use as two 8-bit data registers. R1H and R1L are the same as R0H and R0L. Conversely, R2 and R0 can be combined for use as a 32-bit data register (R2R0). R3R1 is the same as R2R0.

2.2 Address Registers (A0 and A1)

The register A0 consists of 16 bits, and is used for address register indirect addressing and address register relative addressing. They also are used for transfers and logic/logic operations. A1 is the same as A0. In some instructions, registers A1 and A0 can be combined for use as a 32-bit address register (A1A0).

2.3 Frame Base Register (FB)

FB is configured with 16 bits, and is used for FB relative addressing.

2.4 Interrupt Table Register (INTB)

INTB is configured with 20 bits, indicating the start address of an interrupt vector table.

2.5 Program Counter (PC)

PC is configured with 20 bits, indicating the address of an instruction to be executed.

2.6 User Stack Pointer (USP) and Interrupt Stack Pointer (ISP)

Stack pointer (SP) comes in two types: USP and ISP, each configured with 16 bits.

Your desired type of stack pointer (USP or ISP) can be selected by the U flag of FLG.

2.7 Static Base Register (SB)

SB is configured with 16 bits, and is used for SB relative addressing.

2.8 Flag Register (FLG)

FLG consists of 11 bits, indicating the CPU status.

2.8.1 Carry Flag (C Flag)

This flag retains a carry, borrow, or shift-out bit that has occurred in the arithmetic/logic unit.

2.8.2 Debug Flag (D Flag)

The D flag is used exclusively for debugging purpose. During normal use, it must be set to "0".

2.8.3 Zero Flag (Z Flag)

This flag is set to "1" when an arithmetic operation resulted in 0; otherwise, it is "0".

2.8.4 Sign Flag (S Flag)

This flag is set to "1" when an arithmetic operation resulted in a negative value; otherwise, it is "0".

2.8.5 Register Bank Select Flag (B Flag)

Register bank 0 is selected when this flag is "0"; register bank 1 is selected when this flag is "1".

2.8.6 Overflow Flag (O Flag)

This flag is set to "1" when the operation resulted in an overflow; otherwise, it is "0".

2.8.7 Interrupt Enable Flag (I Flag)

This flag enables a maskable interrupt.

Maskable interrupts are disabled when the I flag is "0", and are enabled when the I flag is "1". The I flag is cleared to "0" when the interrupt request is accepted.

2.8.8 Stack Pointer Select Flag (U Flag)

ISP is selected when the U flag is "0"; USP is selected when the U flag is "1".

The U flag is cleared to "0" when a hardware interrupt request is accepted or an INT instruction for software interrupt Nos. 0 to 31 is executed.

2.8.9 Processor Interrupt Priority Level (IPL)

IPL is configured with three bits, for specification of up to eight processor interrupt priority levels from level 0 to level 7.

If a requested interrupt has priority greater than IPL, the interrupt is enabled.

2.8.10 Reserved Area

When write to this bit, write "0". When read, its content is indeterminate.

3. Memory

Figure 3.1 is a memory map of this MCU. The address space extends the 1M bytes from address 00000_{16} to $FFFFFF_{16}$.

The internal ROM (program area) is allocated in a lower address direction beginning with address $0FFFF_{16}$. For example, a 16-Kbyte internal ROM is allocated to the addresses from $0C000_{16}$ to $0FFFF_{16}$.

The fixed interrupt vector table is allocated to the addresses from $0FFDC_{16}$ to $0FFFF_{16}$. Therefore, store the start address of each interrupt routine here.

The internal ROM (data area) is allocated to the addresses from 02000_{16} to $02FFF_{16}$.

The internal RAM is allocated in an upper address direction beginning with address 00400_{16} . For example, a 1-Kbyte internal RAM is allocated to the addresses from 00400_{16} to $007FF_{16}$. In addition to storing data, the internal RAM also stores the stack used when calling subroutines and when interrupts are generated. Special function registers (SFR) are allocated to the addresses from 00000_{16} to $002FF_{16}$. Peripheral function control registers are located here. Of the SFR, any space which has no functions allocated is reserved for future use and cannot be used by users.

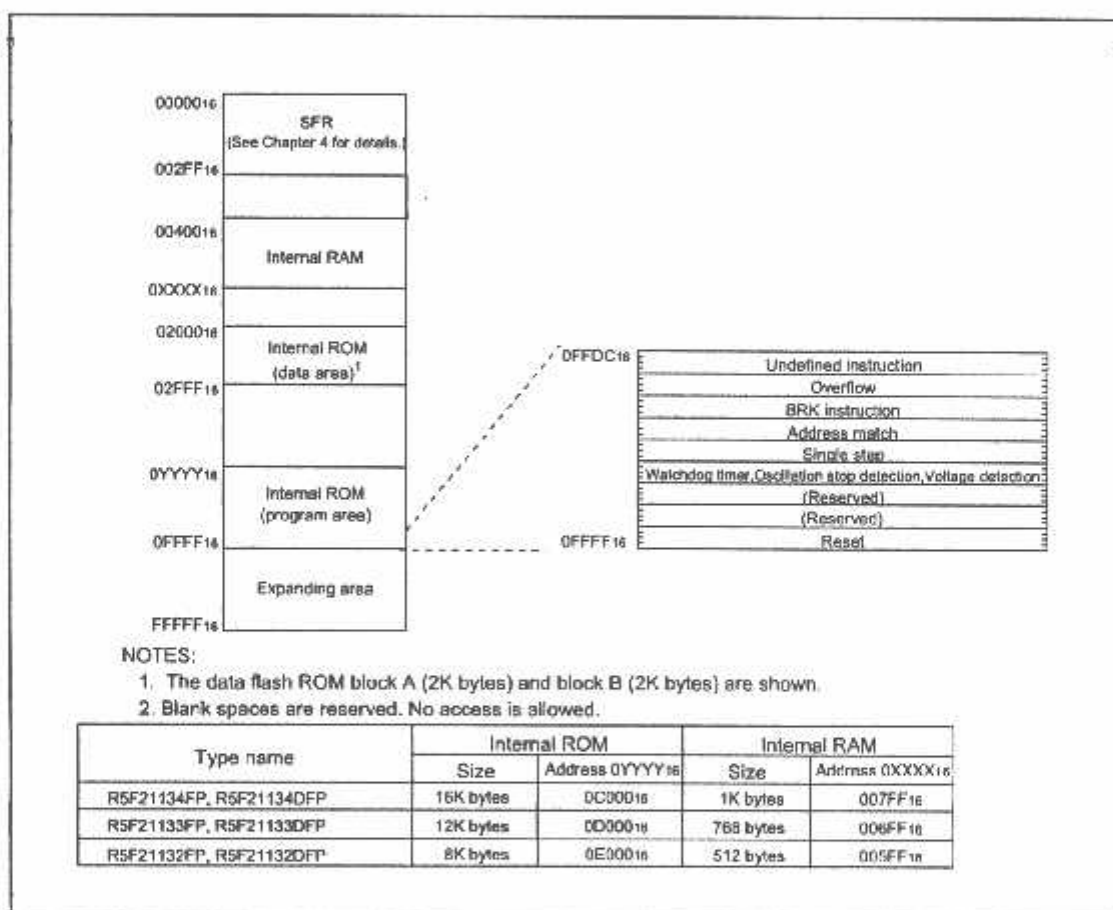


Figure 3.1 Memory Map

4. Special Function Register (SFR)

SFR(Special Function Register) is the control register of peripheral functions. Tables 4.1 to 4.4 list the SFR information

Table 4.1 SFR Information(1)(1)

Address	Register	Symbol	After reset
0000H			
0001H			
0002H			
0003H			
0004H	Processor mode register 0 ¹	PM0	0016
0005H	Processor mode register 1	PM1	0016
0006H	System clock control register 0	CM0	01:010002
0007H	System clock control register 1	CM1	00:000002
0008H	High-speed on-chip oscillator control register 0	HR0	0016
0009H	Address match interrupt enable register	AIER	XXXXXX002
000AH	Protect register	PRCR	00XXXX002
000BH	High-speed on-chip oscillator control register 1	HR1	4016
000CH	Oscillation stop detection register	OCD	00001002
000DH	Watchdog timer reset register	WDTR	XX16
000EH	Watchdog timer start register	WDTS	XX16
000FH	Watchdog timer control register	WDC	000111112
0010H	Address match interrupt register 0	RMAD0	0016
0011H			0016
0012H			X016
0013H			
0014H	Address match interrupt register 1	RMAD1	0016
0015H			0016
0016H			X016
0017H			
0018H			
0019H	Voltage detection register 1 ²	VCR1	00001002
001AH	Voltage detection register 2 ²	VCR2	0016 ³ 10000002 ⁴
001BH			
001CH			
001DH			
001EH	INT0 input filter select register	INT0F	XXXXX0002
001FH	Voltage detection interrupt register ²	D4INT	0016 ³ 01000012 ⁴
0020H			
0021H			
0022H			
0023H			
0024H			
0025H			
0026H			
0027H			
0028H			
0029H			
002AH			
002BH			
002CH			
002DH			
002EH			
002FH			
0030H			
0031H			
0032H			
0033H			
0034H			
0035H			
0036H			
0037H			
0038H			
0039H			
003AH			
003BH			
003CH			
003DH			
003EH			
003FH			

X : Undefined

NOTES:

- Blank columns are all reserved space. No access is allowed.
- Software reset or the watchdog timer reset does not affect this register.
- Owing to Reset input.
- In the case of RESET pin = H retaining.

5. Electrical Characteristics

Table 5.1 Absolute Maximum Ratings

Symbol	Parameter	Condition	Rated value	Unit
V _{CC}	Supply voltage	V _{CC} =AV _{CC}	-0.3 to 6.5	V
AV _{CC}	Analog supply voltage	V _{CC} =AV _{CC}	-0.3 to 6.5	V
V _I	Input voltage		-0.3 to V _{CC} +0.3	V
V _O	Output voltage		-0.3 to V _{CC} +0.3	V
P _d	Power dissipation	T _{opr} =25 °C	300	mW
T _{opr}	Operating ambient temperature		-20 to 85 / -40 to 85 (D version)	°C
T _{stg}	Storage temperature		-65 to 150	°C

Table 5.2 Recommended Operating Conditions

Symbol	Parameter	Conditions	Standard			Unit	
			Min.	Typ.	Max.		
V _{CC}	Supply voltage		2.7		5.5	V	
AV _{CC}	Analog supply voltage			V _{CC} ²		V	
V _{SS}	Supply voltage			0		V	
AV _{SS}	Analog supply voltage			0		V	
V _{IH}	"H" input voltage		0.8V _{CC}		V _{CC}	V	
V _{IL}	"L" input voltage		0		0.2V _{CC}	V	
I _{OH} (sum)	"H" peak all output currents	Sum of all pins' IOH (peak)			-60.0	mA	
I _{OH} (peak)	"H" peak output current				-10.0	mA	
I _{OH} (avg)	"H" average output current				-5.0	mA	
I _{OL} (sum)	"L" peak all output currents	Sum of all pins' IOL (peak)			60	mA	
I _{OL} (peak)	"L" peak output current	Except P1 ₀ to P1 ₇			10	mA	
		P1 ₀ to P1 ₇	Drive ability HIGH		30	mA	
I _{OL} (avg)	"L" average output current	Except P1 ₀ to P1 ₇	Drive ability LOW		10	mA	
			Drive ability HIGH		5	mA	
f (XIN)	Main clock input oscillation frequency		Drive ability LOW		5	MHz	
			3.0V ≤ V _{CC} ≤ 5.5V	0		20	MHz
			2.7V ≤ V _{CC} < 3.0V	0		10	MHz

Note

- 1: Referenced to V_{CC} = AV_{CC} = 2.7 to 5.5V at T_{opr} = -20 to 85 °C / -40 to 85 °C unless otherwise specified.
- 2: The mean output current is the mean value within 100ms.
- 3: Set V_{CC}=AV_{CC}

Table 5.3 A/D Conversion Characteristics

Symbol	Parameter		Measuring condition	Standard			Unit	
				Min.	Typ.	Max.		
-	Resolution		$V_{REF} = V_{CC}$			10	Bit	
-	Absolute accuracy	10 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{REF} = V_{CC} = 5.0 \text{ V}$			± 3	LSB	
		8 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{REF} = V_{CC} = 5.0 \text{ V}$			± 2	LSB	
		10 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{REF} = V_{CC} = 3.3 \text{ V}^3$			± 5	LSB	
		8 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{REF} = V_{CC} = 3.3 \text{ V}^3$			± 2	LSB	
R_{LADDER}	Ladder resistance		$V_{REF} = V_{CC}$	10		40	k Ω	
t_{CONV}	Conversion time		10 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{REF} = V_{CC} = 5.0 \text{ V}$	3.3		μs	
			8 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{REF} = V_{CC} = 5.0 \text{ V}$	2.8		μs	
V_{REF}	Reference voltage				V_{CC}^4		V	
V_{IA}	Analog input voltage			0		V_{REF}	V	
-	A/D operation clock frequency ²		Without sample & hold		0.25		10	MHz
			With sample & hold		1.0		10	MHz

Note

- 1: Referenced to $V_{CC} = AV_{CC} = 2.7$ to 5.5 V at $T_{opr} = -20$ to $85 \text{ }^\circ\text{C}$ / -40 to $85 \text{ }^\circ\text{C}$ unless otherwise specified.
- 2: When f_{AD} is 10 MHz more, divide the f_{AD} and make A/D operation clock frequency (ϕ_{AD}) lower than 10 MHz.
- 3: When the AV_{CC} is less than 4.2V, divide the f_{AD} and make A/D operation clock frequency (ϕ_{AD}) lower than $f_{AD}/2$.
- 4: Set $V_{CC} = V_{REF}$

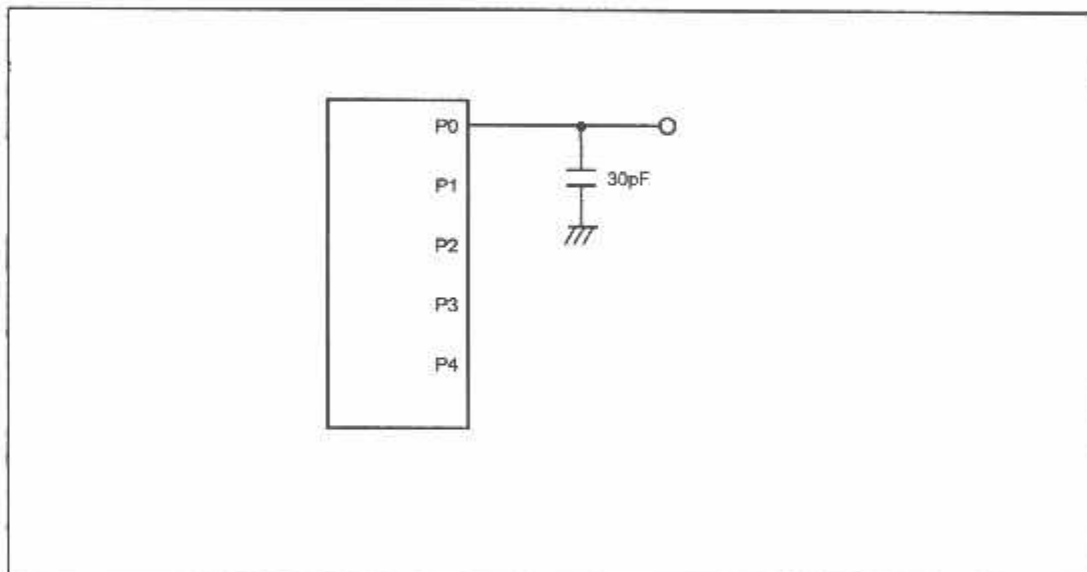


Figure 5.1 Port P0 to P4 measurement circuit

Table 5.4 Flash Memory (Program area) Electrical Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
—	Program/Erase cycle ²		1000 ³	—	—	cycle
—	Byte program time	V _{CC} = 5.0 V at Topr = 25 °C	—	50	—	μs
—	Block erase time	V _{CC} = 5.0 V at Topr = 25 °C	—	0.4	—	s
t _d (SR-ES)	Time delay from Suspend Request until Erase Suspend		—	—	8	ms
—	Erase Suspend Request Interval		10	—	—	ms
—	Program, Erase Voltage		2.7	—	5.5	V
—	Read Voltage		2.7	—	5.5	V
—	Program, Erase Temperature		0	—	60	°C
—	Data-retention duration	Topr = 55 °C	20	—	—	year

Table 5.5 Flash Memory (Data area Block A, Block B) Electrical Characteristics⁴

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
—	Program/Erase endurance ²		10000 ³	—	—	times
—	Byte program time(program/erase endurance ≤1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	50	400	μs
—	Byte program time(program/erase endurance >1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	65	—	μs
—	Block erase time(program/erase endurance ≤1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	0.2	9	s
—	Block erase time(program/erase endurance >1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	0.3	—	s
t _d (SR-ES)	Time delay from Suspend Request until Erase Suspend		—	—	8	ms
—	Erase Suspend Request Interval		10	—	—	ms
—	Program, Erase Voltage		2.7	—	5.5	V
—	Read Voltage		2.7	—	5.5	V
—	Program/Erase Temperature		-20(-40) ⁸	—	85	°C
—	Data-retention duration	Topr = 55 °C	20	—	—	year

Note

1: Referenced to V_{CC}=AV_{CC}=2.7 to 5.5V at Topr = 0°C to 60°C unless otherwise specified.

2: Definition of Program/Erase

The cycle of Program/Erase shows a cycle for each block.

If the program/erase number is "n" (n = 1000, 10000), "n" times erase can be performed for each block.

For example, if performing one-byte write to the distinct addresses on Block A of 2K-byte block 2048 times and then erasing that block, the number of Program/Erase cycles is one time.

However, performing multiple writes to the same address before an erase operation is prohibited (overwriting prohibited).

3: Maximum numbers of Program/Erase cycles for which all electrical characteristics is guaranteed.

4: Table 16.5 applies for Block A or B when the Program/Erase cycles are more than 1000. The byte program time up to 1000 cycles are the same as that of the program area (see Table 5.4).

5: To reduce the number of Program/Erase cycles, a block erase should ideally be performed after writing in series as many distinct addresses (only one time each) as possible. If programming a set of 16 bytes, write up to 128 sets and then erase them one time. This will result in ideally reducing the number of Program/Erase cycles. Additionally, averaging the number of Program/Erase cycles for Block A and B will be more effective. It is important to track the total number of block erases and restrict the number.

6: If error occurs during block erase, attempt to execute the clear status register command, then the block erase command at least three times until the erase error disappears.

7: Customers desiring Program/Erase failure rate information should contact their Renesas technical support representative.

8: -40 °C for D version.

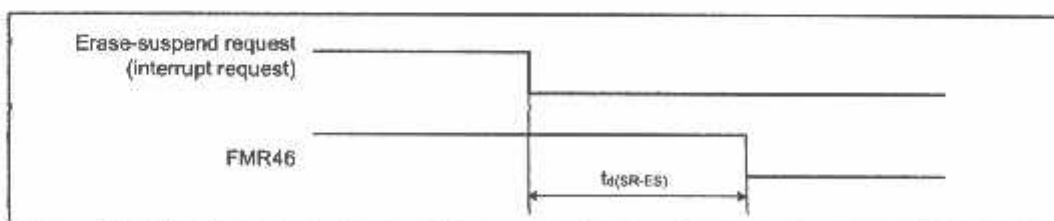
**Figure 5.2 Time delay from Suspend Request until Erase Suspend**

Table 5.6 Voltage Detection Circuit Electrical Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
Vdet	Voltage detection level		3.3	3.8	4.3	V
	Voltage detection interrupt request generating time ²			10		μs
	Voltage detection circuit self consumption current	VC27=1, VCC=5.0V		800		nA
t(E-A)	Waiting time until voltage detection circuit operation starts ³				20	μs
V _{comb}	Microcomputer operation voltage minimum value		2.7			V

NOTES:

- The measuring condition is V_{CC}=AV_{CC}=2.7V to 5.5V and T_{opr}=-40°C to 85°C.
- This shows the time until the voltage detection interrupt request is generated since the voltage passes V_{det}.
- This shows the required time until the voltage detection circuit operates when setting to "1" again after setting the VC27 bit in the VCR2 register to "0".

Table 5.7 Reset Circuit Electrical Characteristics (When Using Hardware Reset 2^{1, 3})

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
Vpor2	Power-on reset valid voltage	-20°C ≤ T _{opr} < 85°C			Vdet	V
t _w (Vpor2-Vdet)	Supply voltage rising time when power-on reset is canceled ²	-20°C ≤ T _{opr} < 85°C, t _w (por2) ≥ 10s ⁴			100	ms

NOTES:

- The voltage detection circuit which is embedded in a microcomputer is a factor to generate the hardware reset 2. Refer to 5.1.2 Hardware Reset 2.
- This condition is not applicable when using V_{CC} ≥ 1.0V.
- When turning power on after the external power has been held below the valid voltage for greater than 10 seconds, refer to Table 16.8 Reset Circuit Electrical Characteristics (When Not Using Hardware Reset 2).
- t_w(por2) is time to hold the external power below effective voltage (Vpor2).

Table 5.8 Reset Circuit Electrical Characteristics (When Not Using Hardware Reset 2)

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
Vpor1	Power-on reset valid voltage	-20°C ≤ T _{opr} < 85°C			0.1	V
t _w (Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	0°C ≤ T _{opr} ≤ 85°C, t _w (por1) ≥ 10s ²			100	ms
t _w (Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	-20°C ≤ T _{opr} < 0°C, t _w (por1) ≥ 30s ²			100	ms
t _w (Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	-20°C ≤ T _{opr} < 0°C, t _w (por1) ≥ 10s ²			1	ms
t _w (Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	0°C ≤ T _{opr} ≤ 85°C, t _w (por1) ≥ 1s ²			0.5	ms

NOTES:

- When not using hardware reset 2, use with V_{CC} ≥ 2.7V.
- t_w(por1) is time to hold the external power below effective voltage (Vpor1).

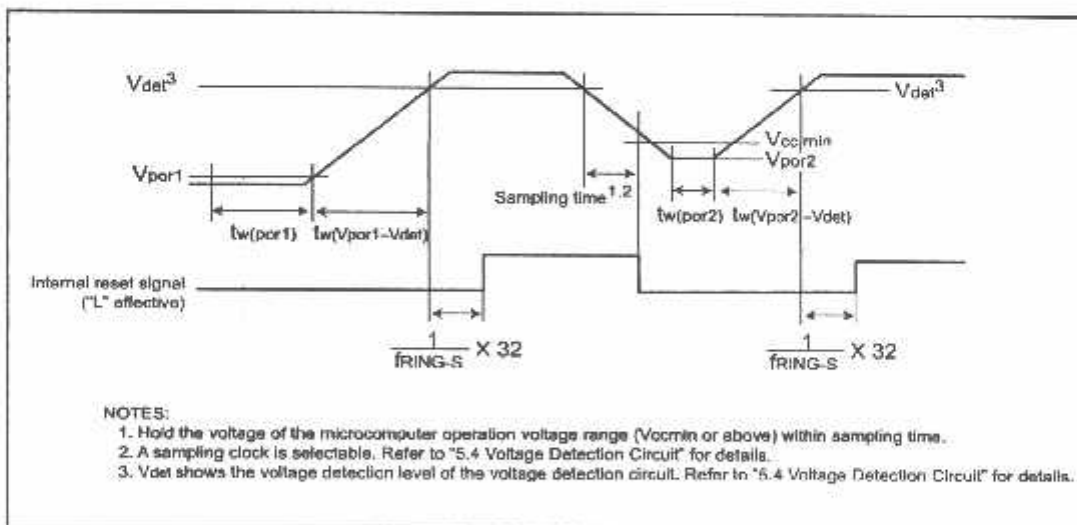


Figure 5.3 Reset Circuit Electrical Characteristics

Table 5.9 High-speed On-Chip Oscillator Circuit Electrical Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
—	High-speed on-chip oscillator frequency $f / (td(HR0)feil) \times td(HR)$ when the reset is released	VCC=5.0V, Topr=25 °C Bit "40h" in the HR1 register	—	±	—	MHz
td(HR0)feil	Settable high-speed on-chip oscillator minimum period	VCC=5.0V, Topr=25 °C Bit "001h" in the HR1 register	—	81	—	ns
td(HR)	High-speed on-chip oscillator period adjusted unit	Differences when setting "0110" and "0010" in the HR register	—	±	—	ns
—	High-speed on-chip oscillator temperature dependence(1)	Frequency fluctuation in temperature range of -10 °C to 50 °C	—	±5	—	%
—	High-speed on-chip oscillator temperature dependence(2)	Frequency fluctuation in temperature range of -40 °C to 85 °C	—	±10	—	%

NOTES:

1. The measuring condition is Vcc=AVcc=5.0 V and Topr=25 °C.

Table 5.10 Power Circuit Timing Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
td(P-R)	Time for internal power supply stabilization during power-on ²		1		2000	µs
td(R-S)	STOP release time ³				150	µs

Note

1: The measuring condition is Vcc=AVcc=2.7 to 5.5 V and Topr=25 °C.

2: This shows the wait time until the internal power supply generating circuit is stabilized during power-on.

3: This shows the time until BCLK starts from the interrupt acknowledgement to cancel stop mode.

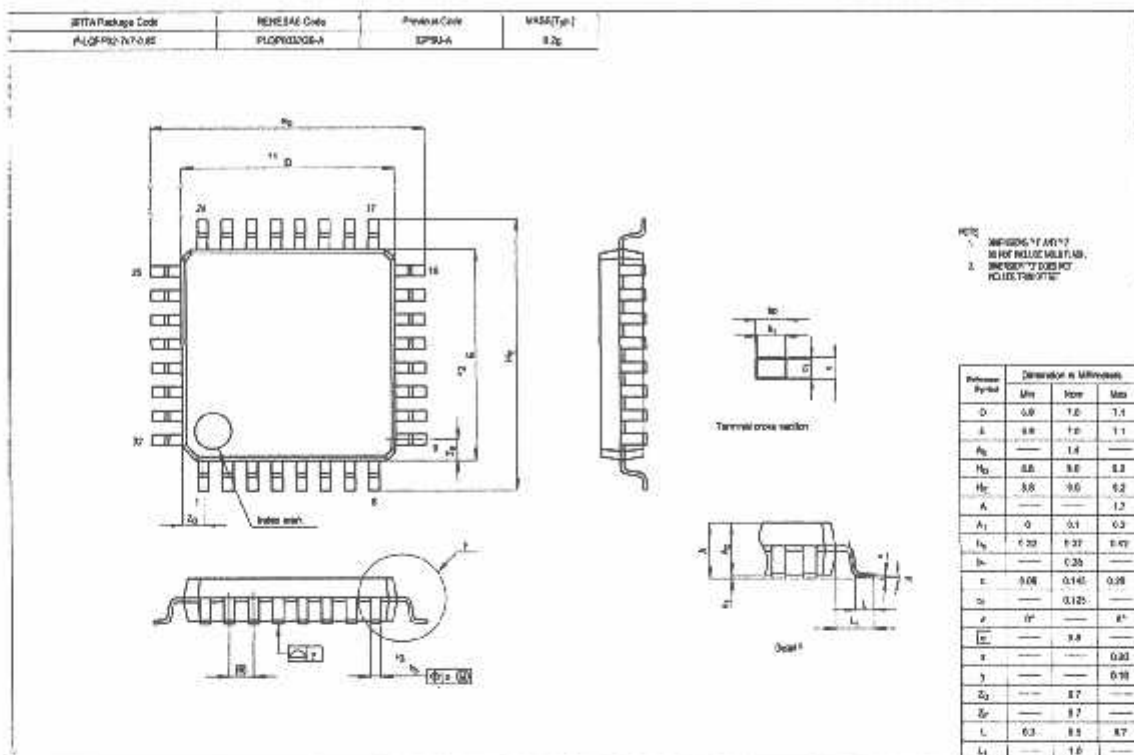
Table 5.11 Electrical Characteristics (1) [Vcc=5V]

Symbol	Parameter	Measuring condition	Standard			Unit		
			Min.	Typ.	Max.			
Voh	"H" output voltage	Except Xout	Ioh=5mA	Vcc-2.0	—	Vcc	V	
			Ioh=200µA	Vcc-0.3	—	Vcc	V	
		Xout	Drive capacity HIGH	Ioh=1 mA	Vcc-2.0	—	Vcc	V
			Drive capacity LOW	Ioh=500µA	Vcc-2.0	—	Vcc	V
Vol	"L" output voltage	P10 to P17 Except Xout	IOL=5 mA	—	—	2.0	V	
			IOL=200 µA	—	—	0.45	V	
		P10 to P17	Drive capacity HIGH	IOL=15 mA	—	—	2.0	V
			Drive capacity LOW	IOL=5 mA	—	—	2.0	V
		Xout	Drive capacity LOW	IOL=200 µA	—	—	0.45	V
			Drive capacity HIGH	IOL=1 mA	—	—	2.0	V
Vih-Vil	Hysteresis	INT0, INT1, INT2, INT3, K0, K1, K2, K3, CNTR0, CNTR1, TGM, RdD0, RdD1, P45		0.2	—	1.0	V	
		RESET		0.2	—	2.2	V	
Iih	"H" input current	V=5V	—	—	5.0	µA		
Iil	"L" input current	V=0V	—	—	-5.0	µA		
Rpullup	Pull-up resistance	V=0V	30	50	187	kΩ		
Rfwn	Feedback resistance	XIN	—	1.0	—	MΩ		
foscLS	Low-speed on-chip oscillator frequency		40	125	250	KHz		
Vhold	RAM retention voltage	At stop mode	2.0	—	—	V		

Note

1: Referenced to Vcc=AVcc=4.2 to 5.5V at Topr = -20 to 85 °C / -40 to 85 °C, f(XIN)=20MHz unless otherwise specified.

Package Dimensions



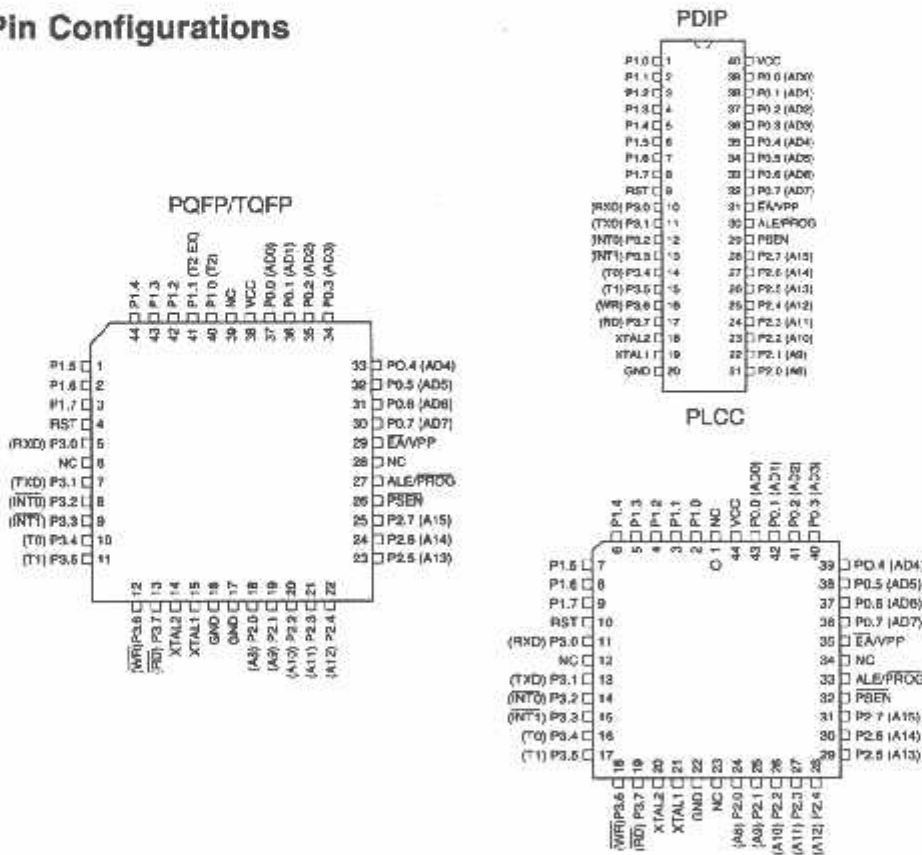
Features

- Compatible with MCS-51™ Products
- 4K Bytes of In-System Reprogrammable Flash Memory
 - Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 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
- Programmable Serial Channel
- Low-power Idle and Power-down Modes

Description

The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard MCS-51 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 Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

Pin Configurations

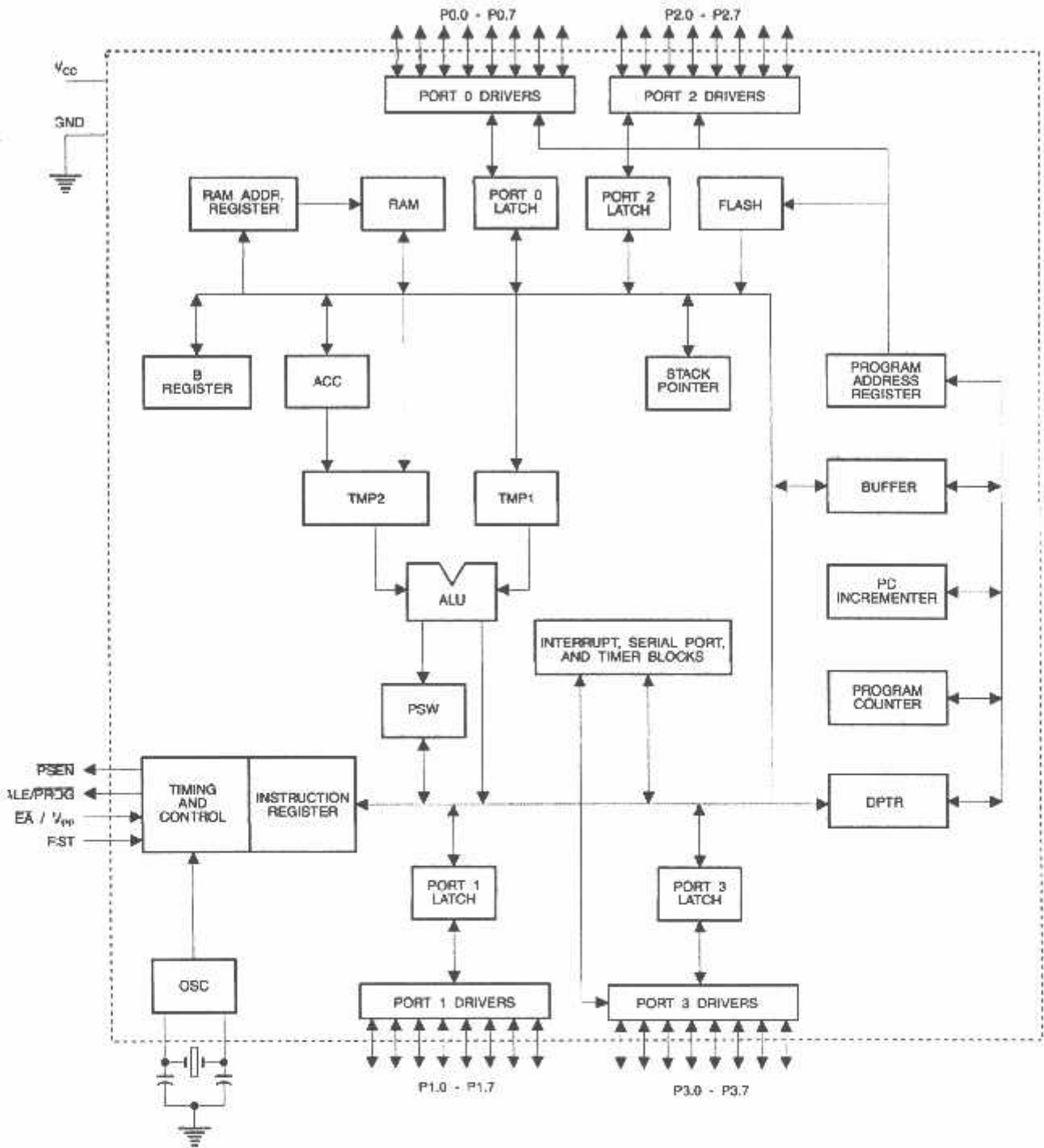


8-bit Microcontroller with 4K Bytes Flash

AT89C51



Block Diagram



The AT89C51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, 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 AT89C51 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 hardware reset.

Pin Description

VCC

Supply voltage.

GND

Ground.

Port 0

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

Port 0 may 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 pullups.

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

Port 1

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

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

Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pullups. 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 pullups 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 pullups.

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, it uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

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

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pullups. 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 pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51 as listed below:

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

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

RST

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

ALE/PROG

Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (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.

PSEN

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

When the AT89C51 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

$\overline{\text{EA}}/\text{VPP}$

External Access Enable. $\overline{\text{EA}}$ 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, $\overline{\text{EA}}$ will be internally latched on reset.

$\overline{\text{EA}}$ 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, for parts that require 12-volt V_{PP} .

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left

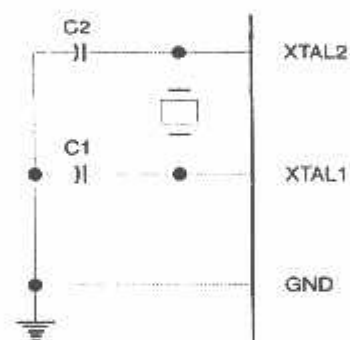
unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

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

It should be noted that when Idle 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 is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Figure 1. Oscillator Connections

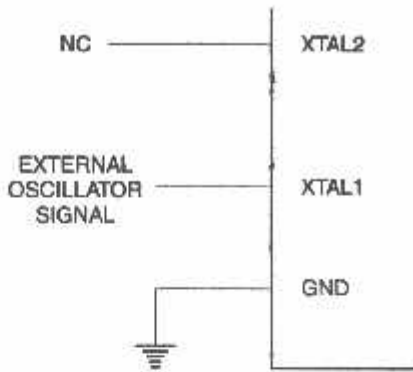


Note: C1, C2 = 30 pF \pm 10 pF for Crystals
= 40 pF \pm 10 pF for Ceramic Resonators

Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	$\overline{\text{PSEN}}$	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

Figure 2. External Clock Drive Configuration



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 Regis-

ters retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. 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.

Program Memory Lock Bits

On the chip are three lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below.

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. It is necessary that the latched value of \overline{EA} be in agreement with the current logic level at that pin in order for the device to function properly.

Lock Bit Protection Modes

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



Programming the Flash

The AT89C51 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (V_{CC}) program enable signal. The low-voltage programming mode provides a convenient way to program the AT89C51 inside the user's system, while the high-voltage programming mode is compatible with conventional third-party Flash or EPROM programmers.

The AT89C51 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

	$V_{PP} = 12V$	$V_{PP} = 5V$
Top-Side Mark	AT89C51 xxxx yyww	AT89C51 xxxx-5 yyww
Signature	(030H) = 1EH (031H) = 51H (032H) = FFH	(030H) = 1EH (031H) = 51H (032H) = 05H

The AT89C51 code memory array is programmed byte-by-byte in either programming mode. *To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.*

Programming Algorithm: Before programming the AT89C51, the address, data and control signals should be set up according to the Flash programming mode table and Figure 3 and Figure 4. To program the AT89C51, 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 for the high-voltage programming mode.
5. Pulse $\overline{ALE}/\overline{PROG}$ 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 1.5 ms. Repeat steps 1 through 5, changing the address

and data for the entire array or until the end of the object file is reached.

Data Polling: The AT89C51 features Data Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on PO.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

Ready/Busy: The progress of byte programming can also be monitored by the RDY/BSY output signal. P3.4 is pulled low after ALE goes high during programming to indicate BUSY. P3.4 is pulled high again when programming is done to indicate READY.

Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase: The entire Flash array is erased electrically by using the proper combination of control signals and by holding $\overline{ALE}/\overline{PROG}$ low for 10 ms. The code array is written with all "1"s. The chip erase operation must be executed before the code memory can be re-programmed.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 030H, 031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

- (030H) = 1EH indicates manufactured by Atmel
- (031H) = 51H indicates 89C51
- (032H) = FFH indicates 12V programming
- (032H) = 05H indicates 5V programming

Programming Interface

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

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

Flash Programming Modes

Mode	RST	PSEN	ALE/PROG	EA/V _{pp}	P2.6	P2.7	P3.6	P3.7
Write Code Data	H	L		H/12V	L	H	H	H
Read Code Data	H	L	H	H	L	L	H	H
Write Lock	H	L		H/12V	H	H	H	H
				H/12V	H	H	L	L
				H/12V	H	L	H	L
Chip Erase	H	L	(1)	H/12V	H	L	L	L
Read Signature Byte	H	L	H	H	L	L	L	L

Note: 1. Chip Erase requires a 10 ms PROG pulse.

Figure 3. Programming the Flash

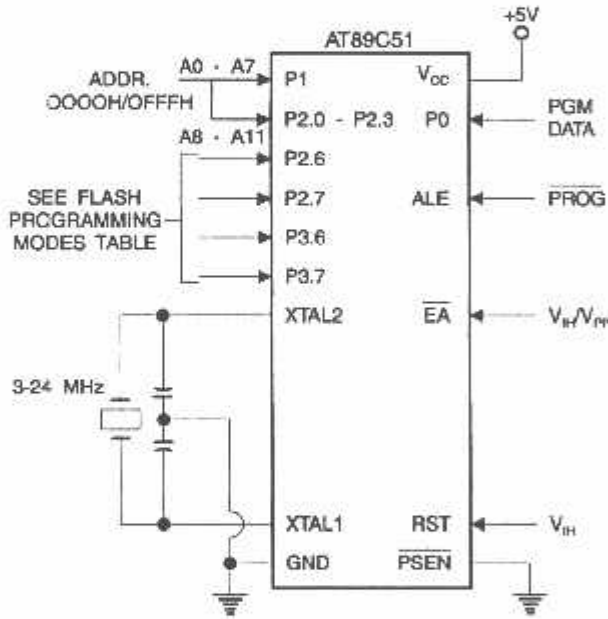
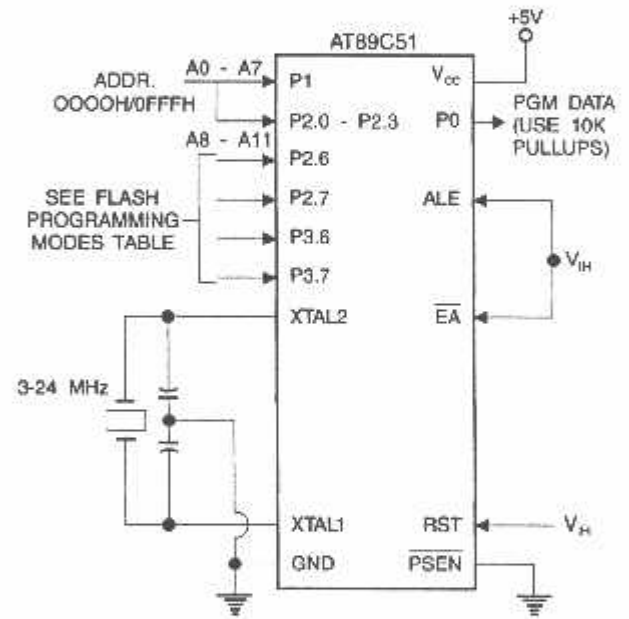
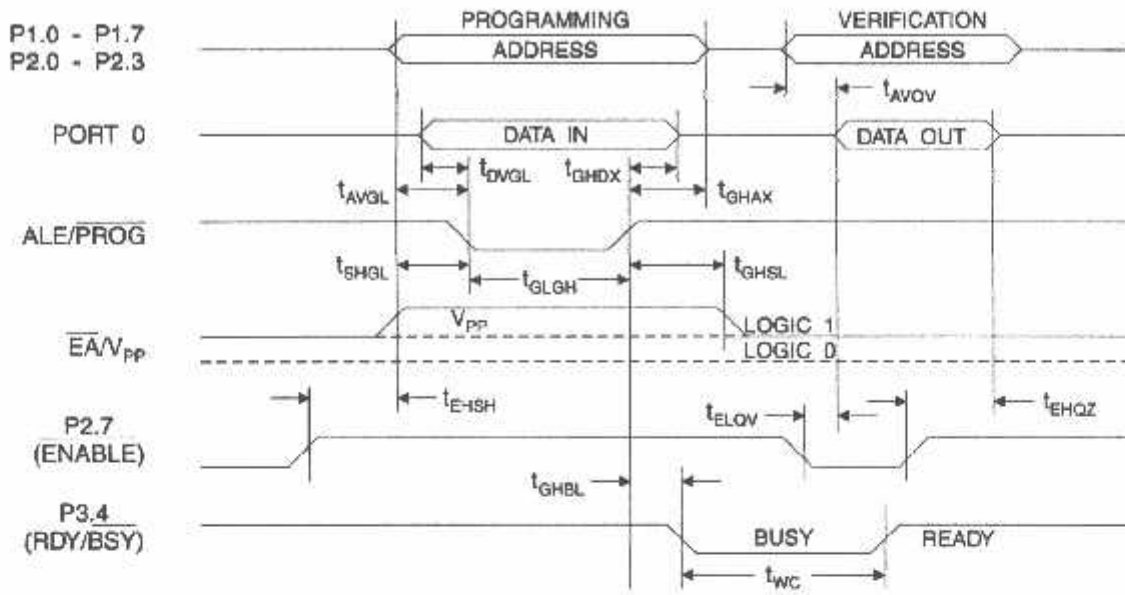


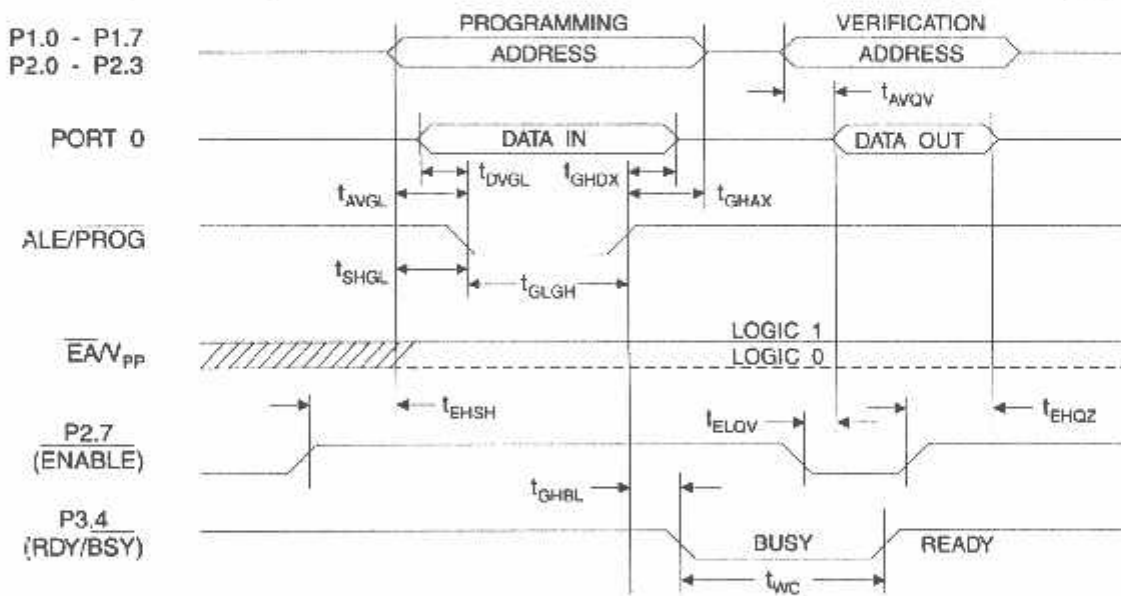
Figure 4. Verifying the Flash



Flash Programming and Verification Waveforms - High-voltage Mode ($V_{PP} = 12V$)



Flash Programming and Verification Waveforms - Low-voltage Mode ($V_{PP} = 5V$)



Flash Programming and Verification Characteristics

$T_A = 0^\circ\text{C to } 70^\circ\text{C}$, $V_{CC} = 5.0 \pm 10\%$

Symbol	Parameter	Min	Max	Units
$V_{PP}^{(1)}$	Programming Enable Voltage	11.5	12.5	V
$I_{PP}^{(1)}$	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold After $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{DVG_L}	Data Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHD_X}	Data Hold After $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{EHS}	P2.7 ($\overline{\text{ENABLE}}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{\text{PROG}}$ Low	10		μs
$t_{GHSL}^{(1)}$	V_{PP} Hold After $\overline{\text{PROG}}$	10		μs
t_{GLG_P}	$\overline{\text{PROG}}$ Width	1	110	μs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{\text{ENABLE}}$ Low to Data Valid		$48t_{CLCL}$	
t_{FH02}	Data Float After $\overline{\text{ENABLE}}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{\text{PROG}}$ High to $\overline{\text{BUSY}}$ Low		1.0	μs
t_{WC}	Byte Write Cycle Time		2.0	ms

Note: 1. Only used in 12-volt programming mode.



ISD2560/75/90/120 Products

Single-Chip Voice Record/Playback Devices

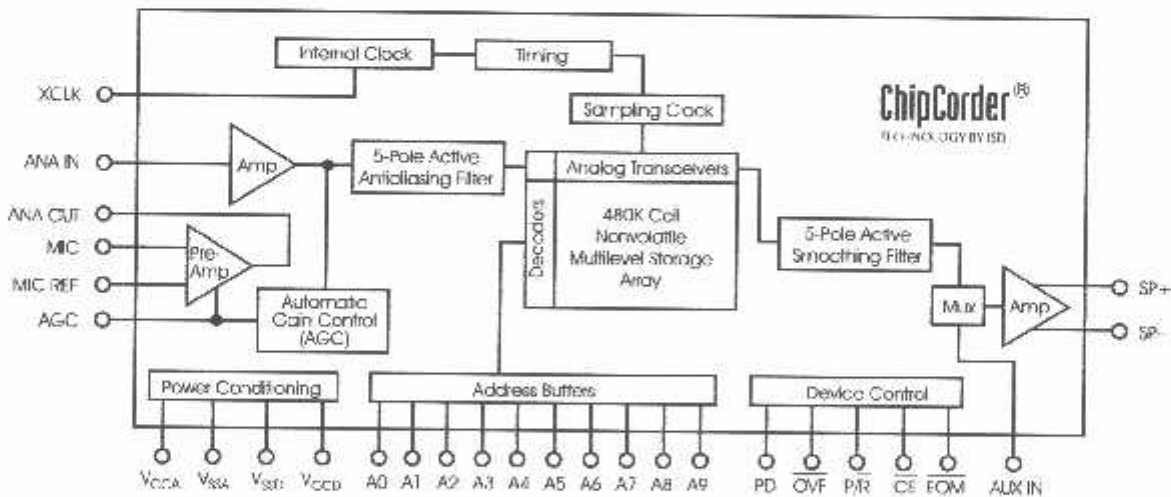
60-, 75-, 90-, and 120-Second Durations

GENERAL DESCRIPTION

Information Storage Devices' ISD2500 ChipCorder[®] Series provides high-quality, single-chip record/playback solutions for 60- to 120-second messaging applications. The CMOS devices include an on-chip oscillator, microphone preamplifier, automatic gain control, antialiasing filter, smoothing filter, speaker amplifier, and high density multilevel storage array. In addition, the ISD2500 is microcontroller compatible, allowing complex messaging and addressing to be achieved.

Recordings are stored in on-chip nonvolatile memory cells, providing zero-power message storage. This unique, single-chip solution is made possible through ISD's patented multilevel storage technology. Voice and audio signals are stored directly into memory in their natural form, providing high-quality, solid-state voice reproduction.

Figure i: ISD2560/75/90/120 Device Block Diagram



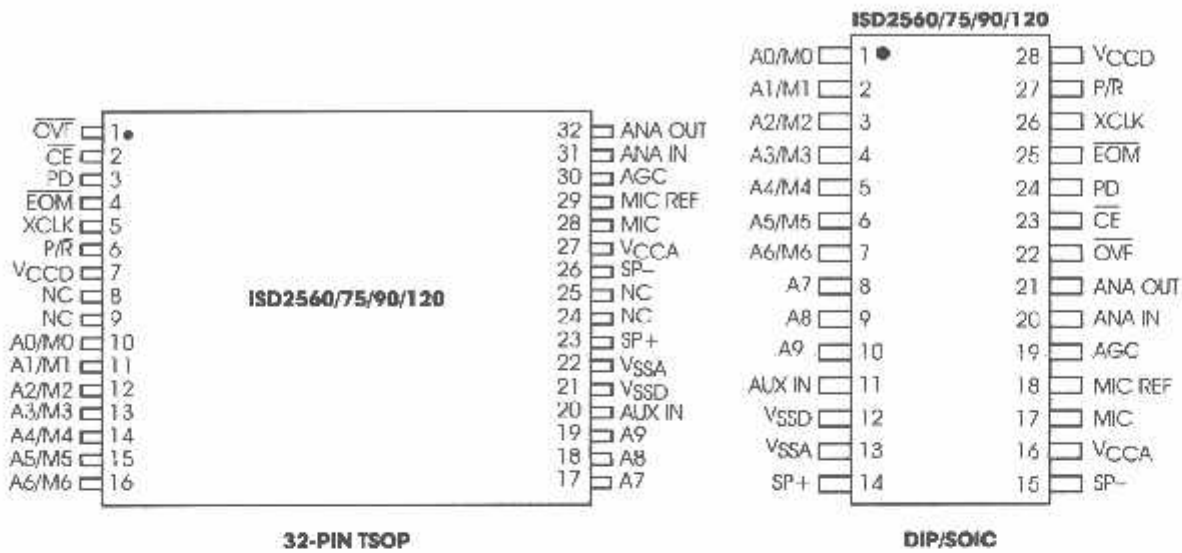
FEATURES

- 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
 - Single-chip durations of 60, 75, 90, and 120 seconds
 - Directly cascadable for longer durations
 - Automatic Power-Down (Push-Button Mode)
 - Standby current 1 μ A (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 clock source
 - Programmer support for play-only applications
 - Single +5 volt power supply
 - Available in die form, DIP, SOIC, and TSOP packaging
-

Table i: ISD2560/75/90/120 Product Summary

Part Number	Duration (Seconds)	Input Sample Rate (KHz)	Typical Filter Pass Band (KHz)
ISD2560	60	8.0	3.4
ISD2575	75	6.4	2.7
ISD2590	90	5.3	2.3
ISD25120	120	4.0	1.7

Figure 1: ISD2560/75/90/120 Device Pinouts



PIN DESCRIPTIONS

VOLTAGE INPUTS (V_{CCA}, V_{CCD})

To minimize noise, the analog and digital circuits in the ISD2500 series devices use separate power busses. These voltage busses are brought out to separate pins and should be tied together as close to the supply as possible. In addition, these supplies should be decoupled as close to the package as possible.

GROUND INPUTS (V_{SSA}, V_{SSD})

The ISD2500 series of devices utilizes separate analog and digital ground busses. These pins should be connected separately through a low-impedance path to power supply ground.

POWER DOWN INPUT (PD)

When not recording or playing back, the PD pin should be pulled HIGH to place the part in a very low power mode (see I_{SB} specification). When overflow (OVF) pulses LOW for an overflow condition, PD should be brought HIGH to reset the address pointer back to the beginning of the record/playback space. The PD pin has additional functionality in the M6 (Push-Button) Operational Mode described later in the Operational Mode section.

CHIP ENABLE INPUT (CE)

The CE pin is taken LOW to enable all playback and record operations. The address inputs and playback/record input (P/R) are latched by the falling edge of CE. CE has additional functionality in the M6 (Push-Button) Operational Mode described later in the Operational Mode section.

PLAYBACK/RECORD INPUT (P/R)

The P/R input is latched by the falling edge of the \overline{CE} pin. A HIGH level selects a playback cycle while a LOW level selects a record cycle. For a record cycle, the address inputs provide the starting address and recording continues until PD or CE is pulled HIGH or an overflow is detected (i.e. the chip is full). When a record cycle is terminated by pulling PD or CE HIGH, an End-Of-Message (EOM) marker is stored at the current address in memory. For a playback cycle, the address inputs provide the starting address and the device will play until an EOM marker is encountered. The device can continue past an EOM marker in an Operational Mode, or if \overline{CE} is held LOW in address mode. (See page 5 for more Operational Modes).

END-OF-MESSAGE / RUN OUTPUT (EOM)

A nonvolatile marker is automatically inserted at the end of each recorded message. It remains there until the message is recorded over. The EOM output pulses LOW for a period of T_{EOM} at the end of each message.

In addition, the ISD2500 series has an internal V_{CC} detect circuit to maintain message integrity should V_{CC} fall below 3.5 V. In this case, EOM goes LOW and the device is fixed in playback-only mode.

When the device is configured in Operational Mode M6 (Push-Button Mode), this pin provides an active-HIGH RUN signal, indicating the device is currently recording or playing. This signal can conveniently drive an LED for a visual indicator of a record or playback operation in process.

OVERFLOW OUTPUT (OVF)

This signal pulses LOW at the end of memory space, indicating the device has been filled and the message has overflowed. The \overline{OVF} output then follows the CE Input until a PD pulse has reset the device. This pin can be used to cascade several ISD2500 devices together to increase record/playback durations.

MICROPHONE INPUT (MIC)

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 24 dB. An external microphone should be AC coupled to this pin via a series capacitor. The capacitor value, together with the internal 10 K Ω resistance on this pin, determines the low-frequency cutoff for the ISD2500 series passband. See Application Information for additional information on low-frequency cutoff calculation.

MICROPHONE REFERENCE INPUT (MIC REF)

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.

AUTOMATIC GAIN CONTROL INPUT (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 whispers to loud sounds to be recorded with minimal distortion. The "attack" time is determined by the time constant of a 5 K Ω internal resistance and an external capacitor (C2 on the schematic on page 18) connected from the AGC pin to V_{SSA} analog ground. The "release" time is determined by the time constant of an external resistor (R2) and an external capacitor (C2) connected in parallel between the AGC Pin and V_{SSA} analog ground. Nominal values of 470 K Ω and 4.7 μ F give satisfactory results in most cases.

ANALOG OUTPUT (ANA OUT)

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.

ANALOG INPUT (ANA IN)

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 K Ω 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.

EXTERNAL CLOCK INPUT (XCLK)

The external clock input for the ISD2500 devices has an internal pull-down device. These devices are configured at the factory with an internal sampling clock frequency centered to ± 1 percent of specification. The frequency is then maintained to a variation of ± 2.25 percent over the entire commercial temperature and operating voltage ranges. The internal clock has a ± 5 percent tolerance over the industrial temperature and voltage range. A regulated power supply is recommended for industrial temperature range parts. If greater precision is required, the device can be clocked through the XCLK pin as follows:

Table 1: External Clock Sample Rates

Part Number	Sample Rate	Required Clock
ISD2560	8.0 KHz	1024 KHz
ISD2575	6.4 KHz	819.2 KHz
ISD2590	5.3 KHz	662.7 KHz
ISD25120	4.0 KHz	512 KHz

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.**

SPEAKER OUTPUTS (SP+ /SP-)

All devices in the ISD2500 series include an on-chip differential speaker driver, capable of driving 50 mW into 16 Ω from AUX IN (12.2 mW from memory).

The speaker outputs are held at V_{SSA} levels during record and power down. It is therefore not possible to parallel speaker outputs of multiple ISD2500 devices or the outputs of other speaker drivers.

NOTE Connection of speaker outputs in parallel may cause damage to the device.

A single output may be used alone (including a coupling capacitor between the SP pin and the speaker). These outputs may be used individually with the output signal taken from either pin. Using the differential outputs results in a 4 to 1 improvement in output power.

NOTE Never ground or drive an unused speaker output.

AUXILIARY INPUT (AUX IN)

The Auxiliary Input is multiplexed through to the output amplifier and speaker output pins when CE is HIGH, P/R is HIGH, and playback is currently not active or if the device is in playback overflow. When cascading multiple ISD2500 devices, the AUX IN pin is used to connect a playback signal from a following device to the previous output speaker drivers. For noise considerations, it is suggested that the auxiliary input not be driven when the storage array is active.

ADDRESS/MODE INPUTS (AX/MX)

The Address/Mode Inputs have two functions depending on the level of the two Most Significant Bits (MSB) of the address (A8 and A9).

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{CE} .

If both MSBs are HIGH, the Address/Mode Inputs are interpreted as Mode bits according to the Operational Mode table. There are six Operational Modes (M0..M6) available as indicated in the table. It is possible to use multiple Operational Modes simultaneously. Operational Modes are sampled on each falling edge of \overline{CE} , and thus Operational Modes and direct addressing are mutually exclusive.

OPERATIONAL MODES

The ISD2500 series is designed with several built-in Operational Modes that provide maximum functionality with minimum additional components. These are described in detail below. The Operational Modes use the address pins on the ISD2500 devices, but are mapped outside the valid address range. When the two Most Significant Bits (MSBs) are HIGH (A8 and A9), 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. First, all operations begin initially at address 0, which is the beginning of the ISD2500 address space. 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, playback to record (except M6 mode), or when a Power-Down cycle is executed.

Second, Operational Modes are executed when \overline{CE} goes LOW and the two MSBs are HIGH. This Operational Mode remains in effect until the next LOW-going \overline{CE} signal, at which point the current address/mode levels are sampled and executed.

Table 2: Operational Modes Table

Mode Control	Function	Typical Use	Jointly Compatible ¹
M0	Message cueing	Fast-forward through messages	M4, M5, M6
M1	Delete EOM markers	Position EOM marker at the end of the last message	M3, M4, M5, M6
M2	Not applicable	Reserved	N/A
M3	Looping	Continuous playback from Address 0	M1, M5, M6
M4	Consecutive addressing	Record/play multiple consecutive messages	M0, M1, M5
M5	\overline{CE} level-activated	Allows message pausing	M0, M1, M3, M4
M6	Push-button control	Simplified device interface	M0, M1, M3

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

OPERATIONAL MODES DESCRIPTION

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

M0 — MESSAGE CUEING

Message Cueing allows the user to skip through messages, without knowing the actual physical addresses of each message. Each \overline{CE} LOW pulse causes the internal address pointer to skip to the next message. This mode should be used for playback only, and is typically used with the M4 Operational Mode.

M1 — DELETE EOM MARKERS

The M1 Operational Mode allows sequentially recorded messages to be combined into a single message with only one EOM marker set at the end of the final message. When this Operational Mode is configured, messages recorded sequentially are played back as one continuous message.

M2 — UNUSED

When Operational Modes are selected, the M2 pin should be LOW.

M3 — MESSAGE LOOPING

The M3 Operational Mode allows for the automatic, continuously repeated playback of the message located at the beginning of the address space. A message can completely fill the ISD2500 device and will loop from beginning to end without \overline{OVF} going LOW.

M4 — CONSECUTIVE ADDRESSING

During normal operations, the address pointer will reset when a message is played through to an EOM marker. The M4 Operational Mode inhibits the address pointer reset on EOM, allowing messages to be played back consecutively.

M5 — \overline{CE} -LEVEL ACTIVATED

The default mode for ISD2500 devices is for \overline{CE} to be edge-activated on playback and level-activated on record. The M5 Operational Mode causes the \overline{CE} pin to be interpreted as level-activated as opposed to edge-activated during playback. This is specifically useful for terminating playback operations using the \overline{CE} signal.

In this mode, \overline{CE} LOW begins a playback cycle, at the beginning of the device memory. The playback cycle continues as long as \overline{CE} is held LOW. When \overline{CE} goes HIGH, playback will immediately end. A new \overline{CE} LOW will restart the message from the beginning unless M4 is also HIGH.

M6 — PUSH-BUTTON MODE

The ISD2500 series of devices contain a Push-Button Operational Mode. The Push-Button mode is used primarily in very low-cost applications and is designed to minimize external circuitry and components, thereby reducing system cost. In order to configure the device in Push-Button Operational Mode, the two most significant address bits must be HIGH, and the M6 mode pin must also be HIGH. A device in this mode always powers down at the end of each playback or record cycle after \overline{CE} goes HIGH.

When this Operational Mode is implemented, several of the pins on the device have alternate functionality:

Table 3: Alternate Functionality in Pins

Pin Name	Alternate Functionality in Push-Button Mode
\overline{CE}	Start/Pause Push-Button (LOW pulse-activated)
PD	Stop/Reset Push-Button (HIGH pulse activated)
EOM	Active-HIGH Run Indicator

CE PIN (START/PAUSE)

In Push-Button Operational Mode, \overline{CE} acts as a LOW-going pulse-activated START/PAUSE signal. If no operation is currently in progress, a LOW-going pulse on this signal will initiate a playback or a record cycle according to the level on the P/\overline{R} pin. A subsequent pulse on the \overline{CE} pin, before an End-Of-Message is reached in playback or an overflow condition occurs, will cause the device to pause. The address counter is not reset, and another \overline{CE} pulse will cause the device to continue the operation from the place where it was paused.

PD PIN (STOP/RESET)

In push-button Operational Mode, PD acts as a HIGH-going pulse-activated STOP/RESET signal. When a playback or record cycle is in progress and a HIGH-going pulse is observed on PD, the current cycle is terminated and the address pointer is reset to address 0, the beginning of the message space.

EOM PIN (RUN)

In Push-Button Operational Mode, EOM becomes an active-HIGH RUN signal which can be used to drive an LED or other external device. It is HIGH whenever a record or playback operation is in progress.

Recording in Push-Button Mode

1. The PD pin should be LOW, usually using a pull-down resistor.
2. The P/\overline{R} pin is taken LOW.
3. The \overline{CE} pin is pulsed LOW. Recording starts, EOM goes HIGH to indicate an operation in progress.
4. The \overline{CE} pin is pulsed LOW. Recording pauses, EOM goes back LOW. The internal address pointers are not cleared, but an EOM marker is stored in memory to point to the message end. The P/\overline{R} pin may be taken HIGH at this time. Any subsequent \overline{CE} would start a playback at address 0.

5. The \overline{CE} pin is pulsed LOW. Recording starts at the next address after the previous set EOM marker. EOM goes back HIGH.

NOTE *If the M1 Operational Mode pin is also HIGH, the just previously written EOM bit is erased, and recording starts at that address.)*

6. When the recording sequences are finished, the final \overline{CE} pulse LOW will end the last record cycle, leaving a set EOM marker at the message end. Recording may also be terminated by a HIGH level on PD, which will leave a set EOM marker.

Playback in Push-Button Mode

1. The PD pin should be LOW.
2. The P/\overline{R} pin is taken HIGH.
3. The \overline{CE} pin is pulsed LOW. Playback starts, EOM goes HIGH to indicate an operation in progress.
4. If the \overline{CE} pin is pulsed LOW or an EOM marker is encountered during an operation, the part will pause. The internal address pointers are not cleared, and EOM goes back LOW. The P/\overline{R} pin may be changed at this time. A subsequent record operation would not reset the address pointers and the recording would begin where playback ended.
5. \overline{CE} is again pulsed LOW. Playback starts where it left off, with EOM going HIGH to indicate an operation in progress.
6. Playback continues as in steps 4 and 5 until PD is pulsed HIGH or overflow occurs.
7. If in overflow, pulling \overline{CE} LOW will reset the address pointer and start playback from the beginning. After a PD pulse, the part is reset to address 0.

NOTE *Push-button mode can be used in conjunction with modes M0, M1, and M3.*

GOOD AUDIO DESIGN PRACTICES

ISD products are very high-quality single-chip voice recording and playback systems. To ensure the highest quality voice reproduction, it is important that good audio design practices on layout and power supply decoupling be followed. See the ISD Application Notes in this book for details.

ISD1000A COMPATIBILITY

The ISD2500 series of devices is designed to provide upward compatibility with the ISD1000A family. When designing with the ISD2500 series, the following differences should be noted.

ADDRESSING

The ISD2560/75/90/120 devices have 480K storage cells designed to provide 60 seconds of storage at a sampling rate of 8.0 KHz. This is approximately four times the storage of the ISD1000A family. To enable the same addressing resolution, two additional address pins have been added. The address space of each device is divisible into 600 increments with valid addressing from 00 to 257 Hex. Some higher addresses are mapped into the Operational Modes. All other addresses are invalid.

OVERFLOW

The ISD1000A series combined two functions on the EOM pin: end-of-message indication and overflow. The ISD2500 separates these two functions. Pin 25 (PDIP package) remains as $\overline{\text{EOM}}$, but outputs only the EOM signal indication. Pin 22 (PDIP package) becomes OVF and pulses LOW only when the device reaches its end of memory, or is "full." This change allows easy message cueing and addressability across device boundaries. This also means that the M2 Operational Mode found in the ISD1000A family is not implemented in the ISD2500 series.

PUSH-BUTTON MODE

The ISD2500 series includes an additional Operational Mode called Push-Button mode. This provides an alternative interface to the record and playback functions of the part. The CE and PD pins become redefined as edge-activated "push-buttons." A pulse on CE initiates a cycle, and if triggered again, pauses the current cycle without resetting the address pointer (i.e., a Start or Pause function). PD stops any current cycle and resets the address pointer to the beginning of the message space (i.e., a Stop and Reset function). Additionally, the EOM pin functions as an active-HIGH run indicator, and can be used to drive an LED indicating a record or playback operation is in progress. Devices in the Push-Button mode cannot be cascaded.

LOOPING MODE

The ISD2500 series can loop with a message that completely fills the memory space.

NOTE Additional descriptions of ISD2500 device functionality and application examples are provided in the ISD Application Notes in this book.

TIMING DIAGRAMS

Figure 2: Record

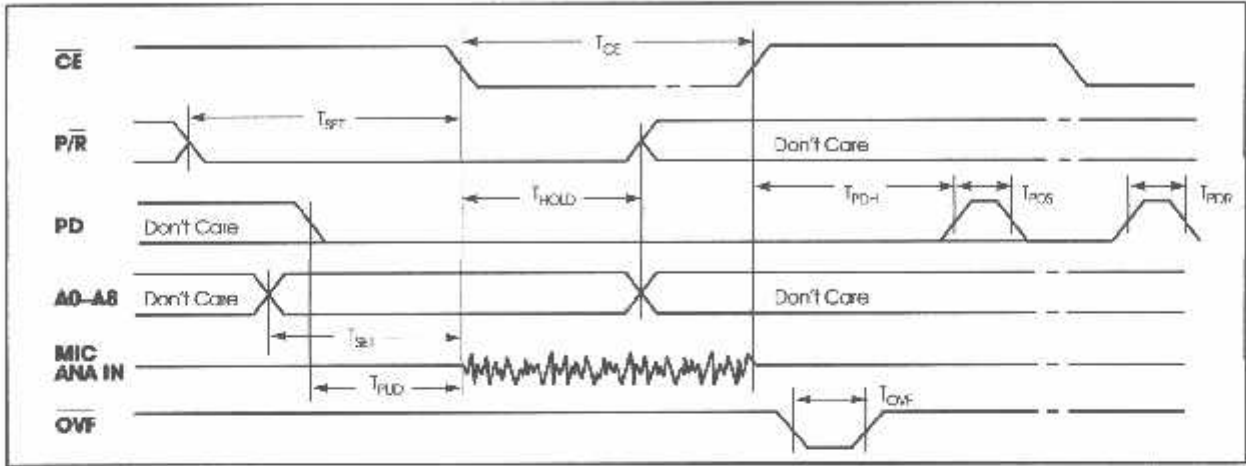


Figure 3: Playback

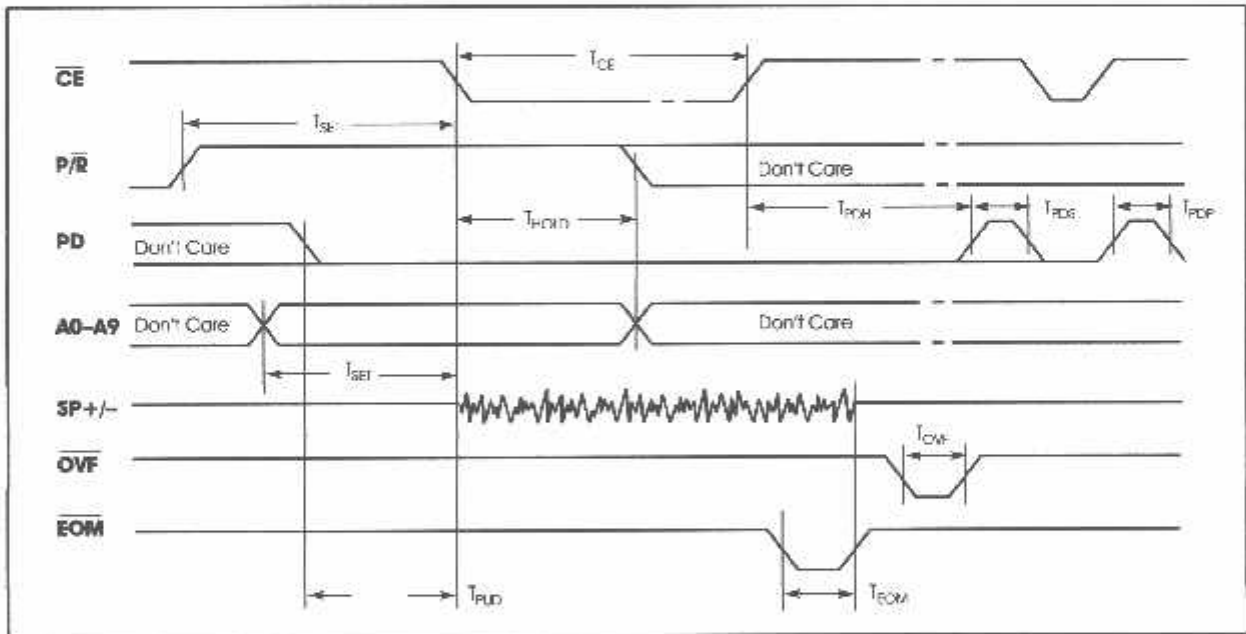


Table 4: Absolute Maximum Ratings (Packaged Parts)⁽¹⁾

Condition	Value
Junction temperature	150°C
Storage temperature range	-65°C to +150°C
Voltage applied to any pin	(V _{SS} - 0.3 V) to (V _{CC} + 0.3 V)
Voltage applied to any pin (Input current limited to ±20 mA)	(V _{SS} - 1.0 V) to (V _{CC} + 1.0 V)
Lead temperature (soldering - 10 seconds)	300°C
V _{CC} - V _{SS}	-0.3 V to +7.0 V

Table 5: Operating Conditions (Packaged Parts)

Condition	Value
Commercial operating temperature range ⁽¹⁾	0°C to +70°C
Industrial operating temperature range ⁽¹⁾	
Supply voltage (V _{CC}) ⁽²⁾	+4.5 V to +5.5 V
Ground voltage (V _{SS}) ⁽³⁾	0 V

1. Case temperature.
2. V_{CC} = V_{CCA} = V_{CCD}.
3. V_{SS} = V_{SSA} = V_{SSD}.
4. Consult factory.

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

Table 6: DC Parameters (Packaged Parts)

Symbol	Parameters	Min ⁽²⁾	Typ ⁽¹⁾	Max ⁽²⁾	Units	Conditions
V _{IL}	Input Low Voltage			0.8	V	
V _{IH}	Input High Voltage	2.0			V	
V _{OL}	Output Low Voltage			0.4	V	I _{OL} = 4.0 mA
V _{OH}	Output High Voltage	V _{CC} -0.4			V	I _{OH} = -10 μA
V _{OH1}	OVF Output High Voltage	2.4			V	I _{OH} = -1.6 mA
V _{OH2}	EOM Output High Voltage	V _{CC} - 1.0	V _{CC} - 0.8		V	I _{OH} = -3.2 mA
I _{CC}	V _{CC} Current (Operating)		25	30	mA	R _{FXT} = ∞ ⁽³⁾
I _{SB}	V _{CC} Current (Standby)		1	10	μA	⁽³⁾
I _{IL}	Input Leakage Current			±1	μA	
I _{ILPD}	Input Current HIGH w/Pull Down			130	μA	Force V _{CC} ⁽⁴⁾
R _{EXT}	Output Load Impedance	16			Ω	Speaker Load
R _{MIC}	Preamp In Input Resistance	4	9	15	KΩ	MIC and MIC REF Pins
R _{AUX}	AUX INPUT Resistance	5	11	20	KΩ	

Table 6: DC Parameters (Packaged Parts)

Symbol	Parameters	Min ⁽²⁾	Typ ⁽¹⁾	Max ⁽²⁾	Units	Conditions
R _{ANA IN}	ANA IN Input Resistance	2.3	3	5	$\lt \Omega$	
A _{PRE1}	Preamp Gain 1	21	24	26	dB	AGC = 0.0 V
A _{PRE2}	Preamp Gain 2		-15	5	dB	AGC = 2.5 V
A _{AUX}	AUX IN/SP+ Gain		0.98	1.0	V/V	
A _{A2P}	ANA IN to SP+/- Gain	21	23	26	dB	
R _{AGC}	AGC Output Resistance	2.5	5	9.5	K Ω	

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

2. All Min/Max limits are guaranteed by ISD via electrical testing or characterization. Not all specifications are 100 percent tested.

3. V_{CCA} and V_{CCB} connected together.

4. XCLK pin only.

Table 7: AC Parameters (Packaged Parts)

Symbol	Characteristic	Min ⁽²⁾	Typ ⁽¹⁾	Max ⁽²⁾	Units	Conditions
F _S	Sampling Frequency	ISD2560		8.0		KHz (7)
		ISD2575		6.4		KHz (7)
		ISD2590		5.3		KHz (7)
		ISD25120		4.0		KHz (7)
F _{CF}	Filter Pass Band	ISD2560		3.4		KHz 3 dB Roll-Off Point (3) (8)
		ISD2575		2.7		KHz 3 dB Roll-Off Point (3) (8)
		ISD2590		2.3		KHz 3 dB Roll-Off Point (3) (8)
		ISD25120		1.7		KHz 3 dB Roll-Off Point (3) (8)
T _{REC}	Record Duration	ISD2560	58.1	60.0	62.0	sec Commercial Operation ⁽⁷⁾
		ISD2560	56.5	60.0	63.8	sec Industrial Operation ⁽⁷⁾
		ISD2575	72.6	75.0	77.5	sec Commercial Operation ⁽⁷⁾
		ISD2575	70.7	75.0	79.7	sec Industrial Operation ⁽⁷⁾
		ISD2590	87.1	90.0	93.0	sec Commercial Operation ⁽⁷⁾
		ISD25120	116.1	120.0	123.9	sec Commercial Operation ⁽⁷⁾
T _{PLAY}	Playback Duration	ISD2560	58.1	60.0	62.0	sec Commercial Operation
		ISD2560	56.5	60.0	63.8	sec Industrial Operation
		ISD2575	72.6	75.0	77.5	sec Commercial Operation
		ISD2575	70.7	75.0	79.7	sec Industrial Operation
		ISD2590	87.1	90.0	93.0	sec Commercial Operation
		ISD25120	116.1	120.0	123.9	sec Commercial Operation
T _{CE}	CE Pulse Width		100		nsec	
T _{SET}	Control/Address Setup Time		300		nsec	
T _{HOLD}	Control/Address Hold Time		0		nsec	

Table 7: AC Parameters (Packaged Parts)

Symbol	Characteristic	Min ⁽²⁾	Typ ⁽¹⁾	Max ⁽²⁾	Units	Conditions	
T _{PUD}	Power-Up Delay	ISD2560	24.1	25.0	27.8	msec	Commercial Operation
		ISD2560	23.5		28.5	msec	Industrial Operation
		ISD2575	30.2	31.3	34.3	msec	Commercial Operation
		ISD2575	29.3	31.3	35.2	msec	Industrial Operation
		ISD2590	36.2	37.5	40.8	msec	Commercial Operation
		ISD25120	48.2	50.0	53.6	msec	Commercial Operation
T _{PDR}	PD Pulse Width Record	ISD2560		25		msec	
		ISD2575		31.25		msec	
		ISD2590		37.5		msec	
		ISD25120		50.0		msec	
T _{PDP}	PD Pulse Width Play	ISD2560		12.5		msec	
		ISD2575		15.625		msec	
		ISD2590		18.75		msec	
		ISD25120		25.0		msec	
T _{PDS}	PD Pulse Width Static		100		nsec	(6)	
T _{PDH}	Power Down Hold		0		nsec		
T _{EOM}	EOM Pulse Width	ISD2560		12.5		msec	
		ISD2575		15.625		msec	
		ISD2590		18.75		msec	
		ISD25120		25.0		msec	
T _{OMF}	Overflow Pulse Width		6.5		μsec		
THD	Total Harmonic Distortion		1	2	%	@ 1 KHz	
P _{OUT}	Speaker Output Power		12.2	50	mW	R _{EXT} = 16 Ω ⁽⁴⁾	
V _{OUT}	Voltage Across Speaker Pins			2.5	V p-p	R _{EXT} = 600 Ω	
V _{IN1}	MIC Input Voltage			20	mV	Peak-to-Peak ⁽⁵⁾	
V _{IN2}	ANA IN Input Voltage			50	mV	Peak-to-Peak	
V _{IN3}	Aux Input Voltage			1.25	V	Peak-to-Peak; R _{EXT} = 16 Ω	

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

2. All Min/Max limits are guaranteed by ISD via electrical 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. From AUX IN; if ANA IN is driven at 50 mV p-p, the P_{OUT} = 12.2 mW, typical.

5. With 5.1 KΩ series resistor at ANA IN.

6. T_{PDS} is required during a static condition, typically overflow.

7. Sampling Frequency and playback Duration can vary as much as ±2.25 percent over the commercial temperature range and voltage range and +5 percent over the industrial temperature and voltage range. For greater stability, an external clock can be utilized (see Pin Descriptions).

8. Filter specification applies to both the anti-aliasing filter and the smoothing filter. Therefore, from input to output, expect a 6 dB drop by nature of passing through both filters.

LIQUID CRYSTAL DISPLAY MODULE

M 1 6 3 2

USER MANUAL

Seiko Instruments Inc.

1. GENERAL

1.1 General

The M1632 is a low-power-consumption dot-matrix liquid crystal display (LCD) module with a high-contrast wide-view TN LCD panel and a CMOS LCD drive controller built in. The controller has a built-in character generator ROM/RAM, and display data RAM. All the display functions are controlled by instructions and the module can easily be interfaced with an MPU. This makes the module applicable to a wide range of purposes including terminal display units for microcomputers and display units for measuring gages.

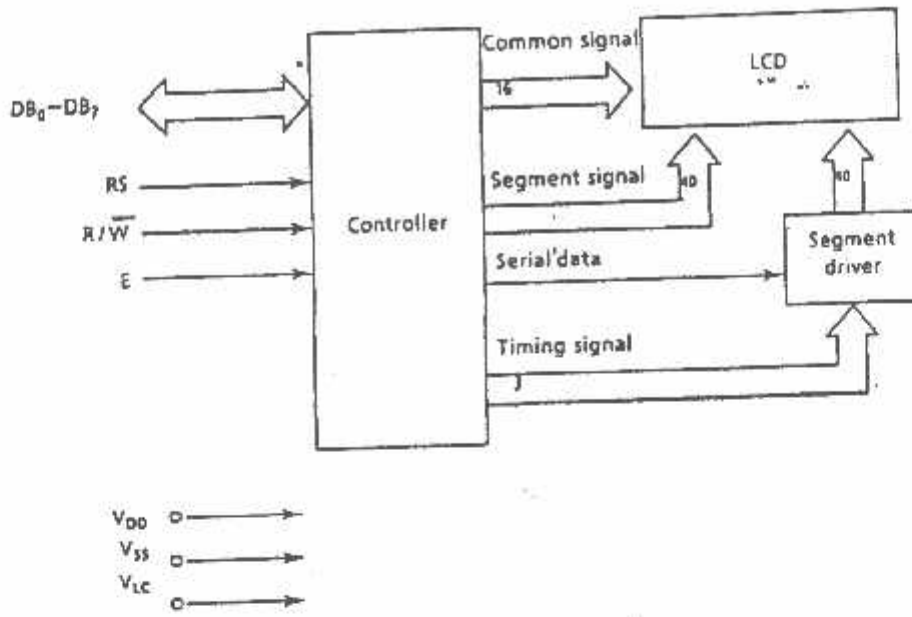
1.2 Features

- 16-character, two-line TN liquid crystal display of 5 x 7 dot matrix + cursor
- Duty ratio: 1/16
- Character generator ROM for 192 character types (character font: 5 x 7 dot matrix)
- Character generator RAM for eight character types (program write) (character font: 5 x 7 dot matrix)
- 80 x 8 bit display data RAM (80 characters maximum)
- Interface with four-bit and eight-bit MPUs possible
- Display data RAM and character generator RAM readable from MPU
- Many instruction functions

Display Clear, Cursor Home, Display ON/OFF, Cursor ON/OFF, Display Character Blink, Cursor Shift, and Display Shift

- Built-in oscillator circuit
- +5 V single power supply
- Built-in automatic reset circuit at power-on
- CMOS process
- Operating temperature range: 0°C to 50°C

1.4 Block Diagram



1.5 Absolute Maximum Ratings

$V_{SS} = 0V$

Item	Symbol	Standard	Unit	Remarks
Power supply voltage	V_{DD}	-0.3 to +7.0	V	
	V_{LC}	$V_{DD} - 13.5$ to $V_{DD} + 0.3$	V	
Input voltage	V_{in}	-0.3 to $V_{DD} + 0.3$	V	
Operating temperature	T_{opr}	0 to +50	°C	
Storage temperature	T_{stg}	-20 to +60	°C	At 50% RH

1.6 Electrical Characteristics

$V_{DD} = 5V \pm 5\%$, $V_{SS} = 0V$, $T_A = 0^\circ C$ to $50^\circ C$

Item	Symbol	Conditions	Standard			Unit
			Min.	Typ.	Max.	
Input voltage	High	V_{IH1}	2.2	-	V_{DD}	V
	Low	V_{IL1}	0	-	0.6	V
Output voltage (TTL)	High	V_{OH1} $-I_{OH} = 0.205 \text{ mA}$	2.4	-	-	V
	Low	V_{OL1} $I_{OL} = 1.2 \text{ mA}$	-	-	0.4	V
Output voltage (CMOS)	High	V_{OH2} $-I_{OH} = 0.04 \text{ mA}$	$0.9V_{DD}$	-	-	V
	Low	V_{OL2} $I_{OL} = 0.04 \text{ mA}$	-	-	$0.1V_{DD}$	V
Power supply voltage	V_{DD}		4.75	5.00	5.25	V
	$-V_{LC}$	$V_{DD} = 5V$, $T_A = 25^\circ C$	-	0.25	-	V
Current consumption	I_{DD}		-	2.0	3.0	mA
	I_{LC}	$V_{LC} = 0.25V$	-	-	1.0	mA
Clock oscillation freq.	f_{osc}	Resistance oscillation	190	270	350	kHz

1.7 Optical Characteristics

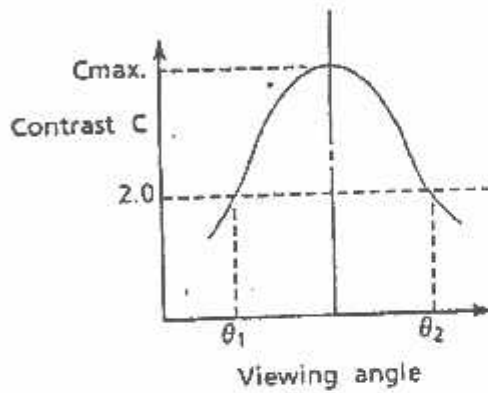
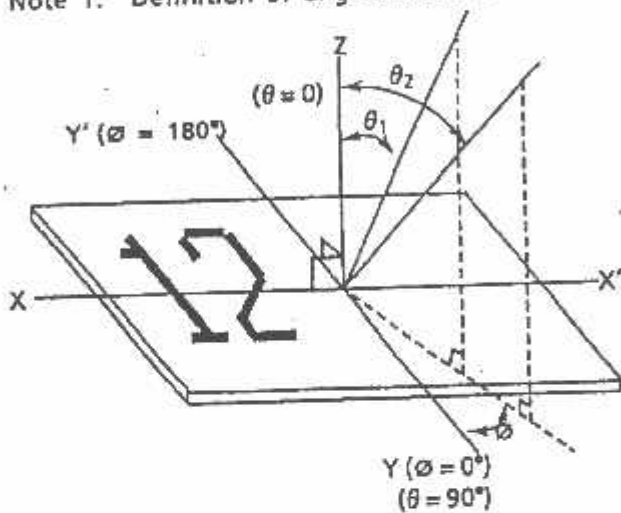
1.7.1 Optical characteristics

Maximum viewing angle: 6 o'clock ($\theta = 0^\circ$)
 $T_A = 25^\circ\text{C}$, $V_{opr} = 4.75\text{ V}$

Item	Symbol	Conditions	Min.	Typ.	Max.	Remarks
Viewing angle	$\theta_2 - \theta_1$	$C \geq 2.0$, $\theta = 0^\circ$	35	-	-	See Notes 1 and 2
Contrast	C	$\theta = 25^\circ$, $\theta = 0^\circ$	5	8	-	See Note 3.
Rise time	t_{on}	$\theta = 25^\circ$, $\theta = 0^\circ$	-	60 ms	70 ms	See Note 4.
Fall time	t_{off}	$\theta = 25^\circ$, $\theta = 0^\circ$	-	150 ms	170 ms	See Note 4.

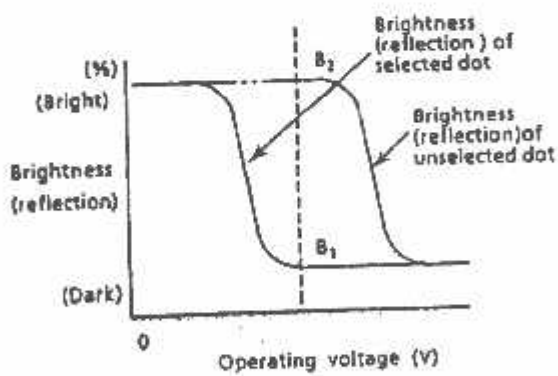
Note 1: Definition of angles θ and θ

Note 2: Definition of viewing angles θ_1 and θ_2

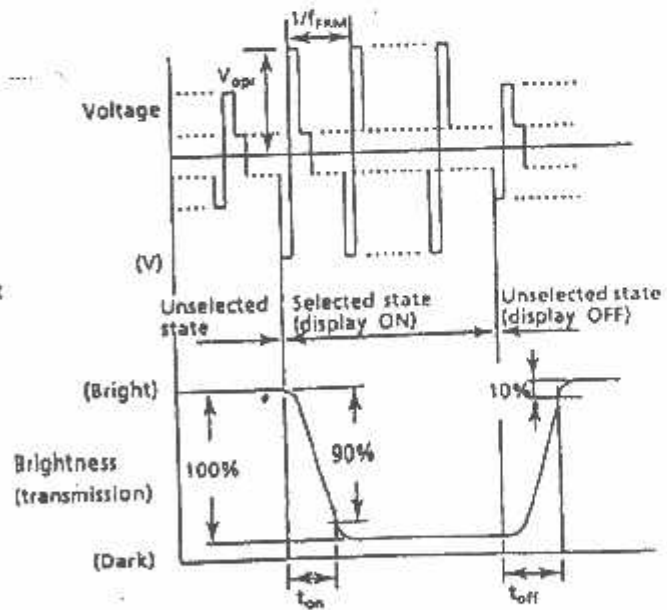


Note 3: Definition of contrast C

$$C = \frac{\text{Brightness (reflection) of unselected dot (B2)}}{\text{Brightness (reflection) of selected dot (B1)}}$$



Note 4: Definition of response time



V_{opr} : Operating voltage (V)
 f_{FRM} : Frame frequency (Hz)
 t_{on} : Response time (rise)(ms)
 t_{off} : Response time (fall)(ms)

1.7.2 Recommended operating voltage

The viewing angle and screen contrast of the LCD panel can be varied by changing the liquid crystal operating voltage (V_{opr}), that is V_{LC} .

The optical characteristics is influenced by an ambient temperature. The recommended value of V_{opr} for an ambient temperatures are shown below.

Temperature (°C)	0	10	25	40	50
Voltage V_{opr} (V)	5.00	4.90	4.75	4.60	4.50

$$V_{opr} = V_{DD} - V_{LC}$$

2. OPERATING INSTRUCTIONS

2.1 Terminal Functions

Table 1 Terminal functions

Signal name	No. of terminals	I/O	Destination	Function
DB ₀ to DB ₃	4	I/O	MPU	Tristate bidirectional lower four data buses: Data is read from the module to the MPU or written to the module from the MPU through the buses. If the interface data is 4 bits, the signals are not used.
DB ₄ to DB ₇	4	I/O	MPU	Tristate bidirectional upper four data buses: Data is read from the module to the MPU or written to the module from the MPU through the buses. DB ₇ is also used as a busy flag.
E	1	Input	MPU	Operation start signal: The signal activates data write or read.
R/W	1	Input	MPU	Read (R) and Write (W) selection signals 0: Write 1: Read
RS	1	Input	MPU	Register selection signals 0: Instruction register (Write) Busy flag and address counter (Read) 1: Data register (Write and Read)
V _{LC}	1	-	Power supply	Power supply terminal for driving liquid crystal display: The screen contrast can be varied by changing V _{LC} .
V _{DD}	1	-	Power supply	+5V
V _{SS}	1	-	Power supply	Ground terminal: 0V

2.2 Basic Operations

2.2.1 Registers

The controller has two kinds of eight-bit registers: the instruction register (IR) and the data register (DR). They are selected by the register select (RS) signal as shown in Table 2.

The IR stores instruction codes such as Display Clear and Cursor Shift, and the address information of display data RAM (DD RAM) and character generator RAM (CG RAM). They can be written from the MPU, but cannot be read to the MPU.

The DR temporarily stores data to be written into DD RAM or CG RAM, or read from DD RAM or CG RAM. When data is written into DD RAM or CG RAM from the MPU, the data in the DR is automatically written into DD RAM or CG RAM by internal operation. However, when data is read from DD RAM or CG RAM, the necessary data address is written into the IR. The specified data is read out to the DR and then the MPU reads it from the DR. After the read operation, the next address is set and DD RAM or CG RAM data at the address is read into the DR for the next read operation.

Table 2 Register selection

RS	R/W	Operation	
0	0	IR selection, IR write.	Internal operation : Display clear
0	1	Busy flag (DB ₇) and address counter (DB ₀ to DB ₆) read	
1	0	DR selection, DR write.	Internal operation : DR to DD RAM or CG RAM
1	1	DR selection, DR read.	Internal operation : DD RAM or CG RAM to DR

2.2.2 Busy flag (BF)

The flag indicates whether the module is ready to accept the next instruction. As shown in Table 2, the signal is output to DB₇ if RS = 0 and R/W = 1. If the value is 1, the module is working internally and the instruction cannot be accepted. If the value is 0, the next instruction can be written. Therefore, the flag status needs to be checked before executing an instruction. If an instruction is executed without checking the flag status, wait for more than the execution time shown by 2.4 Instruction Outline.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Display digit
Line 1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	DD RAM address
Line 2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	

If the display is shifted, DD RAM address 00H to 27H are displayed in line 1 and 40H to 67H in line 2. The following figures are examples of display shifts.

*Left shift

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Display digit
Line 1	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	DD RAM address
Line 2	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	

*Right shift

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Display digit
Line 1	27	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	DD RAM address
Line 2	67	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	

2.2.5 Character generator ROM (CG ROM)

Character generator ROM generates 192 types of 5 x 7 dot-matrix character patterns from eight-bit character codes.

Table 3 shows the correspondence between the CG ROM character codes and character patterns.

2.2.6 Character generator RAM (CG RAM)

CG RAM is used to create character patterns freely by programming. Eight types of character patterns can be written.

Table 4 shows the character patterns created from CG RAM addresses and data. To display a created character pattern, the character code in the left column of the table is written into DD RAM corresponding to the display position (digit). The areas not used for display are available as general data RAM.

Table 3 Correspondence between character codes and character patterns

Upper bit 4 bit Lower bit 4 bit	0 0000	2 0010	3 0011	7 0100	8 0101	6 0110	7 0111	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)		0	A	P	\	Q		→	9	R	0	0
xxxx0001	(2)	!	1	A	Q	a	q	。	7	+	△	△	Q
xxxx0010	(3)	"	2	B	R	b	r	"	4	ウ	×	B	B
xxxx0011	(4)	#	3	C	S	c	s	┘	ウ	7	E	E	e
xxxx0100	(5)	\$	4	D	T	d	t	\	IT	+	+	W	Q
xxxx0101	(6)	%	5	E	U	e	u	。	+	+	1	E	Q
xxxx0110	(7)	&	6	F	V	f	v	ヲ	0	二	B	Q	Σ
xxxx0111	(8)	'	7	G	W	g	w	7	+	7	7	Q	π
xxxx1000	(1)	<	8	H	X	h	x	4	ウ	*	リ	J	π
xxxx1001	(2)	>	9	I	Y	i	y	9	7	リ	W	"	U
xxxx1010	(3)	*	:	J	Z	j	z	。	コ	0	ウ	i	π
xxxx1011	(4)	+	;	K	E	k	e	*	7	E	0	°	π
xxxx1100	(5)	.	<	L	*	l	l	+	E	7	7	+	π
xxxx1101	(6)	---	=	N	I	n	i	。	ア	7	ウ	+	+
xxxx1110	(7)	;	>	N	^	n	+	。	E	+	°	π	
xxxx1111	(8)	/	?	0	...	o	+	ウ	7	°		0	π

Table 4 Relationships between CG RAM addresses and character codes (DD RAM) and character patterns (CG RAM data)

Character code (DD RAM data)		CG RAM address						Character pattern (CG RAM data)																						
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0															
← Upper bit Lower bit →		← Upper bit		Lower bit →				← Upper bit		Lower bit →																				
0 0 0 0 * 0 0 0		0 0 0		0 0 0				* * *										Example of character pattern (R)												
				0 0 0						← Cursor position																				
				0 0 0 0 * 0 0 1		0 0 1				0 0 0				* * *										Example of character pattern (Y)						
										0 0 1						← Cursor position														
										0 0 0 0 * 1 1 1		1 1 1				0 0 0				* * *										
																0 0 1						← Cursor position								

- Notes:
- In CG RAM data, 1 corresponds to Selection and 0 to Non-selection on the display.
 - Character code bits 0 to 2 and CG RAM address bits 3 to 5 correspond with each other (three bits, eight types).
 - CG RAM address bits 0 to 2 specify a line position for a character pattern. Line 8 of a character pattern is the cursor position where the logical sum of the cursor and CG RAM data is displayed. Set the data of line 8 to 0 to display the cursor. If the data is changed to 1, one bit lights, regardless of the cursor.

The character pattern column positions correspond to CG RAM data bits 0 to 4 and bit 4 comes to the left end. CG RAM data bits 5 to 7 are not displayed but can be used as general data RAM.

When reading a character pattern from CG RAM, set to 0 all of character code bits 4 to 7. Bits 0 to 2 determine which pattern will be read out. Since bit 3 is not valid, 00_H and 08_H select the same character.

Timing Characteristics

3.1 Write timing characteristics

 $V_{DD} = 5.0V \pm 5\%$, $V_{SS} = 0V$, $T_A = 0^\circ C$ to $50^\circ C$

Item	Symbol	Standard		Unit	
		Min.	Max.		
Enable cycle time	t_{CYCE}	1000	-	ns	
Enable pulse width	High level	PW_{EH}	450	-	ns
Enable rise and fall time	t_{Er}, t_{Ef}	-	25	ns	
Setup time	RS, $\overline{R/W} - E$	t_{AS}	140	-	ns
Address hold time	t_{AH}	10	-	ns	
Data setup time	t_{DSW}	195	-	ns	
Data hold time	t_H	10	-	ns	

Write operation

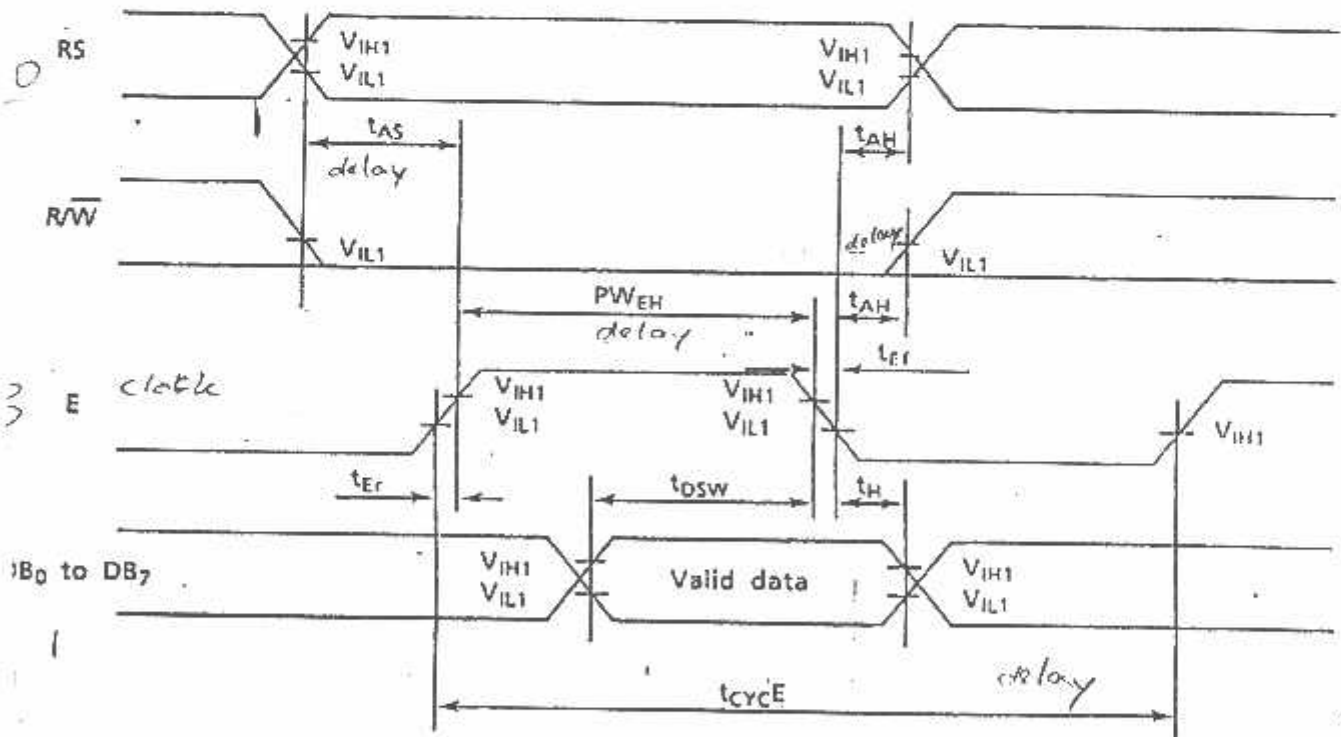


Figure 3 Data write from MPU to module

**03 THRU 2024
GH-VOLTAGE,
GH-CURRENT
ARLINGTON ARRAYS**

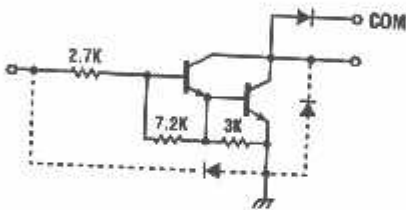
DEVICE PART NUMBER DESIGNATION

$V_{CE(MAX)}$	50 V	95 V
$I_{C(MAX)}$	500 mA	500 mA
Logic	Part Number	
5V TTL, CMOS	ULN2003A* ULN2003L*	ULN2023A* ULN2023L
6-15 V CMOS, PMOS	ULN2004A* ULN2004L*	ULN2024A ULN2024L

* Also available for operation between -40°C and $+85^{\circ}\text{C}$. To order, change prefix from "ULN" to "ULQ".

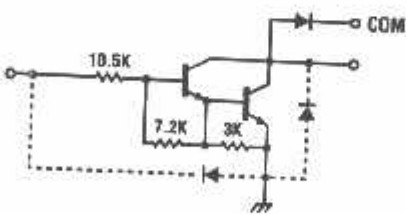
PARTIAL SCHEMATICS

ULN20x3A/L (Each Driver)

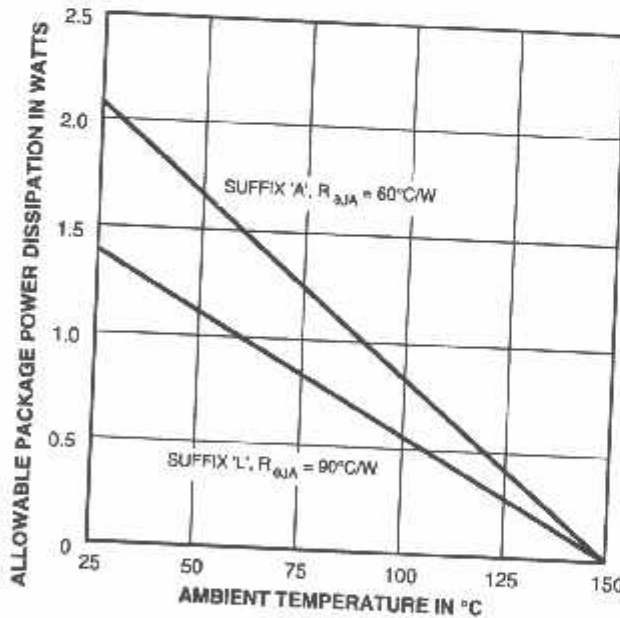


Dwg. No. A-9851

ULN20x4A/L (Each Driver)



Dwg. No. A-9898A



Dwg. GP-005A

X = Digit to identify specific device. Specification shown applies to family of devices with remaining digits as shown. See matrix above.

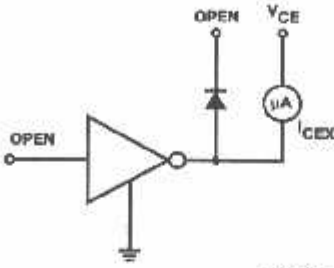


115 Northeast Cutoff, Box 15036
Worcester, Massachusetts 01615-0036 (508) 853-5000
Copyright © 1974, 1998 Allegro MicroSystems, Inc.

2003 THRU 2024
**HIGH-VOLTAGE,
HIGH-CURRENT
DARLINGTON ARRAYS**

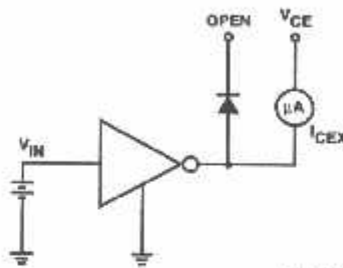
TEST FIGURES

FIGURE 1A



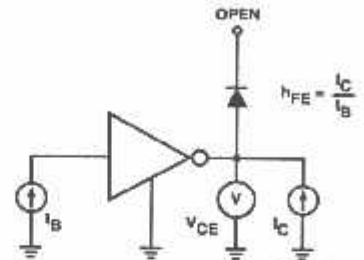
Dwg. No. A-9723A

FIGURE 1B



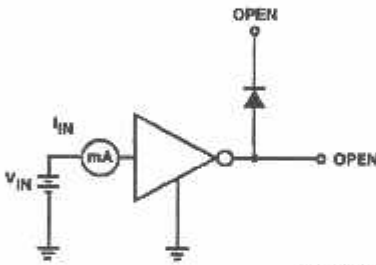
Dwg. No. A-9730A

FIGURE 2



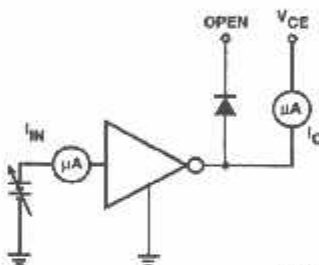
Dwg. No. A-9731A

FIGURE 3



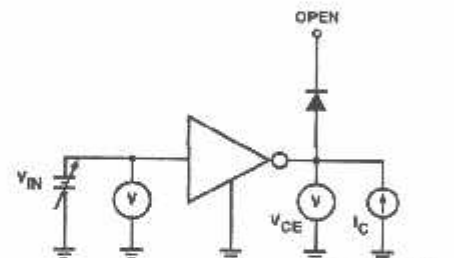
Dwg. No. A-9732A

FIGURE 4



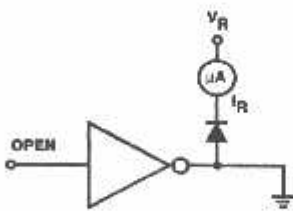
Dwg. No. A-9733A

FIGURE 5



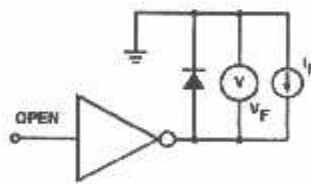
Dwg. No. A-9734A

FIGURE 6



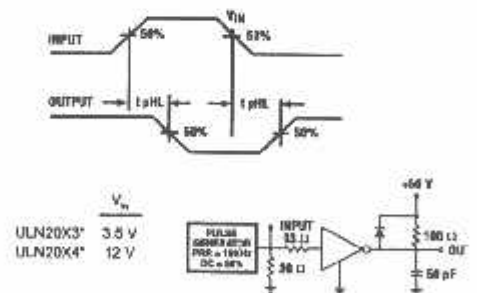
Dwg. No. A-9736A

FIGURE 7



Dwg. No. A-9738A

FIGURE 8

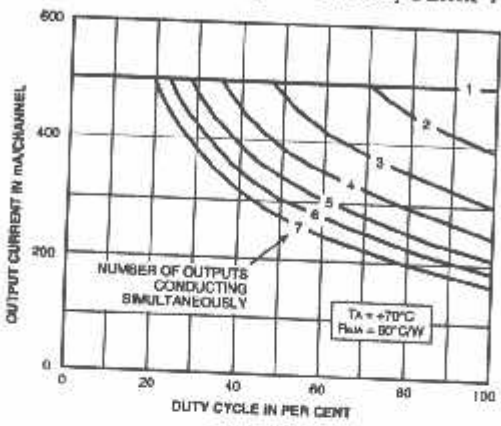


* Complete part number includes a final letter to indicate package.

X = Digit to identify specific device. Specification shown applies to family of devices with remaining digits as shown.

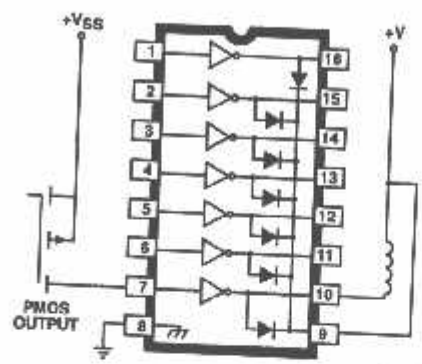
**03 THRU 2024
GH-VOLTAGE,
GH-CURRENT
ARLINGTON ARRAYS**

**ALLOWABLE COLLECTOR CURRENT
AS A FUNCTION OF DUTY CYCLE
(Dual In-line-Packaged Devices, Suffix 'A')**



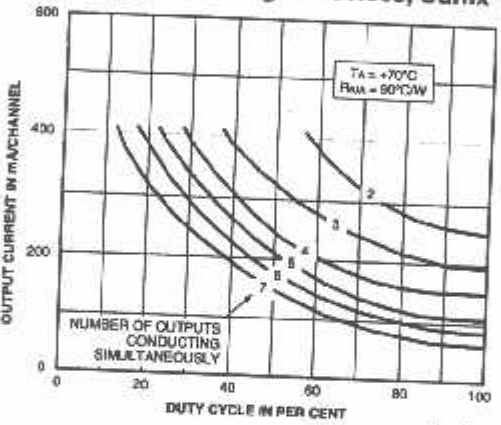
Dwg. 01-47a

TYPICAL APPLICATIONS

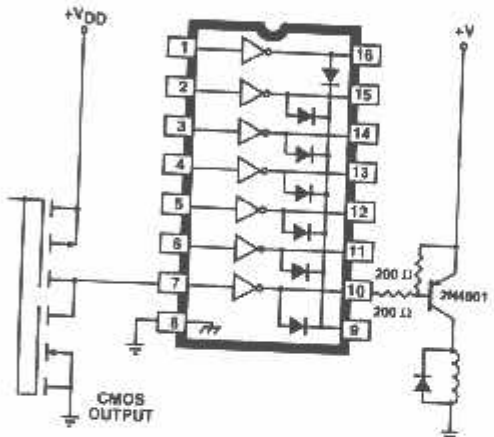


Dwg. No. A-9652

(Small-Outline-Packaged Devices, Suffix 'L')

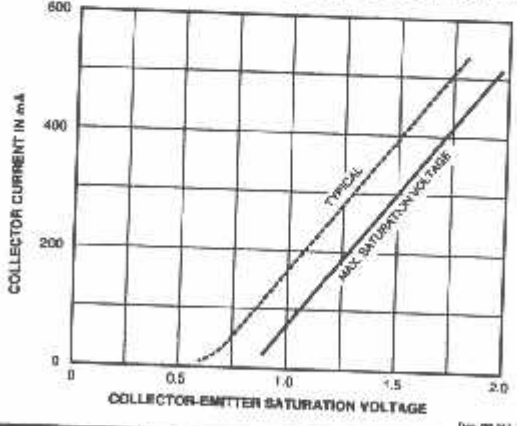


Dwg. 01-47a



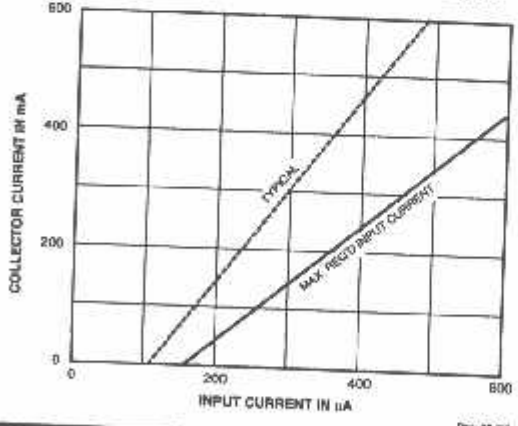
Dwg. No. A-9654A

**SATURATION VOLTAGE
AS A FUNCTION OF COLLECTOR CURRENT**



Dwg. 01-367

**COLLECTOR CURRENT AS A
FUNCTION OF INPUT CURRENT**



Dwg. 01-367

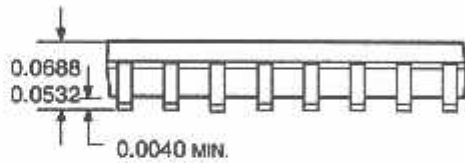
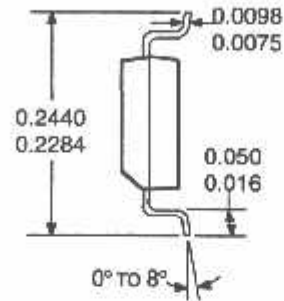
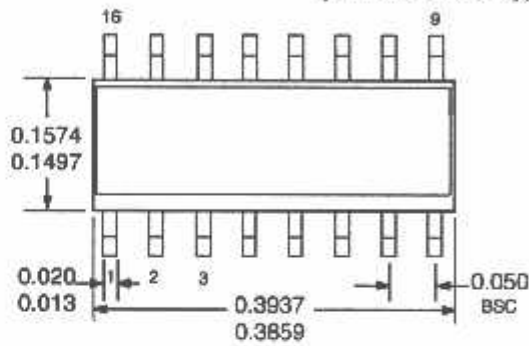


115 Northeast Cutoff, Box 15036
Worcester, Massachusetts 01615-0036 (508) 853-5000

**2003 THRU 2024
HIGH-VOLTAGE,
HIGH-CURRENT
DARLINGTON ARRAYS**

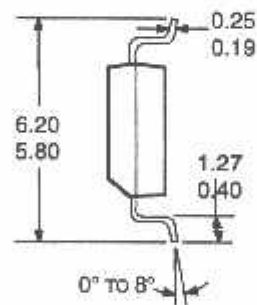
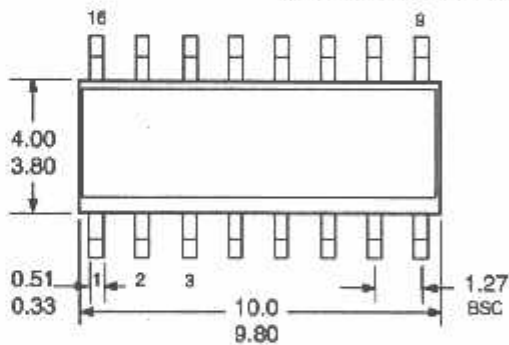
PACKAGE DESIGNATOR "L"

Dimensions in Inches
(for reference only)

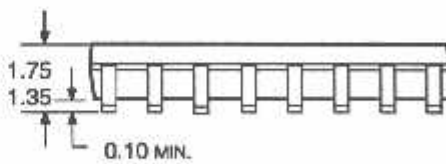


Dep. MA-007-18 in

Dimension in Millimeters
(controlling dimensions)



Dep. MA-007-18A mm



- NOTES: 1. Lead spacing tolerance is non-cumulative.
2. Exact body and lead configuration at vendor's option within limits shown.

