

# SKRIPSI

## PERANCANGAN DAN PEMBUATAN PENAMPIL NOMOR DAN JUDUL LAGU YANG DIPERGUNAKAN DI GEREJA BERBASIS MIKROKONTROLER AT89S52



Disusun Oleh :

**PAULO DAS DORES E SILVA**

**041 22 05**



**JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK ELEKTRONIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
SEPTEMBER 2009**

DEPARTMENT 3000

INSTITUT TEKNOLOGI NASIONAL SURABAYA

PUSAT PENELITIAN DAN PENGABDIAN

KEPERAWATAN LINGKUNGAN BERTANAMAN

JURUSAN TEKNIK PERTANIAN 2-1

011 33 02

MALINDANG DAN SURABAYA

Surabaya 01111



BERSERVIS MENDUKUNG/BUKTIKASI ATAS  
JURUSAN TEKNIK PERTANIAN DI GEDUNG  
PENGABDIAN DAN KEMAJUAN BERKUALITAS MANUSIA

SKRIPSI

# LEMBAR PERSETUJUAN

## PERANCANGAN DAN PEMBUATAN PENAMPIL NOMOR DAN JUDUL LAGU YANG DIPERGUNAKAN DI GEREJA BERBASIS MIKROKONTROLER AT89S52

### SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh*

*Gelar Sarjana Teknik Elektronika Strata Satu (S-1)*


Disusun Oleh :

**PAULO DAS DORES E SILVA**

**041 22 05**

Diperiksa dan Disetujui :

Dosen Pembimbing I



**Ir. TH. Mimien Mustikawati, MT**  
NIP.Y. 103 000 0352

Dosen Pembimbing II



**Sotvohadi, ST**  
NIP.Y. 103 970 0309

Mengetahui,

Ketua Jurusan Teknik Elektro S-1



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

**JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK ELEKTRONIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2009**





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

**BERITA ACARA UJIAN SKRIPSI**

**FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Paulo Das Dores E Silva  
Nim : 041 22 05  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Elektronika  
Judul Skripsi : Perancangan Dan Pembuatan Penampil Nomor Dan Judul Lagu  
Yang Dipergunakan Di Gereja Berbasis Mikrokontroler AT89S52

Dipersidankan di hadapan majelis penguji Skripsi jenjang Sarjana Strata satu (S-1) pada:

Hari : Selasa  
Tanggal : 29 September 2009  
Dengan Nilai : 76,6 (B+) *BY*



**Ketua Majelis Penguji**

(Ir. Sidik Noertjahjono, MT)  
NIP.Y.102 870 0163

**Sekretaris Majelis Penguji**

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

**Penguji I**

(Ir. M. Ibrahim Ashari, ST, MT)  
NIP.P.103 010 0358

**Penguji II**

(Irmalia Suryani Faradisa, ST)  
NIP.P.103 000 0365



**PERANCANGAN DAN PEMBUATAN PENAMPIL NOMOR DAN JUDUL LAGU  
YANG DIPERGUNAKAN DI GEREJA BERBASIS MIKROKONTROLER  
AT89S52**

**Paulo Das Dores E Silva**

**041 22 05**

Jurusan Teknik Elektronika S-1. Institut Teknologi Nasional Malang

*Email:*

[paulo\\_silva04@yahoo.com](mailto:paulo_silva04@yahoo.com)

**Dosen Pembimbing:**

Ir. TH. Mimien Mustikawati, MT

NIP.Y. 103 000 0352

Sotyohadi, ST

NIP.Y. 103 970 0309

***Abstrak***

*Pada zaman sekarang, keyboard PC sudah sangatlah biasa dipergunakan untuk memasukan/mengetik suatu karakter atau informasi. Bagaimana pun juga keyboard PC masih tergantung pada monitor dan komputer (PC) untuk menampilkan karakter atau informasi, mungkin masih sangat sulit untuk menampilkan suatu informasi atau karakter yang diinginkan.*

*Pada perancangan ini, penggerakkan pada dot matriks LED menggunakan sistem scan kolom dan sistem scan kode keyboard PC yang bisa mendapatkan kembali data dari keyboard PC. Bagaimana caranya menghubungkan keyboard PC ke dot matriks LED, dan menampilkan karakter atau informasi yang berasal dari keyboard PC. Oleh karena itu kita membutuhkan dua buah mikrokontroler yang saling terhubung satu sama lain. Terpilihnya mikrokontroler AT89S52 karena memiliki internal memori (RAM) dan port masukan/keluaran yang mampu untuk menghubungkan ke keyboard PC dan dot matriks LED.*

*Kata kunci: Mikrokontroler, Keyboard PC, Dot matriks LED*

## **KATA PENGANTAR**

Puji Syukur saya selaku penulis panjatkan ke hadirat Tuhan Yang Maha Esa yang telah memberikan kesehatan, semangat, kemampuan berpikir dan segala-Nya kepada saya, sehingga penulis dapat menyelesaikan penulisan dan perancangan skripsi ini dengan baik dan lancar.

Penulisan skripsi ini merupakan suatu kewajiban yang harus ditempuh oleh setiap mahasiswa program studi Teknik Elektro Jurusan Teknik Elektronika S-1 Institut Teknologi Nasional Malang, sebagai syarat utama untuk mendapatkan gelar Sarjana Teknik Strata Satu/ST.

Dalam penyelesaian skripsi ini yang berjudul “Perancangan Dan Pembuatan Penampil Nomor Dan Judul Lagu Yang Dipergunakan Di Gereja Berbasis Mikrokontroler AT89S52”, terselesaikannya skripsi ini karena adanya bantuan dari berbagai pihak baik itu secara langsung maupun tidak langsung. Oleh karena itu penyusun mengucapkan banyak terima kasih kepada:

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang
2. Bapak Ir. Sidik Noertjahjono, MT selaku Dekan Elektro
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Kajur Elektro
4. Bapak Ir. Yusuf Ismail Nahkoda, MT selaku Sekretaris Jurusan Elektro
5. Ibu Ir. TH. Mimien Mustikawati, MT selaku dosen Pembimbing I
6. Bapak Sotyohadi, ST selaku dosen Pembimbing II
7. Rekan – rekan mahasiswa yang telah membantu dalam penyelesaian skripsi ini.

Penyusun menyadari bahwa dalam penyusunan skripsi ini masih terdapat kekurangan yang berupa sistem kerja pada semua tombol *Keyboard PC*, namun demikian saya berharap skripsi ini dapat bermanfaat bagi penulis dan penggunaan sistem ini, khususnya di gereja - gereja

Akhir kata semoga skripsi ini dapat memberikan manfaat bagi pembacanya.

Malang, September 2009

Penulis

## DAFTAR ISI

<b>LEMBAR PERSETUJUAN...</b>	<b>i</b>
<b>BERITA ACARA UJIAN SKRIPSI.....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>ABSTRAK.....</b>	<b>iv</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR.....</b>	<b>x</b>
<b>DAFTAR TABEL.....</b>	<b>xiii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
<b>1.1. LATAR BELAKANG.....</b>	<b>1</b>
<b>1.2. RUMUSAN MASALAH.....</b>	<b>2</b>
<b>1.3. TUJUAN.....</b>	<b>2</b>
<b>1.4. BATASAN MASALAH.....</b>	<b>3</b>
<b>1.5. METODELOGI PEMBAHASAN.....</b>	<b>4</b>
<b>1.6. SISTEMATIKA PENULISAN.....</b>	<b>4</b>
<b>BAB II LANDASAN TEORI.....</b>	<b>5</b>
<b>2.1. MIKROKONTROLER.....</b>	<b>5</b>
2.1.1. Deskripsi.....	5
2.1.2. Konfigurasi Pin AT89S52.....	6
2.1.3. Fungsi Masing – masing Pin AT89S52.....	7



2.1.4. Pewaktu CPU ( <i>Central Processing Unit</i> )..	10
2.1.5. Organisasi Memori.....	11
<b>2.2. EEPROM (<i>Electrically Erasable Programmable Read Only Memory</i>).....</b>	<b>17</b>
2.2.1. Diskripsi.....	17
2.2.2. Konfigurasi Pin EEPROM AT28C16.....	17
<b>2.3. Keyboard PC (<i>Personal Computer</i>).....</b>	<b>19</b>
2.3.1. Cara Kerja.....	19
2.3.2. Kode Perintah.....	22
<b>2.4. LCD (<i>Liquid Crystal Display</i>).....</b>	<b>24</b>
2.4.1. Operasi Dasar.....	30
2.4.2. Sinyal Interface M1632.....	32
<b>2.5. IC TTL (<i>Transistor – Transistor Logic</i>).....</b>	<b>34</b>
2.5.1. IC <i>Shift Register</i> 74LS164.....	34
2.5.1.1.Konfigurasi Pin 74LS164.....	35
2.5.2. IC D-Flip – flop 74LS373.....	36
2.5.2.1.Konfigurasi Pin 74LS373.....	37
<b>2.6. Dot Matriks LED.....</b>	<b>38</b>
2.6.1. Dot Matriks LED.....	38
2.6.2. Dekode Matriks LED.....	40
<b>2.7. Konektor PS2.....</b>	<b>40</b>
2.7.1. Konfigurasi Pin Konektor PS2.....	40

<b>BAB III PERANCANGAN DAN PEMBUATAN SISTEM.....</b>	<b>41</b>
<b>3.1. Diagram Blok Rangkaian.....</b>	<b>41</b>
<b>3.2. Perancangan Perangkat Keras (<i>Hardware</i>).....</b>	<b>42</b>
3.2.1. Perancangan Minimum Sistem AT89S52.....	43
3.2.1.1.Rangkaian <i>Clock</i> .. .....	43
3.2.1.2.Rangkaian <i>Reset</i> .. .....	44
3.2.1.3.Perancangan Penggunaan Port – port Pada AT89S52.. .....	46
3.2.2. Perancangan Minimum Sistem EEPROM AT28C16.. .....	50
3.2.3. Driver Dot Matriks.....	51
3.2.4. Perancangan Rangkaian Keyboard PC.....	53
3.2.5. Perancangan Rangkaian LCD.. .....	54
<b>3.3. Perancangan Perangkat Lunak (<i>Software</i>).....</b>	<b>55</b>
3.3.1. Flowchart Kerja Rangkaian .....	56
3.3.1.1.Flowchart Kerja Rangkaian Keseluruhan.. .....	56
3.3.1.2.Flowchart Kerja Pada Mikrokontroler I ( <i>Master</i> ).....	57
3.3.1.3.Flowchart Kerja Pada Mikrokontroler II ( <i>Slave</i> ).. .....	58
3.3.1.4.Flowchart Kerja Pada Keyboard PC.. .....	59
<b>BAB IV PENGUJIAN SISTEM.....</b>	<b>60</b>
<b>4.1. Pengujian Rangkaian Mikrokontroler I (<i>Master</i>).....</b>	<b>60</b>
4.1.1. Tujuan Pengujian.. .....	60
4.1.2. Langkah – langkah .. .....	61
4.1.3. Hasil dan Analisa.. .....	61

<b>4.2. Pengujian Rangkaian Mikrokontroler II (Slave).....</b>	<b>62</b>
4.2.1. Tujuan Pengujian.. .....	62
4.2.2. Langkah – langkah .. .....	62
4.2.3. Hasil dan Analisa.. .....	62
<b>4.3. Pengujian Keyboard PC dan Tampilan Ke LCD.. .....</b>	<b>63</b>
4.3.1. Tujuan Pengujian.. .....	63
4.3.2. Langkah – langkah .. .....	63
4.3.3. Hasil dan Analisa.. .....	63
<b>4.4. Pengujian Rangkaian Dot Matriks LED.. .....</b>	<b>70</b>
4.4.1. Tujuan Pengujian.. .....	70
4.4.2. Langkah – langkah .. .....	70
4.4.3. Hasil dan Analisa.. .....	70
<b>4.5. Pengujian Kelebihan Karakter Pada LCD.....</b>	<b>71</b>
4.5.1. Tujuan Pengujian.. .....	71
4.5.2. Langkah – langkah ... .....	71
4.5.3. Hasil dan Analisa.. .....	72
<b>4.6. Pengujian Pengiriman Data Antara Mikro I Dengan Mikro II.. .....</b>	<b>73</b>
4.6.1. Tujuan Pengujian.. .....	73
4.6.2. Langkah – langkah.. .....	73
4.6.3. Hasil dan Analisa.. .....	73
<b>4.7. Pengujian Rangkaian Keseluruhan Sistem.. .....</b>	<b>74</b>
4.7.1. Tujuan Pengujian.. .....	74
4.7.2. Langkah – langkah. ....	74



4.7.3. Hasil dan Analisa.....	75
<b>BAB V PENUTUP.....</b>	<b>77</b>
<b>5.1. Kesimpulan.....</b>	<b>77</b>
<b>5.2. Saran... ..</b>	<b>78</b>
<b>DAFTAR PUSTAKA.....</b>	<b>79</b>

#### **DAFTAR LAMPIRAN**

- **Lembar Persembahan**
- **Lembar Bimbingan Skripsi**
- **Gambar Rangkaian Keseluruhan Sistem**
- **Listening Program untuk MCU I & II**
- **All DataSheet**

## DAFTAR GAMBAR

### BAB II LANDASAN TEORI

Gambar 2.1 Diagram Blok Arsitektur Mikrokontroler AT89S52.....	6
Gambar 2.2 Konfigurasi Pin Mikrokontroler AT89S52.. .....	7
Gambar 2.3 Rangkaian Pewaktu Internal.....	10
Gambar 2.4 Rangkaian Pewaktu Eksternal.....	11
Gambar 2.5 Perbedaan signal Kendali Pada Memori Program dan Memori Data Eksternal.. .....	12
Gambar 2.6 Memori Data.. .....	13
Gambar 2.7 Organisasi Memori AT89S52.. .....	14
Gambar 2.8 Memori Data 128 awal IDATA dan Pengalamatan BIT.....	15
Gambar 2.9 Konfigurasi Pin AT28C16.. .....	18
Gambar 2.10 Diagram Blok AT28C16.. .....	19
Gambar 2.11 Keyboard PC dan Scan Code.. .....	21
Gambar 2.12.a Sinyal Keyboard Ke Host.....	21
Gambar 2.12.b Sinyal Host Ke Keyboard.....	22
Gambar 2.13 Rangkaian LCD ( <i>Liquid Crystal Display</i> ).. .....	26
Gambar 2.14 Susunan Alamat Pada LCD ( <i>Liquid Crystal Display</i> ).. .....	28
Gambar 2.15 LCD ( <i>Liquid Crystal Display</i> ).....	32
Gambar 2.16 Pola Karakter Pada LCD ( <i>Liquid Crystal Display</i> ).....	33
Gambar 2.17 Mengirim/mengambil Data Ke/dari M1632.....	34
Gambar 2.18 Konfigurasi Pin IC SN74LS164.....	35

Gambar 2.19 Logic Symbol IC SN74LS164..	35
Gambar 2.20 Diagram Logic IC SN74LS164.....	36
Gambar 2.21 Konfigurasi Pin IC SN74LS373.....	37
Gambar 2.22 Logic Diagram IC SN74LS373.....	37
Gambar 2.23 Internal Circuit Diagram Dot Matriks 5x7.....	38
Gambar 2.24 Package Dimension Dot Matriks 5x7.....	39
Gambar 2.25 Konektor PS2 (PS two)....	40

### **BAB III PERANCANGAN DAN PEMBUATAN SISTEM**

Gambar 3.1 Diagram Blok Sistem..	41
Gambar 3.2 Rangkaian Clock..	44
Gambar 3.3 Rangkaian Reset.....	45
Gambar 3.4 Perancangan Rangkaian Mikrokontroler I ( <i>Master</i> )..	47
Gambar 3.5 Perancangan Rangkaian Mikrokontroler II ( <i>Slave</i> ).....	49
Gambar 3.6 Perancangan Rangkaian AT28C16..	51
Gambar 3.7 Perancangan Driver Dot Matriks.....	52
Gambar 3.8 Perancangan Rangkaian Pada Keyboard PC.....	53
Gambar 3.9 Perancangan Rangkaian LCD ( <i>Liquid Crystal Display</i> ).....	54
Gambar 3.10 Flowchart Kerja Rangkaian Keseluruhan..	56
Gambar 3.11 Flowchart Kerja Pada Mikrokontroler I ( <i>Master</i> ).....	57
Gambar 3.12 Flowchart Kerja Pada Mikrokontroler II ( <i>Slave</i> )..	58
Gambar 3.13 Flowchart Kerja Pada Keyboard PC..	59



#### **BAB IV PENGUJIAN SISTEM**

Gambar 4.1 Hasil Pengujian Pada Mikrokontroler I ( <i>Master</i> ).....	61
Gambar 4.2 Hasil Pengujian Pada Mikrokontroler II ( <i>Slave</i> ).....	62
Gambar 4.3 Hasil Pengujian Keyboard PC Dengan Terminal v1.8 By Brayy ++.....	64
Gambar 4.4 Hasil Pengujian Keyboard PC Dan Tampilan Pada LCD.....	68
Gambar 4.5 Hasil Pengujian Penekanan Tombol Yang Tidak Diinginkan..	69
Gambar 4.6 Hasil Pengujian Pada Rangkaian Dot Matriks LED..	70
Gambar 4.7 Hasil Pengujian Kelebihan Karakter Pada LCD..	72
Gambar 4.8 Hasil Pengujian Pengiriman Data Antara Mikrokontroler I & II.....	73
Gambar 4.9 Hasil Pengujian Sistem Keseluruhan... ..	76

## DAFTAR TABEL

### BAB II LANDASAN TEORI

Tabel 2.1 Fungsi Khisis Port 3.....	9
Tabel 2.2 Special Function Register (SFR).....	16
Tabel 2.3 Konfigurasi Pin EEPROM AT28C16.....	19
Tabel 2.4 Fungsi Pada Tiap – tiap LCD.....	27
Tabel 2.5 Instruksi Pada LCD.....	28
Tabel 2.6 Pemilihan Register Pada LCD..	31

### BAB IV PENGUJIAN SISTEM

Tabel 4.1 Hasil Pengujian Tombol Abjad a-z dan A-Z Pada Keyboard PC.....	65
Tabel 4.2 Hasil Pengujian Tombol Angka (0-9) dan Simbol Pada Keyboard PC... ..	66
Tabel 4.3 Hasil Pengujian Keyboard PC.....	67
Tabel 4.4 Pergeseran Pada IC Register SN74LS164..	75

# **BAB I**

## **PENDAHULUAN**

### **1.1. LATAR BELAKANG**

Kemajuan teknologi dibidang elektronika dewasa ini berkembang dengan cepat dan berpengaruh dalam pembuatan alat – alat canggih, yaitu alat yang dapat bekerja secara otomatis dan memiliki ketelitian tinggi dengan bantuan mikrokontroler. Ada beberapa macam kontroler yang dapat digunakan, namun yang saat ini yang paling banyak digunakan adalah kontroler yang merupakan dari Mikroprosesor.

Sistem Mikroprosesor tidak dapat bekerja sendiri tanpa didukung oleh internal sistem (*Software*) dan eksternal sistem (*Hardware*). Apabila sebuah Mikroprosesor dikombinasikan dengan memori (ROM/RAM) dan unit – unit I/O maka akan menghasilkan sebuah mikrokomputer. Kombinasi ini dapat dibuat dalam satu level *Chip* yaitu *Chip* mikrokomputer atau sering disebut juga Mikrokontroler.

Pengunaan sebagai unit – unit kendali sudahlah sangat luas. Hal ini dikarenakan peralatan – peralatan yang dikontrol secara elektronik lebih banyak memberi kemudahan – kemudahan dalam penggunaannya.

Dalam hal ini, akan dibahas khusus mengenai papan angka (*score board*) atau dot matriks LED yang sudah lazim digunakan di kalangan umum atau masyarakat maupun dunia internasional, misalnya dunia olahraga atau entertainment lainnya, hal ini juga berpengaruh pada gereja – gereja yaitu dengan memanfaatkan papan angka yang dapat menampilkan angka atau nomor halaman dari buku nyanyian (Puji syukur), sehingga dapat memudahkan umat, namun papan angka yang ada sekarang ini masih memiliki beberapa kekurangan karena masih bekerja secara manual, yaitu hanya bisa diatur dengan tombol biasa dan langsung dihubungkan dengan papan angka, untuk itu

dalam penulisan skripsi ini, penulis ingin mengembangkannya dengan menggunakan *keyboard* PC, *Dot* matriks LED dan LCD, untuk memudahkan para pemakai dalam mengoperasikan papan angka atau display *dot* matriks LED yang berbasis Mikrokontroler AT89S52.

## 1.2. RUMUSAN MASALAH

Dalam perancangan dan pembuatan alat secara otomatisasi untuk tampilan yang akan dikontrol langsung oleh *keyboard* PC dengan menggunakan Mikrokontroler AT89S52, dapat dirumuskan permasalahannya sebagai berikut:

1. Bagaimana membentuk angka dan karakter pada *Dot* matriks dan LCD.
2. Bagaimana cara mengirim data dari *keyboard* PC ke *Dot* matriks dan LCD.
3. Bagaimana menguji tiap blok diagram yang dirancang.

## 1.3. TUJUAN

Tujuan yang ingin dicapai oleh penulis dari pembuatan dan penyusunan tugas akhir ini adalah:

1. Mengerti cara kerja dan penggunaan *keyboard* PC.
2. Mengerti cara kerja dan penggunaan LED *Dot* matriks.
3. Merancang LED *Dot* matriks yang dikendalikan dengan Mikrokontroler dapat bekerja dengan baik.
4. Memahami sistem komunikasi antara Mikrokontroler dengan perangkat lain.
5. Untuk mempermudah pemakai khususnya digereja – gereja yang saat ini digunakan sebagai papan penampil nomor halaman dan judul lagu buku nyanyian.

#### 1.4. BATASAN MASALAH

Agar permasalahan tidak terlalu luas, maka penulis membatasi hanya pada hal-hal tertentu:

- a. Keyboard yang digunakan adalah *keyboard PC* standart.
- b. *Dot* Matriks LED Display 5 x 7 (16 character).
- c. Liquid Crystal Display matrix (LCD) 16x2 (32 character).
- d. Catu daya +5 volt sampai +12 volt.
- e. Karakter yang ditampilkan hanya sampai 32 karakter.

#### 1.5. METODELOGI PEMBAHASAN

Adapun metode pembahasan yang digunakan dalam menyusun dan menganalisa tugas skripsi ini:

1. Study literature yang berhubungan dengan perancangan dan pembuatan sistem dan mempelajari serta memahami dasar – dasar Mikrokontroler entah itu *Software* maupun *Hardware*.
3. Perencanaan Sistem  
Merencanakan peralatan yang akan dirancang baik *Software* maupun *Hardware* sesuai dengan perencanaan.
3. Pembuatan Sistem  
Membuat sistem yang akan bekerja sesuai dengan apa yang direncanakan.
4. Pengujian Sistem  
Pengujian sistem yaitu untuk mengetahui sistem kerja dari alat dengan baik secara *Software* maupun *Hardware*.

## **1.6. SISTEMATIKA PENULISAN**

Sistematika penulisan dalam tugas akhir ini terdiri dari 5 bab, yaitu:

### **BAB I : PENDAHULUAN**

Pada bab ini berisi tentang latar belakang permasalahan, batasan masalah, rumusan masalah, tujuan pembahasan, metodologi pembahasan, sistematika penulisan.

### **BAB II : LANDASAN TEORI**

Pada bab ini membahas tentang teori dasar alat-alat yang digunakan dalam perencanaan dan pembuatan sistem yaitu Mikrokontroler, *Hardware* dan teori dasar alat – alat pendukung lainnya.

### **BAB III : PERANCANGAN DAN PEMBUATAN SISTEM**

Pada bab ini membahas tentang perencanaan dan pembuatan penampil nomor dan judul lagu yang berbasis Mikrokontroler terdiri dari *Hardware* dan *Software*.

### **BAB IV : PENGUJIAN SISTEM**

Pada bab ini berisi tentang uji coba alat yang telah dibuat, pengoperasian dan spesifikasi.

### **BAB V : PENUTUP**

Pada bab ini merupakan kesimpulan dari pembahasan pada bab – bab sebelumnya dan kemungkinan untuk pengembangan sistem.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Mikrokontroler**

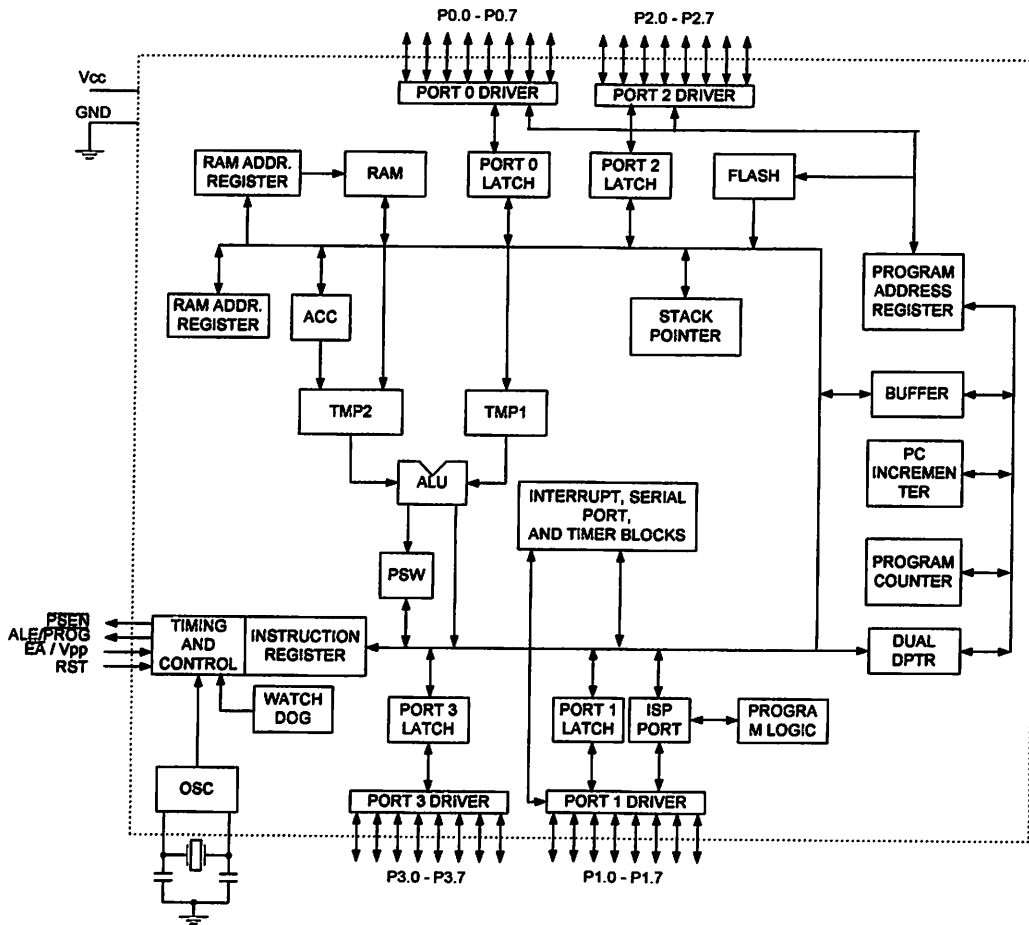
##### **2.1.1. Deskripsi**

Keluarga mikrokontroler MCS-52 merupakan mikrokontroler yang banyak digunakan untuk aplikasi kontrol saat ini. Selain tangguh dan efisien, keluarga ini didukung oleh banyak vendor *Hardware* yang menyediakan banyak fitur tambahan pada sistem mikrokontroler ini.

Sebagai satu vendor yang besar di dunia, Atmel mengeluarkan mikrokontroler AT89S52 yang banyak digunakan pada saat ini. Mikrokontroler AT89S52 memiliki kompatibilitas penuh dengan keluarga mikrokontroler MCS-52. Mikrokontroler AT89S52 memiliki beberapa kemampuan standar, yaitu:

1. Central Processing Unit (CPU) 8 bit dengan register A (acumulator) dan register B.
2. Memiliki 3 level Program *Lock* Memori.
3. 256 x 8 bit Internal RAM.
4. 32 Programmabel I/O lines.
5. 8 Kbyte of In-System Reprogrammable Flash Memory.
6. Memiliki 6 sumber interupsi.
7. 3 *Timer / Counter* 16 bit
8. *Full Duplex serial port* yang dapat diprogram.
9. *On Chip Oscillator*.
10. *Watchdog Timer* dan *Dual data pointer*.

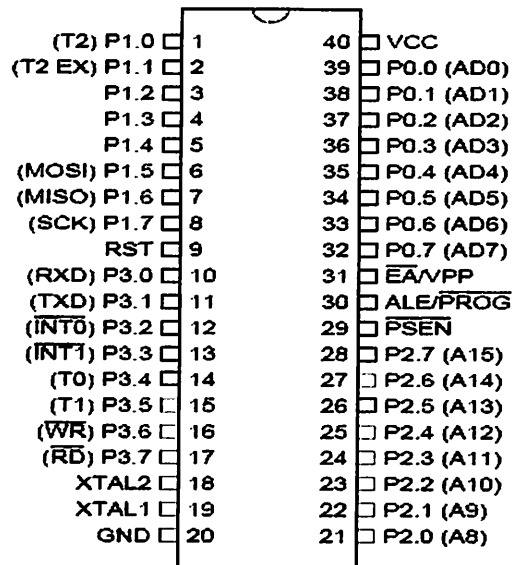




**Gambar 2.1 Diagram Blok Arsitektur Mikrokontroler AT89S52 <sup>[1]</sup>**

### 2.1.2. Konfigurasi Pin AT89S52

Berdasarkan jumlah pin, AT89S52 memiliki 40 pin yang pada dasarnya fungsi pin-pin tersebut hampir sama pada AT89S51 hanya ada sedikit penambahan pada pin P1.0 dan P1.1 pada AT89S52.



**Gambar 2.2 Konfigurasi Pin Mikrokontroler AT89S52 <sup>[1]</sup>**

### 2.1.3. Fungsi Masing-masing Pin AT89S52

#### 1. VCC

Berfungsi sebagai sumber tegangan +5volt.

#### 2. GND

Berfungsi sebagai pentahan (Ground).

#### 3. Port 0

Port 0 adalah masukan/keluaran (I/O) 8 bit dengan nama P0.0 - P0.7 jenisnya cerat terbuka masukan dua arah (open drain bi directional I/O port). Jika port 0 berlogika 1 maka dapat digunakan sebagai masukan yang mempunyai impedansi tinggi.

#### 4. Port 1

Port 1 adalah masukan/keluaran 8 bit dengan nama masing-masing P1.0 – P1.7 yang bersifat dua arah. Port 1 sudah dipasang resistor pull up secara internal. Jika logika

satu dituliskan pada port 1 maka keluaran akan berlogika satu dan digunakan sebagai masukan.

Port 1 sebagai masukan alamat rendah pada saat pemrograman memori flash internal dan verifikasi.

P1.0 untuk T2 (masukan pencacah eksternal ke *timer/counter 2*) keluaran clock.

P1.1 untuk T2EX (*timer/counter 2 reload trigger* dan kendali arah).

## 5. Port 2

Port 2 sama dengan Port 1 yaitu masukan/keluaran 8 bit dengan nama masing-masing P2.0 – P2.7 yang bersifat dua arah. Port 2 juga dipasang resistor pull up secara internal. Jika logika satu dituliskan pada port 2 maka keluaran akan berlogika satu dan digunakan sebagai masukan.

Port 2 juga sebagai byte alamat tinggi (A8 – A15) pada saat menjalankan program pada memori program eksternal dan mengakses data pada memori data eksternal dengan menggunakan pengalamatan 16 bit (instruksi MOVX @DPTR). Sedangkan jika menggunakan pengalamatan 8 bit (instruksi MOVX @R1) maka port 2 berisi SFR P2.

## 6. Port 3

Port 3 sama dengan Port 1 dan Port 2 yaitu masukan/keluaran 8 bit dengan nama masing-masing P2.0 – P2.7 yang bersifat dua arah. Port 2 juga dipasang resistor pull up secara internal. Jika logika satu dituliskan pada port 3 maka keluaran akan berlogika satu dan digunakan sebagai masukan.

Selain sebagai masukan/keluaran biasa, Port 3 juga mempunyai fungsi khusus sebagai berikut :

**Tabel 2.1 Fungsi Khusus Port 3<sup>[1]</sup>**

Port	Fungsi
P3.0	RXD (Port masukan serial)
P3.1	TXD (Port keluaran serial)
P3.2	INT0 (Interupsi eksternal 0, aktif rendah)
P3.3	INT1 (Interupsi eksternal 1, aktif rendah)
P3.4	T0 (Masukan eksternal timer 0)
P3.5	T1 (Masukan eksternal timer 1)
P3.6	WR (Signal tulis untuk memori eksternal aktif rendah)
P3.7	RD (Signal baca untuk memori eksternal aktif rendah)

## 7. RST

RST (*reset*) berfungsi sebagai masukan *reset*. Jika RST diberi logika tinggi dalam waktu 2 siklus mesin mikrokontroler akan direset.

## 8. ALE/PROG

Untuk menahan alamat bawah selama mengakses memori eksternal. Pin ini juga berfungsi sebagai PROG (aktif *low*) yang diaktifkan saat memprogram *internalflash* memori pada mikrokontroler (*on chip*).

## 9. PSEN (*Program Store Enable*)

Sinyal yang digunakan untuk membaca, memindahkan program memori eksternal (ROM / EPROM) ke mikrokontroler (aktif *low*).

## 10. EA/Vpp (*External Access Enable*)

Untuk memilih memori yang akan digunakan, memori program internal ( $EA=V_{cc}$ ) atau memori program eksternal ( $EA=V_{ss}$ ), juga berfungsi sebagai  $V_{pp}$  (*programming supply voltage*) pada saat memprogram internal *flash* memori pada mikrokontroler.

## 11. XTAL-1

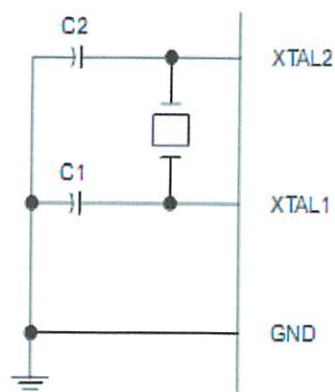
Pin 19 sebagai XTAL 1, masukan ke osilator berpenguatan tinggi, terhubung pada kristal.

## 12. XTAL-2

Pin 18 sebagai XTAL 2, keluaran osilator yang terhubung pada kristal.

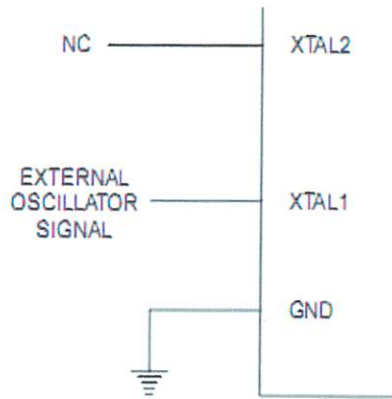
### 2.1.4. Pewaktu CPU (*Central Processing Unit*)

Semua *Mikrokontroler* Atmel mempunyai osilator internal yang digunakan sebagai sumber pewaktu CPU. Penggunaan osilator internal dilakukan dengan menghubungkan kristal atau keramik pada kaki XTAL1 dan XTAL2 seperti pada gambar berikut 2.3.



**Gambar 2.3 Rangkaian Pewaktuan Internal** <sup>[1]</sup>

Atau dapat juga pewaktu diambil dari sumber eksternal dengan koneksinya seperti gambar 2.4.



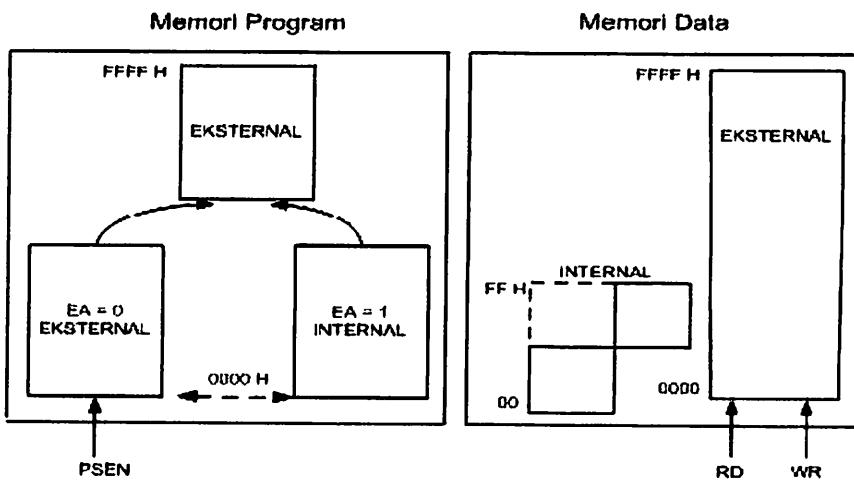
**Gambar 2.4 Masukan Pewaktuan Eksternal<sup>[1]</sup>**

### 2.1.5. Organisasi Memori

#### Organisasi Memori

Semua serpih tunggal dalam keluarga MCS-51 memiliki pembagian ruang alamat untuk program dan data. Pemisahan memori program dan memori data memperbolehkan memori data untuk diakses oleh alamat 8 bit. Sekalipun demikian, alamat data memori 16 bit dapat dihasilkan melalui register DPTR (*Data Point Register*). Memori program hanya bisa dibaca tidak bisa ditulis karena disimpan dalam EPROM. Dalam hal ini EPROM yang tersedia di dalam serpih tunggal AT89S52 sebesar 8 Kbyte Mikrokontroler ATMEL-52 mempunyai organisasi memori yang terdiri atas:

1. Memori Program (CODE).
2. Memori Data (DATA).
3. Memori Data Indirect (IDATA).
4. Memori Data Pengalamatan Bit (BIT).
5. Memori Data External (XDATA).



**Gambar 2.5. Perbedaan Signal Kendali Pada Memori Program dan Memori Data Eksternal [1]**

**a. Memori Program**

Memori program merupakan ruang memori yang digunakan untuk menyimpan kode program dan konstanta yang sifatnya tetap. Memori program bersifat hanya baca saja (*Read Only Memory*) dalam antrian ketika sedang eksekusi program memori ini hanya bersifat dibaca saja tidak dapat diubah isinya.

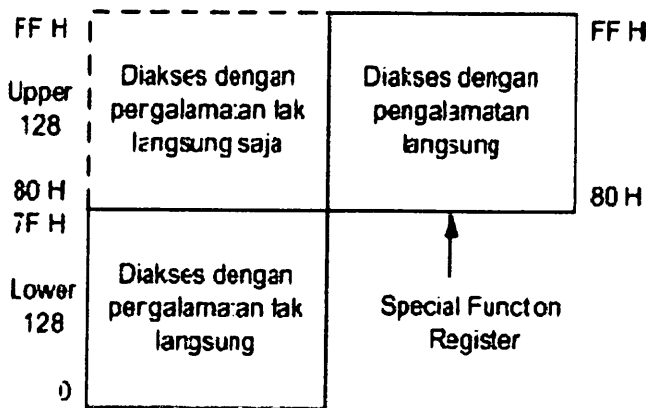
Memori program sebagian terdapat di dalam chip mikrokontroler (*On Chip*) dan sebagian lagi berada diluar *chip* (*Off Chip*). Jika mikrokontroler mempunyai memori program *on-chip* dan diaktifkan maka akan menempati pada alamat awal yang kemudian dilanjutkan



memori program *off-chip*. Kapasitas memori program *on-chip* untuk AT89S52 sebesar 8 kByte.

**b. Memori data**

Memori data internal dipetakan seperti pada gambar di bawah ini ruang memorinya dibagi menjadi tiga blok yaitu bagian 128 bawah, 128 atas, dan ruang SFR (*Special Function Register*)



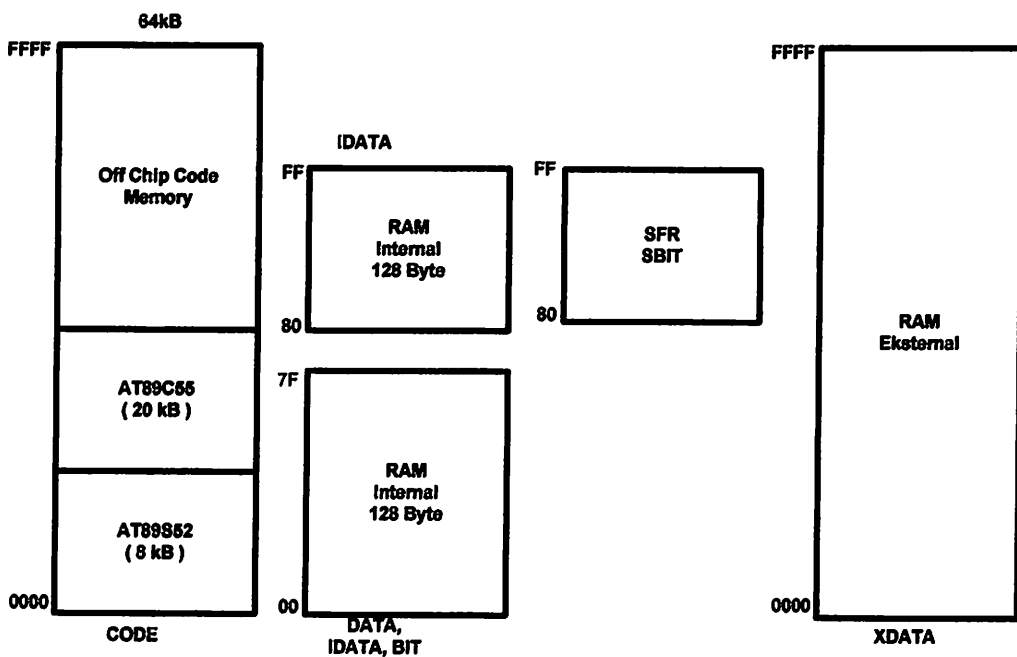
**Gambar 2.6. Memori Data <sup>[1]</sup>**

Bagian RAM 128 byte bawah dipetakan menjadi 32 byte bawah dikelompokkan menjadi 4 bank dan 8 register (R0 sampai R7). Pada bagian 16 byte berikutnya, di atas bank-bank register, membentuk suatu blok ruang memori yang bisa teralamat per bit (*bit addressable*).

Alamat – alamat bit ini adalah 00H hingga 7FH. Semua byte yang berada di dalam 128 bawah dapat diakses baik secara langsung maupun tidak langsung. Bagian 128 atas hanya dapat diakses dengan pengalamatan tidak langsung. Bagian 128 atas dari RAM hanya ada di dalam piranti yang memiliki RAM 256 byte.

**c. Memori Data Indirect (IDATA)**

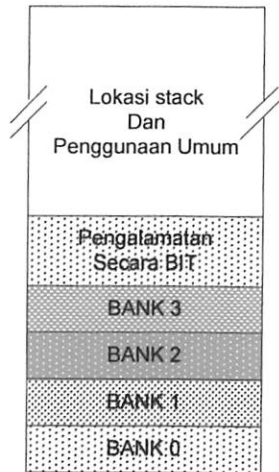
Memori data indirect merupakan segment data seluruh ruang memori data internal yaitu mulai dari alamat 0x00h s/d 0xFFh. 128 byte awal yaitu 00h s/d 7Fh secara fisik sama dengan segment DATA, dan 128 byte diatasnya yaitu mulai alamat 0x80h s/d 0xFF *overloop* dengan *Special Function Register (SFR)*. Untuk mengatasi permasalahan ini dilakukan dengan teknik pengalamatan yaitu *indirect* untuk IDATA dan *direct* untuk SFR. (lihat gambar 2.7 Organisasi Memori AT89S52)



**Gambar 2.7. Organisasi Memori AT89S52<sup>[1]</sup>**

**d. Memori Data Pengalamatan BIT (BIT)**

Secara fisik memori data pengalamatan BIT ini berada pada memori data (DATA) yang dimulai pada alamat 0x20 s/d 0x30 (12 bit). Pada jangkauan alamat ini masing-masing dapat dimanipulasi sendiri – sendiri. Berikut gambar memori Data, 128 awal IDATA dan Pengalamantan BIT.



**Gambar 2.8. Memori Data, 128 awal IDATA dan Pengalamatan BIT<sup>[1]</sup>**

**e. Memori Data Eksternal (XDATA)**

Memori data eksternal (XDATA) adalah ruang memori data *off chip* atau tidak terdapat di dalam chip mikrokontroler. Ruang alamat ini diakses melalui Port 0 (P0) dan Port 2 (P2). Port 0 sebagai bus alamat rendah yang termultipleks dengan bus data sedangkan Port 2 sebagai bus alamat atas. Seluruh jalur alamat sebesar 16 bit sehingga mikrokontroler mampu mengakses memori data sebanyak 64 kbyte.

**f. Memori Data Halaman Eksternal (PDATA)**

Secara fisik PDATA sama dengan XDATA perbedaannya pada PDATA hanya menggunakan Port 0 untuk alamat rendah dan Port 2 tetap terhubung dengan register Port 2 di SFR. Secara intruksi assembly, XDATA menggunakan intruksi `MOVX@DPTR, A` sedangkan PDATA menggunakan `MOVX@R0,A` (atau R1).

**g. Special Function Register (SFR)**

SFR merupakan register dengan fungsi khusus. SFR pada mikrokontroler MCS-52 memiliki alamat 80H – FFH sehingga terdapat 256 lokasi untuk alamat SFR. Dari alamat – alamat ini hanya beberapa saja yang digunakan oleh SFR.

**Tabel 2.2 Special Function Register (SFR) AT89S52 [ 1 ]**

0FBH									0FFH
0F0H	B 00000000								0F7H
0EBH									0EFH
0E0H	ACC 00000000								0E7H
0DBH									0DFH
0D0H	PSW 00000000								0D7H
0CBH	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
0C0H									0C7H
0BBH	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0ABH	IE 0X000000								0AFH
0A0H	P2 11111111								0A7H
9BH	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111								97H
8BH	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DPL 00000000	DPH 00000000				PCON 0XXX0000	87H

## 2.2 EEPROM (*Electrically Erasable Programmable Read Only Memory*)

### 2.2.1. Deskripsi

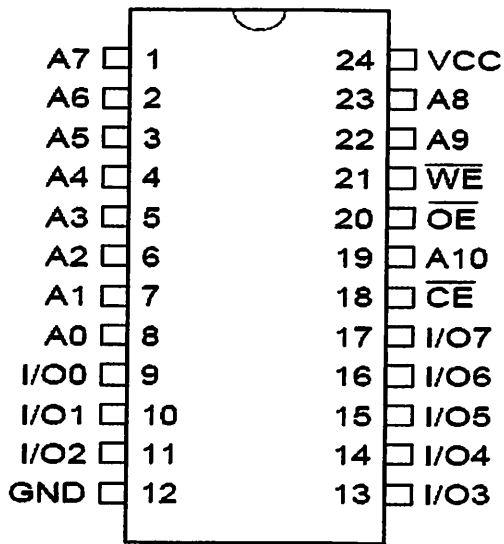
EEPROM merupakan kependekan dari *Electrically Erasable Programmable Read-Only Memory*. EEPROM adalah tipe khusus dari PROM (*Programmable Read-Only Memory*) yang bisa dihapus dengan memakai perintah elektrik. Seperti juga tipe PROM lainnya, EEPROM dapat menyimpan isi datanya, bahkan saat listrik sudah dimatikan.

EEPROM sangat mirip dengan flash memory yang disebut juga flash EEPROM. Perbedaan mendasar antara flash memory dan EEPROM adalah penulisan dan penghapusan EEPROM dilakukan pada data sebesar satu byte, sedangkan pada flash memory penghapusan dan penulisan data ini dilakukan pada data sebesar satu block. Oleh karena itu flash memory lebih cepat.

Dengan ROM biasa, penggantian BIOS hanya dapat dilakukan dengan mengganti chip. Sedangkan pada EEPROM program akan memberikan instruksi kepada pengendali chip supaya memberikan perintah elektronis untuk kemudian mendownload kode BIOS baru untuk dikirimkan kepada *chip*. Hal ini berarti perusahaan dapat dengan mudah mendistribusikan BIOS baru atau *update*, misalnya dengan menggunakan disket. Hal ini disebut juga flash BIOS.

### 2.2.2. Konfigurasi Pin EEPROM AT28C16

Berdasar jumlah pin pada AT28C16 yaitu memiliki 24 pin dan juga memiliki fungsi-fungsi yang berbeda – beda, berikut adalah gambar konfigurasinya.



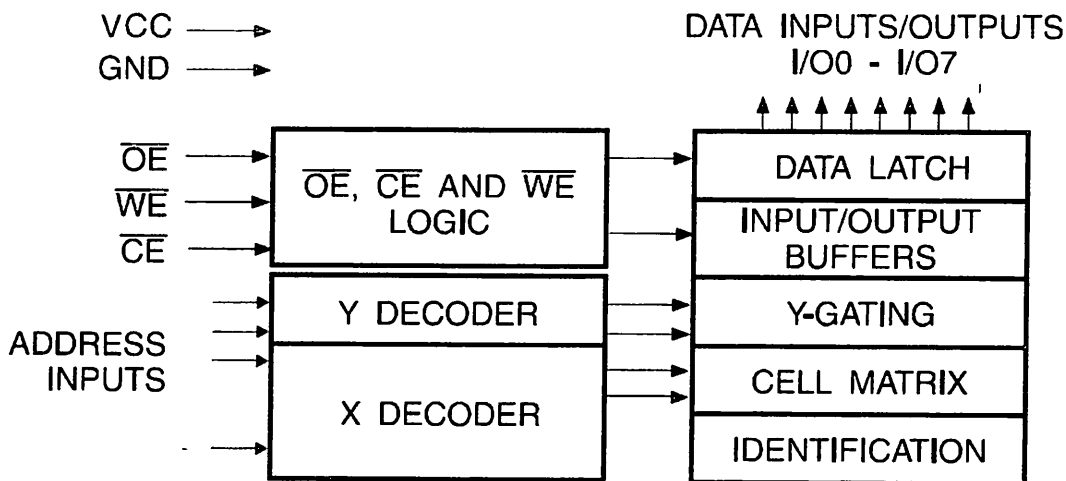
Gambar 2.9. Konfigurasi Pin AT28C16<sup>[1]</sup>

#### Konfigurasi Pin EEPROM AT28C16

- A0 – A10 yaitu sebagai alamat
- $\overline{CE}$  yaitu sebagai Chip Enable
- $\overline{OE}$  yaitu sebagai Output Enable
- $\overline{WE}$  yaitu sebagai Write Enable
- I/O0 – I/O7 yaitu sebagai data masukan atau keluaran
- NC yaitu No Connect
- DC yaitu Don't Connect

**Tabel 2.3. Tabel Konfigurasi Pin EEPROM AT28C16<sup>[1]</sup>**

Pin Name	Function
A0 - A10	Addresses
$\overline{CE}$	Chip Enable
$\overline{OE}$	Output Enable
$\overline{WE}$	Write Enable
I/O0 - I/O7	Data Inputs/Outputs
NC	No Connect
DC	Don't Connect



**Gambar 2.10. Diagram Blok AT28C16<sup>[1]</sup>**

## 2.3 Keyboard PC (*Personal Computer*)

### 2.3.1. Cara Kerja

Setiap kali salah satu tombol *keyboard PC* ditekan atau dilepas, *keyboard PC* akan mengirim kode ke *host* (*host* adalah komputer kalau *keyboard* dihubungkan ke *PC*, atau berupa mikrokontroler kalau *keyboard PC* dihubungkan ke peralatan berbasis mikrokontroler). Kode tersebut dinamakan sebagai *scan code*.



*Scan code* tombol 'S' adalah 1B (angka hex. setara dengan angka biner 00011011). Ketika tombol 'S' ditekan, *keyboard PC* akan mengirim 1B. Jika tombol 'S' ditekan terus, maka *keyboard PC* akan mengirimkan data 1B berikutnya terus menerus, sampai ada tombol lain yang ditekan atau tombol 'S' dilepas. *Keyboard PC* juga mengirim kode saat ada satu tombol yang dilepas kodenya adalah F0 (11110000). Jadi kalau tombol 'S' tadi dilepas, *keyboard PC* akan mengirim data F0 dan 1B.

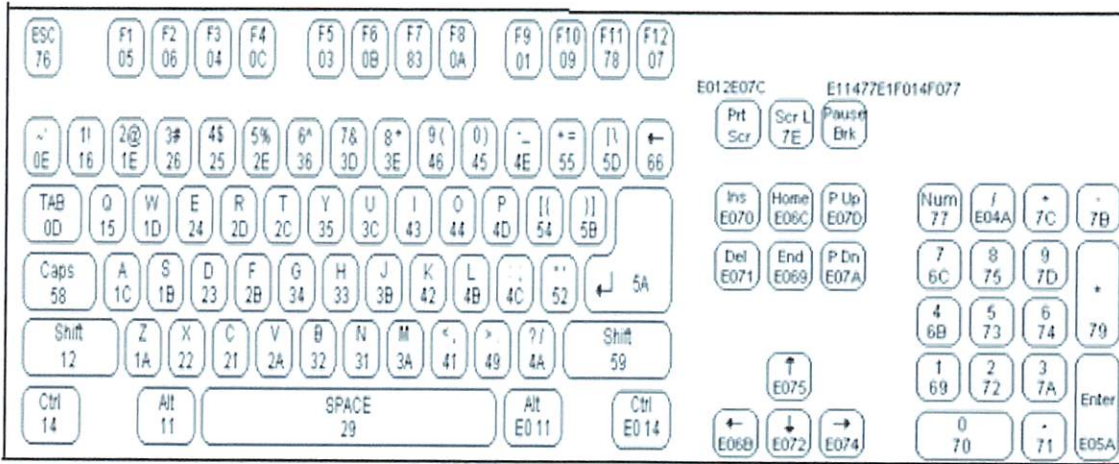
Kode – kode yang akan dikirim ke *keyboard PC* secara seri, artinya dikirimkan satu bit demi satu bit. Misalnya 1B dikirimkan dengan cara : mula-mula dikirim '1', sesaat kemudian '1' lagi dan menyusul '0' sampai akhirnya terkirim sebanyak 8 bit yang berbentuk 00011011 (dikirim mulai dari bit yang paling kanan kemudian bergeser satu persatu sampai yang paling kiri).

Masing – masing tombol pada *keyboard PC* memiliki *scan code* tersendiri, termasuk tombol '*shift*' tombol '*ctrl*' dan lain-lain. Jadi jika tombol '*ctrl*' ditekan bersama dengan tombol 'S' maka *scan* yang dikirim adalah 14 (*scan code* untuk '*ctrl*') dan 1B (*scan code* untuk 'S').

*Scan code* disusun sebagai kode 8 bit bisa dipakai untuk membedakan 256 macam kode, sedangkan *keyboard PC* hanya punya 101 tombol. Jadi sesungguhnya kode 8 bit tadi cukup untuk semua tombol. Tapi tombol di *keyboard PC* dikelompokkan 2 bagian, yaitu bagian utama dan tambahan. Bagian utama cukup dinyatakan dengan *scan code 1 byte* saja, sedangkan bagian tambahan diwakili dengan beberapa *byte scan code* yang selalu diawali dengan E0. Misalnya tombol "*ctrl*" kiri diwakili dengan 14, sedangkan tombol "*ctrl*" kanan diawali dengan E0 14.

Gambar berikut adalah gambar *scan code* masing-masing tombol *keyboard PC*. Terlihat pada gambar tersebut, *scan code* tidak berupa kode ASCII yang biasa dipakai

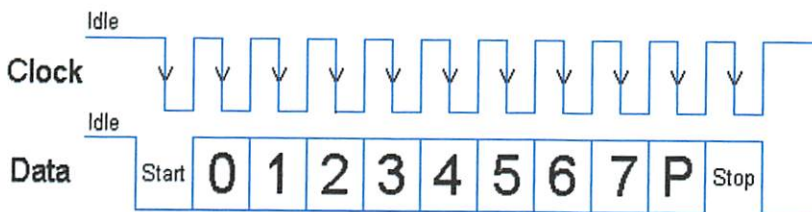
mewakili huruf, dan ditentukan secara acak juga. Sehingga setelah diterima *host*, *scan code* harus diubah menjadi kode ASCII dengan memakai cara pencarian tabel.



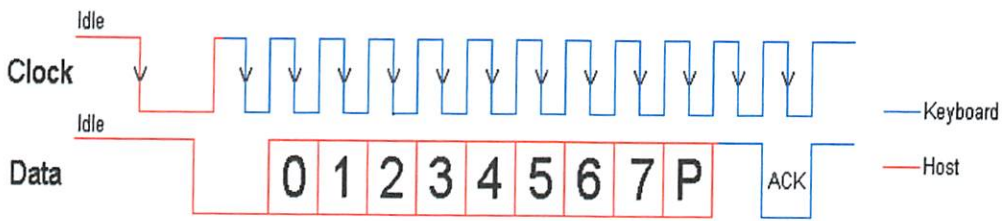
**Gambar 2.11. Keyboard PC dan Scan Code** [ 3 ]

Komunikasi antara *keyboard* dan *host* adalah komunikasi dua arah. *Keyboard PC* mengirim *scan code* ke *host*, *host* bisa mengirim perintah untuk mengatur kerja dari *keyboard PC*.

Urutan pengiriman data dari *keyboard* ke *host* yaitu *start bit*, 8 bit data mulai LSB (*Least Significant Bit*), diikuti dengan *parity* dan diakhiri dengan *stop bit*. Setiap kali data dihasilkan, maka akan di-*clock* antara 10 – 20 KHz, tergantung dari masing – masing *keyboard*. Pada saat tidak ada pengiriman data, sinyal *Kdb Clock* dan *Kdb Data* dalam keadaan ‘1’. Sinyal *Kdb Clock* adalah *clock* yang dihasilkan oleh *keyboard*, sedangkan *Kdb Data* adalah *data* serial yang dihasilkan oleh *keyboard* pula. Sinyal pengiriman data dari *keyboard* ke *host* dan *host* ke *keyboard* akan dijelaskan sebagai gambar berikut :



**Gambar 2.12 a. Sinyal keyboard ke host** [ 3 ]



Gambar 2.12 b. Sinyal *host* ke *keyboard* <sup>[ 3 ]</sup>

Dari gambar diatas dapat dijelaskan bahwa :

- Data mulai dikirimkan dengan meng-nol-kan Kdb *Data* sebagai tanda mulai pengiriman (*start bit*), beberapa saat kemudian setelah Kdb *Data* stabil disusul Kdb *Clock* berubah menjadi '0' dan kembali ke '1' lagi, ini berarti selesai mengirimkan data 1 bit.
- Setelah mengirim '*start bit*', dikirimkan *bit 0*, *bit 1* dan seterusnya sampai *bit 7*.
- Menyusul dikirim *parity bit*, yaitu *bit* kontrol yang berguna bagi penerima data untuk memastikan data yang diterima tidak ada kesalahan. Jika banyaknya *bit* '1' terdapat di *bit* '0' sampai *bit* '7' genap, maka *parity bit* akan bernilai '1'.

Sebagai penutup (*stop bit*) Kdb *Data* dikembalikan pada keadaan normalnya, yaitu '1'.

### 2.3.2. Kode Perintah

Kode perintah untuk *keyboard PC* tidak sebanyak *scan code*, berikut ini daftar kode perintah untuk *keyboard PC* (dalam Heksadesimal) lengkapnya :

**ED** perintah untuk menyalakan/padamkan lampu indikator di *keyboard PC*. Setelah menerima perintah ED dari *host*, *keyboard PC* akan menjawab dengan FA sebagai tanda perintah itu telah dikenali (ACK—acknowledge) dan menunggu 1 byte perintah lagi dari *host* untuk menentukan lampu indikator mana yang perlu dinyalakan/padamkan, 1 *byte* perintah susulan tersebut akan diartikan sebagai

berikut : bit 0 dipakai untuk mengatur lampu indikator *ScrollLock*, bit 1 untuk *NumLock* dan bit 2 untuk *CapsLock*, bit – bit lainnya diabaikan.

**EE** dipakai *host* untuk memeriksa apakah *keyboard PC* masih aktif. Setelah menerima perintah EE dari *host*, *keyboard PC* akan menjawab EE pula, menandakan dirinya masih aktif.

**F0** ada *keyboard PC* yang dilengkapi 3 set *scan code*. Perintah ini dipakai untuk memilih *scan code* yang ingin dipakai. Setelah menerima perintah F0 dari *host*, *keyboard PC* akan menjawab dengan FA sebagai tanda perintah itu telah dikenali (ACK—acknowledge) dan *host* menjawab 1 *byte* lagi (nilainya 1, 2 dan 3) untuk memilih set *scan code*. Jika *byte* yang dikirimkan nilainya 0, *keyboard PC* akan menjawab dengan nomor set *scan code* yang saat itu dipakai.

**F3** dipakai untuk mengatur kecepatan tanggapan *keyboard PC* (*Typematic Repeat Rate*). Setelah menerima perintah F3 dari *host*, *keyboard PC* akan menjawab dengan FA sebagai tanda perintah itu telah dikenali (ACK—acknowledge) dan *host* menjawab 1 *byte* nilai kecepatan tanggapan *keyboard PC* yang dikehendaki.

**F4** dipakai untuk mengaktifkan kembali *keyboard PC*. Setelah menerima perintah F4 ini, *keyboard PC* akan menjawab dengan FA (ACK—acknowledge).

**F5** digunakan untuk menonaktifkan *keyboard PC*. Setelah menerima perintah ini, *keyboard PC* akan menjawab dengan FA (ACK—acknowledge).

**FE** dipakai untuk meminta *keyboard PC* mengirim ulang *scan code* terakhir yang dikirim.

**FF** perintah untuk me-reset *keyboard PC*.

Selain perintah dari *host*, *keyboard PC* juga mempunyai kode – kode lain (selain *scan code* yang dikirim ke *host*), sebagai berikut :

**FA** berarti ACK (acknowledge), yaitu jawaban dari *keyboard PC* bahwa perintah dari *host* sudah dikenali dengan baik.

**AA** berarti *keyboard PC* selesai memeriksa diri dan siap bekerja setelah diberi catu daya.

**EE** dipakai *host* untuk memeriksa apakah *keyboard PC* masih aktif. Setelah menerima perintah EE dari *host*, *keyboard PC* akan menjawab EE pula, menandakan dirinya masih aktif.

**FE** artinya meminta *host* mengulang perintah terakhir yang dikirim.

**FF/00** Berarti terjadi kesalahan di *keyboard PC*.

#### 2.4 LCD (*Liquid Crystal Display*)

Merupakan komponen optoelektronik yaitu komponen yang bekerja atau dipengaruhi oleh sinar (optolistrik), komponen pembangkit cahaya (*Light Emitting*) dan komponen – komponen yang akan mengubah sinar. LCD terbuat dari bahan Kristal cair yang merupakan suatu komponen organik dan mempunyai sifat optik seperti benda padat meskipun bahan tetap cair.

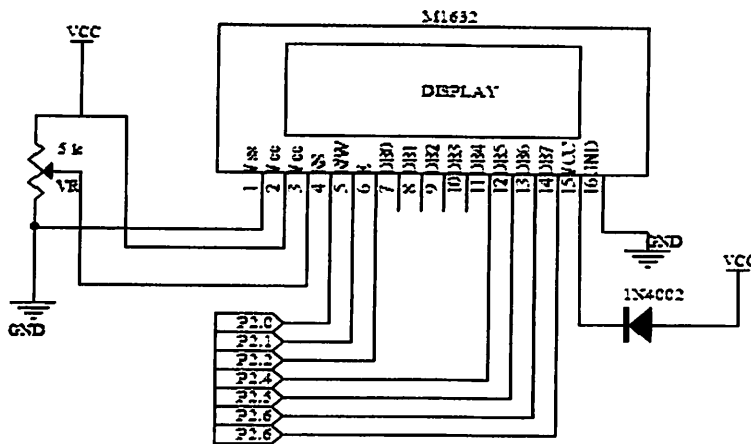
Sel Kristal cair terdiri dari selapis bahan Kristal cair yang diapit antara dua kaca tipis yang transparan. Antara dua lembar kaca tersebut diberi bahan Kristal cair (*liquid crystal*) yang tembus cahaya. Permukaan luar dari masing – masing keping kaca mempunyai lapisan penghantar tembus cahaya seperti oxide timah (*tin oxide*) atau oxide indium (*indium oxide*). Sel mempunyai ketebalan sekitar  $1 \times 10^{-5}$  meter dan diisi dengan *crystal* cair.

Karena sel - sel kristal cair merefleksikan cahaya dan bulan membangkitkan cahaya maka konsumsi daya yang dibutuhkan relatif rendah. Energi yang dipergunakan hanya untuk mengaktifkan kristal cair. Pada dasarnya LCD bekerja pada tegangan rendah

(13 – 15 Vrms), frekuensi rendah ( 25 – 60 Hz) sinyal AC dan memakai arus listrik yang sangat kecil (25 – 300  $\mu$ A). LCD sering kali ditata sebagai tampilan *seven segment* untuk menampilkan angka tetapi juga memiliki keistimewaan lain yaitu, kemampuan untuk menampilkan karakter dan berbagai macam simbol.

Salah satu jenis LCD diantaranya adalah LCD M1632. Suatu jenis piranti dengan konsumsi daya yang rendah, disusun dari *dot* matrik dan dikontrol oleh ROM atau RAM generator karakter dan RAM data display. Pengontrolan utamanya adalah pada ROM generator dan display data RAM yang menghasilkan kode ASCH jika padanya diberikan input ASCH. Untuk dapat difungsikan dengan baik maka perlu diperhatikan proses analisis yang telah ditentukan oleh pabrik pembuatannya. Timing penganalisisan sangat dipertimbangkan, karena jika melesat sampai ordo/*milisecon* maka dapat dipastikan LCD tidak dapat berfungsi.

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



**Gambar 2.13. Rangkaian LCD M1632 [ 2 ]**

Adapun karekteristik dari LCD M1632 antara lain :

- Dengan 16 karakter – 2 baris dalam bentuk *dot* matrik 5 x 7 dan *cursor*
- *Duty ratio* 1/16
- Memiliki ROM pembangkit karakter untuk 192 jenis karakter
- RAM untuk data display sebanyak 80 x 8 *bit*
- Dapat dirangkai dengan MPU 8 bit / 4 *bit*
- RAM data display dan RAM pembangkit karakter dapat dibaca oleh MPU
- Memiliki fungsi intruksi antara lain *display on/off*, *cursor on/off*, *display* karakter *blink*, *cursor shift* dan *display shift*.
- Memiliki rangkaian osilator sendiri
- Catu tegangan tunggal yaitu  $\pm 5V$
- Memiliki rangkaian *reset* otomatis pada catu daya yang dihidupkan.



LCD memiliki 16 pin, masing – masing memiliki fungsi sebagai berikut :

**Tabel 2.3 Fungsi tiap pin LCD <sup>[2]</sup>**

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

Tabel 2.4 Instruksi pada LCD [ 2 ]

Instruksi	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0
Display Clear	0	0	0	0	0	0	0	0	0	1
Cursor Home	0	0	0	0	0	0	0	0	1	*
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S
Display On/Off	0	0	0	0	0	0	1	D	C	B
Cursor Display Shift	0	0	0	0	0	1	S/C	R/L	*	*
Function Set	0	0	0	0	1	DL	1	*	*	*
CG RAM Address Set	0	0	0	1	ACG					
DD RAM Address Set	0	0	1	ADD						
BF/Address Read	0	1	BF	AC						
Data Write to CG RAM	1	0	Write data							
Data Read from CG RAM	1	1	Read data							

ACG : CG RAM *Address* dan ADD

CGRAM merupakan memori untuk menggambarkan pola sebuah karakter, dimana bentuk dari karakter dapat diubah – ubah sesuai dengan keinginan. Namun, memori akan hilang saat *power supply* tidak aktif sehingga pola karakter akan hilang. Berikut tabel pin untuk LCD M1632. Perbedaan-nya dengan LCD standar adalah pada kaki 1 *Vcc*, dan kaki 2 *Ground*. Ini kebalikan dengan LCD standar.

Display	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16						
Line 1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	...
Line 2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	...

Gambar 2.14. Susunan Alamat Pada LCD [ 2 ]

## DD RAM Address

Pada LCD jada terdapat instruksi – instruksi sebagai berikut :

### 1. *Display Clear*

Memberikan tampilan yang ada pada LCD serta menyimpan, sedangkan kursor kembali ke posisi semula.

### 2. *Cursor Home*

Hanya memberikan tampilan dan kursor kembali ke semula

### 3. *Empty Mode Set* : layar beraksi sebagai tampilan tulis

S : 1/10 = menggeser layar

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

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

### 4. *Display On/Off Control*

D : 1 = layar on

D : 0 = layar off

C : 1 = kursor on

C : 0 = kursor off

B : 1 = kursor berkedip – kedip

B : 0 = kursor tidak berkedip – kedip

### 5. *Cursor Display Shift*

S/C : 1 = LCD diidentifikasi sebagai layar

S/C : 0 = LCD diidentifikasi sebagai kursor

R/L : 1 = menggeser satu spasi ke kanan

R/L : 0 = menggeser satu spasi ke kiri

### 6. *Fuction Set*

DL : 1 = panjang data LCD pada 8 bit

DL : 0 = panjang data LCD pada 4 bit

Bit upper di transfer terlebih dahulu kemudian diikuti dengan 4 bit lower

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

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

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

#### 2.4.1. Operasi Dasar

- **Register**

Control dari LCD memiliki 2 buah register 8 *bit* yaitu register intruksi (IR) dan register data (DR). IR memiliki instruksi seperti *display*, *clear*, *cursor shift* dan *display data* (DD RAM) serta karakter (CG RAM). DR menyimpan data untuk ditulis ke DD RAM ataupun membaca data dari DD RAM maka DR secara otomatis menulis data ke DD RAM atau CG RAM. Ketika data pada CG RAM atau DD RAM akan dibaca maka alamat data ditulis pada IR. Sedangkan data akan dimasukkan melalui DR sehingga dapat dibaca oleh mikrokontroler.

**Table 2.5 Pemilihan register pada LCD <sup>[2]</sup>**

RS	RW	Operasi
0	0	Seleksi IR, IR Write Display Clear
0	1	Busy Flag (DB7), @ Counter (DB0 – DB7) Read
1	0	Seleksi DR, DR Write
1	1	Seleksi DR, DR Read

- ***Busy Flag***

Busy Flag menunjukkan bahwa modul siap untuk menerima instruksi selanjutnya sebagaimana terlihat pada tabel diatas. Register seleksi sinyal akan melalui DB7 jika RS=0 dan R/W=1. Jika bernilai 1 maka sedang melakukan kerja internal dan instruksi tidak akan dapat diterima, oleh karena itu status dari flag harus diperiksa sebelum melaksanakan instruksi selanjutnya.

- ***Address Counter (AC)***

*Address Counter* (AC) menunjukkan lokasi memori dalam modul LCD. Pemilihan lokasi alamat lewat AC diberikan lewat register instruksi (IR) ketika data pada A, maka AC secara otomatis menaikkan atau menurunkan alamat tergantung dari *Entry Mode Set*.

- ***Display Data RAM***

Pada LCD, masing – masing line memiliki *range* tersendiri. alamat itu diekspresikan dengan bilangan hexadecimal. Untuk line 1 *range* alamat berkisar antara 40H - 4FH.

- **Character Generator ROM (CG ROM)**

CG ROM memiliki tipe *dot* matrik 5 x 7, dimana pada LCD telah tersedia ROM sebagai pembangkit karakter dalam kode ASCH.

- **Character Generator RAM (CG RAM)**

CG RAM dipakai untuk pembuatan karakter tersendiri melalui program.

Adapun bentuk fisik dari LCD M1632 adalah pada gambar berikut :



**Gambar 2.15. Liquid Crytal Display LCD<sup>[ 2 ]</sup>**

#### **2.4.2. Sinyal interface M1632**

Untuk berhubungan dengan mikrokontroler pemakai, M1632 dilengkapi dengan 8 jalur data (DB0 – DB7) yang dipakai untuk menyalurkan kode ASCH ataupun perintah pengatur kerjanya M1632. Selain itu dilengkapi pula dengan E, R/W, dan RS seperti layaknya komponen yang kompatibel dengan mikroprosesor.

Kombinasi lainnya E dan R/W merupakan sinyal standar pada komponen buatan Motorola. Sebaliknya sinyal – sinyal dari MCS51 merupakan sinyal khas intel dengan kombinasi sinyal WR dan RD.

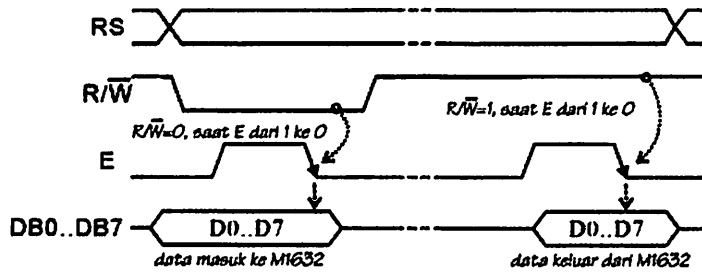
RS, singkatan dari *Register select*, dipakai untuk membedakan jenis data yang dikirim ke M1632, kalau RS=0 data yang dikirim adalah perintah untuk mengatur kerja M1632, sebaliknya kalau RS=1 data yang akan dikirim adalah kode ASCH yang ditampilkan.

HEX	DATA	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0x0000	0	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬	▬
0x0001	1	!	1	À	Q	a	q	À	¡	±	À	Ñ	ã	ñ			
0x0010	16	“	”	2	B	R	b	r	Ж	Г	Ф	З	ã	ö	ä	ó	
0x0011	17	”	#	3	C	S	c	s	З	π	£	Э	ã	ö	ä	ó	
0x00100	16	£	\$	4	D	T	d	t	М	Σ	κ	Р	ã	ö	ä	ó	
0x00101	17	¥	%	5	E	U	e	u	Й	σ	¥	М	ã	ö	ä	ó	
0x00110	16	•	&	6	F	V	f	v	Ј	Д	І	Р	ë	ö	æ	ö	
0x00111	17	€	'	7	G	W	w	Π	τ	§	•	С	×	Ф	÷		
0x001000	16	↑	(	8	H	X	h	x	У	†	‡	ω	é	è	é	è	
0x001001	17	↓	)	9	I	Y	i	y	У	Θ	†	é	ù	é	ù		
0x001010	16	→	*	:	J	Z	j	z	Ч	Ω	∞	∞	é	ù	é	ù	
0x001011	17	←	+	;	K	C	k	c	Ш	∞	*	*	é	ù	é	ù	
0x001100	16	≤	,	<	L	\			Ш	∞	∞	∞	i	ü	i	ü	
0x001101	17	≥	-	=	M	I	m	}	б	∞	∞	∞	i	ÿ	i	ÿ	
0x001110	16	▲	.	>	N	^	n	~	Ы	ε	∞	∞	i	İ	i	İ	
0x001111	17	▼	/	?	O	_	o	o	∞	∞	∞	∞	i	İ	i	ÿ	

Gambar 2.16. Pola Karakter Pada LCD<sup>[2]</sup>

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

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



**Gambar 2.17. Mengirim/Mengambil Data Ke/Dari M1632<sup>[2]</sup>**

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

## 2.5 IC TTL (*Transistor – Transistor Logic*)

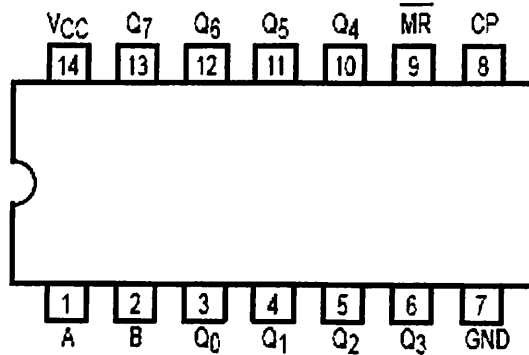
Serkit terpadu atau IC (*Intergrated Circuit*) TTL adalah komponen dasar yang terdiri dari resistor - transistor dan lain - lain. IC adalah komponen yang dipakai sebagai otak peralatan elektronika.

### 2.5.1. IC *Shift Register* 74LS164

Pada perancangan sistem ini untuk *driver Dot* matrik menggunakan IC 74LS164 yaitu sebagai penggerak kolom yang mendapat masukan dari mikrokontroler. 74LS164 yaitu suatu IC yang memiliki kecepatan tinggi 8 bit *Serial Input Parallel Output (SIPO)*

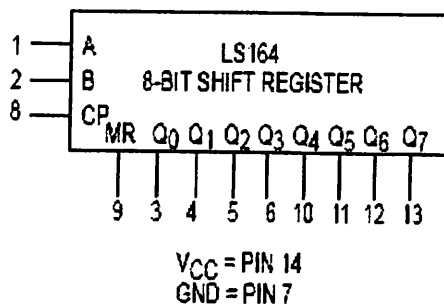


*Shift Register*. Data serial yang masukan melalui dua masukan gerbang AND disinkronisasi dengan tinggi rendahnya suatu peralihan waktu.



**Gambar 2.18. Konfigurasi Pin SN74LS164** <sup>[5]</sup>

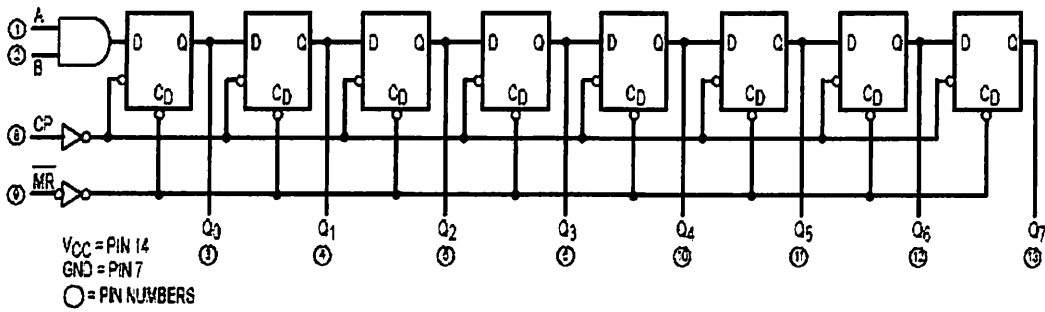
IC 74LS164 adalah IC register geser 8 bit dengan 2 masukan serial dan 8 keluaran paralel. Keluaran IC 74LS164 dapat direset dengan memberikan logika *LOW* pada pin 9 (*Clear*). Jika hanya ingin memberikan satu masukan serial, maka masukan serial yang lain harus diberi logika *HIGH*.



**Gambar 2.19. Logic Symbol** <sup>[5]</sup>

### 2.5.2.1. Konfigurasi Pin 74LS164

- A dan B = Pin A dan B adalah sebagai masukan data.
- CP = Clock (*Active High Going Edge*) Input
- $\overline{MR}$  = Master Reset (*Active Low*) Input
- Q0-Q7 adalah data Keluaran.

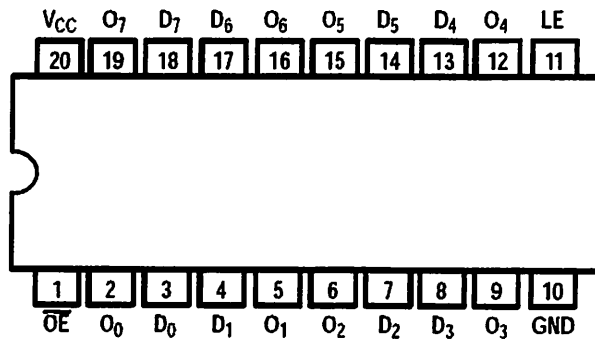


**Gambar 2.20. Diagram Logic 74LS164 [ 5 ]**

Pada digram logic diatas dapat ketahui bahwa masukan pada IC 74LS164 ini adalah pin nomor 1 dan 2 yaitu A dan B sebagai data masukan yang berupa kondisi *High* atau *Low* (1 atau 0). Sedangkan keluaran pada IC 74LS164 adalah Q0-Q7 sebagai data keluaran. Untuk pin nomor 8 sebagai *clock input* (CP) dan MR sebagai *master reset* yaitu mengubah kondisi semua kembali ke kondisi semula atau normal

### 2.5.2. IC D-Flip – flop 74LS373

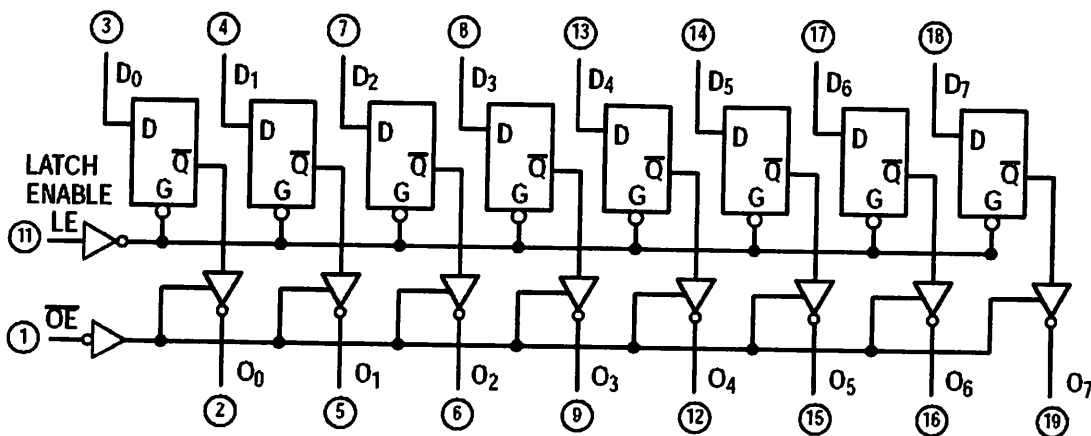
IC 74LS373 adalah sebuah IC yang terdiri atas 8 *latch* dengan 3 keadaan keluaran untuk pengaturan bus sistem operasi. Pada flip – flop ini *data* yang muncul sangatlah jelas ketika Latch Enable dalam keadaan *HIGH*, dan ketika pada keadaan *LOW* maka data tersebut berada pada susunan waktu yaitu latches. Data akan muncul pada bus ketika Output Enable ( $\overline{OE}$ ) pada kondisi *LOW* dan pada saat Output Enable ( $\overline{OE}$ ) ini *HIGH* maka keluaran pada bus dengan keadaan impedansi yang tinggi pula.



Gambar 2.21. Konfigurasi Pin 74LS373 [ 4 ]

### 2.5.2.1. Konfigurasi Pin 74LS373

- D0 – D7 adalah alamat data masukan
- O0 – O7 adalah alamat data keluaran
- $\overline{OE}$  adalah Output Enable
- LE adalah Latch Enable
- VCC yaitu pada pin 20
- GND yaitu pada pin 10



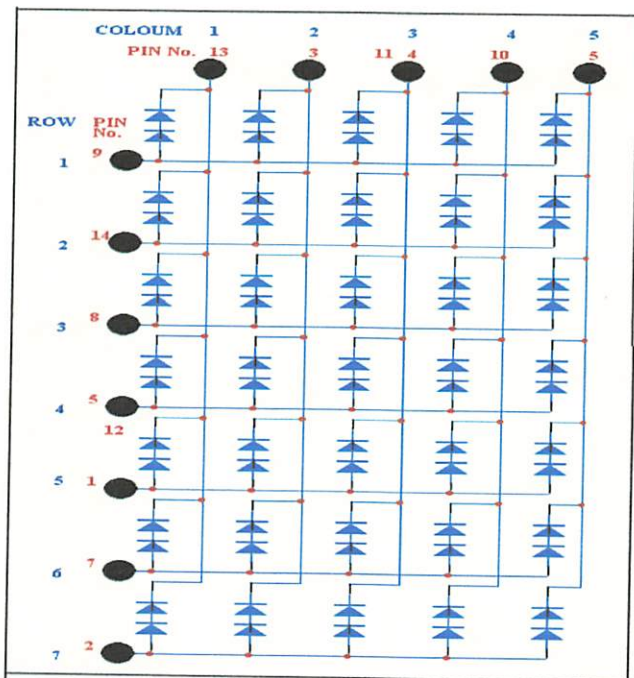
Gambar 2.22. Logic Diagram 74LS373 [ 4 ]

## 2.6 DOT MATRIKS LED

### 2.6.1. Dot Matriks LED

Pada umumnya *dot* Matriks di definisikan sebagai suatu susunan *data* dalam bentuk baris dan kolom, dan *data – data* yang ada pada *dot* matriks tersebut saling berkaitan satu dengan yang lainnya.

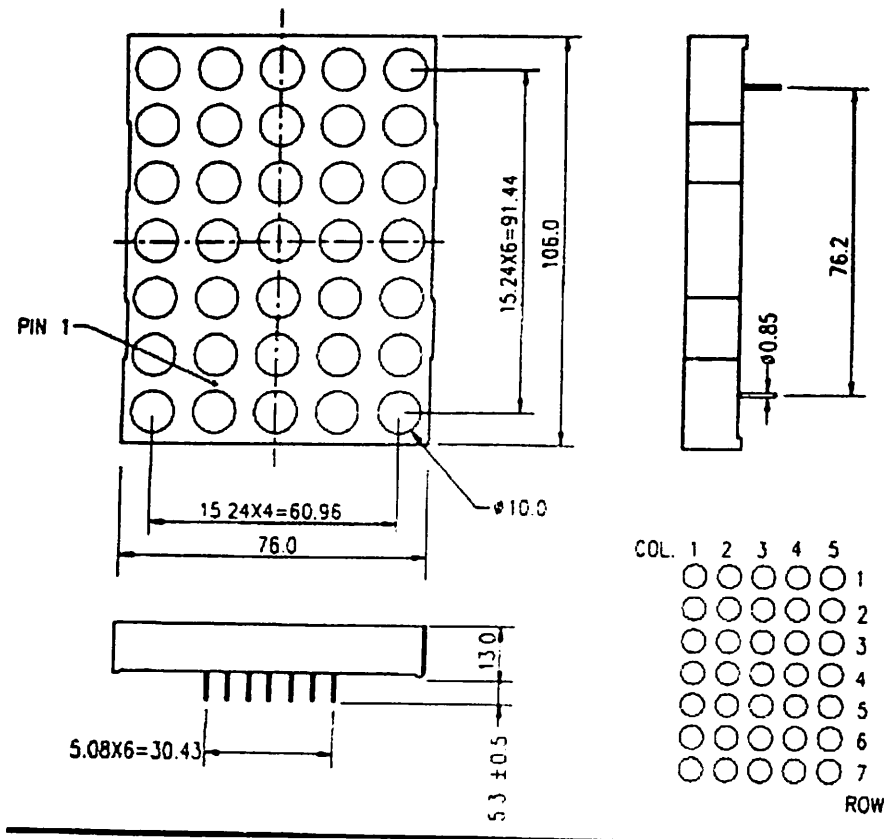
Piranti *dot* matriks mempunyai sejumlah besar sumber cahaya yang berbentuk seperti titik. Contoh umum adalah matriks 5 x 7 yang terlihat pada gambar 2.23 berikut. Untuk menghidupkan sebuah LED dalam matriks ini, harus menarapkan tegangan dalam anodanya dan menggroundkan lebih dari satu baris, maka dapat memperagakan setiap angka decimal, setiap huruf abjad, serta berbagai karakter lain.



Gambar 2.23. Internal Circuit Diagram Dot Matriks 5x7 [ 8 ]

Tampilan matriks titik (*Dot Matrix*) terdiri atas sejumlah LED yang disusun dalam baris dan kolom. Susunan yang paling sering digunakan adalah 5 x 7, yaitu lima kolom

dan tujuh baris yang jumlah titik – titiknya adalah 35 titik, seperti pada gambar 2.24 berikut.



**Gambar 2.24. Package Dimension Dot matriks 5x7 (4")<sup>[8]</sup>**

Matriks 5 x 7 dapat digunakan untuk menyajikan karakter alfanumerik yang lengkap. Proses pembangkitan karakter antara lain melibatkan proses scanning baris atau kolom, memilih LED yang tepat dalam baris atau kolom dan menyalakannya. Proses yang ini diulang untuk baris atau kolom berikutnya. Setelah semua baris atau kolom dipilih dengan urutan tertentu, proses diatas diulang mulai dari baris paling atas atau kolom pertama.

Jika frekuensi scanningnya cukup cepat (sekitar 50Hz), akan diperoleh karakter bebas kedip. Jika matriks discan dari kiri ke kanan, kolom demi kolom disebut *Vertical scanning*, dan jika dilakukan baris demi baris disebut *Horisontal scanning*.

### 2.6.2. Dekode Matriks LED

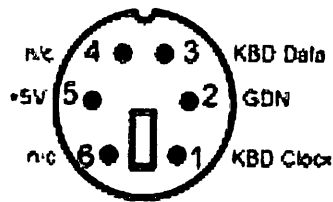
Untuk menyalakan suatu huruf alfanumerik dengan pembacaan *dot* matriks, maka LED yang dibutuhkan tidak dinyalakan dengan serentak atau bersamaan, proses ini dilakukannya yaitu dengan proses scanning pada baris dan kolom sehingga dapat membentuk suatu huruf alfanumerik atau karakter.

Rangkaian yang mendriver matriks LED ini cukup rumit, karena harus melakukan *scan* baris – baris horisontal dan pada saat yang sama memasang tegangan pada kolom yang sesuai. Secara umum pencacah lingkaran (*Ring Counter*) melakukan *scan* dalam baris horisontal, sedangkan memori berisi data kolom LED yang menyala memberikan tegangan dalam kolom – kolom vertikal.

### 2.7 Konektor PS2

Konektor PS2 (PS *two*) adalah sebuah masukan yang sering digunakan pada komputer. Pada umumnya konektor *keyboard* ada beberapa macam, namun digambar berikut adalah salah satu konektor *keyboard* yaitu konektor PS2 (PS *two*) yang memiliki 6 pin.

#### 2.7.1. Konfigurasi Pin Konektor PS2



Gambar 2.25. Konektro PS2 [ 3 ]

1= KBD Clock

4= No Connention (N/C)

2= Ground

5= Vcc (+ 5 Volt)

3= KBD Data

6=NoConnection(N/C)

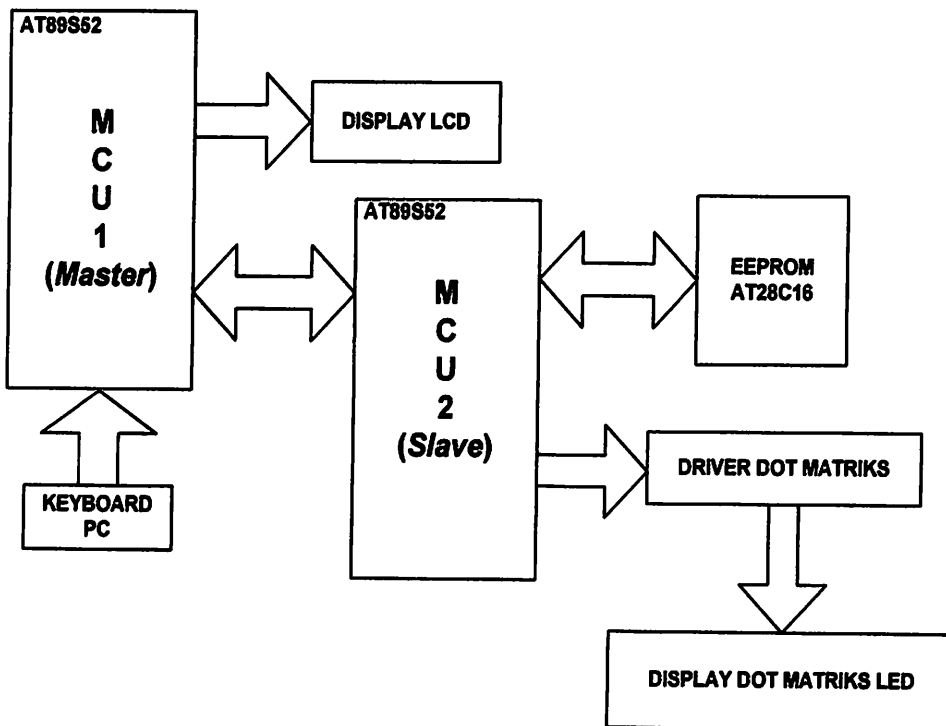
## BAB III

### PERANCANGAN DAN PEMBUATAN SISTEM

Pada bab ini akan di bahas tentang perancangan dan pembuatan sistem yang meliputi perancangan *software* (perangkat lunak) dan *hardware* (perangkat keras) untuk lebih jelas maka akan dibahas pada sub – sub bab berikut:

#### 3.1. Diagram Blok Rangkaian

Secara garis besar, prinsip kerja dari alat ini digambarkan diagram bloknya sebagai berikut:



Gambar 3.1 Diagram Blok Sistem

Prinsip kerja dari masing-masing blok:

➤ Mikrokontroler AT89S52 *Master*

Mikorkontroler *Master* ini adalah sebagai otak pengolah data dari masukan *keyboard PC* yang akan dikirim ke LCD dan mikrokontroler *Slave*.

➤ Mikrokontroler AT89S52 *Slave*

Mikrokontroler *Slave* ini adalah sebagai penyimpan data sementara yang akan dikirim ke display *dot* matriks.

➤ *Keyboard PC*

*Keyboard PC* bekerja sebagai data masukan yang akan dikirim ke LCD dan *Display dot* matriks, data tersebut berupa karakter – karakter dan huruf abjad.

➤ Display LCD

*Display Liquid Crystal Display* (LCD) sebagai hasil tampilan sementara dari masukan *Keyboard PC*.

➤ Memori EEPROM AT28C16

*Electrically Erasable Programmable Read-Only Memory* (EEPROM) sebagai penyimpan data sementara yang akan dikirim ke display *Dot* matriks.

➤ Driver Dot matriks

Driver *dot* matriks yaitu rangkaian yang berfungsi untuk menggerakkan display *dot* matriks yaitu kolom dan baris dan memberikan tegangan pada display *dot* matriks.

➤ Display *Dot* matriks LED

Hasil tampilan akhir dari masukan-masukan yang diberikan yaitu berupa karakter atau huruf abjad.

### 3.2. Perencanaan Perangkat Keras (*Hardware*)

Dalam perencanaan ini, rancangan *hardware* yang dibuat bertujuan guna mendukung dan memberikan kemudahan pada proses kerja perancangan *software* agar



nantinya sesuai dengan kondisi yang diinginkan. Untuk perancangan *hardware* sendiri dibagi menjadi 6 bagian yaitu :

- Minimum Sistem AT89S52
- Minimum Sistem EEPROM AT28C16
- Driver *Dot* matriks
- *Keyboard PC*
- LCD (*Liquid Crystal Display*)
- Display *Dot* matriks

### **3.2.1. Perancangan Minimum Sistem AT89S52**

Penggunaan mikrokontroler AT89S52 harus didukung oleh beberapa rangkaian penunjang agar dapat melakukan fungsinya, antara lain rangkaian *clock* dan rangkaian *reset*. Selain itu juga harus ditentukan penggunaan port – portnya untuk rangkaian pendukung yang lain.

#### **3.2.1.1. Rangkaian *Clock***

Kecepatan proses pengolahan data pada mikrokontroler ditentukan oleh *clock* (waktu) yang dikendalikan oleh mikrokontroler tersebut. Pada mikrokontroler AT89S52 terdapat *internal clock generator* yang berfungsi sebagai sumber *clock*, tapi masih memerlukan rangkaian tambahan untuk membangkitkan *clock* yang diinginkan.

Rangkaian tambahan ini terdiri atas 2 buah kapasitor dan sebuah kristal yang terangkai sedemikian rupa dan kemudian dihubungkan dengan port yang khusus tersedia pada mikrokontroler.

Dalam perancangan rangkaian ini menggunakan :

- 2 Kapasitor 33 pF. Penentuan besarnya kapasitansi disesuaikan dengan spesifikasi pada data sheet.
- Kristal 12 MHz

Dengan demikian perhitungannya dapat dilihat sebagai berikut :

$$f = 12 \text{ MHz}$$

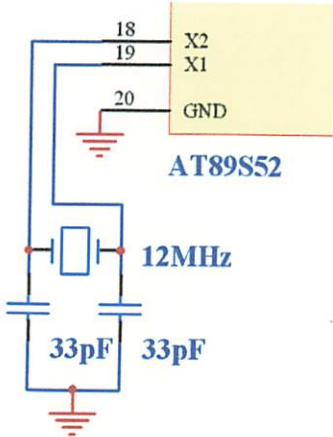
$$T = \frac{1}{f}$$

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

karena 1 siklus mesin = 12T maka

$$1 \text{ siklus mesin} = 12 \times \frac{1}{12 \times 10^6} = 0,999 \mu\text{s}$$

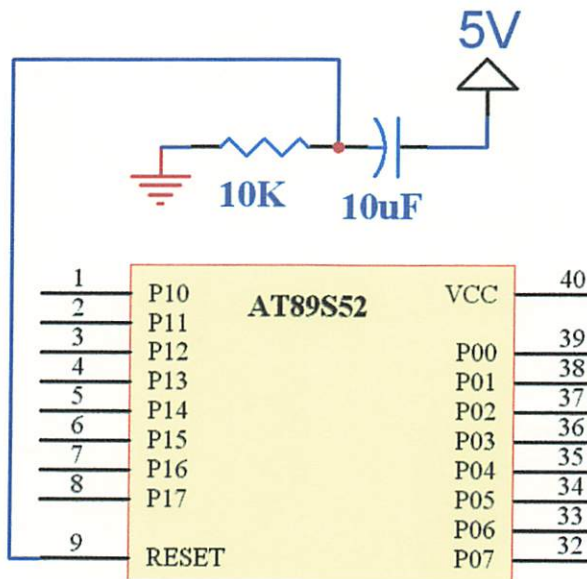
Adapun gambar rangkaian *clock* sebagai berikut :



Gambar 3.2 Rangkaian *Clock*

**3.2.1.2. Rangkaian *Reset***

*Reset* pada mikrokontroler merupakan masukan aktif *high* '1' pulsa transisi dari rendah '0' ketinggian akan me-*reset* mikrokontroler menuju alamat 0000H. Pin *reset* dihubungkan dengan rangkaian *power-on reset* seperti pada gambar 3-3:



**Gambar 3.3 Rangkaian Reset**

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

Rangkaian *reset* terbentuk oleh komponen resistor dan kapasitor yang sudah baku (ditetapkan oleh perusahaan pembuat IC AT89S52). Nilai resistor yang dipakai adalah  $10K\Omega$  dan kapasitor  $10\mu F$ . Karena kristal yang digunakan mempunyai frekuensi sebesar 12 MHz, maka satu periode membutuhkan waktu sebesar :

$$T = \frac{1}{f_{XTAL}} = \frac{1}{12MHz} S = 8,333 \times 10^{-8}$$

Sehingga waktu minimal logika yang dibutuhkan untuk me-*reset* mikrokontroler adalah :

$$Reset \text{ (minimal)} = T \times \text{periode yang dibutuhkan}$$

$$= 8,333 \times 10^{-8} \times 12 = 1,999 \mu\text{s}$$

Jadi mikrokontroller membutuhkan waktu minimal 1,999  $\mu\text{s}$  untuk *me-reset*.

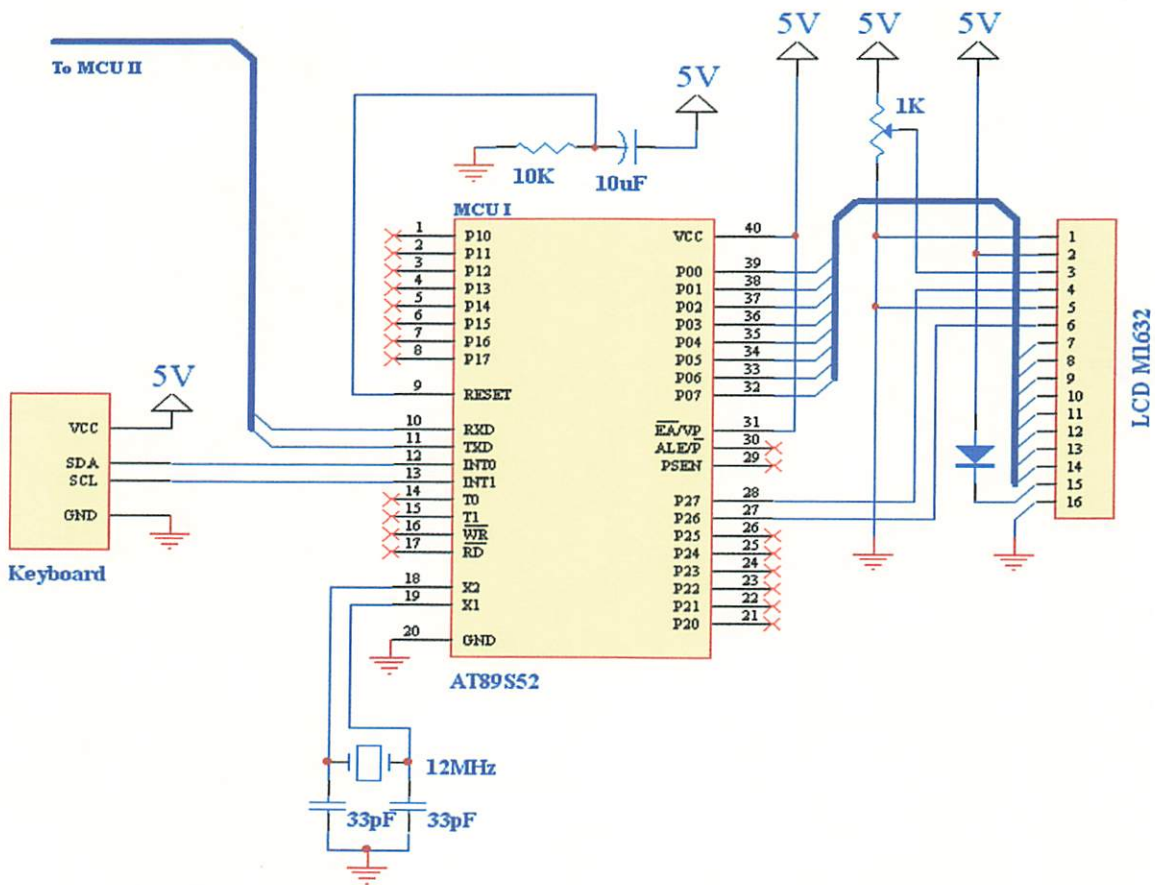
Waktu inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dengan menentukan nilai R = 10 K $\Omega$  dan C = 10  $\mu\text{F}$ , maka :

$$T = 0.357 R.C = 0,357 \times 10000 \Omega \times 10 \cdot 10^{-6} = 35,7 \text{ ms}$$

Jadi dengan nilai komponen R = 10 K $\Omega$  dan C = 10  $\mu\text{F}$  dapat memenuhi syarat minimal untuk waktu yang dibutuhkan mikrokontroler

### **3.2.1.3. Perancangan Penggunaan Port – Port Pada Mikrokontroler AT89S52**

Pada skripsi ini mikrokontroller AT89S52 digunakan sebagai pusat pengendali kerja dari alat yang dibuat Gambar 3-4 menunjukkan rancangan port – port I/O pada mikrokontroler AT89S52 yang dimanfaatkan pada skripsi.



Gambar 3.4 Perancangan mikrokontroler I (Master)

1. Port 0

Port 0.0 – Port 0.7 (pin 32 – 39) digunakan sebagai port keluaran untuk *Liquid Crystal Display (LCD)*

2. Port 1

Port 1.0 – Port 1.7 (pin 1 – 8) pada perancangan ini untuk port 1 tidak digunakan.

3. Port 2

Pin 27 dihubungkan ke pin 6 pada *LCD (Liquid Crystal Display)*

Pin 28 dihubungkan ke pin 4 pada *LCD (Liquid Crystal Display)*

4. Port 3

Pin 10 yaitu RXD dihubungkan ke pin 11 TXD *Micro II (Slave)*

Pin 11 yaitu TXD dihubungkan ke pin 10 RXD *Micro II (Slave)*

5. Pin 9 (*reset*) dihubungkan dengan rangkaian *reset*

6. Pin 18 dan pin 19 dihubungkan dengan rangkaian rangkaian *clock* atau

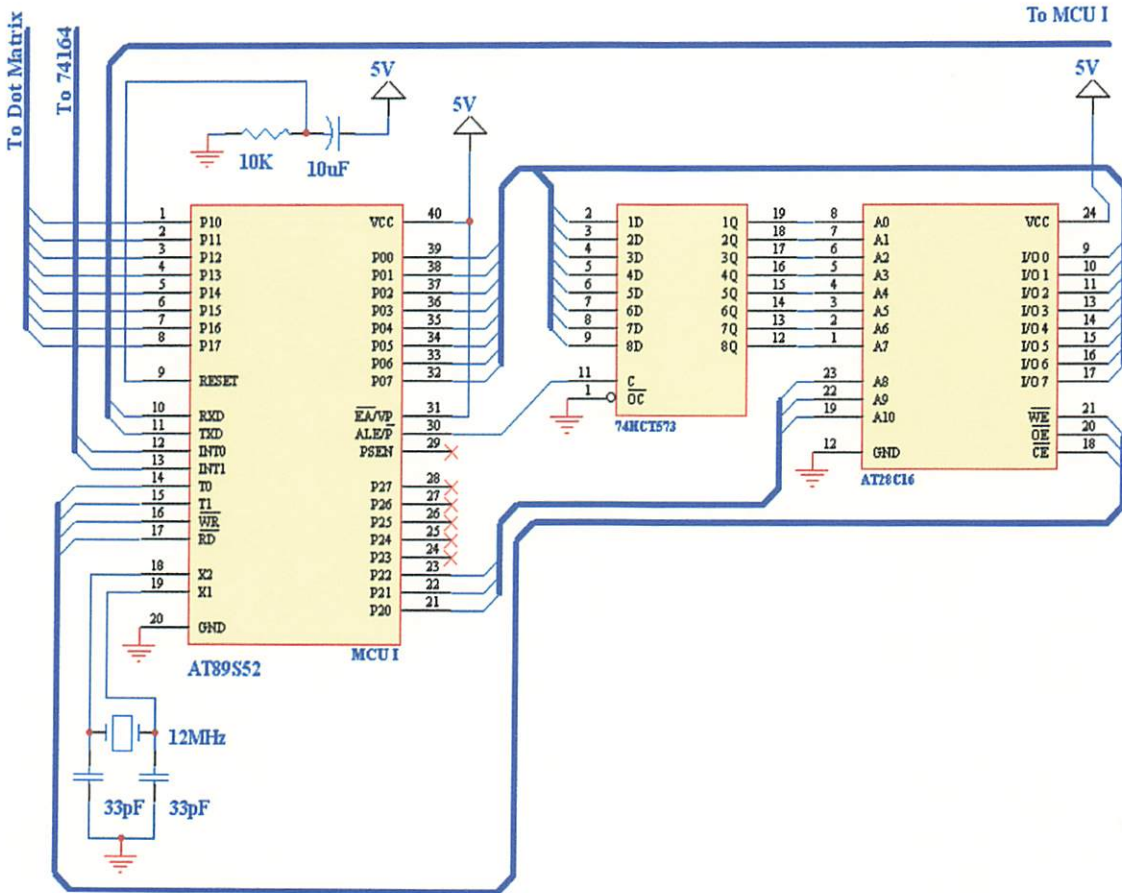
*Oscillator external*

7. Pin 31 (EA) diberi logika tinggi (*high*) atau dihubungkan dengan Vcc maka mikrokontroller akan mengakses program dari ROM internal (EPROM atau *flash memory*)

8. Pin 29 dan pin 30 (ALE/PROG dan PSEN) tidak digunakan karena pada pembuatan alat ini tidak menggunakan atau mengakses *memory eksternal*

9. Pin 40 (Vcc) dihubungkan dengan tegangan *supply +5V*

10. Pin 20 (GND) dihubungkan dengan tegangan *supply ground*



Gambar 3.5 Perancangan mikrokontroler II (*Slave*)

1. Port 0  
 Port 0.0 – Port 0.7 (pin 39 – 32) digunakan sebagai *outputan* data yang dikirim ke IC 74LS373 dan EEPROM AT28C16
2. Port 1  
 Port 1.0 – Port 1.4 (pin 1 – 5) digunakan sebagai *outputan driver* dot matrik
3. Port 2  
 Pin 21 dihubungkan ke pin 19 dari EEPROM AT28C16  
 Pin 22 dihubungkan ke pin 22 dari EEPROM AT28C16  
 Pin 23 dihubungkan ke pin 23 dari EEPROM AT28C16

4. Port 3

Pin 10 yaitu RXD dihubungkan ke pin 11 TXD *Micro I (Master)*

Pin 11 yaitu TXD dihubungkan ke pin 10 RXD *Micro I (Master)*

Pin 12 yaitu INT0 dihubungkan ke pin 8 *Clock 74LS164*

Pin 13 yaitu INT1 dihubungkan ke pin 1 dan 2 dari 74LS164

Pin 14 – pin 17 dihubungkan ke EEPROM AT28C16

5. Pin 9 (*reset*) dihubungkan dengan rangkaian *reset*

6. Pin 18 dan pin 19 dihubungkan dengan rangkaian rangkaian *clock* atau *Oscillator external*

7. Pin 31 (EA) diberi logika tinggi (*high*) atau dihubungkan dengan Vcc maka mikrokontroler akan mengakses program dari ROM internal (EPROM atau *flash memory*)

8. Pin 29 dan pin 30 (ALE/PROG dan PSEN) tidak digunakan karena pada pembuatan alat ini tidak menggunakan atau mengakses *memory eksternal*

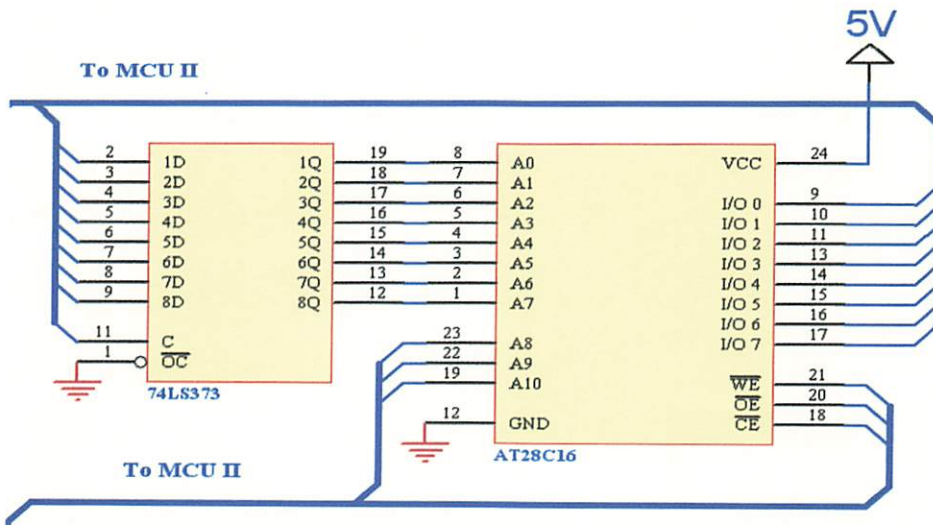
9. Pin 40 (Vcc) dihubungkan dengan tegangan *supply +5V*

10. Pin 20 (GND) dihubungkan dengan tegangan *supply ground*

### 3.2.2. Perancangan Minimum Sistem EEPROM AT28C16

Pada perancangan sistem ini digunakan *Electrically Erasable Programmable Read Only Memory* (EEPROM) AT28C16 sebagai memori tambahan, disini EEPROM berfungsi sebagai penyimpanan sementara data atau karakter yang akan dituliskan pada *dot matriks*.



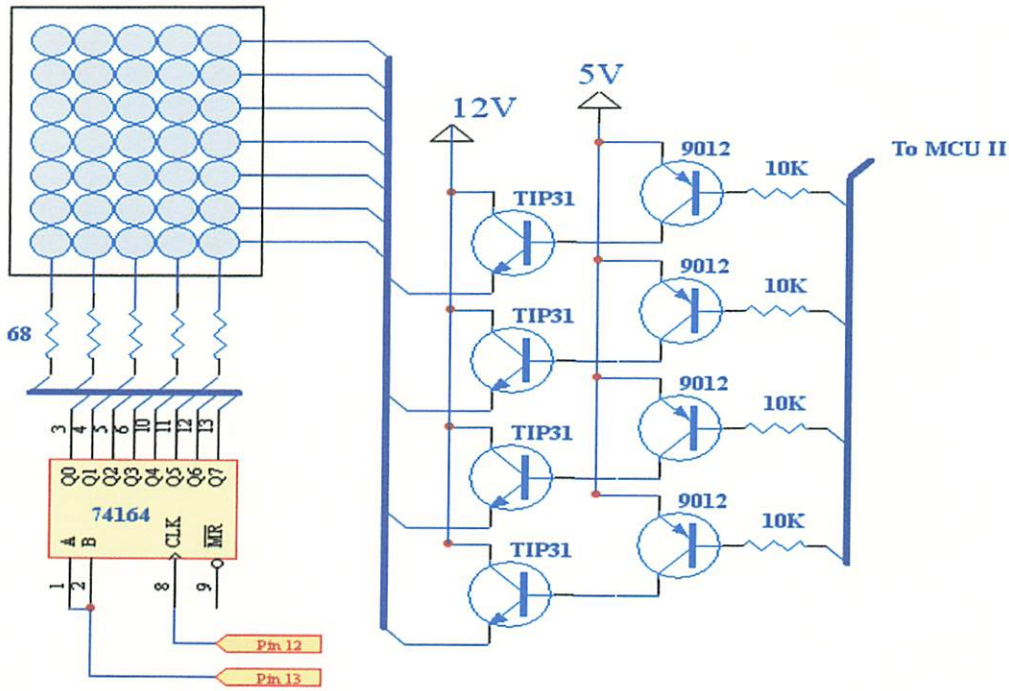


Gambar 3.6 Perancangan AT28C16

- Pin 1 – pin 8 (A7-A0) yaitu sebagai alamat data yang dihubungkan ke keluaran dari IC 74LS373 (pin 12-pin18).
- Pin 9 – pin 17 (I/O0-I/O7) yaitu sebagai *input/output* yang dihubungkan ke port 0 dari mikrokontroler II (Slave) dan ke pin 2 – pin 9 pada IC 74LS373.
- Pin 19, 22 dan 23 yaitu (A10, A9 dan A8) dihubungkan ke port 2 mikrokontroler II (Slave) yaitu pada pin 21, 22 dan 23.

### 3.2.3. Driver Dot matriks

Pada perancangan driver *dot* matrik yaitu digunakan IC register 74LS164 sebagai pengeser data kolom pada *dot* matrik dan transistor TIP31 sebagai pembangkit tegangan atau saklar pada driver *Dot* matriks, berikut adalah gambar perancangan untuk *driver Dot* matrik.



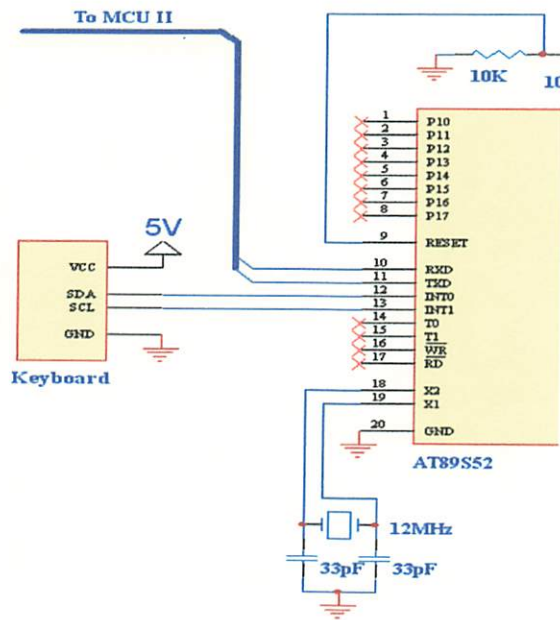
**Gamabar 3.7 Perancangan Rangkaian Driver Dot Matrik**

Konfigurasi pin – pin dari perancangan driver *dot* matrik adalah:

- Pin 1 dan pin 2 adalah inputan data yaitu dihubungkan dengan pin 13 (INT1) dari mikrokontroler II (*Slave*).
- Pin 8 sebagai *clock* yang dihubungkan dengan pin 12 (INT0) dari mikrokontroler II (*Slave*).
- Pin 3, 4, 5, 6, 10, 11, 12 dan 13 sebagai keluaran data yang dihubungkan dengan *dot* matrik yang pada baris – barisnya. Disini pin 13 juga sebagai masukan ke IC 74LS164 yang lain yaitu dihubungkan ke pin 1 dan 2 IC berikutnya.

### 3.2.4. Perancangan Rangkaian *Keyboard PC*

Pada perancangan sistem ini digunakan *keyboard PC* sebagai masukan data, yaitu berupa huruf dan angka atau simbo – simbol yang akan ditampilkan pada LCD dan *dot* matrik, berikut ini adalah gambar perancangan untuk *keyboard PC*



Gambar 3.8 Perancangan Rangkaian pada *Keyboard PC*

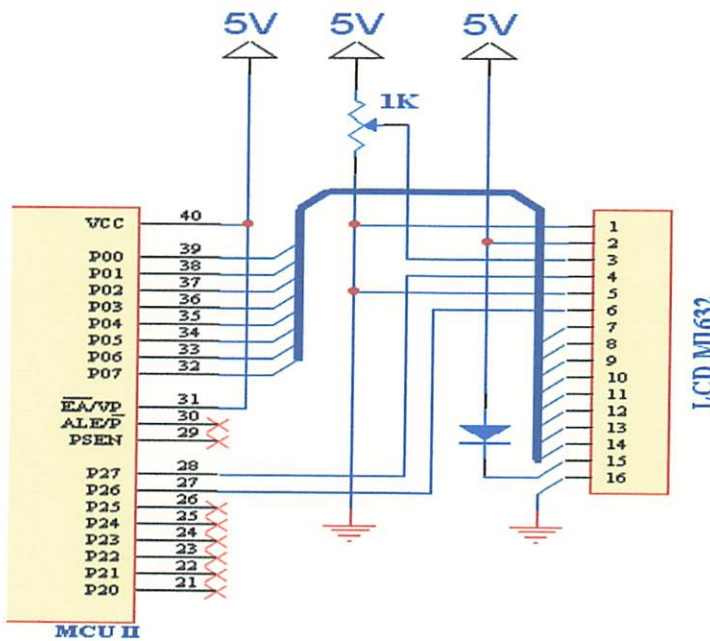
Konfigurasi pin – pin pada *Keyboard PC* yaitu sebagai berikut:

- Pin 5 sebagai *clock* yang dihubungkan dengan pin 13 (INT1) dari mikrokontroler I (*Master*).
- Pin 4 sebagai Vcc yaitu menerima tegangan sumber sebesar +5 Volt DC.
- Pin 3 sebagai *Graund*.
- Pin 1 sebagai Data yaitu dihubungkan dengan pin 12 (INT0) dari mikrokontroler I (*Master*).
- Pin 2 dan pin 6 yaitu *No Connection* (NC) tidak digunakan.

### 3.2.5. Perancangan Rangkaian LCD (*Liquid Crystal Display*)

*Liquid Crystal Display* (LCD) sebagai penampil hasil ketikan yang akan dikirim ke mikrokontroler II (*Slave*) sehingga di tampilkan ke *dot* metrik *display*.

Pada perancangan ini menggunakan LCD M1632/162A sebagai penampilnya, berikut adalah gambar perancangan rangkaian *keyboard PC*.



Gambar 3.9 Perancangan Rangkaian LCD

Konfigurasi pin – pin pada *Liquid Crystal Display* (LCD) sebagai berikut:

- Pin 1, 3 dan 5 dihubungkan dengan trinpot 1K untuk pengaturan kecerahan pada *Liquid Crystal Display* (LCD).
- Pin 2 sebagai Vcc dihubungkan ke tegangan sumber +5 Volt DC.
- Pin 4 sebagai *Sinyal Register* (SR) yang dihubungkan dengan pin 28 pada mikrokontroler I (*Master*).

- Pin 6 sebagai *Enable* (E) yang dihubungkan dengan pin 28 pada mikrokontroler I (*Master*).
- Pin 7 – 14 sebagai *bit bus* data *Lower* dan *Upper* dua arah yang dihubungkan dengan port0 pada mikrokontroler I (*Master*).
- Pin 15 sebagai kecerahan pada LCD yang dihubungkan dengan kaki katoda dioda.
- Pin 16 sebagai *Ground*.

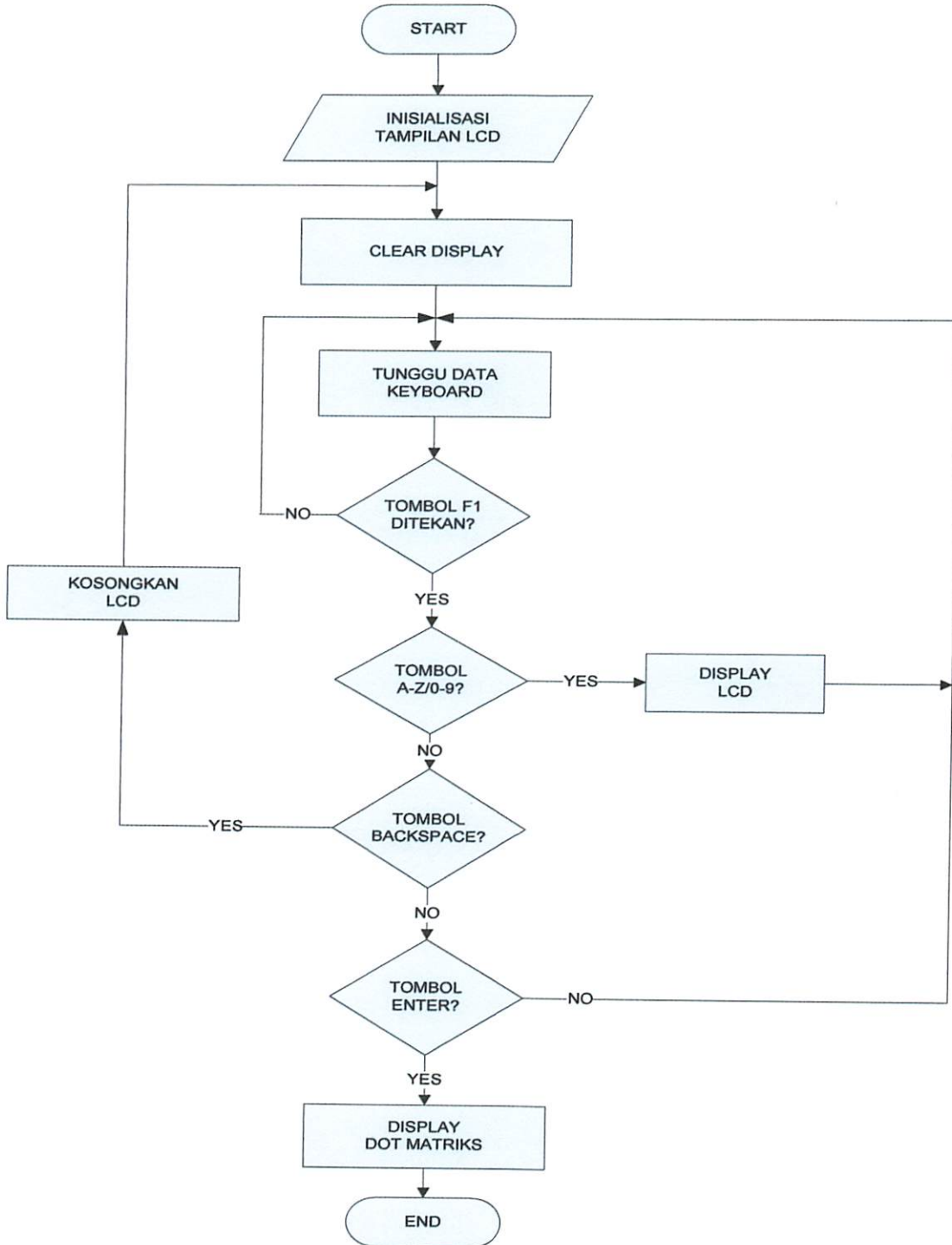
### 3.3. Perancangan Perangkat Lunak (*Software*)

Dalam menunjang kerja sistem secara keseluruhan diperlukan suatu perangkat lunak (*software*). Perangkat lunak (*software*) yang digunakan untuk AT89S52 disini menggunakan bahasa assembler keluarga MCS52. Program yang ditulis dengan bahasa *assembly* terdiri dari *label*; *kode mnemonic* dan lain sebagainya yang pada umumnya dinamakan sebagai program sumber (*source code*) yang belum bisa diterima oleh prosesor untuk dijalankan sebagai program, tetapi harus dijalankan dulu menjadi bahasa mesin dalam bentuk *code biner*.

- Penulisan program dengan menggunakan teks editor dan disimpan dengan ekstensi *Asm*.
- Meng-compile program yang telah ditulis dengan menggunakan Compiler MCS52 sehingga didapatkan file dengan ekstensi *Hex*.
- Mengubah file berekstensi *Hex* menjadi file berekstensi *Bin*.
- Men-download file berekstensi *Bin* ke dalam EPROM Mikrokontroler AT89S52.

### 3.3.1. Flowchart Kerja Rangkaian

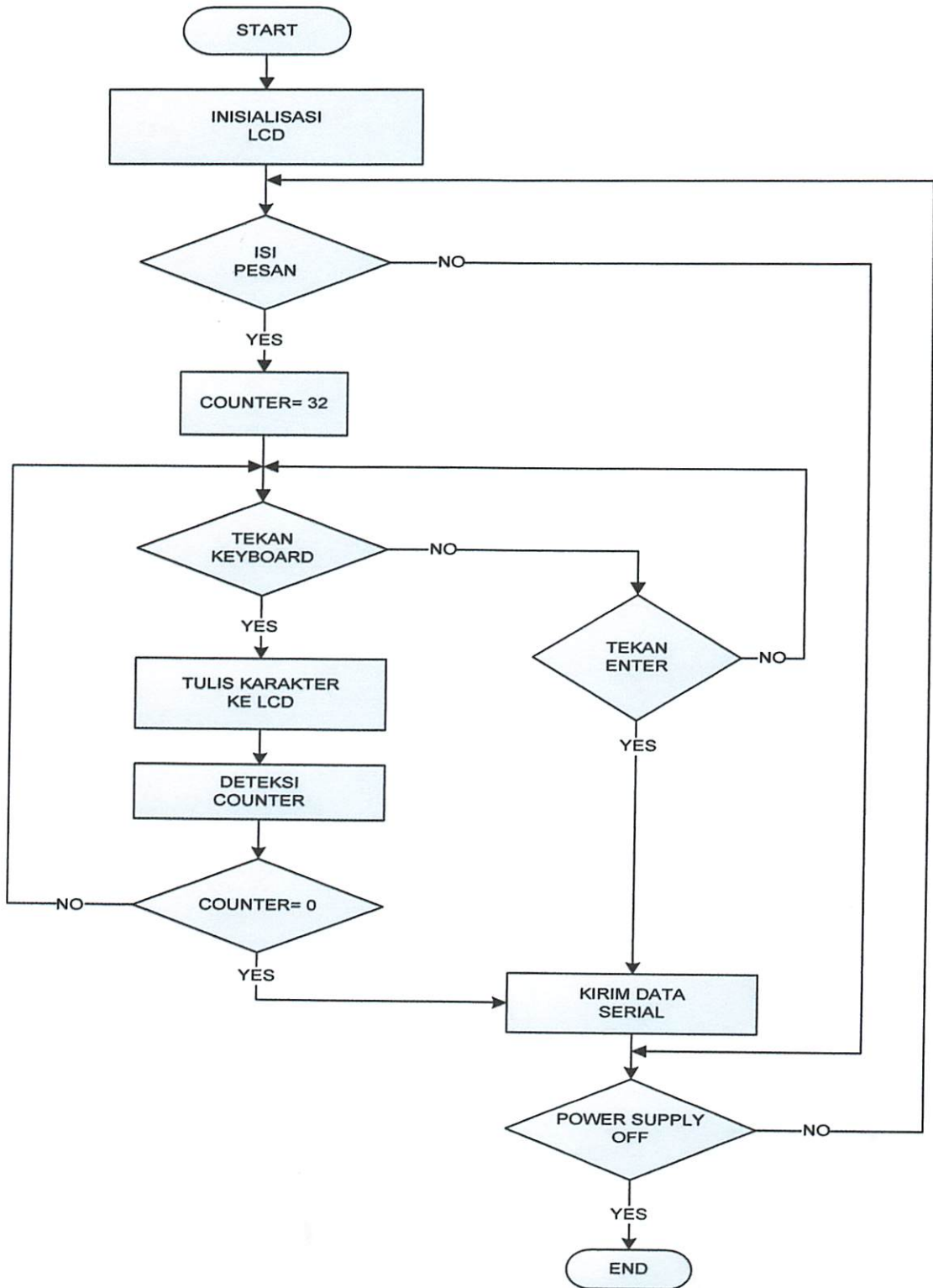
#### 3.3.1.1. Flowchart Kerja Rangkaian Keseluruhan



Gambar 3.10 Flowchart Kerja Rangkaian Keseluruhan

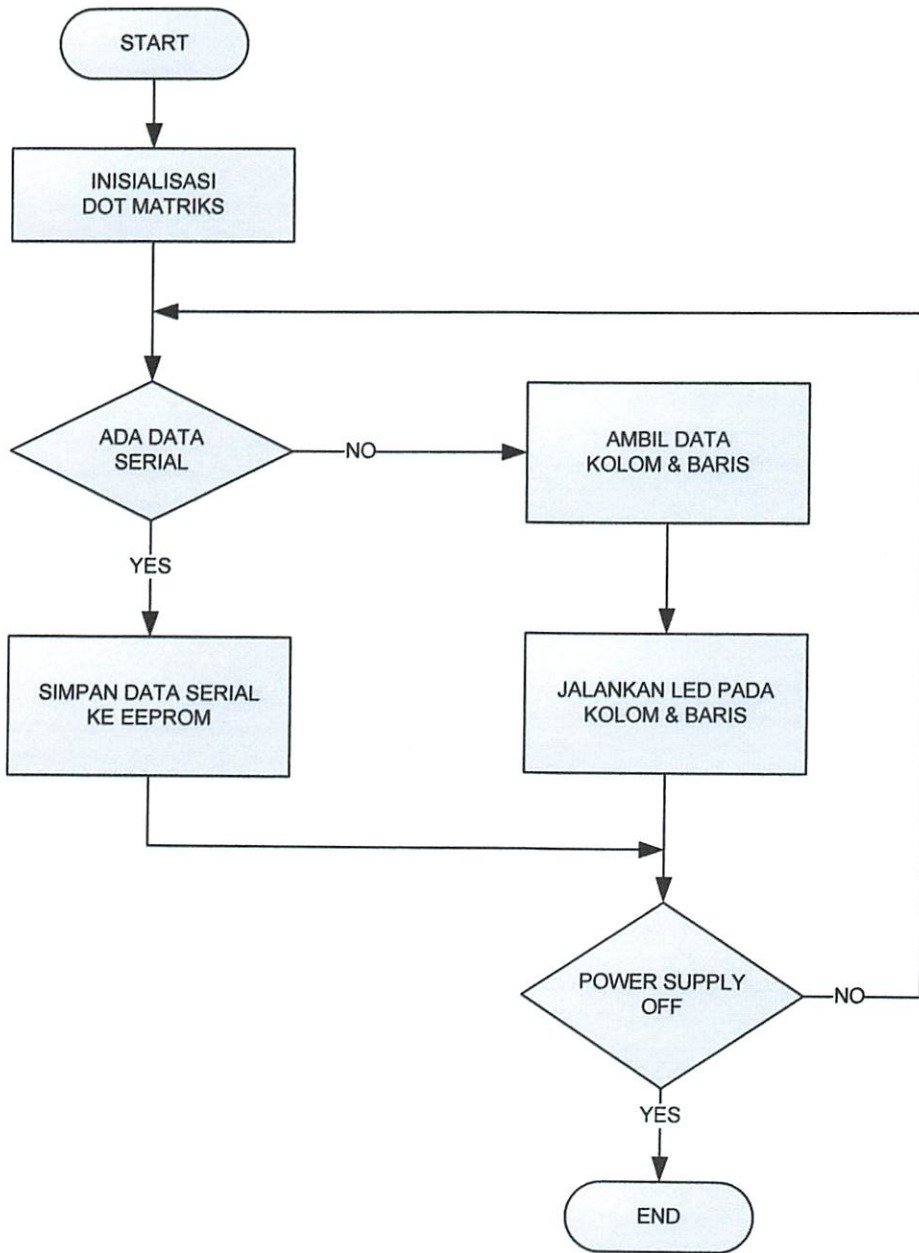


3.3.1.2. Flowhart Kerja Pada Mikrokontroler I (*Master*)



Gambar 3.11 Flowchart Kerja Pada Mikrokontroler I (*Master*)

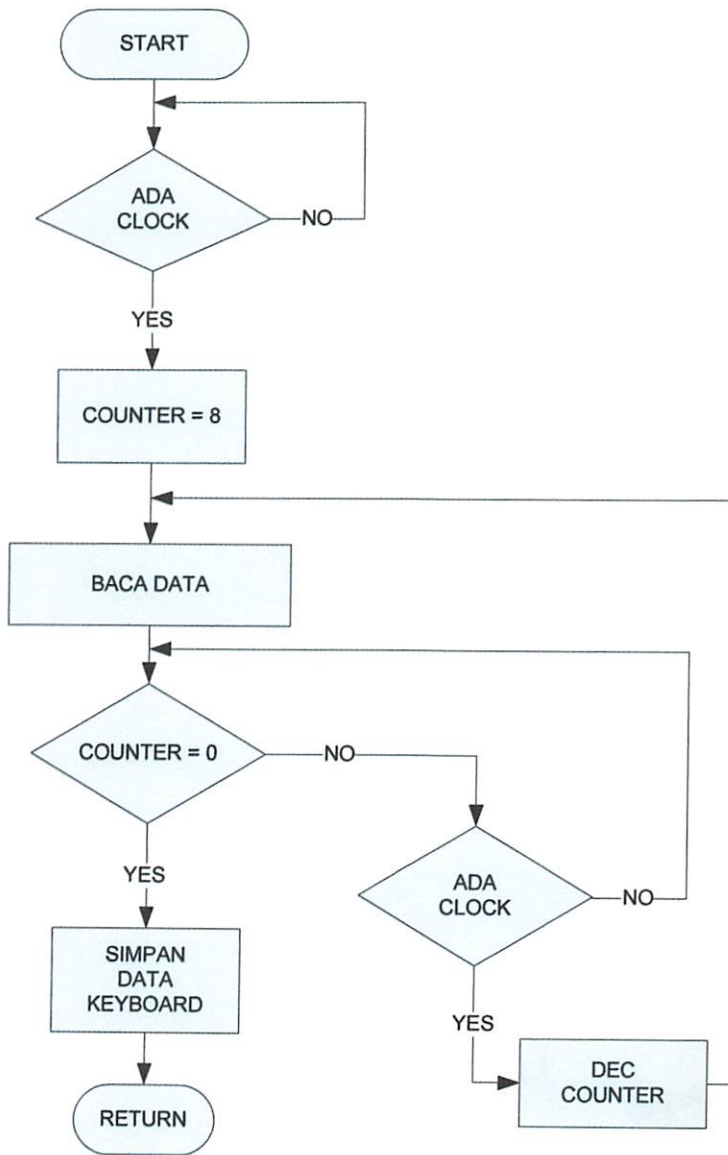
3.3.1.3. Flowchart Kerja Pada Mikrokontroler II (Slave)



Gambar 3.12 Flowchart Kerja Pada Mikrokontroler II (Slave)



3.3.1.4. Flowchart Pada *Keyboard PC*



Gambar 3.13 Flowchart Kerja Pada *Keyboard PC*

## **BAB IV**

### **PENGUJIAN SISTEM**

Pada bab ini akan membahas tentang pengujian alat yang telah dirancang, dirakit serta direalisasikan. Tujuan pengujian alat ini adalah mengetahui kerja dari masing-masing sistem yang dibuat secara per-blok. Dengan demikian dapat diketahui kepersisan kerja dari alat yang direncanakan dan dibuat, pada bab ini akan dibahas tentang pengujian sistem yang telah dirancang, yaitu sebagai berikut :

1. Pengujian terhadap mikrokontroler I (*Slave*) dan *Dot* matriks LED
2. Pengujian terhadap mikrokontroler II (*Master*) dan tampilan pada LCD
3. Pengujian pada *Keyboard PC* dan tampilan ke LCD
4. Pengujian Keseluhan Sistem

Dari pengujian sistim ini tujuannya adalah sebagai berikut:

1. Mengetahui proses kerja dari masing – masing rangkaian (blok).
2. Memudahkan pendataan spesifikasi alat.
3. Mengetahui hasil dari suatu perancangan yang telah dibuat.
4. Memudahkan perawatan dan perbaikan apabila sewaktu – waktu terjadi kerusakan pada sistem tersebut.

#### **4.1. Pengujian Rangkaian Mikrokontroler I (Master)**

##### **4.1.1. Tujuan Pengujian**

Pengujian rangkaian mikrokontroler I (*Master*) bertujuan untuk mengetahui bahwa komunikasi antara mikrokontroler dengan LCD dan *Keyboard PC* tersebut bisa bekerja seperti dalam perancangan sistemnya.

#### 4.1.2. Langkah – langkah Pengujian

Dalam pengujian rangkaian mikrokontroler I (*Master*) ini dilakukan dengan menghubungkan antara mikrokontroler I (*Master*) dengan LCD (*Liquid Crystal Display*). Mikrokontroler (*Master*) diprogram dengan kalimat “PAULO DAS DORES E SILVA 0412205”. Hasil dari pengujian mikrokontroler ini dapat dilihat pada Gambar 4.1

#### 4.1.3. Hasil dan Analisa



**Gambar 4.1 Hasil Pengujian Pada Mikrokontroler I (*Master*)**

Dari hasil pengujian pada mikrokontroler I (*Master*) ini, LCD (*Liquid Crystal Display*) menampilkan beberapa karakter atau tulisan “PAULO DAS DORES E SILVA 0412205” dengan jelas tanpa cacat, maka dapat disimpulkan bahwa pengujian pada mikrokontroler telah berhasil diuji.

## 4.2. Pengujian Rangkaian Mikrokontroler II (Slave)

### 4.2.1. Tujuan Pengujian

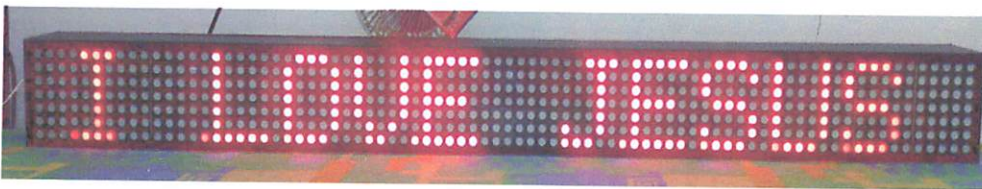
Dalam pengujian rangkaian mikrokontroler II (*Slave*) bertujuan untuk mengetahui sistem kerja dari display *dot* matrik dan IC register yang bekerja sesuai dengan apa yang telah direncanakan dalam perancangan tersebut.

### 4.2.2. Langkah – langkah Pengujian

Langkah – langkah dalam pengujian ini dilakukan dengan menghubungkan mikrokontroler II (*Slave*) dengan LED *dot* matrik. Mikrokontroler (*Slave*) ini diprogram dengan kalimat “I LOVE JESUS”. Hasil dari pengujian ini dapat dilihat pada Gambar 4.2.

### 4.2.3. Hasil dan Analisa

Pada hasil pengujian ini hanya terlihat beberapa kalimat saja dari tampilan LCD, ini disebabkan karena sistem kerja dari *dot* matrik tersebut adalah menampilkan kalimat dengan cara bergeser atau berjalan.



**Gambar 4.2 Hasil Pengujian Pada Mikrokontroler II (*Slave*)**

Dari hasil pengujian pada mikrokontroler II (*Slave*), diketahui bahwa sistem ini bekerja sesuai dengan masukan yang diberikannya berupa tulisan “I LOVE JESUS” yang terlihat pada gambar 4.2 diatas, sehingga dapat disimpulkan bahwa mikrokontroler II (*Slave*) tersebut telah berhasil diuji.

### **4.3. Pengujian Keyboard PC dan Tampilan Ke LCD**

#### **4.3.1. Tujuan Pengujian**

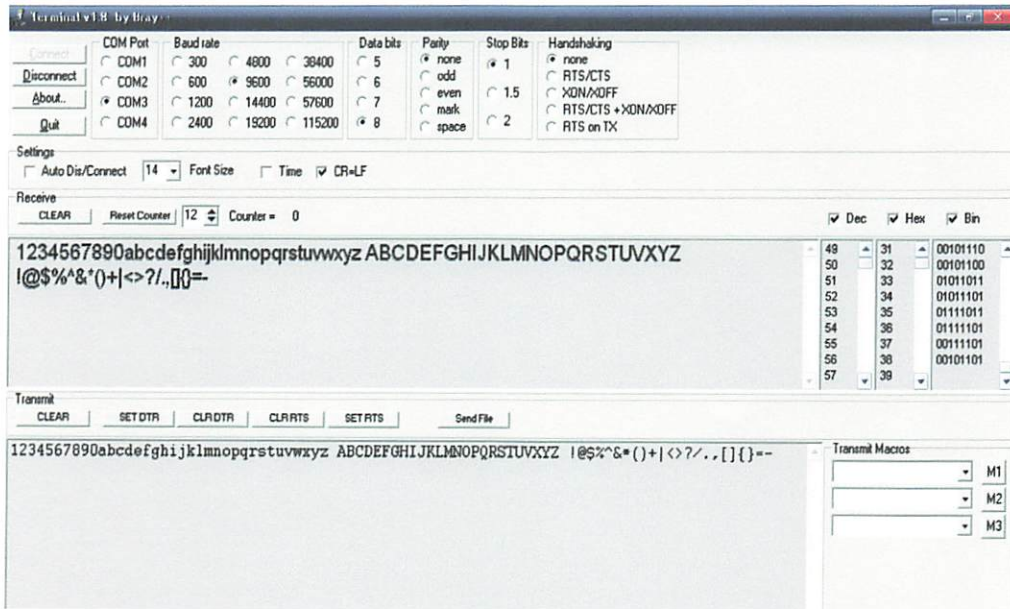
Pengujian *Keyboard PC* dan tampilan ke LCD (*Liquid Crystal Display*) disini bertujuan untuk mengetahui apakah semua tombol – tombol pada *keyboard PC* tersebut bekerja dengan baik dan juga mengetahui tampilan LCD (*Liquid Crystal Display*) yang dipergunakan sesuai dengan cara kerjanya.

#### **4.3.2. Langkah – langkah Pengujian**

Pengujian terhadap *Keyboard PC* dan tampilan ke LCD (*Liquid Crystal Display*) ini dilakukan dengan menghubungkan *keyboard PC* dengan LCD melalui mikrokontroler. Dengan menggunakan *keyboard PC* dicoba mengetikkan beberapa karakter untuk mengetahui apakah tampilan di LCD (*Liquid Crystal Display*) sesuai dengan apa yang diketikkan.

#### **4.3.3. Hasil dan Analisa**

Hasil dari pengujian *Keyboard PC* dan tampilan LCD (*Liquid Crystal Display*) dapat dilihat pada Gambar 4.4, dan disini digunakan juga *software Terminal v1.8 by Bray + +* untuk mengetahui kode – kode pada setiap tombol *keyboard* yang ditekan, berikut ini adalah gambar pengujian tombol *keyboard* dengan *Terminal v1.8 by Bray + +*.



**Gambar 4.3 Pengujian *Keyboard PC* dengan Terminal v1.8 By Bray ++**

Dari hasil pengujian yang dilakukan, maka dapat disimpulkan bahwa kode – kode yang terdapat pada *keyboard PC* berbeda, misalnya untuk huruf abjad kecil ‘a’ kode tombol-Nya 61H, sedangkan untuk huruf abjad kapital ‘A’ kode tombol-Nya 41H. Untuk lebih jelasnya maka dapat dilihat pada Tabel 4.1 dan Tabel 4.2 yaitu pengujian tombol – tombol pada *keyboard PC*.

**Tabel 4.1 Hasil Pengujian tombol abjad a – z dan A – Z pada Keyboard PC**

<b>NO</b>	<b>Jenis Tombol</b>	<b>Kode Tombol (Hex.)</b>	<b>NO</b>	<b>Jenis Tombol</b>	<b>Kode Tombol (Hex.)</b>
1	a	61	29	A	41
2	b	62	30	B	42
3	c	63	31	C	43
4	d	64	32	D	44
5	e	65	33	E	45
6	f	66	34	F	46
7	g	67	35	G	47
8	h	68	36	H	48
9	i	69	37	I	49
10	j	6A	38	J	4A
11	k	6B	39	K	4B
12	l	6C	40	L	4C
13	m	6D	41	M	4D
14	n	6E	42	N	4E
15	o	6F	43	O	4F
16	p	70	44	P	50
17	q	71	45	Q	51
18	r	72	46	R	52
19	s	73	4	S	53
20	t	74	48	T	54
21	u	75	49	U	55
22	v	76	50	V	56
23	w	77	51	W	57
24	x	78	52	X	58
25	y	79	53	Y	59
26	z	7A	54	Z	5A

**Tabel 4.2 Hasil Pengujian pada tombol angka (0 – 9) dan Simbol – simbol  
pada Keyboard PC**

<b>NO</b>	<b>Jenis Tombol</b>	<b>Kode Tombol (Hex.)</b>	<b>NO</b>	<b>Jenis Tombol</b>	<b>Kode Tombol (Hex.)</b>
1	!	21	24	<	3C
2	@	40	25	>	3E
3	#	23	26	?	3F
4	\$	24	27	,	2C
5	%	25	28	.	2E
6	^	5E	29	/	2F
7	&	26	30	'	27
8	*	2A	31	“	22
9	(	28	32	;	3B
10	)	29	33	Esc	1B
11	_	5F	34	Enter	0D
12	+	2B	35	BackSpace	08
13		7C	36	Space	20
14	~	7E	37	0	30
15	`	60	38	1	31
16	-	2D	39	2	32
17	=	3D	40	3	33
18	\	5C	41	4	34
19	{	7B	42	5	35
20	}	7D	43	6	36
21	[	5B	44	7	37
22	]	5D	45	8	38
23	:	3A	46	9	39



Pada pengujian ini, untuk mengetahui apakah *scan code* dalam penekanan tombol *Keyboard PC* berjalan dengan lancar atau tidak. Hasil dari pengujian adalah sebagai berikut:

**Tabel 4.3 Pengujian *Keyboard PC***

<b>Tombol</b>	<b>Keterangan</b>
<i>CapsLock</i> ditekan	Flag Carry=0, karakter diubah menjadi huruf besar
G ditekan	Muncul huruf kapital G di LCD
F1 ditekan	Muncul informasi di LCD yang berisi “Tulis Pesan”
<i>CapsLock</i> ditekan	Flag Carry=0, karakter diubah menjadi huruf kecil
A ditekan	Muncul huruf a di LCD
1 ditekan	Muncul angka 1 di LCD
<i>Space</i> ditekan	Kursor bergeser ke kanan sebanyak satu kali

Dari hasil pengujian ini bisa diketahui bahwa ada beberapa tombol *keyboard PC* yang difungsikan sebagai mana mestinya sesuai dengan fungsi dan cara kerja pada sistem tersebut, yaitu:

- Tombol F1 sebagai perintah untuk penulisan pesan pada tampilan LCD (*Liquid Crystal Display*).
- Tombol *Esc* sebagai perintah untuk memberikan kondisi kembali seperti semula atau pembatalan untuk penulisan pada LCD (*Liquid Crystal Display*).
- Tombol *BackSpace* untuk menghapus tulisan yang dianggap salah pada tampilan LCD (*Liquid Crystal Display*).

- Tombol angka 0 – 9 utama dan kedua.
- Tombol huruf A – Z
- Tombol tanda garis miring (/)
- Tombol tanda titik (.)
- Tombol tanda koma (,)
- Tombol tanda strep (-)
- Tombol *Enter* sebagai perintah untuk mengirimkan data ke tampilan *dot* matriks.
- Tombol *CapsLock* untuk mengubah huruf besar dan kecil.
- Tombol tanda titik dua (:)
- Tombol *Space* untuk memberikan spasi jarak pada penulisan pesan atau karakter pada LCD (*Liquid Crystal Display*).

Berikut ini adalah hasil dari pengujian *Keyboard PC* dan tampilan ke LCD (*Liquid Crystal Display*)



**Gambar 4.4 Hasil Pengujian *Keyboard PC* dan Tampilan pada LCD**

Dari hasil pengujian yang dilakukan terhadap *keyboard PC* dan tampilan ke LCD (*Liquid Crystal Display*) ini dikatakan telah berhasil diuji karena karakter yang diketikkan pada *keyboard PC* semua-Nya ditampilkan ke LCD (*Liquid Crystal Display*).

Dalam perancangan dan pembuatan sistem ini ada beberapa tombol *Keyboard* yang tidak dipergunakan, yaitu tombol – tombol yang tidak disebutkan diatas apabila ditekan maka sistem ini akan terjadi *error* atau pergeseran kursor pada LCD (*Liquid Crystal Display*). Berikut ini adalah gambar dari pengujian *Keyboard PC* dan Tampilan pada LCD (*Liquid Crystal Display*) yang terjadi *error* atau pergeseran kursor.



**Gambar 4.5 Hasil Penekanan Tombol yang tidak diinginkan**

Dari pengujian yang ditunjukkan pada gambar 4.4 diatas, disebabkan karena penekanan pada tombol – tombol yang tidak dipergunakan pada perancangan ini, sehingga terjadi pergeseran kursor pada LCD (*Liquid Crystal Display*).

## 4.4. Pengujian Rangkaian Dot Matriks LED

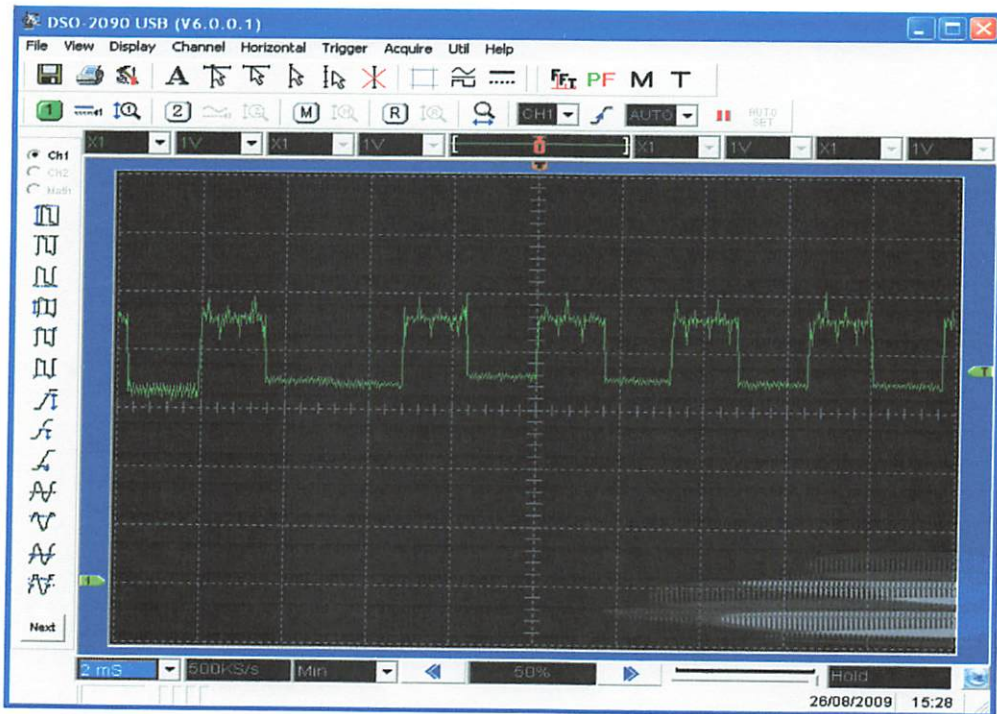
### 4.4.1. Tujuan Pengujian

Pengujian rangkaian *dot* matriks LED bertujuan untuk mengetahui perancangan sistem, khususnya untuk mengetahui waktu yang dipergunakan dalam menyalakan satu kolom *dot* matriks LED.

### 4.4.2. Langkah – langkah Pengujian

Pengujian *dot* matriks ini dilakukan dengan menghubungkan *probe oscilloscope* ke salah satu pin mikrokontroler (*Slave*) yang terhubung dengan salah satu kolom *dot* matrik LED, dalam hal ini digunakan P1.0. sebagai pengukurannya.

### 4.4.3. Hasil dan Analisa



**Gambar 4.6 Hasil Pengujian Pada Rangkaian *Dot* Matriks LED**

Dalam pengukuran ini diketahui bahwa besar waktu yang diperlukan untuk menyalakan satu kolom dot matriks adalah sebesar  $2 \text{ ms} \times 0,8 \text{ kotak} = 1,6 \text{ ms}$ . Jadi

sebenarnya setiap kolom hanya menyala selama 1,6 ms, namun karena sistem scanning ini berjalan dengan cepat, maka kelihatannya menyala bersamaan. Dari hasil pengujian juga dapat dilihat bahwa waktu yang dipergunakan untuk mematikan satu kolom *dot* matriks adalah sebesar  $2 \text{ ms} \times 0,8 \text{ kotak} = 1,6 \text{ ms}$ . Berdasarkan hasil dari pengujian ini, dapat diperoleh bahwa frekuensi dari sistem adalah sebesar :

$$F = 1 / T$$

$$F = 1 / 1,6 \cdot 10^{-3}$$

$$F = 62,50 \text{ Hz}$$

#### **4.5. Pengujian Kelebihan Karakter Pada LCD (*Liquid Crystal Display*)**

##### **4.5.1. Tujuan Pengujian**

Pada pengujian ini, bertujuan untuk mengetahui jumlah karakter yang akan dituliskan pada LCD (*Liquid Crystal Display*), dan mengetahui jika kelebihan karakter tersebut terjadi pada LCD (*Liquid Crystal Display*).

##### **4.5.2. Langkah – langkah**

Untuk melakukan pengujian ini pada sistem, maka dilakukan dengan menghubungkan antara mikrokontroler I (*Master*), mikrokontroler II (*Slave*), *Keyboard PC* dan LCD (*Liquid Crystal Display*) yang terhubung dengan baik. Pada *keyboard PC* diketikkan beberapa karakter misalnya “Institut Teknologi Nasional ITN MALANG” jadi jumlah karakter pada tulisan tersebut adalah 34 huruf dan 4 kali spasi maka jumlah keseluruhan-Nya adalah 38 karakter. Untuk melihat pengujian hasil-Nya dapat dilihat pada gambar 4.6.



### 4.5.3. Hasil dan Analisa

Hasil dari pengujian kelebihan karakter yang dituliskan pada LCD (*Liquid Crystal Display*) dapat dilihat pada gambar berikut:



**Gambar 4.7 Hasil Pengujian Kelebihan Karakter pada LCD  
(*Liquid Crystal Display*)**

Dari hasil pengujian ini, hasil tampilan pada LCD (*Liquid Crystal Display*) dan *dot* matriks LED hanya menampilkan beberapa karakter saja dari kalimat yang diinginkan, yaitu tulisan “Institut Teknologi Nasional ITN MALANG” karena jumlah kalimat tersebut melebihi dari 32 karakter, sehingga pada saat ada penekan tombol berikut-Nya maka tulisan yang pada LCD (*Liquid Crystal Display*) langsung terkirim ke *dot* matriks LED. Dengan hal ini dapat dianalisa bahwa tampilan pada *dot* matriks LED tergantung pada jumlah karakter pada LCD (*Liquid Crystal Display*) bukan pada internal RAM mikrokontroler.

## 4.6. Pengujian Pengiriman Data Antara Mikrokontroler I (*Master*) Dengan Mikrokontroler II (*Slave*)

### 4.6.1. Tujuan Pengujian

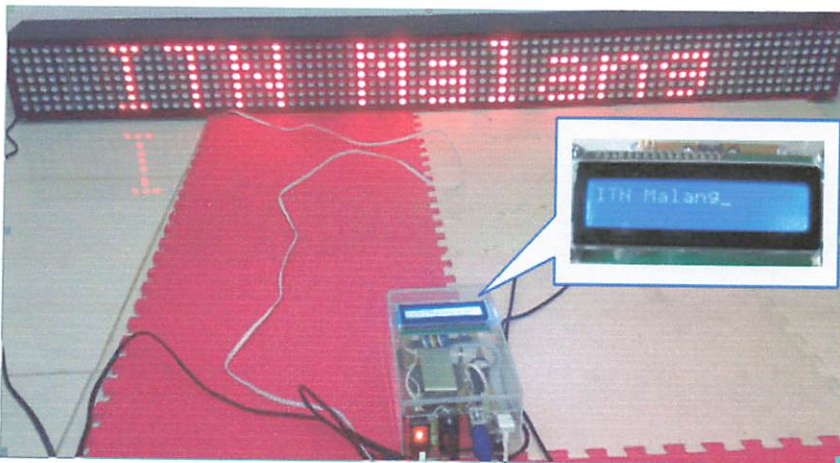
Pada pengujian pengiriman data dari mikrokontroler I (*Master*) terhadap mikrokontroler II (*Slave*) ini, bertujuan untuk mengetahui pengiriman data dari mikrokontroler I (*Master*) ke mikrokontroler II (*Slave*).

### 4.6.2. Langkah – langkah

Untuk melakukan pengujian ini, maka diperlukan hubungan antara mikrokontroler I (*Master*) dengan mikrokontroler II (*Slave*) dan pada mikrokontroler I (*Master*) dituliskan kalimat “ITN Malang” pada LCD (*Liquid Crystal Display*) dan hasil tersebut dikirimkan ke mikrokontroler II (*Slave*) yang ditampilkan ke *dot* matriks LED.

### 4.6.3. Hasil dan Analisa

Dari pengujian ini dapat dilihat pada gambar 4.7 berikut:



Gambar 4.8 Hasil Pengujian Pengiriman Data dari mikrokontroler I (*Master*) ke mikrokontroler II (*Slave*)

Pada pengujian pengiriman data dari mikrokontroler I (*Master*) ke mikrokontroler II (*Slave*) ini diketahui bahwa, data tulisan yang dikirimkan dari mikrokontroler I (*Master*) ke mikrokontroler II (*Slave*) telah berhasil diuji karena hasil akhir yang ditampilkan pada dot matriks LED sama seperti data tulisan yang ada pada mikrokontroler I (*Master*) yaitu tulisan pada LCD (*Liquid Crystal Display*).

#### **4.7. Pengujian Rangkaian Keseluruhan Sistem**

##### **4.7.1. Tujuan Pengujian**









Dalam pengujian sistem keseluruhan ini, bertujuan agar mengetahui cara kerja dari sistem yang telah dirancang bekerja dengan baik, sesuai dengan apa yang telah direncanakan.

##### **4.7.2. Langkah – langkah**

Untuk melakukan pengujian pada sistem keseluruhan ini, maka menghubungkan antara mikrokontroler I (*Master*), mikrokontroler II (*Slave*), *Keyboard PC* dan LCD (*Liquid Crystal Display*) terhubung dengan baik. Pada keyboard PC diketikkan karakter “I LOVE JESUS 0983” yang muncul pada LCD (*Liquid Crystal Display*) sehingga dikirim ke *dot* matriks LED dan hasil tampilan pada *Dot* matriks hanya muncul “I LOVE JESUS” karena hasil tampilan pada *dot* matriks tersebut bersifat pergeseran karakter. Pergeseran ini dilakukan oleh IC 74LS164 yang sebagai IC register untuk mengeser baris pada *dot* matriks LED, berikut ini adalah tabel pergeseran pada IC 74LS164.



**Tabel 4.4 Pergeseran Pada IC 74LS164**

MASUKAN		KELUARAN							
DATA	CLOCK	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
L		L	*	*	*	*	*	*	*
H		H	L	*	*	*	*	*	*
H		H	H	L	*	*	*	*	*
L		L	H	H	L	*	*	*	*
L		L	L	H	H	L	*	*	*
H		H	L	L	H	H	L	*	*
L		L	H	L	L	H	H	L	*
H		H	L	H	L	L	H	H	L

**4.7.3. Hasil dan Analisa**

Dari hasil pengujian sistem keseluruhan ini dapat dilihat pada Gambar 4.5 berikut :



**Gambar 4.9 Hasil Pengujian Sistem Keseluruhan**

Dari hasil pengujian sistem keseluruhan ini, diketahui bahwa tulisan yang diketikkan oleh *keyboard PC* sehingga ditampilkan pada LCD (*Liquid Crystal Display*) dan dikirim ke *dot matriks LED* telah berhasil diuji karena hasil tampilan pada *dot matriks LED* sama seperti tulisan yang diketikkan dari *keyboard PC*.

## **BAB V**

### **PENUTUP**

Bab ini akan dibahas tentang kesimpulan yang diambil dari keseluruhan pada sub bab – bab sebelumnya, yaitu akan dibahas sebagai berikut:

#### **5.1 KESIMPULAN**

Dari hasil perancangan dan pengujian sistem ini maka dapat diambil kesimpulan-nya sebagai berikut:

1. Pada perancangan ini, karakter yang akan dituliskan ke LCD tidak boleh melebihi 32 karakter, jika hal ini terjadi maka tulisan yang diketikkan pada LCD (*Liquid Crystal Display*) langsung terkirim ke *dot* matriks LED.
2. Untuk tombol – tombol *keyboard PC* yang tidak digunakan pada kerja sistem ini, apabila ditekan maka terjadi pergeseran *cursor* pada LCD (*Liquid Crystal Display*).
3. Pada kerja sistem ini, apabila hubungan ke tegangan sumber (AC) terputus, maka tulisan yang terakhir kali tampil pada *dot* matriks LED masih tetap tersimpan, sehingga pada saat sumber tegangan terhubung kembali maka tulisan tersebut masih ditampilkan lagi.

## 5.2 SARAN

Dalam perancangan sistem ini, untuk lebih lanjutnya dalam pengembangan sistem ini, maka penulis menyarankan bahwa :

1. Sistem pengiriman data atau komunikasi antara mikrokontroler I (*Master*) dengan mikrokontroler II (*Slave*) menggunakan komunikasi serial yang bisa pengiriman datanya berjarak semakin jauh.
2. Tombol – tombol pada *keyboard PC* difungsikan semua menurut sistem kerja dan kegunaannya.
3. Untuk penulisan karakter pada LCD (*Liquid Crystal Display*) lebih dari 32 karakter atau penulisan karakter tidak dibatasi.

## DAFTAR PUSTAKA

- [1] [www.Atmel.com](http://www.Atmel.com)
- [2] M1632 LCD Unit User's manual, Seiko Instrument Inc.
- [3] [www.Beyondlogic.org](http://www.Beyondlogic.org)
- [4] Data sheet 74LS373 [www.alldatasheet.com](http://www.alldatasheet.com)
- [5] Data sheet 74LS164 [www.alldatasheet.com](http://www.alldatasheet.com)
- [6] [www.dtechtronic.com](http://www.dtechtronic.com)
- [7] Simulasi Elektronika Digital menggunakan Electronic workbench, **Purnomo Gatot Santosa**, ANDY Yogyakarta 2007.
- [8] [www.digi-ware.com](http://www.digi-ware.com)
- [9] Belajar dengan mudah dan cepat Pemrograman bahasa C dengan Small Device C Compiler pada Mikrokontroler AT89x051/AT89C51/52, Teori Simulasi dan Aplikasi. **Totok Budioko**



**L**

**A**

**M**

**P**

**I**

**R**

**A**

**N**



**LEMBARAN  
PERSEMBAHAN**

*Ku titipkan sedikit kata buat smua-nya:*

## **JESUS CHRIST AND IMMACULATE MOTHER**

Sang Maha Pencipta dan Tuhan Yang Maha Esa **"TUHAN YESUS KRISRUS"** puji syukur ke hadirat-Mu ya **"ALLAH BUNDA MARIA"** yang selalu melimpahkan Berkat Rahmat, Karunia, petunjuk serta hidayah-Nya bagi hamba dan kita semua.

♥ **"PAPA"** N **"MAMA"** tercinta yang telah mengajarkanku berbagai macam hal, yang telah memberikan segalanya untuk membahagiakanku dan selalu mengingatkanku jika aku salah serta selalu mengingatkanku untuk Berdo'a dan ingat kepada **"TUHAN YESUS KRISTUS"**. Begitu banyak kasih sayang yang telah mereka berikan padaku tapi aku tak bisa membalas semuanya, hanya doa yang bisa kupanjatkan kehadirat-Nya semoga **"BAPAK DISURGA"** memberikan yang terbaik selalu untuk mereka. Amin...

♥ **Brothers N Sisters** ku makasih atas segala bimbingan dan support yang kalian semua berikan padaku, thanx ya...!!!

♥ My sweet Child **Vanecha** N **Nandito** always which make happiness and give strong to me for my advance, I love you.

♥ **PaiZhe** N **MaeZhy** thanks you for everything, support and anything to me thanks for all.



- ♥ My Big Family Wai-Da trima kasih untuk semua yang telah kalian semua lakukan untuk aku, thanx atas doa dan supportnya.
  
- ♥ For some1 “*Q-noy*” thanks you baby for you’re supported and prayed during these times, thanks baby for you love.
  
- ♥ Buat saudara/saudariku semua yang selama ini telah membantu saya dalam menyelesaikan studiku ini, thanx ya semuanya.
  
- ♥ Thanx buat saudaraku Elly N Tino yang telah memberikan dorongan dan smagat padaku pada saat aku ada problems internal yang bertepatan dengan ujianku, makasih ya.
  
- ♥ Buat teman – teman seperjuang semua makasih ya atas semua yang telah kalian lakuin ke aku.
  
- ♥ Buat teman – teman dan adik – adik IMTTL ITN Malang yang telah memberikan kepercayaan kepemimpinan dalam pengurusan IMTTL ITN Malang selama ini, trima kasih atas semuanya.



## FORMULIR BIMBINGAN SKRIPSI

Nama : PAULO DAS DORES E SILVA  
Umur : 04.12.205  
Masa Bimbingan : 29 Mei 2009 s/d 29 November 2009 f  
Judul Skripsi : PERANCANGAN DAN PEMBUATAN PENAMPIL JUDUL DAN  
NOMOR LAGU YANG DIPERGUNAKAN DI GEREJA BERBASIS  
MIKROKONTROLER AT89S52

NO	Tanggal	Uraian	Paraf Pembimbing
1	24-6-2009	Bab I + II di revisi	MS
2	29-6-2009	Bab I + II ok	MS
3	11-08-2009	Bab III di revisi	MS
4	13-08-2009	Bab III ok	MS
5	24-08-2009	Demo Alat	MS
6	25-08-2009	Bab IV + V ok	MS
7	04-09-2009	Seminar hasil	MS
8	29-09-2009	kompre	MS
9			
10			

Malang,  
Dosen Pembimbing I

Ir. TH. Mimien Mustikawati, MT  
NIP. 103 000 0352



## FORMULIR BIMBINGAN SKRIPSI

Nama : PAULO DAS DORES E SILVA  
 Tanggal : 04.12.2005  
 Masa Bimbingan : 29 Mei 2009 s/d 29 November 2009  
 Judul Skripsi : PERANCANGAN DAN PEMBUATAN PENAMPIL JUDUL DAN  
 NOMOR LAGU YANG DIPERGUNAKAN DI GEREJA BERBASIS  
 MIKROKONTROLER AT89S52

NO	Tanggal	Uraian	Paraf Pembimbing
1	30-06-2009	BAB I + II	<i>fadi</i>
2	20-08-2009	BAB III	<i>fadi</i>
3	24-08-2009	Demo ALU	<i>fadi</i>
4	25-08-2009	BAB IV	<i>fadi</i>
5	25-08-2009	BAB V	<i>fadi</i>
6	04-09-2009	SEMINAR HASIL	<i>fadi</i>
7	29-09-2009	KOMPRES	<i>fadi</i>
8			
9			
10			

Malang,  
Dosen Pembimbing II

Sotobhadi, ST. MSc  
NIP.1030 970 0309



INSTITUT TEKNOLOGI NASIONAL MALANG  
 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 / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : Paulo das POMES ES  
 NIM : 0412205  
 Perbaikan meliputi :

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---


---

---

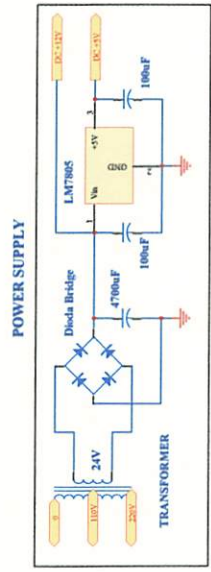
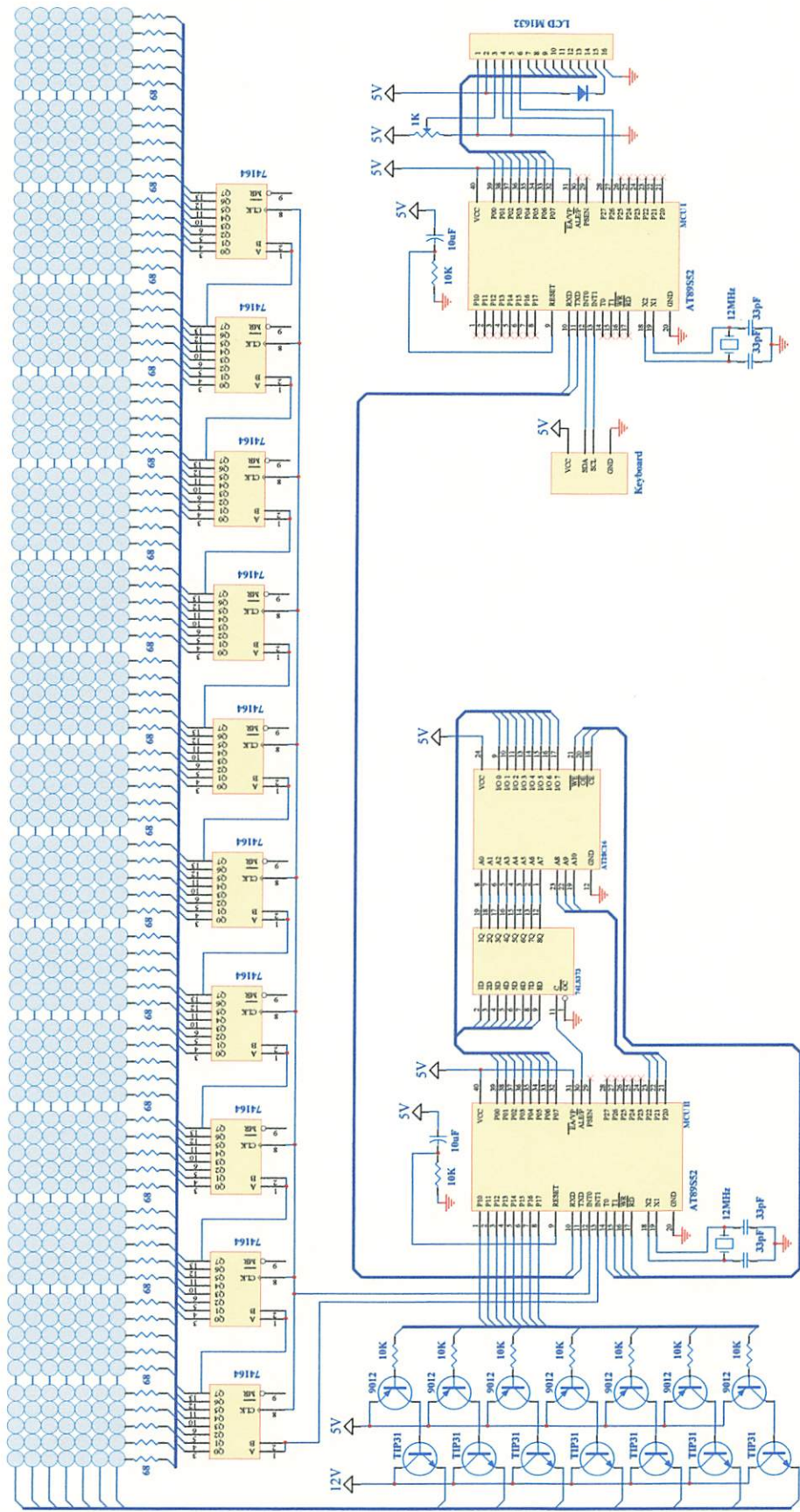
---

---

Malang, 29 sept 2009 .

(  )  
 ( M. Ibrahim aslami )

**GAMBAR RANGKAIAN  
KESELURUHAN SISTEM**



RANGKAIAN SCHEMATIC			
No	Revisi	Revisi	Revisi
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			
27			
28			
29			
30			
31			
32			
33			
34			
35			
36			
37			
38			
39			
40			
41			
42			
43			
44			
45			
46			
47			
48			
49			
50			
51			
52			
53			
54			
55			
56			
57			
58			
59			
60			
61			
62			
63			
64			
65			
66			
67			
68			
69			
70			
71			
72			
73			
74			
75			
76			
77			
78			
79			
80			
81			
82			
83			
84			
85			
86			
87			
88			
89			
90			
91			
92			
93			
94			
95			
96			
97			
98			
99			
100			

**LISTENING PROGRAM  
FOR MCU I**

### Listening Program for PC Keyboard

```
org    00h
ljmp   init
;

org    13h
ljmp   scnkbd
;

Rest   Bit P2.6           ; RS LCD
Enbl   Bit P2.7           ; E LCD
Kdta   Bit P3.2           ; keyboard data
Kclk   Bit P3.3           ; keyboard clock

Stts   Bit 20h.0         ; status char besar / kecil

Dt00   Equ 30h           ; data 00
Dt01   Equ 31h           ; data 01
Dt02   Equ 32h           ; data 02
Dt03   Equ 33h           ; data 03
Dt04   Equ 34h           ; data 04
Dt05   Equ 35h           ; data 05
Dt06   Equ 36h           ; data 06
Dt07   Equ 37h           ; data 07
Dt08   Equ 38h           ; data 08
Dt09   Equ 39h           ; data 09
Dt0A   Equ 3Ah           ; data 0A
Dt0B   Equ 3Bh           ; data 0B
Dt0C   Equ 3Ch           ; data 0C
Dt0D   Equ 3Dh           ; data 0D
Dt0E   Equ 3Eh           ; data 0E
Dt0F   Equ 3Fh           ; data 0F

Dt10   Equ 40h           ; data 10
Dt11   Equ 41h           ; data 11
Dt12   Equ 42h           ; data 12
Dt13   Equ 43h           ; data 13
Dt14   Equ 44h           ; data 14
Dt15   Equ 45h           ; data 15
Dt16   Equ 46h           ; data 16
Dt17   Equ 47h           ; data 17
Dt18   Equ 48h           ; data 18
Dt19   Equ 49h           ; data 19
Dt1A   Equ 4Ah           ; data 1A
Dt1B   Equ 4Bh           ; data 1B
Dt1C   Equ 4Ch           ; data 1C
Dt1D   Equ 4Dh           ; data 1D
Dt1E   Equ 4Eh           ; data 1E
Dt1F   Equ 4Fh           ; data 1F

Char    Equ 50h           ; char LCD
Cnt0    Equ 51h           ; counter 0
Cnt1    Equ 52h           ; counter 1
Dly0    Equ 53h           ; delay0
Dly1    Equ 54h           ; delay1
Dly2    Equ 55h           ; delay2
Dly3    Equ 56h           ; delay3
;
```



```

init:   acall   lcd_in
        acall   srl_in
        clr     Stts           ; reset status char besar /kecil
;
mulai:  mov     DPTR,#nama
        acall   line1
        mov     Char,#16
        acall   tulis
        mov     DPTR,#nim
        acall   line2
        mov     Char,#16
        acall   tulis
        acall   delay2
        mov     DPTR,#jur
        acall   line1
        mov     Char,#16
        acall   tulis
        mov     DPTR,#univ
        acall   line2
        mov     Char,#16
        acall   tulis
        acall   delay2
        ljmp    mulai
;
isidta: lcall   lcdclr
        mov     DPTR,#tptlps
        lcall   line1
        mov     Char,#16
        lcall   tulis
        mov     Dly3,#10
        lcall   delay3
        lcall   rstdt0
        mov     R1,#0           ; set urut = 0
isdt00: lcall   lcdclr
        lcall   delay0
urut00: mov     A,#080h
        orl    A,R1
        mov     P0,A
        lcall   w_ins
        mov     P0,#' '
        lcall   w_chr
        mov     A,#080h
        orl    A,R1
        mov     P0,A
        lcall   w_ins
;
urut01: cjne   R1,#00,urut02
        lcall   tgchar
        mov     Dt00,A
        inc    R1
urut02: cjne   R1,#01,urut03
        lcall   tgchar
        mov     Dt01,A
        inc    R1
urut03: cjne   R1,#02,urut04
        lcall   tgchar
        mov     Dt02,A

```

```

        inc      R1
urut04: cjne    R1,#03,urut05
        lcall   tgchar
        mov     Dt03,A
        inc     R1
urut05: cjne    R1,#04,urut06
        lcall   tgchar
        mov     Dt04,A
        inc     R1
urut06: cjne    R1,#05,urut07
        lcall   tgchar
        mov     Dt05,A
        inc     R1
urut07: cjne    R1,#06,urut08
        lcall   tgchar
        mov     Dt06,A
        inc     R1
urut08: cjne    R1,#07,urut09
        lcall   tgchar
        mov     Dt07,A
        inc     R1
urut09: cjne    R1,#08,urut10
        lcall   tgchar
        mov     Dt08,A
        inc     R1
urut10: cjne    R1,#09,urut11
        lcall   tgchar
        mov     Dt09,A
        inc     R1
urut11: cjne    R1,#10,urut12
        lcall   tgchar
        mov     Dt0A,A
        inc     R1
urut12: cjne    R1,#11,urut13
        lcall   tgchar
        mov     Dt0B,A
        inc     R1
urut13: cjne    R1,#12,urut14
        lcall   tgchar
        mov     Dt0C,A
        inc     R1
urut14: cjne    R1,#13,urut15
        lcall   tgchar
        mov     Dt0D,A
        inc     R1
urut15: cjne    R1,#14,urut16
        lcall   tgchar
        mov     Dt0E,A
        inc     R1
urut16: cjne    R1,#15,urut17
        lcall   tgchar
        mov     Dt0F,A
        inc     R1
;
urut17: mov     A,R1
        mov     B,#16
        clr    C

```

```

        subb    A,B
        mov     R2,A
        mov     A,#0C0h
        orl    A,R2
        mov     P0,A
        lcall   w_ins

        mov     P0,#' '
        lcall   w_chr

        mov     A,R1
        mov     B,#16
        clr    C
        subb   A,B
        mov     R2,A
        mov     A,#0C0h
        orl    A,R2
        mov     P0,A
        lcall   w_ins

        cjne   R1,#16,urut18
        lcall   tgchar
        mov     Dt10,A
        inc    R1
urut18:  cjne   R1,#17,urut19
        lcall   tgchar
        mov     Dt11,A
        inc    R1
urut19:  cjne   R1,#18,urut20
        lcall   tgchar
        mov     Dt12,A
        inc    R1
urut20:  cjne   R1,#19,urut21
        lcall   tgchar
        mov     Dt13,A
        inc    R1
urut21:  cjne   R1,#20,urut22
        lcall   tgchar
        mov     Dt14,A
        inc    R1
urut22:  cjne   R1,#21,urut23
        lcall   tgchar
        mov     Dt15,A
        inc    R1
urut23:  cjne   R1,#22,urut24
        lcall   tgchar
        mov     Dt16,A
        inc    R1
urut24:  cjne   R1,#23,urut25
        lcall   tgchar
        mov     Dt17,A
        inc    R1
urut25:  cjne   R1,#24,urut26
        lcall   tgchar
        mov     Dt18,A
        inc    R1
urut26:  cjne   R1,#25,urut27

```

```

        lcall    tgchar
        mov     Dt19,A
        inc     R1
urut27: cjne    R1,#26,urut28
        lcall    tgchar
        mov     Dt1A,A
        inc     R1
urut28: cjne    R1,#27,urut29
        lcall    tgchar
        mov     Dt1B,A
        inc     R1
urut29: cjne    R1,#28,urut30
        lcall    tgchar
        mov     Dt1C,A
        inc     R1
urut30: cjne    R1,#29,urut31
        lcall    tgchar
        mov     Dt1D,A
        inc     R1
urut31: cjne    R1,#30,urut32
        lcall    tgchar
        mov     Dt1E,A
        inc     R1
urut32: cjne    R1,#31,isdt01
        lcall    tgchar
        mov     Dt1F,A
;
isdt01: lcall    lcdclr
        mov     DPTR,#tpkrdt
        lcall    line1
        mov     Char,#16
        lcall    tulis
;
        mov     A,#1                ; header
        lcall    kr_srl
        mov     Dly3,#1
        acall   delay3
;
        mov     A,Dt00
        lcall    kr_srl
        mov     A,Dt01
        lcall    kr_srl
        mov     A,Dt02
        lcall    kr_srl
        mov     A,Dt03
        lcall    kr_srl
        mov     A,Dt04
        lcall    kr_srl
        mov     A,Dt05
        lcall    kr_srl
        mov     A,Dt06
        lcall    kr_srl
        mov     A,Dt07
        lcall    kr_srl
        mov     A,Dt08
        lcall    kr_srl
        mov     A,Dt09

```

```

        lcall    kr_srl
        mov     A,Dt0A
        lcall    kr_srl
        mov     A,Dt0B
        lcall    kr_srl
        mov     A,Dt0C
        lcall    kr_srl
        mov     A,Dt0D
        lcall    kr_srl
        mov     A,Dt0E
        lcall    kr_srl
        mov     A,Dt0F
        lcall    kr_srl
;
        mov     A,Dt10
        lcall    kr_srl
        mov     A,Dt11
        lcall    kr_srl
        mov     A,Dt12
        lcall    kr_srl
        mov     A,Dt13
        lcall    kr_srl
        mov     A,Dt14
        lcall    kr_srl
        mov     A,Dt15
        lcall    kr_srl
        mov     A,Dt16
        lcall    kr_srl
        mov     A,Dt17
        lcall    kr_srl
        mov     A,Dt18
        lcall    kr_srl
        mov     A,Dt19
        lcall    kr_srl
        mov     A,Dt1A
        lcall    kr_srl
        mov     A,Dt1B
        lcall    kr_srl
        mov     A,Dt1C
        lcall    kr_srl
        mov     A,Dt1D
        lcall    kr_srl
        mov     A,Dt1E
        lcall    kr_srl
        mov     A,Dt1F
        lcall    kr_srl
;
        acall    delay2
        mov     SP,#07h
        ljmp    mulai
;
scnkbd: clr     EX1                ; disable external interupt 1
        mov     Cnt0,#3
scan0:  mov     Cnt1,#8
        jb     Kclk,$
        jnb    Kclk,$
        clr    A

```

```

scan1:  jb      Kclk,$
        mov     C,Kdta
        RRC    A
        jnb    Kclk,$
        djnz   Cnt1,scan1
        jb     Kclk,$
        jnb    Kclk,$
        jb     Kclk,$
        jnb    Kclk,$
        djnz   Cnt0,scan0
scan2:  cjne   A,#0E0h,scan3
        mov     Cnt0,#2
        sjmp   scan0
;
scan3:  cjne   A,#058h,scan4          ; if Caps Lock
        mov     R0,#0                 ; set data = 0
cksts0: jb     Stts,cksts1            ; \
        setb   Stts                   ; |
        ljmp   scan4                  ; |  rubah status
cksts1: jnb    Stts,cksts0           ; |
        clr    Stts                   ; /
;
scan4:  cjne   A,#077h,scan5          ; if Num Lock
;
scan5:  cjne   A,#076h,scan6          ; if esc ?
        mov     R0,#11                ; set data = 11
scan6:  cjne   A,#05Ah,scan7          ; if enter ?
        mov     R0,#12                ; set data = 12
scan7:  cjne   A,#005h,scan8          ; if F1 ?
        mov     R0,#13                ; set data = 13
scan8:  cjne   A,#066h,scan9          ; set data = 14
        mov     R0,#14
scan9:  jb     Stts,scanA
        mov     DPTR,#kbmap0          ; ambil data pointer char kecil
        ljmp   scanB
scanA:  mov     DPTR,#kbmap1          ; ambil data pointer char besar
scanB:  setb   EX1                    ; enable external interrupt 1
        reti                           ; kembali
;
        ;mov    DPTR,#angka           ; smntr
        ;mov    P0,#080h              ; smntr
        ;lcall  w_ins                  ; smntr
        ;lcall  nilai                 ; smntr
;
tgchar: clr    A
        jz     $
        cjne   R0,#00,tgchr0
        mov     R0,#10
        ljmp   tgchar
tgchr0: cjne   R0,#11,tgchr1
        mov     R0,#10
        mov     SP,#07h
        ljmp   mulai
tgchr1: cjne   R0,#12,tgchr2
        mov     R0,#10
        ljmp   isdt01
tgchr2: cjne   R0,#14,tgchr4

```

```

        mov     R0,#10
        dec     R1
        cjne   R1,#255,tgchr3
        mov     R1,#0
tgchr3: mov     SP,#07h
        ljmp   urut00
tgchr4: movc   A,@A+DPTR
        mov     R0,#10
        mov     P0,A
        lcall  w_chr
        ret

```

```

;
rstdt0: mov     Dt00,#' '
        mov     Dt01,#' '
        mov     Dt02,#' '
        mov     Dt03,#' '
        mov     Dt04,#' '
        mov     Dt05,#' '
        mov     Dt06,#' '
        mov     Dt07,#' '
        mov     Dt08,#' '
        mov     Dt09,#' '
        mov     Dt0A,#' '
        mov     Dt0B,#' '
        mov     Dt0C,#' '
        mov     Dt0D,#' '
        mov     Dt0E,#' '
        mov     Dt0F,#' '

```

```

        mov     Dt10,#' '
        mov     Dt11,#' '
        mov     Dt12,#' '
        mov     Dt13,#' '
        mov     Dt14,#' '
        mov     Dt15,#' '
        mov     Dt16,#' '
        mov     Dt17,#' '
        mov     Dt18,#' '
        mov     Dt19,#' '
        mov     Dt1A,#' '
        mov     Dt1B,#' '
        mov     Dt1C,#' '
        mov     Dt1D,#' '
        mov     Dt1E,#' '
        mov     Dt1F,#' '
        ret

```

```

;
nilai: mov     B,#100
        div     AB
        lcall  wr_chr
        mov     A,B
        mov     B,#10
        div     AB
        lcall  wr_chr
        mov     A,B
        lcall  wr_chr
        ret

```

```

;
line1:  mov     P0,#080h
        acall  w_ins
        ret

;
line2:  mov     P0,#0C0h
        acall  w_ins
        ret

;
tulis:  clr     A
        acall  wr_chr
        inc    DPTR
        djnz   Char,tulis
        ret

;
wr_chr:  movc   A,@A+DPTR
        mov    P0,A
        acall  w_chr
        ret

;
w_ins:  clr     Enbl
        clr     Rest
        setb   Enbl
        clr     Enbl
        acall  delay0
        ret

;
w_chr:  clr     Enbl
        setb   Rest
        setb   Enbl
        clr     Enbl
        acall  delay0
        ret

;
lcd_in:  mov     Dly3,#2
        acall  delay3
        mov     P0,#01h           ; Display Clear
        acall  w_ins
        mov     P0,#38h         ; Function Set
        acall  w_ins
        mov     P0,#0Eh         ; Display On, Cursor, Blink
        acall  w_ins
        mov     P0,#06h         ; Entry Mode
        acall  w_ins
        mov     P0,#02h         ; Cursor Home
        acall  w_ins
        ret

;
lcdclr:  mov     P0,#01h           ; Display Clear
        acall  w_ins
        acall  delay0
        acall  delay0
        ret

;
srl_in:  mov     Dly3,#2
        acall  delay3
        mov     TMOD,#20h

```



```

        mov     TH1,#0FDh
        mov     SCON,#50h
        setb   TR1
        setb   EX1           ; enable external interupt 1
        setb   ES
        setb   EA           ; enable all interupt
        ret

;
kr_srl:  clr     ES
        mov     SBUF,A
        jnb    TI,$
        clr     TI
        setb   ES
        lcall  delay0
        lcall  delay0
        lcall  delay0
        ret

;
delay0:  djnz   Dly0,delay0
        ret

;
delay1:  acall  delay0
        cjne   R0,#13,dely10      ; keyboard F1
        mov    R0,#10
        ljmp   isidta
dely10:  djnz   Dly1,delay1
        ret

;
delay2:  mov    Dly2,#15
dely20:  acall  delay1
        djnz   Dly2,dely20
        ret

;
delay3:  lcall  delay0
        djnz   Dly1,delay3
        djnz   Dly3,delay3
        ret

;
nama:    DB     'Paulo DD e Silva'
nim:     DB     ' NIM: 0412205 '
jur:     DB     ' T. Elektro '
univ:    DB     ' ITN Malang '
tptlps:  DB     ' Tulis Pesan '
tpkrdt:  DB     ' Kirim Data '
angka:  DB     '0123456789 '

;
kbmap0:  DB     '           ' ;000-009
        DB     '           ' ;010-019
        DB     ' q1  zsaw'  ;020-029
        DB     '2  cxde43 ' ;030-039
        DB     ' vftr5 n'  ;040-049
        DB     'bhgy6  mj'  ;050-059
        DB     'u78  ,kio0' ;060-069
        DB     '9  ./l;p- ' ;070-079
        DB     ' [= '      ;080-089
        DB     ' ] '      ;090-099
        DB     ' 1 47 '    ;100-109

```

```

DB      ' 0 2568 ' ;110-119
DB      ' 3 9 ' ;120-129
DB      ' ' ;130-139
DB      ' ' ;140-149
DB      ' ' ;150-159
DB      ' ' ;160-169
DB      ' ' ;170-179
DB      ' ' ;180-189
DB      ' ' ;190-199
DB      ' ' ;200-209
DB      ' ' ;210-219
DB      ' ' ;220-229
DB      ' ' ;230-239
DB      ' ' ;240-249
DB      ' ' ;250-255
;
kbmap1: DB      ' ' ;000-009
DB      ' ' ;010-019
DB      ' Q1 ZSAW ' ;020-029
DB      ' 2 CXDE43 ' ;030-039
DB      ' VFTR5 N ' ;040-049
DB      ' BHGY6 MJ ' ;050-059
DB      ' U78 ,KIOO ' ;060-069
DB      ' 9 ./L:P- ' ;070-079
DB      ' [= ' ;080-089
DB      ' ] ' ;090-099
DB      ' 1 47 ' ;100-109
DB      ' 0 2568 ' ;110-119
DB      ' 3 9 ' ;120-129
DB      ' ' ;130-139
DB      ' ' ;140-149
DB      ' ' ;150-159
DB      ' ' ;160-169
DB      ' ' ;170-179
DB      ' ' ;180-189
DB      ' ' ;190-199
DB      ' ' ;200-209
DB      ' ' ;210-219
DB      ' ' ;220-229
DB      ' ' ;230-239
DB      ' ' ;240-249
DB      ' ' ;250-255
;
end

```

**LISTENING PROGRAM  
FOR MCU II**

### Listening Program for Dot Matrix

```
org    00h
ljmp   init
;
org    23h
clr    ES
jnb    RI,$
clr    RI
mov    A,SBUF
mov    R7,A
setb   ES
reti
;
Sdta   Bit P3.3
Sclk   Bit P3.4
Slct   Bit P3.5
Jchr   Equ 30h           ; jumlah character ascii
Jcd0   Equ 31h           ; jumlah colom dot low
Jcd1   Equ 32h           ; jumlah colom dot high
Cnt0   Equ 33h
Cnt1   Equ 34h
Cnt2   Equ 35h
Cnt3   Equ 36h
DpcL   Equ 37h           ; data pointer character low
DpcH   Equ 38h           ; data pointer character high
DpdL   Equ 39h           ; data pointer dot low
DpdH   Equ 3Ah           ; data pointer dot high
Buf0   Equ 3Bh
Buf1   Equ 3Ch
Buf2   Equ 3Dh
Buf3   Equ 3Eh
Buf4   Equ 3Fh
Buf5   Equ 40h
Jrkn   Equ 41h
Dly0   Equ 50h
Dly1   Equ 51h
Dly2   Equ 52h
;
init:  mov    P1,#0FFh      ; clr banner
       lcall  delay2
       lcall  srl_in
       lcall  rstcmd
       clr    Slct         ; memory active
;
       lcall  clrldot
mulai: mov    Jchr,#32      ; jumlah character ascii
       mov    DpcL,#00h    ; ambil dari data pointer
       mov    DpcH,#00h    ; address 0000h
       lcall  cchchr       ; cacah character
zzz:   mov    Jcd0,#024h   ; jalankan terus menerus
       mov    Jcd1,#001h   ; dalam 292 colom dot matrix led
       lcall  tljln1       ; oke
       ljmp   zzz
;
bc_srl: mov   Cnt0,#32
        mov   DPTR,#00h
        lcall tg_srl
```

```

bcsr10: movx    @DPTR,A
        lcall   wt_wr
        inc    DPTR
        djnz   Cnt0,bcsr11
        lcall   rstcmd
        mov    SP,#07h
        ljmp   mulai
bcsr11: lcall   tg_srl
        ljmp   bcsr10
;
clock:  setb   Sclk
        clr    Sclk
        ret
;
tlj1n1: mov    DpdL,#100           ; jalan terus
        mov    DpdH,#000
        mov    Cnt1,#0
        mov    Cnt2,#8
tlj110: mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis0
        mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis1
        mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis2
        mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis3
        mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis4
        mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis5
        mov    DPL,DpdL
        mov    DPH,DpdH
        mov    Cnt0,#80
        lcall   tulis6
        djnz   Cnt2,tlj112
        mov    Cnt2,#8
        lcall   gsrdot
        mov    A,Jcd1
        jz     tlj111
        djnz   Cnt1,tlj112
        djnz   Jcd1,tlj112
tlj111: mov    A,Jcd0
        jz     tlj115
        djnz   Jcd0,tlj112
        ljmp   tlj115
tlj112: cjne   R7,#0FFh,tlj113

```

```

        ljmp      t1j114
t1j113: ljmp      bc_srl
t1j114: ljmp      t1j110
t1j115: ret
;
gsrdot: inc       DpdL
        mov       A,DpdL
        jnz       gsrdt
        inc       DpdH
gsrdt:  ret
;
tulis0: mov       P1,#01111111b
        movx      A,@DPTR
        mov       C,Acc.1
        cpl       C
        mov       Sdta,C
        lcall     clock
        inc       DPTR
        djnz      Cnt0,tulis0
        mov       P1,#10111111b
        lcall     delay1
        mov       P1,#11111111b
        ret
;
tulis1: mov       P1,#01111111b
        movx      A,@DPTR
        mov       C,Acc.2
        cpl       C
        mov       Sdta,C
        lcall     clock
        inc       DPTR
        djnz      Cnt0,tulis1
        mov       P1,#11011111b
        lcall     delay1
        mov       P1,#11111111b
        ret
;
tulis2: mov       P1,#01111111b
        movx      A,@DPTR
        mov       C,Acc.3
        cpl       C
        mov       Sdta,C
        lcall     clock
        inc       DPTR
        djnz      Cnt0,tulis2
        mov       P1,#11101111b
        lcall     delay1
        mov       P1,#11111111b
        ret
;
tulis3: mov       P1,#01111111b
        movx      A,@DPTR
        mov       C,Acc.4
        cpl       C
        mov       Sdta,C
        lcall     clock
        inc       DPTR
        djnz      Cnt0,tulis3

```

```

        mov     P1,#11110111b
        lcall  delay1
        mov     P1,#11111111b
        ret

;
tulis4: mov     P1,#01111111b
        movx   A,@DPTR
        mov     C,Acc.5
        cpl    C
        mov     Sdta,C
        lcall  clock
        inc    DPTR
        djnz   Cnt0,tulis4
        mov     P1,#11110111b
        lcall  delay1
        mov     P1,#11111111b
        ret

;
tulis5: mov     P1,#01111111b
        movx   A,@DPTR
        mov     C,Acc.6
        cpl    C
        mov     Sdta,C
        lcall  clock
        inc    DPTR
        djnz   Cnt0,tulis5
        mov     P1,#11111011b
        lcall  delay1
        mov     P1,#11111111b
        ret

;
tulis6: mov     P1,#01111111b
        movx   A,@DPTR
        mov     C,Acc.7
        cpl    C
        mov     Sdta,C
        lcall  clock
        inc    DPTR
        djnz   Cnt0,tulis6
        mov     P1,#11111110b
        lcall  delay1
        mov     P1,#11111111b
        ret

;
cchchr: mov     DpdL,#180
        mov     DpdH,#000
        mov     DPL,DpcL
        mov     DPH,Dpch
cchch:  movx   A,@DPTR
        lcall  chrdot
        mov     DPL,DpcL
        mov     DPH,Dpch
        inc    DPTR
        mov     DpcL,DPL
        mov     Dpch,DPH
        djnz   Jchr,cchch
        ret
;

```

```

chrdot: mov     DPTR,#chrksg
        cjne   A,#'A',chdt00
        mov   DPTR,#charAb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt00: cjne   A,#'B',chdt01
        mov   DPTR,#charBb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt01: cjne   A,#'C',chdt02
        mov   DPTR,#charCb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt02: cjne   A,#'D',chdt03
        mov   DPTR,#charDb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt03: cjne   A,#'E',chdt04
        mov   DPTR,#charEb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt04: cjne   A,#'F',chdt05
        mov   DPTR,#charFb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt05: cjne   A,#'G',chdt06
        mov   DPTR,#charGb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt06: cjne   A,#'H',chdt07
        mov   DPTR,#charHb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt07: cjne   A,#'I',chdt08
        mov   DPTR,#charIb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt08: cjne   A,#'J',chdt09
        mov   DPTR,#charJb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt09: cjne   A,#'K',chdt10
        mov   DPTR,#charKb
        lcall ambdta
        lcall tismem
        ljmp  chdt99
chdt10: cjne   A,#'L',chdt11
        mov   DPTR,#charLb

```



```

        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt11: cjne    A, #'M', chdt12
        mov     DPTR, #charMb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt12: cjne    A, #'N', chdt13
        mov     DPTR, #charNb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt13: cjne    A, #'O', chdt14
        mov     DPTR, #charOb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt14: cjne    A, #'P', chdt15
        mov     DPTR, #charPb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt15: cjne    A, #'Q', chdt16
        mov     DPTR, #charQb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt16: cjne    A, #'R', chdt17
        mov     DPTR, #charRb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt17: cjne    A, #'S', chdt18
        mov     DPTR, #charSb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt18: cjne    A, #'T', chdt19
        mov     DPTR, #charTb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt19: cjne    A, #'U', chdt20
        mov     DPTR, #charUb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt20: cjne    A, #'V', chdt21
        mov     DPTR, #charVb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt21: cjne    A, #'W', chdt22
        mov     DPTR, #charWb
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99

```

```

chdt22: cjne    A, #'X', chdt23
        mov     DPTR, #charXb
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt23: cjne    A, #'Y', chdt24
        mov     DPTR, #charYb
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt24: cjne    A, #'Z', chdt25
        mov     DPTR, #charZb
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
;
chdt25: cjne    A, #'a', chdt26
        mov     DPTR, #charak
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt26: cjne    A, #'b', chdt27
        mov     DPTR, #charbk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt27: cjne    A, #'c', chdt28
        mov     DPTR, #charck
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt28: cjne    A, #'d', chdt29
        mov     DPTR, #chardk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt29: cjne    A, #'e', chdt30
        mov     DPTR, #charek
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt30: cjne    A, #'f', chdt31
        mov     DPTR, #charfk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt31: cjne    A, #'g', chdt32
        mov     DPTR, #chargk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt32: cjne    A, #'h', chdt33
        mov     DPTR, #charhk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt33: cjne    A, #'i', chdt34
        mov     DPTR, #charik

```

```

        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt34: cjne    A,#'j',chdt35
        mov     DPTR,#charjk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt35: cjne    A,#'k',chdt36
        mov     DPTR,#charkk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt36: cjne    A,#'l',chdt37
        mov     DPTR,#charlk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt37: cjne    A,#'m',chdt38
        mov     DPTR,#charm k
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt38: cjne    A,#'n',chdt39
        mov     DPTR,#charnk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt39: cjne    A,#'o',chdt40
        mov     DPTR,#charok
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt40: cjne    A,#'p',chdt41
        mov     DPTR,#charpk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt41: cjne    A,#'q',chdt42
        mov     DPTR,#charqk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt42: cjne    A,#'r',chdt43
        mov     DPTR,#charrk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt43: cjne    A,#'s',chdt44
        mov     DPTR,#charsk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt44: cjne    A,#'t',chdt45
        mov     DPTR,#chartk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99

```

```

chdt45: cjne    A, #'u', chdt46
        mov     DPTR, #charuk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt46: cjne    A, #'v', chdt47
        mov     DPTR, #charvk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt47: cjne    A, #'w', chdt48
        mov     DPTR, #charwk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt48: cjne    A, #'x', chdt49
        mov     DPTR, #charxk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt49: cjne    A, #'y', chdt50
        mov     DPTR, #charyk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt50: cjne    A, #'z', chdt51
        mov     DPTR, #charzk
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99

chdt51: cjne    A, #'0', chdt52
        mov     DPTR, #charn0
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt52: cjne    A, #'1', chdt53
        mov     DPTR, #charn1
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt53: cjne    A, #'2', chdt54
        mov     DPTR, #charn2
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt54: cjne    A, #'3', chdt55
        mov     DPTR, #charn3
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt55: cjne    A, #'4', chdt56
        mov     DPTR, #charn4
        lcall  ambdta
        lcall  tismem
        ljmp   chdt99
chdt56: cjne    A, #'5', chdt57
        mov     DPTR, #charn5

```

```

        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt57: cjne    A,#'6',chdt58
        mov     DPTR,#charn6
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt58: cjne    A,#'7',chdt59
        mov     DPTR,#charn7
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt59: cjne    A,#'8',chdt60
        mov     DPTR,#charn8
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt60: cjne    A,#'9',chdt61
        mov     DPTR,#charn9
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99

chdt61: cjne    A,#':',chdt62
        mov     DPTR,#chrtrt2
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt62: cjne    A,#'/',chdt63
        mov     DPTR,#chrgrm
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99

chdt63: cjne    A,#'.',chdt64
        mov     DPTR,#chrtrtk
        lcall    ambdta
        lcall    tismem
        ljmp     chdt99
chdt64: cjne    A,#' ',chdt65
        mov     DPTR,#chrsp
        lcall    ambdta
        lcall    tismem
;-----
chdt65: cjne    A,#',',chdt66
        mov     DPTR,#chrkom
        lcall    ambdta
        lcall    tismem
chdt66: cjne    A,#';',chdt67
        mov     DPTR,#chrtrkm
        lcall    ambdta
        lcall    tismem
chdt67: cjne    A,#'[',chdt68
        mov     DPTR,#chrbrk
        lcall    ambdta
        lcall    tismem
chdt68: cjne    A,#']',chdt69

```

```

        mov     DPTR,#chrtpr
        lcall  ambdta
        lcall  tismem
chdt69: cjne   A,#'=',chdt70
        mov     DPTR,#chrsdg
        lcall  ambdta
        lcall  tismem
chdt70: cjne   A,#'-',chdt99
        mov     DPTR,#chrstr
        lcall  ambdta
        lcall  tismem

```

```

;-----
chdt99: ret
;

```

```

ambdta: clr     A
        movc   A,@A+DPTR
        mov    Buf0,A
        inc   DPTR
        clr   A
        movc  A,@A+DPTR
        mov    Buf1,A
        inc   DPTR
        clr   A
        movc  A,@A+DPTR
        mov    Buf2,A
        inc   DPTR
        clr   A
        movc  A,@A+DPTR
        mov    Buf3,A
        inc   DPTR
        clr   A
        movc  A,@A+DPTR
        mov    Buf4,A
        inc   DPTR
        clr   A
        movc  A,@A+DPTR
        mov    Buf5,A
        ret

```

```

;
tismem: mov     DPL,DpdL
        mov     DPH,DpdH
        mov     A,Buf0
        movx    @DPTR,A
        lcall   wt_wr
        inc     DPTR
        mov     A,Buf1
        movx    @DPTR,A
        lcall   wt_wr
        inc     DPTR
        mov     A,Buf2
        movx    @DPTR,A
        lcall   wt_wr
        inc     DPTR
        mov     A,Buf3
        movx    @DPTR,A
        lcall   wt_wr
        inc     DPTR
        mov     A,Buf4

```

```

        movx    @DPTR,A
        lcall   wt_wr
        inc     DPTR
        mov     A,Buf5
        movx    @DPTR,A
        lcall   wt_wr
        inc     DPTR
        mov     DpdL,DPL
        mov     DpdH,DPH
        ret

;
clrldot: mov     DPTR,#100           ; clear colom dot matrix
          mov     Cnt0,#4
          mov     Cnt1,#80
cldot:   mov     A,#0
          movx    @DPTR,A
          lcall   wt_wr
          inc     DPTR
          djnz   Cnt1,cldot
          djnz   Cnt0,cldot
          ret

;
wt_wr:   lcall   delay0
          lcall   delay0
          lcall   delay0
          ret

;
srl_in:  lcall   delay1
          mov     TMOD,#20h
          mov     TH1,#0FDh
          mov     SCON,#50h
          setb   TR1
          setb   ES
          setb   EA
          ret

;
tg_srl:  mov     R7,#0FFh
tgsrl0:  cjne   R7,#0FFh,tgsrl1
          ljmp   tgsrl0
tgsrl1:  ret

;
rstcmd:  mov     R7,#0FFh
          ret

;
delay0:  djnz   Dly0,delay0
          ret

;
delay1:  mov     Dly1,#3
dely10:  lcall   delay0
          djnz   Dly1,dely10
          ret

;
delay2:  mov     Dly2,#5
dely20:  lcall   delay0
          djnz   Dly1,dely20
          djnz   Dly2,dely20
          ret

;

```

charAb:	DB	07Eh,090h,090h,090h,07Eh,000h
charBb:	DB	0FEh,092h,092h,092h,06Ch,000h
charCb:	DB	07Ch,082h,082h,082h,044h,000h
charDb:	DB	0FEh,082h,082h,044h,038h,000h
charEb:	DB	0FEh,092h,092h,092h,082h,000h
charFb:	DB	0FEh,090h,090h,090h,080h,000h
charGb:	DB	07Ch,082h,092h,092h,05Ch,000h
charHb:	DB	0FEh,010h,010h,010h,0FEh,000h
charIb:	DB	000h,082h,0FEh,082h,000h,000h
charJb:	DB	004h,002h,082h,0FCh,080h,000h
charKb:	DB	0FEh,010h,028h,044h,082h,000h
charLb:	DB	0FEh,002h,002h,002h,002h,000h
charMb:	DB	0FEh,040h,030h,040h,0FEh,000h
charNb:	DB	0FEh,020h,010h,008h,0FEh,000h
charOb:	DB	07Ch,082h,082h,082h,07Ch,000h
charPb:	DB	0FEh,090h,090h,090h,060h,000h
charQb:	DB	07Ch,082h,08Ah,084h,07Ah,000h
charRb:	DB	0FEh,090h,098h,094h,062h,000h
charSb:	DB	064h,092h,092h,092h,04Ch,000h
charTb:	DB	080h,080h,0FEh,080h,080h,000h
charUb:	DB	0FCh,002h,002h,002h,0FCh,000h
charVb:	DB	0F8h,004h,002h,004h,0F8h,000h
charWb:	DB	0FCh,002h,01Ch,002h,0FCh,000h
charXb:	DB	0C6h,028h,010h,028h,0C6h,000h
charYb:	DB	0E0h,010h,00Eh,010h,0E0h,000h
charZb:	DB	086h,08Ah,092h,0A2h,0C2h,000h
charak:	DB	004h,02Ah,02Ah,02Ah,01Eh,000h
charbk:	DB	0FEh,012h,022h,022h,01Ch,000h
charck:	DB	01Ch,022h,022h,022h,004h,000h
chardk:	DB	01Ch,022h,022h,012h,0FEh,000h
charek:	DB	01Ch,02Ah,02Ah,02Ah,018h,000h
charfk:	DB	010h,07Eh,090h,080h,040h,000h
chargk:	DB	010h,02Ah,02Ah,02Ah,03Ch,000h
charhk:	DB	0FEh,010h,020h,020h,01Eh,000h
charik:	DB	000h,022h,0BEh,002h,000h,000h
charjk:	DB	004h,002h,022h,0BCh,000h,000h
charkk:	DB	0FEh,008h,014h,022h,000h,000h
charlk:	DB	000h,082h,0FEh,002h,000h,000h
charm:	DB	03Eh,020h,018h,020h,01Eh,000h
charnk:	DB	03Eh,010h,020h,020h,01Eh,000h
charok:	DB	01Ch,022h,022h,022h,01Ch,000h
charpk:	DB	03Eh,028h,028h,028h,010h,000h
charqk:	DB	010h,028h,028h,018h,03Eh,000h
charrk:	DB	03Eh,010h,020h,020h,010h,000h
charsk:	DB	012h,02Ah,02Ah,02Ah,024h,000h
chartk:	DB	020h,0FCh,022h,002h,004h,000h
charuk:	DB	03Ch,002h,002h,004h,03Eh,000h
charvk:	DB	038h,004h,002h,004h,038h,000h
charwk:	DB	03Ch,002h,004h,002h,03Ch,000h
charxk:	DB	022h,014h,008h,014h,022h,000h
charyk:	DB	030h,00Ah,00Ah,00Ah,03Ch,000h
charzk:	DB	022h,026h,02Ah,032h,022h,000h
charn0:	DB	07Ch,082h,082h,082h,07Ch,000h
charn1:	DB	000h,042h,0FEh,002h,000h,000h
charn2:	DB	042h,086h,08Ah,092h,062h,000h
charn3:	DB	084h,082h,0A2h,0D2h,08Ch,000h
charn4:	DB	018h,028h,048h,0FEh,008h,000h



```

charn5: DB      0E4h,0A2h,0A2h,0A2h,09Ch,000h
charn6: DB      03Ch,052h,092h,092h,00Ch,000h
charn7: DB      080h,080h,09Eh,0A0h,0C0h,000h
charn8: DB      06Ch,092h,092h,092h,06Ch,000h
charn9: DB      060h,092h,092h,094h,078h,000h

chrttp2: DB     000h,000h,024h,000h,000h,000h      ; titik dua
chrgrm: DB     004h,008h,010h,020h,040h,000h      ; garis miring
chrttpk: DB     000h,000h,002h,000h,000h,000h      ; titik

chrkom: DB     000h,00Ah,00Ch,000h,000h,000h      ; koma
chrtpk: DB     000h,06Ah,06Ch,000h,000h,000h      ; titik koma
chrbrk: DB     000h,0FEh,082h,082h,000h,000h      ; buka kurung
chrtp: DB      000h,082h,082h,0FEh,000h,000h      ; tutup kurung
chrtdg: DB     028h,028h,028h,028h,028h,000h      ; sama dengan
chrstr: DB     010h,010h,010h,010h,010h,000h      ; strip

chrsp: DB      000h,000h,000h,000h,000h,000h      ; spasi
chrks: DB      000h,000h,000h,000h,000h,000h      ; kosong
;
kbmap0: DB      ' q1  zsaw' ;020-029
DB              '2  cxde43 ' ;030-039
DB              ' vftr5  n' ;040-049
DB              'bhgy6  mj' ;050-059
DB              'u78   ,kio0' ;060-069
DB              '9   ./l;p- ' ;070-079
DB              '    [=    ' ;080-089
DB              '   ]    ' ;090-099
DB              '    1 47 ' ;100-109
DB              '   0 2568 ' ;110-119
DB              '   3  9   ' ;120-129
;
kbmap1: DB      ' Q1  ZSAW' ;020-029
DB              '2  CXDE43 ' ;030-039
DB              ' VFTR5  N' ;040-049
DB              'BHGY6  MJ' ;050-059
DB              'U78   ,KIO0' ;060-069
DB              '9   ./L:P- ' ;070-079
DB              '    [=    ' ;080-089
DB              '   ]    ' ;090-099
DB              '    1 47 ' ;100-109
DB              '   0 2568 ' ;110-119
DB              '   3  9   ' ;120-129
;
end

```

**DATASHEET ATMEL**  
**AT89S52**

## Features

Compatible with MCS-51® Products  
8K Bytes of In-System Programmable (ISP) Flash Memory  
Endurance: 1000 Write/Erase Cycles  
3 to 5.5V Operating Range  
Static Operation: 0 Hz to 33 MHz  
Two-level Program Memory Lock  
64 Bytes of 8-bit Internal RAM  
8 Programmable I/O Lines  
Three 16-bit Timer/Counters  
Eight Interrupt Sources  
Full Duplex UART Serial Channel  
Power Idle and Power-down Modes  
Rapid Recovery from Power-down Mode  
Watchdog Timer  
Data Pointer  
Power-off Flag

## Description

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory. The device is manufactured using high-density nonvolatile memory technology and is compatible with the industry standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a flexible and cost-effective solution to many embedded control applications.

The AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes of internal RAM, 32 I/O lines, Watchdog timer, two data pointers, three 16-bit timer/counters, a full duplex serial port, on-chip oscillator, and low-power circuitry. In addition, the AT89S52 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 content but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset.



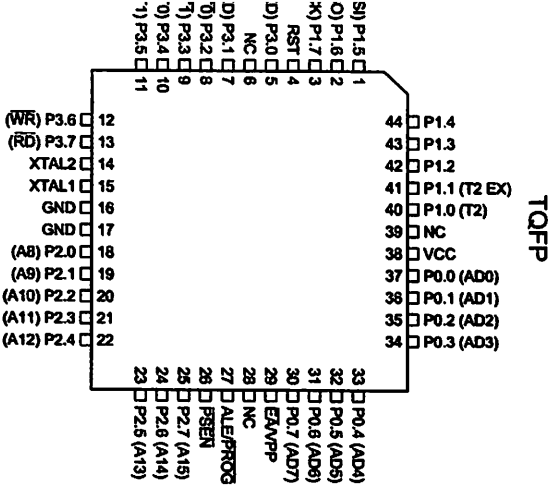
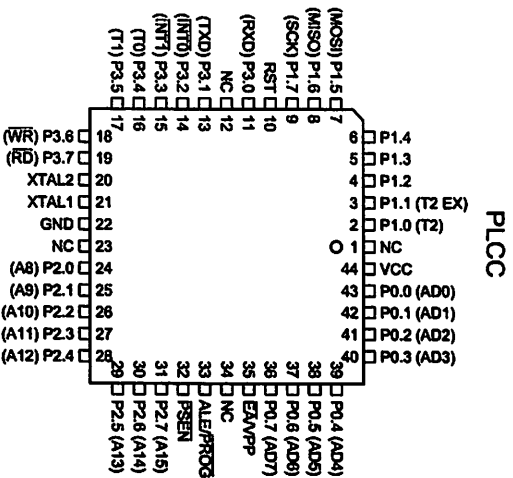
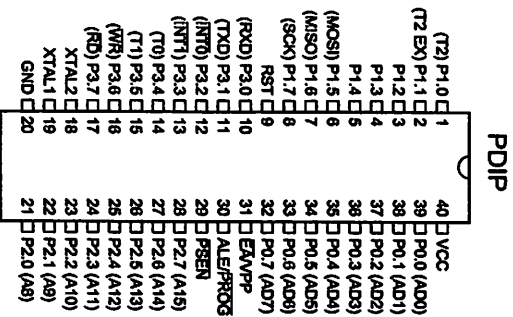
## 8-bit Microcontroller with 8K Bytes In-System Programmable Flash

### AT89S52

Rev. 1919A-07/01

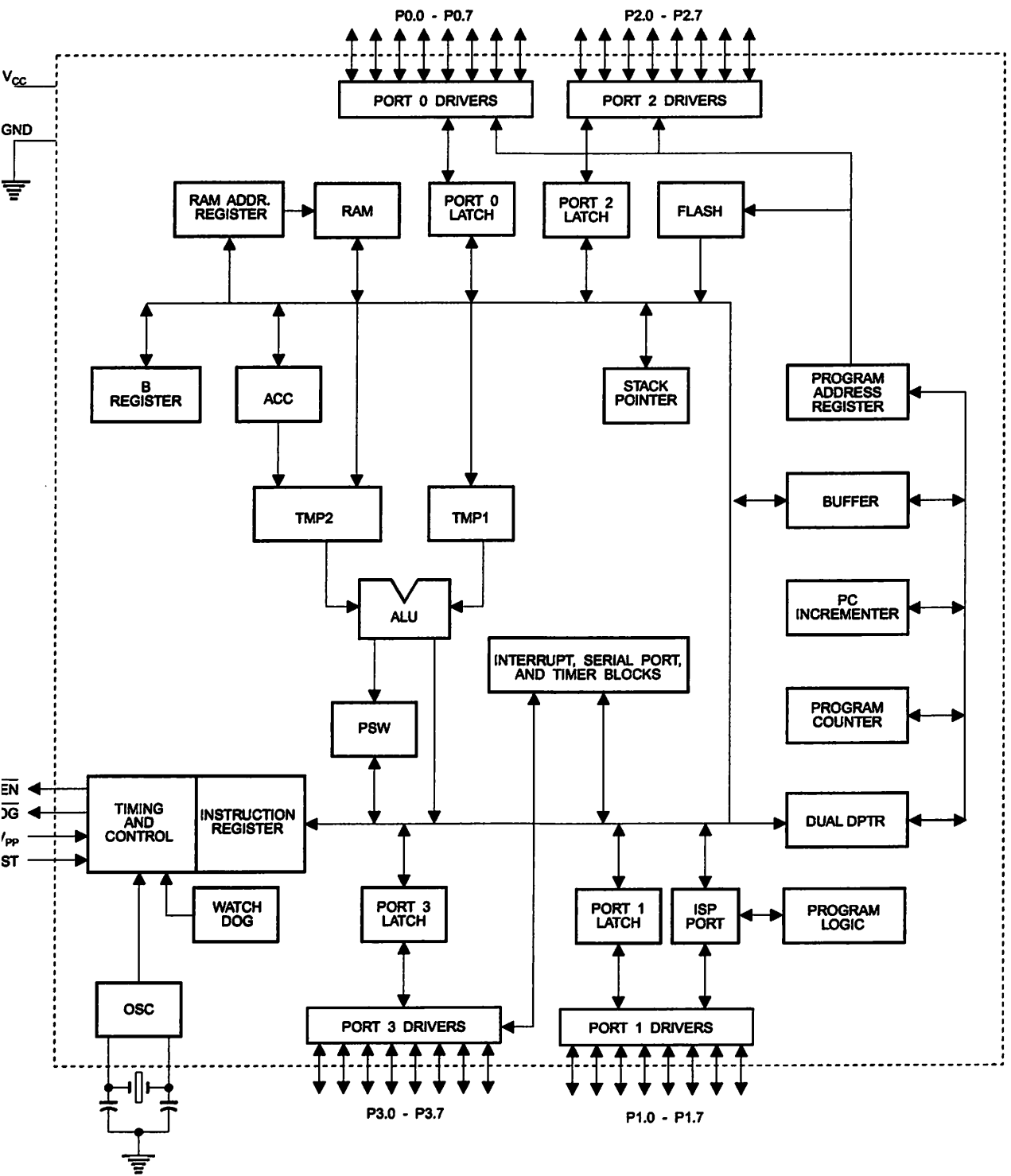


# Configurations



# AT89S52

Block Diagram





## Description

voltage.

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

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

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

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

In addition, P1.0 and P1.1 can be configured to be the Counter 2 external count input (P1.0/T2) and the Counter 2 trigger input (P1.1/T2EX), respectively, as shown in the following table.

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

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)
P1.2	MOSI (used for In-System Programming)
P1.3	MISO (used for In-System Programming)
P1.4	SCK (used for In-System Programming)

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

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to

external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

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

### Port 3

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

Port 3 also serves the functions of various special features of the AT89S52, as shown in the following table.

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

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

### RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives High for 96 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

### ALE/PROG

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

pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

Program Store Enable ( $\overline{\text{PSEN}}$ ) is the read strobe to external program memory.

When the AT89S52 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 access to external data memory.

Parallel 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.

## 4. AT89S52 SFR Map and Reset Values

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  $V_{\text{CC}}$  for internal program executions.

This pin also receives the 12-volt programming enable voltage ( $V_{\text{PP}}$ ) during Flash programming.

### XTAL1

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

### XTAL2

Output from the inverting oscillator amplifier.

								0FFH
B 00000000								0F7H
								0EFH
ACC 00000000								0E7H
								0DFH
PSW 00000000								0D7H
T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
								0C7H
IP XX000000								0BFH
P3 11111111								0B7H
IE 0X000000								0AFH
P2 11111111		AUXR1 XXXXXXX0					WDTRST XXXXXXX	0A7H
SCON 00000000	SBUF XXXXXXXX							9FH
P1 11111111								97H
TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR XXX0XX0		8FH
P0 11111111	SP 00001111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000		PCON 0XXX0000	87H



## Special Function Registers

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

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 0 data, and write accesses will have an indeterminate effect.

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 3) 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.

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

### T2CON – Timer/Counter 2 Control Register

T2CON Address = 0C8H		Reset Value = 0000 000B					
Addressable							
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	$C/\overline{T2}$	$CP/\overline{RL2}$
7	6	5	4	3	2	1	0
<b>Function</b>							
Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.							
Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).							
Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.							
Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.							
Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.							
Start/Stop control for Timer 2. TR2 = 1 starts the timer.							
Timer or counter select for Timer 2. $C/\overline{T2}$ = 0 for timer function. $C/\overline{T2}$ = 1 for external event counter (falling edge triggered).							
Capture/Reload select. $CP/\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. $CP/\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.							



## a. AUXR: Auxiliary Register

Address = 8EH

Reset Value = XXX00XX0B

Not Bit Addressable

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

Reserved for future expansion

Disable/Enable ALE

DISALE      Operating Mode

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

1            ALE is active only during a MOVX or MOVC instruction

Disable/Enable Reset out

DISRTO

0            Reset pin is driven High after WDT times out

1            Reset pin is input only

Disable/Enable WDT in IDLE mode

WDIDLE

0            WDT continues to count in IDLE mode

1            WDT halts counting in IDLE mode

**Data Pointer Registers:** To facilitate accessing both internal and external data memory, two banks of 16-bit Data Pointer Registers are provided: DP0 at SFR address 82H-83H and DP1 at 84H-85H. Bit DPS = 0 in AUXR1 selects DP0 and DPS = 1 selects DP1. 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.

## b. AUXR1: Auxiliary Register 1

Address = A2H

Reset Value = XXXXXXX0B

Not Bit Addressable

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

Reserved for future expansion

Data Pointer Register Select

DPS

0            Selects DPTR Registers DP0L, DP0H

1            Selects DPTR Registers DP1L, DP1H





## Memory Organization

AT89S52 devices have a separate address space for Program and Data Memory. Up to 64K bytes each of external Program and Data Memory can be addressed.

### Program Memory

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

When the  $\overline{EA}$  pin is connected to  $V_{CC}$ , program fetches to addresses 0000H through 1FFFH are directed to on-chip memory and fetches to addresses 2000H through 7FFFH are directed to external memory.

### Data Memory

The AT89S52 implements 256 bytes of on-chip RAM. The lower 128 bytes occupy a parallel address space to the Function Registers. This means that 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 which use direct addressing access the 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.

**Watchdog Timer (Time Enabled with Reset-out)**

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upsets. The WDT consists of a 13-bit counter and the Watchdog Timer (WDRST) SFR. The WDT is defaulted to disable counting until reset. To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDRST register (SFR location 0A6H). When the WDT is enabled, it will increment every machine cycle while the oscillator is running. The WDT timeout period is dependent on the external oscillator frequency. There is no way to disable the WDT through reset (either hardware reset or WDT overflow). When WDT overflows, it will drive an output pin HIGH pulse at the RST pin.

**Service the WDT**

To service the WDT, a user must write 01EH and 0E1H in sequence to the WDRST register (SFR location 0A6H). When the WDT is enabled, the user needs to service it by writing 01EH and 0E1H to WDRST to avoid a WDT overflow. The 13-bit counter overflows when it reaches 8191 machine cycles, and this will reset the device. When the WDT is enabled, it will increment every machine cycle while the oscillator is running. This means the user must reset the WDT at least every 8191 machine cycles. To reset the WDT, the user must write 01EH and 0E1H to WDRST. WDRST is a write-only register. The WDT counter cannot be read or written. When WDT overflows, it will generate an output pulse at the RST pin. The RESET pulse width is  $30 \times T_{OSC}$ , where  $T_{OSC} = 1/F_{OSC}$ . To make the most use of the WDT, it should be serviced in those sections of code that will periodically be executed within the time required to prevent a WDT reset.

**WDT During Power-down and Idle**

When the device enters Power-down mode the oscillator stops, which means the WDT also stops. While in Power-down mode, the user does not need to service the WDT. There are two methods of exiting Power-down mode: by a hardware reset or via a software reset. When Power-down is exited through hardware reset, servicing the WDT should occur as it normally does whenever the AT89S52 is reset. Exiting Power-down with an interrupt is significantly different. The interrupt pin is held low long enough for the oscillator to stabilize. When the interrupt is brought high, the interrupt is serviced. To prevent the WDT from resetting the device, the interrupt pin is held low, the WDT is not started, and the interrupt is pulled high. It is suggested that the user service the reset during the interrupt service for the interrupt to exit Power-down mode.

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

Before going into the IDLE mode, the WDIDLE bit in SFR AUXR is used to determine whether the WDT continues to count if enabled. The WDT keeps counting during IDLE (WDIDLE bit = 0) as the default state. To prevent the WDT from resetting the AT89S52 while in IDLE mode, the user should always set up a timer that will periodically exit IDLE, service the WDT, and reenter IDLE mode.

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

**UART**

The UART in the AT89S52 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 '8051-Architecture Flash Microcontroller', then 'Product Overview'.

**Timer 0 and 1**

Timer 0 and Timer 1 in the AT89S52 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 '8051-Architecture Flash Microcontroller', then 'Product Overview'.

**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/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 3. 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.

**Table 3. 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)



Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input, T2EX. In this function, the external input is sampled during S3P2 of every machine cycle. When the samples are high in one cycle and a low in the next cycle, the counter is incremented. The new count value appears in the registers 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 two machine cycles.

### Capture Mode

In capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer which upon overflow sets bit TF2 in T2CON.

### 5. Timer in Capture Mode

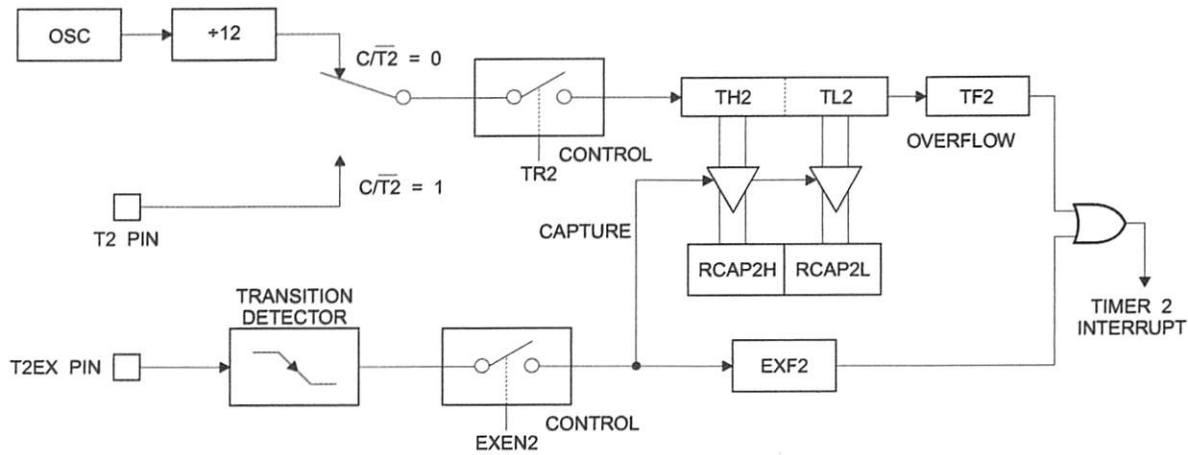


Figure 6 shows Timer 2 automatically counting up when EXEN2 = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0xFFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in RCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered 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.

The DCEN bit enables Timer 2 to count up or down, as shown in Figure 6. In this mode, the T2EX pin controls

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 5.

### Auto-reload (Up or Down Counter)

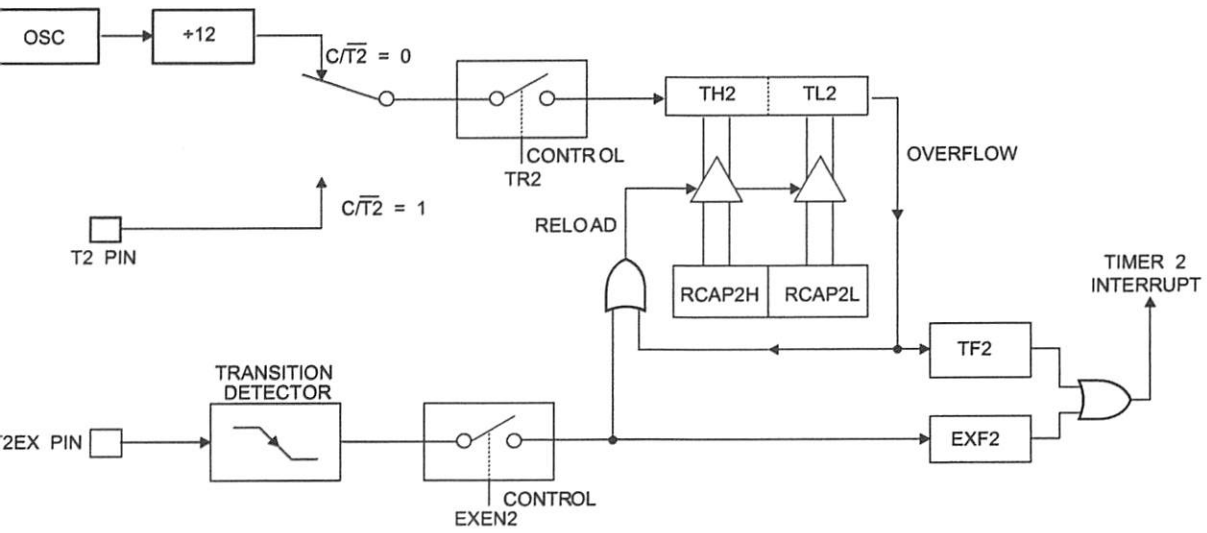
Timer 2 can be programmed to count up or down when configured in its 16-bit auto-reload mode. This feature is invoked by the DCEN (Down Counter Enable) bit located in the SFR T2MOD (see Table 4). Upon reset, the DCEN bit is set to 0 so that timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down, depending on the value of the T2EX pin.

the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0xFFFFH 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 0xFFFFH 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.

## 6. Timer 2 Auto Reload Mode (DCEN = 0)



## 7. T2MOD – Timer 2 Mode Control Register

Register Address = 0C9H

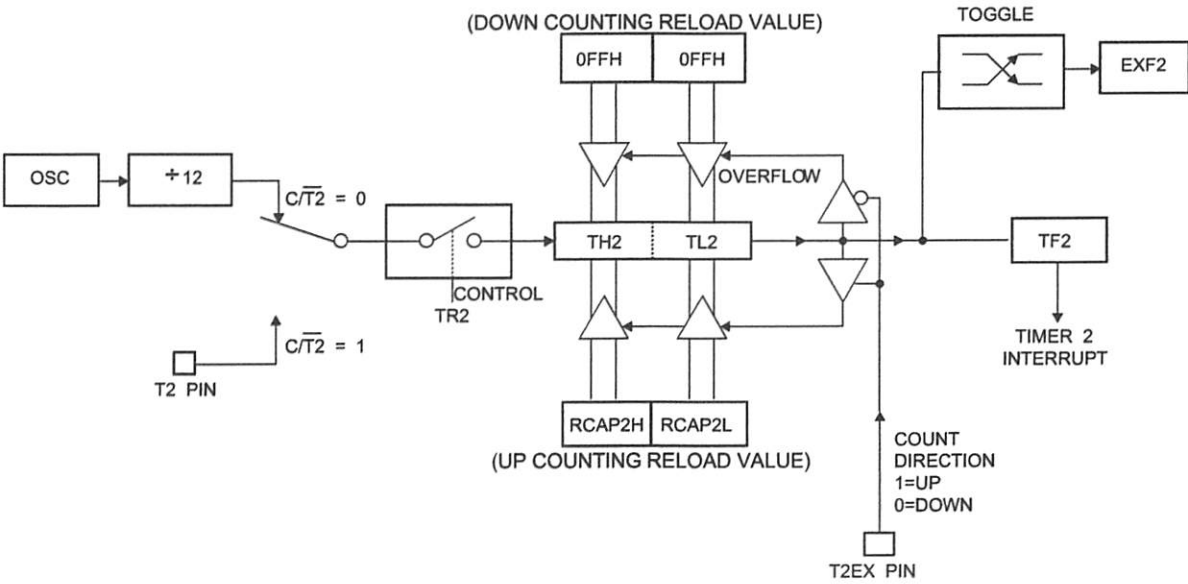
Reset Value = XXXX XX00B

Not Bit Addressable

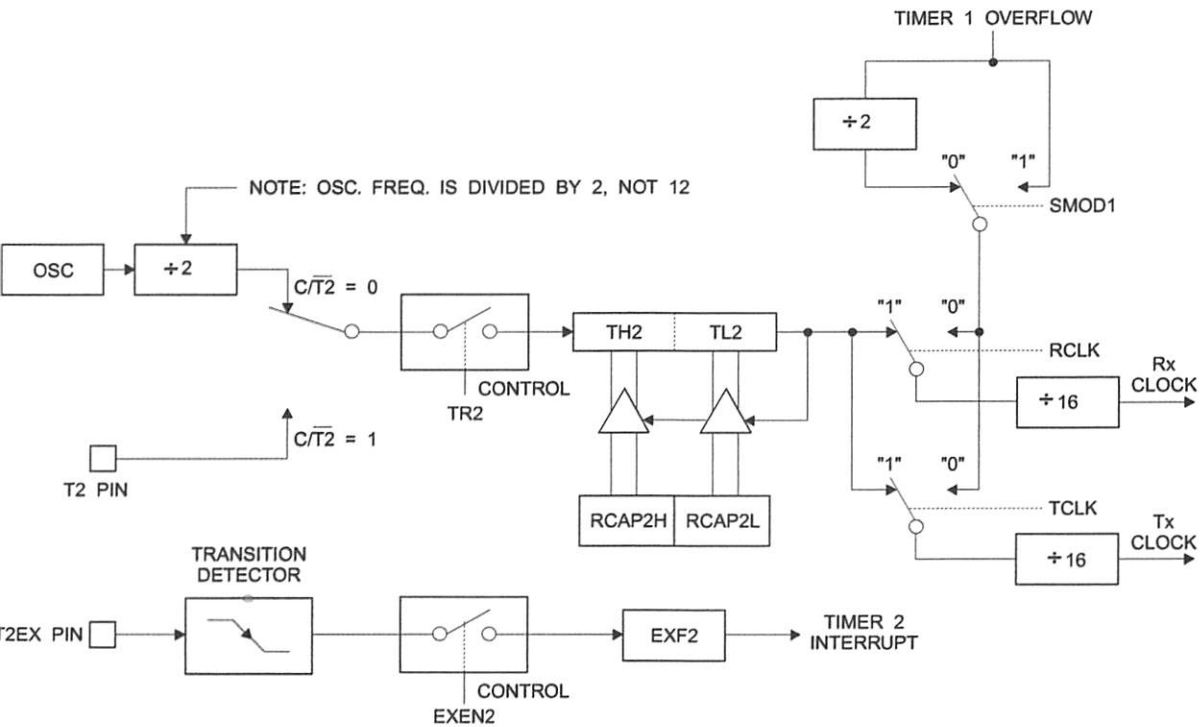
-	-	-	-	-	-	T2OE	DCEN
7	6	5	4	3	2	1	0

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

### 7. Timer 2 Auto Reload Mode (DCEN = 1)



### 8. Timer 2 in Baud Rate Generator Mode



## Rate Generator

is selected as the baud rate generator by setting and/or RCLK in T2CON (Table 2). Note that the rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for another function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 8.

Timer 2 in 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 overflow rate according to the following equation.

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

Timer 2 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 from counter operation when it is used as a baud rate generator. Normally, as a timer, it increments every machine cycle (at 1/12 the oscillator frequency). As a baud rate generator, however, it

increments every state time (at 1/2 the oscillator frequency). The baud rate formula is given below.

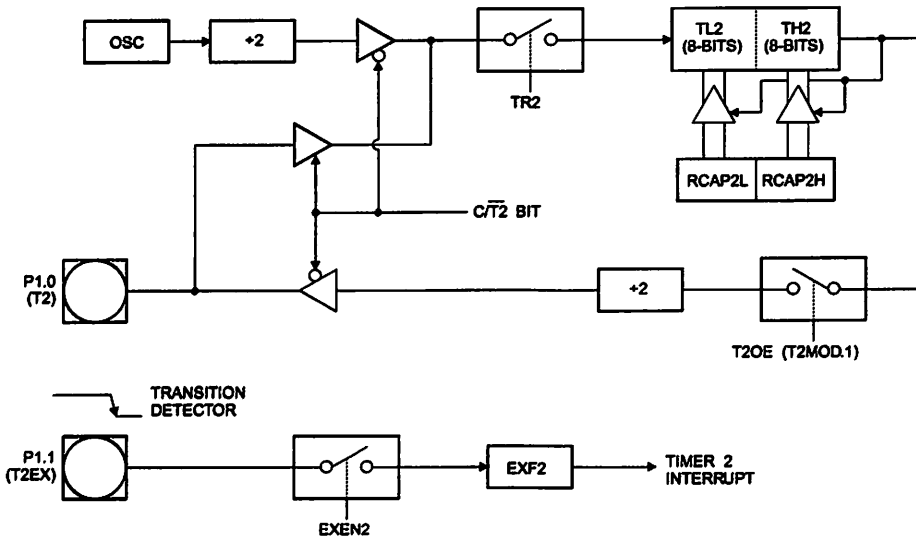
$$\frac{\text{Modes 1 and 3}}{\text{Baud Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - \text{RCAP2H}, \text{RCAP2L}]}$$

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 8. 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.

### 9. Timer 2 in Clock-Out Mode



## Programmable Clock Out

Duty cycle clock can be programmed to come out on pin 1, besides being a register pin, has two alternate functions. It can be programmed to input the external clock for Timer/Counter 2 or output a 50% duty cycle clock ranging from 61 Hz to 4 MHz at a 16 MHz operating frequency.

Configure the Timer/Counter 2 as a clock generator, bit T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

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 clock-out mode, Timer 2 roll-overs 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.

## Interrupts

The AT89S52 has a total of six interrupt vectors: two external interrupts (INT0 and INT1), three timer interrupts (Timer 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 3, IE. IE also contains a global disable bit, EA, which disables all interrupts at once.

As Table 5 shows that bit position IE.6 is unimplemented. In the AT89S52, bit position IE.5 is also unimplemented. User software should not write 1s to these bits, since they may be used in future AT89 products.

The timer interrupt is generated by the logical OR of bits TF0, TF1, and TF2 in register T2CON. Neither of these flags is set by hardware when the service routine is vectored to the interrupt, the service routine may have to determine whether it was TF0 or EXF2 that generated the interrupt, and the bit will have to be cleared in software.

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

Table 5. Interrupt Enable (IE) Register

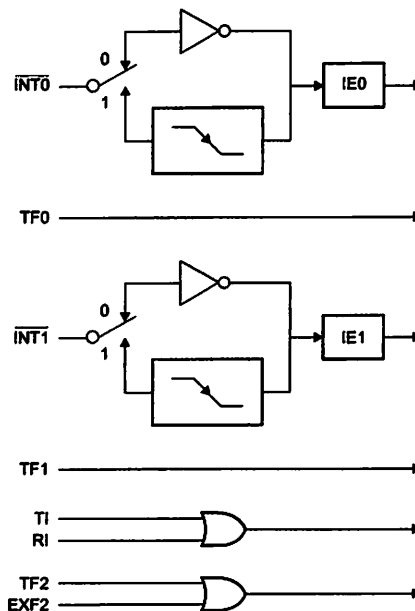
(MSB)								(LSB)
EA	-	ET2	ES	ET1	EX1	ET0	EX0	
Enable Bit = 1 enables the interrupt.								
Enable Bit = 0 disables the interrupt.								

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	Serial Port interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.

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

Figure 10. Interrupt Sources





## Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the oscillator 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 signal, since the input to the internal clocking circuitry is divided by two through a divide-by-two flip-flop, but minimum and maximum high and low time specifications must be observed.

## Idle Mode

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

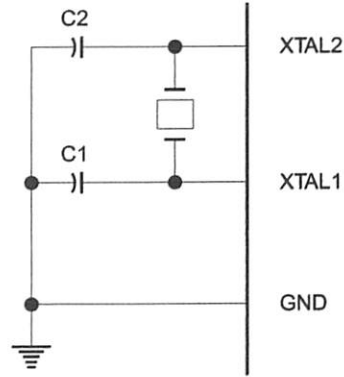
When Idle mode is terminated by a hardware reset, the device normally resumes program execution where it left off, up to two machine cycles before the reset algorithm takes control. On-chip hardware access to internal RAM in this event, but access to I/O pins is not inhibited. To eliminate the possibility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

## Power-down Mode

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

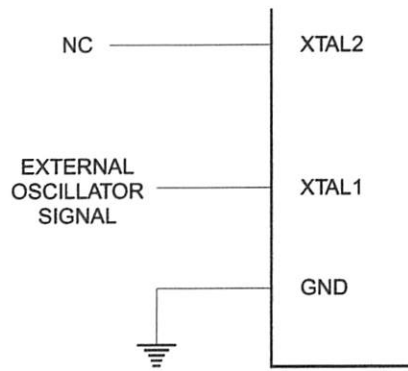
active long enough to allow the oscillator to restart and stabilize.

**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



## Table 1. Status of External Pins During Idle and Power-down Modes

	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
	Internal	1	1	Data	Data	Data	Data
	External	1	1	Float	Data	Address	Data
Idle	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data



## Flash Memory Lock Bits

AT89S52 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the addresses listed in the following table.

### Flash Memory Lock Bit Protection Modes

Program Lock Bits			Protection Type
LB1	LB2	LB3	
U	U	U	No program lock features
P	U	U	MOVX instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further programming of the Flash memory is disabled
P	P	U	Same as mode 2, but verify is also disabled
P	P	P	Same as mode 3, but external execution is also disabled

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

## Programming the Flash – Parallel Mode

AT89S52 is shipped with the on-chip Flash memory ready to be programmed. The programming interface consists of a high-voltage (12-volt) program enable signal and is compatible with conventional third-party Flash or EPROM programmers.

AT89S52 code memory array is programmed byte-by-byte.

**Programming Algorithm:** Before programming the AT89S52, the address, data, and control signals should be set according to the Flash programming mode table and Figures 13 and 14. To program the AT89S52, take the following steps:

1. Put the desired memory location on the address bus.

2.

3. Put the appropriate data byte on the data lines.

4. Activate the correct combination of control signals.

5. Use  $\overline{EA}/V_{PP}$  to 12V.

6. Use  $\overline{ALE}/\overline{PROG}$  once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 50  $\mu$ s.

Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

**Data Polling:** The AT89S52 features  $\overline{Data}$  Polling to indicate the end of a byte write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on P0.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

**Ready/Busy:** The progress of byte programming can also be monitored by the RDY/BSY output signal. P3.0 is pulled low after ALE goes high during programming to indicate  $\overline{BUSY}$ . P3.0 is pulled high again when programming is done to indicate READY.

**Program Verify:** If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The status of the individual lock bits can be verified directly by reading them back.

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

(000H) = 1EH indicates manufactured by Atmel

(100H) = 52H indicates 89S52

(200H) = 06H

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

In the serial programming mode, a chip erase operation is initiated by issuing the Chip Erase instruction. In this mode, chip erase is self-timed and takes about 500 ms.

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

## Programming the Flash – Serial Mode

The Code memory array can be programmed using the serial ISP interface while RST is pulled to  $V_{CC}$ . The serial interface consists of pins SCK, MOSI (input) and MISO (output). After RST is set high, the Programming Enable instruction needs to be executed first before other operations can be executed. Before a reprogramming sequence can occur, a Chip Erase operation is required.

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

Either an external system clock can be supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK)

ncy should be less than 1/16 of the crystal frequency. With a 33 MHz oscillator clock, the maximum SCK frequency is 2 MHz.

## Programming Algorithm

Program and verify the AT89S52 in the serial programming mode, the following sequence is recommended:

Power-up sequence:

Apply power between VCC and GND pins.

Set RST pin to "H".

If a crystal is not connected across pins XTAL1 and XTAL2, apply a 3 MHz to 33 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.

Enable serial programming by sending the Programming Enable serial instruction to pin MISO/P1.5. The frequency of the shift clock supplied at pin SCK/P1.7 needs to be less than the oscillator clock at XTAL1 divided by 16.

The Code array is programmed one byte at a time by supplying the address and data together with the

appropriate Write instruction. The write cycle is self-timed and typically takes less than 1 ms at 5V.

- 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.
- At the end of a programming session, RST can be set low to commence normal device operation.

Power-off sequence (if needed):

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

Set RST to "L".

Turn V<sub>CC</sub> power off.

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

## Serial Programming Instruction Set

The Instruction Set for Serial Programming follows a 4-byte protocol and is shown in Table 10.

## Programming Interface – Parallel Mode

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

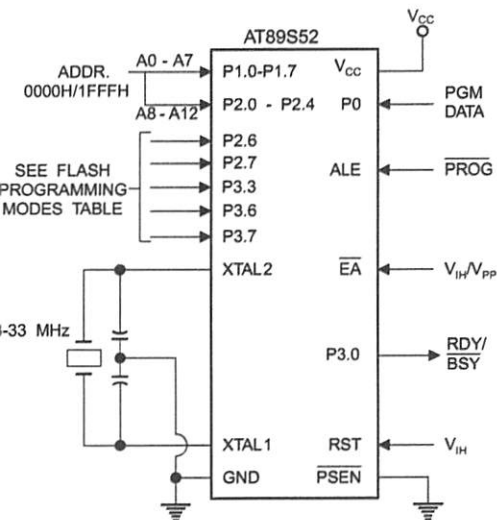
All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

### Flash Programming Modes

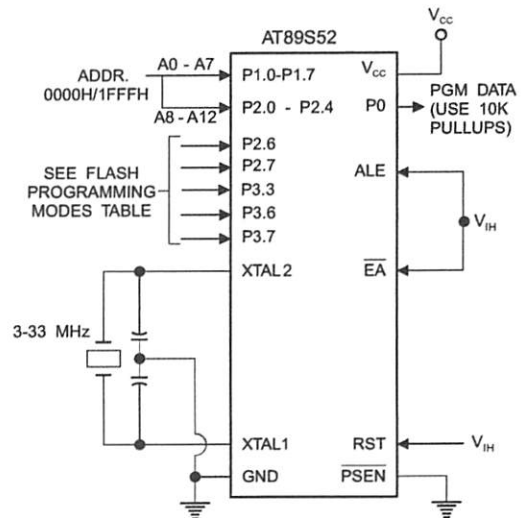
	V <sub>CC</sub>	RST	PSEN	ALE/ PROG	EA/ V <sub>PP</sub>	P2.6	P2.7	P3.3	P3.6	P3.7	P0.7-0 Data	P2.4-0	P1.7-0
												Address	
Write Code Data	5V	H	L		12V	L	H	H	H	H	D <sub>IN</sub>	A12-8	A7-0
Read Code Data	5V	H	L	H	H	L	L	L	H	H	D <sub>OUT</sub>	A12-8	A7-0
Write Lock Bit 1	5V	H	L		12V	H	H	H	H	H	X	X	X
Write Lock Bit 2	5V	H	L		12V	H	H	H	L	L	X	X	X
Write Lock Bit 3	5V	H	L		12V	H	L	H	H	L	X	X	X
Write Lock Bits	5V	H	L	H	H	H	H	L	H	L	P0.2, P0.3, P0.4	X	X
Chip Erase	5V	H	L		12V	H	L	H	L	L	X	X	X
Read Device ID	5V	H	L	H	H	L	L	L	L	L	1EH	X 0000	00H
Write Device ID	5V	H	L	H	H	L	L	L	L	L	52H	X 0001	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	06H	X 0010	00H

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

### 13. Programming the Flash Memory (Parallel Mode)



### Figure 14. Verifying the Flash Memory (Parallel Mode)

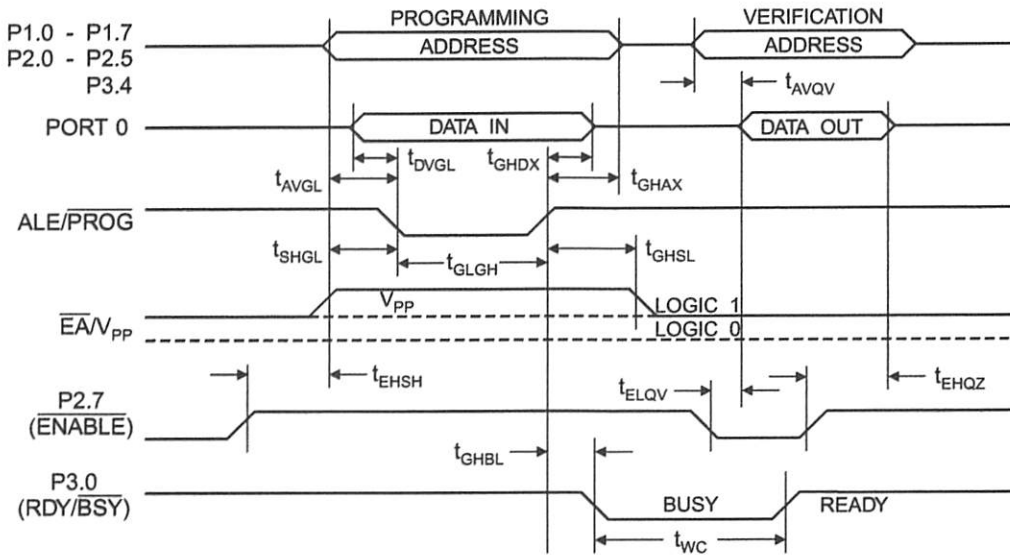


## Programming and Verification Characteristics (Parallel Mode)

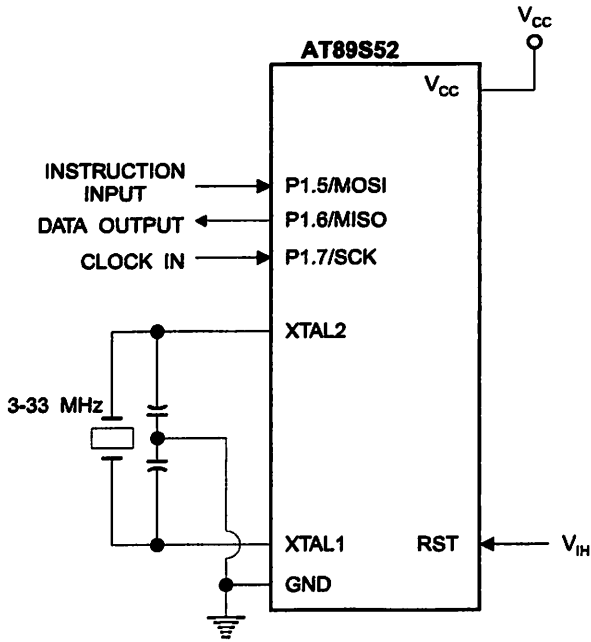
0°C to 30°C,  $V_{CC} = 4.5$  to  $5.5V$

Parameter	Min	Max	Units
Programming Supply Voltage	11.5	12.5	V
Programming Supply Current		10	mA
$V_{CC}$ Supply Current		30	mA
Oscillator Frequency	3	33	MHz
Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
Address Hold After $\overline{PROG}$	$48t_{CLCL}$		
Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
Data Hold After $\overline{PROG}$	$48t_{CLCL}$		
P2.7 ( $\overline{ENABLE}$ ) High to $V_{PP}$	$48t_{CLCL}$		
$V_{PP}$ Setup to $\overline{PROG}$ Low	10		$\mu s$
$V_{PP}$ Hold After $\overline{PROG}$	10		$\mu s$
$\overline{PROG}$ Width	0.2	1	$\mu s$
Address to Data Valid		$48t_{CLCL}$	
$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
Data Float After $\overline{ENABLE}$	0	$48t_{CLCL}$	
$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	$\mu s$
Byte Write Cycle Time		50	$\mu s$

### 15. Flash Programming and Verification Waveforms – Parallel Mode

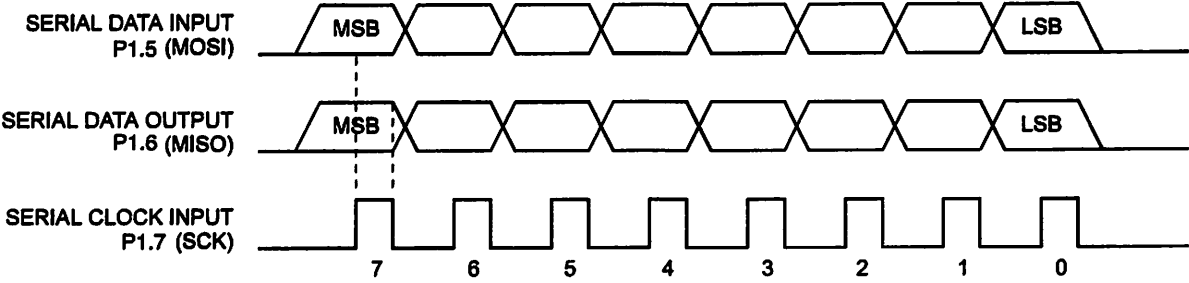


## 16. Flash Memory Serial Downloading



## Programming and Verification Waveforms – Serial Mode

### 17. Serial Programming Waveforms



## Serial Programming Instruction Set

Instruction	Instruction Format				Operation
	Byte 1	Byte 2	Byte 3	Byte 4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx 0110 1001 (Output)	Enable Serial Programming while RST is high
Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash memory array
Read Program Memory (Byte Mode)	0010 0000	xxx A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Read data from Program memory in the byte mode
Write Program Memory (Byte Mode)	0100 0000	xxx A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Write data to Program memory in the byte mode
Write Lock Bits <sup>(2)</sup>	1010 1100	1110 00 B1 B2	xxxx xxxx	xxxx xxxx	Write Lock bits. See Note (2).
Read Lock Bits	0010 0100	xxxx xxxx	xxxx xxxx	xx LB3 LB2 LB1 xx	Read back current status of the lock bits (a programmed lock bit reads back as a '1')
Read Signature Bytes <sup>(1)</sup>	0010 1000	xxx A5 A4 A3 A2 A1 A0	xxx xxxx	Signature Byte	Read Signature Byte
Read Program Memory (Page Mode)	0011 0000	xxx A12 A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Read data from Program memory in the Page Mode (256 bytes)
Write Program Memory (Page Mode)	0101 0000	xxx A12 A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Write data to Program memory in the Page Mode (256 bytes)

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

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

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

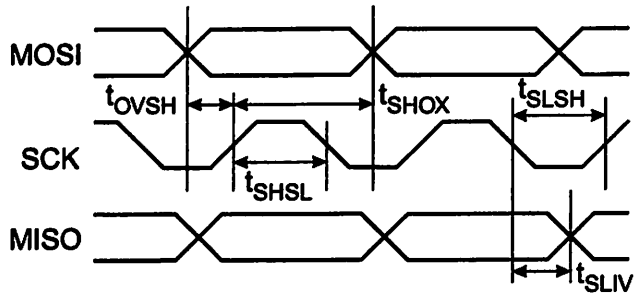
When the Reset signal is high, SCK should be low for at least 64 clocks before it goes high to clock in the enable bytes. No pulsing of Reset signal is necessary. SCK should be no faster than 1/16 of the system clock at

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



## Programming Characteristics

### 18. Serial Programming Timing



### 19. Serial Programming Characteristics, $T_A = -40^\circ\text{C}$ to $85^\circ\text{C}$ , $V_{CC} = 4.0 - 5.5\text{V}$ (Unless otherwise noted)

Parameter	Min	Typ	Max	Units
Oscillator Frequency	0		33	MHz
Oscillator Period	30			ns
SCK Pulse Width High	$2 t_{CLCL}$			ns
SCK Pulse Width Low	$2 t_{CLCL}$			ns
MOSI Setup to SCK High	$t_{CLCL}$			ns
MOSI Hold after SCK High	$2 t_{CLCL}$			ns
SCK Low to MISO Valid	10	16	32	ns
Chip Erase Instruction Cycle Time			500	ms
Serial Byte Write Cycle Time			$64 t_{CLCL} + 400$	$\mu\text{s}$



## Absolute Maximum Ratings\*

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

## Characteristics

Values shown in this table are valid for  $T_A = -40^\circ\text{C}$  to  $85^\circ\text{C}$  and  $V_{CC} = 4.0\text{V}$  to  $5.5\text{V}$ , unless otherwise noted.

Parameter	Condition	Min	Max	Units
Input Low Voltage	(Except $\overline{EA}$ )	-0.5	$0.2 V_{CC} - 0.1$	V
Input Low Voltage ( $\overline{EA}$ )		-0.5	$0.2 V_{CC} - 0.3$	V
Input High Voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
Input High Voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V
Output Low Voltage <sup>(1)</sup> (Ports 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.45	V
Output Low Voltage <sup>(1)</sup> (Port 0, ALE, $\overline{PSEN}$ )	$I_{OL} = 3.2 \text{ mA}$		0.45	V
Output High Voltage (Ports 1,2,3, ALE, $\overline{PSEN}$ )	$I_{OH} = -60 \mu\text{A}, V_{CC} = 5V \pm 10\%$	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
Output High Voltage (Port 0 in External Bus Mode)	$I_{OH} = -800 \mu\text{A}, V_{CC} = 5V \pm 10\%$	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
Logical 0 Input Current (Ports 1,2,3)	$V_{IN} = 0.45\text{V}$		-50	$\mu\text{A}$
Logical 1 to 0 Transition Current (Ports 1,2,3)	$V_{IN} = 2\text{V}, V_{CC} = 5V \pm 10\%$		-650	$\mu\text{A}$
Input Leakage Current (Port 0, $\overline{EA}$ )	$0.45 < V_{IN} < V_{CC}$		$\pm 10$	$\mu\text{A}$
Reset Pull-down Resistor		10	30	$\text{K}\Omega$
Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF
Power Supply Current	Active Mode, 12 MHz		25	mA
	Idle Mode, 12 MHz		6.5	mA
Power-down Mode <sup>(1)</sup>	$V_{CC} = 5.5\text{V}$		50	$\mu\text{A}$

1. Under steady state (non-transient) conditions,  $I_{OL}$  must be externally limited as follows:

Maximum  $I_{OL}$  per port pin: 10 mA

Maximum  $I_{OL}$  per 8-bit port:

Port 0: 26 mA      Ports 1, 2, 3: 15 mA

Maximum total  $I_{OL}$  for all output pins: 71 mA

If  $I_{OL}$  exceeds the test condition,  $V_{OL}$  may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

2. Minimum  $V_{CC}$  for Power-down is 2V.





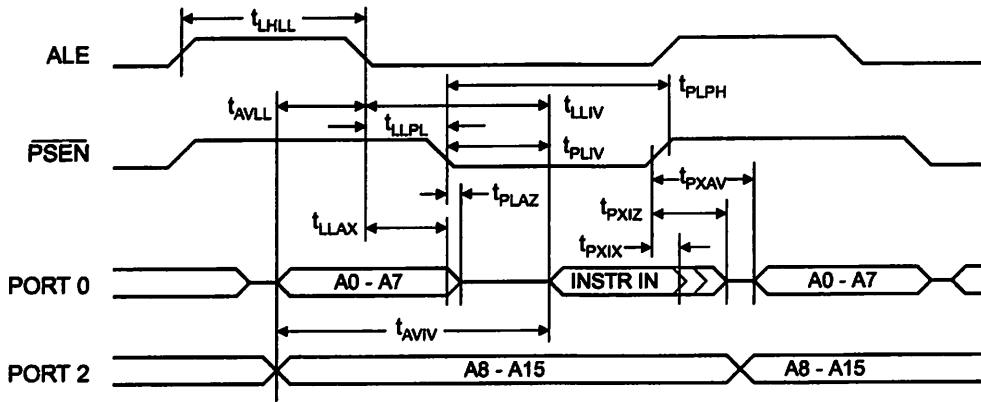
## Characteristics

Operating conditions, load capacitance for Port 0, ALE/PROG, and PSEN = 100 pF; load capacitance for all other = 80 pF.

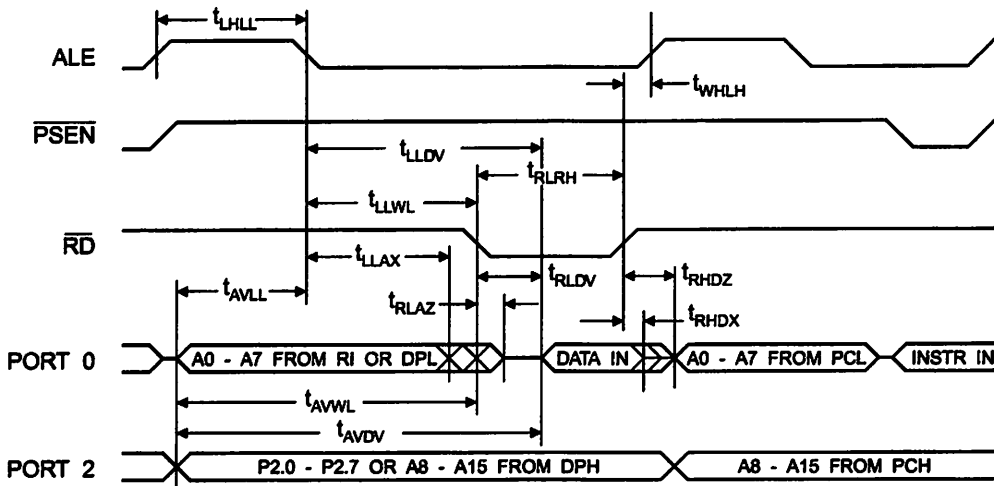
### Normal Program and Data Memory Characteristics

Parameter	12 MHz Oscillator		Variable Oscillator		Units
	Min	Max	Min	Max	
Oscillator Frequency			0	33	MHz
ALE Pulse Width	127		$2t_{CLCL}-40$		ns
Address Valid to ALE Low	43		$t_{CLCL}-25$		ns
Address Hold After ALE Low	48		$t_{CLCL}-25$		ns
ALE Low to Valid Instruction In		233		$4t_{CLCL}-65$	ns
ALE Low to PSEN Low	43		$t_{CLCL}-25$		ns
PSEN Pulse Width	205		$3t_{CLCL}-45$		ns
PSEN Low to Valid Instruction In		145		$3t_{CLCL}-60$	ns
Input Instruction Hold After PSEN	0		0		ns
Input Instruction Float After PSEN		59		$t_{CLCL}-25$	ns
PSEN to Address Valid	75		$t_{CLCL}-8$		ns
Address to Valid Instruction In		312		$5t_{CLCL}-80$	ns
PSEN Low to Address Float		10		10	ns
RD Pulse Width	400		$6t_{CLCL}-100$		ns
WR Pulse Width	400		$6t_{CLCL}-100$		ns
RD Low to Valid Data In		252		$5t_{CLCL}-90$	ns
Data Hold After RD	0		0		ns
Data Float After RD		97		$2t_{CLCL}-28$	ns
ALE Low to Valid Data In		517		$8t_{CLCL}-150$	ns
Address to Valid Data In		585		$9t_{CLCL}-165$	ns
ALE Low to RD or WR Low	200	300	$3t_{CLCL}-50$	$3t_{CLCL}+50$	ns
Address to RD or WR Low	203		$4t_{CLCL}-75$		ns
Data Valid to WR Transition	23		$t_{CLCL}-30$		ns
Data Valid to WR High	433		$7t_{CLCL}-130$		ns
Data Hold After WR	33		$t_{CLCL}-25$		ns
RD Low to Address Float		0		0	ns
RD or WR High to ALE High	43	123	$t_{CLCL}-25$	$t_{CLCL}+25$	ns

Internal Program Memory Read Cycle

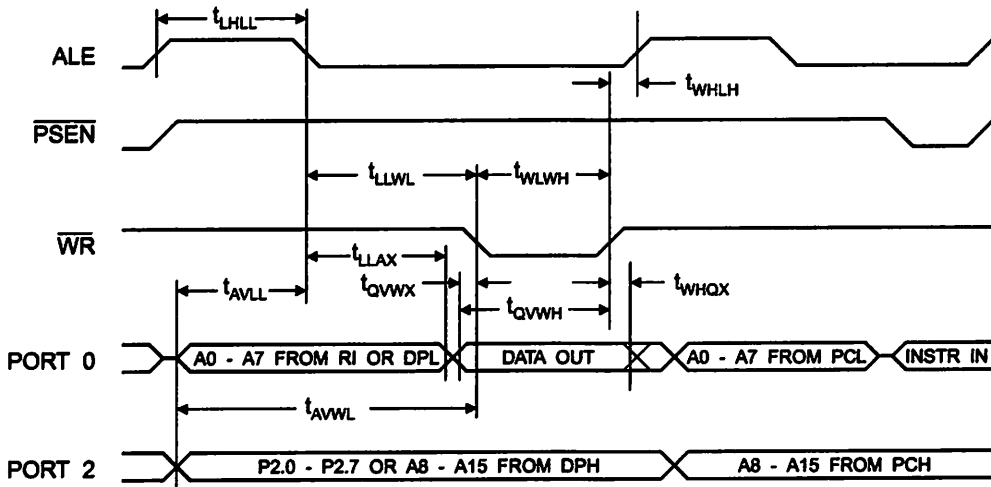


Internal Data Memory Read Cycle

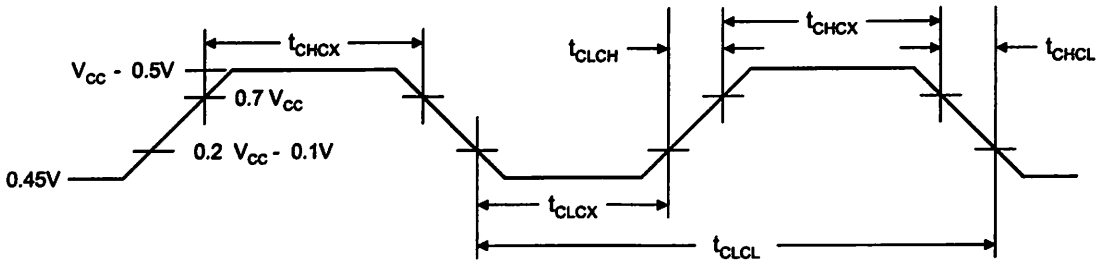




## nal Data Memory Write Cycle



## nal Clock Drive Waveforms



## nal Clock Drive

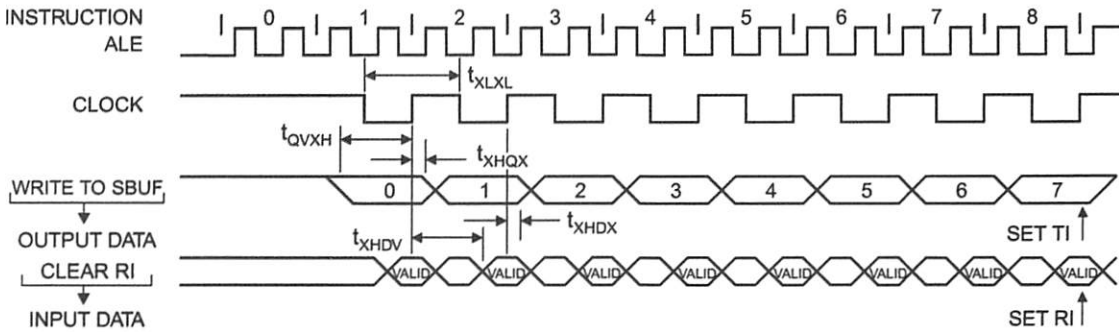
Parameter	Min	Max	Units
Oscillator Frequency	0	33	MHz
Clock Period	30		ns
High Time	12		ns
Low Time	12		ns
Rise Time		5	ns
Fall Time		5	ns

## Serial Port Timing: Shift Register Mode Test Conditions

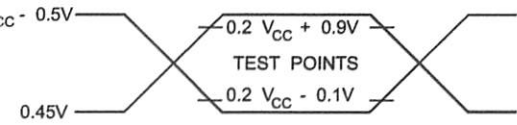
Values in this table are valid for  $V_{CC} = 4.0V$  to  $5.5V$  and Load Capacitance =  $80 pF$ .

Parameter	12 MHz Osc		Variable Oscillator		Units
	Min	Max	Min	Max	
Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		$\mu s$
Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}-133$		ns
Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}-80$		ns
Input Data Hold After Clock Rising Edge	0		0		ns
Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}-133$	ns

## Register Mode Timing Waveforms



## Testing Input/Output Waveforms<sup>(1)</sup>



- 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<sup>(1)</sup>



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



## Ordering Information

Order Code	Power Supply	Ordering Code	Package	Operation Range
	4.0V to 5.5V	AT89S52-24AC	44A	Commercial (0° C to 70° C)
		AT89S52-24JC	44J	
		AT89S52-24PC	40P6	
		AT89S52-24AI	44A	Industrial (-40° C to 85° C)
		AT89S52-24JI	44J	
		AT89S52-24PI	40P6	
	4.5V to 5.5V	AT89S52-33AC	44A	Commercial (0° C to 70° C)
		AT89S52-33JC	44J	
		AT89S52-33PC	40P6	

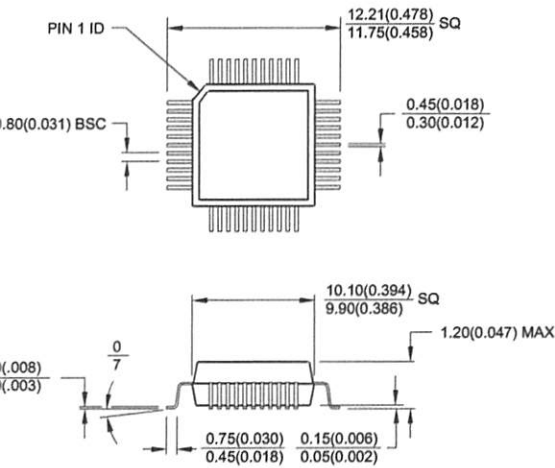
[ ] = Preliminary Availability

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

# AT89S52

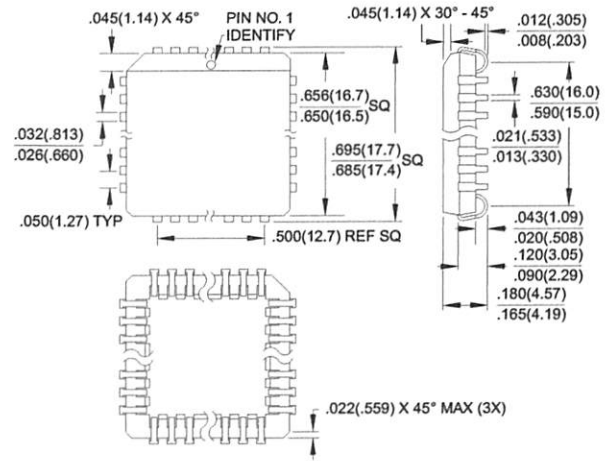
## Mounting Information

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

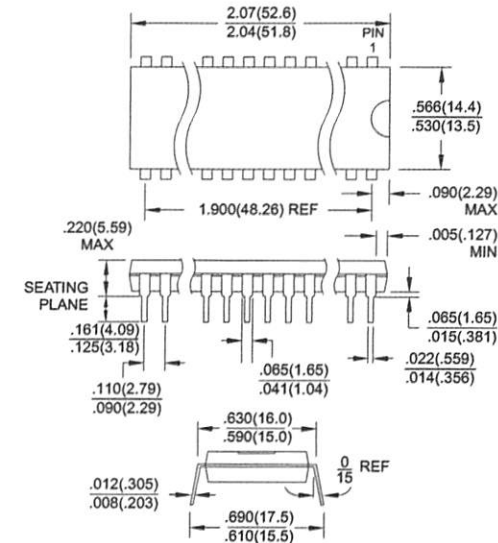


Controlling dimension: millimeters

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



**PDIP6**, 40-pin, 0.600" Wide, Plastic Dual In Line Package (PDIP)  
Dimensions in Inches and (Millimeters)  
DEC STANDARD MS-011 AC





**Headquarters**

**Corporate Headquarters**  
5 Orchard Parkway  
Folsom, CA 95131  
(408) 441-0311  
(408) 487-2600

**Atmel Sarl**  
Parc des Arsenaux 41  
Case Postale 80  
1705 Fribourg  
Switzerland  
(41) 26-426-5555  
(41) 26-426-5500

**Atmel Asia, Ltd.**  
Room 1219  
Lachem Golden Plaza  
100 Tody Road Tsimhatsui  
Kowloon  
Hong Kong  
(852) 2721-9778  
(852) 2722-1369

**Atmel Japan K.K.**  
Tonetsu Shinkawa Bldg.  
1-8 Shinkawa  
1-1-1 Oo-ku, Tokyo 104-0033  
Japan  
(81) 3-3523-3551  
(81) 3-3523-7581

**Atmel Product Operations**

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

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

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

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

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

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

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

**e-mail**  
literature@atmel.com

**Web Site**  
http://www.atmel.com

**BBS**  
1-(408) 436-4309

**Atmel Corporation 2001.**

Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors that may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Atmel is the registered trademark of Atmel.

Intel is the registered trademark of Intel Corporation. Terms and product names in this document may be the trademarks of others.



Printed on recycled paper.



**DATASHEET ATMEL  
AT28C16**

## Features

- Read Access Time - 150 ns
- Byte Write - 200  $\mu$ s or 1 ms
- Timed Byte Write Cycle
- Internal Address and Data Latches
- Internal Control Timer
- Automatic Clear Before Write
- Host Microprocessor Control
- DATA POLLING
- Low Power
- 10 mA Active Current
- 100  $\mu$ A CMOS Standby Current
- High Reliability
- Endurance:  $10^4$  or  $10^5$  Cycles
- Data Retention: 10 Years
- 100% Supply
- 5V & TTL Compatible Inputs and Outputs
- JEDEC Approved Byte Wide Pinout
- Commercial and Industrial Temperature Ranges

## Description

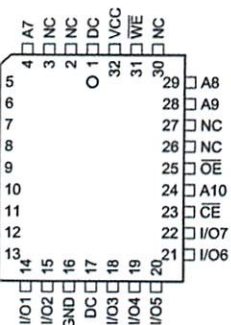
AT28C16 is a low-power, high-performance Electrically Erasable and Programmable Read Only Memory with easy to use features. The AT28C16 is a 16K memory organized as 2,048 words by 8 bits. The device is manufactured with Atmel's reliable 1 $\mu$ m CMOS technology.

(continued)

## Pin Configurations

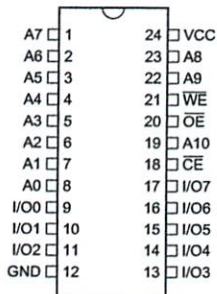
Pin Name	Function
A0-A9	Addresses
CE	Chip Enable
OE	Output Enable
WE	Write Enable
I/O0-I/O7	Data Inputs/Outputs
NC	No Connect
NC	Don't Connect

PLCC  
Top View



PLCC package pins 1 and 17  
DO NOT CONNECT.

PDIP, SOIC  
Top View



## 16K (2K x 8) Parallel EEPROMs

### AT28C16

Rev. 0540B-10/98



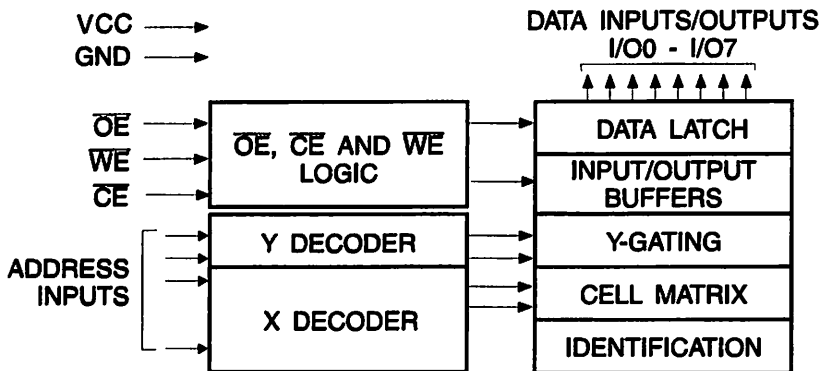


28C16 is accessed like a static RAM for the read or write cycles without the need of external components. During a write cycle, the address and data are latched internally, freeing the microprocessor address and data bus for other operations. Following the initiation of a write cycle, the chip will go to a busy state and automatically clear the latched data using an internal control timer. The duration of a write cycle can be determined by the  $\overline{\text{DATA}}_7$  signal of I/O<sub>7</sub>. Once the end of a write cycle has been reached, a new access for a read or a write can begin.

The CMOS technology offers fast access times of 150 ns at low power dissipation. When the chip is deselected the standby current is less than 100  $\mu\text{A}$ .

Atmel's 28C16 has additional features to ensure high quality and manufacturability. The device utilizes error correction internally for extended endurance and for improved data retention characteristics. An extra 32 bytes of EEPROM are available for device identification or tracking.

### Block Diagram



### Absolute Maximum Ratings\*

Temperature Under Bias .....	-55°C to +125°C
Storage Temperature .....	-65°C to +150°C
Input Voltages (including NC Pins) with respect to Ground .....	-0.6V to +6.25V
Output Voltages with respect to Ground .....	-0.6V to $V_{CC} + 0.6V$
Voltage on $\overline{\text{OE}}$ and A9 with respect to Ground .....	-0.6V to +13.5V

**\*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.

## Operation

The AT28C16 is accessed like a Static RAM. When  $\overline{CE}$  and  $\overline{OE}$  are low and  $\overline{WE}$  is high, the data stored in the memory location determined by the address pins is placed on the outputs. The outputs are put in a high impedance state whenever  $\overline{CE}$  or  $\overline{OE}$  is high. This dual line architecture gives designers increased flexibility in preventing contention.

**WRITE:** Writing data into the AT28C16 is similar to writing into a Static RAM. A low pulse on the  $\overline{WE}$  or  $\overline{CE}$  with  $\overline{OE}$  high and  $\overline{CE}$  or  $\overline{WE}$  low (respectively) initiates a byte write. The address location is latched on the falling edge of  $\overline{WE}$  (or  $\overline{CE}$ ); the new data is latched on the rising edge. Internally, the device performs a self-timed write. Once a byte write has been started, it automatically times itself to completion. Once a programming operation has been initiated and for the duration of a read operation will effectively be a polling operation.

**BYTE WRITE:** The AT28C16E offers a byte write time of 200  $\mu$ s maximum. This feature allows the entire array to be rewritten in 0.4 seconds.

**POLLING:** The AT28C16 provides  $\overline{DATA}$  POLLING during the completion of a write cycle. During a write

cycle, an attempted read of the data being written results in the complement of that data for  $I/O_7$  (the other outputs are indeterminate). When the write cycle is finished, true data appears on all outputs.

**WRITE PROTECTION:** Inadvertent writes to the device are protected against in the following ways: (a)  $V_{CC}$  sense—if  $V_{CC}$  is below 3.8V (typical) the write function is inhibited; (b)  $V_{CC}$  power on delay—once  $V_{CC}$  has reached 3.8V the device will automatically time out 5 ms (typical) before allowing a byte write; and (c) write inhibit—holding any one of  $\overline{CE}$  low,  $\overline{OE}$  high, or  $\overline{WE}$  high inhibits byte write cycles.

**CHIP CLEAR:** The contents of the entire memory of the AT28C16 may be set to the high state by the CHIP CLEAR operation. By setting  $\overline{CE}$  low and  $\overline{OE}$  to 12 volts, the chip is cleared when a 10 msec low pulse is applied to  $\overline{WE}$ .

**DEVICE IDENTIFICATION:** An extra 32 bytes of EEPROM memory are available to the user for device identification. By raising A9 to  $12 \pm 0.5V$  and using address locations 7E0H to 7FFH the additional bytes may be written to or read from in the same manner as the regular memory array.



## AC Operating Range

		AT28C16-15
Operating Temperature (Case)	Com.	0°C - 70°C
	Ind.	-40°C - 85°C
Power Supply		5V ± 10%

## Operating Modes

	$\overline{CE}$	$\overline{OE}$	$\overline{WE}$	I/O
Output Enable	$V_{IL}$	$V_{IL}$	$V_{IH}$	$D_{OUT}$
Input Enable	$V_{IL}$	$V_{IH}$	$V_{IL}$	$D_{IN}$
Read/Write Inhibit	$V_{IH}$	X <sup>(1)</sup>	X	High Z
Output Inhibit	X	X	$V_{IH}$	
Input Inhibit	X	$V_{IL}$	X	
Output Disable	X	$V_{IH}$	X	High Z
Input Disable	$V_{IL}$	$V_H$ <sup>(3)</sup>	$V_{IL}$	High Z

1. X can be  $V_{IL}$  or  $V_{IH}$ .

2. Refer to AC Programming Waveforms.

3.  $V_H = 12.0V \pm 0.5V$

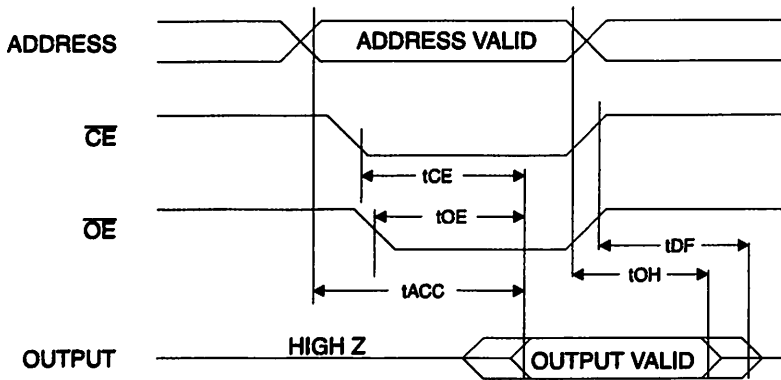
## Characteristics

Parameter	Condition	Min	Max	Units
Input Load Current	$V_{IN} = 0V$ to $V_{CC} + 1V$		10	$\mu A$
Output Leakage Current	$V_{I/O} = 0V$ to $V_{CC}$		10	$\mu A$
$V_{CC}$ Standby Current CMOS	$\overline{CE} = V_{CC} - 0.3V$ to $V_{CC} + 1.0V$		100	$\mu A$
$V_{CC}$ Standby Current TTL	$\overline{CE} = 2.0V$ to $V_{CC} + 1.0V$	Com.	2	mA
		Ind.	3	mA
$V_{CC}$ Active Current AC	$f = 5$ MHz; $I_{OUT} = 0$ mA $\overline{CE} = V_{IL}$	Com.	30	mA
		Ind.	45	mA
Input Low Voltage			0.8	V
Input High Voltage		2.0		V
Output Low Voltage	$I_{OL} = 2.1$ mA		.4	V
Output High Voltage	$I_{OH} = -400$ $\mu A$	2.4		V

## Lead Characteristics

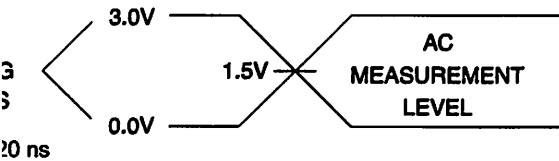
Parameter	AT28C16-15		Units
	Min	Max	
Address to Output Delay		150	ns
$\overline{CE}$ to Output Delay		150	ns
$\overline{OE}$ to Output Delay	10	70	ns
$\overline{CE}$ or $\overline{OE}$ High to Output Float	0	50	ns
Output Hold from $\overline{OE}$ , $\overline{CE}$ or Address, whichever occurred first	0		ns

## Lead Waveforms (1)(2)(3)(4)

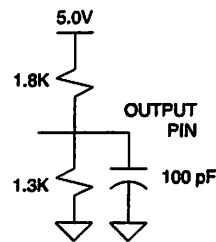


- $\overline{CE}$  may be delayed up to  $t_{ACC} - t_{CE}$  after the address transition without impact on  $t_{ACC}$ .
- $\overline{OE}$  may be delayed up to  $t_{CE} - t_{OE}$  after the falling edge of  $\overline{CE}$  without impact on  $t_{CE}$  or by  $t_{ACC} - t_{OE}$  after an address change without impact on  $t_{ACC}$ .
- $t_{DF}$  is specified from  $\overline{OE}$  or  $\overline{CE}$  whichever occurs first ( $C_L = 5 \text{ pF}$ ).
- This parameter is characterized and is not 100% tested.

## Test Waveforms and Measurement Level



## Output Test Load



## Capacitance

at  $V_{IN} = 0V$ ,  $T = 25^\circ\text{C}$ (1)

Parameter	Typ	Max	Units	Conditions
	4	6	pF	$V_{IN} = 0V$
	8	12	pF	$V_{OUT} = 0V$

- This parameter is characterized and is not 100% tested.



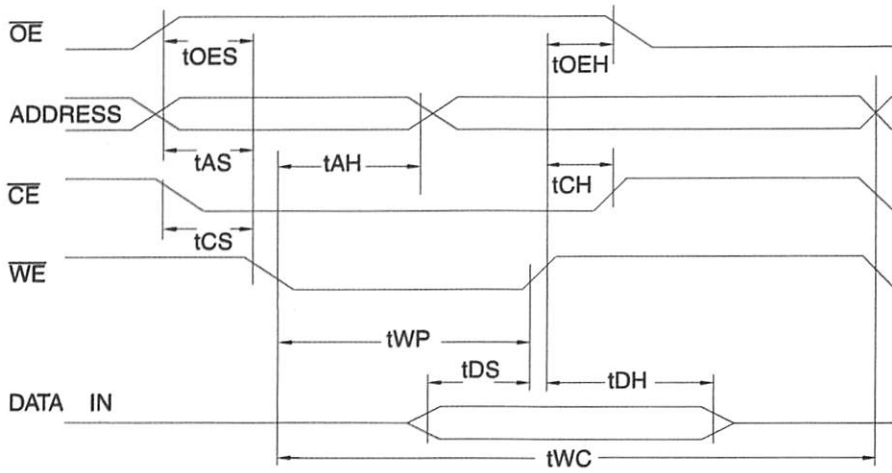


## Write Characteristics

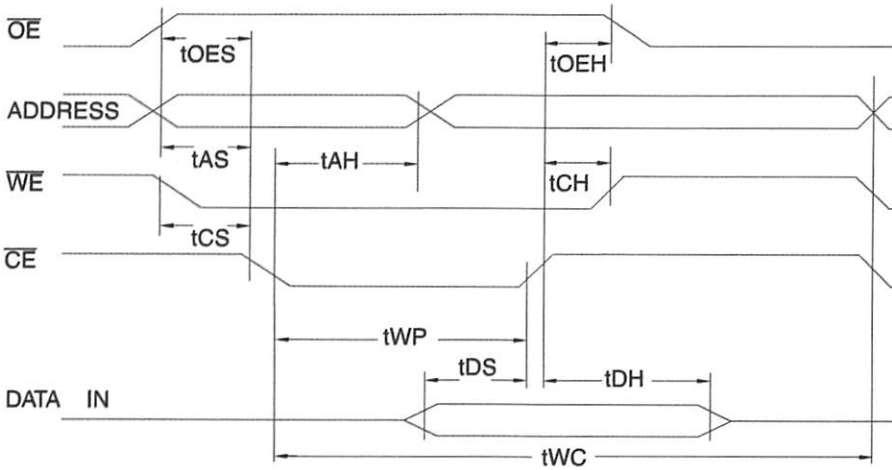
Symbol	Parameter	Min	Typ	Max	Units
$t_{s}$	Address, $\overline{OE}$ Set-up Time	10			ns
	Address Hold Time	50			ns
	Write Pulse Width ( $\overline{WE}$ or $\overline{CE}$ )	100		1000	ns
	Data Set-up Time	50			ns
$t_{h}$	Data, $\overline{OE}$ Hold Time	10			ns
	$\overline{CE}$ to $\overline{WE}$ and $\overline{WE}$ to $\overline{CE}$ Set-up and Hold Time	0			ns
	Write Cycle Time		0.5	1.0	ms
			100	200	$\mu$ s

## Write Waveforms

### Controlled



### Controlled



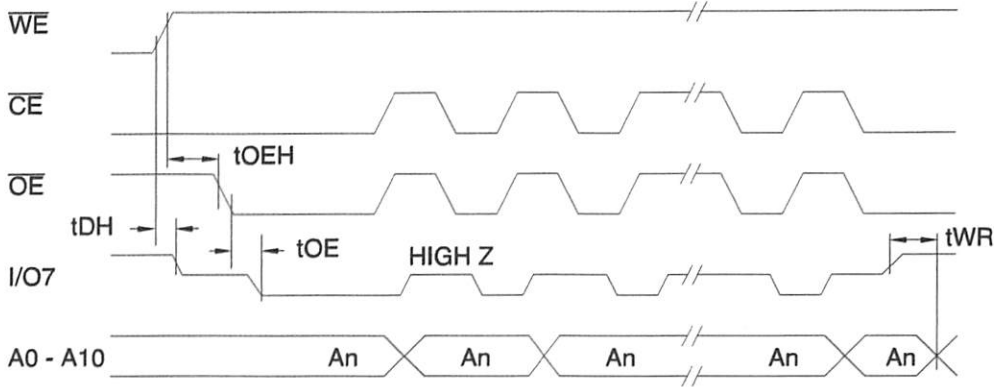
**AT28C16**

## Polling Characteristics<sup>(1)</sup>

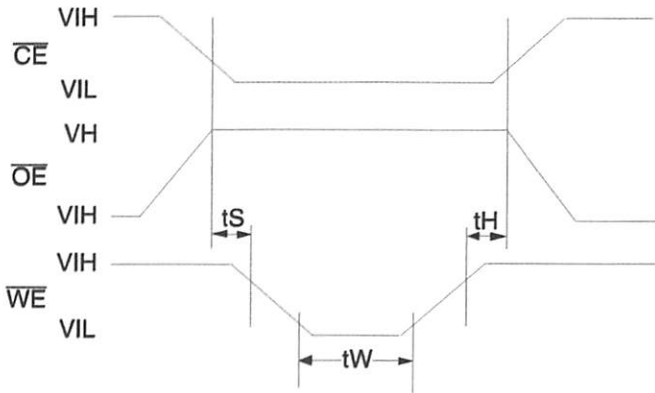
Parameter	Min	Typ	Max	Units
Data Hold Time	10			ns
$\overline{OE}$ Hold Time	10			ns
$\overline{OE}$ to Output Delay <sup>(2)</sup>				ns
Write Recovery Time	0			ns

1. These parameters are characterized and not 100% tested.
2. See AC Characteristics.

## Polling Waveforms



## Erase Waveforms

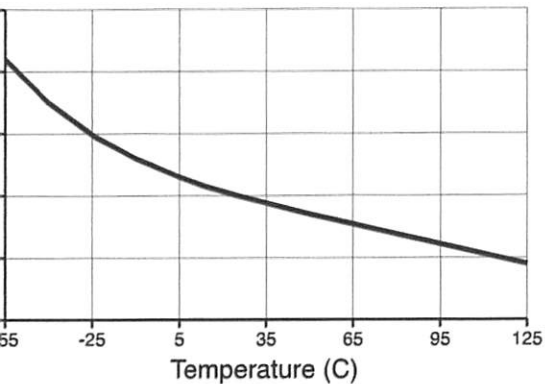


μsec (min.)  
 μsec (min.)  
 V ± 0.5V

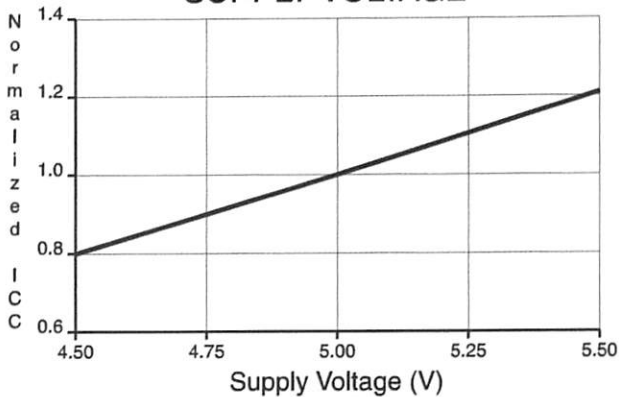




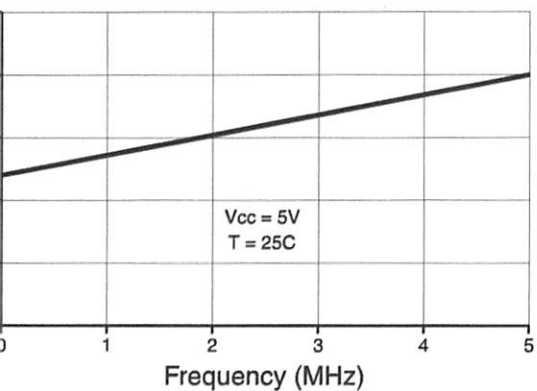
**NORMALIZED SUPPLY CURRENT vs. TEMPERATURE**



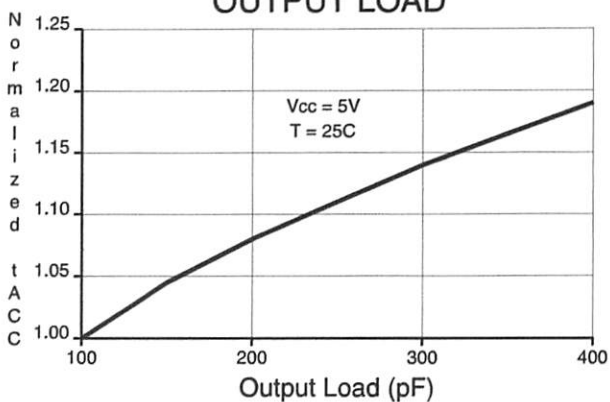
**NORMALIZED SUPPLY CURRENT vs. SUPPLY VOLTAGE**



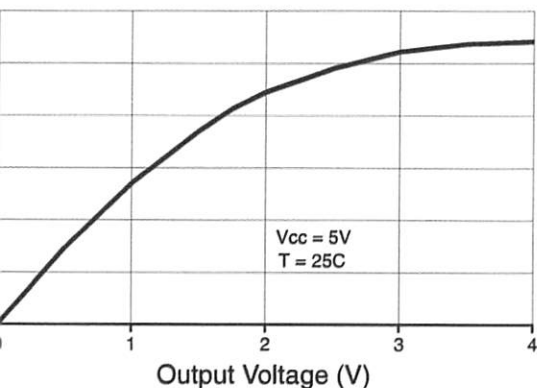
**NORMALIZED SUPPLY CURRENT vs. ADDRESS FREQUENCY**



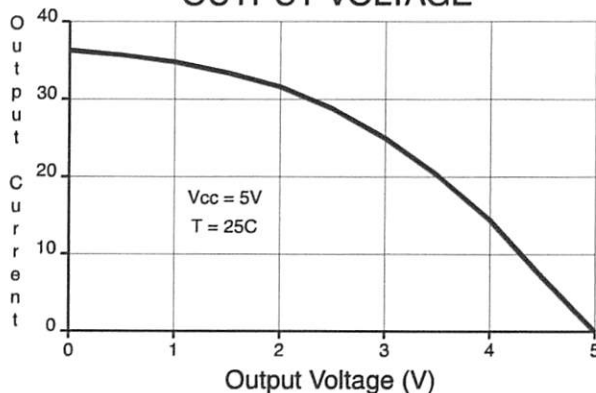
**NORMALIZED ACCESS TIME vs. OUTPUT LOAD**



**OUTPUT SINK CURRENT vs. OUTPUT VOLTAGE**



**OUTPUT SOURCE CURRENT vs. OUTPUT VOLTAGE**



## ing Information<sup>(1)</sup>

$I_{CC}$ (mA)		Ordering Code	Package	Operation Range
Active	Standby			
30	0.1	AT28C16(E)-15JC AT28C16(E)-15PC AT28C16(E)-15SC	32J 24P6 24S	Commercial (0°C to 70°C)
45	0.1	AT28C16(E)-15JI AT28C16(E)-15PI AT28C16(E)-15SI	32J 24P6 24S	Industrial (-40°C to 85°C)

1. See Valid Part Numbers table below.
2. The 28C16 200 ns and 250 ns speed selections have been removed from valid selections table and are replaced by the faster 150 ns  $T_{AA}$  offering.
3. The 28C16 ceramic package offerings have been removed. New designs should utilize the 28C256 ceramic offerings.

## Part Numbers

Following table lists standard Atmel products that can be ordered.

Numbers	Speed	Package and Temperature Combinations
16	15	JC, JI, PC, PI, SC, SI
16E	15	JC, JI, PC, PI, SC, SI
16	-	W

## Products

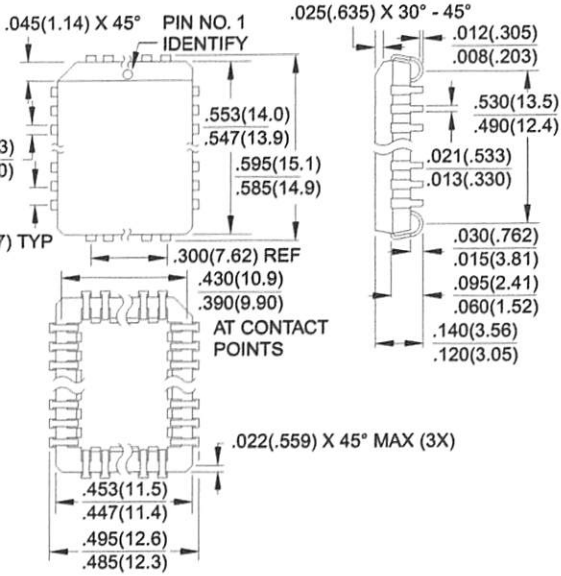
Device Section: Parallel EEPROM Die Products

Package Type	
	32 Lead, Plastic J-Leaded Chip Carrier (PLCC)
	24 Lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)
	Die
Options	
	Standard Device: Endurance = 10K Write Cycles; Write Time = 1 ms
	High Endurance Option: Endurance = 100K Write Cycles; Write Time = 200 $\mu$ s

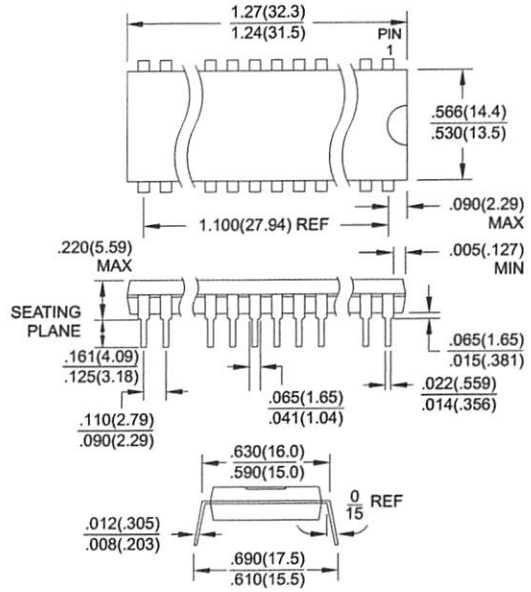


## ing Information

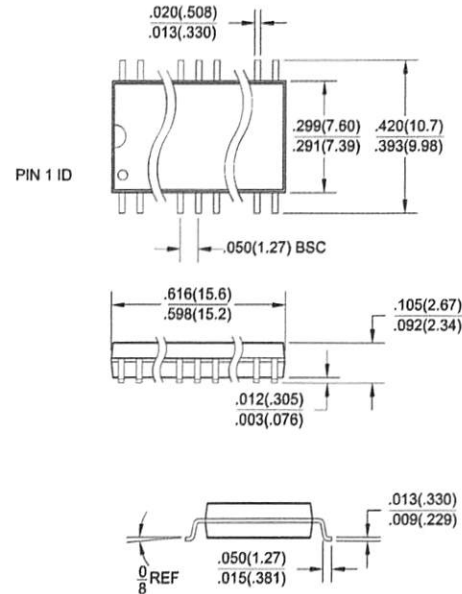
32-Lead, Plastic J-Leaded Chip Carrier (PLCC)  
 Dimensions in Inches and (Millimeters)  
 JEDEC STANDARD MS-018 AA



24P6, 24-Lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)  
 Dimensions in Inches and (Millimeters)  
 JEDEC STANDARD MS-011 AA



24-Lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)  
 Dimensions in Inches and (Millimeters)





## Headquarters

### Corporate Headquarters

Orchard Parkway  
Folsom, CA 95131  
(408) 441-0311  
(408) 487-2600

Atmel U.K., Ltd.  
Atmel Business Centre  
The Valley Way  
Wokingham, Surrey GU15 3YL  
and  
(44) 1276-686677  
(44) 1276-686697

Atmel Asia, Ltd.  
Room 1219  
The Gateway Golden Plaza  
200 Nathan Road  
Kowloon, Hong Kong  
(852) 27219778  
(852) 27221369

Atmel Japan K.K.  
Atmel Shinkawa Bldg., 9F  
1-8-1 Shinkawa  
Nishi-ku, Tokyo 104-0033  
and  
(81) 3-3523-3551  
(81) 3-3523-7581

## Atmel Operations

### Atmel Colorado Springs

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

### Atmel Rousset

Zone Industrielle  
13106 Rousset Cedex, France  
TEL (33) 4 42 53 60 00  
FAX (33) 4 42 53 60 01

---

### *Fax-on-Demand*

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

*e-mail*  
literature@atmel.com

*Web Site*  
<http://www.atmel.com>

*BBS*  
1-(408) 436-4309

### Atmel Corporation 1998.

Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's website. The Company assumes no responsibility for errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property rights are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not intended for use as critical components in life support devices or systems.

Atmel, Atmel logo and/or <sup>TM</sup> are registered trademarks and trademarks of Atmel Corporation.

Other product names in this document may be trademarks of others.



Printed on recycled paper.

0540B-10/98/xM

**DATASHEET TTL**  
**74LS373**

# SN74LS373 SN74LS374

## Octal Transparent Latch with 3-State Outputs; Octal D-Type Flip-Flop with 3-State Output

The SN74LS373 consists of eight latches with 3-state outputs for organized system applications. The flip-flops appear transparent to the data (data changes asynchronously) when Latch Enable (LE) is HIGH. When LE is LOW, the data that meets the setup times is latched. Data appears on the bus when the Output Enable ( $\overline{OE}$ ) is LOW. When  $\overline{OE}$  is HIGH the bus output is in the high impedance state. The SN74LS374 is a high-speed, low-power Octal D-type Flip-Flop offering separate D-type inputs for each flip-flop and 3-state outputs for bus oriented applications. A buffered Clock (CP) and Output Enable (OE) is common to all flip-flops. The SN74LS374 is manufactured using advanced Low Power Schottky technology and is compatible with all ON Semiconductor TTL families.

- Eight Latches in a Single Package
- 3-State Outputs for Bus Interfacing
- Hysteresis on Latch Enable
- Edge-Triggered D-Type Inputs
- Buffered Positive Edge-Triggered Clock
- Hysteresis on Clock Input to Improve Noise Margin
- Input Clamp Diodes Limit High Speed Termination Effects

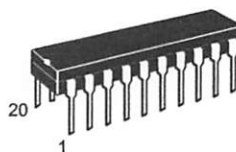
### GUARANTEED OPERATING RANGES

Symbol	Parameter	Min	Typ	Max	Unit
$V_{CC}$	Supply Voltage	4.75	5.0	5.25	V
$T_A$	Operating Ambient Temperature Range	0	25	70	°C
$I_{OH}$	Output Current – High			-2.6	mA
$I_{OL}$	Output Current – Low			24	mA

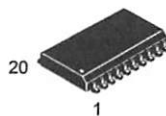


**ON Semiconductor**  
Formerly a Division of Motorola  
<http://onsemi.com>

### LOW POWER SCHOTTKY



PLASTIC  
DIP SUFFIX  
CASE 738



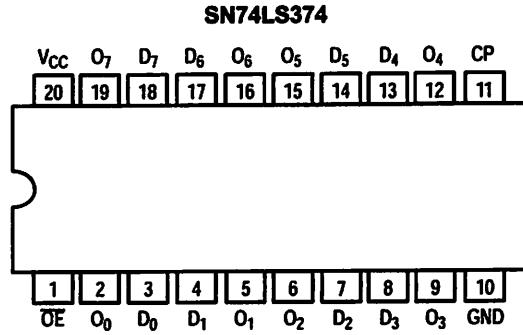
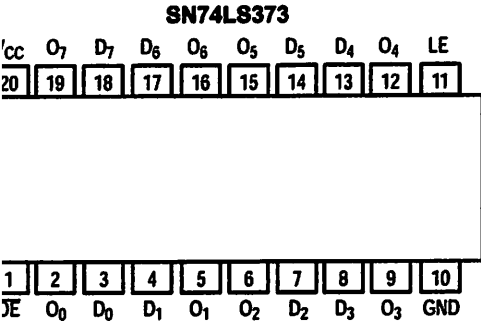
SOIC  
DW SUFFIX  
CASE 751D

### ORDERING INFORMATION

Device	Package	Shipping
SN74LS373N	16 Pin DIP	1440 Units/Box
SN74LS373DW	16 Pin	2500/Tape & Reel
SN74LS374N	16 Pin DIP	1440 Units/Box
SN74LS374DW	16 Pin	2500/Tape & Reel

# SN74LS373 SN74LS374

## CONNECTION DIAGRAM DIP (TOP VIEW)



**NOTE:**  
The Flatpak version has the same pinouts (Connection Diagram) as the Dual In-Line Package.

### PIN NAMES

D <sub>0</sub> - D <sub>7</sub>	Data Inputs
LE	Latch Enable (Active HIGH) Input
CP	Clock (Active HIGH Going Edge) Input
OE	Output Enable (Active LOW) Input
O <sub>0</sub> - O <sub>7</sub>	Outputs

### NOTES:

a) 1 TTL Unit Load (U.L.) = 40 μA HIGH/1.6 mA LOW.

### LOADING (Note a)

	HIGH	LOW
D <sub>0</sub> - D <sub>7</sub>	0.5 U.L.	0.25 U.L.
LE	0.5 U.L.	0.25 U.L.
CP	0.5 U.L.	0.25 U.L.
OE	0.5 U.L.	0.25 U.L.
O <sub>0</sub> - O <sub>7</sub>	65 U.L.	15 U.L.

### TRUTH TABLE

#### LS373

D <sub>n</sub>	LE	OE	O <sub>n</sub>
H	H	L	H
L	H	L	L
X	L	L	Q <sub>0</sub>
X	X	H	Z*

#### LS374

D <sub>n</sub>	LE	OE	O <sub>n</sub>
H		L	H
L		L	L
X	X	H	Z*

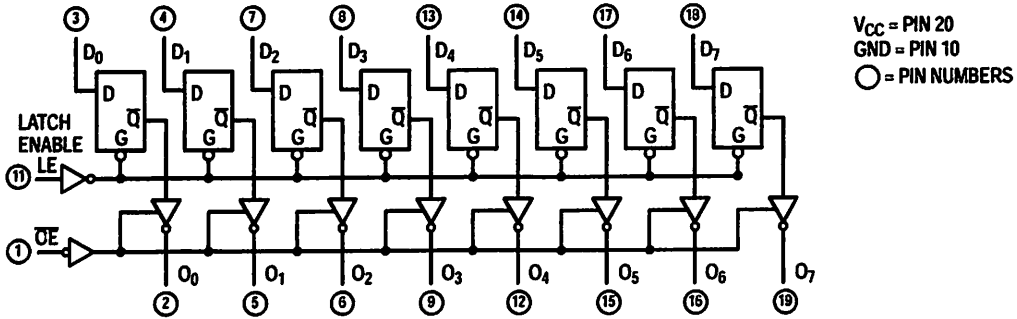
H = HIGH Voltage Level  
L = LOW Voltage Level  
X = Immaterial  
Z = High Impedance

\* Note: Contents of flip-flops unaffected by the state of the Output Enable input (OE).

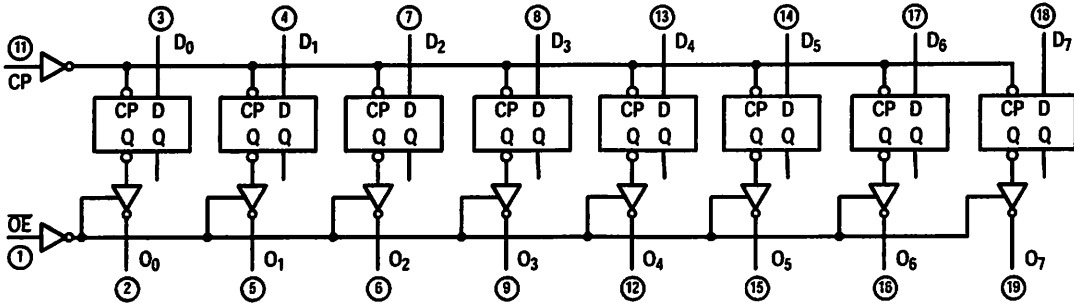
# SN74LS373 SN74LS374

## LOGIC DIAGRAMS

74LS373



74LS374



### CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
V <sub>IH</sub>	Input HIGH Voltage	2.0			V	Guaranteed Input HIGH Voltage for All Inputs
V <sub>IL</sub>	Input LOW Voltage			0.8	V	Guaranteed Input LOW Voltage for All Inputs
V <sub>IK</sub>	Input Clamp Diode Voltage		-0.65	-1.5	V	$V_{CC} = \text{MIN}$ , $I_{IN} = -18 \text{ mA}$
V <sub>OH</sub>	Output HIGH Voltage	2.4	3.1		V	$V_{CC} = \text{MIN}$ , $I_{OH} = \text{MAX}$ , $V_{IN} = V_{IH}$ or $V_{IL}$ per Truth Table
V <sub>OL</sub>	Output LOW Voltage		0.25	0.4	V	$I_{OL} = 12 \text{ mA}$
			0.35	0.5	V	$I_{OL} = 24 \text{ mA}$
I <sub>OZH</sub>	Output Off Current HIGH			20	$\mu\text{A}$	$V_{CC} = \text{MAX}$ , $V_{OUT} = 2.7 \text{ V}$
I <sub>OZL</sub>	Output Off Current LOW			-20	$\mu\text{A}$	$V_{CC} = \text{MAX}$ , $V_{OUT} = 0.4 \text{ V}$
I <sub>IH</sub>	Input HIGH Current			20	$\mu\text{A}$	$V_{CC} = \text{MAX}$ , $V_{IN} = 2.7 \text{ V}$
				0.1	mA	$V_{CC} = \text{MAX}$ , $V_{IN} = 7.0 \text{ V}$
I <sub>IL</sub>	Input LOW Current			-0.4	mA	$V_{CC} = \text{MAX}$ , $V_{IN} = 0.4 \text{ V}$
I <sub>SC</sub>	Short Circuit Current (Note 1)	-30		-130	mA	$V_{CC} = \text{MAX}$
I <sub>CC</sub>	Power Supply Current			40	mA	$V_{CC} = \text{MAX}$

Note 1: Not more than one output should be shorted at a time, nor for more than 1 second.



# SN74LS373 SN74LS374

## CHARACTERISTICS (T<sub>A</sub> = 25°C, V<sub>CC</sub> = 5.0 V)

Symbol	Parameter	Limits						Unit	Test Conditions
		LS373			LS374				
		Min	Typ	Max	Min	Typ	Max		
t <sub>MAX</sub>	Maximum Clock Frequency				35	50		MHz	C <sub>L</sub> = 45 pF, R <sub>L</sub> = 667 Ω
t <sub>PH</sub> t <sub>PL</sub>	Propagation Delay, Data to Output		12 12	18 18				ns	
t <sub>PH</sub> t <sub>PL</sub>	Clock or Enable to Output		20 18	30 30		15 19	28 28	ns	
t <sub>PH</sub> t <sub>PL</sub>	Output Enable Time		15 25	28 36		20 21	28 28	ns	
t <sub>HZ</sub> t <sub>Z</sub>	Output Disable Time		12 15	20 25		12 15	20 25	ns	C <sub>L</sub> = 5.0 pF

## SETUP REQUIREMENTS (T<sub>A</sub> = 25°C, V<sub>CC</sub> = 5.0 V)

Symbol	Parameter	Limits				Unit
		LS373		LS374		
		Min	Max	Min	Max	
	Clock Pulse Width	15		15		ns
	Setup Time	5.0		20		ns
	Hold Time	20		0		ns

## DEFINITION OF TERMS

**SETUP TIME (t<sub>s</sub>)** — is defined as the minimum time required for the correct logic level to be present at the logic input prior to LE transition from HIGH-to-LOW in order to be recognized and transferred to the outputs.

**HOLD TIME (t<sub>h</sub>)** — is defined as the minimum time following the LE transition from HIGH-to-LOW that the logic level must be maintained at the input in order to ensure continued recognition.

# SN74LS373 SN74LS374

## SN74LS373

### AC WAVEFORMS

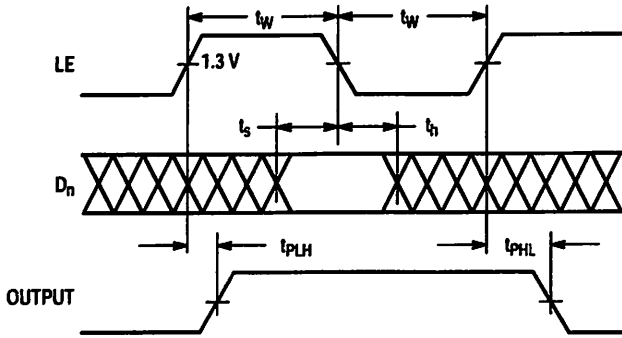


Figure 1.

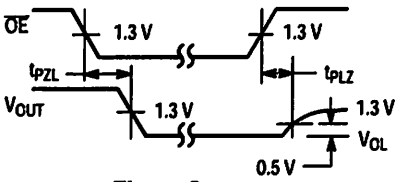


Figure 2.

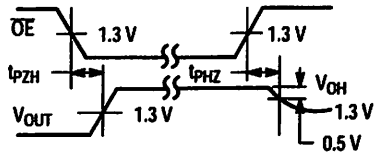
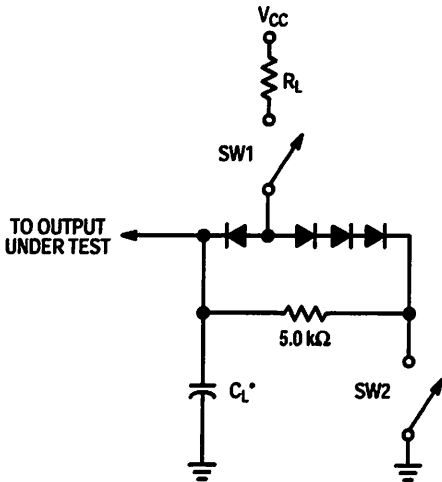


Figure 3.

### AC LOAD CIRCUIT



\* Includes Jig and Probe Capacitance.

Figure 4.

### SWITCH POSITIONS

SYMBOL	SW1	SW2
tPZH	Open	Closed
tPZL	Closed	Open
tPLZ	Closed	Closed
tPHZ	Closed	Closed

# SN74LS373 SN74LS374

## SN74LS374

### AC WAVEFORMS

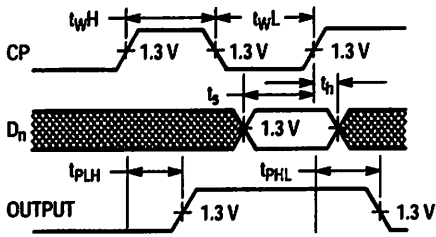


Figure 5.

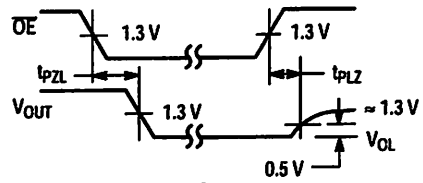


Figure 6.

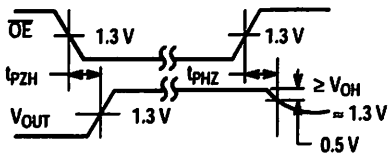
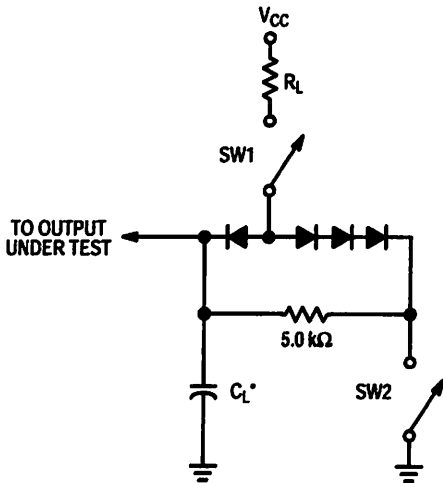


Figure 7.

### AC LOAD CIRCUIT



\* Includes Jig and Probe Capacitance.

Figure 8.

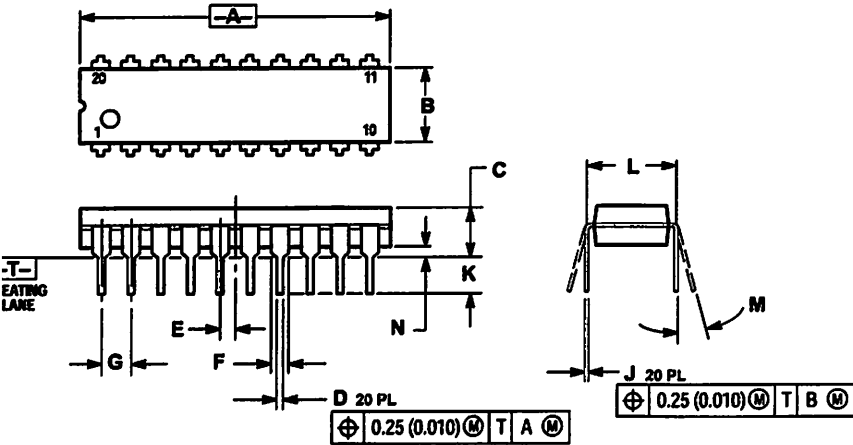
### SWITCH POSITIONS

SYMBOL	SW1	SW2
$t_{PZH}$	Open	Closed
$t_{PZL}$	Closed	Open
$t_{PLZ}$	Closed	Closed
$t_{PHZ}$	Closed	Closed

# SN74LS373 SN74LS374

## PACKAGE DIMENSIONS

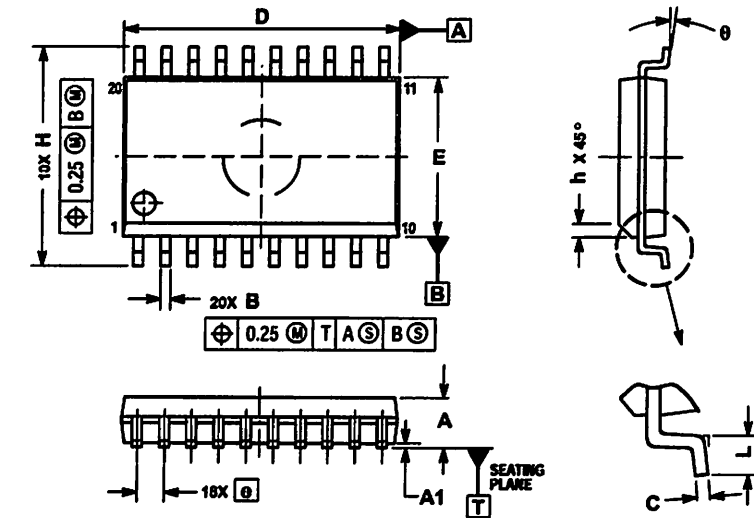
### N SUFFIX PLASTIC PACKAGE CASE 738-03 ISSUE E



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
  2. CONTROLLING DIMENSION: INCH.
  3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
  4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	1.010	1.070	25.66	27.17
B	0.240	0.260	6.10	6.60
C	0.150	0.180	3.81	4.57
D	0.015	0.022	0.39	0.55
E	0.050 BSC		1.27 BSC	
F	0.050	0.070	1.27	1.77
G	0.100 BSC		2.54 BSC	
J	0.008	0.015	0.21	0.39
K	0.110	0.140	2.80	3.55
L	0.300 BSC		7.62 BSC	
M	0°	15°	0°	15°
N	0.020	0.040	0.51	1.01

### D SUFFIX PLASTIC SOIC PACKAGE CASE 751D-05 ISSUE F



- NOTES:
1. DIMENSIONS ARE IN MILLIMETERS.
  2. INTERPRET DIMENSIONS AND TOLERANCES PER ASME Y14.5M, 1994.
  3. DIMENSIONS D AND E DO NOT INCLUDE MOLD PROTRUSION.
  4. MAXIMUM MOLD PROTRUSION 0.15 PER SIDE.
  5. DIMENSION B DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE PROTRUSION SHALL BE 0.13 TOTAL IN EXCESS OF B DIMENSION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS	
	MIN	MAX
A	2.35	2.65
A1	0.10	0.25
B	0.35	0.49
C	0.23	0.32
D	12.65	12.95
E	7.40	7.60
e	1.27 BSC	
H	10.05	10.55
h	0.25	0.75
L	0.50	0.90
θ	0°	7°

**DATASHEET TTL**  
**74LS164**

# SN74LS164

## Serial-In Parallel-Out Shift Register

The SN74LS164 is a high speed 8-Bit Serial-In Parallel-Out Shift Register. Serial data is entered through a 2-Input AND gate asynchronous with the LOW to HIGH transition of the clock. The device features an asynchronous Master Reset which clears the register setting all outputs LOW independent of the clock. It utilizes a Schottky diode clamped process to achieve high speeds and is fully compatible with all ON Semiconductor TTL products.

Typical Shift Frequency of 35 MHz

Asynchronous Master Reset

Gated Serial Data Input

Fully Synchronous Data Transfers

Input Clamp Diodes Limit High Speed Termination Effects

ESD > 3500 Volts

### GUARANTEED OPERATING RANGES

Symbol	Parameter	Min	Typ	Max	Unit
V <sub>CC</sub>	Supply Voltage	4.75	5.0	5.25	V
T <sub>A</sub>	Operating Ambient Temperature Range	0	25	70	°C
I <sub>OH</sub>	Output Current – High			-0.4	mA
I <sub>OL</sub>	Output Current – Low			8.0	mA

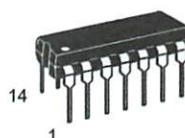


**ON Semiconductor**

Formerly a Division of Motorola

<http://onsemi.com>

**LOW  
POWER  
SCHOTTKY**



PLASTIC  
N SUFFIX  
CASE 646



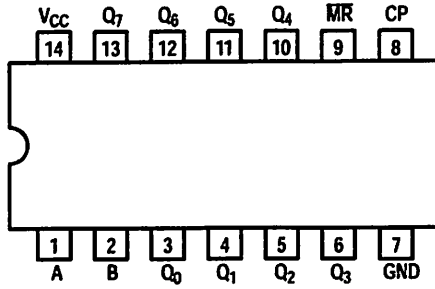
SOIC  
D SUFFIX  
CASE 751A

### ORDERING INFORMATION

Device	Package	Shipping
SN74LS164N	14 Pin DIP	2000 Units/Box
SN74LS164D	14 Pin	2500/Tape & Reel

# SN74LS164

## CONNECTION DIAGRAM DIP (TOP VIEW)



**NOTE:**  
The Flatpak version has the same pinouts (Connection Diagram) as the Dual In-Line Package.

### PIN NAMES

A, B	Data Inputs
CP	Clock (Active HIGH Going Edge) Input
MR	Master Reset (Active LOW) Input
Q <sub>0</sub> - Q <sub>7</sub>	Outputs

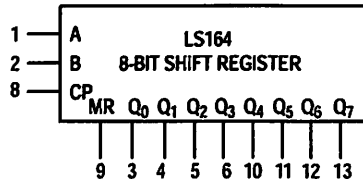
### LOADING (Note a)

HIGH	LOW
0.5 U.L.	0.25 U.L.
0.5 U.L.	0.25 U.L.
0.5 U.L.	0.25 U.L.
10 U.L.	5 U.L.

### NOTES:

a) 1 TTL Unit Load (U.L.) = 40  $\mu$ A HIGH/1.6 mA LOW.

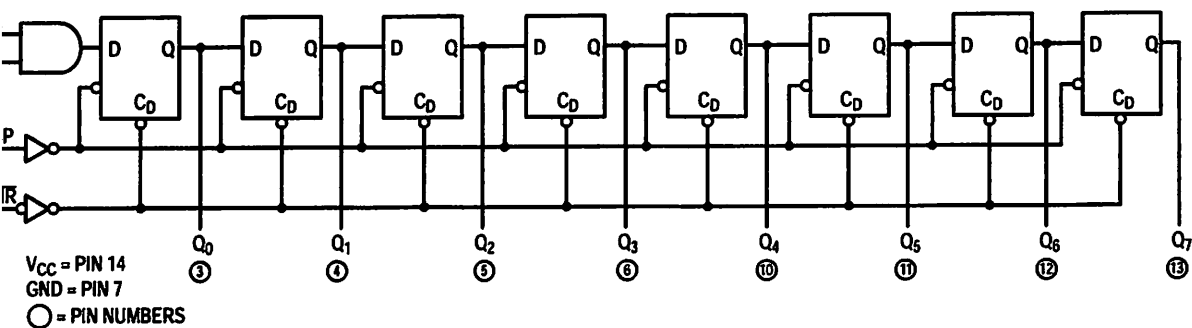
### LOGIC SYMBOL



V<sub>CC</sub> = PIN 14  
GND = PIN 7

# SN74LS164

## LOGIC DIAGRAM



## FUNCTIONAL DESCRIPTION

The LS164 is an edge-triggered 8-bit shift register with serial data entry and an output from each of the eight stages. Data is entered serially through one of two inputs (A or B); either of these inputs can be used as an active HIGH Enable for serial data entry through the other input. An unused input must be tied HIGH, or both inputs connected together.

Each LOW-to-HIGH transition on the Clock (CP) input shifts data one place to the right and enters into Q<sub>0</sub> the logical AND of the two data inputs (A•B) that existed before the rising clock edge. A LOW level on the Master Reset (MR) input overrides all other inputs and clears the register asynchronously, forcing all Q outputs LOW.

## MODE SELECT — TRUTH TABLE

OPERATING MODE	INPUTS			OUTPUTS	
	MR	A	B	Q <sub>0</sub>	Q <sub>1</sub> -Q <sub>7</sub>
Reset (Clear)	L	X	X	L	L - L
Shift	H	l	l	L	Q <sub>0</sub> - Q <sub>6</sub>
	H	l	h	L	Q <sub>0</sub> - Q <sub>6</sub>
	H	h	l	L	Q <sub>0</sub> - Q <sub>6</sub>
	H	h	h	H	Q <sub>0</sub> - Q <sub>6</sub>

L (l) = LOW Voltage Levels

H (h) = HIGH Voltage Levels

X = Don't Care

q<sub>n</sub> = Lower case letters indicate the state of the referenced input or output one set-up time prior to the LOW to HIGH clock transition.



# SN74LS164

## CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
	Input HIGH Voltage	2.0			V	Guaranteed Input HIGH Voltage for All Inputs
	Input LOW Voltage			0.8	V	Guaranteed Input LOW Voltage for All Inputs
	Input Clamp Diode Voltage		-0.65	-1.5	V	$V_{CC} = \text{MIN}$ , $I_{IN} = -18 \text{ mA}$
	Output HIGH Voltage	2.7	3.5		V	$V_{CC} = \text{MIN}$ , $I_{OH} = \text{MAX}$ , $V_{IN} = V_{IH}$ or $V_{IL}$ per Truth Table
L	Output LOW Voltage		0.25	0.4	V	$I_{OL} = 4.0 \text{ mA}$
			0.35	0.5	V	$I_{OL} = 8.0 \text{ mA}$
	Input HIGH Current			20	$\mu\text{A}$	$V_{CC} = \text{MAX}$ , $V_{IN} = 2.7 \text{ V}$
				0.1	mA	$V_{CC} = \text{MAX}$ , $V_{IN} = 7.0 \text{ V}$
	Input LOW Current			-0.4	mA	$V_{CC} = \text{MAX}$ , $V_{IN} = 0.4 \text{ V}$
	Short Circuit Current (Note 1)	-20		-100	mA	$V_{CC} = \text{MAX}$
	Power Supply Current			27	mA	$V_{CC} = \text{MAX}$

1: Not more than one output should be shorted at a time, nor for more than 1 second.

## CHARACTERISTICS ( $T_A = 25^\circ\text{C}$ )

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
MAX	Maximum Clock Frequency	25	36		MHz	$V_{CC} = 5.0 \text{ V}$ $C_L = 15 \text{ pF}$
L	Propagation Delay MR to Output Q		24	36	ns	
H L	Propagation Delay Clock to Output Q		17 21	27 32	ns	

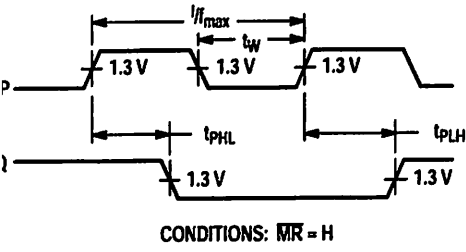
## SETUP REQUIREMENTS ( $T_A = 25^\circ\text{C}$ )

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
	CP, MR Pulse Width	20			ns	$V_{CC} = 5.0 \text{ V}$
	Data Setup Time	15			ns	
	Data Hold Time	5.0			ns	
	MR to Clock Recovery Time	20			ns	

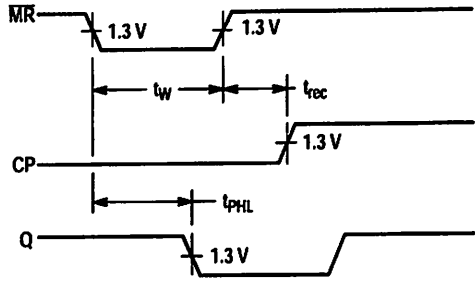
# SN74LS164

## AC WAVEFORMS

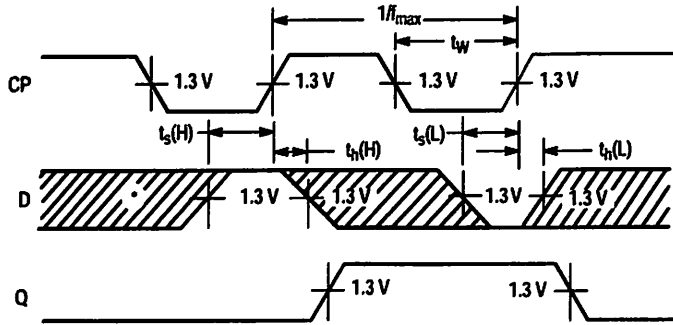
\*The shaded areas indicate when the input is permitted to change for predictable output performance.



**Figure 1. Clock to Output Delays and Clock Pulse Width**



**Figure 2. Master Reset Pulse Width, Master Reset to Output Delay and Master Reset to Clock Recovery Time**

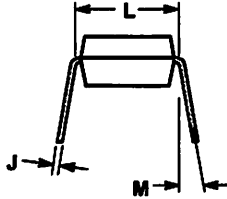
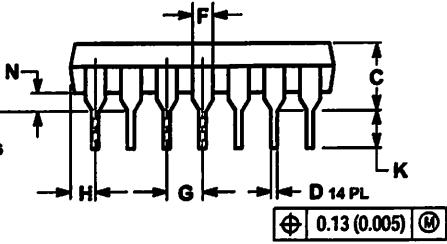
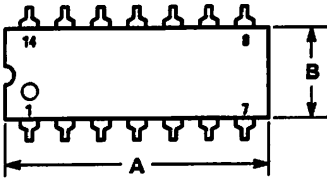


**Figure 3. Data Setup and Hold Times**

# SN74LS164

## PACKAGE DIMENSIONS

**N SUFFIX**  
**PLASTIC PACKAGE**  
**CASE 646-06**  
**ISSUE M**



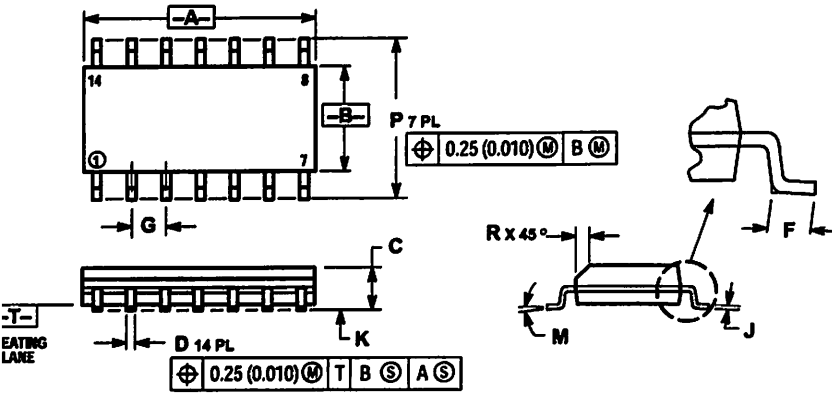
**NOTES:**

1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
5. ROUNDED CORNERS OPTIONAL.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.715	0.770	18.16	18.80
B	0.240	0.260	6.10	6.60
C	0.145	0.185	3.69	4.69
D	0.015	0.021	0.38	0.53
F	0.040	0.070	1.02	1.78
G	0.100 BSC		2.54 BSC	
H	0.052	0.095	1.32	2.41
J	0.008	0.015	0.20	0.38
K	0.115	0.135	2.92	3.43
L	0.290	0.310	7.37	7.87
M	— 10°		— 10°	
N	0.015	0.039	0.38	1.01

# SN74LS164

## D SUFFIX PLASTIC SOIC PACKAGE CASE 751A-03 ISSUE F



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
  2. CONTROLLING DIMENSION: MILLIMETER.
  3. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION.
  4. MAXIMUM MOLD PROTRUSION 0.15 (0.005) PER SIDE.
  5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.127 (0.005) TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	8.55	8.75	0.337	0.344
B	3.80	4.00	0.150	0.157
C	1.35	1.75	0.054	0.068
D	0.35	0.49	0.014	0.019
F	0.40	1.25	0.016	0.049
G	1.27 BSC		0.050 BSC	
J	0.19	0.25	0.008	0.009
K	0.10	0.25	0.004	0.009
M	0°	7°	0°	7°
P	5.80	6.20	0.228	0.244
R	0.25	0.50	0.010	0.019

**DATASHEET TRANSISTOR POWER  
TIP31**

## COMPLEMENTARY SILICON PLASTIC POWER TRANSISTORS

designed for use in general purpose power amplifier and switching applications.

### FEATURES:

Collector-Emitter Sustaining Voltage -

$V_{CE(sust)}$  = 40V(Min)- TIP31, TIP32

60V(Min)- TIP31A, TIP32A

80V(Min)- TIP31B, TIP32B

100V(Min)-TIP31C, TIP32C

Collector-Emitter Saturation Voltage-  $V_{CE(sat)}$  = 1.2V(Max) @  $I_C = 3.0 A$

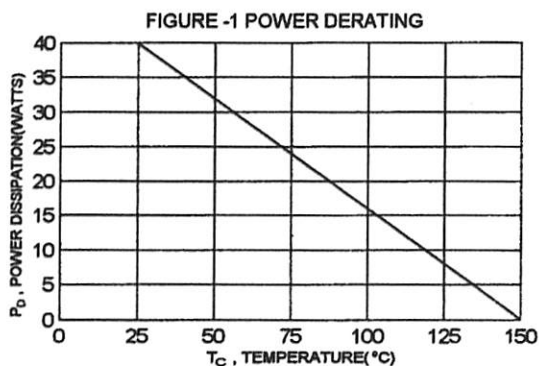
Current Gain-Bandwidth Product  $f_T = 3.0 MHz (Min) @ I_C = 500 mA$

### MAXIMUM RATINGS

Characteristic	Symbol	TIP31 TIP32	TIP31A TIP32A	TIP31B TIP32B	TIP31C TIP32C	Unit
Collector-Emitter Voltage	$V_{CE}$	40	60	80	100	V
Collector-Base Voltage	$V_{CB}$	40	60	80	100	V
Emitter-Base Voltage	$V_{EB}$	5.0				V
Collector Current - Continuous - Peak	$I_C$	3.0 5.0				A
Base Current	$I_B$	1.0				A
Power Dissipation @ $T_C = 25^\circ C$ Derate above $25^\circ C$	$P_D$	40 0.32				W W/ $^\circ C$
Operating and Storage Junction Temperature Range	$T_J, T_{STG}$	-65 to +150				$^\circ C$

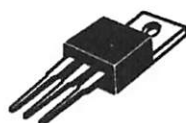
### TERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance Junction to Case	$R_{\theta jc}$	3.125	$^\circ C/W$

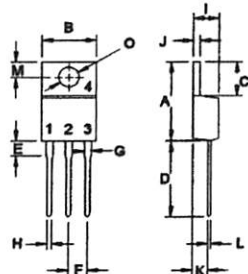


NPN	PNP
TIP31	TIP32
TIP31A	TIP32A
TIP31B	TIP32B
TIP31C	TIP32C

3 AMPERE  
COMPLEMENTARY SILICON  
POWER TRANSISTORS  
40 -100 VOLTS  
40 WATTS



TO-220



PIN 1.BASE  
2.COLLECTOR  
3.EMITTER  
4.COLLECTOR(CASE)

DIM	MILLIMETERS	
	MIN	MAX
A	14.68	15.31
B	9.78	10.42
C	5.01	6.52
D	13.06	14.62
E	3.57	4.07
F	2.42	3.66
G	1.12	1.36
H	0.72	0.86
I	4.22	4.98
J	1.14	1.38
K	2.20	2.97
L	0.33	0.55
M	2.48	2.98
O	3.70	3.90

TIP31, TIP31A, TIP31B, TIP31C NPN / TIP32, TIP32A, TIP32B, TIP32C PNP

ELECTRICAL CHARACTERISTICS (  $T_c = 25^\circ\text{C}$  unless otherwise noted )

Characteristic	Symbol	Min	Max	Unit
<b>DC CHARACTERISTICS</b>				
Collector-Emitter Sustaining Voltage(1) ( $I_C = 30\text{ mA}$ , $I_B = 0$ )	TIP31, TIP32 TIP31A, TIP32A TIP31B, TIP32B TIP31C, TIP32C	$V_{CE0(sus)}$	40 60 80 100	V
Collector Cutoff Current ( $V_{CE} = 30\text{ V}$ , $I_B = 0$ ) ( $V_{CE} = 60\text{ V}$ , $I_B = 0$ )	TIP31, TIP32, TIP31A, TIP32A TIP31B, TIP32B, TIP31C, TIP32C	$I_{C0}$	0.3 0.3	mA
Collector Cutoff Current ( $V_{CE} = 40\text{ V}$ , $V_{EB} = 0$ ) ( $V_{CE} = 60\text{ V}$ , $V_{EB} = 0$ ) ( $V_{CE} = 80\text{ V}$ , $V_{EB} = 0$ ) ( $V_{CE} = 100\text{ V}$ , $V_{EB} = 0$ )	TIP31, TIP32 TIP31A, TIP32A TIP31B, TIP32B TIP31C, TIP32C	$I_{C0}$	0.2 0.2 0.2 0.2	mA
Emitter Cutoff Current ( $V_{EB} = 5.0\text{ V}$ , $I_C = 0$ )		$I_{E0}$	1.0	mA

DC CHARACTERISTICS (1)

DC Current Gain ( $I_C = 1.0\text{ A}$ , $V_{CE} = 4.0\text{ V}$ ) ( $I_C = 3.0\text{ A}$ , $V_{CE} = 4.0\text{ V}$ )	$h_{FE}$	25 10	50	
Collector-Emitter Saturation Voltage ( $I_C = 3.0\text{ A}$ , $I_B = 375\text{ mA}$ )	$V_{CE(sat)}$		1.2	V
Base-Emitter On Voltage ( $I_C = 3.0\text{ A}$ , $V_{CE} = 4.0\text{ V}$ )	$V_{BE(on)}$		1.8	V

AC CHARACTERISTICS

Current Gain - Bandwidth Product (2) ( $I_C = 500\text{ mA}$ , $V_{CE} = 10\text{ V}$ , $f_{TEST} = 1\text{ MHz}$ )	$f_T$	3.0		MHz
Small Signal Current Gain ( $I_C = 500\text{ mA}$ , $V_{CE} = 10\text{ V}$ , $f = 1\text{ kHz}$ )	$h_{fe}$	20		

Pulse Test: Pulse width  $\leq 300\ \mu\text{s}$ , Duty Cycle  $\leq 2.0\%$

$$f_T = |h_{fe}| \cdot f_{TEST}$$

TIP31, TIP31A, TIP31B, TIP31C NPN / TIP32, TIP32A, TIP32B, TIP32C PNP

FIGURE 2 - SWITCHING TIME EQUIVALENT CIRCUIT

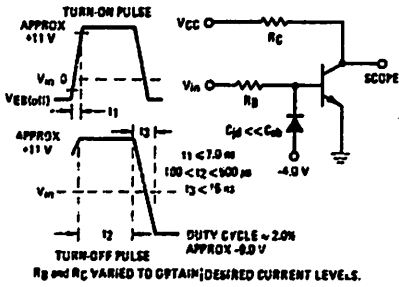


FIG-3 TURN-ON TIME

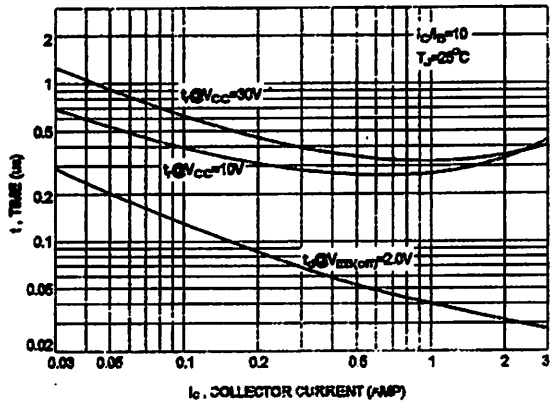


FIG-4 DC CURRENT GAIN

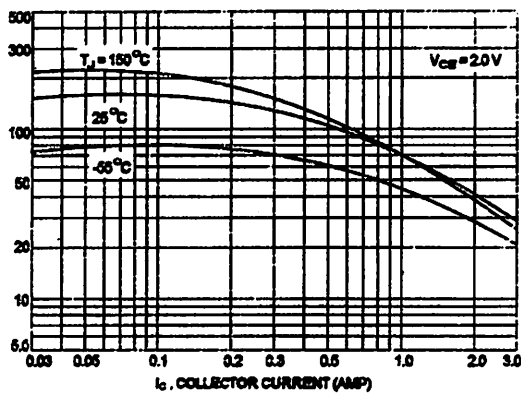


FIG-5 TURN-OFF TIME

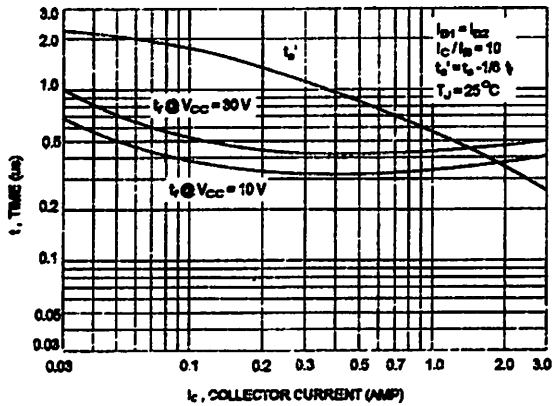
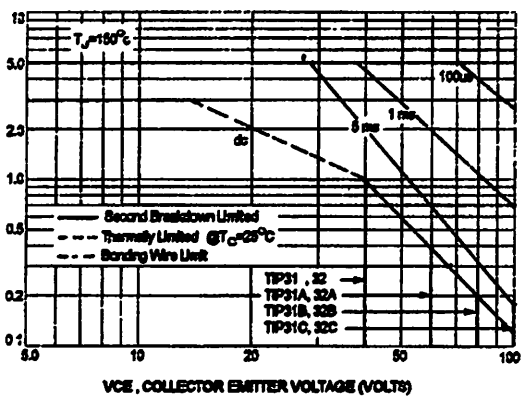


FIG-6 ACTIVE REGION SAFE OPERATING AREA



There are two limitation on the power handling ability of a transistor: average junction temperature and second breakdown safe operating area curves indicate  $I_C$ - $V_{CE}$  limits of the transistor that must be observed for reliable operation i.e., the transistor must not be subjected to greater dissipation than curves indicate.

The data of FIG-6 curve is base on  $T_{j,avg} = 150^\circ\text{C}$ ;  $T_C$  is variable depending on power level, second breakdown pulse limits are valid for duty cycles to 10% provided  $T_{j,avg} \leq 150^\circ\text{C}$ . At high case temperatures, thermal limitation will reduce the power that can be handled to values less than the limitations imposed by second breakdown.



TIP31, TIP31A, TIP31B, TIP31C NPN / TIP32, TIP32A, TIP32B, TIP32C PNP

FIG-7 COLLECTOR SATURATION REGION

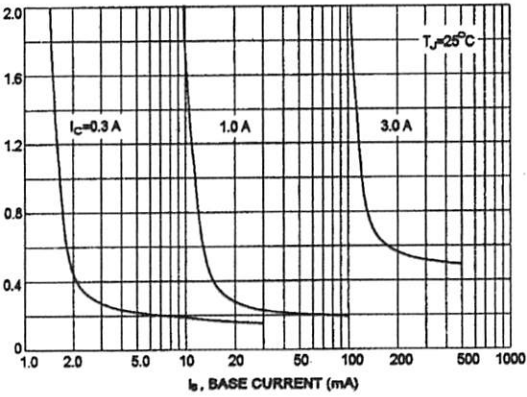


FIG-8 CAPACITANCES

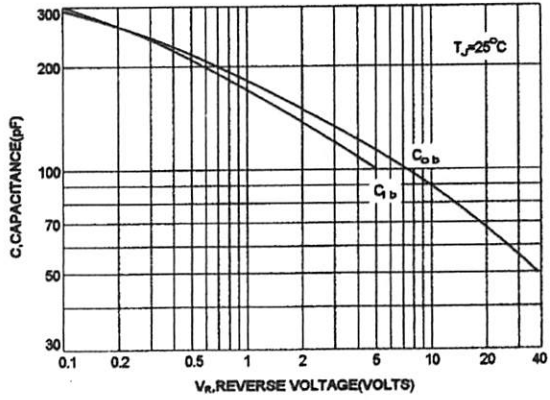


FIG-9 "ON" VOLTAGE

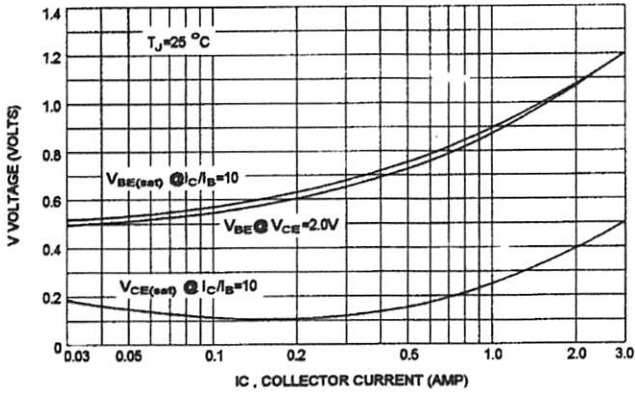
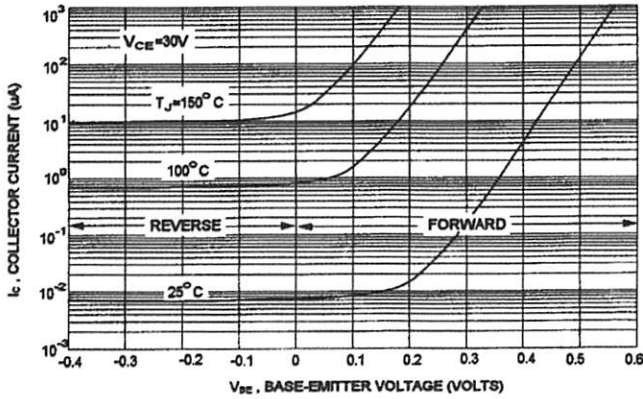


FIG-10 COLLECTOR CUT-OFF REGION



**DATASHEET IC REGULATOR  
LM7805**

# MC78XX/LM78XX/MC78XXA

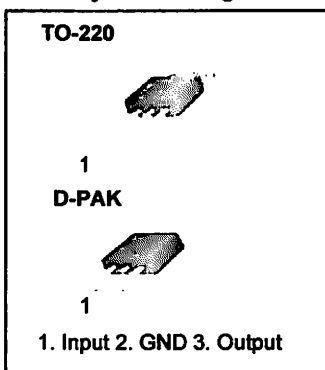
## Terminal 1A Positive Voltage Regulator

### Features

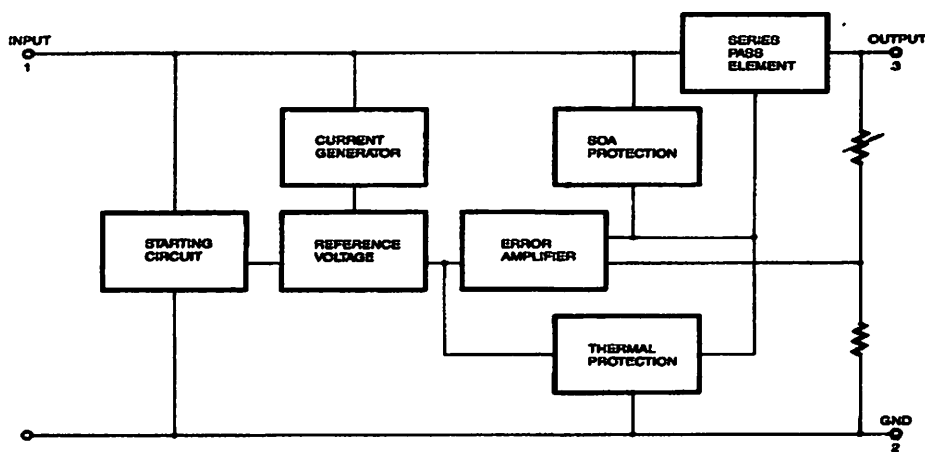
- Output Current up to 1A
- Output Voltages of 5, 6, 9, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

### Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



### Internal Block Diagram



Rev. 1.0.1

## Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$ )	$V_I$	35	V
or $V_O = 24V$ )	$V_I$	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}C/W$
Operating Temperature Range	TOPR	0 ~ +125	$^{\circ}C$
Storage Temperature Range	TSTG	-65 ~ +150	$^{\circ}C$

## Electrical Characteristics (MC7805/LM7805)

Refer to test circuit,  $0^{\circ}C < T_J < 125^{\circ}C$ ,  $I_O = 500mA$ ,  $V_I = 10V$ ,  $C_I = 0.33\mu F$ ,  $C_O = 0.1\mu F$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7805/LM7805			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V	
		$5.0mA \leq I_O \leq 1.0A$ , $P_O \leq 15W$ $V_I = 7V$ to $20V$	4.75	5.0	5.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}C$	$V_O = 7V$ to $25V$	-	4.0	100	mV
			$V_I = 8V$ to $12V$	-	1.6	50	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}C$	$I_O = 5.0mA$ to $1.5A$	-	9	100	mV
			$I_O = 250mA$ to $750mA$	-	4	50	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}C$	-	5.0	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5mA$ to $1.0A$	-	0.03	0.5	mA	
		$V_I = 7V$ to $25V$	-	0.3	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5mA$	-	-0.8	-	$mV / ^{\circ}C$	
Output Noise Voltage	$V_N$	$f = 10Hz$ to $100KHz$ , $T_A = +25^{\circ}C$	-	42	-	$\mu V / V_O$	
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB	
Dropout Voltage	$V_{Drop}$	$I_O = 1A$ , $T_J = +25^{\circ}C$	-	2	-	V	
Output Resistance	$r_O$	$f = 1KHz$	-	15	-	$m\Omega$	
Short Circuit Current	ISC	$V_I = 35V$ , $T_A = +25^{\circ}C$	-	230	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}C$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7806)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 11\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7806			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	5.75	6.0	6.25	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 8.0\text{V to } 21\text{V}$	5.7	6.0	6.3		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 8\text{V to } 25\text{V}$	-	5	120	mV
			$V_I = 9\text{V to } 13\text{V}$	-	1.5	60	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	9	120	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	3	60	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1\text{A}$	-	-	0.5	mA	
		$V_I = 8\text{V to } 25\text{V}$	-	-	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{kHz}$ , $T_A = +25^{\circ}\text{C}$	-	45	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 9\text{V to } 19\text{V}$	59	75	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	$r_O$	$f = 1\text{kHz}$	-	19	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7808)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 14\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7808			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	7.7	8.0	8.3	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 10.5\text{V to } 23\text{V}$	7.6	8.0	8.4		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 10.5\text{V to } 25\text{V}$	-	5.0	160	mV
			$V_I = 11.5\text{V to } 17\text{V}$	-	2.0	80	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5.0\text{mA to } 1.5\text{A}$	-	10	160	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	5.0	80	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1.0\text{A}$	-	0.05	0.5	mA	
		$V_I = 10.5\text{A to } 25\text{V}$	-	0.5	1.0		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{KHz}$ , $T_A = +25^{\circ}\text{C}$	-	52	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $V_I = 11.5\text{V to } 21.5\text{V}$	56	73	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	$r_O$	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7809)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 15\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7809			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	8.65	9	9.35	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 11.5\text{V to } 24\text{V}$	8.6	9	9.4		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 11.5\text{V to } 25\text{V}$	-	6	180	mV
			$V_I = 12\text{V to } 17\text{V}$	-	2	90	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	12	180	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	4	90	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	mA	
		$V_I = 11.5\text{V to } 26\text{V}$	-	-	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/°C	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{kHz}$ , $T_A = +25^{\circ}\text{C}$	-	58	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 13\text{V to } 23\text{V}$	56	71	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	$r_O$	$f = 1\text{kHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7810)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 16\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7810			Unit	
			Min.	Typ.	Max.		
Output Voltage	VO	$T_J = +25^{\circ}\text{C}$	9.6	10	10.4	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 12.5\text{V to } 25\text{V}$	9.5	10	10.5		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 12.5\text{V to } 25\text{V}$	-	10	200	mV
			$V_I = 13\text{V to } 25\text{V}$	-	3	100	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	12	200	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	4	400	
Quiescent Current	IQ	$T_J = +25^{\circ}\text{C}$	-	5.1	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	mA	
		$V_I = 12.5\text{V to } 29\text{V}$	-	-	1.0		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	VN	$f = 10\text{Hz to } 100\text{kHz}$ , $T_A = +25^{\circ}\text{C}$	-	58	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 13\text{V to } 23\text{V}$	56	71	-	dB	
Dropout Voltage	VDrop	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	rO	$f = 1\text{kHz}$	-	17	-	m $\Omega$	
Short Circuit Current	ISC	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.



**Electrical Characteristics (MC7812)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 19\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7812			Unit	
			Min.	Typ.	Max.		
Output Voltage	V <sub>O</sub>	T <sub>J</sub> = +25 °C	11.5	12	12.5	V	
		5.0mA ≤ I <sub>O</sub> ≤ 1.0A, P <sub>O</sub> ≤ 15W V <sub>I</sub> = 14.5V to 27V	11.4	12	12.6		
Line Regulation (Note1)	Regline	T <sub>J</sub> = +25 °C	V <sub>I</sub> = 14.5V to 30V	-	10	240	mV
			V <sub>I</sub> = 16V to 22V	-	3.0	120	
Load Regulation (Note1)	Regload	T <sub>J</sub> = +25 °C	I <sub>O</sub> = 5mA to 1.5A	-	11	240	mV
			I <sub>O</sub> = 250mA to 750mA	-	5.0	120	
Quiescent Current	I <sub>Q</sub>	T <sub>J</sub> = +25 °C	-	5.1	8.0	mA	
Quiescent Current Change	ΔI <sub>Q</sub>	I <sub>O</sub> = 5mA to 1.0A	-	0.1	0.5	mA	
		V <sub>I</sub> = 14.5V to 30V	-	0.5	1.0		
Output Voltage Drift	ΔV <sub>O</sub> /ΔT	I <sub>O</sub> = 5mA	-	-1	-	mV/°C	
Output Noise Voltage	V <sub>N</sub>	f = 10Hz to 100KHz, T <sub>A</sub> = +25 °C	-	76	-	μV/V <sub>O</sub>	
Ripple Rejection	RR	f = 120Hz V <sub>I</sub> = 15V to 25V	55	71	-	dB	
Dropout Voltage	V <sub>Drop</sub>	I <sub>O</sub> = 1A, T <sub>J</sub> = +25 °C	-	2	-	V	
Output Resistance	r <sub>O</sub>	f = 1KHz	-	18	-	mΩ	
Short Circuit Current	I <sub>SC</sub>	V <sub>I</sub> = 35V, T <sub>A</sub> = +25 °C	-	230	-	mA	
Peak Current	I <sub>PK</sub>	T <sub>J</sub> = +25 °C	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in V<sub>O</sub> due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7815)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 23\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7815			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	14.4	15	15.6	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 17.5\text{V to } 30\text{V}$	14.25	15	15.75		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 17.5\text{V to } 30\text{V}$	-	11	300	mV
			$V_I = 20\text{V to } 26\text{V}$	-	3	150	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	12	300	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	4	150	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.2	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	mA	
		$V_I = 17.5\text{V to } 30\text{V}$	-	-	1.0		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{kHz}$ , $T_A = +25^{\circ}\text{C}$	-	90	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 18.5\text{V to } 28.5\text{V}$	54	70	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	$r_O$	$f = 1\text{kHz}$	-	19	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7818)**

Refer to test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 27\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7818			Unit	
			Min.	Typ.	Max.		
Output Voltage	VO	$T_J = +25^{\circ}\text{C}$	17.3	18	18.7	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 21\text{V to } 33\text{V}$	17.1	18	18.9		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 21\text{V to } 33\text{V}$	-	15	360	mV
			$V_I = 24\text{V to } 30\text{V}$	-	5	180	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	15	360	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	5.0	180	
Quiescent Current	IQ	$T_J = +25^{\circ}\text{C}$	-	5.2	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	mA	
		$V_I = 21\text{V to } 33\text{V}$	-	-	1		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	VN	$f = 10\text{Hz to } 100\text{KHz}$ , $T_A = +25^{\circ}\text{C}$	-	110	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 22\text{V to } 32\text{V}$	53	69	-	dB	
Dropout Voltage	VDrop	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	rO	$f = 1\text{KHz}$	-	22	-	$\text{m}\Omega$	
Short Circuit Current	ISC	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7824)**

for test circuit,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 500\text{mA}$ ,  $V_I = 33\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7824			Unit	
			Min.	Typ.	Max.		
Output Voltage	VO	$T_J = +25^{\circ}\text{C}$	23	24	25	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 27\text{V to } 38\text{V}$	22.8	24	25.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 27\text{V to } 38\text{V}$	-	17	480	mV
			$V_I = 30\text{V to } 36\text{V}$	-	6	240	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	15	480	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	5.0	240	
Quiescent Current	IQ	$T_J = +25^{\circ}\text{C}$	-	5.2	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1.0\text{A}$	-	0.1	0.5	mA	
		$V_I = 27\text{V to } 38\text{V}$	-	0.5	1		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1.5	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	VN	$f = 10\text{Hz to } 100\text{KHz}$ , $T_A = +25^{\circ}\text{C}$	-	60	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 28\text{V to } 38\text{V}$	50	67	-	dB	
Dropout Voltage	VDrop	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	rO	$f = 1\text{KHz}$	-	28	-	m $\Omega$	
Short Circuit Current	ISC	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Notes:

Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7805A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 10\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	4.9	5	5.1	V	
		$I_O = 5\text{mA}$ to $1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 7.5\text{V}$ to $20\text{V}$	4.8	5	5.2		
Line Regulation (Note1)	Regline	$V_I = 7.5\text{V}$ to $25\text{V}$ $I_O = 500\text{mA}$	-	5	50	mV	
		$V_I = 8\text{V}$ to $12\text{V}$	-	3	50		
		$T_J = +25^{\circ}\text{C}$	$V_I = 7.3\text{V}$ to $20\text{V}$	-	5		50
			$V_I = 8\text{V}$ to $12\text{V}$	-	1.5		25
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA}$ to $1.5\text{A}$	-	9	100	mV	
		$I_O = 5\text{mA}$ to $1\text{A}$	-	9	100		
		$I_O = 250\text{mA}$ to $750\text{mA}$	-	4	50		
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.0	6	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA}$ to $1\text{A}$	-	-	0.5	mA	
		$V_I = 8\text{V}$ to $25\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$V_I = 7.5\text{V}$ to $20\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz}$ to $100\text{kHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 8\text{V}$ to $18\text{V}$	-	68	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	$r_O$	$f = 1\text{kHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7806A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 11\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	5.58	6	6.12	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 8.6\text{V to } 21\text{V}$	5.76	6	6.24		
Line Regulation (Note1)	Regline	$V_I = 8.6\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	5	60	mV	
		$V_I = 9\text{V to } 13\text{V}$	-	3	60		
		$T_J = +25^{\circ}\text{C}$	$V_I = 8.3\text{V to } 21\text{V}$	-	5		60
			$V_I = 9\text{V to } 13\text{V}$	-	1.5		30
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	9	100	mV	
		$I_O = 5\text{mA to } 1\text{A}$	-	4	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	5.0	50		
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	4.3	6	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1\text{A}$	-	-	0.5	mA	
		$V_I = 9\text{V to } 25\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$V_I = 8.5\text{V to } 21\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 9\text{V to } 19\text{V}$	-	65	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	$r_O$	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	ISC	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7808A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 14\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	VO	$T_J = +25^{\circ}\text{C}$	7.84	8	8.16	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 10.6\text{V to } 23\text{V}$	7.7	8	8.3		
Line Regulation (Note1)	Regline	$V_I = 10.6\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	6	80	mV	
		$V_I = 11\text{V to } 17\text{V}$	-	3	80		
		$T_J = +25^{\circ}\text{C}$	$V_I = 10.4\text{V to } 23\text{V}$	-	6		80
			$V_I = 11\text{V to } 17\text{V}$	-	2		40
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	12	100	mV	
		$I_O = 5\text{mA to } 1\text{A}$	-	12	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50		
Quiescent Current	I <sub>Q</sub>	$T_J = +25^{\circ}\text{C}$	-	5.0	6	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5\text{mA to } 1\text{A}$	-	-	0.5	mA	
		$V_I = 11\text{V to } 25\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$V_I = 10.6\text{V to } 23\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	V <sub>N</sub>	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 11.5\text{V to } 21.5\text{V}$	-	62	-	dB	
Dropout Voltage	V <sub>Drop</sub>	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r <sub>O</sub>	$f = 1\text{KHz}$	-	18	-	m $\Omega$	
Short Circuit Current	I <sub>SC</sub>	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	I <sub>PK</sub>	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7809A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 15\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	VO	$T_J = +25^{\circ}\text{C}$	8.82	9.0	9.18	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_{O} \leq 15\text{W}$ $V_I = 11.2\text{V to } 24\text{V}$	8.65	9.0	9.35		
Line Regulation (Note1)	Regline	$V_I = 11.7\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	6	90	mV	
		$V_I = 12.5\text{V to } 19\text{V}$	-	4	45		
		$T_J = +25^{\circ}\text{C}$	$V_I = 11.5\text{V to } 24\text{V}$	-	6		90
			$V_I = 12.5\text{V to } 19\text{V}$	-	2		45
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.0\text{A}$	-	12	100	mV	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	12	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50		
Quiescent Current	IQ	$T_J = +25^{\circ}\text{C}$	-	5.0	6.0	mA	
Quiescent Current Change	$\Delta I_Q$	$V_I = 11.7\text{V to } 25\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8	mA	
		$V_I = 12\text{V to } 25\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	VN	$f = 10\text{Hz to } 100\text{kHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 12\text{V to } 22\text{V}$	-	62	-	dB	
Dropout Voltage	VDrop	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2.0	-	V	
Output Resistance	$r_O$	$f = 1\text{kHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	ISC	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.



**Electrical Characteristics (MC7810A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 16\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	9.8	10	10.2	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 12.8\text{V to } 25\text{V}$	9.6	10	10.4		
Line Regulation (Note1)	Regline	$V_I = 12.8\text{V to } 26\text{V}$ $I_O = 500\text{mA}$	-	8	100	mV	
		$V_I = 13\text{V to } 20\text{V}$	-	4	50		
		$T_J = +25^{\circ}\text{C}$	$V_I = 12.5\text{V to } 25\text{V}$	-	8		100
			$V_I = 13\text{V to } 20\text{V}$	-	3		50
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	12	100	mV	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	12	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50		
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.0	6.0	mA	
Quiescent Current Change	$\Delta I_Q$	$V_I = 13\text{V to } 26\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.5	mA	
		$V_I = 12.8\text{V to } 25\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 14\text{V to } 24\text{V}$	-	62	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2.0	-	V	
Output Resistance	$r_O$	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7812A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 19\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	VO	$T_J = +25^{\circ}\text{C}$	11.75	12	12.25	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 14.8\text{V to } 27\text{V}$	11.5	12	12.5		
Line Regulation (Note1)	Regline	$V_I = 14.8\text{V to } 30\text{V}$ $I_O = 500\text{mA}$	-	10	120	mV	
		$V_I = 16\text{V to } 22\text{V}$	-	4	120		
		$T_J = +25^{\circ}\text{C}$	$V_I = 14.5\text{V to } 27\text{V}$	-	10		120
			$V_I = 16\text{V to } 22\text{V}$	-	3		60
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	12	100	mV	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	12	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50		
Quiescent Current	IQ	$T_J = +25^{\circ}\text{C}$	-	5.1	6.0	mA	
Quiescent Current Change	$\Delta I_Q$	$V_I = 15\text{V to } 30\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8	mA	
		$V_I = 14\text{V to } 27\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	VN	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 14\text{V to } 24\text{V}$	-	60	-	dB	
Dropout Voltage	VDrop	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2.0	-	V	
Output Resistance	rO	$f = 1\text{KHz}$	-	18	-	m $\Omega$	
Short Circuit Current	ISC	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7815A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 23\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	V <sub>O</sub>	T <sub>J</sub> = +25 °C	14.7	15	15.3	V	
		I <sub>O</sub> = 5mA to 1A, P <sub>O</sub> ≤ 15W V <sub>I</sub> = 17.7V to 30V	14.4	15	15.6		
Line Regulation (Note1)	Regline	V <sub>I</sub> = 17.9V to 30V I <sub>O</sub> = 500mA	-	10	150	mV	
		V <sub>I</sub> = 20V to 26V	-	5	150		
		T <sub>J</sub> = +25 °C	V <sub>I</sub> = 17.5V to 30V	-	11		150
			V <sub>I</sub> = 20V to 26V	-	3		75
Load Regulation (Note1)	Regload	T <sub>J</sub> = +25 °C I <sub>O</sub> = 5mA to 1.5A	-	12	100	mV	
		I <sub>O</sub> = 5mA to 1.0A	-	12	100		
		I <sub>O</sub> = 250mA to 750mA	-	5	50		
Quiescent Current	I <sub>Q</sub>	T <sub>J</sub> = +25 °C	-	5.2	6.0	mA	
Quiescent Current Change	ΔI <sub>Q</sub>	V <sub>I</sub> = 17.5V to 30V, T <sub>J</sub> = +25 °C	-	-	0.8	mA	
		V <sub>I</sub> = 17.5V to 30V, I <sub>O</sub> = 500mA	-	-	0.8		
		I <sub>O</sub> = 5mA to 1.0A	-	-	0.5		
Output Voltage Drift	ΔV/ΔT	I <sub>O</sub> = 5mA	-	-1.0	-	mV/°C	
Output Noise Voltage	V <sub>N</sub>	f = 10Hz to 100KHz T <sub>A</sub> = +25 °C	-	10	-	μV/V <sub>O</sub>	
Ripple Rejection	RR	f = 120Hz, I <sub>O</sub> = 500mA V <sub>I</sub> = 18.5V to 28.5V	-	58	-	dB	
Dropout Voltage	V <sub>Drop</sub>	I <sub>O</sub> = 1A, T <sub>J</sub> = +25 °C	-	2.0	-	V	
Output Resistance	r <sub>O</sub>	f = 1KHz	-	19	-	mΩ	
Short Circuit Current	I <sub>SC</sub>	V <sub>I</sub> = 35V, T <sub>A</sub> = +25 °C	-	250	-	mA	
Peak Current	I <sub>PK</sub>	T <sub>J</sub> = +25 °C	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in V<sub>O</sub> due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7818A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 27\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	17.64	18	18.36	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 21\text{V to } 33\text{V}$	17.3	18	18.7		
Line Regulation (Note1)	Regline	$V_I = 21\text{V to } 33\text{V}$ $I_O = 500\text{mA}$	-	15	180	mV	
		$V_I = 21\text{V to } 33\text{V}$	-	5	180		
		$T_J = +25^{\circ}\text{C}$	$V_I = 20.6\text{V to } 33\text{V}$	-	15		180
			$V_I = 24\text{V to } 30\text{V}$	-	5		90
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	15	100	mV	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	15	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	7	50		
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.2	6.0	mA	
Quiescent Current Change	$\Delta I_Q$	$V_I = 21\text{V to } 33\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8	mA	
		$V_I = 21\text{V to } 33\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 22\text{V to } 32\text{V}$	-	57	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2.0	-	V	
Output Resistance	$r_O$	$f = 1\text{KHz}$	-	19	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

**Electrical Characteristics (MC7824A)**

Refer to the test circuits.  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $I_O = 1\text{A}$ ,  $V_I = 33\text{V}$ ,  $C_I = 0.33\mu\text{F}$ ,  $C_O = 0.1\mu\text{F}$ , unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	
Output Voltage	$V_O$	$T_J = +25^{\circ}\text{C}$	23.5	24	24.5	V	
		$I_O = 5\text{mA to } 1\text{A}$ , $P_O \leq 15\text{W}$ $V_I = 27.3\text{V to } 38\text{V}$	23	24	25		
Line Regulation (Note1)	Regline	$V_I = 27\text{V to } 38\text{V}$ $I_O = 500\text{mA}$	-	18	240	mV	
		$V_I = 21\text{V to } 33\text{V}$	-	6	240		
		$T_J = +25^{\circ}\text{C}$	$V_I = 26.7\text{V to } 38\text{V}$	-	18		240
			$V_I = 30\text{V to } 36\text{V}$	-	6		120
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	15	100	mV	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	15	100		
		$I_O = 250\text{mA to } 750\text{mA}$	-	7	50		
Quiescent Current	$I_Q$	$T_J = +25^{\circ}\text{C}$	-	5.2	6.0	mA	
Quiescent Current Change	$\Delta I_Q$	$V_I = 27.3\text{V to } 38\text{V}$ , $T_J = +25^{\circ}\text{C}$	-	-	0.8	mA	
		$V_I = 27.3\text{V to } 38\text{V}$ , $I_O = 500\text{mA}$	-	-	0.8		
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5		
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.5	-	$\text{mV}/^{\circ}\text{C}$	
Output Noise Voltage	$V_N$	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = 25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ , $I_O = 500\text{mA}$ $V_I = 28\text{V to } 38\text{V}$	-	54	-	dB	
Dropout Voltage	$V_{\text{Drop}}$	$I_O = 1\text{A}$ , $T_J = +25^{\circ}\text{C}$	-	2.0	-	V	
Output Resistance	$r_O$	$f = 1\text{KHz}$	-	20	-	$\text{m}\Omega$	
Short Circuit Current	$I_{\text{SC}}$	$V_I = 35\text{V}$ , $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	$I_{\text{PK}}$	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

Load and line regulation are specified at constant junction temperature. Change in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

# Typical Performance Characteristics

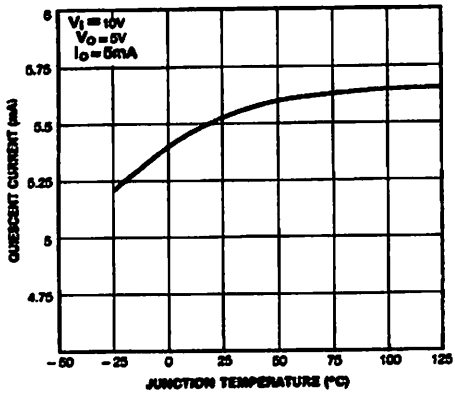


Figure 1. Quiescent Current

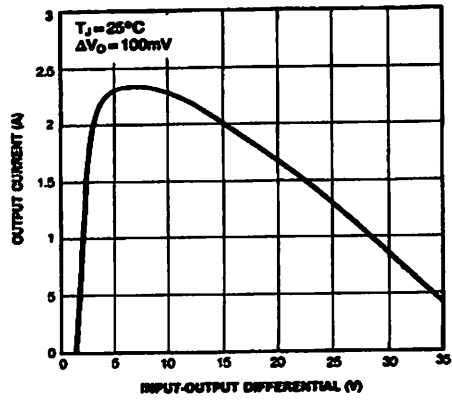


Figure 2. Peak Output Current

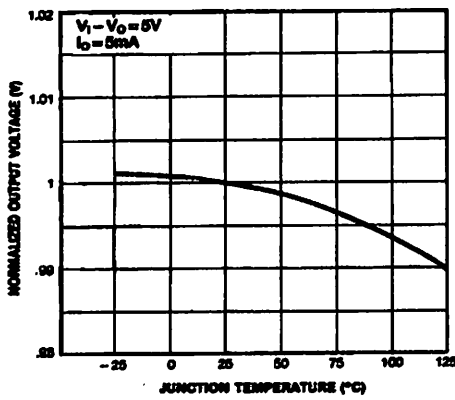


Figure 3. Output Voltage

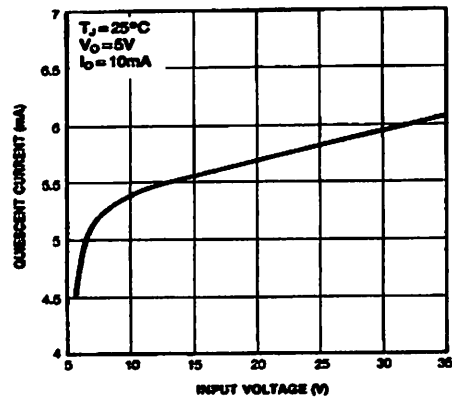


Figure 4. Quiescent Current

# Typical Applications

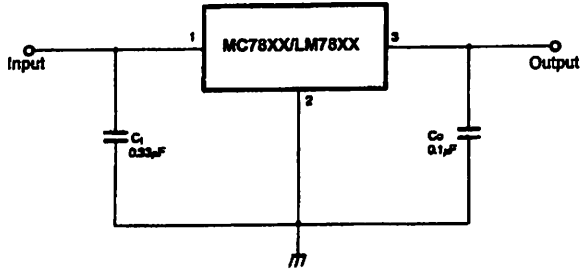


Figure 5. DC Parameters

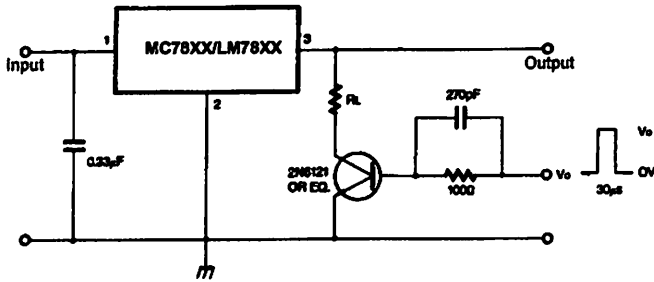


Figure 6. Load Regulation

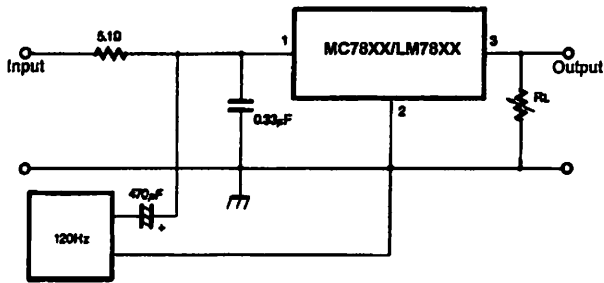


Figure 7. Ripple Rejection

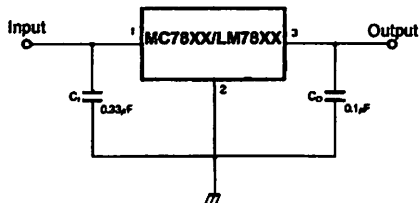


Figure 8. Fixed Output Regulator

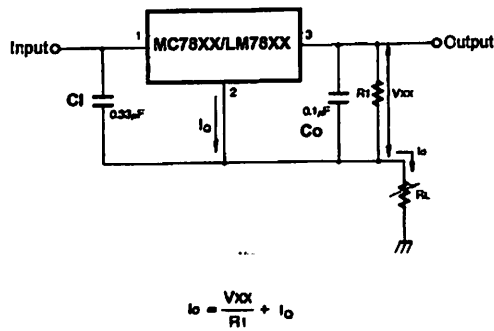


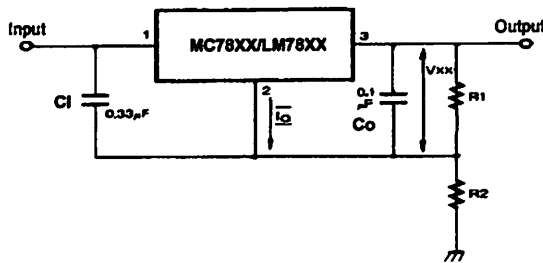
Figure 9. Constant Current Regulator

es:

To specify an output voltage, substitute voltage value for "XX." A common ground is required between the input and the Output voltage. The input voltage must remain typically 2.0V above the output voltage even during the low point on the input ripple voltage.

C1 is required if regulator is located an appreciable distance from power Supply filter.

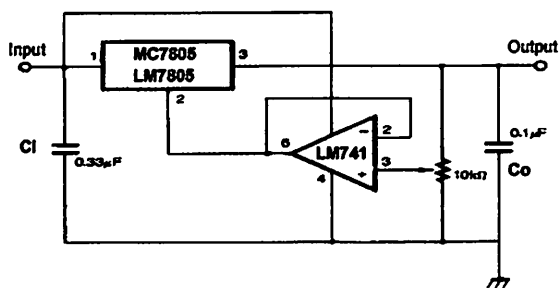
Co improves stability and transient response.



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{xx}(1+R_2/R_1)+I_Q R_2$$

Figure 10. Circuit for Increasing Output Voltage



$$I_{RI} \geq 5 I_Q$$

$$V_O = V_{xx}(1+R_2/R_1)+I_Q R_2$$

Figure 11. Adjustable Output Regulator (7 to 30V)



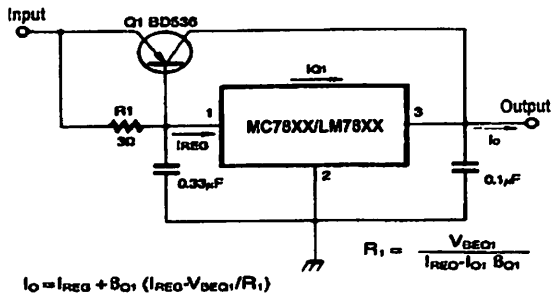


Figure 12. High Current Voltage Regulator

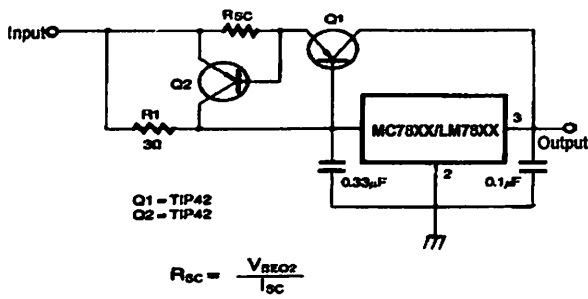


Figure 13. High Output Current with Short Circuit Protection

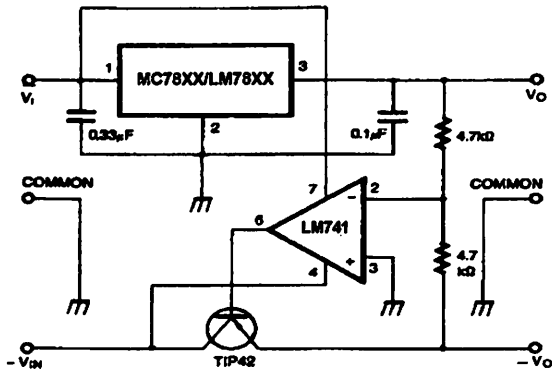


Figure 14. Tracking Voltage Regulator

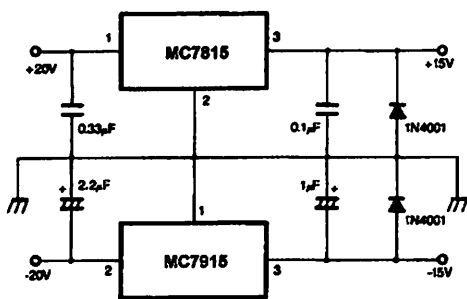


Figure 15. Split Power Supply ( $\pm 15V-1A$ )

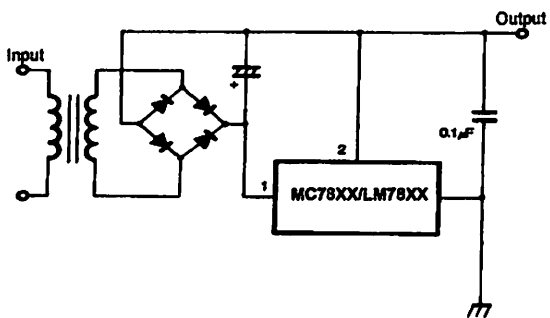


Figure 16. Negative Output Voltage Circuit

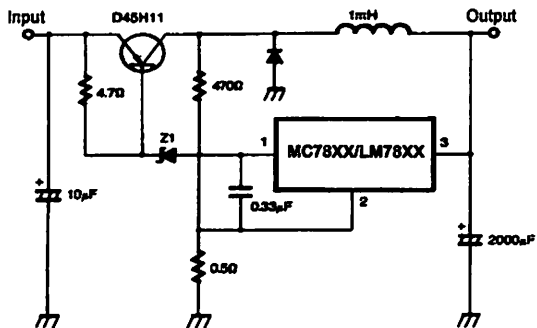
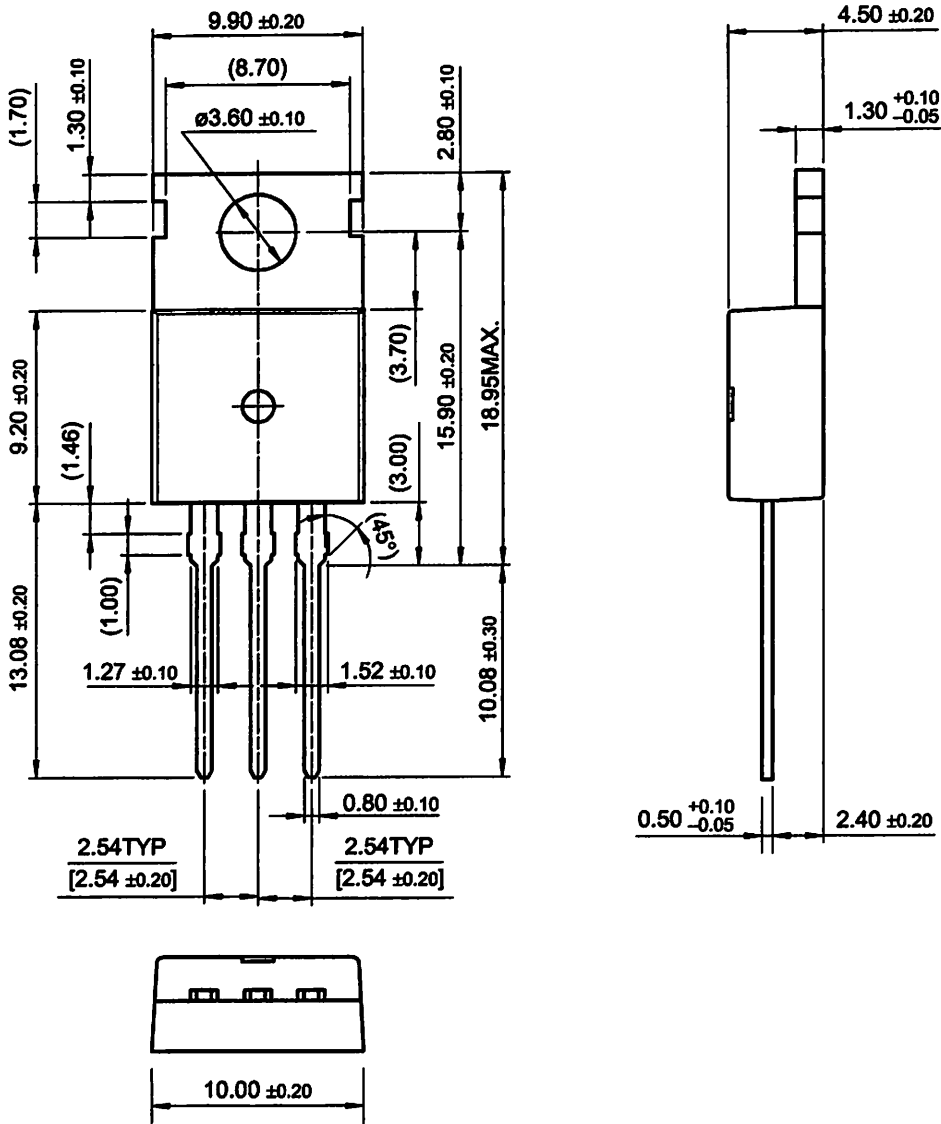


Figure 17. Switching Regulator

Mechanical Dimensions

Package

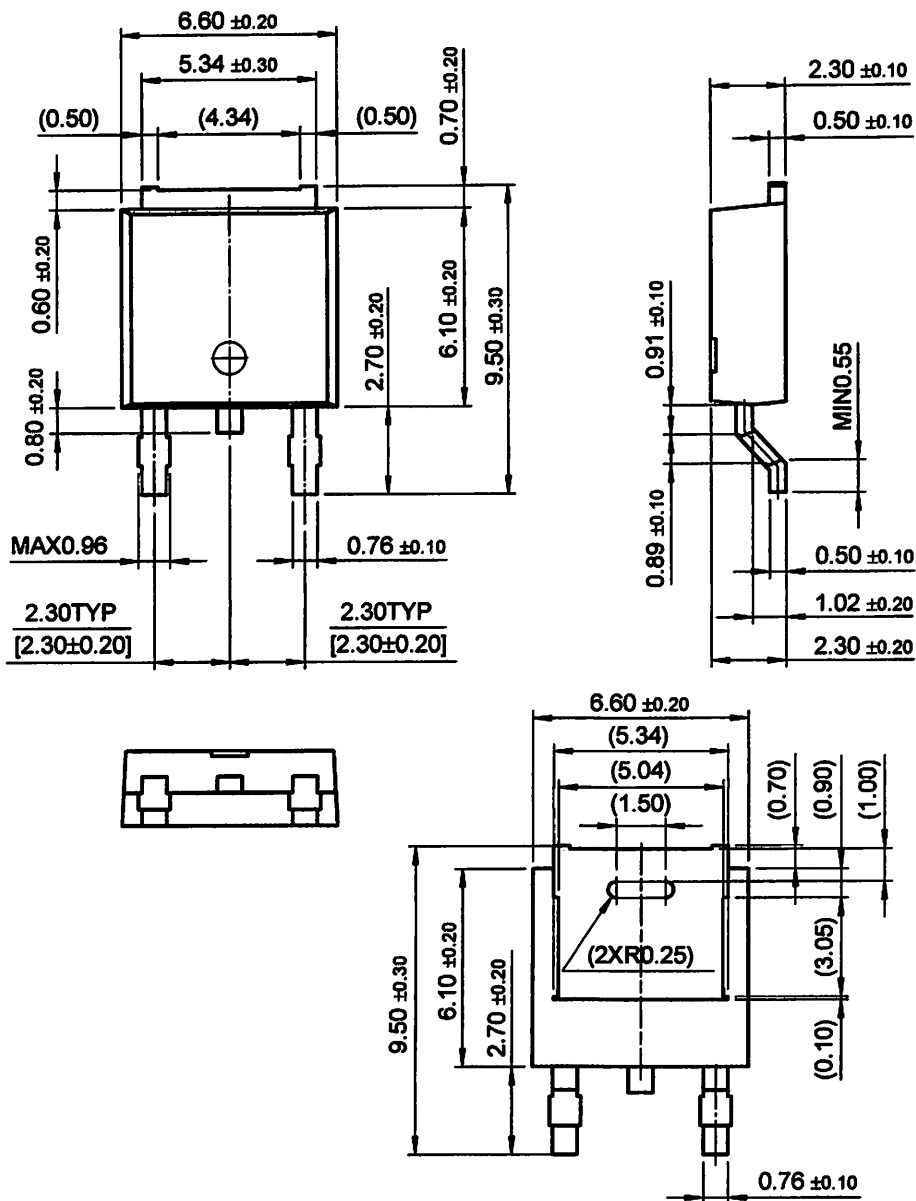
TO-220



Mechanical Dimensions (Continued)

Package

### D-PAK



## Ordering Information

Product Number	Output Voltage Tolerance	Package	Operating Temperature
LM7805CT	±4%	TO-220	0 ~ + 125°C

Product Number	Output Voltage Tolerance	Package	Operating Temperature
MC7805CT	±4%	TO-220	0 ~ + 125°C
MC7806CT			
MC7808CT			
MC7809CT			
MC7810CT			
MC7812CT			
MC7815CT			
MC7818CT		D-PAK	
MC7824CT			
MC7805CDT			
MC7806CDT			
MC7808CDT			
MC7809CDT			
MC7810CDT			
MC7812CDT			
MC7805ACT	±2%	TO-220	
MC7806ACT			
MC7808ACT			
MC7809ACT			
MC7810ACT			
MC7812ACT			
MC7815ACT			
MC7818ACT			
MC7824ACT			