

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



**PERANCANGAN DAN PEMBUATAN ALAT BANTU POLISI DALAM
MENGIDENTIFIKASI KENDARAAN DENGAN MEMANFAATKAN
TEKNOLOGI *RFID***

SKRIPSI

**Disusun Oleh :
Donny Mahendra
NIM. 02.17.150**

SEPTEMBER 2006

LEMBAR PERSETUJUAN



**PERANCANGAN DAN PEMBUATAN ALAT BANTU POLISI DALAM
MENGIDENTIFIKASI KENDARAAN DENGAN MEMANFAATKAN
TEKNOLOGI RFID**

SKRIPSI

*Disusun dan Diajukan untuk Melengkapi dan Memenuhi Syarat Guna
Mencapai Gelar Sarjana Teknik*

Disusun Oleh :

DONNY MAHENDRA

NIM. 02.17.150

Diperiksa dan Disetujui

Ketua Jurusan Elektro S-1

Dosen Pembimbing



(Ir. F. Yudi Limpraptono, MT)

NIP. Y. 130 950 0274

(Ir. Yusuf Ismail Nakhoda, MT)

NIP. Y. 101 880 0189

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

2006



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

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : DONNY MAHENDRA
NIM : 02.17.150
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : Perancangan Dan Pembuatan Alat Bantu Polisi Dalam Mengidentifikasi Kendaraan Dengan Memanfaatkan Teknologi *RFID*

Dipertahankan Dihadapan Tim Penguji Skripsi Jenjang Program Strata Satu (S-1)

Pada :

Hari : Selasa
Tanggal : 26 September 2006
Dengan Nilai : B+ (79,6)



Panitia Ujian Skripsi

Ketua

(Ir.Mochtar Asroni, MSME)
NIP. Y. 101 810 0036

Sekretaris

(Ir.F.Yudi Limpraptono, MT)
NIP.Y. 103 950 0274

Anggota Penguji

Penguji I

(Ir. Widodo Pudi Muljanto, MT)
NIP. Y. 102 870 0171

Penguji II

(M Ashar, ST, MT)

ABSTRAKSI

PERANCANGAN DAN PEMBUATAN ALAT BANTU POLISI DALAM MENGIDENTIFIKASI KENDARAAN DENGAN MEMANFAATKAN TEKNOLOGI *RFID*

(Donny Mahendra, 02.17.150, Teknik Elektronika S.1)
(Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT)

Kata Kunci : Mikrokontroler AT89S8252, *RFID*, *LCD* dan RS-232.

Perkembangan ilmu pengetahuan dan teknologi menawarkan sebuah solusi pengidentifikasian suatu objek dengan memanfaatkan frekuensi radio. Dalam hal ini frekuensi radio diaplikasikan dalam suatu paket yang diberi nama *RFID* (*Radio Frekuensi Identification*). Teknologi *RFID* ini digunakan untuk proses identifikasi kendaraan. *Radio Frequency Identification (RFID)* adalah teknologi *wireless* yang kompak yang berpotensi sangat besar untuk kemajuan perniagaan. *RFID* menggunakan chip yang dapat dideteksi pada range beberapa meter oleh *RFID reader*. *Tag RFID* yang telah diperbaharui mempunyai beberapa keunggulan dibandingkan dengan teknologi identifikasi lainnya dan dapat juga digunakan untuk sistem keamanan.

Pada perancangan dan pembuatan alat ini terdapat tiga blok sistem. Blok pertama yaitu kartu *tag RFID* yang memiliki nomor yang berbeda-beda, blok kedua adalah alat portable yang digunakan untuk mendeteksi *tag RFID* yang terdiri dari mikrokontroler AT89S8252, *reader RFID*, dan *LCD*, dan blok yang ketiga yaitu *PC* yang digunakan sebagai *data base*. Cara kerjanya adalah sebagai berikut setiap pemilik kendaraan harus memiliki kartu *Tag RFID*, kemudian dengan bantuan *reader RFID* beserta mikrokontroler AT89S8252, dan *LCD tag* tersebut dapat diidentifikasi. Untuk menampilkan data pemilik kendaraan berdasarkan kode dari *tag RFID* yang telah teridentifikasi dilakukan dengan bantuan komputer.

Pada tahap pengujian didapatkan hasil yaitu jarak pembacaan *RFID* tanpa penghalang paling baik kurang dari 5 cm, sementara dengan penghalang pembacaan paling baik kurang dari 3 cm kecuali jika penghalang terbuat dari bahan logam. Kemudian pengujian tegangan tiap pin pada mikrokontroler dan max 232 yang dipakai didapatkan hasil bahwa tegangan tidak mengalami perubahan yang *signifikan* baik pada saat *standby* maupun terdeteksi.

KATA PENGANTAR

Atas berkat rahmat Allah SWT, sehingga penulis dapat menyelesaikan laporan skripsi dengan judul :

“PERANCANGAN DAN PEMBUATAN ALAT BANTU POLISI DALAM MENGIDENTIFIKASI KENDARAAN DENGAN MEMANFAATKAN TEKNOLOGI *RFID*”

Pembuatan skripsi ini disusun guna memenuhi nsyarat akhir kelulusan pendidikan jenjang strata-1 di Institut Teknologi Nasional Malang. Laporan skripsi ini merupakan tanggung jawab tertulis atas ilmu pengetahuan yang didapat selama penyusun mengikuti kuliah.

Atas terselesainya skripsi ini, penulis mengucapkan terima kasih kepada :

- Bapak Prof. Dr. Ir. Abrahan Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
- Bapak Ir. Mochtar Asroni, MSME, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
- Bapak Ir. Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro S-1.
- Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Dosen Pembimbing.
- Ibu Ir. Mimien Mustikawati, selaku Sekretaris Jurusan Teknik Elektro S-1.
- Serta pihak-pihak yang tidak dapat disebutkan satu per satu yang telah banyak membantu dalam penerjaan skripsi ini sehingga dapat terselesaikan.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan.

Oleh sebab itu kritik dan saran yang bersifat membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, September 2006

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
BERITA ACARA UJIAN SKRIPSI.....	iii
ABSTRAKSI.....	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	3
1.4. Tujuan.....	3
1.5. Metodologi	4
1.6. Sistematika Penulisan.....	4
BAB II LANDASAN TEORI	6
2.1. <i>RFID (Radio Frekuensi Identifikasi)</i>	6
2.2. Mikrokontroler ATMEL AT89S8252.....	11
2.2.1. Fitur Mikrokontroler AT89S8252.....	12
2.2.2. Konfigurasi Pin Pada Mikrokontroler AT89S8252	14

2.2.3. SFR Tambahan Pada Mikrokontroler Atmel At89S8252	16
2.2.3.1. SFR Untuk <i>Timer 2</i>	17
2.2.3.2. SFR Untuk <i>Watchdog</i> dan Memori	20
2.2.3.3. SFR Pengontrol SPI	21
2.2.4. Organisasi Memori	26
2.2.4.1. Program Memori Internal	26
2.2.4.2. Data Memori RAM Internal	26
2.2.5. <i>Programmable Watchdog Timer (WDT)</i>	27
2.2.6. <i>Timer 2</i>	28
2.2.6.1. <i>Mode Capture</i>	30
2.2.6.2. <i>Auto Reload (Up/Down Counter)</i>	30
2.2.6.3. <i>Baud Rate Generator</i>	32
2.2.6.4. <i>Programmable Clock Out</i>	33
2.3. LCD (<i>Liquid Cristal Display</i>)	34
2.4. Komunikasi Data Serial	37
2.4.1. Pengaturan <i>Baud Rate</i> Port Serial	37
2.4.2. Komunikasi Serial Antara PC Dengan Mikrokontroler	37
2.4.3. <i>Interface Unit RS-232</i>	38

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1. Perancangan Dan Pembuatan Perangkat Keras	41
3.1.1. Perancangan Dan Pembuatan Diagram Blok Sistem	41
3.1.1.1. Keterangan Masing-masing Blok	41
3.1.1.2. Prinsip Kerja Dari Sistem	42

3.1.2. Perancangan dan Pembuatan Rangkaian <i>RFID</i>	43
3.1.2.1. Frekuensi Kerja <i>RFID</i>	44
3.1.3. Perancangan Dan Pembuatan Mikorokontroller Sebagai Minimum Sistem	45
3.1.3.1. Perancanagn Rangkaian <i>Clock</i>	46
3.1.3.2. Perancangan Rangkaian <i>Reset</i>	47
3.1.4. Perancangan Dan Pembuatan Tampilan (<i>LCD</i>)	49
3.1.5. Perancangan dan Pembuatan Antar Muka Dengan <i>PC</i>	51
3.2. Perancangan Perangkat Lunak	51
3.2.1. Mikrokontroller	51
3.2.2. Program Utama Pada <i>PC</i>	53
 BAB IV PENGUJIAN ALAT	55
4.1. Pengujian <i>RFID</i>	55
4.2. Pengujian Pin-pin Mikrokontroller	58
4.3. Pengujian Pin-pin RS-232.....	59
4.4. Pengujian Keseluruhan Sistem.....	60
 BAB V PENUTUP	62
5.1. Kesimpulan.....	62
5.2. Saran-saran	62
 DAFTAR PUSTAKA	
 LAMPIRAN-LAMPIRAN	

DAFTAR GAMBAR

BAB II LANDASAN TEORI

Gambar 2.1. Komunikasi Antara <i>Reader</i> Dengan <i>Transponder (Tag)</i>	7
Gambar 2.2. Output Format ASCII	9
Gambar 2.3. Konfigurasi Pin ID-10	10
Gambar 2.4. Diagram Blok At89S8252	11
Gambar 2.5. Konfigurasi Pin AT89S8252	14
Gambar 2.6. Koneksi SPI <i>Master</i> Dan <i>Slave</i>	22
Gambar 2.7. Format <i>Transfer</i> SPI Dengan CPHA=0.....	23
Gambar 2.8. Format <i>Transfer</i> SPI Dengan CPHA=1.....	23
Gambar 2.9. Timer 2 Pada <i>Mode Capture</i>	30
Gambar 2.10. Timer 2 Pada <i>Mode Auto Reload</i> DCEN=0 (<i>Atas</i>), DCEN=1 (<i>Bawah</i>)	32
Gambar 2.11. Timer 2 Sebagai <i>Baud Rate</i> Generator	33
Gambar 2.12. Timer 2 Dalam <i>Clock Out Mode</i>	34
Gambar 2.13. Bentuk Fisik Dari <i>LCD</i>	35
Gambar 2.14. Level Logika Standart RS-232	49
Gambar 2.15. Konektor DB-9	39

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Gambar 3.1. Diagram Blok Sistem	41
Gambar 3.2. Rangkaian <i>RFID Reader</i>	44
Gambar 3.3. Mikrokontroler Sebagai Minimum Sistem.....	45

Gambar 3.4. Rangkaian <i>Clock</i>	47
Gambar 3.5. Rangkaian <i>Reset</i>	48
Gambar 3.6. Rangkaian Tampilan <i>LCD</i>	50
Gambar 3.7. Rangkaian Komunikasi Serial ke <i>PC</i>	51
Gambar 3.8. Flowchart Program Mikrokontroler	52
Gambar 3.9. Flowchart Program Pada Komputer	53

BAB IV PENGUJIAN ALAT

Gambar 4.1. Kotak Dialog <i>Connection Description</i>	56
Gambar 4.2. Kotak Dialog <i>connect to</i>	56
Gambar 4.3. Kotak Dialog <i>COM1 Properties</i>	57
Gambar 4.4. Identifikasi <i>Reader</i> Terhadap Kartu	57
Gambar 4.5. Tampilan Awal Kode <i>RFID</i> Pada <i>PC</i>	61
Gambar 4.6. Data Base data Pemilik Kendaraan	61

DAFTAR TABEL

BAB II LANDASAN TEORI

Tabel 2-1 Fungsi Pin Dan Format Data	10
Tabel 2-2 Fungsi Khusus Pada <i>Port 1</i> AT89S8252	14
Tabel 2-3 Pemilihan Periode Waktu <i>watchdog timer</i>	28
Tabel 2-4 Mode Operasi Timer 2	30
Tabel 2-5 Definisi Pin-pin LCD	36
Tabel 2-6 <i>Baud Rate Mode Serial</i>	37
Tabel 2-7 Fungsi Pin RS-232 Dalam DB-9	40

BAB IV PENGUJIAN ALAT

Tabel 4-1 Hasil Pengujian Pembacaan <i>RFID</i>	58
Tabel 4-2 Hasil Pengujian Pin-pin Mikrokontroller	59
Tabel 4-3 Hasil Pengujian Pin-pin RS-232	60

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi (IPTEK) dewasa ini sangat pesat, utamanya teknologi elektronika. Dalam perkembangannya elektronika sendiri sekarang ini telah mengalami kemajuan yang sangat pesat, hal tersebut dapat dibuktikan dengan adanya perkembangan di berbagai bidang terutama teknologi identifikasi dengan menggunakan frekuensi radio..

Seiring dengan padatnya penduduk dan semakin banyak pula kendaraan pribadi, tingkat kriminalitas pun semakin merajalela. Misalnya kasus pencurian kendaraan yang sering terjadi sekarang ini. Para pelaku pencurian bukan orang sembarangan, mereka sering kali mengganti dan memalsukan surat-surat kendaraan serta memodifikasi kembali kendaraan hasil curiannya tersebut, sehingga pemilik atau pihak kepolisian sulit untuk mengidentifikasi kendaraan tersebut, selain hal tersebut pada saat heregistrasi kendaraan sering kali pihak kepolisian harus melakukan pengecekan nomor mesin dan rangka dengan cara menggosok-gosok nomor yang ada di mesin kendaraan. Hal ini dirasa kurang efektif dan efisien.

Oleh karena hal itulah penulis mencoba untuk menawarkan sebuah solusi yaitu dengan memanfaatkan teknologi *RFID* untuk proses identifikasi kendaraan. *Radio Frequency Identification (RFID)* adalah teknologi *wireless* yang kompak yang berpotensi sangat besar untuk kemajuan

perniagaan. *RFID* menggunakan chip yang dapat dideteksi pada range beberapa meter oleh *RFID reader*. *Tag RFID* yang telah diperbaharui mempunyai beberapa keunggulan dibandingkan dengan teknologi identifikasi lainnya dan dapat juga digunakan untuk sistem keamanan. Caranya sangat mudah yaitu setiap pemilik kendaraan harus memiliki kartu *Tag RFID*, kemudian dengan bantuan *reader RFID* beserta mikrokontroller AT89S8252, dan *LCD tag* tersebut dapat diidentifikasi. Untuk menampilkan data pemilik kendaraan berdasarkan kode dari *tag RFID* yang telah teridentifikasi dilakukan dengan bantuan komputer. Dengan dirancangnya sistem ini diharapkan pihak kepolisian tidak kesulitan untuk mengidentifikasi kendaraan dan kriminalitas dapat sedikit ditekan.

1.2. Rumusan Masalah

Dari latar belakang yang telah dikemukakan di atas, maka rumusan masalah pada perancangan ini adalah sebagai berikut :

1. Bagaimana merancang dan membuat alat untuk mengidentifikasi kendaraan dengan memanfaatkan teknologi *RFID*?
2. Bagaimana menyusun perangkat lunak/*software* yang mendukung *hardware* secara baik?

Sehubungan dengan permasalahan tersebut di atas, maka dalam skripsi ini dipilih judul :

**“PERANCANGAN DAN PEMBUATAN ALAT BANTU POLISI
DALAM MENGIDENTIFIKASI KENDARAAN DENGAN
MEMANFAATKAN TEKNOLOGI *RFID*”**

1.3. Batasan Masalah

Agar permasalahan yang dibahas tidak terlalu meluas maka perlunya pembatasan permasalahan, adapun batasan masalahnya meliputi :

1. Data yang muncul pada komputer berupa kode kartu, nama pemilik kendaraan, alamat pemilik, nomor rangka, bahan bakar, kode lokasi, nomor polisi, nomor mesin, jenis kendaraan, warna kendaraan, nomor urut pendaftaran, dan tanggal jatuh tempo masa berlaku STNK.
2. Tidak membahas catu daya, dan frekuensi-frekuensi radio.
3. Mikrokontroller yang digunakan adalah AT89S8252, *RFID Reader* yang digunakan adalah ID-10, dan *LCD* yang digunakan adalah tipe M1632.
4. Menggunakan RS 232 untuk *interface* antara komputer dengan mikrokontroller dan tidak membahas protokoler data secara mendetail.
5. Kapasitas EEPROM yang digunakan adalah sebesar 40 lokasi memori.

1.4. Tujuan

Tujuan dari perancangan dan pembuatan alat ini adalah merancang dan membuat suatu alat yang memanfaatkan teknologi *RFID* sebagai proses identifikasi kendaraan.

1.5. Metodologi

Metodologi penelitian yang dipakai dalam pembuatan tugas akhir ini adalah :

1. *Study* Literatur

Metode yang digunakan diawali dengan melakukan studi literatur terhadap data-data yang diperlukan guna menunjang kelancaran didalam proses pembuatan alat. Studi ini akan dilaksanakan dengan cara mencari buku-buku yang menjadi landasan dari tiap teori maupun mencari data lewat sarana internet. Kemudian data ini akan diproses untuk mencari yang sesuai dengan data yang kita kehendaki.

2. Perancangan Dan Pembuatan Alat

Melaksanakan perancangan dan pembuatan alat baik *hardware* maupun *software* sesuai dengan rancangan yang telah disusun.

3. Pelaksanaan Uji Coba Alat

Pengujian alat dilakukan saat mencapai tahap akhir untuk menemukan kesalahan atau kekurangan pada alat tersebut untuk kemudian dilakukan perbaikan.

4. Penyusunan Laporan Tugas Akhir

Bertujuan untuk penyusunan data laporan berpedoman pada alat yang selesai dibuat beserta cara kerja alat.

1.6. Sistematika Penulisan

Sistematika Penulisan dalam tugas akhir ini sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, batasan masalah, tujuan, metodologi, dan sistematika penulisan.

BAB II Dasar Teori

Membahas teori-teori yang menunjang dalam perancangan dan perealisasiian alat.

BAB III Perancangan Dan Pembuatan Alat

Menjelaskan tentang perancangan perangkat elektronika dan perancangan perangkat lunak yang diperlukan dalam perealisasiian alat.

BAB IV Pengujian Dan Analisis

Memuat hasil pengujian terhadap alat yang telah direalisasiikan.

BAB V Penutup

Berisi kesimpulan perancangan ini dan saran-saran yang perlu untuk melakukan pengembangan aplikasi selanjutnya.



BAB II

LANDASAN TEORI

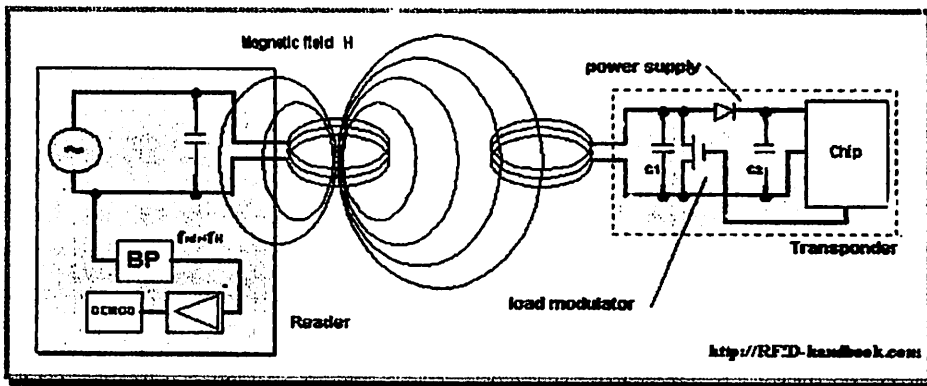
Pada bab ini akan dibahas mengenai dasar-dasar teori yang dapat menunjang dalam perancangan dan pembuatan alat bantu polisi dalam mengidentifikasi kendaraan dengan menggunakan teknologi *RFID*. Teori penunjang ini akan membahas komponen dan peralatan pendukung pada alat yang dibuat. Uraian teori dalam bab ini meliputi :

- *RFID* dengan ID-10 sebagai *RFID reader*.
- Mikrokontroler AT89S8252
- *Liquid Crystal Display* (LCD) M1632
- RS-232 Sebagai Komunikasi Serial Antara PC Dengan Mikrokontroler.

2.1. *RFID (Radio Frekuensi Identification)*

RFID adalah proses identifikasi seseorang atau objek dengan menggunakan frekuensi transmisi radio. *RFID* menggunakan frekuensi radio untuk membaca informasi dari sebuah devais kecil yang disebut *tag* atau *transponder (Transmitter + Responder)*. *Tag RFID* akan mengirimkan informasi yang ada di dalam dirinya pada saat berada di dalam gelombang elektromagnetik yang dipancarkan devais yang kompatibel yaitu pembaca *RFID (RFID Reader)*. Gelombang elektromagnetik tersebut digunakan sebagai catu daya untuk membangkitkan transmitter sehingga informasi di dalam *tag* dapat terbaca. Jarak pembacaan antara *tag* dengan *reader* berkisar

± 3 cm (tergantung jenis *tag* yang digunakan) serta bekerja pada frekuensi 125 khz seperti terlihat pada gambar berikut ini :



Gambar 2.1
Komunikasi Antara *Reader* Dan *Transponder (Tag)*^[1]

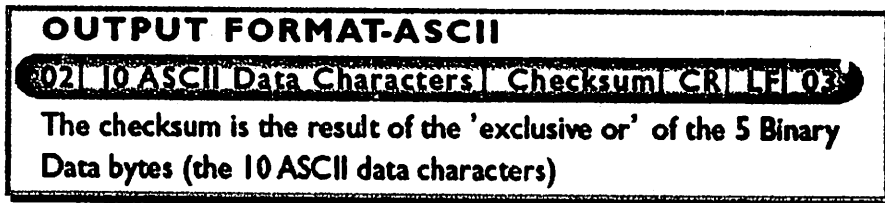
RFID dapat disediakan dalam devais yang hanya dapat dibaca saja (*Read Only*) atau dapat dibaca dan ditulis (*Read/Write*), tidak memerlukan kontak langsung maupun jalur cahaya untuk dapat beroperasi, dapat berfungsi pada berbagai variasi kondisi lingkungan. Karena teknologi ini sulit untuk dipalsukan, maka *RFID* dapat menyediakan tingkat keamanan yang tinggi.

Pada sistem *RFID* umumnya, *tag* atau *transponder* ditempelkan pada suatu objek. Setiap *tag* dapat membawa informasi yang unik, diantaranya : serial *number*, warna, tempat perakitan, dan data lain dari objek tersebut. Ketika *tag* melewati medan yang dihasilkan oleh *RFID reader* yang kompatibel, *tag* akan menstransmisikan informasi yang ada pada *tag* kepada *reader*, sehingga proses identifikasi objek dapat dilakukan.

Sistem *RFID* terdiri dari empat komponen, diantaranya seperti dapat lihat pada gambar 2.1, dengan penjelasan sebagai berikut :

- *Tag* : ini adalah devais kecil yang menyimpan informasi untuk identifikasi objek. *Tag RFID* sering juga disebut sebagai *transponder*. Format dari *tag* pada perancangan ini adalah EM4001 atau *tag* kompatibel lainnya.
- Antena : untuk mentransmisikan sinyal frekuensi radio antara pembaca *RFID* dengan *tag RFID*.
- Pembaca *RFID* : adalah devais kecil yang kompatibel dengan *tag RFID* yang akan berkomunikasi secara wireless dengan tag. Digunakan tipe ID-10 sebagai *RFID reader* pada perancangan ini.
- Software Aplikasi : adalah aplikasi pada sebuah *workstation* atau PC yang dapat membaca data dari *tag* melalui *RFID reader*. Baik *tag* maupun *RFID reader* dilengkapi dengan antenna sehingga dapat menerima dan memancarkan gelombang elektromagnetik.

Tipe *RFID reader* yang digunakan dalam alat ini adalah ID-10. *RFID reader* ini memiliki dua bentuk output serial yaitu : *ASCII* dan *Wiegand 26-bit*. Pada perancangan alat ini digunakan output dengan format *ASCII*, karena output ini sangat mudah untuk dihubungkan pada mikrokontroler. Output yang memiliki format *ASCII* memiliki struktur sebagai berikut :



Gambar 2.2
Output Format ASCII^[2]

Checksum merupakan hasil *EXOR (Exclusive OR)* dari 5 biner data *byte*. Untuk lebih jelasnya tentang cara pembacaan format *ASCII*, lihat contoh berikut ini.

Misalnya data *output* serial (dalam *hexadesimal*) yang kita tangkap adalah sebagai berikut:

02	30	34	36	32	30	31	44	37	36	43	44	43	0D	0A	03
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

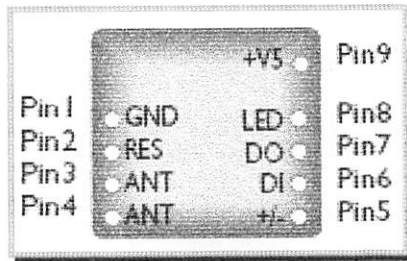
Langkah pertama adalah merubah semua nilai data diatas menjadi karakter *ASCII*. Misalnya 30H menjadi karakter "0", 34H menjadi karakter "4", dst. Langkah kedua adalah menyusun data – data tersebut ke dalam Format data *ASCII* seperti gambar 2.2. Kemudian ambil 10 data karakter *ASCII*. Dalam contoh ini berarti data tersebut adalah:

30	34	36	32	30	31	44	37	36	43	Data Hexa
		6	2	0	1	D	7	6	C	Data ASCII

Untuk data dengan warna biru merupakan data untuk jenis – jenis kartu dan tidak digunakan dalam proses konversi, yang akan dipakai disini adalah data yang ke 3 s/d 10. Hasil konversi dari data heksa ke dalam data *ASCII* adalah "6201D76C". Gabungkan karakter data *ASCII* menjadi bilangan *Hexadesimal*, kemudian konversikan bilangan *hexadesimal* tersebut ke dalam desimal. Hasilnya sebagai berikut: 6201D76C H menjadi

1644287852 (ini merupakan nomor kartu sebenarnya yang tertera pada badan kartu tersebut). Cara ini hanya berlaku pada kartu yang tidak dienkripsi.

RFID reader memiliki 9 pin seperti terlihat pada gambar berikut ini :



Gambar 2.3
Konfigurasi Pin ID-10 (*RFID Reader*)^[2]

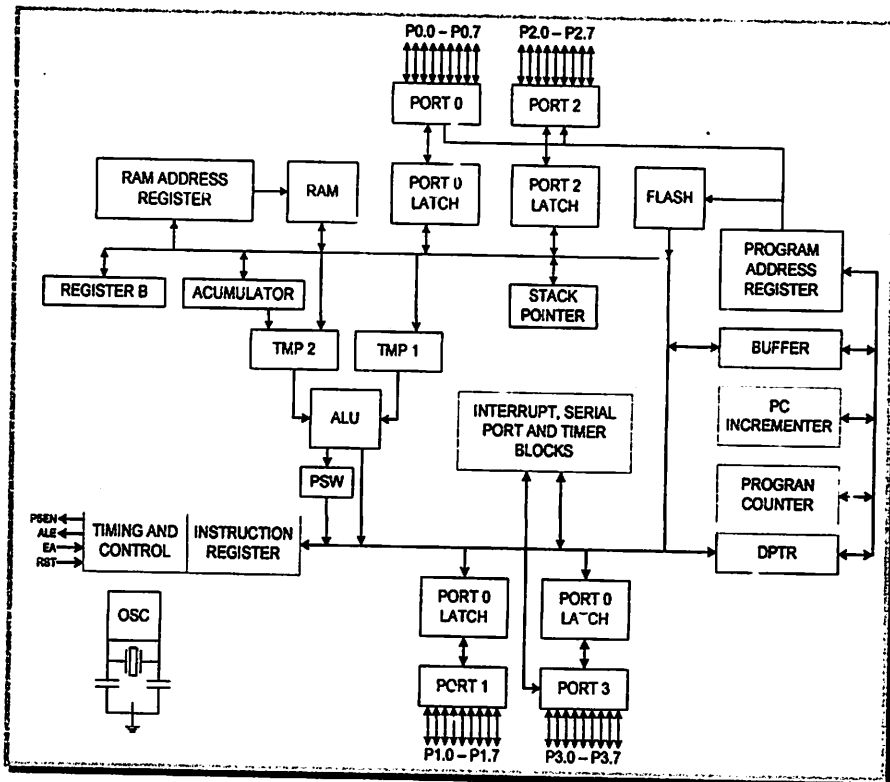
Fungsi masing-masing pin dan format data dapat dilihat pada tabel berikut ini :

Tabel 2-1
Fungsi Pin Dan Format Data^[2]

Pin	ASCII		WIEGAND 26	
Pin 1	Ground 0V	Zero Volts and Tunning Capacitor Ground	Ground 0V	Zero Volts and Tunning Capacitor Ground
Pin 2	Reset Bar	Strap to +5V	Reset Bar	Strap to +5V
Pin 3	Antenna	NC	Antenna	NC
Pin 4	Antenna	NC	Antenna	NC
Pin 5	Strap to Ground		Strap to Pin +5V	
Pin 6	CMOS	Serial ASCII	One Output	
Pin 7	TTL Data	Serial ASCII Inverted	Zero Output	
Pin 8	Beep/LED	2.7Khz Logic	Beep/LED	2.7Khz Logic
Pin 9	+4.6 Through to +5.5V	Supply DC Volts	+4.6 Through to +5.5V	Supply DC Volts

2.2. Mikrokontroler ATMEL AT89S8252

Mikrokontroler Atmel AT89S8252 merupakan pengembangan dari mikrokontroler standar MCS-51. Hal-hal yang terdapat pada penjelasan mikrokontroler MCS-51 juga berlaku untuk mikrokontroler Atmel AT89S8252. Hanya mikrokontroler Atmel AT89S8252 memiliki beberapa kelebihan bila dibandingkan dengan mikrokontroler MCS-51. Karena adanya fitur tambahan yang tidak terdapat pada mikrokontroler MCS-51, maka mikrokontroler Atmel AT89S8252 dapat menggantikan mikrokontroler MCS-51, tetapi tidak demikian sebaliknya. AT89S8252 merupakan sebuah mikrokontroler 8-bit CMOS, *Low Power* dengan 8 Kbyte *Flash Programmable and Erasable Read Only Memory (PEROM)*.



Gambar 2.4
Diagram Blok AT89S8252^[3]

2.2.1. Fitur Mikrokontroler Atmel AT89S8252

IC ini dibuat sesuai dengan standar industri konfigurasi pin dan *instruction set* dari MCS-8252. Arsitektur dan kelengkapan Mikrokontroler AT89S8252 adalah sebagai berikut:

- a. Kompatibel dengan mikrokontroler MCS-51
- b. 8K byte *Downloadable Flash Memory*
- c. 2K byte EEPROM
- d. 3 level *program memory lock*
- e. 256 byte RAM *internal*
- f. 32 I/O yang dapat dipakai semua
- g. 3 buah *Timer/Counter* 16 bit
- h. *Programmable* UART (*serial port*)
- i. *SPI Serial Interface*
- j. *Programmable Watchdog Timer*
- k. *Dual Data Pointer*
- l. Frekuensi kerja 0 sampai 24 MHz
- m. Tegangan operasi 2,7 Volt sampai 6 Volt
- n. CPU (*Central Processing Unit*) 8-bit dengan register A (*Accumulator*) dan B
- o. 16-bit *Program Counter* (PC) dan *Data Pointer* (DPTR)
- p. 8-bit *Program Status Word* (PSW)
- q. 4-bit *Stack Pointer* (SP)
- r. 4 *bank register*, masing-masing berisi 8 register
 - 16 *byte* yang dapat dialamati pada bit *Level*

- 80 byte *general purpose Memory data*
- s. *Receiver Register*, yaitu: TCON, TMOP, SCON, IP dan IE
- t. 5 buah sumber *interrupt* (2 buah sumber *interrupt eksternal* dan 3 buah sumber *interrupt internal*)
- u. *Oscillator* dan *Clock*

Terlihat bahwa mikrokontroler Atmel AT89S8252 memiliki banyak fitur yang menguntungkan. Dipakainya *Downloadable Flash Memory* memungkinkan mikrokontroler ini bekerja sendiri tanpa diperlukan tambahan chip lainnya. Sementara *Flash Memory*nya mampu diprogram hingga seribu kali. Hal lain yang menguntungkan adalah sistem pemrograman menjadi lebih sederhana dan tidak memerlukan rangkaian yang rumit seperti rangkaian untuk memprogram produk Atmel lainnya yaitu AT89C51.

Timer/Counter juga bertambah satu dari standar 2 buah pada MCS-51. Selain itu frekuensi kerja yang lebar dan rancangan statik sangat membantu untuk proses *debugging*. Dengan adanya beberapa fitur tambahan itu, maka akan mengakibatkan bertambahnya SFR (*Special Function Register*).

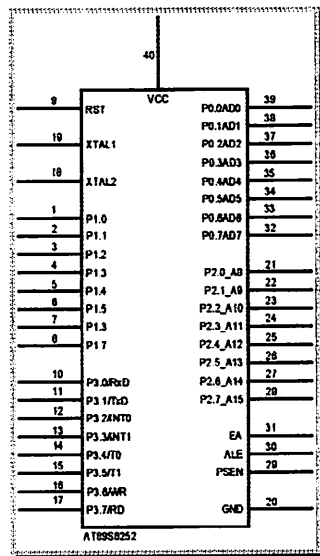
Semua pin pada mikrokontroler Atmel AT89S8252 adalah sama dengan mikrokontroler MCS-51. Namun pada port 1 mikrokontroler Atmel AT89S8252 terdapat beberapa fungsi khusus yang tidak terdapat pada mikrokontroler MCS-51. Fungsi khusus tersebut dijelaskan pada tabel berikut:

Tabel 2 – 2
Fungsi Khusus pada Port 1 AT89S8252^[3]

Port Pin	Fungsi Khusus
P1.0	T2 (Masukan luar untuk <i>Timer/Counter 2</i>)
P1.1	T2 EX (<i>Timer/Counter 2 capture/reload trigger dan control arah</i>)
P1.2	-
P1.3	-
P1.4	SS (<i>Slave port select input</i>)
P1.5	MOSI (<i>Master data output, Slave data input untuk kanal SPI</i>)
P1.6	MISO (<i>Master data input, Slave data Output untuk kanal SPI</i>)
P1.7	SCK (<i>Master clock output, Slave clock input untuk kanal SPI</i>)

2.2.2. Konfigurasi Pin pada mikrokontroler AT89S8252

Konfigurasi kaki-kaki AT89S8252 terdiri dari 40 pin, seperti pada gambar 2 – 18 berikut:



Gambar 2.5
Konfigurasi Pin-Pin AT89S8252^[3]

Fungsi dari tiap-tiap pin adalah sebagai berikut:

1. VCC (*Supply tegangan*)

2. GND (*ground*)

3. Port 0

Merupakan *port input-output* dua arah dan dikonfigurasi sebagai *multiplex* dua bus alamat rendah (A0-A7) dan data selama pelaksanaan *program Memory* dan data *Memory Internal*.

4. Port 1

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

5. Port 2

Merupakan *port input-output* dua arah dengan *Internal pull-up*, mengeluarkan *address* tinggi selama pengambilan (*fetch*) program *memory eksternal* dan selama pelaksanaan ke data *Memory Port 2* mengeluarkan isi P2PSFR (*Special Function Register*) menerima *address* tinggi dan beberapa sinyal kontrol selama pemrograman dan verifikasi.

6. Port 3

Merupakan *port input-output* dua arah dengan *Internal pull-up*, yang juga memiliki fungsi khusus, yaitu:

- RXD (P3.0) : *Port Input Serial*
- TXD (P3.1) : *Port Output Serial*
- INT0 (P3.2) : *Interrupt 0 eksternal*
- INT1 (P3.3) : *Interrupt 1 eksternal*
- T0 (P3.4) : *Input external Timer 0*
- T1 (P3.5) : *Input external Timer 1*
- WR (P3.6) : *Strobe tulis data Memory eksternal*
- RD (P3.7) : *Strobe tulis data Memory eksternal*

7. RST

Input Reset

8. ALE / PROG

Pulsa output ALE digunakan untuk proses “*latching*” *Byte address* rendah (A0-A7) selama pengaksesan ke *external Memory*. Pin ini juga untuk memasukkan pulsa program selama pemrograman. Pada operasi normal ALE mengeluarkan rate konstan yaitu 16 frekuensi osilasi dan boleh digunakan untuk *timing external*.

9. PSEN

Merupakan strobe baca ke program *Memory External*.

10. EA / VPP

External address enable EA digroundkan jika mengakses *Memory External*. Untuk mengakses *Memory Internal*, maka dihubungkan ke VCC.

11. XTAL 1 dan XTAL 2

Kaki ini dihubungkan dengan kristal bila menggunakan *oscillator Internal*. XTAL 1 merupakan *input inverting oscillator amplifier*, sedangkan dengan XTAL 2 merupakan *output inverting oscillator amplifier*.

2.2.3. SFR tambahan pada mikrokontroler Atmel AT89S8252

Selain memiliki SFR seperti halnya pada mikrokontroler MCS-51, mikrokontroler Atmel AT89S8252 memiliki tambahan SFR. Hal ini tak lain adalah karena terdapatnya tambahan fitur pada mikrokontroler ini. Jadi SFR

tambahan ini adalah SFR untuk mengontrol alat tambahan pada mikrokontroler Atmel AT89S8252.

SFR tambahan ini meliputi:

- T2CON (*Timer 2 Register*), dengan alamat 0C8H
- T2MOD (*Timer 2 Mode*), dengan alamat 0C9H
- WMCON (*Watchdog and Memory Control Register*), dengan alamat 96H
- SPCR (*SPI Control Register*), dengan alamat D5H
- SPSR (*SPI Status Register*), dengan alamat AAH
- SPDR (*SPI Data Register*), dengan alamat 86H

2.2.3.1. SFR untuk Timer 2

Pada mikrokontroler Atmel AT89S8252 terdapat tambahan sebuah *Timer/Counter* yang diberi nama *Timer 2* (sehingga mikrokontroler Atmel AT89S8252 memiliki 3 *Timer/Counter* yaitu *Timer/Counter 0*, *Timer/Counter 1*, dan *Timer/Counter 2*).

Pada *Timer/Counter 2* ini dikendalikan oleh *special function register* yang bernama T2CON (*Timer 2 Control*), T2MOD (*Timer 2 Mode*), dan sepasang register RCAP2H, RCAP2L merupakan register *capture/reload* untuk *Timer 2* dalam 16 bit *capture mode* atau *auto reload mode*.

Register T2CON yang beralamat di 0C8H memiliki bit-bit sebagai berikut:

MSB							LSB
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2

Bit	Keterangan
TF2	Bendera <i>overflow Timer 2</i> , diset oleh <i>Timer 2</i> dan harus di- <i>clear</i> lewat <i>software</i>
EXF2	Bendera luar <i>Timer 2</i> diset saat suatu <i>capture</i> atau <i>reload</i> disebabkan oleh transisi negatif pada T2EX dan EXEN2=1. Jika sela <i>Timer 2</i> diaktifkan, EXF2 =1 akan menyebabkan CPU mencabang ke rutin sela <i>Timer 2</i> . EXF2 harus di- <i>clear</i> dengan <i>software</i> . EXF2 tidak menyebabkan sela pada <i>mode up/down counter</i> (DCFN=1).
RCLK	<i>Receive clock enable</i> . Jika diset menyebabkan serial port menggunakan pulsa <i>overflow Timer 2</i> sebagai detak penerimaan pada serial port untuk mode 1 dan 3. jika RCLK=0 menyebabkan pulsa <i>overflow Timer 1</i> yang digunakan sebagai detak.
TCLK	<i>Transmit clock enable</i> . Jia diset menyebabkan serial port menggunakan pulsa <i>overflow Timer 2</i> sebagai detak pengiriman. Jika TCLK=0 menyebabkan pulsa <i>overflow Timer 1</i> yang digunakan sebagai detak pengiriman.
EXEN2	<i>Timer 2 eksternal enable</i> . Jika diset memungkinkan <i>capture</i> atau <i>reload</i> terjadi sebagai hasil dari transisi negative pada pin T2EX jika <i>Timer 2</i> tidak sedang digunakan sebagai <i>baud rate generator</i> untuk serial port. Jika EXEN2=0 menyebabkan <i>Timer 2</i> akan melakukan apa-apa kejadian pada pin T2EX.

- TR2** Bit untuk mengatur *start/stop* untuk *Timer 2*. Jika TR2=1 *Timer* akan aktif.
- C/T2** Bit pemilihan *Timer* atau *Counter* untuk *Timer 2*. Jika C/T2=0 maka terpilih fungsi *timer*. C/T2=1 untuk fungsi *counter*.
- CP/RL2** Pemilihan *Capture/Reload*. Jika diset maka proses *capture* akan terjadi pada transisi negatif pada pin T2EX jika bit EXEN2=1. Jika bit ini di-clear maka proses *reload* otomatis akan terjadi saat *Timer 2* overflow atau transisi negatif terjadi pada pin T2EX saat bit EXEN2=1. Jika bit RCLK atau TCLK diset maka bit ini menjadi tidak diprhitungkan (*ignore*). Hal ini karena *Timer 2* dipakai sebagai *baud rate generator* pada serial port.

SFR ini memiliki nilai pada saat reset : 0000 0000B.

Timer 2 juga memiliki SFR yang bernama T2MOD (*Timer 2 Mode Control Register*) yang beralamat di 0C9H dan memiliki nilai pada saat reset XXXX XX00B. Bit-bit pada T2MOD adalah sebagai berikut :

MSB		LSB
-	- - - - -	T2OE DCEN

Bit	Keterangan
------------	-------------------

T2OE	<i>Timer 2 Output Enable</i> bit
-------------	----------------------------------

DCEN	Jika diset memungkinkan <i>Timer/Counter</i> sebagai <i>up/down counter</i> .
-------------	---

2.2.3.2. SFR untuk Watchdog dan Memori

Untuk menggunakan watchdog timer atau memori, maka dapat dilakukan dengan mengatur SFR yang bernama WMCON dengan alamat 96H. Bit-bit pada SFR ini dapat dijelaskan sebagai berikut :

Nilai SFR ini pada saat reset adalah 0000 0000B

MSB		LSB					
PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTTRST	WDTEN
Bit	Keterangan						
PS2	-						
PS1	-						
PS0	Ketiga bit ini merupakan bit prescaler untuk watchdog timer. Jika ketiga bit ini di-clear maka periode watchdog timer adalah 16 ms. Jika ketiga bit di atas di-set maka nominal periode waktu watchdog timer adalah 2048 ms.						
EEMWE	Bit pengaktif penulisan EEPROM Data Memori. Bit ini harus di-set sebelum penulisan ke EEPROM dengan instruksi MOVX. Setelah selesai penulisan maka bit ini harus di-clear.						
EEMEN	Bit pengaktif pengaksesan internal EEPROM. Saat EEMEN=1 instruksi MOVX dengan DPTR akan mengakses internal EEPROM bukan pada data memori luar. Jika EEMEN=0 instruksi MOVX dengan DPTR akan mengakses data memori luar.						

- DPS** *Data Pointer Register Select*. Jika bit ini di-clear akan memilih bank pertama dari *Data Pointer Register* (DP0). Jika bit ini di-set akan terpilih bank kedua (DP1).
- WDTRST** Watchdog Timer Reset dan bit bendera EEPROM *ready/busy*. Tiap saat bit ini di-set ke 1 oleh software pengguna, suatu pulsa akan dihasilkan untuk me-reset watchdog timer. Bit ini kemudian secara otomatis akan di-clear. Bit ini bersifat hanya dapat ditulisi. Bit ini juga sebagai bit bendera *ready/busy* pada *mode Read Only* selama penulisan EEPROM. RDY/BSY=1 berarti bahwa EEPROM siap untuk deprogram. Selama operasi pemrograman berlangsung, bit ini akan berlogika '0' dan secara otomatis akan di-reset ke '1' saat pemrograman selesai.
- WDTEN** Bit pengaktif Watchdog Timer. Jika WDTEN=1 akan mengaktifkan Watchdog Timer, jika WDTEN=0 akan melumpuhkannya.

2.2.3.3. SFR Pengontrol SPI

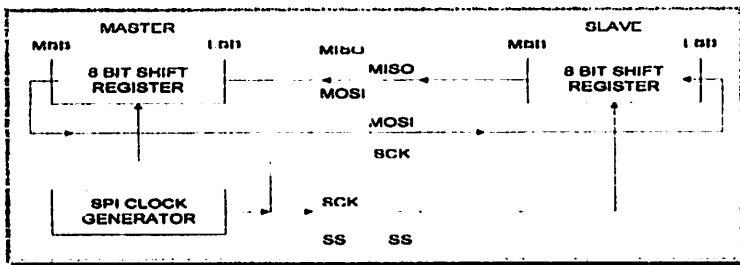
AT89S8252 memiliki fasilitas SPI (*Serial Peripheral Interface*) yang memungkinkan transfer data kecepatan tinggi secara sinkron mikrokontroler Atmel AT89S8252 dengan peripheral atau antar AT89S8252.

Fitur SPI pada AT89S8252 meliputi :

- a. *Full Duplex*. 3 kawat dengan transfer data secara sinkron
- b. Operasi Master atau Slave
- c. Frekuensi maksimum 6 MHz
- d. Sistem data transfer MSB dahulu atau LSB dahulu

- e. 4 bit rate terprogram
- f. Bendera sela pada akhir transmisi
- g. *Write Collision Flag Protection*
- h. Bangun dari mode *idle* (hanya untuk mode slave)

Gambar berikut menunjukkan hubungan antara CPU master dan slave :



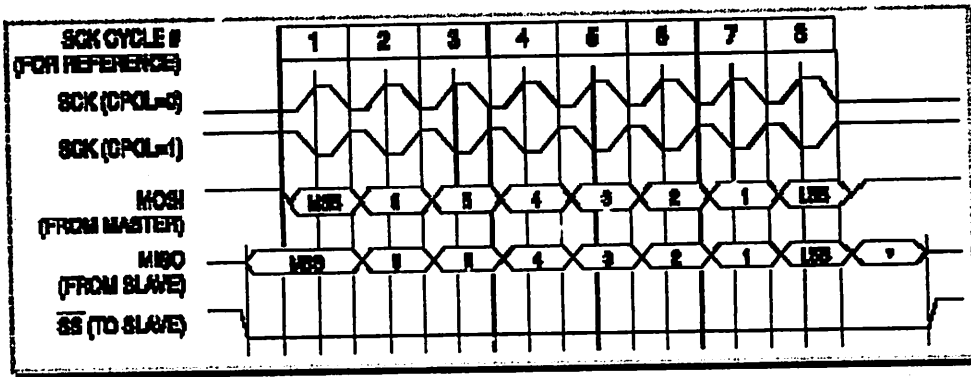
Gambar 2.6
Koneksi SPI Master dan Slave^[3]

Pin SCK adalah keluaran detak pada mode master, tetapi merupakan detak masukan pada mode slave. Menulis ke SPI data register pada CPU master akan memulai SPI *clock generator*, dan data yang ditulis digeser keluar pada pin MOSI dan menuju pin MOSI pada CPU slave.

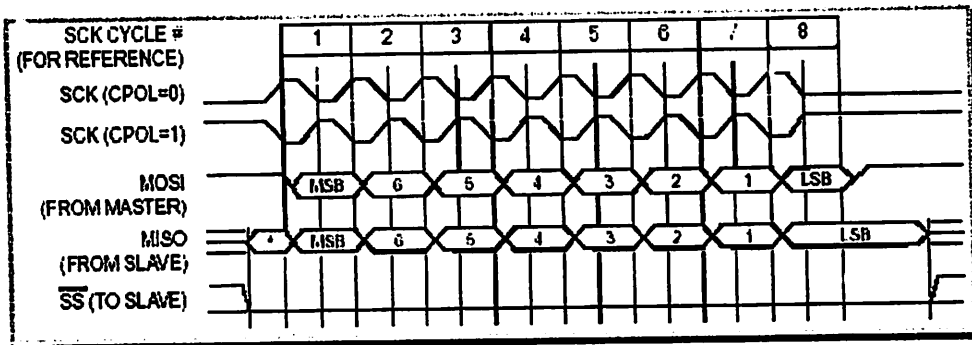
Setelah menggeser 1 byte, SPI *clock generator* akan berhenti, dan akan mengaktifkan bendera (*flag*) selesai pengiriman (SPIF). Jika kedua bit pengaktif serial port (ES) di-set, suatu sela akan dibutuhkan.

Bit pemilih *slave select* (SS) pada port 1 pin 4 (P1.4) dibuat rendah untuk memilih suatu alat SPI individual sebagai slave. Jika pena ini tinggi, maka port SPI tidak diaktifkan dan pin MOSI (P1.6) dapat digunakan sebagai masukan.

Berikut adalah format transfer SPI dengan CPHA= 0 dan CPHA= 1 ;



Gambar 2.7
Format Transfer SPI dengan CPHA = 0^[3]



Gambar 2.8
Format Transfer SPI dengan CPHA = 1^[3]

Sedangkan *Special Function Register* untuk mengontrol penggunaan SPI adalah SPCR (*SPI Control Register* dengan alamat D5H) dan SPSR (*SPI Status Register* dengan alamat AAH). Bit-bit pada SFR SPCR dijelaskan sebagai berikut:

MSB							LSB
SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
Bit	Keterangan						

SPIE Bit pengaktif sela SPI. Bit ini berhubungan dengan bit ES pada register IE, memungkinkan pengaktifan sela SPI. Jika SPIE=1 dan ES=1 akan mengaktifkan sela SPI. SPIE=0 melumpuhkan sela SPI.

SPE Bit pengaktif SPI. Jika SPE=1 mengaktifkan kanal SPI dan menghubungkan SS, MOSI, MISO, dan SCK ke pin P1.4, P1.5, P1.6, dan P1.7. Jika SPE=0 akan melumpuhkan kanal SPI.

DORD Data Order. Jika DORD=1 akan memilih LSB terlebih dahulu untuk dikirimkan. Jika DORD=0 akan memilih MSB terlebih dahulu untuk dikirimkan.

MSTR Bit pemilih fungsi Master atau Slave. Jika MSTR=1 akan memilih mode Master. Jika MSTR=0 akan memilih mode Slave.

CPOL Polaritas detak. Jika CPOL=1, SCK akan tinggi saat kondisi idle. Jika CPOL=0, SCK pada alat Master akan rendah jika tidak sedang mengirimkan data.

CPHA Fasa detak. Bit ini bersama dengan bit CPOL mengontrol hubungan antara detak dengan data antara Master dan Slave.

SPR1 SPI Clock Rate Select. Kedua bit ini mengontrol rate SCK pada alat yang dikonfigurasi sebagai Master.

SPR0 SPR1 dan SPR0 tidak berefek pada slave. Hubungan antara SPR1, SPR0, dan SCK adalah sebagai berikut :

SPR1	SPR0	SCK=F osc dibagi dengan
0	0	4
0	1	16
1	0	64
1	1	128

Pada saat reset, register ini akan bernilai 0000 01XXB. (X=*don't care*)

Register SPSR (SPI Status Register) yang beralamat di AAH dan memiliki nilai reset 0000 0000B mempunyai bit-bit sebagai berikut :

MSB		LSB
SPIF	WCOL - - - - - - -	

Bit	Keterangan
SPIF	Bendèra sela SPI. Jika suatu pengiriman secara serial telah selesai, bit SPIF akan di-set dan suatu sela akan dibangkitkan jika bit SPIE=1 dan bit ES=1. Bit SPIF di-clear dengan membaca register status SPI dengan SPIF dan WCOL di-set, dan kemudian mengakses data register SPI.
WCOL	<i>Write Collision Flag</i> . Bit WCOL akan di-set jika SPI data register sedang menulis saat transfer data. Selama transfer data, hasil pembacaan register SPDR mungkin akan salah, dan penulisan kepadanya tidak berefek.

Register terakhir adalah register SPDR (SPI Data Register) dengan alamat 86H dan memiliki nilai pada saat reset yang tidak berubah. Bit-bit pada register ini adalah sebagai berikut :

MSB		LSB
SPD7	SPD6 SPD5 SPD4 SPD3 SPD2 SPD1	SPD0

AT89S8252 mempunyai dua buah *Power-Saving Mode* yang dapat diatur melalui *software*, yaitu: IDE Mode yang akan menghentikan CPU sebagai RAM, *Timer/Counter*, *Serial Port* dan *Interrupt system* yang tetap berfungsi. *Power Down Mode* ini yang akan menyimpan di RAM dan akan

menahan *Oscillator* untuk tidak mengaktifkan *chip* yang lain sampai terjadi reset secara *hardware*.

2.2.4. Organisasi Memori

Dalam IC AT89S8252 ruang alamat telah dibedakan untuk program *Memory* dan data *Memory*.

2.2.4.1. Program Memori Internal

AT89S8252 memiliki program Memori Internal sebesar 8 Kbyte dengan alamat 0000H-0FFFH. Jika alamat-alamat program lebih tinggi daripada 0FFFH, yang melebihi kapasitas ROM/*Flash Memory Internal* menyebabkan AT89S8252 secara otomatis mengambil *Code Byte* dari program Memori External. *Code Byte* juga dapat diambil hanya dari *External Memory* dengan alamat 0000H-FFFFH dengan cara menghubungkan Pin EA ke *Ground*.

AT89S8252 juga dilengkapi dengan data memori yang berupa EEPROM (*Electrically Erasable Programmable Read Only Memory*). EEPROM yang ditanamkan ini besarnya 2 Kbyte dan dipakai untuk menyimpan data.

EEPROM on chip ini diakses dengan mengeset bit EEMEN pada register WMCON pada alamat 96H. alamat EEPROM ini adalah C0H-7FFH. Instruksi MOV X digunakan untuk mengakses EEPROM internal ini.

2.2.4.2. Data Memori RAM Internal

RAM yang ada pada mikrokontroler AT89S8252 adalah berkapasitas 256 byte, terbagi atas tiga daerah, yaitu:

- Empat Bank Register

Setiap bank terdiri dari 8 register (R0-R7), sehingga jumlah register untuk keempat bank register (bank 0-bank 3) menjadi 32 buah register yang menempati alamat 00H-1FH. Mengaktifkan salah satu bank register yang dapat dilakukan dengan mengatur RS0-RS1 pada PSW (*Program Status Word*)

- *Bit Addressable*

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

- *General Purpose*

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

2.2.5. Programmable Watchdog Timer (WDT)

Pada Mikrokontroler AT89S8252 juga dilengkapi dengan *Watchdog Timer*. *Watchdog Timer* ini menggunakan detak tersendiri. Untuk mengatur rentang waktu (periode) pada WDT ini maka terdapat prescaler yang dapat mengatur rentang waktu yang dibutuhkan.

Bit prescaler ini adalah bit PS0, PS1, dan PS2 pada register WMCON. Periode waktu pada WDT ini berkisar antara 16 ms sampai 2048 ms. Karena bit prescaler-nya ada tiga, maka akan ada 8 buah kemungkinan seperti yang tertera pada tabel di bawah ini :

Tabel 2-3
Pemilihan Periode Waktu Watchdog Timer^[3]

PS2	PS1	PS0	PERIODA
0	0	0	16 ms
0	0	1	32 ms
0	1	0	64 ms
0	1	1	128 ms
1	0	0	256 ms
1	0	1	512 ms
1	1	0	1024 ms
1	1	1	2048 ms

Watchdog Timer (WDT) dilumpuhkan oleh *Power on Reset* (POR) dan selama *Power Down*. WDT diaktifkan dengan men-setting bit WDTEN pada SFR WMCON (alamat 96H). Jika perhitungan waktu WDT telah selesai (*time out*) tanpa ada reset atau dilumpuhkan, maka suatu pulsa reset internal akan dihasilkan untuk mereset CPU.

2.2.6. Timer 2

Pada mikrokontroler Atmel AT89S8252 terdapat tambahan Timer 2. *Timer/Counter* dapat digunakan sebagai *generator baud rate* untuk serial port. Pada standar MCS-51 biasanya digunakan Timer 1 sebagai penghasil baud rate. Pada AT89S8252 selain dapat menggunakan Timer 1 sebagai penghasil baud rate (untuk menjaga kompatibilitas dengan MC-51) juga dapat menggunakan Timer 2 sebagai penghasil baud rate untuk serial port.

Timer 2 ini merupakan *Timer/Counter* yang berukuran 16 bit yang dapat beroperasi sebagai Timer (dengan detak dari sistem detak mikrokontroler) atau dapat beroperasi sebagai penghitung kejadian (*event counter*) dengan detak dari luar. Untuk mengatur fungsi ini dilakukan

dengan mengatur bit C/T2 pada SFR T2CON. Jika bit ini tinggi maka akan terpilih fungsi counter (C), tetapi jika bit ini rendah maka akan terpilih fungsi Timer 2.

Timer 2 ini memiliki 3 mode operasi, yaitu *capture*, *auto reload (up dan down counting)*, dan *baud rate generator*. Untuk memilih mode ini dilakukan dengan mengatur bit pada SFR T2CON (Timer 2 Control Register).

Timer 2 terdiri dari 2 buah timer 8 bit register yaitu TH2 dan TL2. Pada fungsi timer, register TL2 dinaikkan (*increment*) tiap siklus mesin. Karena siklus mesin terdiri dari 12 periode osilasi, maka *count rate* menjadi $\frac{1}{12}$ dari frekuensi osilator.

Pada fungsi counter, register dinaikkan berdasarkan tanggapan adanya transisi tinggi ke rendah pada pena yang bersesuaian (dalam hal ini pin T2 atau P1.0). Pada fungsi ini, masukan luar akan disampling selama S5P2 dari tiap siklus mesin. Jika hasil *sampling* menunjukkan logika tinggi pada selama satu siklus dan rendah pada siklus selanjutnya, maka akan terdeteksi transisi dari tinggi ke rendah dan akibatnya penghitungan akan dinaikkan. Nilai penghitungan yang baru akan muncul pada register selama S3P1 dari siklus setelah transisi tinggi ke rendah terdeteksi.

Tabel berikut menunjukkan mode operasi yang dapat dijalankan Timer 2 :

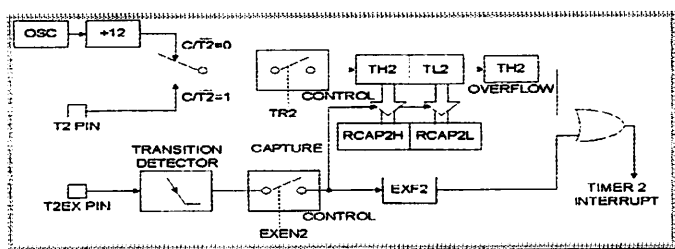
Tabel 2-4
Mode Operasi Timer 2^[3]

RCLK+TCLK	CP/RL2	TR2	MODE
0	0	1	16 bit <i>auto reload</i>
0	1	1	16 bit <i>capture</i>
1	X	1	<i>baud rate generator</i>
X	X	0	off

2.2.6.1. Mode Capture

Pada mode ini dua pilihan dipilih oleh bit EXEN2 pada SFR T2CON. Jika EXEN2=0, Timer 2 merupakan 16 bit timer atau counter yang jika telah overflow (melimpah) akan mengeset bit TF2 pada T2CON.

Bit ini kemudian dapat digunakan untuk menghasilkan sela. Jika EXEN2=1, Timer 2 akan berlaku sama, tetapi suatu transisi tinggi ke rendah (1 to 0) pada pin T2EX (P1.1) akan menyebabkan nilai sekarang pada TH2 dan TL2 untuk ditangkap dan disimpan ke RCAP2H dan RCAP2L. sebagai tambahan transisi tinggi ke rendah pada T2EX menyebabkan bit EXF2 pada T2CON diset. Bit EXF2 sama halnya dengan bit TF2 dapat menghasilkan sela.



Gambar 2.9
Timer 2 pada Mode Capture^[3]

2.2.6.2. Auto Reload (Up/Down Counter)

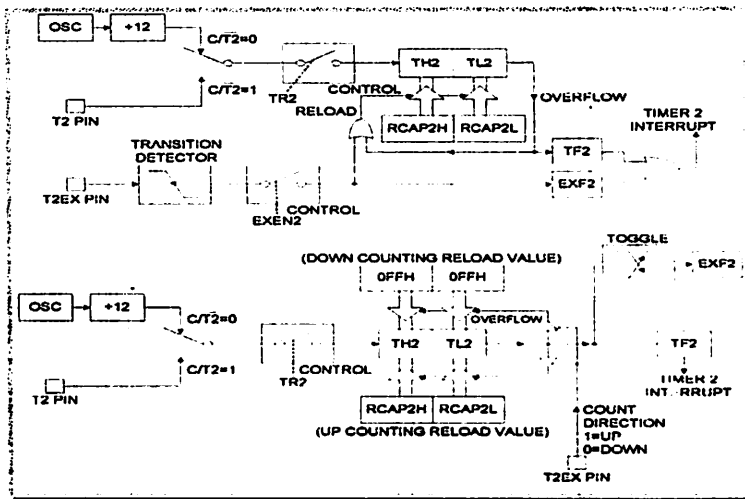
Timer 2 dapat diprogram untuk menghitung naik (maju) atau menghitung mundur jika dikonfigurasi sebagai mode 16 bit auto reload.

Fitur ini dapat dimatikan dengan mengatur bit DCEN (*Down Counter Enable*) pada SFR T2MOD. Pada saat reset DCEN akan berlogika rendah maka Timer 2 akan memiliki default untuk menghitung maju (*Up Counter*). Jika bit DCEN diset, Timer 2 dapat menghitung maju atau mundur tergantung pada nilai logika pada pin T2EX.

Pada gambar berikut Timer 2 secara otomatis menghitung naik (maju) pada saat DCEN=0. Pada mode ini, 2 pilihan dipilih oleh bit EXEN2 pada SFR T2CON. Jika EXEN2=0, Timer 2 akan naik menjadi 0FFFFH dan kemudian akan mengeset bit TF2 jika telah melimpah (*overflow*). *Overflow* juga menyebabkan terjadinya *register timer* diisi kembali dengan nilai 16 bit dari RCAP2H dan RCAP2L. Jika EXEN2=1 suatu 16 bit *reload* dapat diaktifkan (di-triger) baik oleh *overflow* atau oleh transisi tinggi ke rendah pada pin T2EX. Transisi ini juga akan mengeset bit EXF2. Kedua bit TF2 dan EXF2 dapat menimbulkan sela jika diaktifkan.

Mengeset bit DCEN akan mengaktifkan Timer 2 untuk menghitung naik atau mundur. Pada mode ini pin T2EX akan mengontrol arahnya (maju/mundur). Suatu logika '1' pada pin T2EX membuat Timer 2 menghitung maju (*count up*). Timer akan melimpah pada hitungan 0FFFFH dan mengeset bit TF2. *Overflow* ini juga menyebabkan nilai 16 bit pada RCAP2H dan RCAP2L diisikan kembali (*reloaded*) ke register timer yaitu TH2 dan TL2. Suatu logika '0' pada pin T2EX membuat Timer 2 menghitung maju (*count down*). Timer akan mengalami *underflow* saat TH2 dan TL2 sama dengan nilai yang tersimpan pada RCAP2H dan RCAP2L.

underflow ini akan mengeset bit TF2 dan menyebabkan 0FFFFH akan diisikan kembali (reloaded) ke timer register.



Gambar 2.10
 Timer 2 pada Mode *Auto Reload*
 DCEN=0 (atas), DCEN=1 (bawah)^[3]

2.2.6.3. Baud Rate Generator

Timer 2 dapat dipilih sebagai *baud rate generator* dengan menseting TCLK dan RCLK pada SFR T2CON. *Baud rate* untuk pengiriman dan penerimaan dapat berbeda jika Timer 2 digunakan untuk penerimaan atau pengiriman, sementara Timer 1 digunakan untuk tugas lain.

Baud rate generator secara teknis sama dengan mode *auto reload*, dimana pelimpahan TH2 menyebabkan register Timer 2 diisi kembali dengan nilai 16 bit pada register RCAP2H dan RCAP2L yang telah diisi (*preset*) oleh program pemakai.

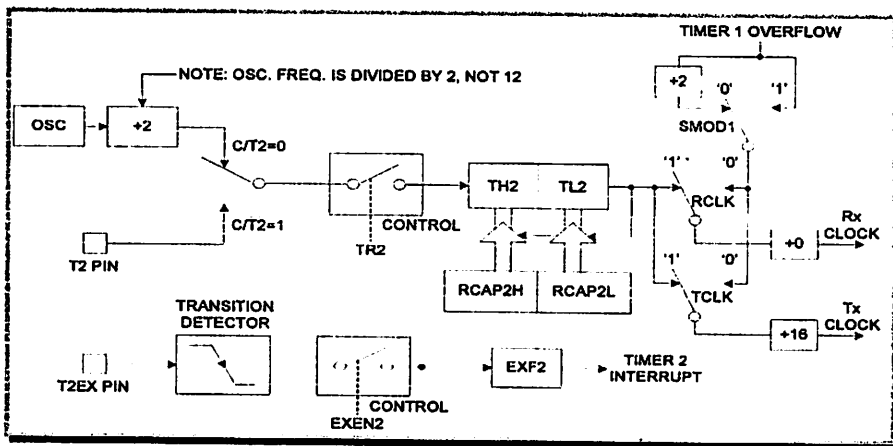
Baud rate pada mode 1 dan 3 ditentukan oleh *rate overflow Timer* berdasarkan persamaan :

$$\text{Baud Rate [mode 1,3]} = \frac{\text{Timer2OverflowRate}}{16}$$

Timer dapat dikonfigurasi sebagai operasi timer atau counter. Pada kebanyakan pemakaian adalah difungsikan sebagai timer dimana bit CP/T2 dibuat '0'. Operasi timer adalah berbeda jika Timer 2 difungsikan sebagai baud rate generator. Secara normal sebagai timer, akan naik setiap siklus mesin (pada 1/12 frekuensi osilator). Sebagai *baud rate generator* akan dinaikkan tiap *state time* (pada 1/2 frekuensi osilator). Rumus untuk baud rate adalah sebagai berikut :

$$\text{Baud Rate [mode 1,3]} = \frac{\text{Frekuensi Osilator}}{32 \times [65536 - (\text{RCAP2H}, \text{RCAP2L})]}$$

Dimana (RCAP2H,RCAP2L) adalah isi dari register RCAP2H dan RCAP2L yang diambil sebagai 16 bit integer tak bertanda.



Gambar 2.11
Timer 2 sebagai *Baud Rate Generator*^[3]

2.2.6.4. Programmable Clock Out

Fungsi terakhir dari Timer 2 adalah untuk menghasilkan suatu detak/pulsa (*clock*). Detak dengan siklus tugas 50% dapat diprogramkan sehingga keluar dari P1.0. Pin ini selain sebagai pin I/O biasa juga memiliki dua fungsi alternative. Dapat diprogram sebagai pin masukan untuk *eksternal clock Timer/Counter 2* atau sebagai penghasil clock dengan siklus

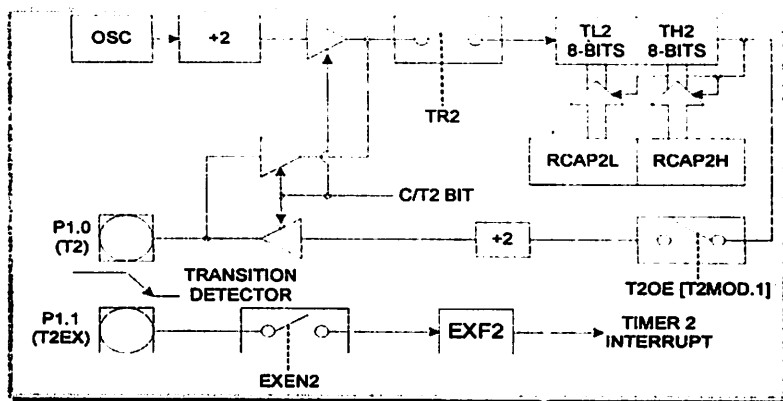
tugas 50% dengan rentang dari 61 Hz sampai 4 MHz jika dipakai 16 MHz kristal sebagai frekuensi mikrokontroler.

Untuk membuat *Timer/Counter 2* sebagai generator detak, bit C/T2 (T2CON) harus dibuat rendah dan bit T2OE (T2MOD.1) harus diset. Bit TR2 (T2CON.2) sebagai *start/stop* timer.

Frekuensi detak yang dihasilkan bergantung kepada frekuensi osilator yang dipakai dan nilai reload pada *capture register Timer 2* (RCAP2H,RCAP2L) seperti dalam persamaan berikut ini :

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

Gambar berikut ini menunjukkan *Timer 2* sebagai *clock out generator* :



Gambar 2.12
Timer 2 dalam *Clock Out Mode*^[3]

2.3. LCD (*Liquid Cristal Display*)

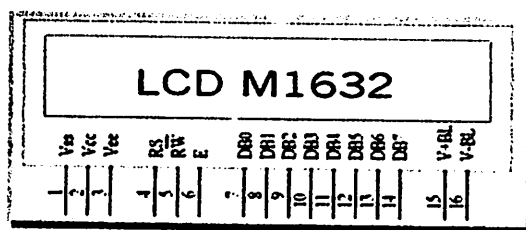
Liquid Cristal Display adalah modul tampilan yang mempunyai konsumsi daya yang relatif rendah dan terdapat sebuah kontroler CMOS didalamnya. Kontroler tersebut berfungsi sebagai pembangkit ROM / RAM

dan display data RAM. Semua fungsi tampilan dikontrol oleh suatu instruksi, modul LCD dapat dengan mudah diinterfacekan dengan MPU.

LCD yang digunakan dalam Tugas Akhir ini adalah LCD yang memiliki kemampuan sebagai berikut :

- a. Meliputi 32 karakter yang dibagi menjadi 2 baris dengan display dot matrik 5 x 7 ditambah cursor.
- b. Karakter generator ROM dengan 192 karakter.
- c. Karakter generator RAM dengan 8 tipe karakter.
- d. Dilengkapi fungsi tambahan yaitu display clear, cursor home, display ON/OFF, cursor ON / OFF, display character blink, cursor shift dan display shift.
- e. Internal data.
- f. 80 x 8 bit display data RAM.
- g. Dapat diinterfacekan dengan MPU 8 atau 4 bit.
- h. Internal otomatis dan reset pada power ON.
- i. + 5 Volt power supply tunggal.

Gambar dibawah ini menunjukkan LCD beserta pin – pinnya :



Gambar 2.13
Bentuk fisik dari LCD^[4]

Dimana untuk definisi pin-pin yang terdapat dalam LCD tersebut dapat dilihat pada tabel berikut

Tabel 2-5
Definisi Pin-Pin LCD^[4]

NAMA PIN	JUMLAH TERMINAL	NOMOR TERMINAL	I/O	LEVEL	TUJUAN	FUNGSI
DB0-DB3	4	7-10	I/O	H/L	μ C	4 bit bus data <i>lower tristate</i> dua arah, dapat dibaca atau ditulis terhadap μ C.
DB4-DB7	4	11- 14	I	H/L	μ C	4 bus data <i>upper tristate</i> dua arah, dapat ditulis atau dibaca terhadap μ C, DB7 juga sebagai <i>busy flag</i> .
E	1	6	I	H/L	μ C	Sinyal tanda mulai operasi. Aktif saat operasi <i>write</i> atau <i>read</i> .
R/W	1	5	I	H/L	μ C	0 : Write 1 : Read
RS	1	4	-	H/L	μ C	Sinyal pilih register : 0 : <i>Instruction register (write) busy flag</i> dan <i>address counter (read)</i> 1 : <i>Data register (write dan read)</i>
V _{LC}	1	3	-	-	<i>Power Supply</i>	<i>Power supply</i> untuk mendrive LCD guna mengatur kontrasnya.
V _{DD}	1	3	-	-	<i>Power Supply</i>	+5 Volt
V _{SS}	1	1	-	-	<i>Power Supply</i>	0V: <i>Terminal Ground</i>
V+BL	1	15	-	-	<i>Back Light Supply</i>	4 - 4,2 Volt 50 - 200 mA
V-BL	1	16	-	-	<i>Back Light Supply</i>	0 Volt (<i>ground</i>)

2.4. Komunikasi Data Serial

Pengiriman data serial biasanya untuk jarak yang relative jauh. Pada port serial 89S8252 dapat digunakan untuk komunikasi data secara sinkron maupun asinkron. Komunikasi data serial sinkron merupakan bentuk komunikasi data serial yang memerlukan sinyal *clock* untuk sinkronisasi. Sinyal *clock* tersebut akan tersulut pada setiap bit pengiriman data, sedangkan komunikasi data serial asinkron tidak memerlukan sinyal *clock* sebagai sinkronisasi. Pengirimannya akan dimulai dari LSB (*Least Significant Bit*) dan diakhiri oleh MSB (*Most Significant Bit*).

2.4.1. Pengaturan *Baud Rate Port Serial*

Baud rate dari *port serial* 89S8252 yang dapat diatur adalah Mode 1 dan mode 3, sedangkan pada mode 0 dan Mode 2 mempunyai *baud rate* dengan kecepatan yang permanen. Kecepatan *baud rate* yang digunakan ditunjukkan dalam table berikut ini :

Tabel 2-6
Baud Rate Mode Serial^[5]

$$BR = \frac{f_{osc}}{12 \times (256 - TH1) \times 32}$$

2.4.2. Komunikasi Serial Antara PC Dengan MCU

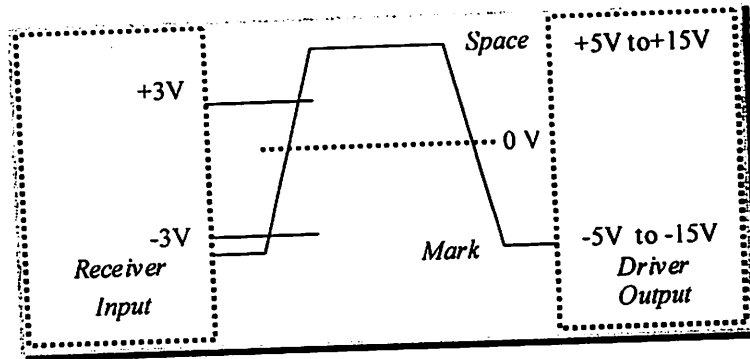
IC digital termasuk mikrokontroller, umumnya bekerja pada level tegangan TTL (transistor-transistor logic), yang dibuat atas dasar tegangan catu daya +5 Volt. Rangkaian input TTL menganggap tegangan kurang dari 0,8 Volt sebagai level tegangan '0' dan tegangan lebih dari 2,0 volt

dianggap sebagai level tegangan '1'. Level tegangan ini sering dikatakan sebagai level tegangan TTL. Sedangkan pada PC / *serial port* tegangan antara +3 sampai +15 Volt dianggap sebagai level tegangan '0', dan tegangan antara -3 sampai -15 Volt dianggap sebagai level tegangan '1'. Dari perbedaan acuan tegangan tersebut diperlukan RS-232 sebagai jembatan untuk menghubungkan antara MCU dengan PC, sehingga *transfer* data dapat dilakukan.

2.4.3. *Interface* Unit RS-232

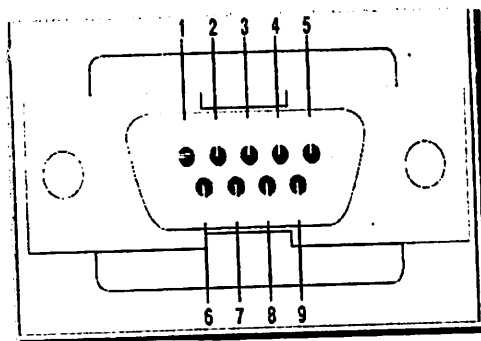
RS-232 merupakan seperangkat alat yang berfungsi sebagai *interface* dalam proses *transfer* data secara serial. Metode pengiriman secara serial RS-232 adalah asinkron. Pengiriman asinkron berarti tidak memerlukan pewaktu sebagai sinkronisasi. Dalam pengiriman serial asinkron, *clock* tidak dikirimkan, tetapi dikondisikan oleh *timing start bit* yang merupakan isyarat dari sumber ke tujuan untuk mengkodekan adanya pengiriman karakter sudah selesai dikirim.

Karakteristik elektris dari sistem RS-232 adalah mempunyai tegangan keluaran antara -15 sampai +15 Volt. Tegangan +5 sampai +15 Volt untuk mewakili level rendah (logika '0' / *spacing*) dan tegangan -5 sampai -15 volt mewakili level tinggi (logika '1' / *marking*). Hal tersebut dapat ditunjukkan pada gambar berikut ini :



Gambar 2.14
Level Logika Standar RS-232^[6]

Di dalam komputer terdapat fasilitas komunikasi serial yang menggunakan standar RS-232, yaitu terletak pada COM1 dan COM2. Kedua fasilitas ini menggunakan konektor DB9 atau DB25 sebagai penghubung dengan piranti luar. Gambar konektor dapat dilihat pada gambar berikut ini :



Gambar 2.15
Konektor DB-9^[7]

Fungsi masing-masing pin pada DB-9 seperti terdapat pada tabel berikut :

Tabel 2-7
Fungsi Pin RS-232 Dalam DB-9¹⁷¹

PIN	NAMA	FUNGSI
1	DCD (<i>Data Carrier Detect</i>)	Mendeteksi sinyal <i>carrier</i> dari modem lain
2	RD (<i>Receive Data Line</i>) / (R _r D)	Pengiriman data serial dari DCE ke DTE
3	TD (<i>Transmit Dta Line</i>) / TxD)	Pengiriman data serial dari DTE ke DCE
4	DTR (<i>Data Terminal Ready</i>)	Memberitahu DCE bahwa DTE telah aktif dan siap untuk bekerja
5	<i>Ground</i>	Referensi semua tegangan antar muka
6	DSR (<i>Data Set Ready</i>)	Memberitahu DTE bahwa DCE telah aktif dan siap untuk bekerja
7	RTS (<i>Request To Send</i>)	Memberitahu DCE bahwa DTE akan mcngirim data
8	CTS (<i>Clear To Send</i>)	Memberitahu DTE bahwa DCE siap menerima data
9	RI (<i>Ring Indikator</i>)	Aktif jika modem menerima sinyal ring pada jalur telepon

Jalur data (*TxD* dan *RxD*) untuk *transport* data, *TxD* adalah jalur output pada komputer, data dikirim dari pin ini. Sedangkan *RxD* adalah penerima untuk komputer, data yang datang akan diterima oleh pin ini. Pin keempat adalah output (*RTS*) di mana sebuah sinyalakan diberikan pada alat yang dihubungkan dengan maksud meminta kiriman data. *CTS* adalah sinyal masukan yang menunggu sinyal dari alat yang terhubung. Ketika alat tersebut menerima sinyal *RTS* dan bisa menerima data maka ia akan mengirimkan sinyal balik yang merupakan *CTS*. *DTR* adalah sinyal keluaran yang memberi tanda bahwa ada alat yang terhubung dan akan mengirimkan data. *DSR* merupakan sinyal input yang mana jika alat yang terhubung menerima sinyal *DTR* ia akan memberi sinyal balik kemudian diterima sebagai sinyal *DSR*.

BAB III

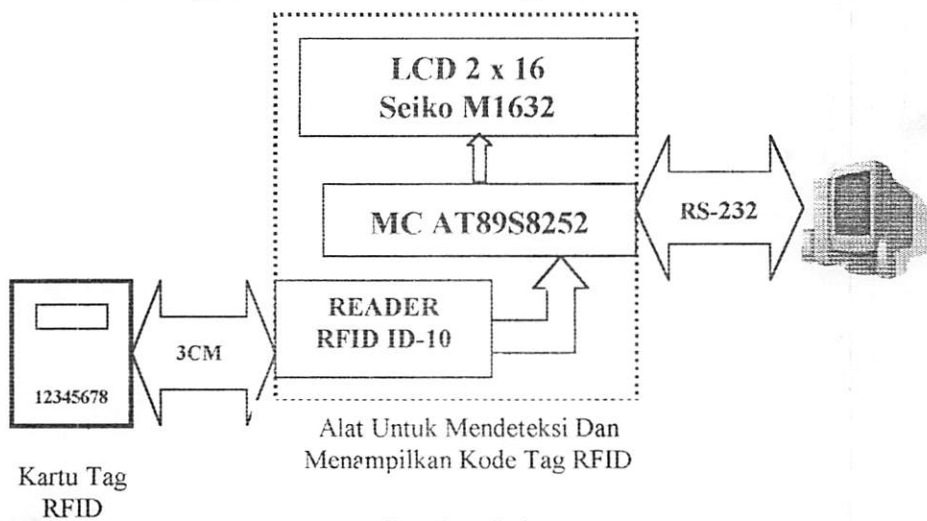
PERANCANGAN DAN PEMBUATAN ALAT

Dalam perancangan alat ini dapat dibagi menjadi dua bagian yang terdiri atas perancangan perangkat keras dan perancangan perangkat lunak yang mendukung system kerja alat ini.

Pada perancangan perangkat keras meliputi perancangan diagram blok sistem, perancangan rangkaian *RFID*, mikrokontroller AT89S8252, serta *display LCD*. Sedangkan perancangan perangkat lunaknya, menjelaskan perangkat lunak yang digunakan untuk menjalankan sistem.

3.1. Perancangan Dan Pembuatan Perangkat Keras

3.1.1. Perancangan Dan Pembuatan Diagram Blok Sistem



Gambar 3.1
Diagram Blok Sistem

3.1.1.1. Keterangan Masing-masing Blok

Keterangan masing-masing blok dapat dilihat sebagai berikut :

1. *RFID reader* : untuk pembaca pola kode yang terdapat pada *tag RFID*.

2. *Tag RFID* : sebagai identitas yang akan dideteksi oleh *RFID reader*.
3. Mikrokontroler : sebagai penerjemah dari kode *tag RFID* untuk selanjutnya dikirim menuju *personal computer*.
4. RS-232 : sebagai *interface* antara mikrokontroler dengan *PC*.
5. *LCD* : digunakan untuk melihat secara visual nomor *ID* dari *Tag RFID* yang terdeteksi.
6. *PC* : untuk menampilkan kode dari *tag RFID* dan memasukkan data dari pemilik kendaraan bermotor berupa nama pemilik kendaraan bermotor, alamat pemilik, nomor rangka, bahan bakar, kode lokasi, nomor polisi, nomor mesin, jenis kendaraan, warna kendaraan, nomor urut pendaftaran, dan tanggal jatuh tempo masa berlaku STNK, dan untuk menghapus data kode *RFID*.

3.1.1.2. Prinsip Kerja Dari Sistem :

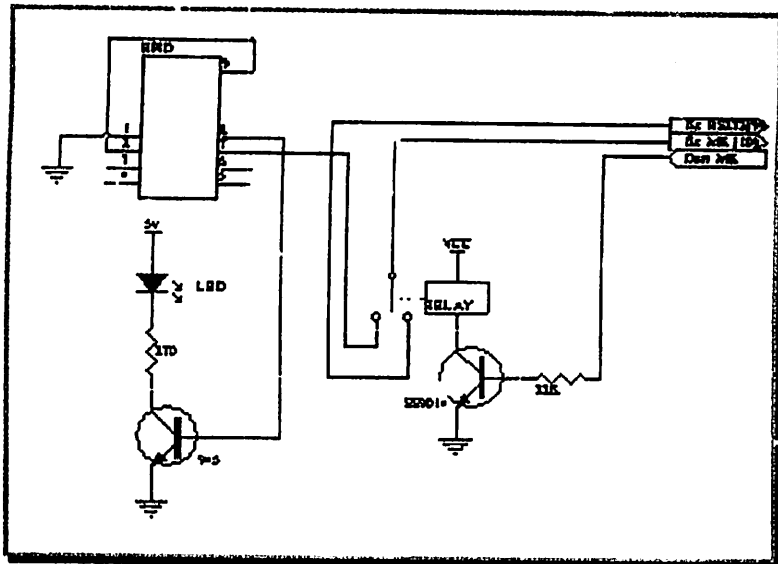
Pertama-tama para pemilik kendaraan bermotor harus memiliki *tag RFID* yang berisikan kode yang berbeda-beda yang ditempel pada badan kendaraanya masing-masing. Kemudian kode tersebut akan dideteksi oleh *reader RFID*, setelah itu kode tersebut diproses oleh mikrokontroler dan ditampilkan pada *LCD*. Kode tersebut juga dikirim ke EEPROM yang ada dimikrokontroler untuk disimpan agar apabila catu daya dimatikan kode-

kode tersebut tidak akan hilang. Kemudian kode tersebut dikirim ke *personal* komputer untuk ditampilkan beserta dengan data-data pemilik kendaraan sesuai dengan yang ada di STNK. Apabila sebelumnya sudah terdaftar maka data yang ada di STNK pemilik kendaraan tersebut akan muncul pada layar komputer, tetapi apabila belum terdaftar sebelumnya maka operator akan mengisikan data-data tersebut sesuai yang ada di STNK. Selain untuk menampilkan data-data fungsi dari komputer adalah untuk menghapus kode *RFID*.

3.1.2. Perancangan Dan Pembuatan Rangkaian *RFID*

Pada sistem *RFID* umumnya, *tag/transponder* ditempelkan pada suatu objek. Setiap *tag* dapat membawa informasi yang unik, diantaranya : *serial number*, model, warna, dan data lain dari objek tersebut. Ketika *Tag* ini melalui medan yang dihasilkan oleh pembawa *RFID* yang *kompatible*, *tag* akan mentransmisikan informasi yang ada pada *tag* kepada pembaca *RFID*, sehingga proses identifikasi objek dapat dilakukan.

Pada perancangan alat ini juga digunakan LED sebagai indicator pendeteksian *RFID*. Dalam perancangan rangkaian *RFID* hanya memerlukan pin P3.0 yang digunakan sebagai pengiriman data ke mikrokontroller seperti terlihat pada gambar berikut ini :



Gambar 3.2
Rangkaian *RFID Reader*

3.1.2.1. Frekuensi Kerja *RFID*

Faktor penting yang harus diperhatikan dalam *RFID* adalah frekuensi kerja dari system *RFID*, yaitu frekuensi yang digunakan untuk komunikasi *wireless* antara pembaca *RFID* dengan *tag RFID*.

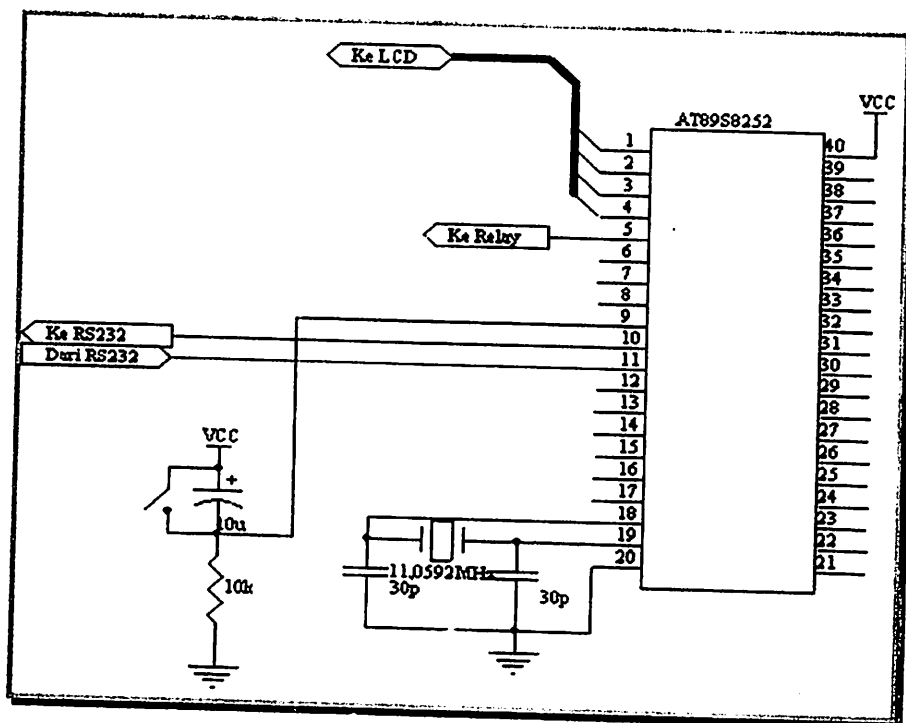
Ada beberapa *band* frekuensi yang digunakan untuk system *RFID*. Pemilihan dari frekuensi kerja system *RFID* akan mempengaruhi jarak komunikasi, *interferensi* dengan frekuensi system radio lain. Untuk frekuensi yang rendah umumnya digunakan *tag pasif* dan inilah yang dipakai dalam perancangan dan untuk frekuensi tinggi digunakan *tag aktif*.

Pada frekuensi rendah *tag pasif* tidak dapat mentransmisikan data jarak yang jauh, karena keterbatasan daya yang diperoleh dari medan elektromagnetik. Akan tetapi dapat dilakukan tanpa kontak langsung. Pada kasus ini hal yang perlu mendapatkan perhatian adalah *tag pasif* harus terletak jauh dari objek logam karena logam secara signifikan mengurangi

fluida dari medan magnet. Akibatnya tag RFID tidak bekerja dengan baik karena tag tidak menerima daya minimum untuk dapat bekerja.

3.1.3. Perancangan Dan Pembuatan Mikrokontroller Sebagai Minimum Sistem

Rangkaian minimum dari mikrokontroller AT89S8252 terdiri dari 3 kapasitor, 1 IC mikrokontroller, 1 resistor, dan 1 kristal. Dengan rangkaian yang sederhana ini penulis membuat mikrokontroller sebagai minimum sistem menjadi pengontrol alat, disamping itu rangkaian ini dapat dibuat bermacam-macam alat dengan menambah sedikit komponen tambahan lainnya. Dari rangkaian tersebut yang berpengaruh terhadap kecepatan proses menjalankan program adalah kristal. Adapun rangkaiannya ditunjukkan pada gambar berikut ini :



Gambar 3.3
Mikrokontroller Sebagai Minimum Sistem

Fungsi MCU yang digunakan sebagai *slave* aplikasi MCU yang mengontrol masukan dan keluaran data dalam sistem informasi ini adalah :

- a. Pin 5 (P1.4), relay berfungsi memilih data input dari *RFID* atau komputer.
- b. Pin 9 (RST), reset aktif tinggi yang terhubung dengan rangkaian *power on reset* dan jika diaktifkan akan mereset mikrokontroler AT89S8252.
- c. Pin 10 (RXD), *port* komunikasi input serial yang berfungsi sebagai pengiriman data dari *RFID* dan *PC*.
- d. Pin 11 (TXD), *port* komunikasi output serial yang berfungsi sebagai pengiriman data ke RS-232 yang diteruskan ke *PC*.
- e. Pin 18 (XTAL2), untuk *clock* pada mikrokontroler.
- f. Pin 19 (XTAL1), untuk *clock* pada mikrokontroler.
- g. Pin 20 (GND), digunakan sebagai ground mikrokontroler.
- h. Pin 40 (Vcc), digunakan sebagai sumber tegangan.
- i. Pin 1,2,3, dan 4 (P1.0, P1.1, P1.2, P1.3), digunakan sebagai output dari mikrokontroler yang dihubungkan ke *LCD*.

3.1.3.1. Perancangan Rangkaian *Clock*

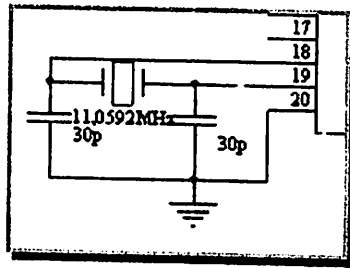
Kecepatan proses pengolahan data pada mikrokontroler ditentukan oleh *clock* (waktu) yang dikendalikan oleh mikrokontroler tersebut. Pada mikrokontroler AT89S8252 terdapat *internal clock*. *Internal clock* generator berfungsi sebagai sumber *clock*, tapi masih memerlukan rangkaian tambahan untuk membangkitkan *clock* yang diperlukan. Rangkaian *clock* ini terdiri dari 2 buah kapasitor dan 1 buah kapasitor yang

dirangkai sedemikian rupa dan kemudian ke pin 18, pin 19, dan pin 20 pada port 3 dari mikrokontroller. Dalam perancangan rangkaian ini menggunakan :

1. C1 dan C2 = $30 \text{ pF} \pm 10 \text{ pf}$ (digunakan kristal)
= $40 \text{ pF} \pm 10 \text{ pF}$ (digunakan untuk keramik resonator)
(data ini disesuaikan dengan spesifikasi pada data sheet).

2. Kristal = 11.0592 MHz

Adapun rangkaian *clock* dapat ditunjukkan pada gambar berikut ini :

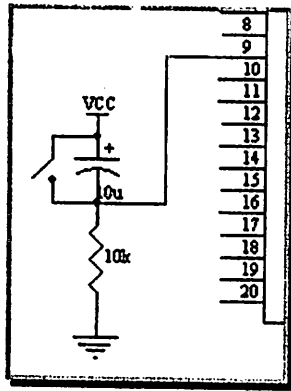


Gambar 3.4
Rangkaian *Clock*

3.1.3.2. Perancangan Rangkaian *Reset*

Reset dapat dilakukan secara manual atau otomatis pada saat catu daya diaktifkan (*power on reset*). Rangkaian ini akan me-*reset* mikrokontroller secara otomatis setiap kali catu daya diaktifkan. Reset terjadi dengan adanya logika 1 selama minimal dua siklus mesin serta oscillator telah bekerja pada penyemat RST. Setelah penyemat RST kembali ke logika 0 maka mikrokontroller akan mulai menjalankan program.

Adapun rangkaian *reset* dapat ditunjukkan pada gambar berikut ini :



Gambar 3.5
Rangkaian *Reset*

Rangkaian *reset* terbentuk oleh komponen R dan C yang sudah baku (ditetapkan oleh perusahaan pembuat IC AT89S8252). Nilai R yang dipakai adalah 10 KΩ dan C= 10 µF.

Sedangkan untuk mencari frekuensi dari *reset* tersebut menggunakan rumus sebagai berikut :

$$F_o = \frac{1}{1,1RC}$$

Sehingga dengan komponen resistor dengan nilai 10 KΩ serta kapasitor dengan nilai 10 µF akan dihasilkan frekuensi :

$$F_o = \frac{1}{1,1RC}$$

$$F_o = \frac{1}{1,1 \cdot 10^3 \cdot 10^{-6}}$$

$$F_o = 9,09 \text{ hz}$$

maka periode *clock* $T = \frac{1}{f_o}$

$$T = \frac{1}{9,09}$$

$$= 0,11 \text{ detik}$$

3.1.4. Perancangan Dan Pembuatan Tampilan (LCD)

LCD Display module M 1632 buatan *Seiko instrument Inc.* terdiri dari 2 bagian, yang pertama merupakan panel *LCD* sebagai media penampil informasi dalam bentuk huruf atau angka 2 baris, masing-masing baris bisa menampung 16 huruf atau angka.

Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroller yang ditempelkan dibalik panel *LCD*, berfungsi mengatur tampilan informasi serta berfungsi mengatur komunikasi M 1632 dengan mikrokontroller utama. Dengan demikian pemakaian M 1632 menjadi sederhana dibandingkan dengan sistem lain, karena M. 1632 cukup mengirim kode *ASCII* dari informasi yang ditampilkan seperti layaknya memakai sebuah printer.

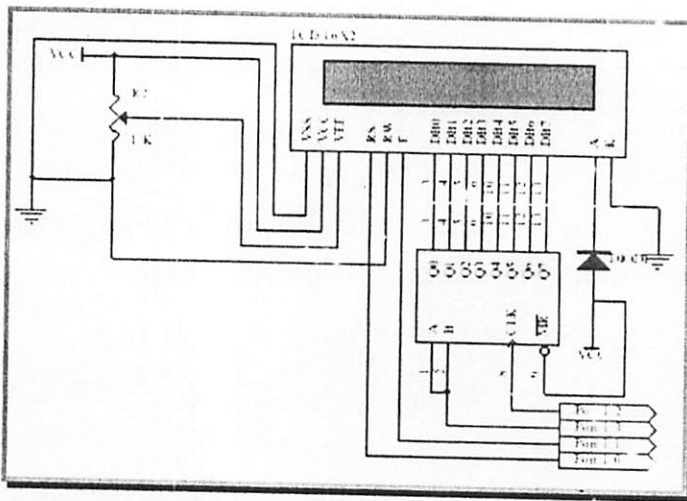
Rangkaian *LCD M 1632* ini adalah komponen *display* yang umum digunakan. *Display LCD M 1632* ini memiliki ROM sebagai penyimpanan karakter sebanyak 192 buah. Sebelum mengoperasikan *LCD* sebagai penampil karakter, terlebih dahulu ditentukan format penulisan *LCD*.

Dalam penulisan format *LCD* terada beberapa aturan yang diberikan oleh pabrik pembuatnya (dalam *data sheet*) yaitu:

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

Jika penginisialisasian sudah selesai, langkah selanjutnya adalah menulis karakter yang diinginkan. Misalnya tampilan yang diinginkan

adalah "123", maka format data yang ditansfer ke jalur data adalah format data *BCD* angka 123 yang masing-masing disertai data posisi baris. Data yang dikirim ke *LCD* cukup satu kali, selanjutnya data akan terus tampil berulang-ulang oleh *LCD* itu sendiri selama tidak ada instruksi untuk membersihkan layar. Hubungan pin data dengan dengan pin kontrol *LCD* dengan *MCU* ditunjukkan pada gambar di bawah ini :



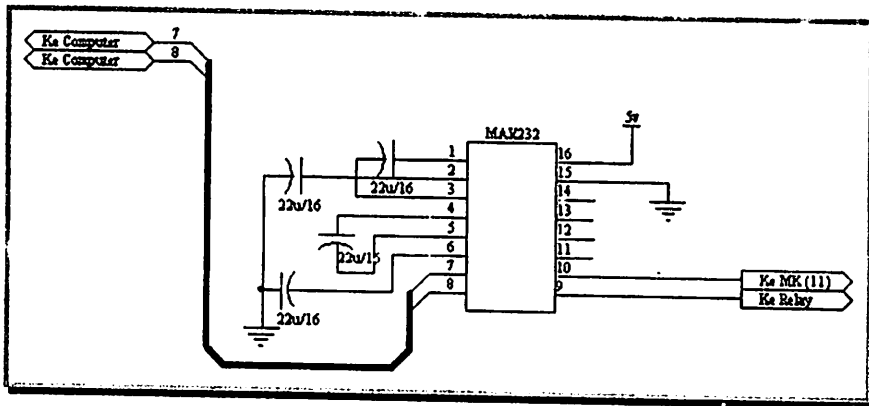
Gambar 3.6
Rangkaian Tampilan *LCD*

Fungsi dari masing-masing pin *LCD* yang digunakan adalah :

- Pin RS dihubungkan dengan *port* 1.0 dari *MCU* untuk membedakan sinyal antara instruksi program atau instruksi penulisan data
- Pin E dihubungkan dengan *port* 1.1 dari *MCU* untuk memberikan instruksi penulisan pada alamat *LCD*
- Pin DB0 – DB7 dihubungkan pada Q0 – Q7 IC DM 74LS164 *shif register* dimana input A dan B terhubung pada *port* 1.3 dan *CLK* terhubung pada *port* 1.2 sebagai pengontrolan data yang akan masuk ke *LCD*.

3.1.5. Perancangan dan Pembuatan Antar Muka Dengan *Personal Komputer (PC)*

Perangkat ini digunakan sebagai alat untuk memasukkan data *base* yang belum terdapat dalam data base memori alat. Dengan *PC* ini kita dapat menyimpan data *base* yang telah ada pada alat ke komputer sehingga dokumen tetap tersimpan dengan rapi di dalam komputer. Komunikasi antara *PC* dan mikrokontroller menggunakan RS-232. Dalam perancangan ini *port* yang digunakan hanyalah pin P3.0 dan pin P3.1 sebagai jalur pengiriman dan penerimaan data, seperti terlihat pada gambar berikut ini :



Gambar 3.7
Rangkaian Komunikasi Serial Ke *PC*

3.2. Perancangan Perangkat Lunak

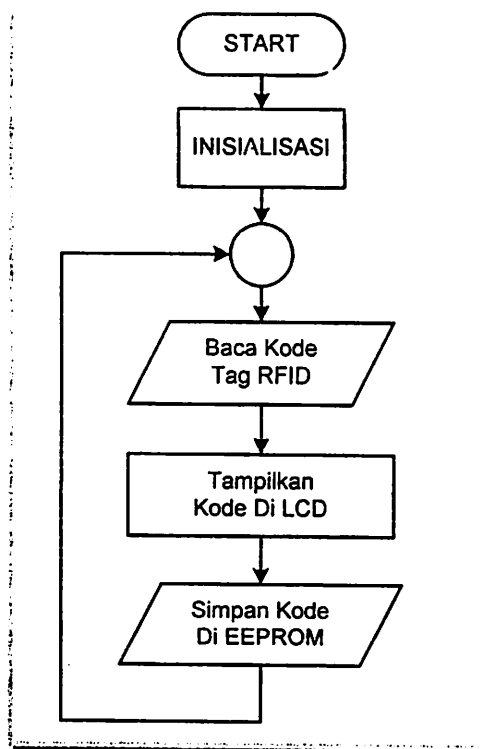
Perangkat lunak dibagi atas dua macam, yaitu program mikrokontroller dan program komputer.

3.2.1. Mikrokontroller

Program mikrokontroller bertujuan untuk mengontrol masukan dan keluaran. Hal-hal yang dikontrol oleh mikrokontroller adalah proses

komunikasi serial dengan *RFID* dan komputer, dan proses penampilan kode *tag RFID* pada *LCD*.

Proses pendeteksian berawal dari pembacaan *tag RFID* oleh *reader RFID* yang kemudian ditampilkan ke *LCD* dan setelah itu disimpan dalam *EEPROM*, untuk kemudian akan ditampilkan ke komputer bersama dengan data *base*-nya. Proses program mikrokontroller dapat dilihat pada flowchart program mikrokontroller berikut ini :



Gambar 3.8
Flowchart Program Mikrokontroller

Di dalam proses komunikasi serial antara mikrokontroller dengan komputer terlebih dahulu ditentukan baud rate yang digunakan. Baud rate yang dibangkitkan menggunakan timer 1 dengan timer mode 2 (8 auto

baud rate sebesar 9600 dengan SMOD = 0 dengan menggunakan $f_{osc} = 11,0592$.

$$\text{Baudrate} = \frac{2^{SMOD}}{32} \times \frac{f_{osc}}{12 \cdot [256 - (TH1)]}$$

$$9600 = \frac{11,0592 \text{ MHz}}{384 \cdot [256 - (TH1)]}$$

$$3686400 \cdot [256 - (TH1)] = 11,0592 \cdot 10^6$$

$$256 - (TH1) = 3$$

$$TH1 = 256 - 3$$

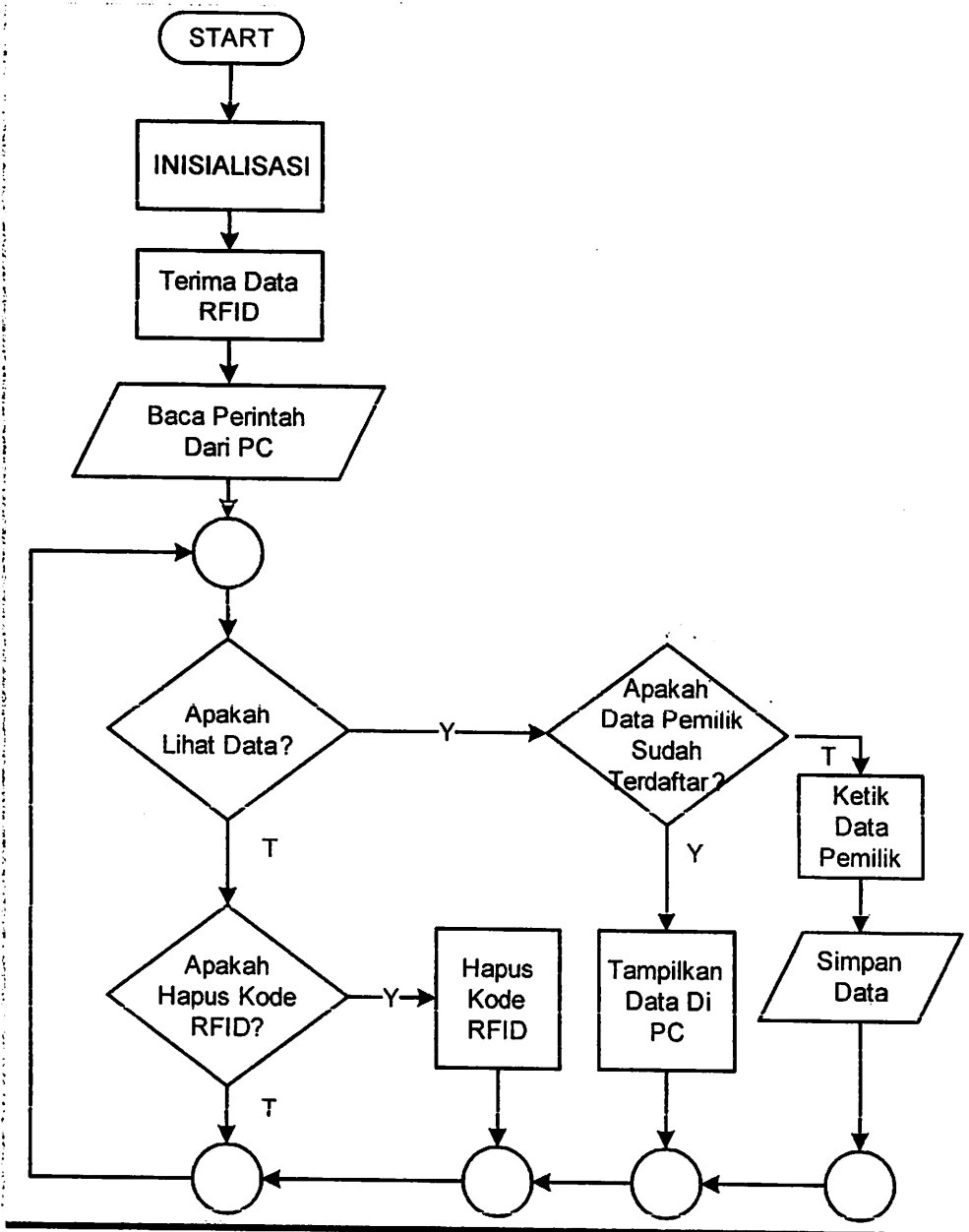
$$= 253$$

$$= FD_{11}$$

3.2.2. Program Utama Pada PC

Pada dasarnya program komputer bertujuan untuk mengorganisir komunikasi antara komputer sebagai pengolah data dengan mikrokontroller sebagai saluran masukan dan keluaran, sehingga untuk perangkat lunak dalam program komputer diketahui terlebih dahulu aturan-aturan yang digunakan untuk mengatur jalannya komunikasi antara komputer dengan mikrokontroller.

Pada program komputer ini terdiri dari dua menu yaitu pembacaan EEPROM beserta dengan tampilan data pemilik kendaraan, dan hapus semua kode *RFID*. Proses keseluruhan dapat dilihat pada flowchart program komputer berikut ini :



Gambar 3.9
Flowchart Program Pada Komputer

BAB IV

PENGUJIAN ALAT

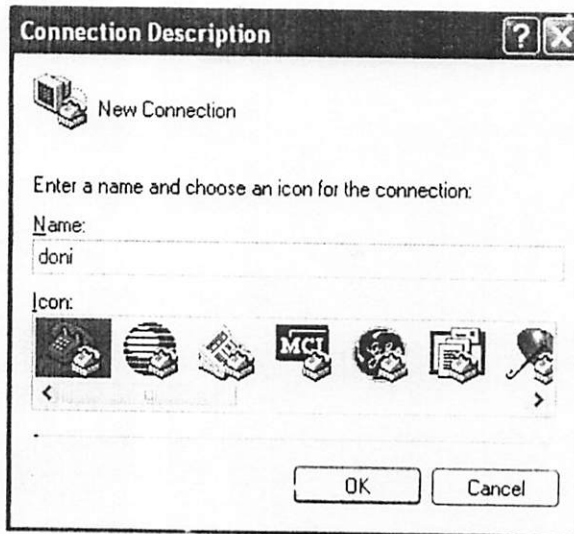
Dalam bab ini membahas tentang pengujian dan pengukuran dari peralatan yang dibuat. Secara umum pengujian ini bertujuan untuk mengetahui apakah piranti yang telah direalisasikan dapat bekerja sesuai dengan perencanaan yang telah direncanakan.

4.1. Pengujian *RFID*

Tujuan dari pengujian ini adalah untuk mengetahui apakah *tag RFID* bisa dibaca oleh *reader RFID*. Adapun cara pengujiannya adalah dengan merangkai rangkaian *RFID* dan kemudian menghubungkan ke *COM1 PC*. Untuk menguji reader bisa membaca kartu *RFID* dilakukan melalui *Hyper terminal*.

Prosedur pengujian kartu pada reader :

- a. Menghubungkan rangkaian *RFID* ke *COM1 PC*.
- b. Membuka *hyper terminal* (*start* → *all program* → *accessories* → *comunication* → *hyper terminal*).
- c. Memberi nama dan memilih *icon* pada *connection description*.



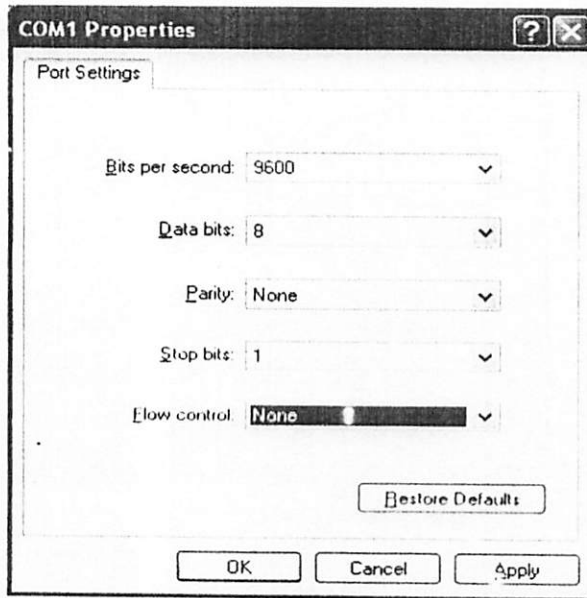
Gambar 4.1
Kotak Dialog *Connection Description*

d. Memilih COM1 pada kotak dialog *connect to*



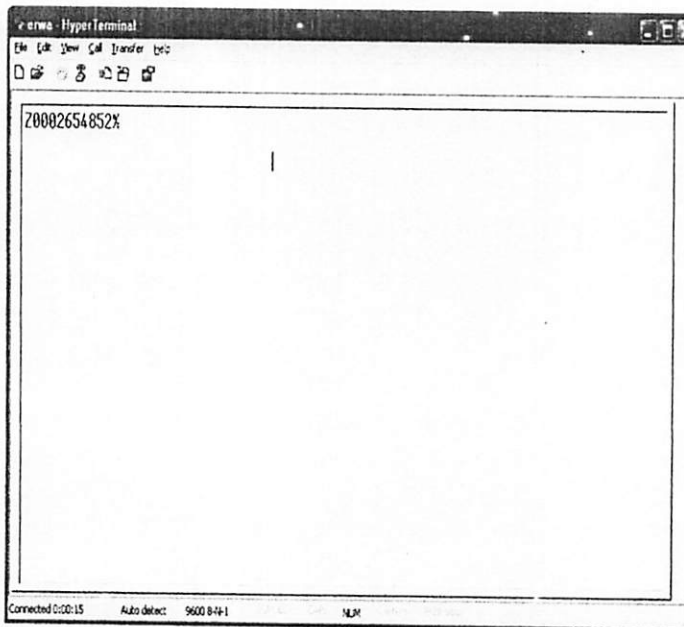
Gambar 4.2
Kotak Dialog *connect to*

e. Pada Com1 *propertis* mengubah *bit rate per second* menjadi 9600 dan *flow control* menjadi *none*.



Gambar 4.3
Kotak Dialog COM1 Properties

- f. Menempatkan kartu pada jarak yang dijangkau *reader* sehingga menampilkan angka dari kartu tersebut.



Gambar 4.4
Identifikasi *Reader* Terhadap Kartu

Hasil Pengujian Pembacaan RFID

Tabel 4-1
Hasil Pengujian Pembacaan RFID

Jarak (cm)	Percobaan									
	1	2	3	4	5	6	7	8	9	10
1	√	√	√	√	√	√	√	√	√	√
2	√	√	√	√	√	√	√	√	√	√
3	√	√	√	√	√	√	√	√	√	√
4	√	√	√	√	√	√	√	√	√	√
5	√	√	√	√	√	√	√	√	√	√
6	-	-	-	-	-	-	√	-	-	-
7	-	-	-	-	-	√	-	-	-	-
8	-	-	-	-	-	-	-	-	-	-
9	-	-	-	-	-	-	-	-	-	-
10	-	-	-	-	-	-	-	-	-	-

Tabel di atas merupakan hasil pengujian dimana kartu yang menghadap *reader* adalah bagian depan. Jarak yang baik untuk bisa teridentifikasi adalah 5 cm. Untuk bagian belakang menghasilkan data yang sama, tetapi untuk pengujian dimana kartu tegak lurus dengan *reader* hanya bisa saat kartu berjarak sangat dekat dengan *reader*.

4.2. Pengujian Pin Mikrokontroler

Tujuan dari pengujian ini adalah untuk menguji pin-pin yang ada di mikrokontroler sehingga dapat bekerja dengan normal.

Prosedur pengujian :

- a. Mengukur dengan multimeter DT890B.
- b. Menguji pin-pin yang digunakan dalam perancangan ini baik *stand by* maupun pada waktu identifikasi.

Hasil pengujian terdapat dalam tabel berikut :

Tabel 4-2
Hasil Pengujian Pin-pin Mikrokontroller

Pin	Posisi	Tegangan
5 (relay)	<i>Stand by</i> identifikasi	5,8 V 0,83V
9 (reset)	<i>Stand by</i> identifikasi	0,92V 0,87V
10 (RXD)	<i>Stand by</i> identifikasi	6,51V 6,45V
11 (TXD)	<i>Stand by</i> identifikasi	6,51V 6,48V
18 (XTAL2)	<i>Stand by</i> identifikasi	3,42V 3,34V
19 (XTAL2)	<i>Stand by</i> identifikasi	3,23V 3,26V
20 (GND)	<i>Stand by</i> identifikasi	0,03V 0,51V
40 (VCC)	<i>Stand by</i> identifikasi	6,6V 6,65V

Dari tabel di atas dapat disimpulkan bahwa ketika keadaan *stand by* maupun saat terjadi identifikasi, tidak terlalu banyak perbedaan tegangan yang terjadi.

4.3. Pengujian Pin-pin Pada Max -232

Tujuan dari pengujian ini adalah untuk menguji pin-pin yang ada pada Max 232 sehingga dapat diketahui bekerja dengan normal.

Prosedur pengujian :

- a. Mengukur dengan menggunakan multimeter DT890B.
- b. Menguji pin-pin yang digunakan dalam perancangan ini baik pada saat *stand by* maupun identifikasi.

Hasil pengujian terdapat pada tabel berikut :

Tabel 4-3
Hasil Pengujian RS-232

Pin	Posisi	Tegangan
7 (RS-232 Output)	<i>Stand by</i> identifikasi	-8,82V -8.70V
8 (RS-232 input)	<i>Stand by</i> identifikasi	-10,15V -10,17V
9 (TTL/CMOS Output)	<i>Stand by</i> Identifikasi	6,51V 6,48V
10 (TTL/CMOS Input)	<i>Stand by</i> Identifikasi	6,62V 6,66V

Dari tabel di atas dapat disimpulkan bahwa ketika keadaan *stand by* maupun saat terjadi identifikasi, tidak terlalu banyak perbedaan tegangan yang terjadi.

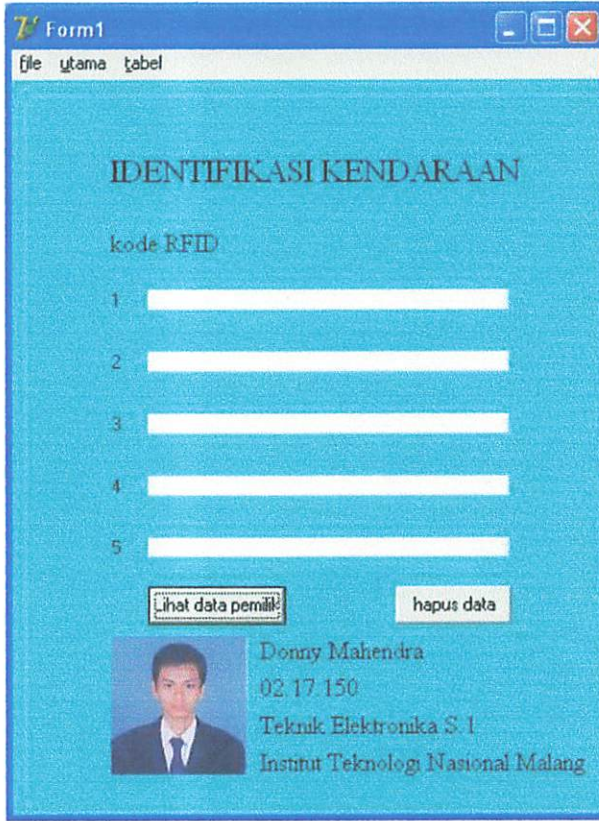
4.4. Pengujian Keseluruhan Sistem

Tujuan dari pengujian ini adalah untuk mengetahui apakah semua sistem berjalan dengan normal dan juga untuk mengetahui error yang terjadi.

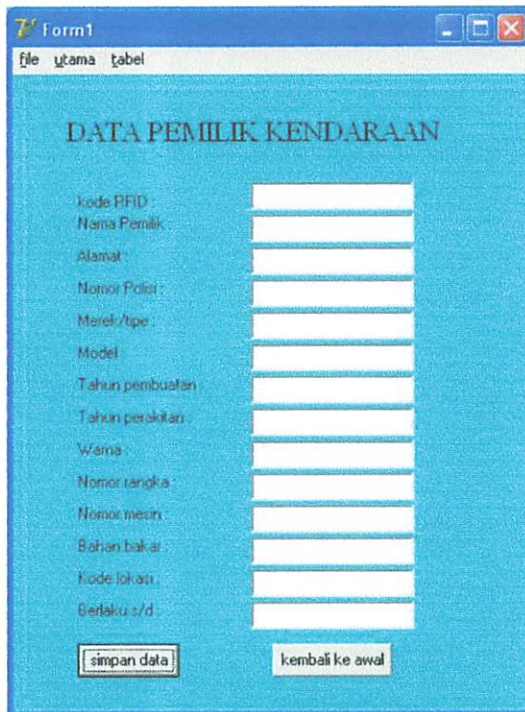
Alat ini memiliki dua menu utama yaitu pendeteksian (Mendeteksi Nomor Kode *Tag RFID*), dan komunikasi *PC*. Menu pertama dioperasikan dengan menggunakan *LCD*, *tag RFID*, mikrokontroler dan *RFID Reader*, yang dirangkai dalam sebuah *hardware*. Sedangkan menu kedua yaitu melihat data pemilik kendaraan beserta dengan kode *tag RFID* dioperasikan dengan menggunakan *PC*.

Hasil pengujian keseluruhan sistem dapat dilihat sebagai berikut :

Pengujian pertama kali dengan melakukan pendeteksian terhadap *tag RFID*, kemudian kode tag tersebut ditampilkan ke *LCD* dan disimpan dalam *EEPROM* dalam mikrokontroler. Setelah itu kode dikirim secara



Gambar 4.5
Tampilan Awal Kode *RFID* Pada *PC*



Gambar 4.6
Data Base Data Pemilik Kendaraan

BAB V

PENUTUP

5.1. Kesimpulan

Kesimpulan yang dapat diambil dari perancangan dan pembuatan alat bantu polisi dalam mengidentifikasi kendaraan dengan memanfaatkan teknologi *RFID* ini adalah sebagai berikut :

1. Alat ini dapat mendeteksi *tag RFID* tanpa penghalang dengan baik ± 5 cm.
2. Pengujian Dengan Menggunakan *Hyper terminal* membuktikan bahwa fungsi pin 11 (p 3.1/TxD) dapat berjalan dengan baik, dengan kata lain mikrokontroller dapat mengirimkan data berupa nomor kode tag RFID menuju ke komputer dengan baik.
3. Pada pin-pin mikrokontroller dan pin-pin RS-232 yang digunakan perubahan tegangan yang dihasilkan pada saat *standby* maupun terdeteksi tidak mengalami perubahan yang cukup *signifikan*.

5.2. Saran

Saran yang dapat penyusun sampaikan pada kesempatan ini adalah sebagai berikut :

1. Alat bantu polisi dalam mengidentifikasi kendaraan dengan memanfaatkan teknologi *RFID* ini dapat mendeteksi dengan jarak ± 5 cm, dapat dilakukan pengembangan untuk pendeteksian dengan jarak yang lebih jauh lagi.

2. Komunikasi dengan *PC* dapat dilakukan dengan sistem *nirkabel* sehingga komunikasi tidak menggunakan serial melainkan dilakukan secara *wireless*

DAFTAR PUSTAKA

- [1]. Antono, Danardono Dwi. 2004. "*RFID, Sebuah Teknologi Identifikasi Pengancam Privasi?*". Inovasi Vol.1/ xiv/ Agustus, 2004. hal. 31-34. Persatuan Pelajar Indonesia (PPI) Jepang.
- [2]. Catalog, Sunbest. 2004. "*Id-10 Series Low Cost Proximity Reader.*" <http://www.digi-ware.com/RFID.pdf>, 2004.
- [3]. Malik, Moh. Ibnu,ST. 2003. "*Belajar Mikrokontroller ATMEL AT89S8252.*" Yogyakarta: Gava Media.
- [4]. Seiko Instrument. 1987. "*Liquid Crystal Display Module M1632 User Manual.*" <http://www.alldatasheet.com/M1632.pdf>, 1987.
- [5]. Nalwan, Paulus Andi. 2003. "*Panduan Praktis Teknologi Antarmuka Dan Pemrograman MCAT89C5*"1. Jakarta: Pt. Elex Media Komutindo.
- [6]. Dallas Semikonduktor. "*Level Logika Standar RS-232*" <http://www.alldatasheet.com/Max232.pdf>.
- [7]. Indriantos. 2004. "*Konektor DB-9 Dan DB-25*". <http://library.gunadarma.ac.id/files/disk1/3/jbtgunadarma-gdl-s2-04-indriantos-122-babii.pdf>2004

LAMPIRAN

FOTO ALAT






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

**FORMULIR BIMBINGAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : DONNY MAHENDRA
NIM : 02.17.150
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Fakultas : Fakultas Teknologi Industri
Judul Skripsi : Perancangan Dan Pembuatan Alat Bantu Polisi
Dalam Mengidentifikasi Kendaraan dengan
Memanfaatkan Teknologi *RFID*.
Tanggal Mengajukan Skripsi : 24 Juni 2006
Selesai Menulis : 26 September 2006
Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT
Telah dievaluasi dengan nilai : 88

Malang, September 2006

**Diperiksa dan Disetujui,
Dosen Pembimbing**

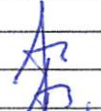
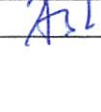


Ir. Yusuf Ismail Nakhoda, MT
NIP. Y. 101 880 0189



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

FORMULIR PERBAIKAN UJIAN SKRIPSI

Nama : DONNY MAHENDRA
N.I.M : 02.17.150
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Hari/Tanggal : Selasa, 26 September 2006
Judul : Perancangan Dan Pembuatan Alat Bantu Polisi Dalam
Mengidentifikasi Kendaraan Dengan Memanfaatkan
Teknologi *RFID*.

No.	MATERI PERBAIKAN	PARAF
1.	Pengujian Sistem	
2.	Kesimpulan	
3.	Flowchart Sistem	

**Mengetahui
Dosen Pembimbing**



**(Ir. Yusuf Ismail Nakhoda, MT)
NIP. Y. 101 880 0189**

**Diperiksa Dan Disetujui
Dosen Penguji II**



(M Ashar, ST, MT)



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

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

NAMA :
N I M :
Perbaikan meliputi :

- Pengisian system
- Komputer
- flowchart system

Malang,


(M. ASHAR)



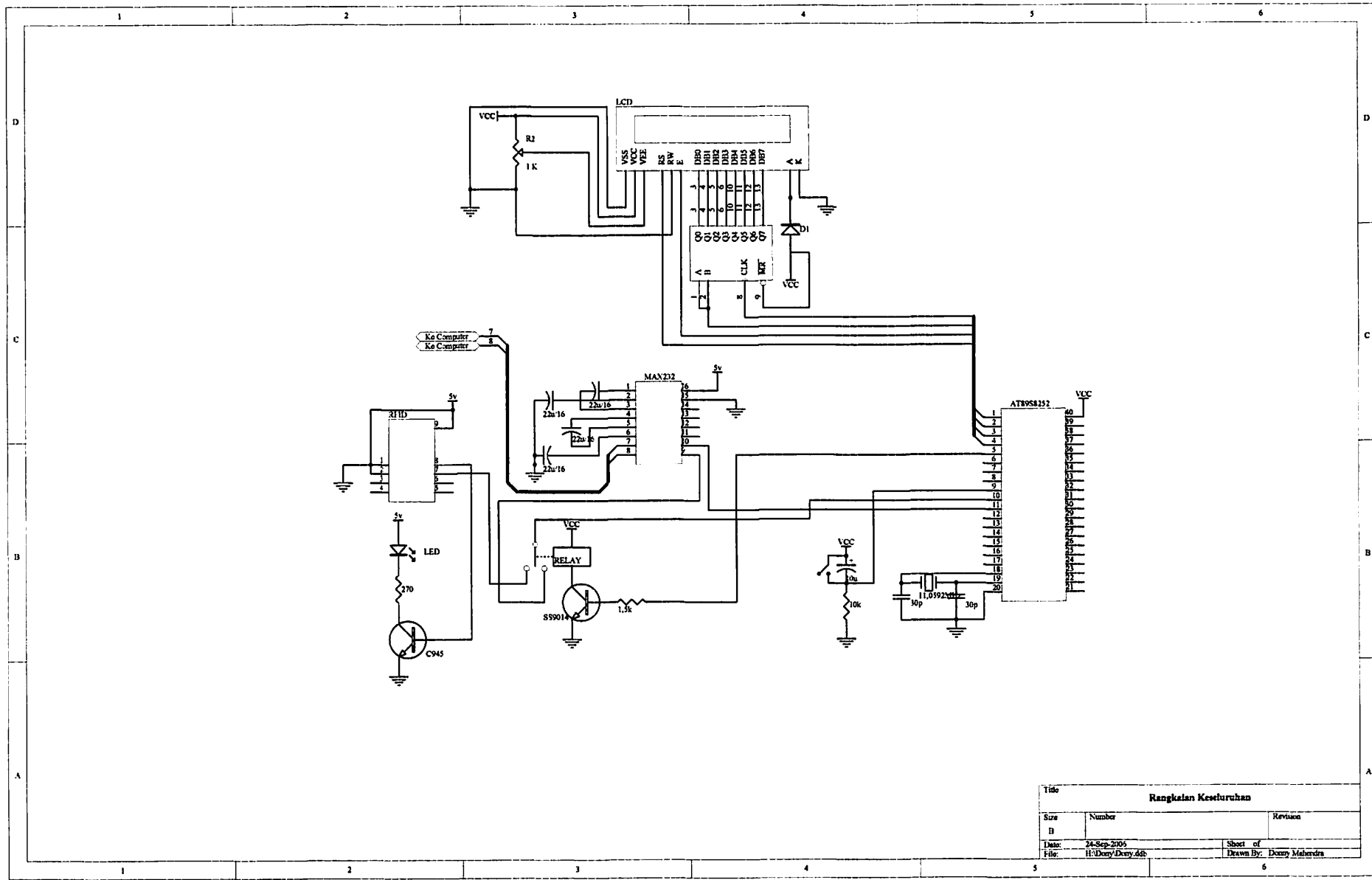
· FORMULIR BIMBINGAN SKRIPSI

Nama : DONNY MAHENDRA
Nim : 02.17.150
Masa Bimbingan : 24 Juni s/d 24 Desember 2006
Judul Skripsi : **PERANCANGAN DAN PEMBUATAN ALAT BANTU
POLISI DALAM MENGIDENTIFIKASI KENDARAAN
DENGAN MEMANFAATKAN TEKNOLOGI *RFID***

No.	Tanggal	Uraian	Parap Pembimbing
1.	04-09-2006	Demo Alat	
2.	07-09-2006	Format Penulisan Diperbaiki	
3.	11-09-2006	Memperbaiki Abstraksi	
4.	13-09-2006	Memperbaiki <i>Flowchart</i>	
5.	15-09-2006	Melengkapi Bab Pengujian Alat Dengan Menambahkan Tabel Hasil Pengujian Pembacaan <i>RFID</i> .	
6.	16-09-2006	Memperbaiki Kesimpulan Dengan Menambahkan Hasil Perencanaan Dan Pengujian Alat.	
7.	18-09-2006	Periksa Makalah Seminar Hasil, Menambahkan Gambar Rangkaian Keseluruhan	
8.	22-09-2006	Melengkapi Batasan Masalah Sesuai Dengan Revisi Pada Saat Seminar Hasil	
9.	23-09-2006	ACC Keseluruhan Bab.	
10.	25-09-2006	Penandatanganan Lembar Persetujuan	

Malang, September 2006
Dosen Pembimbing,

Ir. Yusuf Ismail N., MT
Nip. Y.101 8800 189



Tide		
Rangkaian Keseluruhan		
Size	Number	Revision
B		
Date:	24-Sep-2005	Sheet of
File:	H:\Dony\Dony.ddt	Drawn By: Dony Mahendra

doni

```
dispclear      equ      00000001b
funcset        equ      00111000b
entrmod        equ      00000110b
dispon         equ      00001100b
cursor         equ      00001110b
blink          equ      00001101b
E              equ      p1.1
RS             equ      p1.0
con_lcd        equ      p1.2
data_lcd       equ      p1.3
tombo1         equ      p2.0
tambah         equ      20h
rfid0          equ      21h
rfid1          equ      22h
rfid2          equ      23h
rfid3          equ      24h
rfid4          equ      25h
rfid5          equ      26h
rfid6          equ      27h
rfid7          equ      28h
buffer1        equ      29h
buffer2        equ      2ah
buffer3        equ      2bh
low            equ      2ch
high           equ      2dh
hitmem         equ      2eh
hasil0         equ      65h
hasil1         equ      66h
hasil2         equ      67h
hasil3         equ      68h
hasil4         equ      69h
hasil5         equ      6ah
hasil6         equ      6bh
hasil7         equ      6ch
hasil8         equ      6dh
hasil9         equ      6eh
CS             equ      P1.4
Buzer          equ      p2.5

wmcon  data    96h
eemen  equ     00001000b
eemwe  equ     00010000b
wdtrst equ     00000010b
alamat equ     0000h
```

```
=====
;
;                               Mapping Ram Internal
;
=====
```

```
Data1      Equ      30h      ;Data UNTUK OPERASI ARITMATIKA
Data2      Equ      31h
Data3      Equ      32h
Data4      Equ      33h
Data5      Equ      34h
Data6      Equ      35h
Data7      Equ      36h
Data8      Equ      37h
Data9      Equ      38h
Data10     Equ      39h
Data11     Equ      3Ah
Data12     Equ      3Bh
Data13     Equ      3Ch
Data14     Equ      3Dh
Data15     Equ      3Eh
Data16     Equ      3Fh
Data17     Equ      40h
```

doni

```
Data18      Equ      41h
Data19      Equ      42h
Data20      Equ      43h
Data21      Equ      44h
Data22      Equ      45h
Data23      Equ      46h
Cacah       Equ      47h
Buffer_Data1 Equ      48h
Buffer_Data2 Equ      49h
Buffer_Data3 Equ      4Ah
Buffer_Data4 Equ      4Bh
Buffer_Data5 Equ      4Ch
DataTegR    Equ      4Dh
DataTegS    Equ      4Eh
DataTegT    Equ      4Fh
DataArsR    Equ      50h
DataArsS    Equ      51h
DataArsT    Equ      52h
SudutR     Equ      53h
SudutS     Equ      54h
SudutT     Equ      55h
Cosphi     Equ      56h
RButton    Equ      57h
Milsec     Equ      58h
NomTam     Equ      59h
TimeRHi    Equ      5Ah
TimeRLo    Equ      5Bh
TimeSHi    Equ      5Ch
TimeSLo    Equ      5Dh
TimeTHi    Equ      5Eh
TimeTLo    Equ      5Fh
Total1     Equ      60h
Total2     Equ      61h
Total3     Equ      62h
Total4     Equ      63h
Total5     Equ      64h
```

```
org      0h
jmp      mulai
crg      23h      ; alamat serial interupt
ljmp    serial
```

serial:

```
tele1:  cjne a,#'?',H1
         mov r0,#16
         mov r1,#2
         ljmp akhir
H1:     cjne a,#'$',H_1
         lcall busek
         mov r5,#1
         lcall baris1
         ;mov r0,#16
         ljmp akhir
H_1:   cjne a,#'%',H_2
         lcall dly1000ms
         lcall busek
         mov r0,#17
         ljmp akhir
H_2
         cjne r0,#16,tele2
         cjne r1,#5,h10
         jmp h13
h10:
```

doni

```
    mov r1,#4
    cjne a,#30h,h12
    mov a,#20h
    jmp h13
h12:  mov r1,#5
h13:  lcall dataout

H2:   ljmp akhir
tele2: cjne a,#'G',tele3
      mov r1,#3
      ljmp akhir
tele3:
      cjne a,#2,sel1
      mov r1,#21h
      inc r0
      jmp akhir
sel1:  cjne r0,#1,sel2
      inc r0
      jmp akhir
sel2:  cjne r0,#2,sel3
      inc r0
      ;lcall mbalik
      jmp akhir
sel3:  lcall atoh
      mov @r1,a
      mov a,#11
      subb a,r0
      jc sel4
      inc r0
      inc r1
      jmp akhir
sel4:  cjne r0,#15,sel5
      mov r1,#1
      jmp akhir
sel5:  inc r0
akhir: pop acc
      reti
```

;===program utama

mulai:

```
;clr    cs
setb    EA
;mov    TMOD,#20h
mov     TH1,#0fdh ;
setb    TR1
mov     SCON,#50h
setb    ES
```

```
setb buzer
lcall reset
lcall initlcd
mov dptr,#0000h
```

mulail:

```
clr cs ;setb cs
mov r0,#0
mov buffer3,#0 ;===*
```

awal:

```
mov r5,#1
mov dptr,#rental1
lcall cetak1
mov dptr,#rental2
lcall cetak2
```

awal0:

doni

```
    cjne r1,#1,janu1
    lcall busek
    mov r5,#1
    mov dptr,#id
    lcall cetak1
    lcall tampilcd
    lcall mpw
    ;lcall tampilkom
    mov r1,#0
    mov r0,#0
    lcall dly1000ms
    lcall dly1000ms
janu1:  jmp awal

    jb tombol,awal0
    jnb tombol,$
    setb cs ;clr cs
    lcall dly1000ms
    lcall tampilkom
    clr cs
jagir:  jmp awal0

;==subprogram

enterkom:
    mov r1,#65h
    mov a,r4
    orl a,#30h
    lcall kirimser
enterkom0:
    mov r3,#4
enterkom1:
    mov a,@r1
    lcall kirimser
    inc r1
    djnz r3,enterkom1
    djnz r4,enterkom2
    mov a,#5ah
    lcall kirimser
    ret
enterkom2:
    mov a,#2ch
    lcall kirimser
    jmp enterkom0

enter:  mov r3,#4
        mov r4,#0
        mov r1,#65h
enter1: lcall busek
        mov r5,#1
        mov dptr,#kode
        lcall cetak1

enter12:
    lcall key
    mov @r1,a
    lcall dataout
    inc r1
    djnz r3,enter12
enter2: lcall key
        cjne a,#46h,enter3
        inc r4
        mov r3,#4
        jmp enter1
enter3: ;lcall key
        cjne a,#45h,enter2
```


doni

```
inc r4
lcall busek
mov r5,#1
mov dptr,#selesai1
lcall cetak1
mov dptr,#selesai2
lcall cetak2
ret
```

tampilkom:

```
mov r1,#65h
mov dptr,#0000h
```

tampilkom0:

```
orl    wmcon,#eemen
movx a,@dptr
cjne a,#'Z',tampilak ;tampilkom2
lcall kirimser
```

```
orl    wmcop,#eemen
orl    wmcon,#eemwe
mov a,#'0'
movx @dptr,a
xrl    wmcon,#eemwe
lcall dly10ms
```

```
inc dptr
mov r3,#10
```

tampilkom1:

```
orl    wmcop,#eemen
movx a,@dptr
lcall kirimser
inc dptr
djnz r3,tampilkom1
sjmp tampilkom0
```

tampilkom2:

```
mov dptr,#0000h
orl    wmcon,#eemcon
orl    wmcon,#eemwe
mov a,#'0'
movx @dptr,a
inc dptr
xrl    wmcon,#eemwe
lcall dly10ms
```

tampilak:

```
mov a,#'%'
lcall kirimser
ret
```

lompat11:

```
mov r4,#11
```

lompat111:

```
inc dptr
djnz r4,lompat111
ret
```

mpw:

```
mov r0,#65h
mov r3,#10
mov dptr,#0000h
orl    wmcon,#eemen
movx a,@dptr
cjne a,#'Z',isiid
mpw0: lcall lompat11
orl    wmcon,#eemen
movx a,@dptr
cjne a,#'Z',isiid
sjmp mpw0
```

isiid:

```

                                doni
                                orl     wmcon,#eemen
                                orl     wmcon,#eemwe
                                mov a,#'Z'
                                movx @dptr,a
                                inc dptr
                                xrl     wmcon,#eemwe
                                lcall dly10ms
mpw1:
                                orl     wmcon,#eemen
                                orl     wmcon,#eemwe
                                mov a,@r0
                                movx @dptr,a
                                inc dptr
                                inc r0
                                xrl     wmcon,#eemwe
                                lcall dly10ms
                                djnz r3,mpw1
                                ret
key:
                                mov p2,#0feh
                                jb p2.4,key1
                                mov a,#31h
                                jnb p2.4,$
                                ljmp keyakhir
key1:
                                jb p2.5,key2
                                mov a,#32h
                                jnb p2.5,$
                                ljmp keyakhir
key2:
                                jb p2.6,key3
                                mov a,#33h
                                jnb p2.6,$
                                ljmp keyakhir
key3:
                                jb p2.7,key4
                                mov a,#41h
                                jnb p2.7,$
                                ljmp keyakhir
key4:
                                mov p2,#1111101b
                                jb p2.4,key5
                                mov a,#34h
                                jnb p2.4,$
                                ljmp keyakhir
key5:
                                jb p2.5,key6
                                mov a,#35h
                                jnb p2.5,$
                                ljmp keyakhir
key6:
                                jb p2.6,key7
                                mov a,#36h
                                jnb p2.6,$
                                ljmp keyakhir
key7:
                                jb p2.7,key8
                                mov a,#42h
                                jnb p2.7,$
                                ljmp keyakhir
key8:
                                mov p2,#11111011b
                                jb p2.4,key9
                                mov a,#37h
                                jnb p2.4,$

```

doni

```
key9:    ljmp keyakhir
         jb p2.5,key10
         mov a,#38h
         jnb p2.5,$
         ljmp keyakhir
key10:   jb p2.6,key11
         mov a,#39h
         jnb p2.6,$
         ljmp keyakhir
key11:   jb p2.7,key12
         mov a,#43h
         jnb p2.7,$
         ljmp keyakhir

key12:   mov p2,#11110111b
         jb p2.4,key13
         mov a,#44h
         jnb p2.4,$
         ljmp keyakhir
key13:   jb p2.5,key14
         mov a,#30h
         jnb p2.5,$
         ljmp keyakhir
key14:   jb p2.6,key15
         mov a,#45h
         jnb p2.6,$
         ljmp keyakhir
key15:   jb p2.7,key16
         mov a,#46h
         jnb p2.7,$
         ljmp keyakhir

key16:

keyakhir: ljmp key
         ret

kirimsr: clr     ES
         mov    sbuf,A
         jnb   ti,$           ; kirim lewat serial comm.
         clr   ti
         setb  ES
         ret

busek:   mov r5,#1
         mov dptr,#kos
         lcall cetak1
         mov dptr,#kos
         lcall cetak2
         ret

atoh:    cjne a,#41h,atoh1
         mov a,#0ah
         jmp atohakhir
atoh1:   cjne a,#42h,atoh2
         mov a,#0bh
         jmp atohakhir
atoh2:   cjne a,#43h,atoh3
```

doni

```
    mov a,#0ch
    jmp atohakhir
atoh3:  cjne a,#44h,atoh4
        mov a,#0dh
        jmp atohakhir
atoh4:  cjne a,#45h,atoh5
        mov a,#0eh
        jmp atohakhir
atoh5:  cjne a,#46h,atoh6
        mov a,#0fh
        jmp atohakhir
atoh6:  anl a,#0fh
atohakhir:
        ret

reset:  clr     A
        mov    Data1,A
        mov    Data2,A
        mov    Data3,A
        mov    Data4,A
        mov    Data5,A
        ret

tampilcd:
        mov buffer1,rfid7
        mov buffer2,rfid6
        lcall gabung
        mov data5,a

        mov buffer1,rfid5
        mov buffer2,rfid4
        lcall gabung
        mov data4,a

        mov buffer1,rfid3
        mov buffer2,rfid2
        lcall gabung
        mov data3,a

        mov buffer1,rfid1
        mov buffer2,rfid0
        lcall gabung
        mov data2,a
        mov data1,#0

        lcall Todecim48

        mov a,data17
        lcall pisah
        mov hasil0,buffer1
        mov hasil1,buffer2

        mov a,data18
        lcall pisah
        mov hasil2,buffer1
        mov hasil3,buffer2

        mov a,data19
        lcall pisah
        mov hasil4,buffer1
        mov hasil5,buffer2

        mov a,data20
        lcall pisah
        mov hasil6,buffer1
        mov hasil7,buffer2

        mov a,data21
```

doni

```
lcall pisah
mov hasil8,buffer1
mov hasil9,buffer2
```

```
ret
```

```
isi:  orl    wmcon,#eemen
      orl    wmcon,#eemwe
      mov  a,#'Z'
      movx @dptr,a
      inc  dptr
      xrl    wmcon,#eemwe
      ret
```

```
pisah: mov  b,#10h
       div  ab
       mov  buffer3,b
       orl  a,#30h
       mov  buffer1,a
       lcall dataout
       mov  a,buffer3
       orl  a,#30h
       mov  buffer2,a
       lcall dataout
       ret
```

```
gabung: mov  b,#10h
        mov  a,buffer2
        mul  ab
        add  a,buffer1
        ret
```

```
-----
; Konversi Data Hexa 5 byte ke 6 Byte Desimal
; Input       : Data1 - Data5      Msb - Lsb
; Output      : Data16 - Data21    Msb - Lsb
-----
```

```
Todecim48      Mov    Cacah,#11
Bagi10         Clr    A
              Mov    Data6,A
              Mov    Data7,A
              Mov    Data8,A
              Mov    Data9,A
              Mov    Data10,#10
              Call   bagi48
              Mov    Data16,Data5
              Dec    Cacah
              Mov    A,Cacah
              Cjne   A,#0,Putar
              Ret

Putar          Clr    C
              Mov    R7,#4
Putar2        Mov    A,Data16
              RRC    A
              Mov    Data16,A
              Mov    A,Data17
              RRC    A
              Mov    Data17,A
              Mov    A,Data18
              RRC    A
              Mov    Data18,A
              Mov    A,Data19
              RRC    A
              Mov    Data19,A
              Mov    A,Data20
              RRC    A
```

```

                                doni
Mov     Data20,A
MOV     A,Data21
RRC     A
MOV     Data21,A
Djnz   R7,Putar2
Mov     Data5,Data15
MOV     Data4,Data14
MOV     Data3,Data13
MOV     Data2,Data12
MOV     Data1,Data11
Jmp     Bagi10

```

```

;*****
;*****

```

```

; Pembagian 48 BIT (5 BYTE)
; Input : Data1 - Data5 = Msb - Lsb Terbagi = Sisa
;         Data6 - Data10 = Msb - Lsb Pembagi
; Output: Data1 - Data5 = Msb - Lsb Sisa (Fraksi)
;         Data11 - Data15 = Msb - Lsb HasilBagi
; BUFFER: Data23 = PangkatSisa
;         R3      = PangkatBagi
;*****
;*****

```

```

Bagi48      Mov A,Data6           ; Periksa Pembagi (Msb)
            Cjne A,#00,Div48A
            Mov A,Data7
            Cjne A,#00,Div48A
            Mov A,Data8
            Cjne A,#00,Div48A
            Mov A,Data9
            Cjne A,#00,Div48A
            Mov A,Data10
            Cjne A,#00,Div48A
Div48Z      Mov A,#0FFH           ;Terjadi Pembagian oleh
no1

```

```

            Mov Data11,A          ;Buat Hasil = $FFFFFF
            Mov Data12,A          ; Fraksi = $FFFFFF

```

(TAK TERHINGGA)

```

Mov Data13,A
MOV Data14,A
MOV Data15,A
MOV Data1,A
MOV Data2,A
MOV Data3,A
MOV Data4,A
MOV Data5,A
Ret

```

```

Div48A      Mov Data11,#0         ; Awal, Hasil = 0
            Mov Data12,#0
            Mov Data13,#0
            Mov Data14,#0
            Mov Data15,#0
            Mov R3,#0             ; Pangkat Pembagi (M)
            Mov Data23,#0        ; Pangkat Sisa (N) atau

```

```

selisih Pangkat (Q)
Div48B      Call Cmp64           ; Banding Sisa DG
Pembagi

```

```

Bagi;Lanjut Jnc Div48B1           ; Jika Sisa >=

```

```

Pulang      Ret                 ; Jika Sisa < Bagi,

```

```

;PENYESUAIAN Pangkat (UNTUK MENGHINDARI
PENGURANGAN BERULANG YANG_
;TERLALU PANJANG

```

```

Div48B1     Mov A,Data6           ;Cek Bagi
            Cjne A,#80H,B1o1      ;

```

```

doni
B1o1          Jc      Div48C          ;Jika Bagi < $800000,
Lanjut
Div48C        Sjmp    Div48F
              Mov     A,Data6          ;Cek Bagi
              Cjne   A,#40H,B1o2      ;
              Jc     Div48E          ;Jika Bagi < $400000,
B1o2
KE Div48E
Div48D        Mov     A,Data1
              JB     ACC.7,Div48F      ;Cek Sisa, Jika BIT-7 =
1, Lanjut
              Call   GsrSisa          ;Sisa = Sisa * 2
              Inc    Data23           ;N = N + 1 (PangkatSisa
+ 1)
              Mov    A,Data23
              Cjne   A,#48,Div48D      ;Sisa = 0
              Mov    Data1,#0
              Mov    Data2,#0
              Mov    Data3,#0
              Mov    Data4,#0
              Mov    Data5,#0
Div48E        Ret
              Clr    C
              Mov    A,Data10
              RLC    A
              Mov    Data10,A          ;
              Mov    A,Data9
              RLC    A
              Mov    Data9,A          ;
              Mov    A,Data8
              RLC    A
              Mov    Data8,A          ;
              Mov    A,Data7
              RLC    A
              Mov    Data7,A          ;
              Mov    A,Data6
              RLC    A
              Mov    Data6,A          ;
              Inc    R3                ;M = M + 1
(PangkatPembagi + 1)
              Mov    A,R3
              Cjne   A,#47,Div48C
Div48F        Jmp     Div48Z          ;terbagi oleh nol
              Mov    A,R3              ;Operasi Pangkat
Bhs1          Cjne   A,Data23,Bhs1    ;Jika M >= N, Lanjut
              Jnc    Div48F1
              Clr    C
              Mov    A,Data1
              RRC    A
              Mov    Data1,A          ;Jika M < N, Sisa =
Sisa/2
              Mov    A,Data2
              RRC    A
              Mov    Data2,A          ;[0] -> [DT1] -> [DT2] ->
[DT1]
              Mov    A,Data3
              RRC    A
              Mov    Data3,A
              Mov    A,Data4
              RRC    A
              Mov    Data4,A
              Mov    A,Data5
              RRC    A
              Mov    Data5,A
              Dec    Data23
              Sjmp   Div48F          ;Ulang sampai diperoleh M
              Clr    C
              Subb  A,Data23          ;Selisih Pangkat = M - N
=> N
Div48F1

```

```

                                doni
Div48G      Mov    Data23,A          ;Simpan Sbg Q
            Call   Cmp64          ;Banding Sisa DG Bagi
            Jc     Div48L         ;Jika Sisa < Bagi, KE L
Div48I      Clr     A
            Inc    Data15         ;Hasil = Hasil + 1
            Mov    A,Data15
            Jnz    Div48K
            Inc    Data14
            Mov    A,Data14
            Jnz    Div48K
            Inc    Data13
            Mov    A,Data13
            Jnz    Div48K
            Inc    Data12
            Mov    A,Data12
            Jnz    Div48K
            Inc    Data11
Div48K      Clr     C
            Mov    A,Data5         ;Sisa = Sisa - Bagi
            Subb   A,Data10
            Mov    Data5,A
            Mov    A,Data4
            Subb   A,Data9
            Mov    Data4,A
            Mov    A,Data3
            Subb   A,Data8
            Mov    Data3,A
            Mov    A,Data2
            Subb   A,Data7
            Mov    Data2,A
            Mov    A,Data1
            Subb   A,Data6
            Mov    Data1,A
            Sjmp   Div48G         ;ulang
Div48L      ;
            ;UNTUK Sisa < Pembagi
            Mov    A,Data23       ;Periksa Q
            Cjne   A,#00,OraNo11  ;Q = 0 ?
            Sjmp   Div48M         ;Jika YA, KE M
OraNo11    Call   GsrSisa        ;Sisa = Sisa * 2
            Clr    C
            Mov    A,Data15
            RLC    A
            Mov    Data15,A       ;Hasil = Hasil * 2
(GESER KIRI)
            Mov    A,Data14
            RLC    A
            Mov    Data14,A
            Mov    A,Data13
            RLC    A
            Mov    Data13,A
            Mov    A,Data12
            RLC    A
            Mov    Data12,A
            Mov    A,Data11
            RLC    A
            Mov    Data11,A
            Dec    Data23
Div48M      Sjmp   Div48G         ;Ulang
            Mov    A,R3           ;Periksa PangkatPembagi
            Cjne   A,#00,Noteq    ;Jika Pangkat Pembagi =
0, USAI    Ret
Noteq      Clr     C
            Mov    A,Data1
            RRC    A
            Mov    Data1,A       ;Sisa = Sisa / 2
            Mov    A,Data2

```


doni

```

RRC    A
MOV    Data2,A
MOV    A,Data3
RRC    A
MOV    Data3,A
MOV    A,Data4
RRC    A
MOV    Data4,A
MOV    A,Data5
RRC    A
MOV    Data5,A
Dec    R3
Sjmp  Div48M
DivEnd Ret

GsrSisa    Clr    C
           Mov    A,Data5
           RLC    A
           Mov    Data5,A           ;Sisa = Sisa * 2 (GESER
KIRI)      Mov    A,Data4
           RLC    A
           Mov    Data4,A
           Mov    A,Data3
           RLC    A
           Mov    Data3,A
           Mov    A,Data2
           RLC    A
           Mov    Data2,A
           Mov    A,Data1
           RLC    A
           Mov    Data1,A
           Ret

Cmp64      Mov    A,Data1           ;Banding Sisa DG Pembagi
           Cjne  A,Data6,DivE
           Mov    A,Data2
           Cjne  A,Data7,DivE
           Mov    A,Data3
           Cjne  A,Data8,DivE
           Mov    A,Data4
           Cjne  A,Data9,DivE
           Mov    A,Data5
           Cjne  A,Data10,DivE
DivE:      Ret
;=====routine lcd
baris2:    mov a,r5
           add a,#0c0h ;11000000b
           sjmp posisisub
baris1:    mov a,r5
           add a,#80h; 10000000b
posisisub: dec a
           lcall controlout
           ret
cetak2:    lcall baris2
           sjmp ansa
cetak1:    lcall baris1
ansa:      sjmp outstring
loop:      lcall dataout
           inc dptr
outstring:
```

doni

```
    clr a
    movc a,@a+dptr
    cjne a,#'$',loop
    ret
controlout:
    push dph
    push dpl
    clr rs
    sjmp out
dataout:
    push dph
    push dpl
    setb rs
out:
    setb e
    lcall geser
    mov r6,#250
    djnz r6,$
    pop dpl
    pop dph
    clr e
    ret

geser:
    clr con_lcd
    mov  b,#8
a1:
    rrc a
    mov data_lcd,c
    nop
    nop
    nop
    nop
    setb con_lcd
    nop
    nop
    nop
    nop
    clr con_lcd
    djnz b,a1
    ret
delay:
    mov r6,#00h
dlylcdlp:
    mov r7,#00h
    djnz r7,$
    djnz r6,dlylcdlp
    ret
ldelay:
ldelay2:
    mov r3,#00h
ldelay1:
    lcall tdelay
    djnz r3,ldelay1
    djnz r2,ldelay2
    ret
tdelay:
    mov r7,#00h
    djnz r7,$
    ret
initlcd:
    mov a,#dispclear
    lcall controlout
    lcall delay
    mov a,#funcset
    lcall controlout
    mov a,#dispon
    lcall controlout
```

doni

```
    mov a,#entrmod
    lcall controlout
    ret

;-- Routine delay --
d50ms:  push    tmod
        mov    tmod,#21h
        mov    th0,#03fh ;3cb0
        mov    tl0,#0f0h
        setb  tr0
t50ms:  jbc    tf0,sudah
        ajmp  t50ms
sudah:  clr    tr0
        pop   tmod
        ret

dldet:  mov    r7,#20
dldet1: call  d50ms
        djnz  r7,dldet1
        ret

;=====
;rutin delay
;=====
dly1000ms:
    mov    r2,#10
U1De1:  lcall  dly100ms
        djnz  r2,U1De1
        ret
dly100ms:
    mov    r6,#200
    ljmp  waktu
dly10ms:
    mov    r6,#20
    ljmp  waktu

; Timer 0,5 mS
; R6 = perkalian
waktu:  mov    r7,#247
        djnz  r7,$
        djnz  r6,waktu
        ret

;=====
kos:
    db    '          $'
RENTAL1:
    db    '    ID MOBIL    $'
RENTAL2:
    db    '    ITN MALANG    $'
```

end

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, Menus, Grids, DBGrids, DB, ADODB, ExtCtrls, StdCtrls, DBCtrls,
CPort, jpeg;

type

```
TForm1 = class(TForm)
  Notebook1: TNotebook;
  ADOConnectiondata: TADOConnection;
  ADOQuerydata: TADOQuery;
  DataSource1: TDataSource;
  DBGrid1: TDBGrid;
  MainMenu1: TMainMenu;
  file1: TMenuItem;
  close1: TMenuItem;
  Label1: TLabel;
  Label2: TLabel;
  lbkode1: TLabel;
  Label3: TLabel;
  lbkode2: TLabel;
  Label5: TLabel;
  lbkode3: TLabel;
  Label4: TLabel;
  lbkode4: TLabel;
  lbkode5: TLabel;
  Label6: TLabel;
  Label9: TLabel;
  Label10: TLabel;
  Label11: TLabel;
  Label12: TLabel;
  Label7: TLabel;
  Label13: TLabel;
  Label14: TLabel;
  Label15: TLabel;
  Label16: TLabel;
  Label17: TLabel;
  Label18: TLabel;
  Label19: TLabel;
  Label20: TLabel;
  Label21: TLabel;
  Label22: TLabel;
  Label23: TLabel;
  Label24: TLabel;
  Label25: TLabel;
  Label26: TLabel;
  Label27: TLabel;
```

btnsimpan: TButton;
btnkembali: TButton;
edkoderfid: TEdit;
Ednamapemilik: TEdit;
ednomorpol: TEdit;
Edmerek: TEdit;
Edmodel: TEdit;
Edthbuat: TEdit;
Edthrakit: TEdit;
Edwarna: TEdit;
Ednorangka: TEdit;
Ednomesin: TEdit;
Edbhnbakar: TEdit;
Edkodelokasi: TEdit;
Edmasaberlaku: TEdit;
edalamat: TEdit;
btnlihatdatapemilik: TButton;
Button3: TButton;
utama1: TMenuItem;
tabel1: TMenuItem;
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Button1: TButton;
Button2: TButton;
DBNavigator1: TDBNavigator;
ADOConnection1: TADOConnection;
ADOQuery1: TADOQuery;
ADOConnection2: TADOConnection;
ADOQuery2: TADOQuery;
lbciptat: TLabel;
button: TImage;
GroupBox1: TGroupBox;
Label28: TLabel;
edbt: TEdit;
buton: TButton;
lbpassciptat: TLabel;
Image1: TImage;
Label8: TLabel;
procedure btnlihatdatapemilikClick(Sender: TObject);
procedure btnkembaliClick(Sender: TObject);
procedure btnsimpanClick(Sender: TObject);
procedure close1Click(Sender: TObject);
procedure utar.1a1Click(Sender: TObject);
procedure tabel1Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure Button2Click(Sender: TObject);

```
procedure ComPort1RxChar(Sender: TObject; Count: Integer);
procedure Button3Click(Sender: TObject);
procedure butonClick(Sender: TObject);
procedure buttonClick(Sender: TObject);
```

```
private
  { Private declarations }
public
  { Public declarations }
end;
```

```
var
  Form1: TForm1;
  i1,i2:integer;
  dtserial,s1:string;
```

```
implementation
```

```
uses Unit5, Unit3;
```

```
{$R *.dfm}
```

```
procedure TForm1.btnlihatdatapemilikClick(Sender: TObject);
begin
  inc(i1);
```

```
case i1 of
```

```
1 :begin
```

```
  edkoderfid.Text:=lbkode1.Caption;
  ADOQuerydata.Close;
  ADOQuerydata.SQL.Clear;
  ADOQuerydata.SQL.Add('select * from dt where
kodeRFID='+QuotedStr(lbkode1.Caption));
  ADOQuerydata.Open;
  if
```

```
(lbkode1.Caption=ADOQuerydata.FieldName('kodeRFID').AsString)and(lbkode1.
Caption<>") then
```

```
  begin
```

```
    Ednamapemilik.Text:=ADOQuerydata.FieldName('nama_pemilik').AsString;
    edalamat.Text:=ADOQuerydata.FieldName('alamat').AsString;
    ednomorpol.Text:=ADOQuerydata.FieldName('nomor_polisi').AsString;
    Edmerek.Text:=ADOQuerydata.FieldName('merek/tipe').AsString;
    Edmodel.Text:=ADOQuerydata.FieldName('model').AsString;
    Edthbuat.Text:=ADOQuerydata.FieldName('th_pembuatan').AsString;
    Edthraktit.Text:=ADOQuerydata.FieldName('th_perakitan').AsString;
    Edwarna.Text:=ADOQuerydata.FieldName('warna').AsString;
    Ednorangka.Text:=ADOQuerydata.FieldName('nomor_rangka').AsString;
    Ednomesin.Text:=ADOQuerydata.FieldName('nomor_mesin').AsString;
    Edbhnbakar.Text:=ADOQuerydata.FieldName('bahan_bakar').AsString;
```

```
Edkodelokasi.Text:=ADOQuerydata.FieldByName('kode_lokasi').AsString;
Edmasaberlaku.Text:=ADOQuerydata.FieldByName('masa_berlaku').AsString;
end;
Notebook1.ActivePage:='data kdm';
end;
```

2 :begin

```
edkoderfid.Text:=lbkode2.Caption;
ADOQuerydata.Close;
ADOQuerydata.SQL.Clear;
ADOQuerydata.SQL.Add('select * from dt where
kodeRFID='+QuotedStr(lbkode2.Caption));
ADOQuerydata.Open;
```

if

```
(lbkode2.Caption=ADOQuerydata.FieldByName('kodeRFID').AsString)and(lbkode2.
Caption<>'') then
```

begin

```
Ednamapemilik.Text:=ADOQuerydata.FieldByName('nama_pemilik').AsString;
edalamat.Text:=ADOQuerydata.FieldByName('alamat').AsString;
ednomorpol.Text:=ADOQuerydata.FieldByName('nomor_polisi').AsString;
Edmerek.Text:=ADOQuerydata.FieldByName('merek/tipe').AsString;
Edmodel.Text:=ADOQuerydata.FieldByName('model').AsString;
Edthbuat.Text:=ADOQuerydata.FieldByName('th_pembuatan').AsString;
Edthraktit.Text:=ADOQuerydata.FieldByName('th_perakitan').AsString;
Edwarna.Text:=ADOQuerydata.FieldByName('warna').AsString;
Ednorangka.Text:=ADOQuerydata.FieldByName('nomor_rangka').AsString;
Ednomesin.Text:=ADOQuerydata.FieldByName('nomor_mesin').AsString;
Edbhnbakar.Text:=ADOQuerydata.FieldByName('bahan_bakar').AsString;
Edkodelokasi.Text:=ADOQuerydata.FieldByName('kode_lokasi').AsString;
Edmasaberlaku.Text:=ADOQuerydata.FieldByName('masa_berlaku').AsString;
```

end;

```
Notebook1.ActivePage:='data kdm';
```

end;

3 :begin

```
edkoderfid.Text:=lbkode3.Caption;
ADOQuerydata.Close;
ADOQuerydata.SQL.Clear;
ADOQuerydata.SQL.Add('select * from dt where
kodeRFID='+QuotedStr(lbkode3.Caption));
ADOQuerydata.Open;
```

if

```
(lbkode3.Caption=ADOQuerydata.FieldByName('kodeRFID').AsString)and(lbkode3.
Caption<>'') then
```

begin

```
Ednamapemilik.Text:=ADOQuerydata.FieldByName('nama_pemilik').AsString;
edalamat.Text:=ADOQuerydata.FieldByName('alamat').AsString;
ednomorpol.Text:=ADOQuerydata.FieldByName('nomor_polisi').AsString;
Edmerek.Text:=ADOQuerydata.FieldByName('merek/tipe').AsString;
Edmodel.Text:=ADOQuerydata.FieldByName('model').AsString;
Edthbuat.Text:=ADOQuerydata.FieldByName('th_pembuatan').AsString;
Edthraktit.Text:=ADOQuerydata.FieldByName('th_perakitan').AsString;
```

```

Edwarna.Text:=ADOQuerydata.FieldByName('warna').AsString;
Ednorangka.Text:=ADOQuerydata.FieldByName('nomor_rangka').AsString;
Ednomesin.Text:=ADOQuerydata.FieldByName('nomor_mesin').AsString;
Edbhnbakar.Text:=ADOQuerydata.FieldByName('bahan_bakar').AsString;
Edkodelokasi.Text:=ADOQuerydata.FieldByName('kode_lokasi').AsString;
Edmasaberlaku.Text:=ADOQuerydata.FieldByName('masa_berlaku').AsString;
end;
Notebook1.ActivePage:='data kdm';
end;
4 :begin
edkoderfid.Text:=lbkode4.Caption;
ADOQuerydata.Close;
ADOQuerydata.SQL.Clear;
ADOQuerydata.SQL.Add('select * from dt where
kodeRFID='+QuotedStr(lbkode4.Caption));
ADOQuerydata.Open;
if
(lbkode4.Caption=ADOQuerydata.FieldByName('kodeRFID').AsString)and(lbkode4.
Caption<>"") then
begin
Ednamapemilik.Text:=ADOQuerydata.FieldByName('nama_pemilik').AsString;
edalamat.Text:=ADOQuerydata.FieldByName('alamat').AsString;
ednomorpol.Text:=ADOQuerydata.FieldByName('nomor_polisi').AsString;
Edmerek.Text:=ADOQuerydata.FieldByName('merek/tipe').AsString;
Edmodel.Text:=ADOQuerydata.FieldByName('model').AsString;
Edthbuat.Text:=ADOQuerydata.FieldByName('th_pembuatan').AsString;
Edthraktit.Text:=ADOQuerydata.FieldByName('th_perakitan').AsString;
Edwarna.Text:=ADOQuerydata.FieldByName('warna').AsString;
Ednorangka.Text:=ADOQuerydata.FieldByName('nomor_rangka').AsString;
Ednomesin.Text:=ADOQuerydata.FieldByName('nomor_mesin').AsString;
Edbhnbakar.Text:=ADOQuerydata.FieldByName('bahan_bakar').AsString;
Edkodelokasi.Text:=ADOQuerydata.FieldByName('kode_lokasi').AsString;
Edmasaberlaku.Text:=ADOQuerydata.FieldByName('masa_berlaku').AsString;
end;
Notebook1.ActivePage:='data kdm';
end;
5 :begin
edkoderfid.Text:=lbkode5.Caption;
ADOQuerydata.Close;
ADOQuerydata.SQL.Clear;
ADOQuerydata.SQL.Add('select * from dt where
kodeRFID='+QuotedStr(lbkode5.Caption));
ADOQuerydata.Open;
if
(lbkode5.Caption=ADOQuerydata.FieldByName('kodeRFID').AsString)and(lbkode5.
Caption<>"") then
begin
Ednamapemilik.Text:=ADOQuerydata.FieldByName('nama_pemilik').AsString;
edalamat.Text:=ADOQuerydata.FieldByName('alamat').AsString;
ednomorpol.Text:=ADOQuerydata.FieldByName('nomor_polisi').AsString;

```



```
Edmerek.Text:=ADOQuerydata.FieldByName('merek/tipe').AsString;
Edmodel.Text:=ADOQuerydata.FieldByName('mcdel').AsString;
Edthbuat.Text:=ADOQuerydata.FieldByName('th_pembuatan').AsString;
Edthrakit.Text:=ADOQuerydata.FieldByName('th_perakitan').AsString;
Edwarna.Text:=ADOQuerydata.FieldByName('warna').AsString;
Ednorangka.Text:=ADOQuerydata.FieldByName('nomor_rangka').AsString;
Ednomesin.Text:=ADOQuerydata.FieldByName('nomor_mesin').AsString;
Edbhnbakar.Text:=ADOQuerydata.FieldByName('bahan_bakar').AsString;
Edkodelokasi.Text:=ADOQuerydata.FieldByName('kode_lokasi').AsString;
Edmasaberlaku.Text:=ADOQuerydata.FieldByName('masa_berlaku').AsString;
```

```
end;
```

```
Notebook1.ActivePage:='data kdrn';
```

```
il:=0;
```

```
lbkode1.Caption:="";
```

```
lbkode2.Caption:="";
```

```
lbkode3.Caption:="";
```

```
lbkode4.Caption:="";
```

```
lbkode5.Caption:="";
```

```
end;
```

```
end;
```

```
end;
```

```
procedure TForm1.btnkembaliClick(Sender: TObject);
```

```
begin
```

```
Notebook1.ActivePage:='tampilan';
```

```
Ednamapemilik.Text:="";
```

```
edalamat.Text:="";
```

```
ednomorpol.Text:="";
```

```
Edmerek.Text:="";
```

```
Edmodel.Text:="";
```

```
Edthbuat.Text:="";
```

```
Edthrakit.Text:="";
```

```
Edwarna.Text:="";
```

```
Ednorangka.Text:="";
```

```
Ednomesin.Text:="";
```

```
Edbhnbakar.Text:="";
```

```
Edkodelokasi.Text:="";
```

```
Edmasaberlaku.Text:="";
```

```
end;
```

```
procedure TForm1.btnsimpanClick(Sender: TObject);
```

```
begin
```

```
ADOQuerydata.Close;
```

```
ADOQuerydata.SQL.Clear;
```

```
ADOQuerydata.SQL.Add('select * from dt where
```

```
kodeRFID='+Quote1Str(edkoderfid.Text));
```

```
ADOQuerydata.Open;
```

```

if
(edkoderfid.Text <> ADOQuerydata.FieldByName('kodeRFID').AsString) and (edkoder
fid.Text <> "") then
begin
ADOQuerydata.Close;
ADOQuerydata.SQL.Clear;
ADOQuerydata.SQL.Add('insert into dt values(:a,:b,:c,:d,:e,:f,:g,:h,:i,:j,:k,:l,:m,:n)');
ADOQuerydata.Parameters.ParamByName('a').Value:=edkoderfid.Text;
ADOQuerydata.Parameters.ParamByName('b').Value:=Ednamapemilik.Text;
ADOQuerydata.Parameters.ParamByName('c').Value:=edalamat.Text;
ADOQuerydata.Parameters.ParamByName('d').Value:=ednomorpol.Text;
ADOQuerydata.Parameters.ParamByName('e').Value:=Edmerek.Text;
ADOQuerydata.Parameters.ParamByName('f').Value:=Edmodel.Text;
ADOQuerydata.Parameters.ParamByName('g').Value:=Edthbuat.Text;
ADOQuerydata.Parameters.ParamByName('h').Value:=Edthrakit.Text;
ADOQuerydata.Parameters.ParamByName('i').Value:=Edwarna.Text;
ADOQuerydata.Parameters.ParamByName('j').Value:=Ednorangka.Text;
ADOQuerydata.Parameters.ParamByName('k').Value:=Ednomesin.Text;
ADOQuerydata.Parameters.ParamByName('l').Value:=Edbhnbakar.Text;
ADOQuerydata.Parameters.ParamByName('m').Value:=Edkodelokasi.Text;
ADOQuerydata.Parameters.ParamByName('n').Value:=Edmasaberlaku.Text;
ADOQuerydata.ExecSQL;
end;

```

```
end;
```

```

procedure TForm1.close1Click(Sender: TObject);
begin
close;
end;

```

```

procedure TForm1.utama1Click(Sender: TObject);
begin
Notebook1.ActivePage:='tampilan';
Ednamapemilik.Text:="";
edalamat.Text:="";
ednomorpol.Text:="";
Edmerek.Text:="";
Edmodel.Text:="";
Edthbuat.Text:="";
Edthrakit.Text:="";
Edwarna.Text:="";
Ednorangka.Text:="";
Ednomesin.Text:="";
Edbhnbakar.Text:="";
Edkodelokasi.Text:="";
Edmasaberlaku.Text:="";
end;

```

```

procedure TForm1.tabell1Click(Sender: TObject);

```

```

begin
Notebook1.ActivePage:='tblkdrm';

end;

procedure TForm1.Button1Click(Sender: TObject);
begin
lbkode1.Caption:=Edit1.Text;
lbkode2.Caption:=Edit2.Text;
lbkode3.Caption:=Edit3.Text;
lbkode4.Caption:=Edit4.Text;
lbkode5.Caption:=Edit5.Text;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
i1:=0;
i2:=0;
s1:='';
Notebook1.ActivePage:='tampilan';
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
ADOQuerydata.Close;
ADOQuerydata.SQL.Clear;
ADOQuerydata.SQL.Add('select * from dt');
ADOQuerydata.Open;
end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var
dataedit:TStringList;
begin
ComPort1.ReadStr(dtserial,Count);
//Edit1.Text:=Edit1.Text+dtserial;
s1:=s1+dtserial;
//Edit2.Text:=s1;
i2:=i2+count;
//Edit3.Text:=IntToStr(i2);

if(i2=56)then
begin
//Edit1.Text:='saya';
dataedit:=TStringList.Create;
dataedit.Clear;
dataedit.Delimiter:='Z';
dataedit.DelimitedText:=s1;

Edit1.Text:=dataedit[1];

```

```
Edit2.Text:=dataedit[2];  
Edit3.Text:=dataedit[3];  
Edit4.Text:=dataedit[4];  
Edit5.Text:=copy(dataedit[5],0,10);  
Button1.Click;  
i2:=0;  
s1:='';  
end;
```

```
//Edit3.Text:=inttostr(length(Edit1.Text));  
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);  
begin  
    lbkode1.Caption:='';  
    lbkode2.Caption:='';  
    lbkode3.Caption:='';  
    lbkode4.Caption:='';  
    lbkode5.Caption:='';  
end;
```

```
procedure TForm1.buttonClick(Sender: TObject);  
begin  
    DM5.g;  
end;
```

```
procedure TForm1.buttonClick(Sender: TObject);  
begin  
    DM3.a;  
end;
```

```
end.
```

Features

Compatible with MCS[®]51 Products
8K Bytes of In-System Reprogrammable Downloadable Flash Memory
– SPI Serial Interface for Program Downloading
– Endurance: 1,000 Write/Erase Cycles
2K Bytes EEPROM
– Endurance: 100,000 Write/Erase Cycles
1.8V to 6V Operating Range
Fully Static Operation: 0 Hz to 24 MHz
Two-level Program Memory Lock
6 x 8-bit Internal RAM
Programmable I/O Lines
Three 16-bit Timer/Counters
Six Interrupt Sources
Programmable UART Serial Channel
I²C Serial Interface
Low-power Idle and Power-down Modes
Rapid Recovery from Power-down
Programmable Watchdog Timer
Two Data Pointers
Power-off Flag

Description

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

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

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



**8-bit
Microcontroller
with 8K Bytes
Flash**

AT89S8252

**Not Recommended
for New Designs.
Use AT89S8253.**

0401G-MICRO-3/06



Features

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

Description

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

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

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



**8-bit
Microcontroller
with 8K Bytes
Flash**

AT89S8252

**Not Recommended
for New Designs.
Use AT89S8253.**

0401G-MICRO-3/06



Features

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

Description

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

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

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



**8-bit
Microcontroller
with 8K Bytes
Flash**

AT89S8252

**Not Recommended
for New Designs.
Use AT89S8253.**

0401G-MICRO-3/06





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

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

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

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

Port 2

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

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ R1), Port 2 emits the contents of the P2 Special Function Register.

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

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pull-ups.

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

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

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

RST

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

ALE/PROG

Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{\text{PROG}}$) 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 AT89S8252 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

$\overline{\text{EA}}$ /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 when 12-volt programming is selected.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.



Special Function Registers

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

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

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

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

Table 1. AT89S8252 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000					SPCR 000001XY			0D7H
0C8H	T2CON 00000000	T2MOD XXXXXXXX	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX						0AFH
0A0H	P2 11111111								0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111						WMCON 00000010		97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

Table 2. T2CON – Timer/Counter 2 Control Register

T2CON Address = 0C8H				Reset Value = 0000 0000B				
Bit Addressable								
	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T $\bar{2}$	CP/RL $\bar{2}$
Bit	7	6	5	4	3	2	1	0

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



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

Table 3. WMCON—Watchdog and Memory Control Register

WMCON Address = 96H						Reset Value = 0000 0010B		
	PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN
Bit	7	6	5	4	3	2	1	0

Symbol	Function
PS2 PS1 PS0	Prescaler Bits for the Watchdog Timer. When all three bits are set to "0", the watchdog timer has a nominal period of 16 ms. When all three bits are set to "1", the nominal period is 2048 ms.
EEMWE	EEPROM Data Memory Write Enable Bit. Set this bit to "1" before initiating byte write to on-chip EEPROM with the MOVX instruction. User software should set this bit to "0" after EEPROM write is completed.
EEMEN	Internal EEPROM Access Enable. When EEMEN = 1, the MOVX instruction with DPTR will access on-chip EEPROM instead of external data memory. When EEMEN = 0, MOVX with DPTR accesses external data memory.
DPS	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1
WDTRST RDY/BSY	Watchdog Timer Reset and EEPROM Ready/Busy Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The WDTRST bit is then automatically reset to "0" in the next instruction cycle. The WDTRST bit is Write-Only. This bit also serves as the RDY/BSY flag in Read-Only mode during EEPROM write. RDY/BSY = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/BSY bit equals "0" and is automatically reset to "1" when programming is completed.
WDTEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

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

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

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

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

Table 4. SPCR – SPI Control Register

SPCR Address = D5H				Reset Value = 0000 01XXB				
Bit	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
	7	6	5	4	3	2	1	0

Symbol	Function															
SPIE	SPI Interrupt Enable. This bit, in conjunction with the ES bit in the IE register, enables SPI interrupts: SPIE = 1 and ES = 1 enable SPI interrupts. SPIE = 0 disables SPI interrupts.															
SPE	SPI Enable. SPI = 1 enables the SPI channel and connects \overline{SS} , MOSI, MISO and SCK to pins P1.4, P1.5, P1.6, and P1.7. SPI = 0 disables the SPI channel.															
DORD	Data Order. DORD = 1 selects LSB first data transmission. DORD = 0 selects MSB first data transmission.															
MSTR	Master/Slave Select. MSTR = 1 selects Master SPI mode. MSTR = 0 selects Slave SPI mode.															
CPOL	Clock Polarity. When CPOL = 1, SCK is high when idle. When CPOL = 0, SCK of the master device is low when not transmitting. Please refer to figure on SPI Clock Phase and Polarity Control.															
CPHA	Clock Phase. The CPHA bit together with the CPOL bit controls the clock and data relationship between master and slave. Please refer to figure on SPI Clock Phase and Polarity Control.															
SPR0 SPR1	SPI Clock Rate Select. These two bits control the SCK rate of the device configured as master. SPR1 and SPR0 have no effect on the slave. The relationship between SCK and the oscillator frequency, F_{osc} , is as follows: <table border="1" style="margin-left: 20px;"> <tr> <td>SPR1</td> <td>SPR0</td> <td>SCK = F_{osc} divided by</td> </tr> <tr> <td>0</td> <td>0</td> <td>4</td> </tr> <tr> <td>0</td> <td>1</td> <td>16</td> </tr> <tr> <td>1</td> <td>0</td> <td>64</td> </tr> <tr> <td>1</td> <td>1</td> <td>128</td> </tr> </table>	SPR1	SPR0	SCK = F_{osc} divided by	0	0	4	0	1	16	1	0	64	1	1	128
SPR1	SPR0	SCK = F_{osc} divided by														
0	0	4														
0	1	16														
1	0	64														
1	1	128														



Table 5. SPSR – SPI Status Register

SPSR Address = AAH				Reset Value = 00XX XXXXB			
Bit	SPIF	WCOL	-	-	-	-	-
	7	6	5	4	3	2	1
							0

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

Table 6. SPDR – SPI Data Register

SPDR Address = 86H				Reset Value = unchanged			
Bit	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1
	7	6	5	4	3	2	1
							0

Data Memory – EEPROM and RAM

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

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

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

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

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

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

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

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

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

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

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

Programmable Watchdog Timer

The programmable Watchdog Timer (WDT) operates from an independent internal oscillator. The prescaler bits, PS0, PS1 and PS2 in SFR WMCON are used to set the period of the Watchdog Timer from 16 ms to 2048 ms. The available timer periods are shown in the following table and the actual timer periods (at $V_{CC} = 5V$) are within $\pm 30\%$ of the nominal.

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

Table 7. Watchdog Timer Period Selection

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

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S8252 operate the same way as Timer 0 and Timer 1 in the AT89C51 and AT89C52. For further information on the timers' operation, refer to the Atmel web site (<http://www.atmel.com>). From the home page, select "Products", then "Microcontrollers, then "8051-Architecture". Click on "Documentation", then on "Other Documents". Open the document "AT89 Series Hardware Description".

Timer 2

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

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

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



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

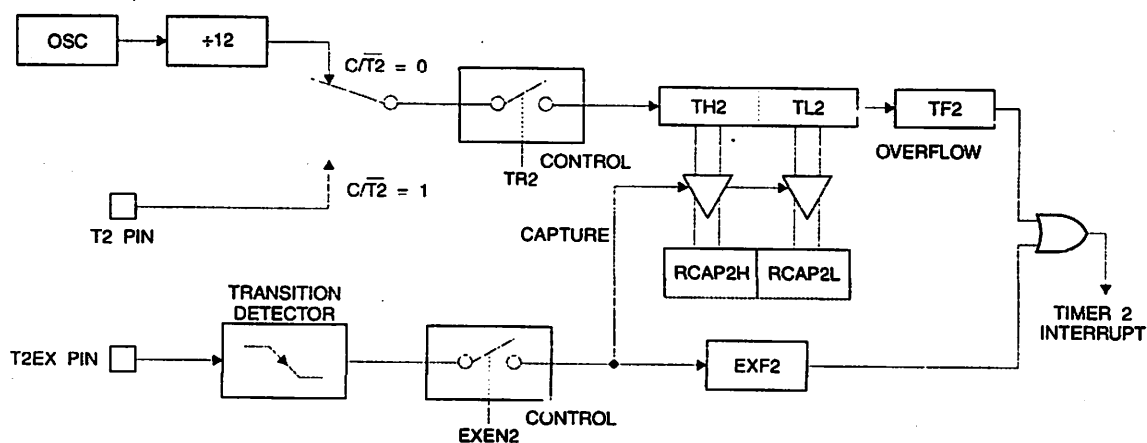
Table 8. Timer 2 Operating Modes

RCLK + TCLK	CP/RL2	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

Capture Mode

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

Figure 1. Timer 2 in Capture Mode



Auto-reload (Up or Down Counter)

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

Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in RCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

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

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

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

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

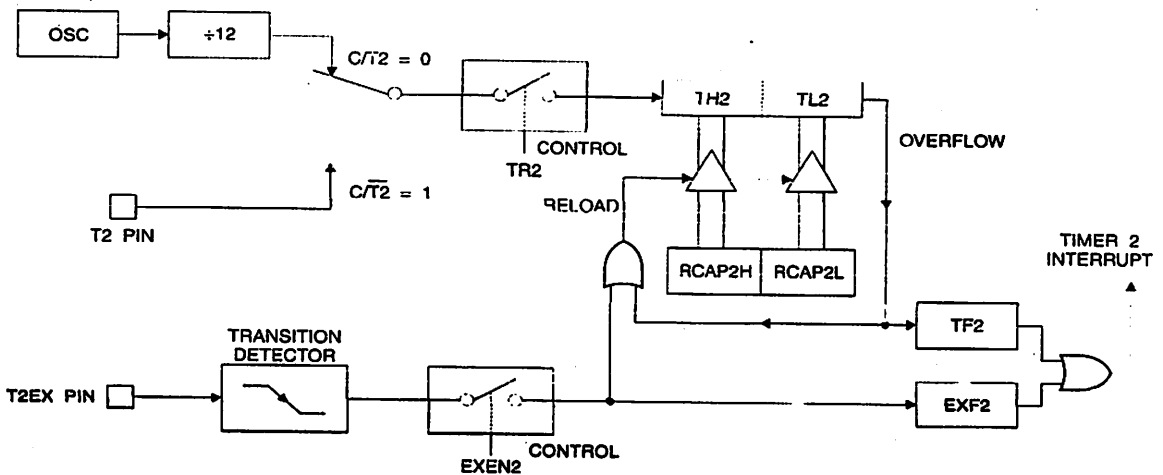


Table 9. T2MOD – Timer 2 Mode Control Register

T2MOD Address = 0C9H							Reset Value = XXXX XX00B	
Not Bit Addressable								
Bit	7	6	5	4	3	2	T2OE	DCEN
	-	-	-	-	-	-	1	0

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

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

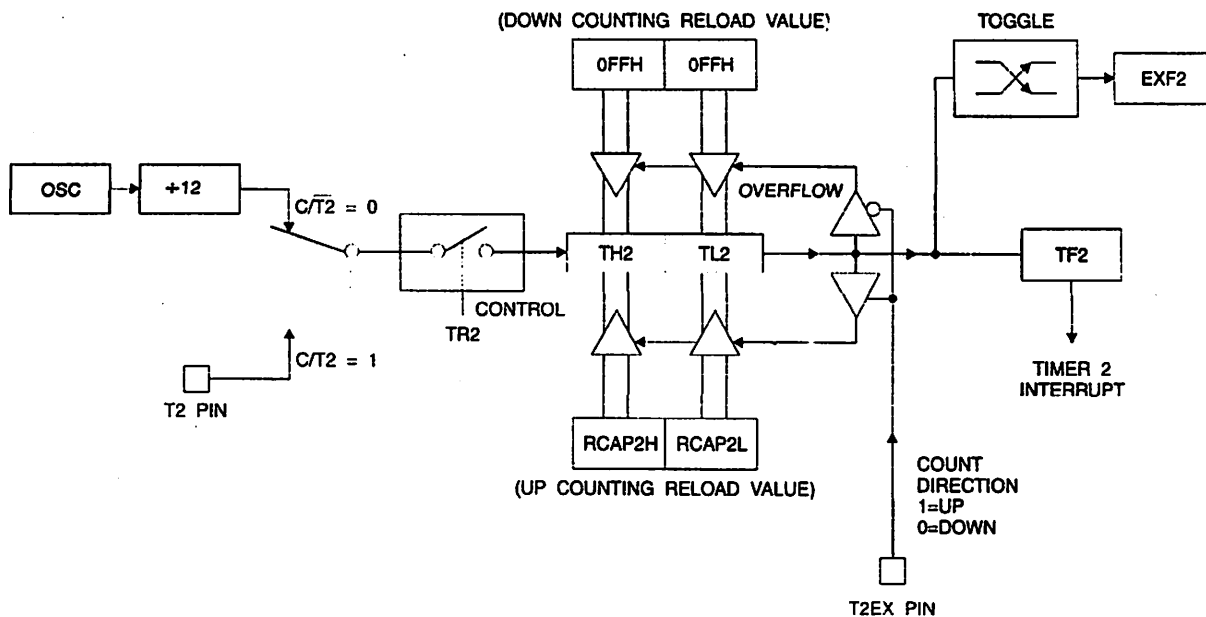
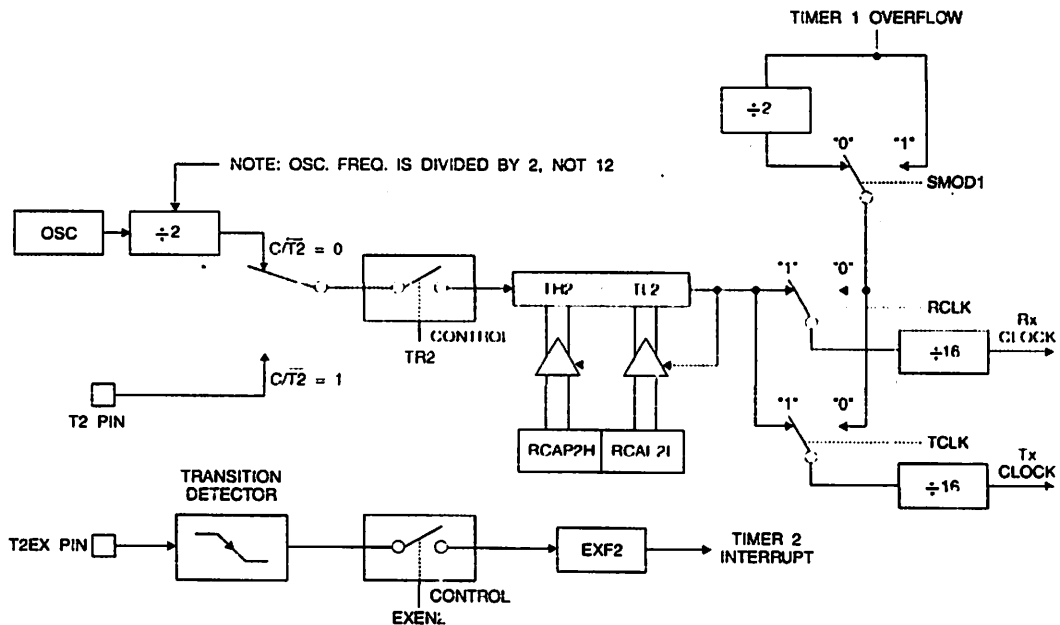


Figure 4. Timer 2 in Baud Rate Generator Mode



Baud Rate Generator

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

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

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

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

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

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

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



Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

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

Programmable Clock Out

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

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

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

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

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

Figure 5. Timer 2 in Clock-out Mode

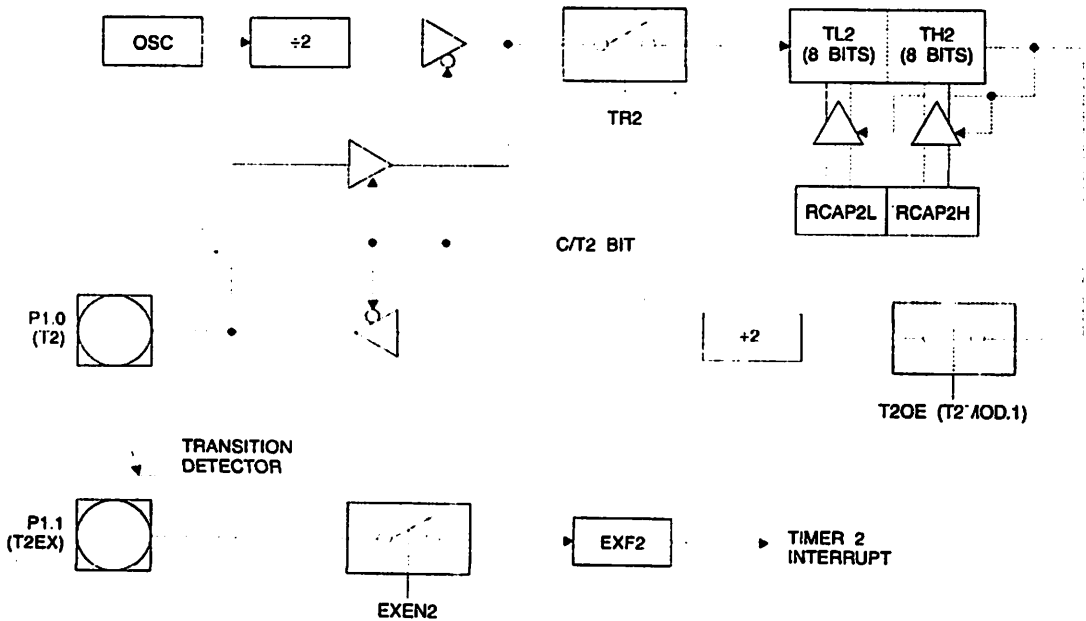
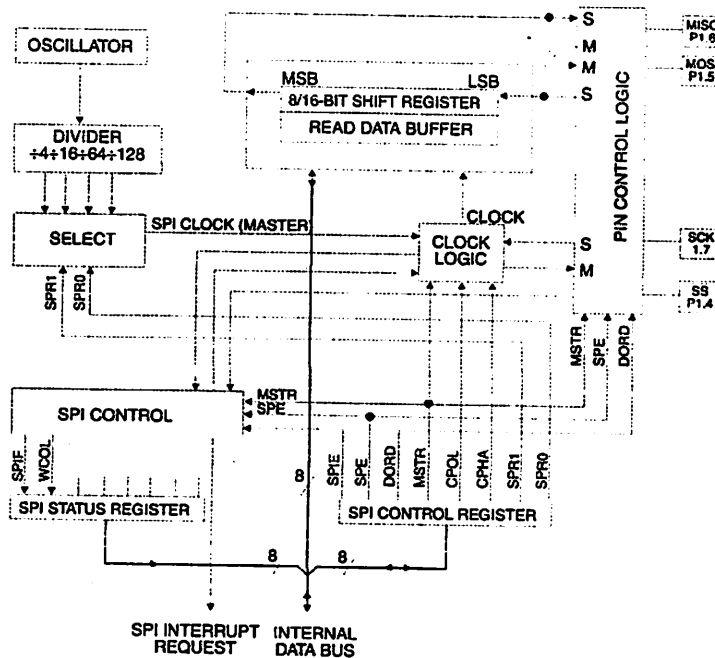


Figure 6. SPI Block Diagram





UART

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

Serial Peripheral Interface

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

- Full-Duplex, 3-Wire Synchronous Data Transfer
- Master or Slave Operation
- 1.5 MHz Bit Frequency (max.)
- LSB First or MSB First Data Transfer
- Four Programmable Bit Rates
- End of Transmission Interrupt Flag
- Write Collision Flag Protection
- Wakeup from Idle Mode (Slave Mode Only)

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

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

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

Figure 7. SPI Master-slave Interconnection

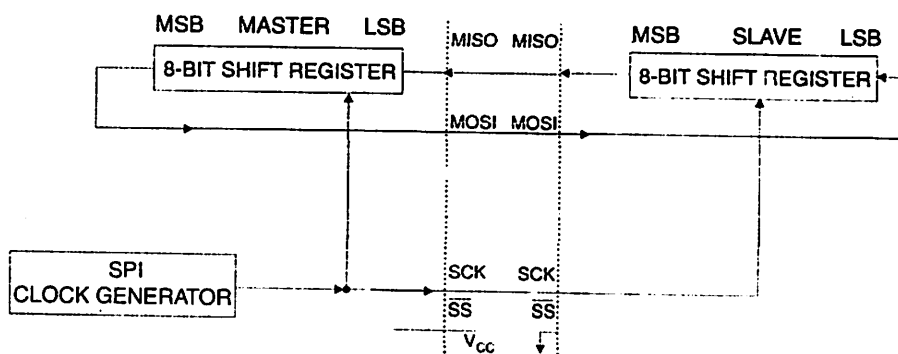
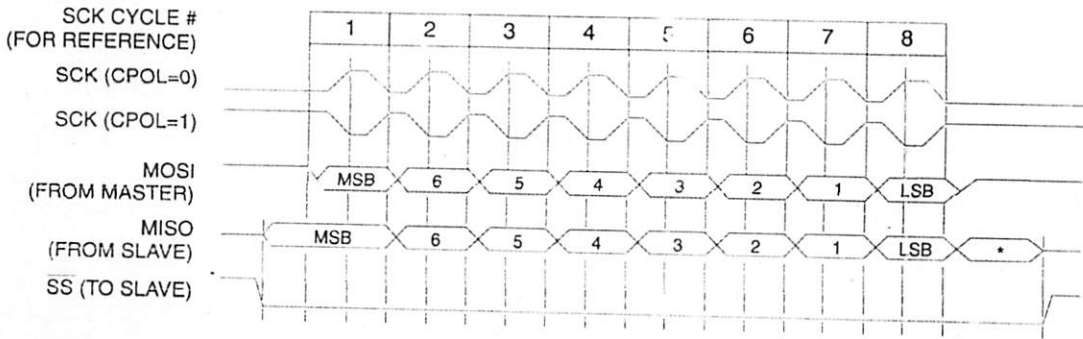
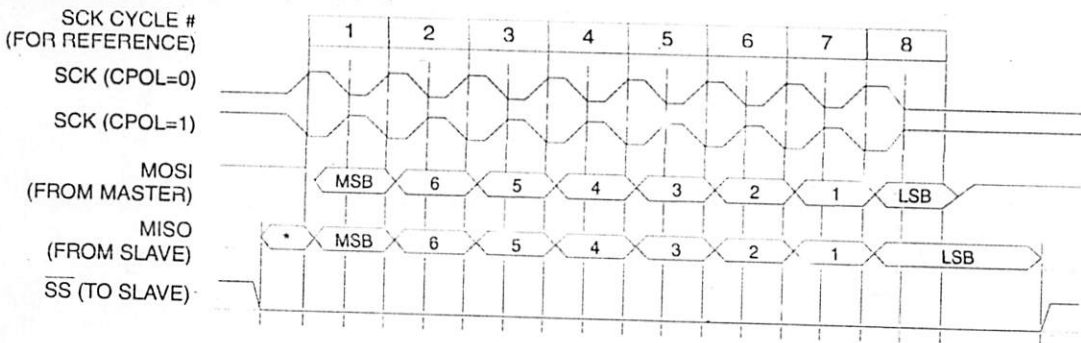


Figure 8. SPI transfer Format with CPHA = 0


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

Figure 9. SPI Transfer Format with CPHA = 1


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

Interrupts

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

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

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

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

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



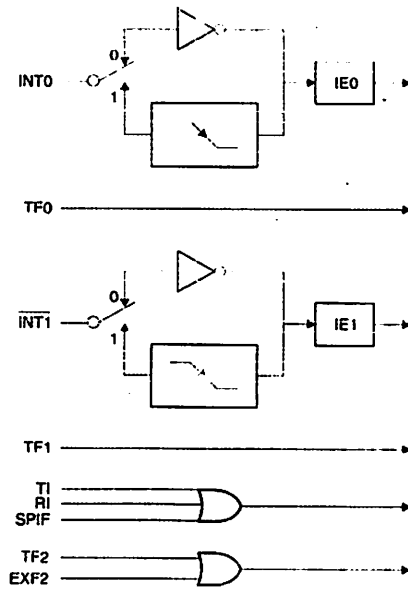
Table 10. Interrupt Enable (IE) Register

(MSB)(LSB)							
EA	-	ET2	ES	ET1	EX1	ET0	EX0
Enable Bit = 1 enables the interrupt.							
Enable Bit = 0 disables the interrupt.							

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
-	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	SPI and UART interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.

User software should never write 1s to unimplemented bits, because they may be used in future AT89 products.

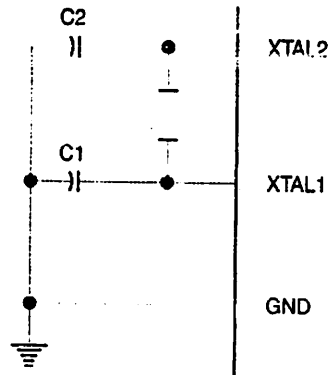
Figure 10. Interrupt Sources



Oscillator

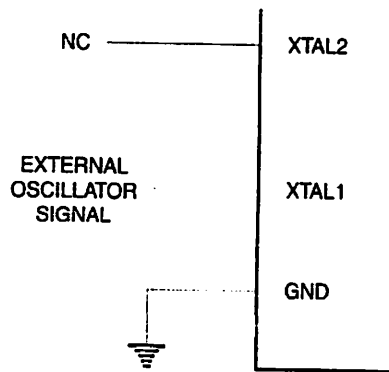
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected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

Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration



Programming the Flash and EEPROM

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

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

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

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

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

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

1. Power-up sequence:
 - Apply power between V_{CC} and GND pins.
 - Set RST pin to "H".
 - Apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Set \overline{PSEN} pin to "L"
 - ALE pin to "H"
 - \overline{EA} pin to "H" and all other pins to "H".
3. Apply the appropriate combination of "H" or "L" logic levels to pins P2.6, P2.7, P3.6, P3.7 to select one of the programming operations shown in the Flash Programming Modes table.
4. Apply the desired byte address to pins P1.0 to P1.7 and P2.0 to P2.5.
 - Apply data to pins P0.0 to P0.7 for Write Code operation.
5. Raise \overline{EA}/V_{PP} to 12V to enable Flash programming, erase or verification.
6. Pulse ALE/ \overline{PROG} once to program a byte in the Code memory array, the Data memory array or the lock bits. The byte-write cycle is self-timed and typically takes 1.5 ms.
7. To verify the byte just programmed, bring pin P2.7 to "L" and read the programmed data at pins P0.0 to P0.7.
8. Repeat steps 3 through 7 changing the address and data for the entire 2K or 8K bytes array or until the end of the object file is reached.
9. Power-off sequence:
 - Set XTAL1 to "L".
 - Set RST and \overline{EA} pins to "L".
 - Turn V_{CC} power off.





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

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

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

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

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

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

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

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

The AT89S8252 is shipped with the Serial Programming Mode enabled.

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

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

Programming Interface

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

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

Serial Downloading

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

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

The Code and Data memory arrays have separate address spaces:

0000H to 1FFFFH for Code memory and 0000H to 7FFFH for Data memory.

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

Serial Programming Algorithm

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

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



Serial Programming Instruction

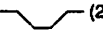

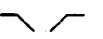
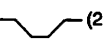
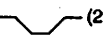
The Instruction Set for Serial Programming follows a 3-byte protocol and is shown in the following table:

Instruction Set

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

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

Flash and EEPROM Parallel Programming Modes

Mode	RST	PSEN	ALE/PROG	EA/V _{PP}	P2.6	P2.7	P3.6	P3.7	Data I/O P0.7:0	Address P2.5:0 P1.7:0
Serial Prog. Modes	H	h ⁽¹⁾	h ⁽¹⁾	x						
Chip Erase	H	L	 (2)	12V	H	L	L	L	X	X
Write (10K bytes) Memory	H	L		12V	L	H	H	H	DIN	ADDR
Read (10K bytes) Memory	H	L	H	12V	L	L	H	H	DOUT	ADDR
Write Lock Bits:	H	L		12V	H	L	H	L	DIN	X
Bit - 1									P0.7 = 0	X
Bit - 2									P0.6 = 0	X
Bit - 3									P0.5 = 0	X
Read Lock Bits:	H	L	H	12V	L	H	L	L	DOUT	X
Bit - 1									@P0.2	X
Bit - 2									@P0.1	X
Bit - 3									@P0.0	X
Read Atmel Code	H	L	H	12V	L	L	L	L	DOUT	30H
Read Device Code	H	L	H	12V	L	L	L	L	DOUT	31H
Serial Prog. Enable	H	L	 (2)	12V	L	H	L	H	P0.0 = 0	X
Serial Prog. Disable	H	L	 (2)	12V	L	H	L	H	P0.0 = 1	X
Read Serial Prog. Fuse	H	L	H	12V	H	H	L	L	@P0.0	X

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



Figure 13. Programming the Flash/EEPROM Memory

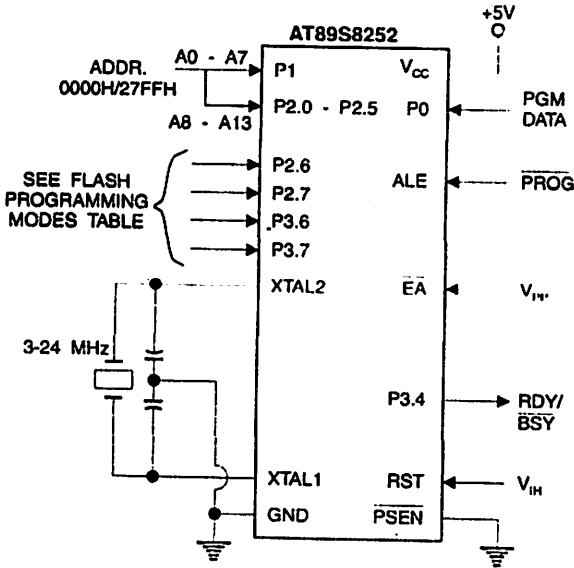


Figure 15. Flash/EEPROM Serial Downloading

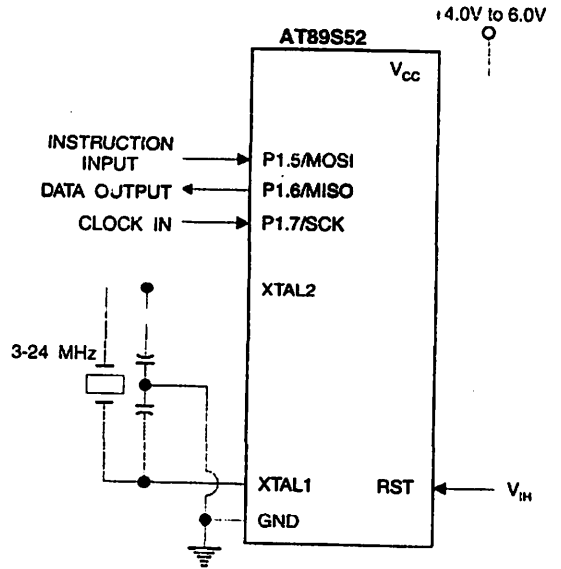
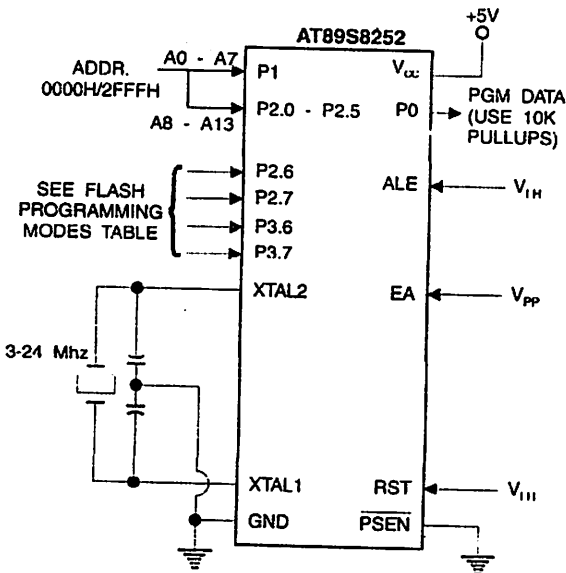


Figure 14. Verifying the Flash/EEPROM Memory

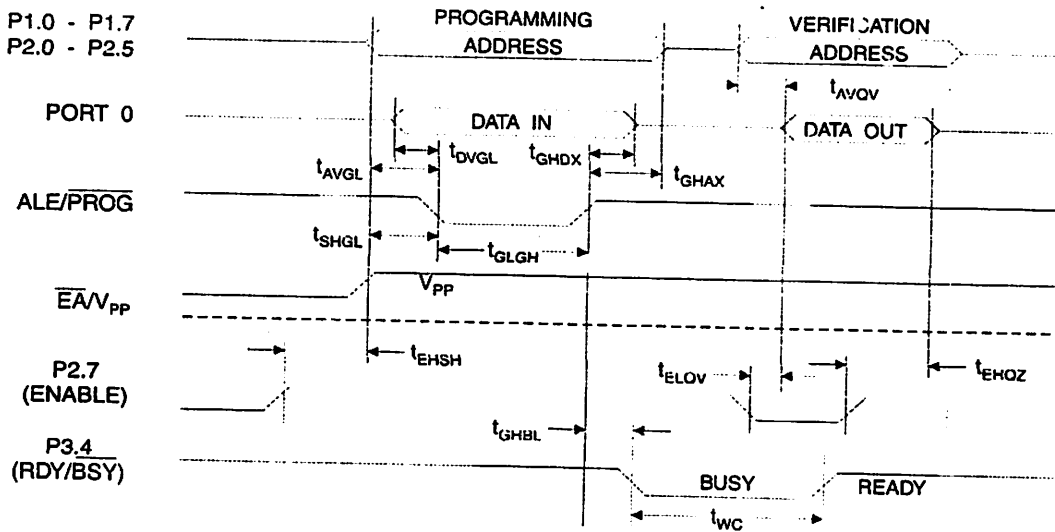


Flash Programming and Verification Characteristics – Parallel Mode

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

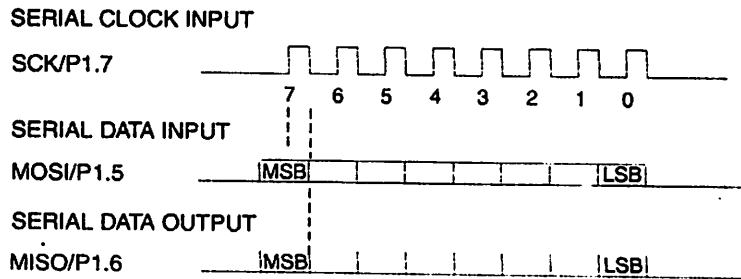
Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold after $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold after $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{EHS1}	P2.7 (ENABLE) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{\text{PROG}}$ Low	10		μs
t_{GLGH}	$\overline{\text{PROG}}$ Width	1	110	μs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELOV}	ENABLE Low to Data Valid		$48t_{CLCL}$	
t_{EHOZ}	Data Float after 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

Flash/EEPROM Programming and Verification Waveforms – Parallel Mode





Serial Downloading Waveforms



Serial Programming Characteristics

Figure 16. Serial Programming Timing

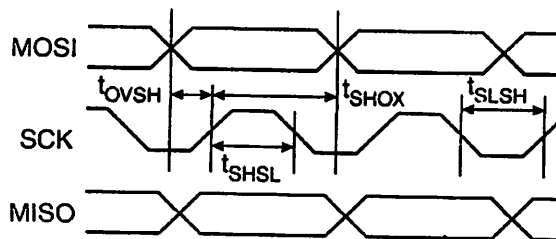


Table 11. Serial Programming Characteristics, $T_A = -40^\circ\text{C}$ to 85°C , $V_{CC} = 4.0 - 6.0\text{V}$ (Unless Otherwise Noted)

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

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current.....	15.0 mA

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

DC Characteristics

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

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except \overline{EA})	-0.5	$0.2 V_{CC} - 0.1$	V
V_{IL1}	Input Low-voltage (\overline{EA})		-0.5	$0.2 V_{CC} - 0.3$	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V
V_{OL}	Output Low-voltage ⁽¹⁾ (Ports 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.5	V
V_{OL1}	Output Low-voltage ⁽¹⁾ (Port 0, ALE, \overline{PSEN})	$I_{OL} = 3.2 \text{ mA}$		0.5	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, \overline{PSEN})	$I_{OH} = -60 \mu\text{A}, V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -25 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -10 \mu\text{A}$	$0.9 V_{CC}$		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	$I_{OH} = -800 \mu\text{A}, V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -300 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -80 \mu\text{A}$	$0.9 V_{CC}$		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	$V_{IN} = 0.45\text{V}$		-50	μA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	$V_{IN} = 2\text{V}, V_{CC} = 5\text{V} \pm 10\%$		-650	μA
I_L	Input Leakage Current (Port 0, \overline{EA})	$0.45 < V_{IN} < V_{CC}$		± 10	μA
RRST	Reset Pull-down Resistor		50	300	$\text{K}\Omega$
C_{iO}	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode ⁽²⁾	$V_{CC} = 6\text{V}$		100	μA
		$V_{CC} = 3\text{V}$		40	μA

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





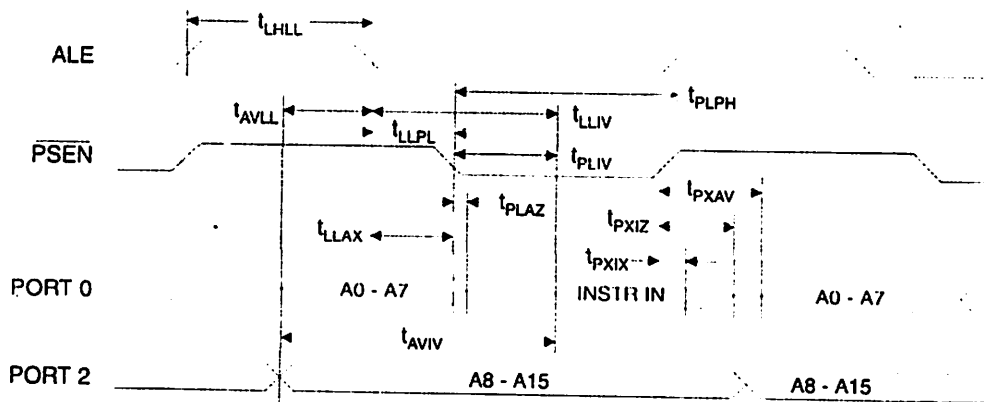
AC Characteristics

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

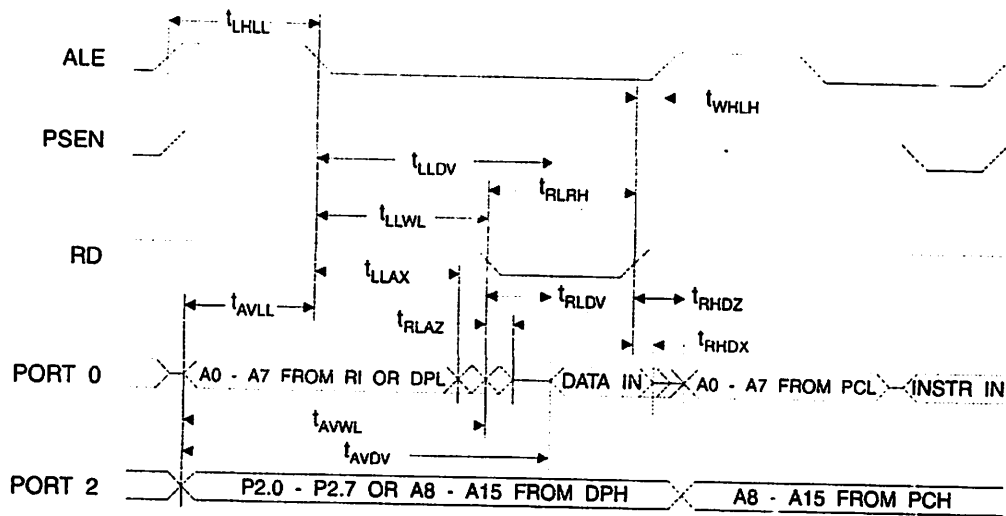
External Program and Data Memory Characteristics

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

External Program Memory Read Cycle

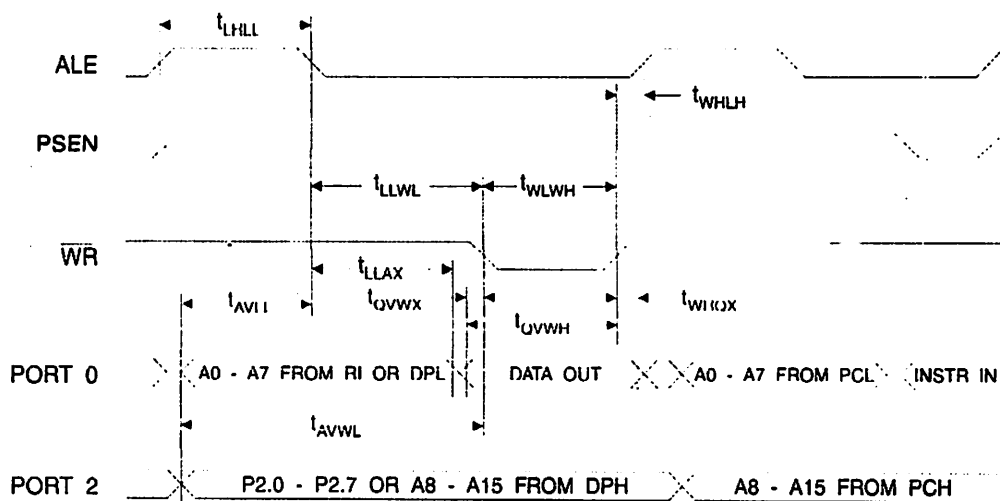


External Data Memory Read Cycle

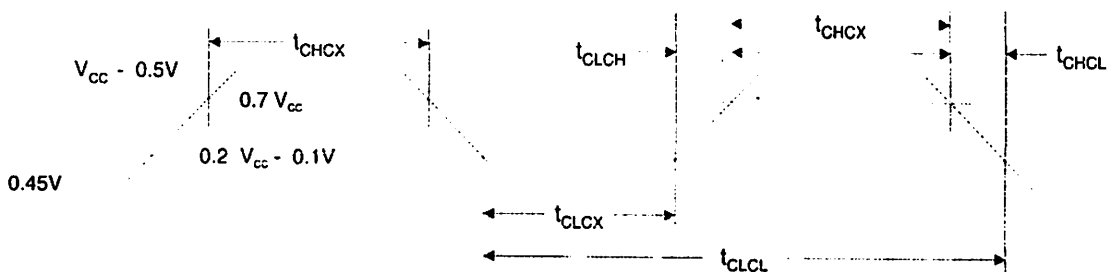




External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

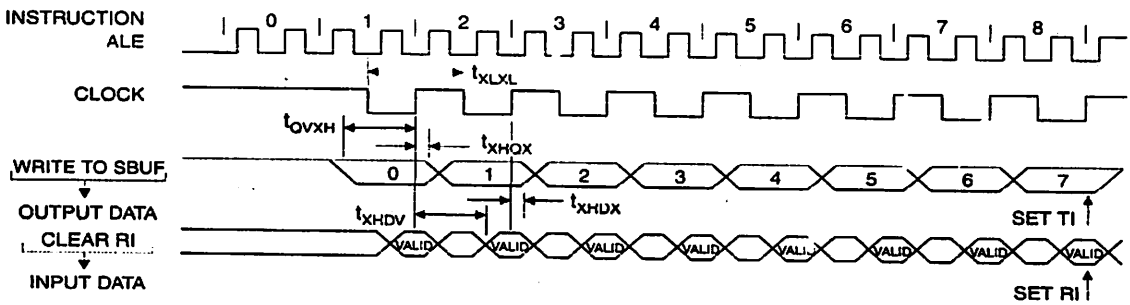
Symbol	Parameter	$V_{CC} = 4.0V \text{ to } 6.0V$		Units
		Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{CLCL}	Clock Period	41.6		ns
t_{CHCX}	High Time	15		ns
t_{CLCX}	Low Time	15		ns
t_{CLCH}	Rise Time		20	ns
t_{CHCL}	Fall Time		20	ns

Serial Port Timing: Shift Register Mode Test Conditions

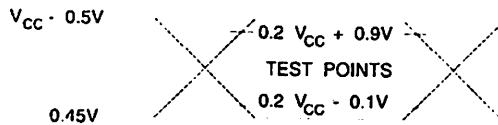
The values in this table are valid for $V_{CC} = 4.0V$ to $6V$ and Load Capacitance = 80 pF .

Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	$12t_{CLCL}$		μS
t_{QVXH}	Output Data Setup to Clock Rising Edge	$10t_{CLCL} - 133$		ns
t_{XHGX}	Output Data Hold after Clock Rising Edge	$2t_{CLCL} - 117$		ns
t_{YHDX}	Input Data Hold after Clock Rising Edge	0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		$10t_{CLCL} - 133$	ns

Shift Register Mode Timing Waveforms

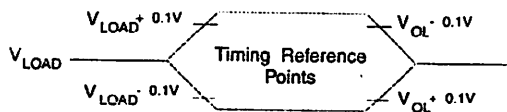


AC Testing Input/Output Waveforms⁽¹⁾

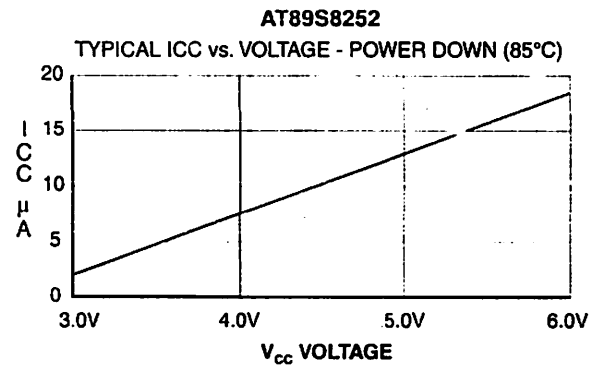
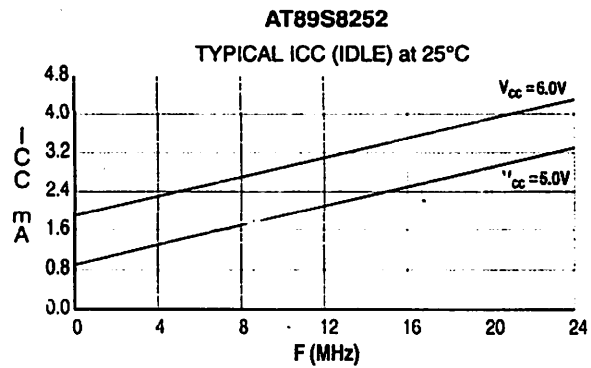
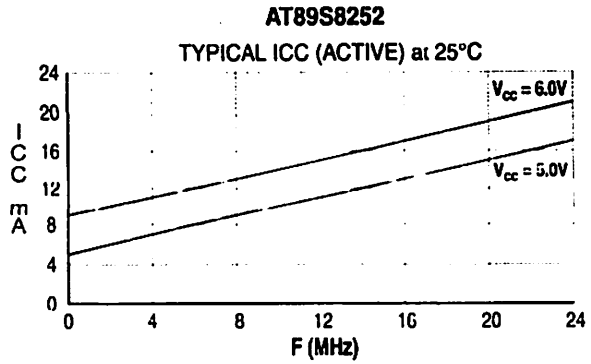


Note: 1. AC Inputs during testing are driven at $V_{CC} - 0.5V$ for a logic 1 and $0.45V$ for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

Float Waveforms⁽¹⁾



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



- Notes: 1. XTAL1 tied to GND for I_{cc} (power-down)
2. Lock bits programmed

Ordering Information

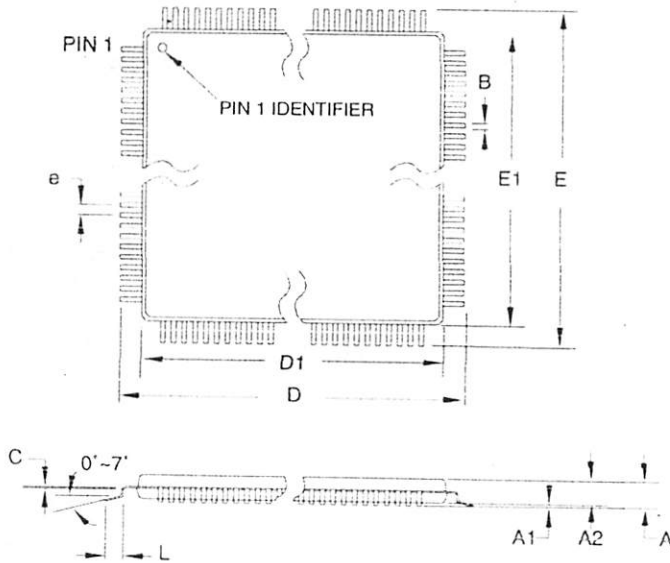
Speed (MHz)	Power Supply	Ordering Code	Package	Operatic n Range
24	4.0V to 6.0V	AT89S8252-24AC	44A	Commercial (0°C to 70°C)
		AT89S8252-24JC	44J	
		AT89S8252-24PC	40P6	
	4.0V to 6.0V	AT89S8252-24AI	44A	Industrial (-40°C to 85°C)
		AT89S8252-24JI	44J	
		AT89S8252-24PI	40P6	

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



Packaging Information

14A – TQFP



COMMON DIMENSIONS
(Unit of Measure = mm)

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

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

10/5/2001



2325 Orchard Parkway
San Jose, CA 95131

TITLE

44A, 44-lead, 10 x 10 mm Body Size, 1.0 mm Body Thickness,
0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)

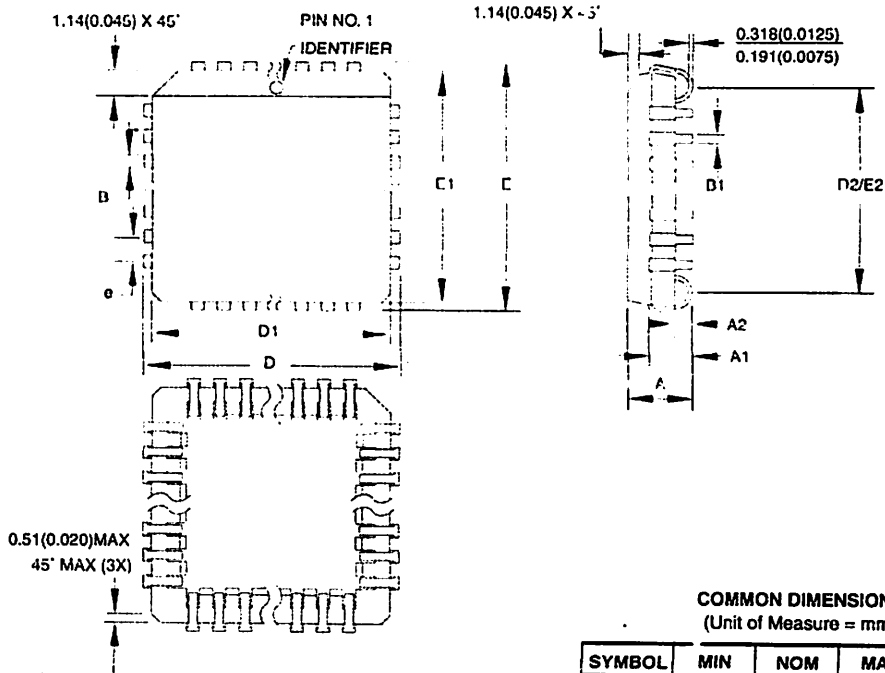
DRAWING NO.

44A

REV.

B

44J - PLCC



COMMON DIMENSIONS
(Unit of Measure = mm)

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

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

10/04/01



2325 Orchard Parkway
San Jose, CA 95131

TITLE

44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)

DRAWING NO.

44J

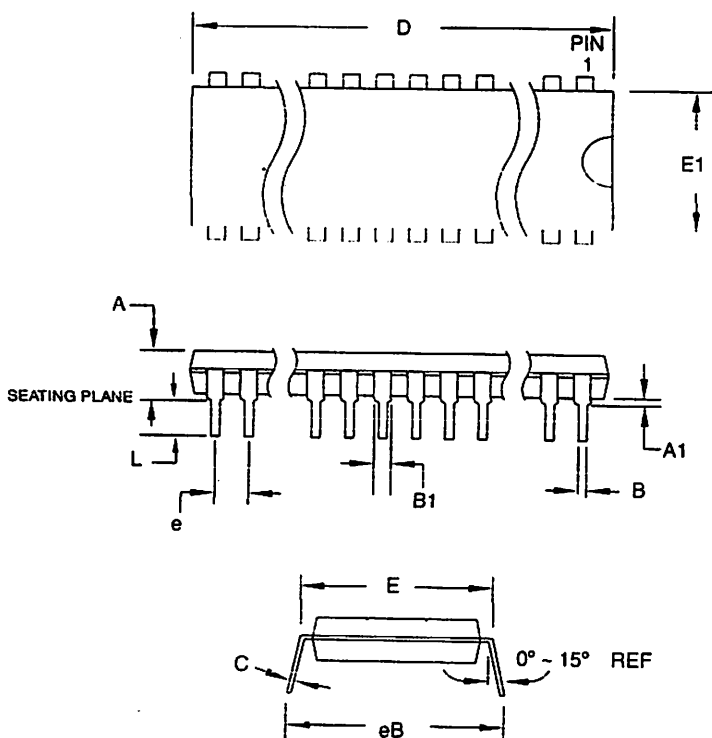
REV.

B





40P6 - PDIP



COMMON DIMENSIONS
(Unit of Measure = mm)

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

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

09/28/01



2325 Orchard Parkway
San Jose, CA 95131

TITLE

40P6, 40-lead (0.600"/15.24 mm Wide) Plastic Dual In-line Package (PDIP)

DRAWING NO.

40P6

REV.

B



Atmel Corporation

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

Regional Headquarters

Europe

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

Asia

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

Japan

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

Atmel Operations

Memory

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

Microcontrollers

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

La Chanterrie

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

ASIC/ASSP/Smart Cards

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

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

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

RF/Automotive

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

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

Bio-metrics/Imaging/Hi-Rel MPU/ High-Speed Converters/RF Datacom

Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests

www.atmel.com/literature

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. EXCEPT AS SET FORTH IN ATMEL'S TERMS AND CONDITIONS OF SALE LOCATED ON ATMEL'S WEB SITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

© Atmel Corporation 2006. All rights reserved. Atmel®, logo and combinations thereof, Everywhere You Are® and others, are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.



Printed on recycled paper.

0401G-MICRO-3/06

xM

- 16 Characters x 2 line
- 5 x 7 Dot Matrix + Cursor
- 1/16 Duty
- 5V single Power Supply
- Available in EL and LED Backlight type

■ Absolute Maximum Ratings

V_{SS}=0V, Ta=25°C

Item	Symbol	Conditions	Min.	Max.	Unit
Power supply voltage	V _{DD}		-0.3	6.0	V
	V _{LC}		-0.3	V _{DD}	V
Input voltage	V _{in}		-0.3	V _{DD} +0.3	V
Operating temperature					
Normal	T _{opr}		0	+50	°C
Wide	T _{opr}		-20	+70	°C
Storage temperature					
Normal	T _{stg}		-20	+60	°C
Wide	T _{stg}		-30	+80	°C
Storage humidity		< 48 hrs	+20	+85	%RH
		<1000 hrs	+20	+65	%RH

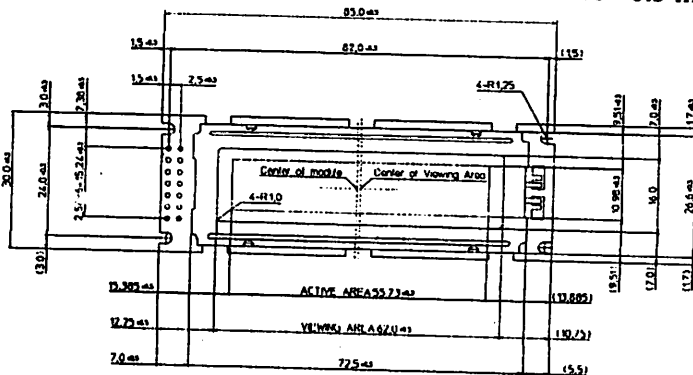
■ Mechanical Characteristics

Item	Specifications	Unit
Module size (H x V)	85.0 x 30.0	mm
Thickness	Reflective/EL	10.1
	LED	15.8
Viewing area (H x V)	62.0 x 16.0	mm
Character size with cursor (H x V)	2.78 x 4.89	mm
Mounting hole distance (H x V)	82.0 x 24.0	mm
Weight	Reflective	25
	EL backlight	30
	LED backlight	40

H : Horizontal, V : Vertical

■ Dimensions

M1632 (2x16) Unit: mm, General tolerance ±0.5 mm



■ Electrical Characteristics

V_{DD}=5V±5%, V_{SS}=0V, Ta=0 ~ 50°C

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Power supply voltage	V _{DD}		4.75	5.00	5.25	V
	V _{DD} - V _{LC}		3.0	-	6.3	V
Input voltage	V _{IH1}		2.2	-	V _{DD}	V
	V _{IL1}		0	-	0.6	V
Output voltage	V _{OH1}	-I _{OH} =0.205mA	2.4	-	-	V
	V _{OL1}	-I _{OL} =1.2mA	-	-	0.4	V
Current consumption *						
Normal Temp. type	b _{DD}	Ta=25°C	-	1.6	2.5	mA
	LC	V _{LC} =0.25V	-	0.2	1.0	mA
Wide Temp. type	b _{DD}	Ta=25°C	-	1.6	2.5	mA
	LC	V _{LC} =-0.6V	-	0.3	1.0	mA

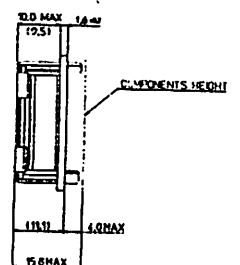
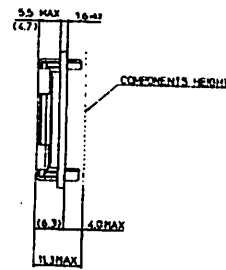
* test pattern : check board pattern

■ Pin Function

No.	Name	Function
1	VSS	GND
2	VDD	Power supply voltage +5V
3	VLC	Liquid crystal driving voltage
4	RS	L : Instruction code input
		H : Data input
5	R/W	L : Data write from MPU to LCM
		H : Data read from LCM to MPU
6	E	Enable
7	DB0	Data bus line
8	DB1	Data bus line
9	DB2	Data bus line
10	DB3	Data bus line
11	DB4	Data bus line
12	DB5	Data bus line
13	DB6	Data bus line
14	DB7	Data bus line
(15)	A	Anode (+) for LED backlight
(16)	K	Cathode (-) for LED backlight

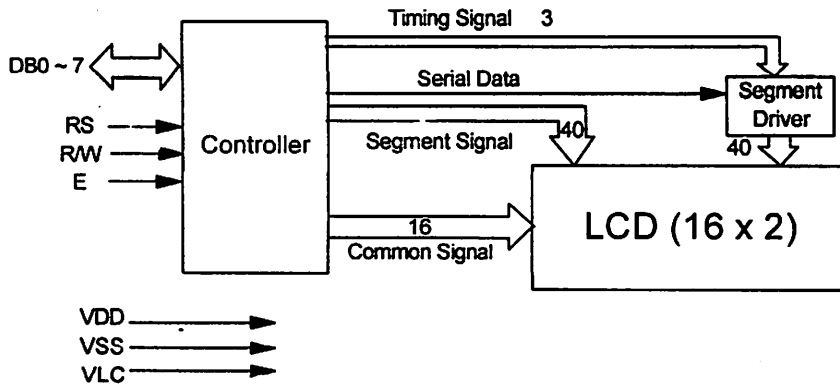
Reflective/EL Backlight

LED Backlight



Note : Only dimension changes between Reflective/EL and LED backlight is the thickness.

■ Circuit Block diagram



■ Recommended Operating Voltage

The recommended value (Vopr) for an ambient temperature is as follows.

$Vopr = VDD - VLc$

Temperature (°C)	-20	0	+25	+50	+70
Vopr (V) Normal	-	5.00	4.75	4.50	-
Vopr (V) Wide	6.20	5.90	5.60	5.40	5.20

■ Optical Characteristics

1. Normal Temperature Range Type

$Ta = 21^{\circ}C$, 1/16 Duty, $Vopr = 4.75V$

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Viewing angle	$\theta 1$	$C \geq 2$	-	-	-15	deg.
	$\theta 2$	$\phi = 0^{\circ}$	55	-	-	
	$\theta 2 - \theta 1$		70	-	-	
Contrast	C	$\theta = +2.5^{\circ}, \phi = 0^{\circ}$	-	5	-	-
Response time	trn (rise)	$\theta = 0^{\circ}$	-	150	200	msec
	toff (fall)	$\phi = 0^{\circ}$	-	200	220	msec
	ton (rise)	$\theta = 0^{\circ}, \phi = 0^{\circ}$	-	750	800	msec
	toff (fall)	$Ta = 0^{\circ}C, Vopr = 5.0V$	-	600	700	msec

Measuring equipment : Canon illuminater LC-4SR

2. Wide Temperature Range Type

$Ta = 21^{\circ}C$, 1/16 Duty, $Vopr = VDD - VLc$

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Viewing angle	$\theta 1$	$C \geq 2$	-	-	-15	deg.
	$\theta 2$	$\phi = 0^{\circ}$	55	-	-	
	$\theta 2 - \theta 1$	$Vopr = 5.6v$	70	-	-	
Contrast	C	$\theta = +20^{\circ}, \phi = 0^{\circ}$ $Vopr = 5.6v$	-	5	-	-
Response time	ton (rise)	$\theta = 0^{\circ}, \phi = 0^{\circ}$	-	150	200	msec
	toff (fall)	$Ta = 21^{\circ}C, Vopr = 5.6V$	-	200	220	msec
	ton (rise)	$\theta = 0^{\circ}, \phi = 0^{\circ}$	-	750	800	msec
	toff (fall)	$Ta = 0^{\circ}C, Vopr = 5.9V$	-	600	700	msec

Measuring equipment : Canon illuminater LC-4SR

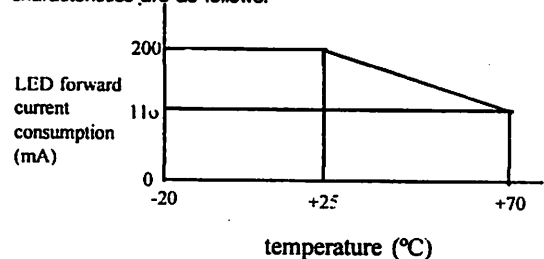
■ LED Backlight

1. Absolute Maximum Ratings

$Ta = 25^{\circ}C$

Item	Symbol	Specifications	Unit
LED forward current consumption *	I _F	200	mA
LED reverse voltage	V _R	8	V
Allowable loss	P _D	0.92	W
Operating Temperature	Topr	-20 ~ +70	°C
Storage Temperature	Tstg	-40 ~ +80	°C

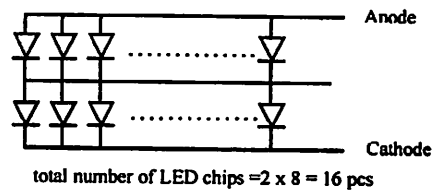
* LED forward current consumption and operating temperature characteristics are as follows.



2. Electrical Characteristics

$Ta = 25^{\circ}C$

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
LED forward input voltage	V _F	I _F = 112mA	3.8	4.1	4.4	V
LED reverse current	I _R	V _R = 8V	-	-	0.16	mA
Brightness	L	I _F = 112mA*	40	50	-	cd/m ²



■ EL Backlight

1. Absolute Maximum Ratings

Item	Symbol	Standard	Unit
Operating voltage	Vopr	AC 150V, 1KHz Sinew ave	V
Operating temperature	Topr	-10 ~ +50	°C
Storage temperature	Tstg	-20 ~ + 60	°C
Storage humidity		0 ~ 10 % RH (60 °C) 0 ~ 30 % RH (40 °C)	

2. Brightness, Current, Life Characteristics

Item	Conditions	Specifications	Unit
Brightness	100V, 400Hz	30 min.	cd/m ²
	Sinew ave	35 typ.	
Current	100V, 400Hz	1.2 typ.	mA
	Sinew ave	1.7 max	
Life *	100V, 400Hz, Sinew ave 25°C, 50%RH	1,500	hrs
	Using 5S Inverter 25°C, 50%RH	4,000	

* Definition of Life : Used continuously down to 10 cd/m²

3. Suitable Inverter 5S

3.1 Electrical Characteristics

(When combined with EL lamp)

Ta=25°C

Item	Symbol	Conditions	Specifications	Unit
Oscillating frequency	f _{INV}	V _{IN} =5VDC	550 typ.	Hz
Output voltage	V _{OUT}	V _{IN} =5VDC	100 typ.	V
Output current	I _{OUT}	V _{IN} =5VDC	1.5 typ.	mA
Input current voltage	V _{IN}		5 typ	VDC
	I _{IN}	V _{IN} =5VDC	10 typ	mA
Initial brightness	B	V _{IN} =5VDC	35 typ.	cd/m ²
Surface brightness (panel upper side)	B _P	V _{IN} =5VDC Vopr=0V	7 typ.	cd/m ²

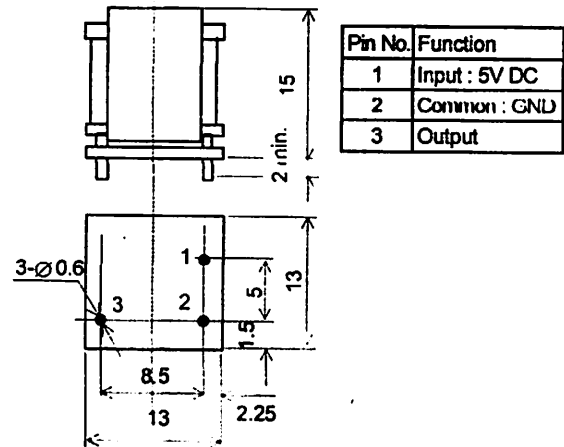
3.2 Tolerance (Inverter only)

Item	Specifications	Unit
Input voltage	3.0 to 6.0	V
Load range	5 to 15	cm ²

3.3 Maximum Ratings (Inverter only)

Item	Specifications	Unit
Input voltage	7.0	V
Load range	50	cm ²
Operating temperature	-10 to +60	°C
Storage temperature	-20 to +70	°C

3.4 Inverter Dimensions (unit : mm)

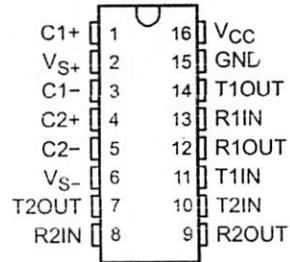


MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L – FEBRUARY 1989 – REVISED MARCH 2004

- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Operates From a Single 5-V Power Supply With 1.0- μ F Charge-Pump Capacitors
- Operates Up To 120 kbit/s
- Two Drivers and Two Receivers
- ± 30 -V Input Levels
- Low Supply Current . . . 8 mA Typical
- ESD Protection Exceeds JESD 22 – 2000-V Human-Body Model (A114-A)
- Upgrade With Improved ESD (15-kV HBM) and 0.1- μ F Charge-Pump Capacitors is Available With the MAX202
- Applications
 - TIA/EIA-232-F, Battery-Powered Systems, Terminals, Modems, and Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept ± 30 -V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T _A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube of 25	MAX232N	MAX232N
	SOIC (D)	Tube of 40	MAX232D	MAX232
		Reel of 2500	MAX232DR	
	SOIC (DW)	Tube of 40	MAX232DW	MAX232
		Reel of 2000	MAX232DWR	
SOP (NS)	Reel of 2000	MAX232NSR	MAX232	
-40°C to 85°C	PDIP (N)	Tube of 25	MAX232IN	MAX232IN
	SOIC (D)	Tube of 40	MAX232ID	MAX232I
		Reel of 2500	MAX232IDR	
	SOIC (DW)	Tube of 40	MAX232IDW	MAX232I
		Reel of 2000	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

LinASIC is a trademark of Texas Instruments.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

 **TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 2004, Texas Instruments Incorporated

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L - FEBRUARY 1989 - REVISED MARCH 2004

Function Tables

EACH DRIVER

INPUT TIN	OUTPUT TOUT
L	H
H	L

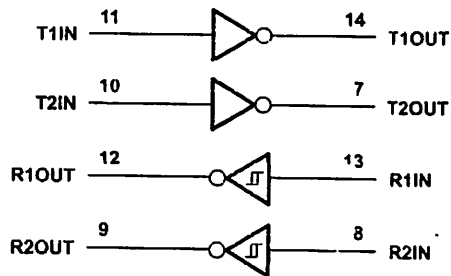
H = high level, L = low level

EACH RECEIVER

INPUT RIN	OUTPUT ROUT
L	H
H	L

H = high level, L = low level

Logic diagram (positive logic)



MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLI.S047L - FEBRUARY 1989 - REVISED MARCH 2004

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1)	-0.3 V to 6 V
Positive output supply voltage range, V_{S+}	$V_{CC} - 0.3$ V to 15 V
Negative output supply voltage range, V_{S-}	-0.3 V to -15 V
Input voltage range, V_I : Driver	-0.3 V to $V_{CC} + 0.3$ V
Receiver	± 30 V
Output voltage range, V_O : T1OUT, T2OUT	$V_{S-} - 0.3$ V to $V_{S+} + 0.3$ V
R1OUT, R2OUT	-0.3 V to $V_{CC} + 0.3$ V
Short-circuit duration: T1OUT, T2OUT	Unlimited
Package thermal impedance, θ_{JA} (see Notes 2 and 3): D package	73°C/W
DW package	57°C/W
N package	67°C/W
NS package	64°C/W
Operating virtual junction temperature, T_J	150°C
Storage temperature range, T_{stg}	-65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

- NOTES: 1. All voltages are with respect to network GND.
 2. Maximum power dissipation is a function of $T_J(\text{max})$, θ_{JA} , and T_A . The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_J(\text{max}) - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can affect reliability.
 3. The package thermal impedance is calculated in accordance with JESD 51-7.

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage (T1IN, T2IN)	2			V
V_{IL}	Low-level input voltage (T1IN, T2IN)			0.8	V
R1IN, R2IN	Receiver input voltage			± 30	V
T_A	Operating free-air temperature	MAX232	0	70	°C
		MAX232I	-40	85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Note 4 and Figure 4)

PARAMETER	TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
I_{CC}	Supply current				
	$V_{CC} = 5.5$ V, All outputs open, $T_A = 25^\circ\text{C}$		8	10	mA

‡ All typical values are at $V_{CC} = 5$ V and $T_A = 25^\circ\text{C}$.

NOTE 4: Test conditions are C1-C4 = 1 μF at $V_{CC} = 5 \text{ V} \pm 0.5 \text{ V}$.



POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L - FEBRUARY 1989 - REVISED MARCH 2004

DRIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 4)

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V _{OH}	High-level output voltage	T1OUT, T2OUT	R _L = 3 kΩ to GND	5	7		V
V _{OL}	Low-level output voltage‡	T1OUT, T2OUT	R _L = 3 kΩ to GND		-7	-5	V
r _o	Output resistance	T1OUT, T2OUT	V _{S+} = V _{S-} = 0, V _O = ±2 V	300			Ω
I _{OS} §	Short-circuit output current	T1OUT, T2OUT	V _{CC} = 5.5 V, V _O = 0		±10		mA
I _{IS}	Short-circuit input current	T1IN, T2IN	V _I = 0			200	μA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.

‡ The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

§ Not more than one output should be shorted at a time.

NOTE 4: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 4)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
SR	Driver slew rate	R _L = 3 kΩ to 7 kΩ, See Figure 2			30	V/μs
SR(t)	Driver transition region slew rate	See Figure 3		3		V/μs
	Data rate	One TOUT switching		120		kbit/s

NOTE 4: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

RECEIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 4)

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V _{OH}	High-level output voltage	R1OUT, R2OUT	I _{OH} = -1 mA	3.5			V
V _{OL}	Low-level output voltage‡	R1OUT, R2OUT	I _{OL} = 3.2 mA			0.4	V
V _{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN	V _{CC} = 5 V, T _A = 25°C		1.7	2.4	V
V _{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN	V _{CC} = 5 V, T _A = 25°C	0.8	1.2		V
V _{hys}	Input hysteresis voltage	R1IN, R2IN	V _{CC} = 5 V	0.2	0.5	1	V
η	Receiver input resistance	R1IN, R2IN	V _{CC} = 5, T _A = 25°C	3	5	7	kΩ

† All typical values are at V_{CC} = 5 V, T_A = 25°C.

‡ The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

NOTE 4: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 4 and Figure 1)

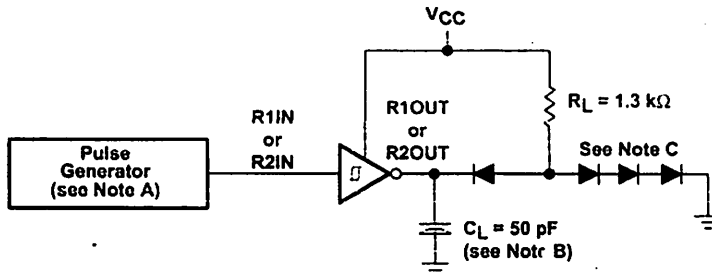
PARAMETER		TYP	UNIT
t _{PLH} (R)	Receiver propagation delay time, low- to high-level output	500	ns
t _{PHL} (R)	Receiver propagation delay time, high- to low-level output	500	ns

NOTE 4: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

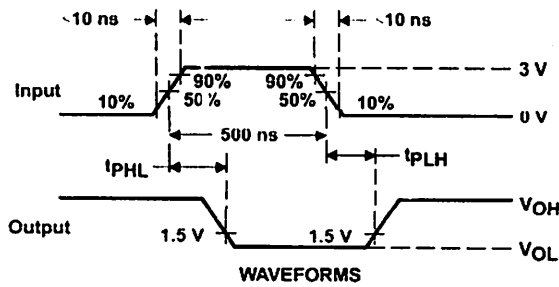


POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

PARAMETER MEASUREMENT INFORMATION



TEST CIRCUIT



WAVEFORMS

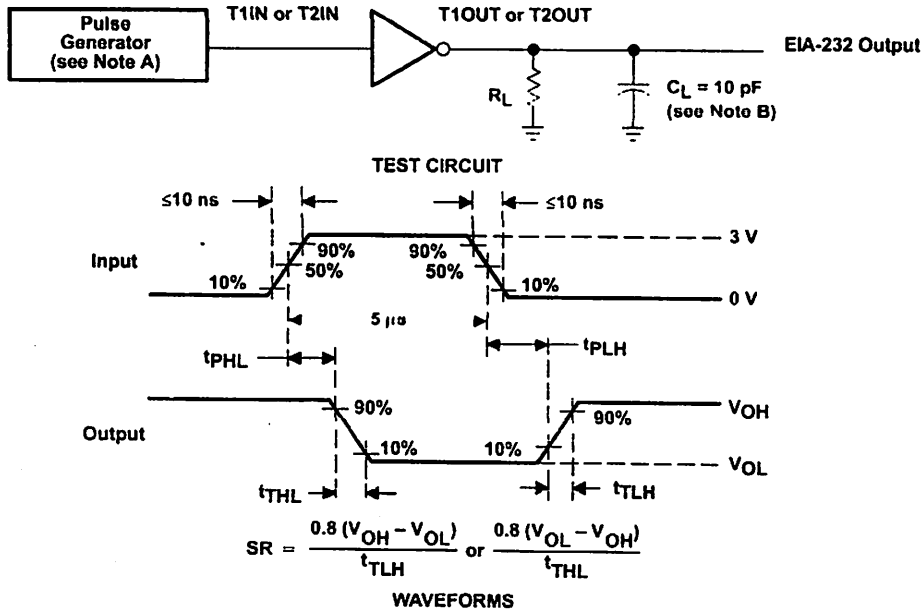
- NOTES: A. The pulse generator has the following characteristics: $Z_O = 50 \Omega$, duty cycle $\leq 50\%$.
 B. C_L includes probe and jig capacitance.
 C. All diodes are 1N3064 or equivalent.

Figure 1. Receiver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements

MAX232, MAX2321
DUAL EIA-232 DRIVERS/RECEIVERS

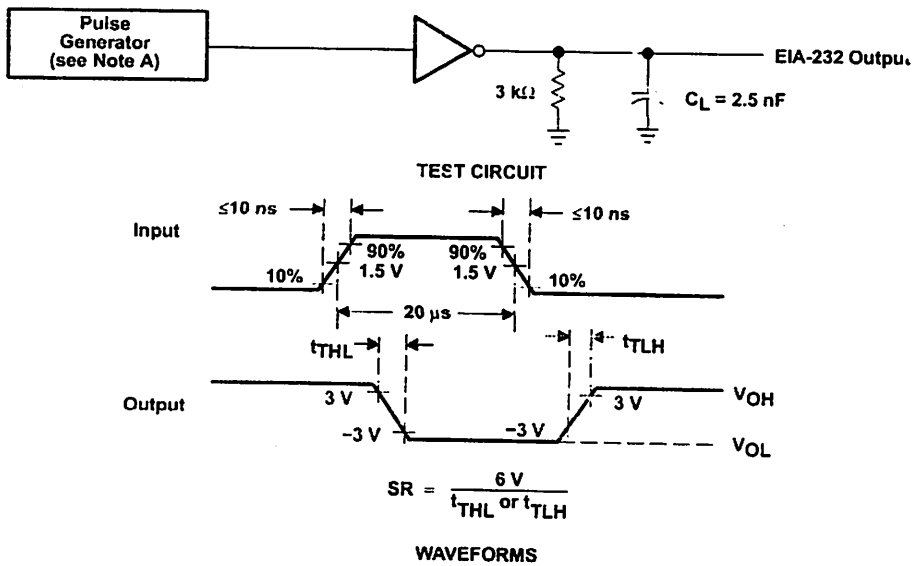
0LLS047L - FEBRUARY 1989 - REVISED MARCH 2004

PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: $Z_0 = 50 \Omega$, duty cycle $\leq 50\%$.
 B. C_L includes probe and jig capacitance.

Figure 2. Driver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements (5- μs Input)



NOTE A: The pulse generator has the following characteristics: $Z_0 = 50 \Omega$, duty cycle $\leq 50\%$.

Figure 3. Test Circuit and Waveforms for t_{THL} and t_{TLH} Measurements (20- μs Input)

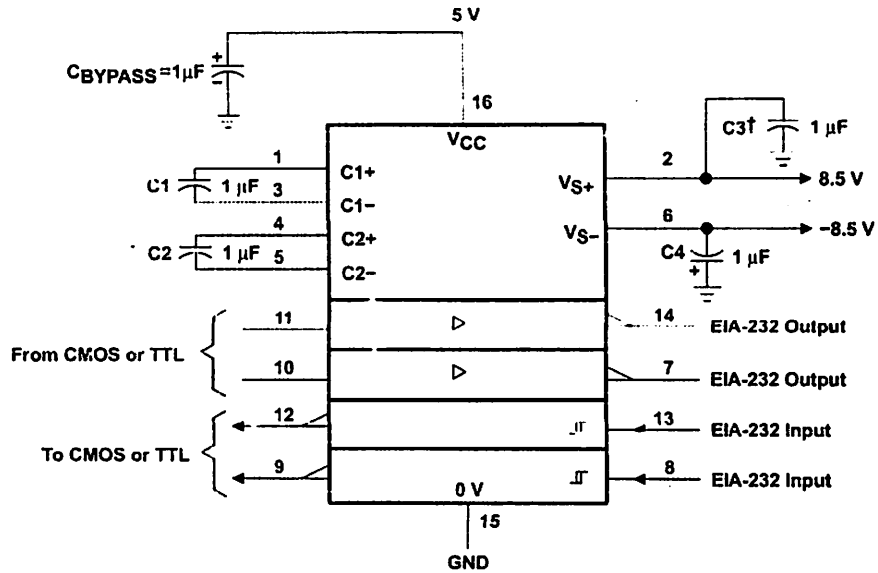


POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L - FEBRUARY 1989 - REVISED MARCH 2004

APPLICATION INFORMATION



† C3 can be connected to V_{CC} or GND.

NOTES: A. Resistor values shown are nominal.

B. Nonpolarized ceramic capacitors are acceptable. If polarized tantalum or electrolytic capacitors are used, they should be connected as shown. In addition to the 1- μ F capacitors shown, the MAX202 can operate with 0.1- μ F capacitors.

Figure 4. Typical Operating Circuit

KSC945

KSC945

Audio Frequency Amplifier & High Frequency OSC.

- Complement to KSA733
- Collector-Base Voltage : $V_{CBO}=60V$
- High Current Gain Bandwidth Product : $f_T=300MHz$ (TYP)
- Suffix "-C" means Center Collector (1. Emitter 2. Collector 3. Base)



TO-92
1. Emitter 2. Base 3. Collector

NPN Epitaxial Silicon Transistor

Absolute Maximum Ratings $T_a=25^\circ C$ unless otherwise noted

Symbol	Parameter	Value	Units
V_{CBO}	Collector-Base Voltage	60	V
V_{CEO}	Collector-Emitter Voltage	50	V
V_{EBO}	Emitter-Base Voltage	5	V
I_C	Collector Current	150	mA
P_C	Collector Power Dissipation	250	mW
T_J	Junction Temperature	150	$^\circ C$
T_{STG}	Storage Temperature	-55 ~ 150	$^\circ C$

Electrical Characteristics $T_a=25^\circ C$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units
BV_{CBO}	Collector-Base Breakdown Voltage	$I_C=100\mu A, I_E=0$	60			V
BV_{CEO}	Collector-Emitter Breakdown Voltage	$I_C=10mA, I_B=0$	50			V
BV_{EBO}	Emitter-Base Breakdown Voltage	$I_E=10\mu A, I_C=0$	5			V
I_{CBO}	Collector Cut-off Current	$V_{CB}=40V, I_E=0$			0.1	μA
I_{EBO}	Emitter Cut-off Current	$V_{EB}=3V, I_C=0$			0.1	μA
h_{FE}	DC Current Gain	$V_{CE}=6V, I_C=1.0mA$	40		700	
$V_{CE(sat)}$	Collector-Emitter Saturation Voltage	$I_C=100mA, I_B=10mA$		0.15	0.3	V
f_T	Current Gain Bandwidth Product	$V_{CE}=6V, I_C=10mA$		300		MHz
C_{ob}	Output Capacitance	$V_{CB}=6V, I_E=0, f=1MHz$		2.5		pF
NF	Noise Figure	$V_{CE}=6V, I_C=0.5mA, f=1KHz, R_S=500\Omega$		4.0		dB

h_{FE} Classification

Classification	R	O	Y	G	L
h_{FE}	40 ~ 80	70 ~ 140	120 ~ 240	200 ~ 400	350 ~ 700

Typical Characteristics

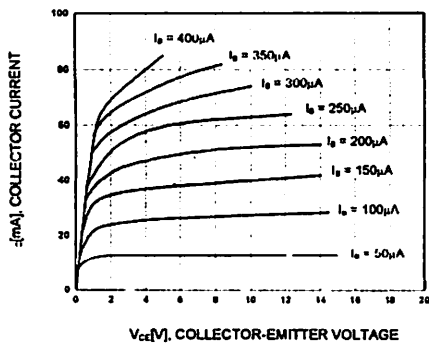


Figure 1. Static Characteristic

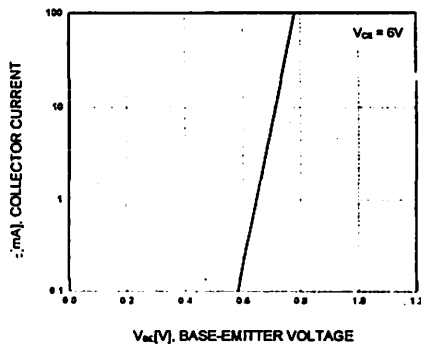


Figure 2. Transfer Characteristic

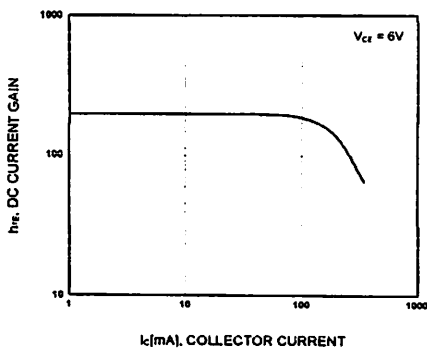


Figure 3. DC current Gain

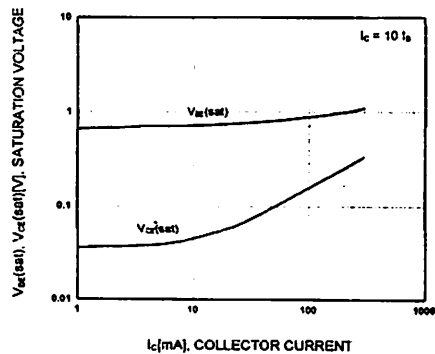


Figure 4. Base-Emitter Saturation Voltage
Collector-Emitter Saturation Voltage

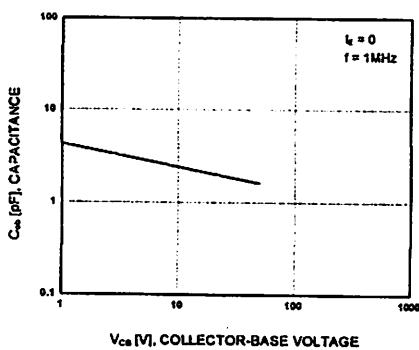


Figure 5. Output Capacitance

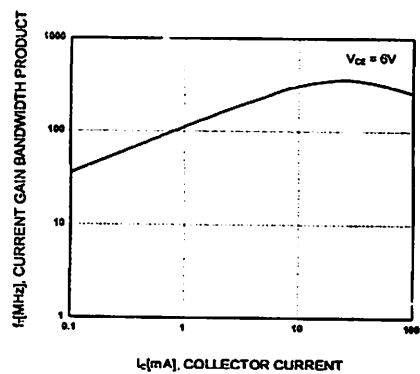
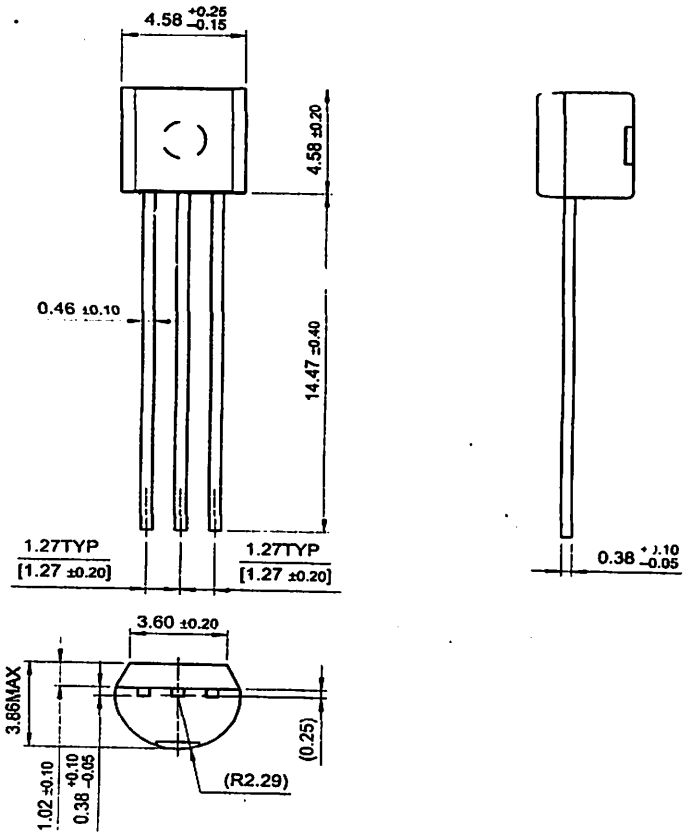


Figure 6. Current Gain Bandwidth Product

Package Dimensions

TO-92



Dimensions in Millimeters

TRADEMARKS

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

ACE ^x ™	FACT™	ImpliedDisconnect™	PACMAN™	SPM™
ActiveArray™	FACT Quiet series™	ISOPLANAR™	POP™	Stealth™
Bottomless™	FAST®	LittleFET™	Power247™	SuperSOT™-3
CoolFET™	FAST _r ™	MicroFET™	PowerTrench®	SuperSOT™-6
CROSSVOLT™	FRFET™	MicroPak™	QFET™	SuperSOT™-8
DOME™	GlobalOptoisolator™	MICROWIRE™	QS™	SyncFET™
EcoSPARK™	GTO™	MSX™	QT Optoelectronics™	TinyLogic™
E ² CMOS™	HiSeC™	MSXPro™	Quiet Series™	TruTranslation™
EnSigna™	i ² C™	OCX™	RapidConfigure™	UHC™
Across the board. Around the world.™		OCXPro™	RapidConnect™	UltraFET®
The Power Franchise™		OPTOLOGIC®	SILENT SWITCHER®	VCX™
Programmable Active Droop™		OPTOPLANAR™	SMART START™	

DISCLAIMER

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

LIFE SUPPORT POLICY

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

As used herein:

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

PRODUCT STATUS DEFINITIONS

Definition of Terms

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

PACKAGING INFORMATION

Orderable Device	Status ⁽¹⁾	Package Type	Package Drawing	Pins	Package Qty	Eco Plan ⁽²⁾	Lead/Ball Finish	MSL Peak Temp ⁽³⁾
MAX232D	ACTIVE	SOIC	D	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DE4	ACTIVE	SOIC	D	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DR	ACTIVE	SOIC	D	16	2500	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DRE4	ACTIVE	SOIC	D	16	2500	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DW	ACTIVE	SOIC	DW	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DWE4	ACTIVE	SOIC	DW	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DWR	ACTIVE	SOIC	DW	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232DWRE4	ACTIVE	SOIC	DW	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
M/ X232ID	ACTIVE	SOIC	D	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDE4	ACTIVE	SOIC	D	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDR	ACTIVE	SOIC	D	16	2500	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDRE4	ACTIVE	SOIC	D	16	2500	Green (RoHS & no Sb/Br)	CJ NIPDAU	Level-1-260C-UNLIM
MAX232IDW	ACTIVE	SOIC	DW	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDWE4	ACTIVE	SOIC	DW	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDWG4	ACTIVE	SOIC	DW	16	40	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDWR	ACTIVE	SOIC	DW	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDWRE4	ACTIVE	SOIC	DW	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IDWRG4	ACTIVE	SOIC	DW	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232IN	ACTIVE	PDIP	N	16	25	Pb-Free (RoHS)	CU NIPDAU	N / A for Pkg Type
MAX232INE4	ACTIVE	PDIP	N	16	25	Pb-Free (RoHS)	CU NIPDAU	N / A for Pkg Type
MAX232N	ACTIVE	PDIP	N	16	25	Pb-Free (RoHS)	CU NIPDAU	N / A for Pkg Type
MAX232NE4	ACTIVE	PDIP	N	16	25	Pb-Free (RoHS)	CU NIPDAU	N / A for Pkg Type
MAX232NSR	ACTIVE	SO	NS	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM
MAX232NSRE4	ACTIVE	SO	NS	16	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM

⁽¹⁾ The marketing status values are defined as follows:

ACTIVE: Product device recommended for new designs.

LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.

NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

PREVIEW: Device has been announced but is not in production. Samples may or may not be available.

OBSOLETE: TI has discontinued the production of the device.

⁽²⁾ **Eco Plan** - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check <http://www.ti.com/productcontent> for the latest availability information and additional product content details.

TBD: The Pb-Free/Green conversion plan has not been defined.

Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.

Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.

Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

⁽³⁾ **MSL, Peak Temp.** - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

Important Information and Disclaimer: The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

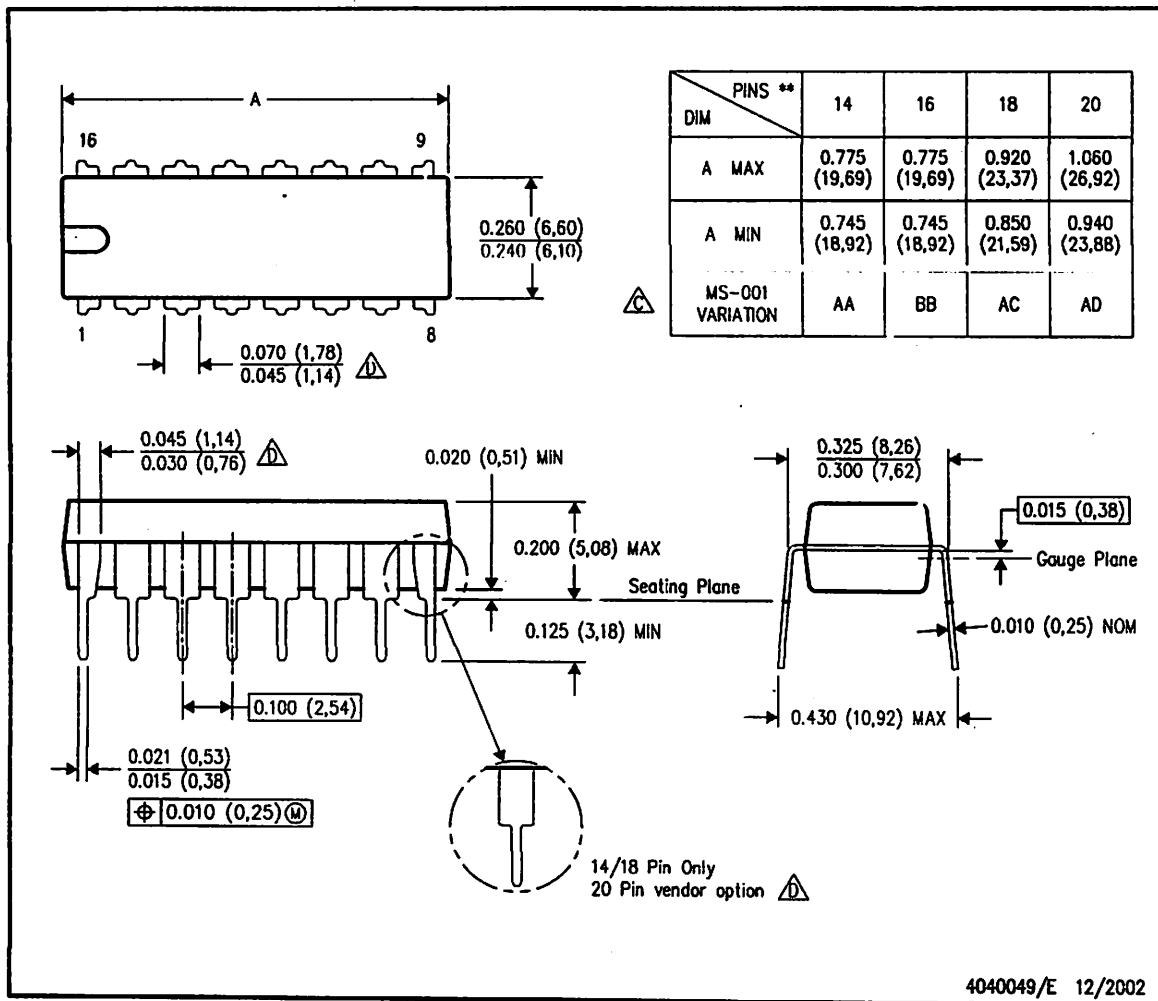
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

MECHANICAL DATA

N (R-PDIP-T**)

PLASTIC DUAL-IN-LINE PACKAGE

16 PINS SHOWN

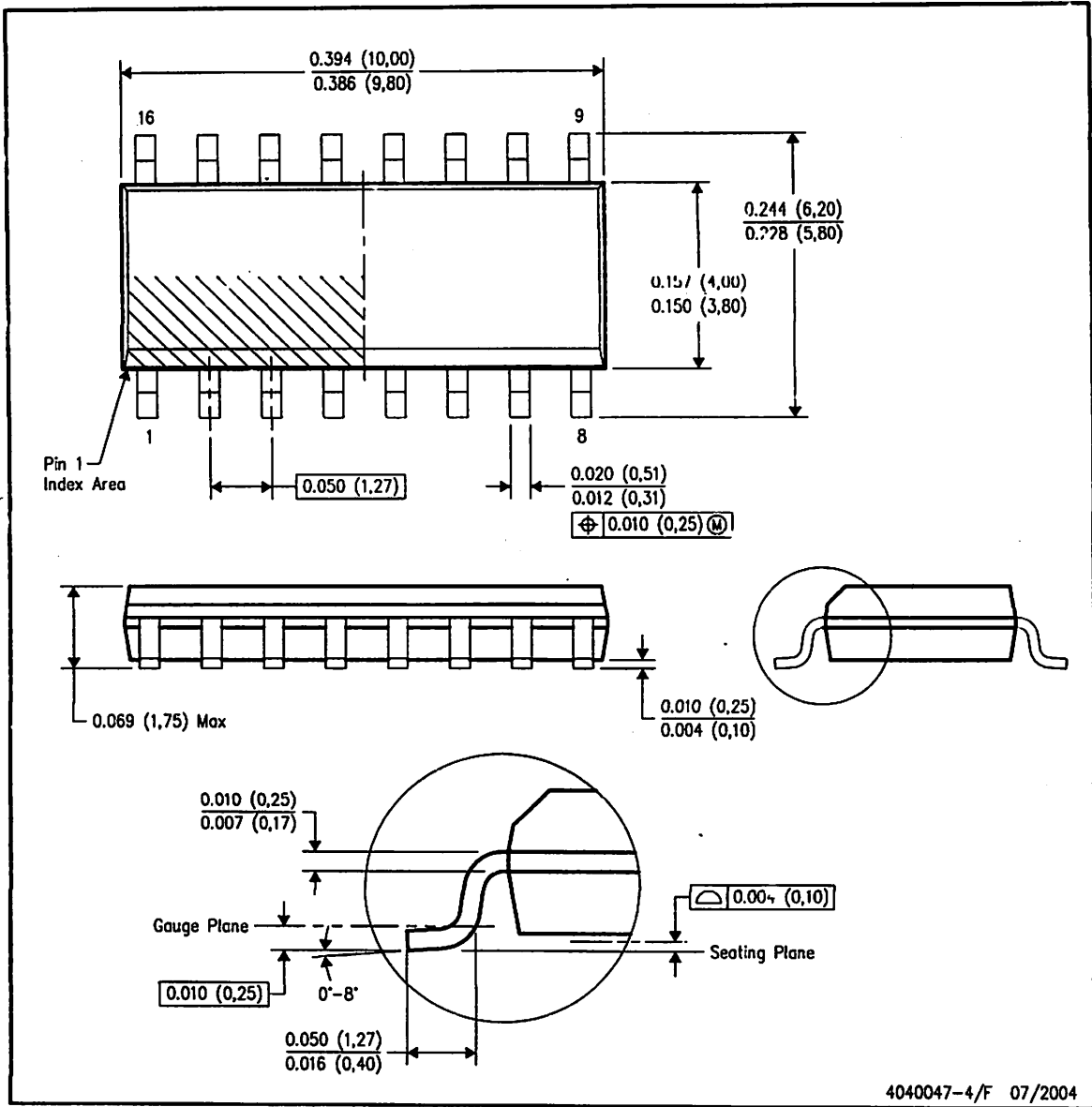


- NOTES:
- A. All linear dimensions are in inches (millimeters).
 - B. This drawing is subject to change without notice.
 - Falls within JEDEC MS-001, except 18 and 20 pin minimum body length (Dim A).
 - The 20 pin end lead shoulder width is a vendor option, either half or full width.

MECHANICAL DATA

D (R-PDSO-G16)

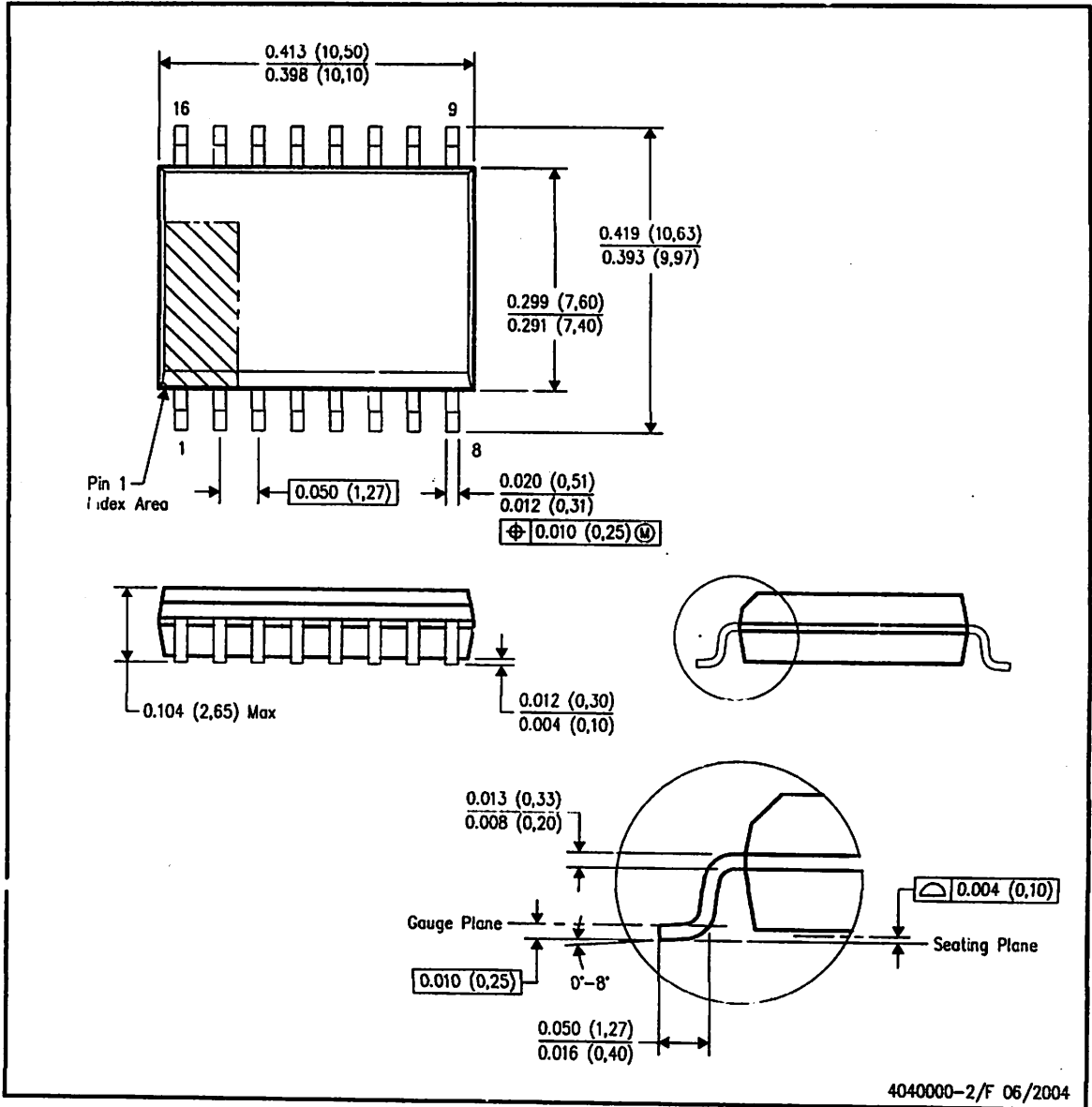
PLASTIC SMALL-OUTLINE PACKAGE



- NOTES:
- A. All linear dimensions are in inches (millimeters).
 - B. This drawing is subject to change without notice.
 - C. Body dimensions do not include mold flash or protrusion not to exceed 0.006 (0,15).
 - D. Falls within JEDEC MS-012 variation AC.

DW (R-PDSO-G16)

PLASTIC SMALL-OUTLINE PACKAGE



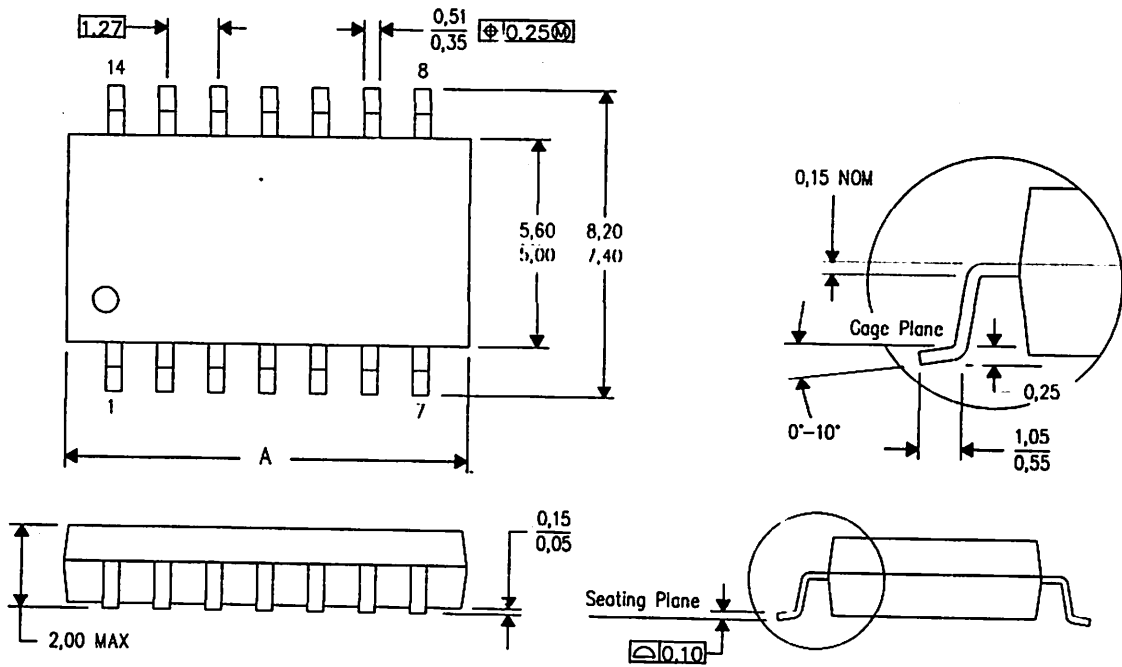
- NOTES:
- A. All linear dimensions are in inches (millimeters).
 - B. This drawing is subject to change without notice.
 - C. Body dimensions do not include mold flash or protrusion not to exceed 0.006 (0,15).
 - D. Falls within JEDEC MS-013 variation AA.

MECHANICAL DATA

NS (R-PDSO-G**)

PLASTIC SMALL-OUTLINE PACKAGE

14-PINS SHOWN



DIM \ PINS **	14	16	20	24
A MAX	10,50	10,50	12,90	15,30
A MIN	9,90	9,90	12,30	14,70

4040062/C 03/03

- NOTES:
- A. All linear dimensions are in millimeters.
 - B. This drawing is subject to change without notice.
 - C. Body dimensions do not include mold flash or protrusion, not to exceed 0,15.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warrant, or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		Applications	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
Low Power Wireless	www.ti.com/lpw	Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Copyright © 2006, Texas Instruments Incorporated

SS9014

SS9014

Pre-Amplifier, Low Level & Low Noise

- High total power dissipation. ($P_T=450\text{mW}$)
- High h_{FE} and good linearity
- Complementary to SS9015



TO-92

1. Emitter 2. Base 3. Collector

NPN Epitaxial Silicon Transistor

Absolute Maximum Ratings $T_a=25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Ratings	Units
V_{CBO}	Collector-Base Voltage	50	V
V_{CEO}	Collector-Emitter Voltage	45	V
V_{EBO}	Emitter-Base Voltage	5	V
I_C	Collector Current	100	mA
P_C	Collector Power Dissipation	450	mW
T_J	Junction Temperature	150	$^\circ\text{C}$
T_{STG}	Storage Temperature	-55 ~ 150	$^\circ\text{C}$

Electrical Characteristics $T_a=25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units
BV_{CBO}	Collector-Base Breakdown Voltage	$I_C=100\mu\text{A}, I_E=0$	50			V
BV_{CEO}	Collector-Emitter Breakdown Voltage	$I_C=1\text{mA}, I_B=0$	45			V
BV_{EBO}	Emitter-Base Breakdown Voltage	$I_E=100\mu\text{A}, I_C=0$	5			V
I_{CBO}	Collector Cut-off Current	$V_{CB}=50\text{V}, I_E=0$			50	nA
I_{EBO}	Emitter Cut-off Current	$V_{EB}=5\text{V}, I_C=0$			50	nA
h_{FE}	DC Current Gain	$V_{CE}=5\text{V}, I_C=1\text{mA}$	60	280	1000	
$V_{CE(sat)}$	Collector-Base Saturation Voltage	$I_C=100\text{mA}, I_B=5\text{mA}$		0.14	0.3	
$V_{BE(sat)}$	Base-Emitter Saturation Voltage	$I_C=100\text{mA}, I_B=5\text{mA}$		0.84	1.0	V
$V_{BE(on)}$	Base-Emitter On Voltage	$V_{CE}=5\text{V}, I_C=2\text{mA}$	0.58	0.63	0.7	V
C_{ob}	Output Capacitance	$V_{CB}=10\text{V}, I_E=0$ $f=1\text{MHz}$		2.2	3.5	pF
f_T	Current Gain Bandwidth Product	$V_{CE}=5\text{V}, I_C=10\text{mA}$	150	270		MHz
NF	Noise Figure	$V_{CE}=5\text{V}, I_C=0.2\text{mA}$ $f=1\text{KHz}, R_S=2\text{K}\Omega$		0.9	10	dB

h_{FE} Classification

Classification	A	B	C	D
h_{FE}	60 ~ 150	100 ~ 300	200 ~ 600	400 ~ 1000

Typical Characteristics

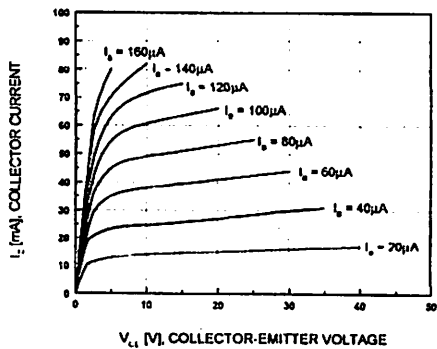


Figure 1. Static Characteristic

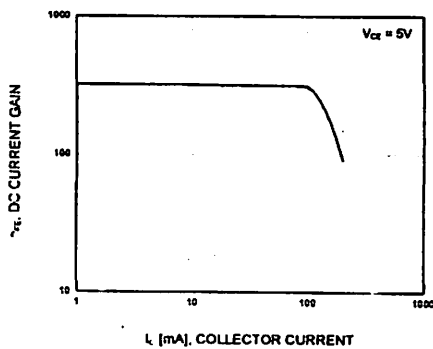


Figure 2. DC current Gain

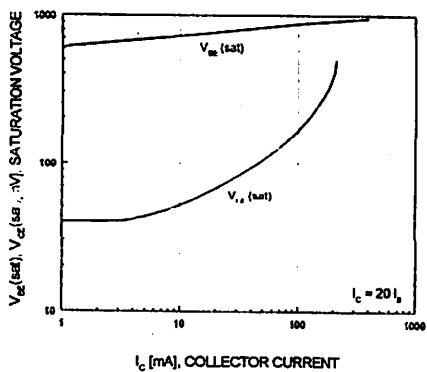


Figure 3. Base-Emitter Saturation Voltage
Collector-Emitter Saturation Voltage

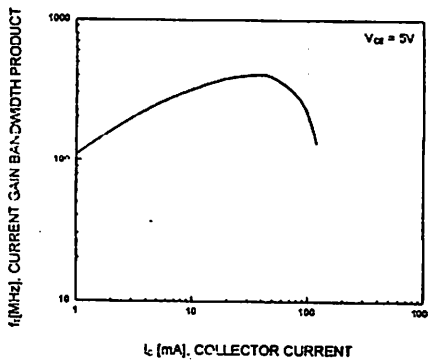
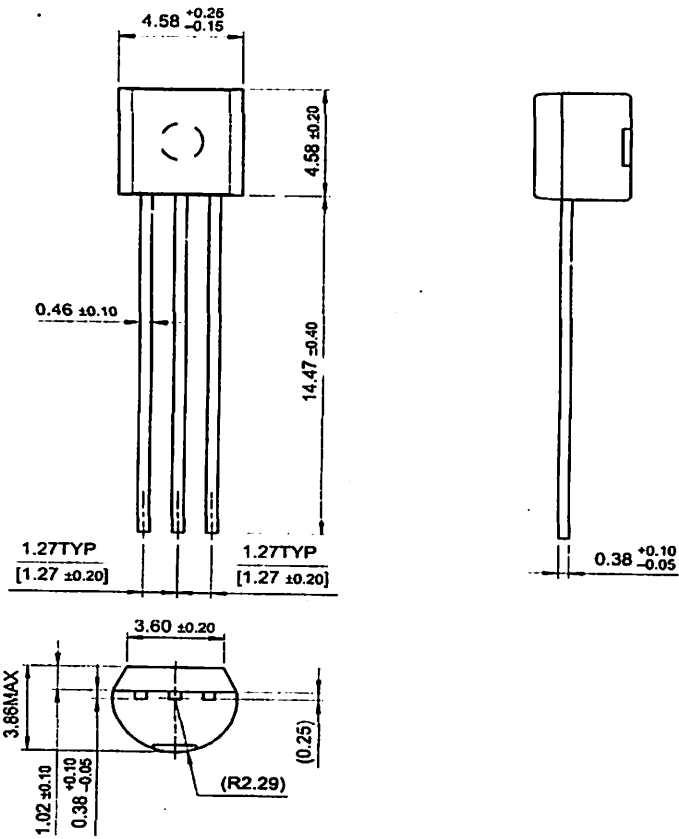


Figure 4. Current Gain Bandwidth Product

Package Dimensions

TO-92



Dimensions in Millimeters

TRADEMARKS

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

ACEx™	FACT™	ImpliedDisconnect™	PACMAN™	SPM™
ActiveArray™	FACT Quiet series™	ISOPANAR™	POP™	Stealth™
Bottomless™	FAST®	LittleFET™	Power247™	SuperSOT™-3
CoolLT™	FASTr™	MicroLT™	PowerTrench®	SuperSOT™-6
CROSSVOLT™	FRFET™	MicroPak™	QFET™	SuperSOT™-8
DOME™	GlobalOptoisolator™	MICROWIRE™	QS™	SyncFET™
EcoSPARK™	GTO™	MSX™	QT Optoelectronics™	TinyLogic™
E ² C MOS™	HiSeC™	MSXPro™	Quiet Series™	TruTranslation™
EnSigna™	IP™	OCX™	RapidConfigure™	UHC™
Across the board. Around the world.™		OCXPro™	RapidConnect™	UltraFET®
The Power Franchise™		OPTOLOGIC®	SILENT SWITCHER®	VCX™
Programmable Active Droop™		OPTOPLANAR™	SMART START™	

DISCLAIMER

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

LIFE SUPPORT POLICY

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

As used herein:

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

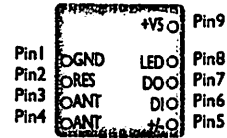
PRODUCT STATUS DEFINITIONS

Definition of Terms

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

ID-10 SERIES LOW COST PROXIMITY READER

BOOTOM VIEW



Power Requirement : 5V@13mA nominal

Card Format : EM4001 or compatible

Frequency : 125KHz

Encoding : Manchester 62bit, modulus 64

I/O Output Current : 20mA sink/source

Drive Current : 300mA

Antenna Volt : 100 Volt PKPK

OUTPUT FORMAT-ASCII

02 10 ASCII Data Characters Checksum CR LF 03

The checksum is the result of the 'exclusive or' of the 5 Binary Data bytes (the 10 ASCII data characters)

DATA STRUCTURE WIEGAND 26 BIT

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

P E E E E E E E E E E E E O O O O O O O O O O O P

EVEN Parity(E) ODD Parity(O)

P=Parity Start Bit and Stop Bit

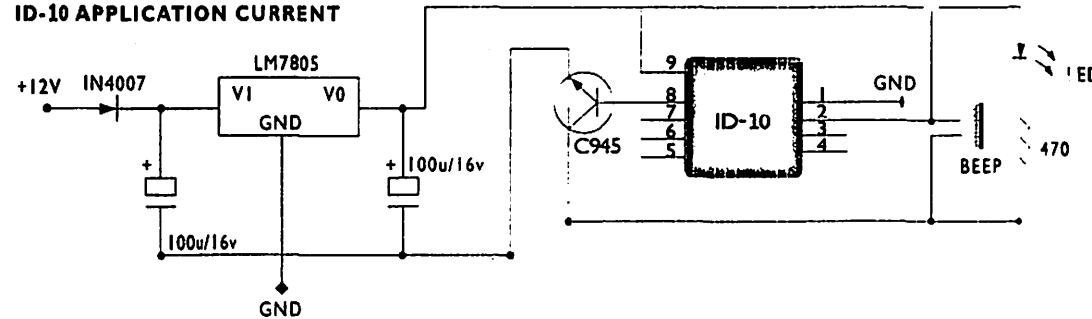
ASCII (RS232)

Pin1	Ground 0V	Zero volts and Tuning Capacitor Ground
Pin2	Reset Bar	Strap to +5V
Pin3	Antenna	NC
Pin4	Antenna	NC
Pin5	Strap to Ground	
Pin6	CMOS	Serial ASCII
Pin7	TTL Data	Serial ASCII inverted
Pin8	BEEP/LED	2.7KHz Logic
Pin9	+4.6 through +5.5V	Supply DC volts

WIEGAND 26

Pin1	Ground 0V	Zero volts and Tuning Capacitor Ground
Pin2	Reset Bar	Strap to +5V
Pin3	Antenna	NC
Pin4	Antenna	NC
Pin5	Strap to Pin +5V	
Pin6	One Output	
Pin7	Zero Output	
Pin8	BEEP/LED	2.7KHz Logic
Pin9	+4.6 through +5.5V	Supply DC volts

ID-10 APPLICATION CURRENT



ASCII Table	Scan Codes & EBCDIC	HTML Codes	Unicode V4 Tables	Domain-Reg Only \$5.95!	AllConversions Unit Converter	ToneShop RingTones	Mobile Games	Link2Me Exchange
-------------	---------------------	------------	-------------------	-------------------------	-------------------------------	--------------------	--------------	------------------

ASCII Table and Description

ASCII stands for American Standard Code for Information Interchange. Computers can only understand numbers, so an ASCII code is the numerical representation of a character such as 'a' or '@' or an action of some sort. ASCII was developed a long time ago and now the non-printing characters are rarely used for their original purpose. Below is the ASCII character table and this includes descriptions of the first 32 non-printing characters. ASCII was actually designed for use with teletypes and so the descriptions are somewhat obscure. If someone says they want your CV however in ASCII format, all this means is they want 'plain' text with no formatting such as tabs, bold or underscoring - the raw format that any computer can understand. This is usually so they can easily import the file into their own applications without issues. Notepad.exe creates ASCII text, or in MS Word you can save a file as 'text only'

Hx Oct Char	Dec Hx Oct	Html Chr	Dec Hx Oct	Html Chr	Dec Hx Oct	Html Chr
0 000 NUL (null)	32 20 040	 Space	64 40 100	@ @	96 60 140	` `
1 001 SOH (start of heading)	33 21 041	! !	65 41 101	A A	97 61 141	a a
2 002 STX (start of text)	34 22 042	" "	66 42 102	B B	98 62 142	b b
3 003 ETX (end of text)	35 23 043	# #	67 43 103	C C	99 63 143	c c
4 004 EOT (end of transmission)	36 24 044	$ \$	68 44 104	D D	100 64 144	d d
5 005 ENQ (enquiry)	37 25 045	% %	69 45 105	E E	101 65 145	e e
6 006 ACK (acknowledge)	38 26 046	& &	70 46 106	F F	102 66 146	f f
7 007 BEL (bell)	39 27 047	' '	71 47 107	G G	103 67 147	g g
8 010 BS (backspace)	40 28 050	((72 48 110	H H	104 68 150	h h
9 011 TAB (horizontal tab)	41 29 051))	73 49 111	I I	105 69 151	i i
A 012 LF (NL line feed, new line)	42 2A 052	* *	74 4A 112	J J	106 6A 152	j j
B 013 VT (vertical tab)	43 2B 053	+ +	75 4B 113	K K	107 6B 153	k k
C 014 FF (NP form feed, new page)	44 2C 054	, ,	76 4C 114	L L	108 6C 154	l l
D 015 CR (carriage return)	45 2D 055	- -	77 4D 115	M M	109 6D 155	m m
E 016 SO (shift out)	46 2E 056	. .	78 4E 116	N N	110 6E 156	n n
F 017 SI (shift in)	47 2F 057	/ /	79 4F 117	O O	111 6F 157	o o
10 020 DLE (data link escape)	48 30 060	0 0	80 50 120	P P	112 70 160	p p
11 021 DC1 (device control 1)	49 31 061	1 1	81 51 121	Q Q	113 71 161	q q
12 022 DC2 (device control 2)	50 32 062	2 2	82 52 122	R R	114 72 162	r r
13 023 DC3 (device control 3)	51 33 063	3 3	83 53 123	S S	115 73 163	s s
14 024 DC4 (device control 4)	52 34 064	4 4	84 54 124	T T	116 74 164	t t
15 025 NAK (negative acknowledge)	53 35 065	5 5	85 55 125	U U	117 75 165	u u
16 026 SYN (synchronous idle)	54 36 066	6 6	86 56 126	V V	118 76 166	v v
17 027 ETB (end of trans. block)	55 37 067	7 7	87 57 127	W W	119 77 167	w w
18 030 CAN (cancel)	56 38 070	8 8	88 58 130	X X	120 78 170	x x
19 031 EM (end of medium)	57 39 071	9 9	89 59 131	Y Y	121 79 171	y y
1A 032 SUB (substitute)	58 3A 072	: :	90 5A 132	Z Z	122 7A 172	z z
1B 033 ESC (escape)	59 3B 073	; ;	91 5B 133	[[123 7B 173	{ {
1C 034 FS (file separator)	60 3C 074	< <	92 5C 134	\ \	124 7C 174	|
1D 035 GS (group separator)	61 3D 075	= =	93 5D 135]]	125 7D 175	} }
1E 036 RS (record separator)	62 3E 076	> >	94 5E 136	^ ^	126 7E 176	~ ~
1F 037 US (unit separator)	63 3F 077	? ?	95 5F 137	_ _	127 7F 177	 DEL

Source: www.LookupTables.com

Extended ASCII Codes

As people gradually required computers to understand additional characters and non-printing characters the ASCII set became restrictive. As with technology, it took a while to get a single standard for these extra characters and hence there are few varying 'extended' sets. The most popular is presented below.

128	Ç	144	E	161	ı	177	ı	193	ı	209	ı	225	ı	241	ı
129	ü	145	æ	162	ó	178	ı	194	T	210	T	226	T	242	ı
130	é	146	Æ	163	ü	179		195	T	211	T	227	π	243	≤
131	â	147	ö	164	ñ	180	ı	196	-	212	E	228	Σ	244	ı
132	ä	148	ö	165	ñ	181	ı	197	+	213	F	229	ü	245	J
133	à	149	ó	166	°	182	ı	198	+	214	F	230	ı	246	ı
134	â	150	ó	167	°	183	ı	199	ı	215	ı	231	ı	247	ı
135	ç	151	ü	168	ı	184	ı	200	ı	216	ı	232	ı	248	ı
136	é	152	ı	169	ı	185	ı	201	ı	217	J	233	ı	249	ı
137	ê	153	ı	170	ı	186	ı	202	ı	218	T	234	ı	250	ı
138	è	154	ü	171	ı	187	ı	203	ı	219	ı	235	ı	251	ı
139	ı	156	ı	172	ı	188	ı	204	ı	220	ı	236	ı	252	ı
140	ı	157	ı	173	ı	189	ı	205	=	221	ı	237	ı	253	ı
141	ı	158	ı	174	ı	190	ı	206	ı	222	ı	238	ı	254	ı
142	ı	159	ı	175	ı	191	ı	207	ı	223	ı	239	ı	255	ı
143	ı	160	ı	176	ı	192	ı	208	ı	224	ı	240	ı		

Source : www.LookupTables.com

Useful Sites

[Download Free MP3](#)

[Music!](#)

PixelTraffic.com


Link Trading System

How quick are you?

Press 'start', then press 'stop'
 when the border goes

red



INOVASI *online*

KATEGORI ARTIKEL

- Opik utama
- ditorial
- edaksi
- putan khusus
- asional
- okoh
- umaniora
- esehatan
- ovasi
- teknik
- uku
- at
- orum



ISSN : 0917-8376 | Edisi Vol.1/XVI/Agustus 2004 - IPTEK

HALAMAN UTAMA

RFID, Sebuah Teknologi Identifikasi Pengancam Privasi

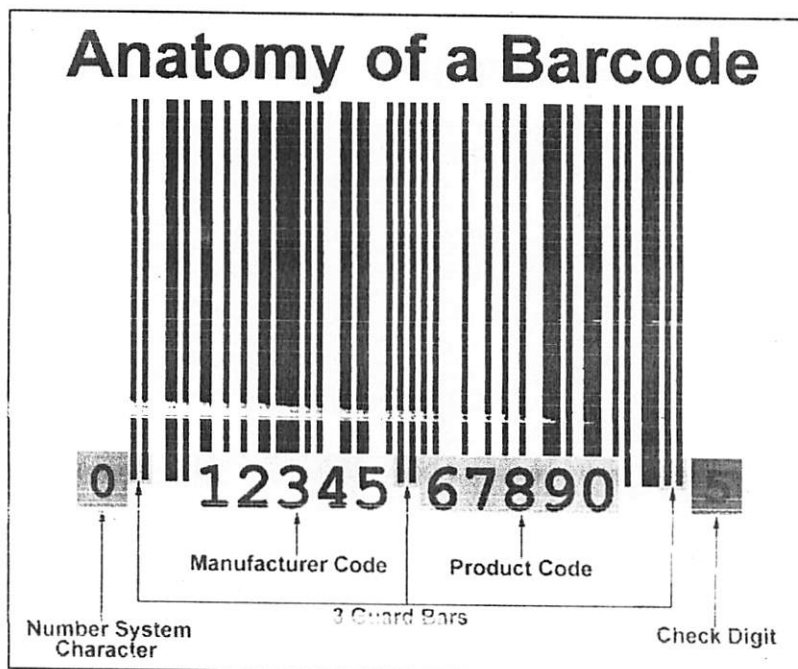
Oleh : Danardono Dwi Antono



Anda tentu mengenal barcode? Sebuah kode-kode tertentu yang diekspresikan dengan susunan garis-garis hitam (*bar*) dan putih (*space*) yang berbeda ketebalannya seperti ditunjukkan pada Gambar 1. Kebutuhan terhadap identifikasi keberadaan suatu barang (*item*) secara otomatis (*Auto-ID*) di bidang industri, perdagangan dan logistik melahirkan penggunaan barcode ini lebih dari 30 tahun yang lalu.

EDISI LAIN

- Vol.6/XVIII/Mar 2006
- Vol.5/XVII/Nov 2005
- Vol.4/XVII/Agt 2005
- Vol.3/XVII/Mar 2005
- Vol.2/XVI/Nov 2004



Gambar 1. Barcode.

Buku-buku dan produk-produk yang dijual di toko buku dan supermarket misalnya selalu dilengkapi dengan barcode untuk memudahkan identifikasinya saat pendataan atau pembayaran di kasir dengan menggunakan sebuah pembaca (*reader*). Satu set barcode terdiri dari beberapa kode, dimana satu kode terdiri dari 7 unit garis hitam atau putih yang jika berdampingan akan terlihat garis hitam atau putih yang lebih tebal.

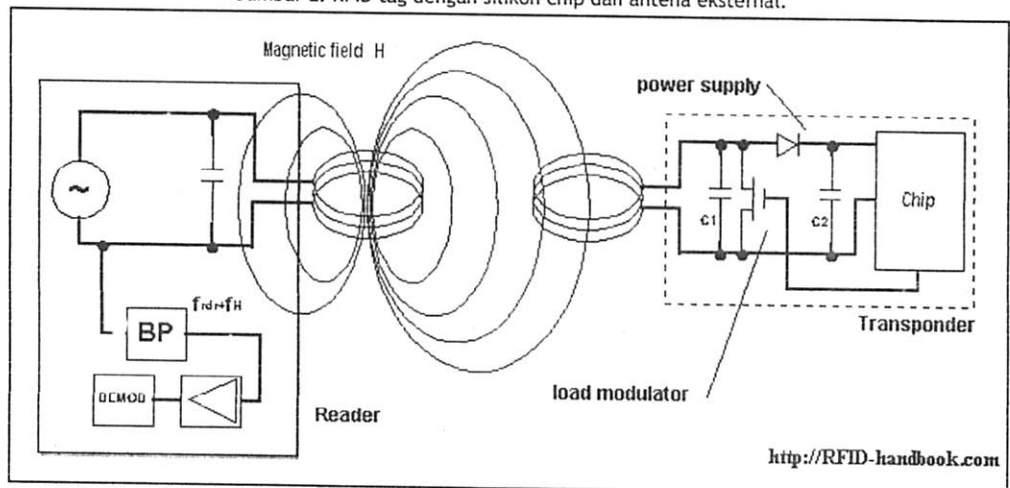
1. Dari barcode menuju ke RFID

Walau terbukti murah dan dapat dipakai di berbagai bidang, barcode ini ternyata mempunyai banyak kelemahan, yaitu selain karena hanya bisa diidentifikasi dengan cara mendekatkan barcode tersebut ke sebuah reader, juga karena mempunyai kapasitas penyimpanan data yang sangat terbatas dan tidak bisa diprogram ulang sehingga menyulitkan untuk menyimpan dan memperbaharui data dalam jumlah besar untuk sebuah *item*.

Salah satu solusi menarik yang kemudian muncul adalah menyimpan data tersebut pada suatu silikon chip. RFID merupakan singkatan dari *Radio Frequency Identification* merupakan teknologi identifikasi baru yang dalam pengoperasiannya terjadi kontak antara *transponder (tag)* atau divais pembawa data yang terbuat dari silikon dilengkapi sebuah radio antena kecil dan reader yang terhubung dengan sistem komputer. Kontak antara RFID dengan reader tidak dilakukan secara kontak langsung atau mekanik melainkan dengan pengiriman gelombang elektromagnet. Berbeda dengan *smart card* yang biasa dipakai di kartu telepon atau kartu bank yang juga menggunakan silikon chip, kode-kode RFID tag bisa dibaca pada jarak yang cukup jauh.



Gambar 2. RFID tag dengan silikon chip dan antena eksternal.



Gambar 3. Komunikasi antara reader dan transponder (tag)

Sekarang ini RFID tag standard biasanya mampu menyimpan tidak lebih dari 128 bit. Sebagian besar memori dipakai untuk kode produk elektronik yang berisi informasi produsen, jenis produk, dan nomor serial. Karena RFID tag adalah unik, maka dua buah kaleng minuman ringan dengan jenis yang sama akan memiliki kode yang berbeda, dimana sebaliknya jika menggunakan barcode semua produk sejenis akan menggunakan kode yang sama. Perbedaan lain antara barcode dan RFID adalah, RFID tag memerlukan sumber tenaga listrik untuk menggerakkan sirkuit rangkaian terpadu di dalam tag tersebut, dan biasanya dan tentunya RFID tag tidak bisa menggunakan baterai yang membuat biayanya menjadi mahal. Pemecahannya adalah dengan cara mengirimkan energi listrik melalui medan elektromagnet dari reader ke RFID tag. Sebaliknya reader dapat membaca banyak RFID tag dalam waktu bersamaan dalam jarak antara beberapa cm sampai 10 meter atau lebih.

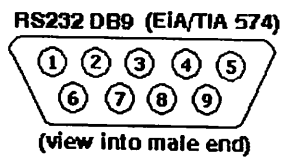
2. Aplikasi sederhana RFID

Untuk sebuah produk hasil pertanian yang dijual di supermarket, jika selama ini dengan menggunakan barcode hanya data jenis produk yang mampu tersimpan, di masa datang diharapkan RFID tag mampu menyimpan tidak hanya data jenis produk namun juga misalnya untuk sebuah produk beras dapat diketahui daerah asal produk beras, kapan beras itu pertama kali ditanam dan dipanen, metode penanaman dan pembuatannya, bahkan riwayat data petaninya secara otomatis. Keuntungan lain adalah kasir maupun pembeli dapat mengetahui total harga

Pin-pin yang terdapat pada RS-232 :

Tabel 2.1 : Keterangan Pin RS-232

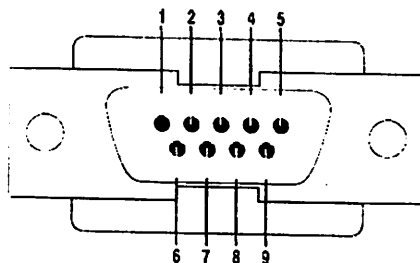
Pin No.	Name	Notes/Description
1	DCD	Data Carrier Detect
2	RD	Receive Data (a.k.a RxD, Rx)
3	TD	Transmit Data (a.k.a TxD, Tx)
4	DTR	Data Terminal Ready
5	SGND	Ground
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI	Ring Indicator



Gambar 2.10 : Pin RS-232

2.10 KONEKTOR DB9

Pada Perancangan ini rakaian pengukur suhu menggunakan konektor DB9 female, pin yang dipakai 4 pin seperti yang terlihat pada gambar 2.6 dibawah ini.



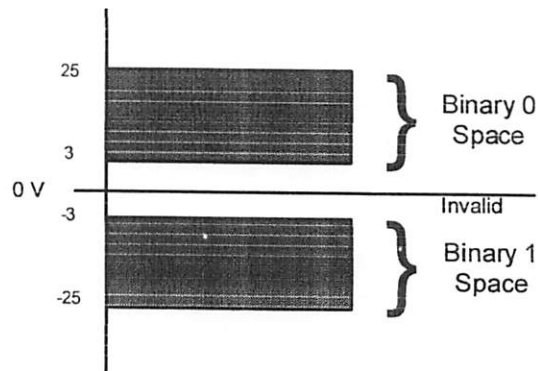
Gambar 2.11. DB9 Female

Alat ini disebut interface. Karena interface merupakan semacam pintu gerbang maka interface sering disebut sebagai I/O port.

Karena itulah maka pada perancangan ini penulis mengambil level tegangan RS232 pada komputer menggunakan konektor DB9. rangkaian pengukur suhu ini tidak membutuhkan power supply dari luar sehingga tegangan yang diambil dari level tegangan RS232 pada komputer. Kegunaan RS 232 adalah sebagai driver, yang akan mengkonversi tegangan dari hardware agar sesuai dengan tegangan pada komputer sehingga data dapat dibaca.

Pada RS232 biner 1 disebut mark dan bisa memiliki tegangan dari -3 sampai -25 volt. Biner 0 disebut space dan memiliki jangkauan tegangan antara +3 sampai +25 volt.

Dibawah ini dapat terlihat gambar 2.9 yang menggambarkan level tegangan pada RS232.[11,12]



Gambar 2.9 : Jangkauan Tegangan RS-232