

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



**PERENCANAAN DAN PEMBUATAN PROTOTYPE ROBOT  
YANG MENIRUKAN GERAKAN JARI TANGAN BERBASIS  
MIKROKONTROLER AT89S8252**

**SKRIPSI**

*Oleh:*

**Eko Yudik Hariyunantok  
NIM : 01.17.075**



**MARET 2007**

---

# LEMBAR PERSETUJUAN



## PERENCANAAN DAN PEMBUATAN PROTOTYPE ROBOT YANG MENIRUKAN GERAKAN JARI TANGAN BERBASIS MIKROKONTROLER AT89S8252

### SKRIPSI

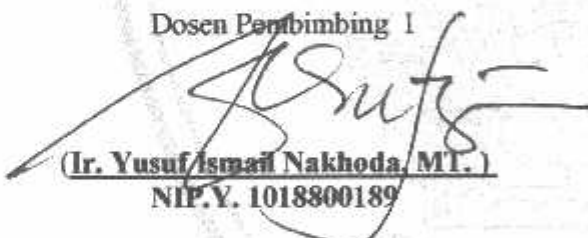
*Disusun Dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar  
Sarjana Teknik Elektronika Strata Satu (S1)*

Oleh :

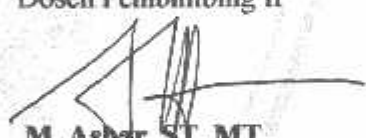
**EKO YUDIK HARIYUNANTOK**  
NIM : 01.17. 075

Diperiksa dan Disetujui

Dosen Pembimbing I

  
(Ir. Yusuf Ismail Nakhoda, MT.)  
NIP.Y. 1018800189

Dosen Pembimbing II

  
M. Ashar, ST, MT,  
NIP .

Ketua Jurusan Teknik Elektronika S-1

  
( Ir. F. Yudi Limpraptono )  
NIP.X. 1039500274

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

2007



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

BERITA ACARA UJIAN SKRIPSI  
FAKULTAS TEKNOLOGI INDUSTRI

1. Nama : Eko Yudik Hariyunantok
2. NIM : 01.17.075
3. Jurusan : Teknik Elektro S-1
4. Konsentrasi : Elektronika
5. Judul Skripsi : **Perencanaan dan Pembuatan Prototype Robot Yang Menirukan Gerakan Jari Tangan Berbasis Mikrokontroller AT89S8252**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1)

Pada :

Hari : Jum'at

Tanggal : 23 Maret 2007

Dengan Nilai : 81,25 (A) *ef*



Panitia Ujian Skripsi

Ketua

*[Signature]*  
Ir. Mochtar Asroni, MSME  
NIP.Y. 1018100036

Sekretaris

*[Signature]*  
Ir. F. Yudi Limpraptono, MT.  
NIP.Y. 1039500274

Anggota Penguji

Penguji I

*[Signature]*  
Ir. Usman Djuanda, MM  
NIP.Y. 1018700143

Penguji II

*[Signature]*  
Ir. Eko Nurcahyo  
NIP.Y. 1028700172

Eko Yudik Hy, 2007, "**Perancangan dan Pembuatan Prototype Robot yang Menirukan Gerakan Jari berbasis Mikrokontroller AT89S8252**" Teknik Elektro S-1 / Teknik Elektronika, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang. Dosen Pembimbing I : Ir. Yusuf Ismail Nakhoda, MT. Dosen Pembimbing II : M. Asar, ST, MT.

---

#### ABSTRAKSI

Kata kunci : Robot Tangan Tino, AT89S8252

Pembuatan robot ini Pengembangan teknologi baru bertujuan untuk memberikan fasilitas dan kemudahan kepada manusia dalam melakukan aktivitas kerja. Karena manusia memiliki keterbatasan dalam hal kemampuan fisiknya, sehingga dalam aktivitasnya memiliki batas-batas yang tertentu pula. Dengan demikian akan berpengaruh langsung pada tingkat efisiensi dari kerja manusia

Di dalam tugas akhir ini, dibahas tentang penerapan mikrokontroler AT89S8252 dalam perencanaan dan pembuatan prototype robot jari. Alat yang dibuat meliputi perencanaan perangkat keras dan perangkat lunak. Perencanaan perangkat keras meliputi: rangkaian sensor gerak potensiometer, rangkaian ADC, mikrokontroler dan rangkaian *driver* relay. Perencanaan perangkat lunak berupa *flowchart* cara kerja sistem. Kerja sistem bermula pada perubahan nilai resistansi pada potensiometer yang akan menentukan berapa lama motor bergerak dan menghasilkan perubahan sudut pada robot jari. Proses ini akan berulang selama ada sumber tegangan pada masing rangkain dan ada perubahan nilai pada potensio meter.

---

## KATA PENGANTAR

Dengan memanjatkan Puji syukur kehadiran Tuhan Yang Maha Esa, karena atas berkat dan anugrah-Nya penulis dapat menyelesaikan skripsi yang berjudul **“Perancangan dan Pembuatan Prototype Robot yang Menirukan Gerakan Jari Tangan Berbasis Mikrokontroller AT89S8252”**. Penyusunan skripsi merupakan syarat yang harus ditempuh mahasiswa jurusan Teknik Elektro S-1 Institut Teknologi Nasional Malang untuk memperoleh gelar Sarjana Teknik. Penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSFE selaku Rektor Institut Teknologi Nasional Malang.
  2. Bapak Ir. Mochtar Asroni, MSFE selaku Dekan Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.
  3. Bapak Ir. F. Yudi Limpraptono, MT selaku Kajar Teknik Elektro S1 di Institut Teknologi Nasional Malang.
  4. Bapak Ir. Yusuf Ismail Nakhoda, MT selaku Dosen pembimbing I
  5. Bapak M. Asar, ST, MT selaku Dosen Pembimbing II
  6. Kedua Orang Tua Penyusun yang telah banyak membantu dan memberikan dorongan serta doa restu pada penyusun dalam penyusunan skripsi ini.
  7. Semua teman-teman seperjuangan dan semua pihak yang telah membantu dalam proses penyusunan Skripsi ini.
-

Penulis menyadari bahwa skripsi ini mungkin kurang sempurna, sehingga penulis mengharapkan saran dan kritik kepada pembaca serta pengamat untuk pengembangan dan penyempurnaan, sehingga mendapatkan hasil yang lebih baik.

Malang, Maret 2007

**Penyusun**

---

## DAFTAR ISI

<b>LEMBAR PERSETUJUAN</b> .....	i
<b>KATA PENGANTAR</b> .....	ii
<b>ABSTRAKSI</b> .....	iii
<b>DAFTAR ISI</b> .....	iv
<b>DAFTAR GAMBAR</b> .....	vii
<b>DAFTAR TABEL</b> .....	viii
<b>BAB I PENDAHULUAN</b> .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	1
1.3. Tujuan Perencanaan .....	2
1.4. Batasan Masalah .....	2
1.5. Metodologi .....	2
1.6. Kontribusi .....	3
1.7. Sistem Penulisan .....	3
<b>BAB II DASAR TEORI</b> .....	5
2.1. Mikrokontroler AT89S8252 .....	5
2.1.1. Penjelasan Fungsi Pin AT89S8252 .....	8
2.1.2. Masukan dan Keluaran .....	10
2.1.3. Osilator .....	11

---

2.1.4. Reset .....	12
2.1.5. Data Memori (EEPROM) dan RAM .....	13
2.1.6. Special Function Register (SFR) .....	15
2.1.7. Timer dan Konter .....	17
2.1.8. Idle Mode .....	19
2.1.9. Sistem Interup .....	19
2.2. ADC (Analog To Digital Converter) .....	20
2.2.1. Successive Approximation .....	22
2.2.2. ADC 0808 .....	23
2.3. IC ULN 2004A .....	28
2.4. Relay .....	29
2.5. Motor DC .....	30
2.5.1. Cara Kerja Motor DC .....	32
2.6. Potensiometer .....	33
2.7. Aktuator .....	33
2.7.1. Kinematik dan Dinamik Robot .....	36
<b>BAB III PERANCANGAN DAN PEMBUATAN ALAT ...</b>	<b>39</b>
3.1. Pendahuluan .....	39
3.2. Perancangan dan Pembuatan Perangkat Keras .....	39
3.2.1. Diagram Blok .....	39
3.2.2. Mikrokontroler Sebagai Minimumsistem ...	41
3.2.3. Perencanaan ADC .....	42

---



3.2.4.	Drifer Motor DC .....	44
3.2.5.	Sensor Gerak ( Potensiometer) .....	45
3.2.6.	Perancangan Robot .....	46
3.3.	Perancangan Peranagkat Lunak .....	49
3.3.1.	Flowchart .....	51
 <b>BAB IV PENGUJIAN ALAT .....</b>		<b>52</b>
4.1.	Pendahuluan .....	52
4.2.	Pengujian Rangkaian Sensor .....	52
4.2.1.	Tujuan .....	52
4.2.2.	Peralatan yang Digunakan .....	52
4.2.3.	Prosedur Pengujian .....	53
4.3.	Pengujian Relay Pengerak Motor .....	56
4.3.1.	Tujuan .....	56
4.3.2.	Peralatan yang Digunakan .....	57
4.3.3.	Prosedur Pengujian .....	57
4.4.	Pengujian Rangkaian ADC .....	58
4.4.1.	Tujuan .....	58
4.4.2.	Peralatan yang Digunakan .....	58
4.4.3.	Prosedur Pengujian .....	58
4.5.	Pengujian Mekanik .....	64
4.6.	Pengujian Keseluruhan Alat .....	67
4.6.1.	Tujuan .....	67

---

<b>BAB V PENUTUP</b>	.....	<b>69</b>
5.1. Kesimpulan	.....	69
5.2. Saran-saran	.....	70
<b>DAFTAR PUSTAKA</b>	.....	<b>71</b>
<b>LAMPIRAN</b>		

---

## DAFTAR TABEL

<b>Tabel 2-1</b>	Fungsi Penganti Port 3 .....	11
<b>Tabel 2-2</b>	Fungsi Kusus pada Port 1 AT89S8252 .....	11
<b>Tabel 2-3</b>	Mode Operasi Timer/ Counter 0 dan 1 .....	18
<b>Tabel 2-4</b>	Mode IOPerasi Timer .....	18
<b>Tabel 2-4</b>	Alamat Sumber Interupsi .....	18
<b>Tabel 4-1</b>	Hasil Pengukuran Dan Perhitungan Potensiometer Sebagai Sensor Gerak .....	54
<b>Tabel 4-2</b>	Prosentase Kesalahan Antara Tegangan Hasil Pengukuran dan Perhitungan pada Potensiometer .....	56
<b>Tabel 4-3</b>	Hasil Pengujian Untuk Mengetahui Respon Driver Relay..	58
<b>Tabel 4-4</b>	Data Hasil Pengukuran Rangkaian Konversi ADC .....	61
<b>Tabel 4-5</b>	Data Hasil Pengukuran Rangkaian Konversi ADC .....	63
<b>Tabel 4-6</b>	Hasil Pengukuran Jarak Tempuh Lengan .....	65
<b>Tabel 4-7</b>	Prosentase Konversi Tegangan Output Sensor Gerak Terhadap Perubahan Sudut pada Robot .....	66

---

## DAFTAR GAMBAR

<b>Gambar</b>	<b>2-1</b>	Blok Diagram AT89S8252 .....	7
<b>Gambar</b>	<b>2-2</b>	Susunan Pin AT89S8252 .....	8
<b>Gambar</b>	<b>2-3</b>	Konfigurasi Osilator Megunakan Kristal ..	12
<b>Gambar</b>	<b>2-4</b>	Konfigurasi Osilator Megunakan External Oscilator Signal .....	12
<b>Gambar</b>	<b>2-5</b>	Rangkai Power On Reset .....	13
<b>Gambar</b>	<b>2-6</b>	Memori Data Internal (RAM).....	14
<b>Gambar</b>	<b>2-7</b>	AT89S8252 SFR Map dan Reset Value ..	15
<b>Gambar</b>	<b>2-8</b>	Struktur Pin-pin ADC 0808 .....	25
<b>Gambar</b>	<b>2-9</b>	Karakteristik Kerja ADC .....	28
<b>Gambar</b>	<b>2-10</b>	Pin-pin Koneksi Dalam IC ULN 2004A ..	29
<b>Gambar</b>	<b>2-11</b>	Jenis Relay .....	29
<b>Gambar</b>	<b>2-12</b>	Medan Yang Dhasilkan Oleh Kutup ..	31
<b>Gambar</b>	<b>2-13</b>	Medan Yang Dhasilkan Arus yang Mengalir pada Penghantar .....	32
<b>Gambar</b>	<b>2-14</b>	Interaksi Kedua Medan Menghasilkan Gaya ...	32
<b>Gambar</b>	<b>2-15</b>	Cara Kerja .....	32
<b>Gambar</b>	<b>2-16</b>	Potensiometer .....	34
<b>Gambar</b>	<b>2-17</b>	Diagran model Dinamik Robot .....	35
<b>Gambar</b>	<b>2-18</b>	Transformasi Kinematik Maju dan Kinematik Invers .....	35

---

<b>Gambar</b>	<b>2-19</b>	Konfigurasi Robot Tangan satu Sendi ....	37
<b>Gambar</b>	<b>2-20</b>	Konfigurasi Robot Tangan Planar 2 Sendi ....	38
<b>Gambar</b>	<b>3-1</b>	Diagram Sistem Prototype Robot yang Menirukan Gerakan Jari Tangan .....	40
<b>Gambar</b>	<b>3-2</b>	Rangkaian Minimum Sistem .....	42
<b>Gambar</b>	<b>3-3</b>	Rangkaian ADC 0808 .....	43
<b>Gambar</b>	<b>3-4</b>	Rangkaian Driver Relay .....	44
<b>Gambar</b>	<b>3-5</b>	Potensiometer .....	46
<b>Gambar</b>	<b>3-6</b>	Foto Robot Tangan Dua Sendi .....	47
<b>Gambar</b>	<b>3-7</b>	Sekema Rangkaian Kontrol Berbasis Mikrokontroler AT89S8252 .....	49
<b>Gambar</b>	<b>3-8</b>	Flowchart Program .....	51
<b>Gambar</b>	<b>4-1</b>	Diagram Blok Pengukuran Sensor Gerak Potensiometer .....	52
<b>Gambar</b>	<b>4-2</b>	Rangkaian Penguji Potensiometer .....	53
<b>Gambar</b>	<b>4-3</b>	Rangkaian Penguji Relay .....	57
<b>Gambar</b>	<b>4-4</b>	Rangkaian Penguji ADC 0808 .....	49

---

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Perkembangan teknologi Elektronika saat ini banyak di pengaruhi oleh keberadaan mikroprosesor dan mikrokontroler yaitu suatu istilah yang umum digunakan untuk mengabarkan komputer/program komputer yang mampu berfikir memecahkan masalah seperti yang dilakukan manusia.

Melalui berbagai keunggulan mikrokontroler sebagai sistem pengontrol dalam pengendalian, penulis mengaplikasikan pada pembuatan robot yang mampu bergerak seperti halnya manusia.

Oleh karena itu dalam tugas akhir ini penulis mencoba membuat robot yang mampu bergerak seperti manusia dan mampu menirukan gerakan-gerakan jari manusia yang terkontrol dan terkendali oleh sistem yang digunakan untuk menggantikan fungsi jari tangan, dan mengurangi resiko kecelakaan pada jari tangan yang melakukan pekerjaan beresiko pada jari jari manusia.

### **1.2. Rumusan Masalah**

Mengacu pada permasalahan yang telah di uraikan pada latar belakang sebelumnya, maka rumusan masalah di tekankan pada :

- a. Bagaimana merancang dan membuat robot jari yang menirukan gerak jari manusia.

- b. Bagaimana merancang dan membuat *Hardware* dan *Software* untuk mengatur kerja dari robot yang meirukan gerak jari.

### **1.3. Tujuan Perencanaan**

Tujuan dari penulisan tugas akhir merancang dan membuat prototype robot yang menirukan gerakan jari tangan yang berbasis mikrokontroler AT 89S8252.

### **1.4. Batasan Masalah**

Dalam penulisan tugas ahkir ini akan di berikan batasan-batasan masalah supaya tidak menyimpang dari tujuan .

Penulisan batasan-batan masalah tersebut meliputi :

- Pengendali utama dari robot yang direncanakan dan dibuat menggunakan mikrokontroler AT 89S8252 .
- Tidak membahas catu daya yang di gunakan
- Tidak membahas jenis motor yang digunakan.

### **1.5. Metodologi**

Metodologi penulisan tugas akhir ini adalah sebagai berikut :

- Studi literature, yaitu melakukan pencarian informasi dan data-data dari referensi-referensi yang berhubungan dengan alat seperti mikrokontroller dan potensiometer sebagai sensor gerak.

- Perancangan *hardware* dan *software*, yaitu melakukan kegiatan pembuatan program, perencanaan gambar rangkaian per-blok hingga pembuatan PCB rangkaian keseluruhan sampai melakukan perakitan komponen.
- Pengujian dan analisa, pada bagian ini melakukan uji coba *hardware* dan *software*, kemudian melakukan analisa berdasarkan hasil pengujian yang dilakukan.

### 1.6. Kontribusi

Alat ini direncanakan untuk kedepannya mampu menggantikan fungsi tangan manusia untuk melakukan pekerjaan yang dianggap berbahaya bagi kesehatan maupun fisik manusia, alat ini juga direncanakan kedepannya untuk melakukan pekerjaan manusia dengan jarak jauh.

### 1.7. Sistematika Penulisan

Untuk mempermudah dan memperjelas pembuatan laporan akhir ini, maka penulisan disusun secara sistematis sebagai berikut :

#### **BAB I      Pendahuluan**

Meliputi latar belakang, rumusan masalah, tujuan, metodologi, dan sistematika penulisan

#### **BAB II     Landasan Teori**

Meliputi uraian mengenai teori-teori yang mendukung perencanaan dan pembuatan alat



**BAB III Perancangan dan Pembuatan**

Meliputi penjelasan blok diagram serta perencanaan dan pembuatan *hardware* dan *software*

**BAB IV Pembahasan**

Meliputi prinsip kerja, pengujian, dan spesifikasi Robot

**BAB V Penutup**

Meliputi kesimpulan dan saran

## **BAB II LANDASAN TEORI**

Landasan teori ini sangat membantu untuk dapat memahami suatu sistem. Disamping itu dapat juga dijadikan sebagai bahan acuan didalam merencanakan suatu sistem. Dengan pertimbangan hal-hal tersebut maka landasan teori merupakan bagian yang harus dipahami untuk pembahasan selanjutnya.

### **2.1 Mikrokontroler AT89S8252**

Mikrokontroler AT89S8252 merupakan mikrokontroler 8 bit kompatibel dengan Standar industri MCS-51<sup>TM</sup> baik atas segi pemrograman maupun kaki tiap pin. Mikrokontroler AT89S8252 mempunyai 8 Kbyte (*Flash Programmable and Read Only Memori* ) pada dasarnya mikrokontroler adalah terdiri atas mikroprosesor, *timer*, dan *counter*, perangkat I/O dan internal memori. Mikrokontroler termasuk perangkat yang sudah didesain dalam *chip* tunggal.

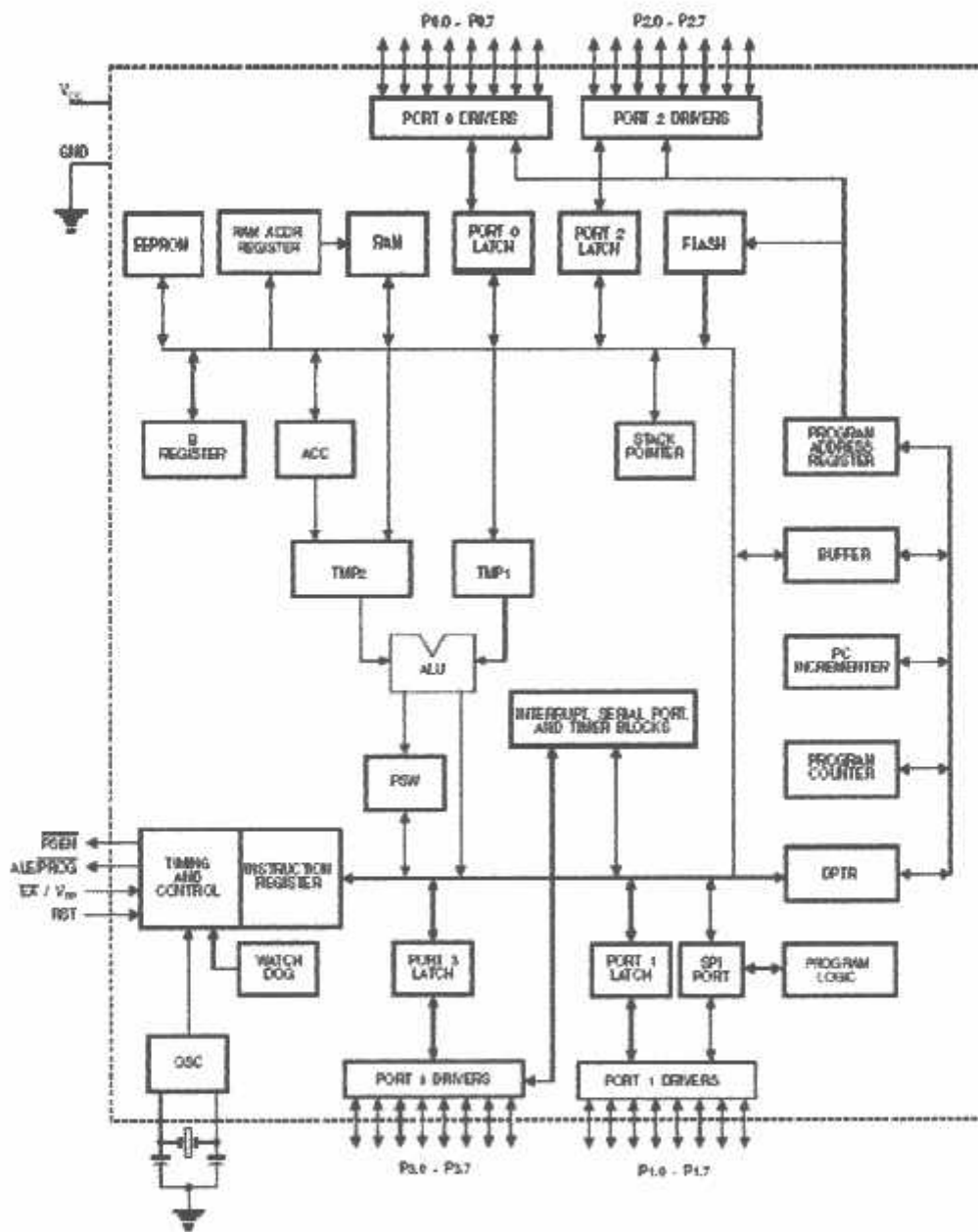
Pada dasarnya mikrokontroler mempunyai fungsi yang sama dengan mikroprosesor yaitu untuk mengontrol suatu kerja sistem. Selain itu mikrokontroler juga dikemas dalam satu *chip* (*single chip*). Didalam mikrokontroler juga terdapat CPU, ALU, PC, SP, dan register seperti dalam mikroprosesor, tetapi juga ditambah dengan perangkat-perangkat lain seperti ROM, RAM, PIO, SIO, *counter* dan sebuah rangkaian *clock*. Mikroprosesor didesain dengan intruksi-intruksi lebih luas dan 8 bit instruksi yang digunakan membaca data instruksi dari internal memori ke ALU. Sebagai suatu sistem control mikrokontroler bila dibandingkan dengan mikroprosesor memiliki kemampuan dan segi ekonomis yang bisa diandalkan karena dalam

mikrokontroler sudah terdapat RAM dan ROM. Sedangkan mikroprosesor didalamnya tidak terdapat keduanya. Terlihat bahwa mikrokontroler Atmel AT89S8252 memiliki banyak fitur yang menguntungkan. Dipakainya Downloadable flash memori memungkinkan mikrokontroler ini bekerja sendiri tanpa diperlukan tambahan *chip* lainnya. Sementara *Flash* memorinya mampu diprogram hingga seribu kali. Hal lain yang menguntungkan adalah system pemrograman menjadi lebih sederhana dan tidak memerlukan rangkaian yang rumit seperti rangkaian untuk memprogram produk Atmel lainnya. Secara umum konfigurasi yang dimiliki mikrokontroler AT89S8252 ad

- Sebuah *port* serial dengan control *full duplex* UART (Universal Asynchronous Receiver Transmitter).
- 256 byte RAM internal.
- 32 I/O yang dapat dipakai
- Sebuah CPU 8 bit dengan menggunakan teknologi dari Atmel.
- 8K byte Downloadable Flash Memori.
- 2K byte EEPROM
- 3 buah Timer/Counter 16 bit .
- SPI Serial Interface.
- Programmable Watchdog Timer .
- Dual Data Pointer.
- Frekuensi kerja 0 sampai 24 MHz
- Tegangan operasi 2,7 Volt sampai 6Volt.

- Kemampuan melaksanakan operasi perkalian, pembagian, dan operasi *Bolean* (bit)

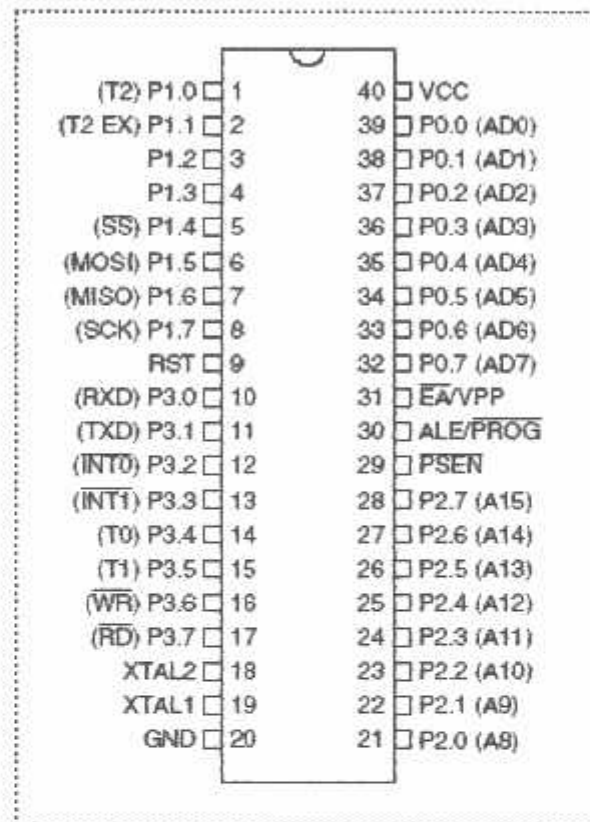
Sedangkan untuk blok diagram AT89S8252 diperlihatkan dalam gambar 2-1



Gambar 2-1 Blok Diagram AT89S8252

### 2.1.1. Penjelasan Fungsi Pin AT89S8252

Mikrktroler AT89S8252 mempunyai 40 pin seperti yang ditunjukkan dalam gambar 2-2 Fungsi-fungsi pin dijelaskan sebagai berikut :



Gambar 2-2 Susunan Pin AT89S8252<sup>[1]</sup>

- Pin 1 sampai 8

*Port* 1 yang terdiri atas pin 1 sampai 8 merupakan saluran masukan/keluaran dua arah.

- Pin 9

RST merupakan saluran dua masukan untuk *mereset* mikrokontroler dengan cara memberi masukan logika tinggi.

- Pin 10 sampai 17  
*Port 3* yang terdiri atas pin 10 sampai pin 17 merupakan saluran masukan/keluaran dua arah dan mempunyai fungsi khusus seperti yang terlihat dalam tabel 2-1
- Pin 18 dan 19  
 $XTAL_1$  dan  $XTAL_2$  merupakan saluran untuk mengatur pewaktuan sistem. Untuk pewaktuan dapat menggunakan pewaktuan internal maupun eksternal.
- Pin 20  
 $V_{ss}$  merupakan hubungan ke *ground* dari rangkaian.
- Pin 21 sampai 28  
*Port 2* yang terdiri atas pin 21 sampai 28 merupakan saluran masukan/keluaran dua arah. *Port* ini mengeluarkan 8 bit bagian alamat tinggi (A8-A15) selama pengambilan instruksi dari memori program eksternal dan pengambilan data memori eksternal yang menggunakan mode pengalamatan 16 bit.
- Pin 29  
 $PSEN$  (*Program Store Enable*) merupakan sinyal baca untuk mengaktifkan memori program eksternal.
- Pin 30  
 $ALE/PROG$  (*Address Latch Enable*) merupakan pulsa yang berfungsi untuk menahan alamat rendah (A0-A7) dalam *port 0*, selama proses baca/tulis memori eksternal. Frekuensi ALE adalah 1/6 kali frekuensi osilator, dan digunakan sebagai pewaktu. Pin ini juga berfungsi sebagai

saluran program selama dilakukan pemrograman jika menggunakan memori program eksternal.

- Pin 31

*EA/VPP (External Access Enable)* untuk mengatur penggunaan memori program eksternal dan internal. Pin ini harus dihubungkan dengan *ground* bila menggunakan memori program eksternal dan dihubungkan dengan *VPP* sebesar 12 Volt jika menggunakan memori program eksternal.

- Pin 32 sampai 39

*Port 0* yang terdiri atas pin 32 sampai 39 merupakan saluran masukan/keluaran. *Port 0* merupakan saluran alamat rendah (A0-A7) yang dimultipleks dengan saluran *bus data* (D0-D7).

- Pin 40

*Vcc* merupakan saluran masukan untuk catu daya positif sebesar 5 Volt DC dengan toleransi kurang lebih 10 %.

### 2.1.2. Masukan dan Keluaran

Untuk masukan dan keluaran terdapat 4 buah *port* yang masing-masing 8 bit. Saluran ini bersifat 2 arah (*bidirectional*) yang berarti dapat difungsikan sebagai masukan atau keluaran, serta dapat dialamati per bit. *Port3* selain digunakan sebagai *port* masukan dan keluaran juga dapat digunakan sebagai fungsi pengganti sebagaimana yang terdapat dalam tabel 2-1. Sedangkan AT89S8252 memiliki fitur tambahan yang terdapat pada *port 1* seperti pada tabel 2-2.

Tabel 2-1 Fungsi Pengganti Port 3<sup>(1)</sup>

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

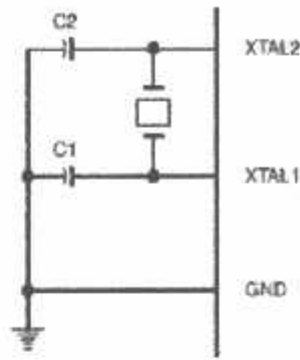
Tabel 2-2 Fungsi khusus pada port 1 AT89S8252<sup>(1)</sup>

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

### 2.1.3. Osilator

Jantung dari AT89S8252 adalah rangkaian yang membangkitkan pulsa clock yang mesinkronkan semua operasi internal. Mikrokontroler AT89S8252 memiliki osilator internal (*on chip oscillator*) yang dapat digunakan sebagai sumber pewaktu (*clock*) bagi CPU. Untuk menggunakan internal diperlukan sebuah kristal atau resonator keramik antara pin XTAL1 dan pin XTAL2 dan sebuah kapasitor ke *ground*. Konfigurasinya dapat dilihat pada gambar berikut.

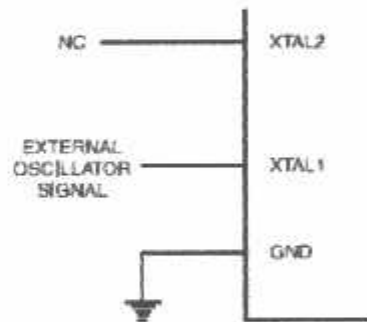




Gambar 2-3 Konfigurasi Osilator Menggunakan Kristal<sup>[1]</sup>

Nilai C1 dan C2 adalah 10 pF – 30 pF bila menggunakan kristal, dan bernilai 10 pF – 40 pF bila menggunakan resonator keramik.

Untuk penggunaan dengan external clock, XTAL2 harus dibiarkan dalam kondisi tidak terhubung. Konfigurasinya dapat dilihat pada gambar berikut ini :

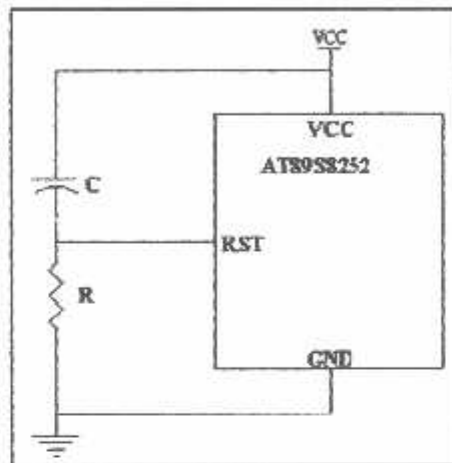


Gambar 2-4 Konfigurasi Osilator Menggunakan External Oscilator Signal<sup>[1]</sup>

#### 2.1.4. Reset

Rangkaian *power on reset* diperlukan untuk mereset mikrokontroler secara otomatis setiap catu daya *on*. Gambar 2-5 menunjukkan rangkaian *power on reset*.

Ketika catu daya diaktifkan, rangkaian *reset* menahan logika tinggi pin RST dengan jangka waktu yang ditentukan oleh besarnya pengisian muatan C.



Gambar 2-5 Rangkaian Power On Reset<sup>(1)</sup>

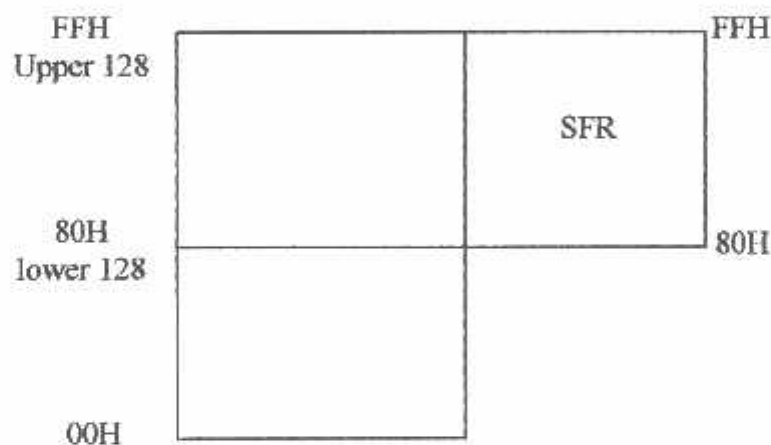
#### 2.1.5. Data Memori (EEPROM) Dan RAM

Berbeda dengan mikrokontroler standard MCS-51, mikrokontroler Atmel AT89S8252 juga dilengkapi dengan data memori yang berupa EEPROM (*Electrically Erasable Programmable Read Only Memory*). EEPROM yang dimaksud ini besarnya 2 *kilo byte* (2K) dan dipakai untuk penyimpanan data.

EEPROM *on-chip* ini diakses dengan mengeset bit EEMEN pada register WMCON pada alamat 96H. Alamat EEPROM ini adalah 000H sampai 7FFH. Intruksi *move* digunakan untuk mengakses EEPROM internal ini. Bit EEMWE pada register WMCON harus diset ke-1 sebelum seberang lokasi pada EEPROM dapat ditulisi. Program pengguna harus *mereset* bit EEMWE ke '0' jika proses penulisan ke EEPROM tidak diperlukan lagi. Proses penulisan ke EEPROM dapat

dilihat dengan membaca bit RDY/BSY pada SFR WMCON. Jika bit ini berlogika rendah maka berarti penulisan EEPROM sedang berlangsung, jika bit ini berlogika tinggi berarti penulisan sudah selesai dan penulisan lain dapat dimulai lagi.

Sedangkan RAM yang ada pada mikrokontroler AT89S8252 adalah berkapasitas 256 *byte* dan kompatibel dengan RAM yang ada pada mikrokontroler standard MCS-51.



Gambar 2-6. Memori Data internal (RAM) <sup>[11]</sup>

Pada *lower 128* lokasi memori dibagi menjadi 3 bagian :

1. Register bank 0 – 3

Lokasi bank register dimulai dari alamat 00H – 1 H yang terdiri dari 32 *bytes*. Register bank ini terdiri dari 4 buah register 8 bit yang dapat dipilih melalui pengaturan *program status word* register.

2. Bit Addressing

Terdiri dari 16 *bytes* yang dimulai dari 20H – 2FH. Masing-masing dari 128 bit lokasi ini dapat dialamati secara langsung yaitu dari 00H sampai 7FH.

### 3. *Scratch Pad Area*

Lokasi dari alamat 30H – 7FH atau sebanyak 80 *bytes* yang dapat digunakan sebagai alamat bagi RAM.

#### 2.1.6. Special Function Register (SFR)

Special Function Register merupakan register dengan tugas khusus. SFR pada mikrokontroler AT89S8252 kompatibel dengan mikrokontroler keluarga MCS-51 dan memiliki alamat 80H - FFH sehingga terdapat 128 lokasi alamat untuk SFR. Namun demikian pada mikrokontroler ini tidak berarti memiliki SFR sebanyak 128 buah. Berikut ini adalah gambar letak dari lokasi alamat SFR.

0F0H								0F0H	
0F0H	B 00000000							0F7H	
0E0H								0EFH	
0D0H	ACC 00000000							0E7H	
0C0H								0DFH	
0C0H	PSW 00000000					SPCR 00000100		0D7H	
0C0H	T2CON 00000000	T2MOD 00000000	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TMR 00000000		0CFH	
0C0H								0C7H	
0B0H	IP 00000000							0BFH	
0B0H	PI 11111111							0B7H	
0A0H	IE 00000000		SPCR 00000000					0A7H	
0A0H	IP2 11111111							0A7H	
0B0H	SCON 00000000	SBUF 00000000						0F0H	
001H	PI 11111111					WBCON 00000010		07H	
00H	PCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	T10 00000000	T10 00000000		0FH	
00H	PD 11111111	SP 00001111	DPOL 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR 00000000	PCON 00000000	07H

Gambar 2-7. AT89S8252 SFR Map dan Reset Value<sup>111</sup>

Selain itu mikrokontroler AT89S8252 memiliki tambahan SFR . Hal ini tak lain adalah karena terdapatnya tambahan fitur pada mikrokontroler ini. SFR tambahan ini meliputi : T2CON (Timer 2 Register dengan alamat 0C8H), T2MOD (Timer 2 Mode dengan alamat 0C9H), WMCON (*Watchdog and Memory Control Register* dengan alamat 96H), SPCR (*SPI Control Register* dengan alamat D5H), SPSR (*SPI Status Register* dengan alamat AAH), SPDR (*SPI Data Register* dengan alamat 86H).

- SFR untuk Timer 2

Mikrokontroler AT89S8252 terdapat tambahan sebuah Timer/Counter yang diberi nama timer 2 (sehingga AT89S8252 memiliki 3 Timer /Counter yaitu Timer/Counter 0, Timer/Counter 1, Timer/Counter 2). Pada Timer/Counter 2 ini dikendalikan oleh *Special Function Register* yang bernama T2CON (Timer 2 Control), T2MOD (Timer @ MODE) dan sepasang register RCAP2H, RCAP2L merupakan register *capture/reload* untuk Timer 2 dalam 16 bit *capture mode/auto reload mode*.

- SFR untuk *Watchdog* Memori.

Untuk menggunakan *Watchdog* timer atau memori, maka dapat dilakukan dengan mengatur SFR yang bernama WMCON dengan alamat 96H.

- SFR pengontrol SPI

Berbeda dengan mikrokontroler MCS-51, AT89S8252 memiliki fasilitas SPI (*Serial Peripheral Interface*). Fasilitas ini memungkinkan transfer data kecepatan tinggi secara sinkron antara mikrokontroler dengan peripheral atau antar mikrokontroler AT89S8252. Fitur ini meliputi :

- a. Full Duplex, 3 kawat dengan transfer data secara sinkron .
- b. Operasi Master atau Slave.
- c. Frekuensi maksimum 6 MHz.
- d. 4 bit rate terprogram.

#### 2.1.7. Timer dan Counter

Dalam mikrokontroler AT89S8252 terdapat tiga buah pewaktu/pencacah (timer/counter 16) 16 bit yang dapat diatur melalui perangkat lunak, yaitu pewaktu / pencacah 0 dan pewaktu / pencacah 1. Timer/counter ini diatur oleh *special function register* yaitu *Timer/Counter Control* (TCON alamat 88H), dan *Timer/Counter Mode Control* (TMOD alamat 89H). Selain itu nilai byte bawah dan byte atas dari Timer/Counter disimpan dalam register TL dan TH.

Jika difungsikan sebagai Timer, maka akan menggunakan system clock sebagai sumber masukan pulsanya. Jika sebagai Counter (pencacah), maka akan menggunakan pulsa dari luar (eksternal) sebagai masukan pulsanya. Pada Port 3 terdapat fungsi khusus yaitu TO (masukan luar untuk Timer/Counter 0) dan T1 (masukan luar untuk Timer/Counter 1). Pemilihan mode Timer/Counter dikontrol oleh register TMOD. Dengan memberikan nilai tertentu pada register TMOD dapat dipilih mode operasi untuk Timer/Counter 0 dan Timer/Counter 1 seperti terlihat dalam Tabel 2-3.

Tabel 2-3 Mode Operasi Timer/Counter 0 dan 1

M2	M1	Mode	Keterangan
0	0	0	13 bit Timer
0	1	1	16 bit Timer
1	0	2	8 bit auto-reload
1	1	3	Split Mode

Pada mikrokontroler AT89S8252 terdapat tambahan Timer 2. Timer yang lain adalah Timer 0 dan Timer 1. Timer 2 ini merupakan Timer/Counter 16 bit dan memiliki 3 mode operasi yaitu *capture*, *auto reload* (*up down counting*) dan *baud rate generator*. Untuk memilih mode ini dilakukan dengan mengatur bit pada SFR T2CON (Timer 2 Control Register). Timer 2 ini terdiri dari 2 buah timer 8 bit register yaitu TH2 dan TL2. pada fungsi Timer, register TL2 dinaikkan (*increment*) tiap siklus mesin. Karena siklus mesin terdiri dari 12 periode osilasi, maka *count rate* menjadi 1/12 dari frekuensi osilator. Sedangkan pada fungsi Counter, register dinaikkan berdasarkan tanggapan adanya transisi tinggi ke rendah pada pena yang bersesuaian (dalam hal ini pin T2 atau P1.0). Tabel berikut menunjukkan mode operasi yang dapat dijalankan pada timer 2.

Tabel 2-4 Mode Operasi Timer 2<sup>[11]</sup>

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)

RCLK = Receive clock enable. Jika diset menyebabkan serial port menggunakan pulsa overflow Timer 2 sebagai detak penerimaan pada serial port. Jika RCLK = 0 Timer 1 yang digunakan.

TCLK = Transmit clock enable. Jika diset menyebabkan serial port menggunakan pulsa overflow Timer 2 sebagai detak pengiriman. Jika TCLK = 0 pulsa overflow timer 1 yang digunakan.

CP/RL2 = Pemilihan capture/Reload. Jika diset maka proses capture yang terjadi sedangkan jika bit ini diclear maka proses reload.

TR2 = Bit untuk mengatur start/stop untuk timer 2 jika TR2 = 1 Timer akan aktif.

#### **2.1.8. Idle Mode**

Saat *Idle Mode* mikrokontroler tidak melakukan apa-apa. Tetapi peralatan lain yang terhubung tetap aktif. Kondisi ini dapat dihentikan dengan sebuah *interrupt* atau dengan *me-reset system*.

#### **2.1.9. Sistem Interrupt**

Mikrokontroler AT89S8252 mempunyai 6 buah sumber interrupt yang dapat membangkitkan permintaan interrupt, yaitu INTO, INT1, T0, T1, T2 dan port serial.

Saat terjadinya interupt, mikrokontroler secara otomatis akan menuju ke *sub rutin* pada alamat tersebut. Setelah interrupt service selesai dikerjakan, mikrokontroler akan mengerjakan program semula. Dua sumber interrupt *external* adalah INTO dan INT1, dimana kedua interupsi eksternal akan aktif atau aktif transisi tergantung isi dari IT0 dan IT1 pada register TCON. Interupsi T0, T1, T2



aktif pada saat timer yang sesuai mengalami *roll over*, interupsi serial dibangkitkan dengan melakukan operasi OR pada R1 dan T1. Tiap-tiap sumber interupsi dapat *enable* atau *disable* secara otomatis.

Tingkat prioritas semua sumber interupsi dapat diprogram sendiri-sendiri dengan set atau *clear bit* pada SFRS IP (*interrupt Priority*).

Tabel 2-5 Alamat Sumber Interupsi<sup>[1]</sup>

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

## 2.2. ADC (*Analog to Digital Converter*)

Mikrokontroler hanya dapat mengolah (memproses) data dalam bentuk *biner* saja, atau lebih sering disebut *digital*, oleh sebab itu setiap data *analog* yang akan diproses oleh mikrokontroler harus diubah terlebih dahulu kedalam bentuk kode *biner* (*digital*).

Jadi untuk menghubungkan sistem *analog* yang ada diluar mikrokontroller kedalam mikrokontroller, dibutuhkan suatu pengubah atau konversi *analog* ke *digital*. Fungsi dasar pengubah *analog* ke *digital* adalah mengubah tegangan *analog* kedalam bentuk kode-kode *biner* (*digital*) sehingga dapat diproses oleh mikrokontroller. Tegangan *analog* yang merupakan masukan dari ADC yang berasal dari sensor, rangkaian ini mempunyai *range* tertentu dan disebut *analog*. Kode *biner* hasil konversi ini dipakai data untuk diolah oleh mikrokontroller yang kemudian dilewatkan ke *driver* agar dapat mengendalikan *output-an*.

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

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

- **Resolution (Resolusi)**

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

- **Accuracy (Ketelitian)**

Adalah jumlah dari semua kesalahan, misalnya kesalahan *non linier*, skala penuh, skala *not* dan lain-lain. Dapat juga menyatakan perbedaan antara tegangan *input analog* secara teoritis yang dibutuhkan untuk menghasilkan suatu kode *biner* tertentu terhadap tegangan *input* nyata yang menghasilkan tegangan kode *biner* tersebut.

- **Waktu Konversi**

Waktu yang dibutuhkan untuk mengkonversikan *analog* ke *digital* setiap sample atau waktu yang diperlukan untuk menyelesaikan suatu konversi. Disini tidak akan dijelaskan keseluruhan dari jenis ADC, tetapi hanya dititik beratkan pada ADC jenis *Successive Approximation*. Karena dalam perencanaan pembuatan alat digunakan ADC jenis ini. *Successive Approximation* ADC merupakan golongan ADC *medium*. ADC tipe ini dapat dikatakan merupakan perpaduan yang baik antara kecepatan (kecepatan menengah) dan tingkat kerumitan rangkaian (menengah). Waktu konversi ADC ini selalu tetap, tidak tergantung pada besarnya sinyal *analog (input)*, tetapnya waktu konversi ini merupakan keuntungan penggunaan ADC tipe ini.

### 2.2.1. *Successive Approximation*

ADC jenis ini dibentuk dari beberapa komponen DAC (*Digital to Analog Converter*), *Comparator* dan beberapa *register* serta rangkaian kontrol.

Teknik konversi yang dipakai adalah dengan melakukan perbandingan antara tegangan *input* yang tidak diketahui dengan tegangan *output* yang dihasilkan oleh DAC.

DAC diatur oleh *shift register* dan rangkaian logika untuk kontrol dan hasil konversi dapat diambil dari *register* keluaran. Konversi dimulai dengan memberikan sinyal *input start conversion*. Dengan sinyal *start conversion* tersebut mengakibatkan *input DAC* akan dibuat aktif sehingga MSB sama dengan 1 dan *bit* yang lain berharga 0. *Output DAC* akan terjadi setengah *full scale* dibandingkan dengan *input analog yang dikonversi*. Dari *input* komparator dapat diketahui mana yang lebih besar, bila tegangan *output DAC* lebih besar maka *bit* MSB tetap, sedang bila *output DAC* lebih kecil maka MSB berubah menjadi 0. Kemudian *bit* berikutnya (*bit* lebih rendah dari MSB) dibuat sama dengan 1 dengan keadaan MSB tetap seperti proses perbandingan pertama tadi. Dari keadaan *input digital* ini akan dihasilkan *output analog* (dari DAC) yang juga akan dibandingkan lagi dengan tegangan *input* untuk menentukan lagi apakah keadaan *bit* kedua akan berubah atau tidak. Proses diatas berlangsung terus sampai LSB, sehingga diperoleh *output* dari *input DAC* yang terakhir. Proses pergeseran *bit* dari MSB sampai LSB dilakukan oleh *shift register* yang waktunya diatur oleh pulsa *clock*.

Jika proses konversi telah selesai, *output* status *bit* akan mengeluarkan "End of Conversion" yang menyatakan bahwa *output digital* ADC dapat diambil sebagai besaran *digital* yang mewakili *input analog* yang dikonversikan.

### 2.2.2. ADC 0808

*Analog to Digital Converter (ADC) 0808* dikemas dalam IC *monolitik*, ADC ini menggunakan metode pengubah berturut-turut (*Successive Approximation Register*) dengan 8 saluran *input analog* dan dilengkapi beberapa

kontrol logika sesuai dengan standart mikroprosesor. ADC 0808 terdiri dari tiga bagian utama, yaitu :

- *MUX Analog 8 Channel.*
- *Analog to Digital Converter (ADC) 8 bit.*
- *Three State Buffer (Penyangga Terkunci 3 Kanal).*

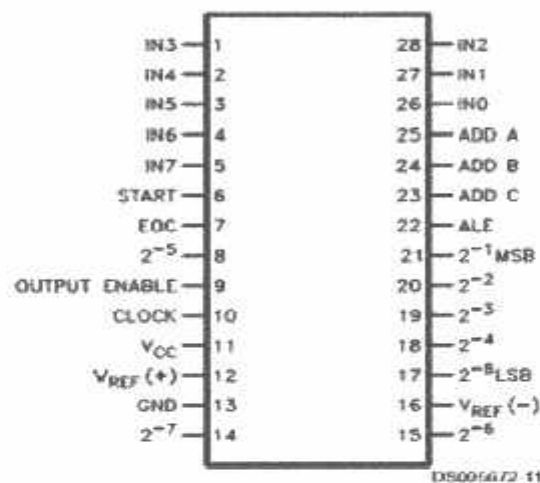
*Multiplexer analog 8 channel* berfungsi untuk mengolah *input analog* secara bergantian untuk memilih *input* mana yang dikehendaki. *Output multiplexer* disediakan 3 *bit* kontrol pemilih saluran *input* A, B, C yang dalam perencanaan nantinya semua *input* tersedia. *Analog to Digital Converter (ADC)* merupakan *successive approximation register 8 bit* yang terdiri dari komparator, SAR, dan *Clock Three-State Latch Buffer* berfungsi untuk menampung keluaran ADC 8 bit. Spesifikasi ADC 0808 adalah :

- *Supply* tegangan yang dibutuhkan sebesar 5V
- Waktu konversi adalah 100 $\mu$  detik
- Mudah di-*interface*-kan dengan mikrokontroler / berdiri sendiri
- Tegangan masuk *analog* 0-5 Volt
- Pengaturan kenaikan tegangan per-*bit* dapat diatur dengan mengukur tegangan *Vref*
- Frekwensi *clock input* 10-1280 KHz
- Konsumsi daya rendah

ADC ini merupakan level tegangan terbatas untuk sinyal *input*-nya yang tergantung pada *input referensi* diberi tegangan negatif dan *input referensi* positif diberi tegangan positif sehingga tegangan *input analog* dapat berharga diantara

kedua tegangan *referensi* tadi dan ini disebut *bipolar*, artinya kedua *input*-nya mempunyai polaritas positif dan negatif.

Jika input *referensi* positif diberi tegangan positif dan *referensi* negatif diberi tegangan 0 Volt, maka dapat disebut *unipolar* karena *analog*-nya berada diatas 0 Volt (berharga positif saja). Gambar struktur pin ADC 0808 dapat dilihat pada gambar dibawah ini.



Gambar 2-8 Struktur Pin-Pin ADC 0808

- In0 - In7 (*Input Analog*)

Adalah sejumlah 8 *bit analog* yang dapat mengatur kombinasi *bit* pada pin Add A, B, C.

- Add A, B, C (*Address*)

Untuk mengatur pemilihan *analog input* yang akan dikonversikan.

- *Clock*

Pin *clock* adalah untuk memberikan pengaturan keserempakan kerja *internal circuit* ADC. *Typical* yang disarankan 640 KHz dan maksimumnya adalah 1280 KHz.

- Vref (+) dan Vref (-)

Tegangan *referensi* untuk menentukan *bit* keluaran, misalnya diberikan tegangan *referensi* 5 volt. Kombinasi yang mungkin dari 8 *bit* keluaran adalah 256 ( $2^8$ ), maka kenaikan tegangan *per-bit* adalah

$$\frac{5}{(2^8 - 1)} = 19,6mV. \text{ Sehingga tegangan } input \text{ yang bisa terdeteksi adalah}$$

19,6 mV dengan keluaran digital  $00000001_2$  akan naik menjadi  $00000010_2$  jika tegangan naik menjadi 39,2 mV demikian seterusnya.

- $2^0$  sampai  $2^7$

Tempat keluarannya data digital hasil konversi yang dihubungkan dengan data *bus system* jika di-*interface*-kan dengan mikroprosesor.

- ALE (*Address Latch Enable*)

Untuk *Latch* / pengunci alamat yang telah ada pada Add A, B, C sesuai dengan *input* yang akan dikonversikan. Pin *input* adalah aktif *high*.

- *Start*

Untuk membantu ADC memulai konversi tegangan *input* yang telah dipilih dengan memberi kombinasi pada 3 *bit* alamat dan mengenainya dengan ALE, pin ini aktif *high*.

- EOC

Pin keluaran ini akan memberikan sinyal *low* jika ADC telah selesai mengkonversi tegangan *analog* yang diberikan. Pin ini dapat dihubungkan ke suatu port sebagai status / ke pin *interrupt* dari prosesor sebagai tanda pemberitahuan agar CPU mengambil data yang telah siap.

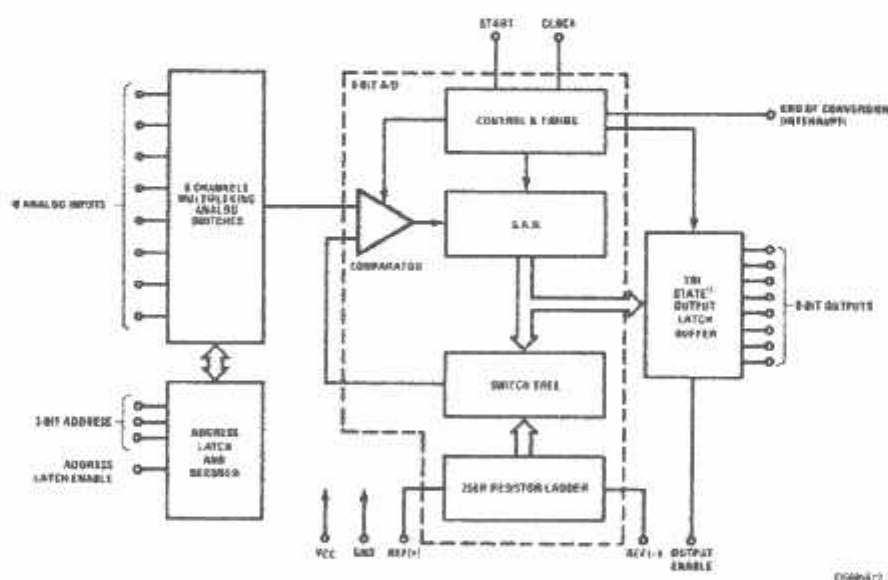
- OE (*Output Enable*)

Setelah EOC memberikan pulsa *low* sejenak, maka berarti data telah siap di *buffer internal* ADC untuk mengambil data yang masih di *buffer*, agar keluar ke data *bus* perlu diberikan pulsa *high* pada pin OE.

Dapat digambar secara garis besar, urutan proses konversi dari besaran *analog* ke digital dengan menggunakan ADC 0808 adalah :

- Memberikan *input* yang akan dikonversi dengan memberikan kombinasi *bit* pada *address*.
- Memberikan pulsa ALE untuk menguji pilihan tersebut *analog*.
- Memberikan pulsa *start* untuk memulai konversi.
- Menanti pulsa EOC dari ADC sebagai tanda selesainya konversi.
- Memberikan pulsa OE untuk aktifkan *output*.

Pada gambar dibawah ini dapat dilihat karakteristik kerja ADC.



Gambar 2-9 Karakteristik Kerja ADC

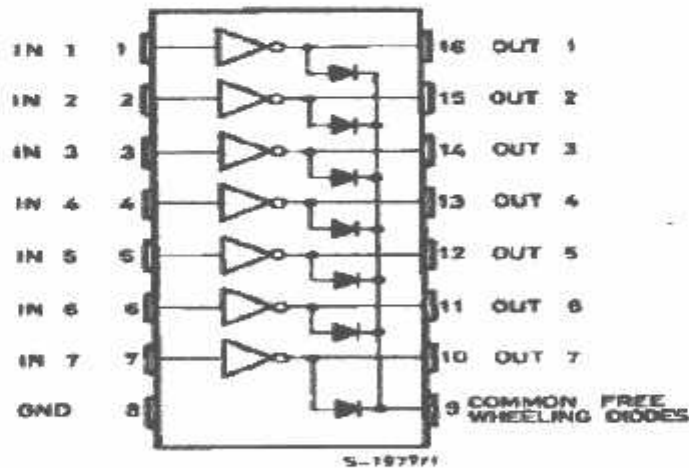


### 2.3. IC ULN2004A

Pada IC ULN2004A di dalamnya terdapat rangkaian penguat Darlington. Untuk satu IC ULN2004A terdapat 7 pasang rangkaian Darlington NPN yang tersusun dalam rangkaian *common catoda*. Rangkaian Darlington ini digunakan sebagai saklar. Pada masing-masing rangkaian Darlington arus kolektornya sebesar 500 mA. Rangkaian Darlington yang ada di dalam IC ULN2004A dapat diparalel guna untuk kebutuhan arus yang besar.

Karenanya IC ini dapat diaplikasikan untuk *driver relay*, *driver lampu*, *driver display* dan *logic huffer*. Pada skripsi ini rangkaian Darlington yang ada pada IC ULN2004A digunakan sebagai *driver relay* yang digunakan untuk mengendalikan relay.

Adapun pin-pin koneksi yang ada dalam IC ULN2004A dapat dilihat pada gambar 2.11 berikut ini:



Gambar 2.10. Pin-Pin Koneksi Dalam IC ULN2004A

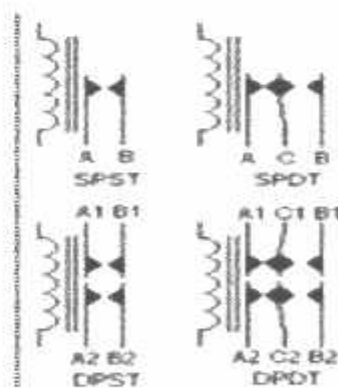
IC ULN2004A dapat terpicu dengan tegangan 5 Volt dan arus maksimum yang diperbolehkan sebesar 500 mA dengan suhu kerja dari -20°C sampai 70°C dengan data di atas maka IC ULN2004A mampu digunakan untuk menghidupkan dan mematikan relay yang memiliki tegangan maksimal sebesar 12 Volt.

#### 2.4. Relay

*Relay* adalah komponen elektronika yang terdiri dari sebuah lilitan kawat (kumparan) yang terlilit pada sebuah inti besi lunak. Jika kumparan dialiri oleh arus listrik, maka inti besi akan timbul magnet listrik yang dapat memutuskan atau menghubungkan kontak. Jenis susunan kontak sebagai berikut :

- *Normally Open* (Normal terbuka)  
Yaitu kontak akan terhubung pada rangkaian saat kumparan *relay* dialiri arus listrik.
- *Normally Close* (Normal Tertutup)  
Yaitu kontak tidak akan terhubung pada rangkaian saat kumparan *relay* dialiri arus listrik.

Sedangkan jenis *relay* berdasarkan cara dalam pengontakan dapat dilihat pada gambar berikut ini :



Gambar 2-11. Jenis Relay

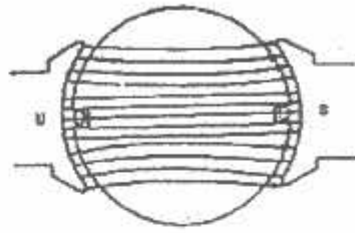
- SPST (*single pole single throw*)  
Memiliki 1 kontak kerja dan 1 kontak diam.
- SPDT (*single pole double throw*)  
Memiliki 1 kontak kerja dan 2 kontak diam.
- DPST (*double pole single throw*)  
Memiliki 2 kontak kerja dan 1 kontak diam.
- DPDT (*double pole double throw*)  
Memiliki 2 kontak kerja dan 2 kontak diam.

## 2.5. Motor DC

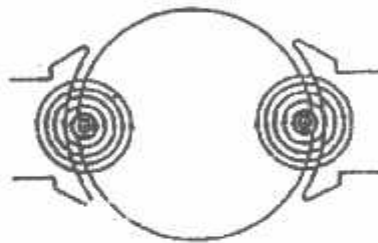
Motor Arus Searah (DC) adalah mesin yang berfungsi mengubah energi listrik menjadi energi mekanik. Prinsip kerja motor DC didasarkan pada penghantar yang membawa arus ditempatkan dalam suatu medan magnet, maka penghantar tersebut akan mengalami gaya. Gaya menimbulkan torsi yang menghasilkan rotasi mekanik, sehingga motor akan berputar. Dalam sistematika kerjanya bisa disimpulkan sebagai berikut :

- Adanya garis-garis gaya medan magnet (*Fluk*) antara kutub yang berbeda di stator.
- Penghantar dialiri arus ditempatkan pada jangkar dalam medan magnet.
- Pada penghantar timbul gaya menghasilkan torsi.

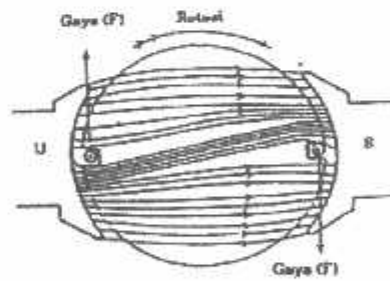
Prinsip kerja motor arus searah tersebut seperti yang terlihat pada gambar berikut :



Gambar 2-12. Medan yang dihasilkan oleh kutub



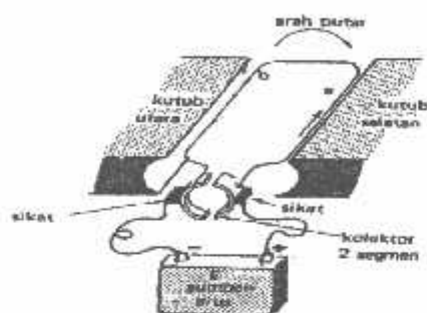
Gambar 2-13. Medan yang dihasilkan arus yang mengalir pada penghantar



Gambar 2-14. Interaksi kedua medan menghasilkan gaya

### 2.5.1. Cara Kerja Motor DC

Adapaun cara kerja motor DC dapat dilihat pada gambar di bawah ini :



Gambar 2-15. Cara Kerja Motor DC

Ada satu lilitan kawat A-B berada di dalam medan magnet. Lilitan ini dapat berputar dengan bebas, lilitan ini biasa disebut jangkar (*armour*). Pada jangkar dimasukan arus yang berasal dari sumber (Baterai) E. koneksi baterai dengan jangkar melalui sikat-sikat. Sikat-sikat ini terpasang pada sebuah cincin yang terbelah dua, yang disebut kolektir. Adapun tujuan dari kontruksi ini adalah agar lilitan kawat dapat berputar apabila ada arus listrik yang melewatinya.

Pada kawat yang berada di kanan arus mengalir dari depan ke belakang dalam kawat yang di kiri, arus mengalir dari belakang. kawat A dan B secara berganti-gantian berada di kiri dan kanan. Karena itu arah arus di A dan arah arus di B selalu membolak-balik. Pembalikan arah arus itu terjadi pada saat lilitan kawat melintasi posisi megnetik. disini kolektor berfungsi bagaikan penyearah mekanik.

*Fluk* magnet yang ditimbulkan magnet, magnetiknya disebut medan magnetnya motor. dalam gambar arah fluk magnetik adalah dari kiri ke kanan.

Adapun gaya yang bekerja pada penghantar B adalah ke atas, sementara gaya yang bekerja pada penghantar A adalah ke bawah. Gaya-gaya yang bekerja sama kuatnya, jadi ada kopel yang bekerja pada kawat sehingga lilitan pun dapat berputar. Setelah berputar  $90^{\circ}$  arah arus berbalik, pada saat itu penghantar A dan penghantar B bertukar tempat. Akibatnya arah gerak putaran tidak berubah.

## 2.6 Potensiometer

Potensiometer merupakan pembagi tegangan yang paling banyak digunakan pada pengatur volume, bass, treble, dan penguat, pengatur cerah dan kontras dalam penerima TV. Potensio yang ada di pasar ada 2 kelas yaitu kelas A dan B dalam hal ini kita menggunakan potensio kelas A, potensio berfungsi sebagai sensor gerak jari sehingga dibutuhkan potensio yang linier.

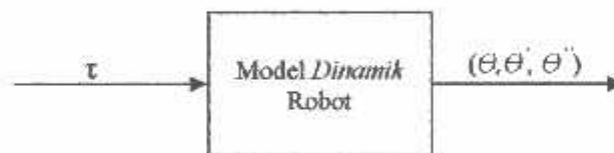


Gambar 2-16 Potensiometer

## 2.7 Aktuator (Robot)

Robot secara fisik adalah suatu benda yang memiliki struktur tertentu dengan masa tertentu yang dalam pergerakannya tunduk pada hukum-hukum alam

yang berkaitan dengan gravitasi atau masa/ kelembaman. Jika robot berada di permukaan bumi maka kedua efek, gravitasi dan masa akan mempengaruhi gerakan. Sedang bila robot berada di luar angkasa yang bebas gravitasi maka masa saja yang yang dapat menimbulkan efek *inersia*/kelembaman. Dalam konteks inilah dikatakan bahwa model dinamik dari suatu robot berhubungan dengan struktur dan masa. Setiap struktur dan masa yang berbeda akan memberikan efek *inersia* yang berbeda pula sehingga penanganan dalam pemberian torsi pada setiap sendi dengan kata lain aktuator/motor tiap sendi seharusnya berbeda pula.

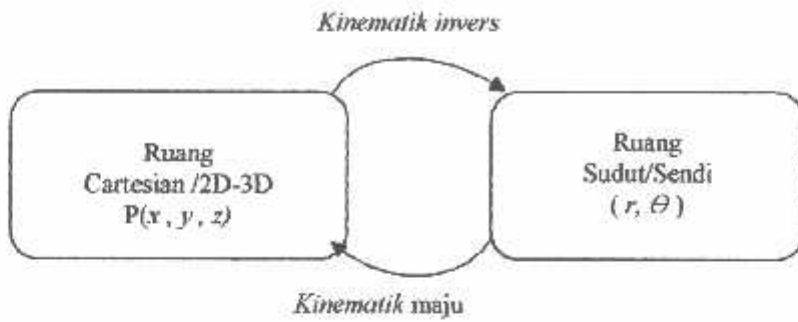


Gambar 2-17 Diagram Model Dinamik Robot

Dapat di lihat dari gambar 2-14 jika  $u$  adalah sinyal aktuator pada *aktuator* motor DC-*torsi* Maka input yang masuk pada model dinamik robot dapat dinyatakan torsi  $\tau$ ,

$$\tau = I \cdot Ktn \dots\dots\dots (2.5)$$

Seperti yang ditunjuk pada Gambar 2.14 dengan  $I$  adalah sinyal analog motor (arus motor) yang di keluarkan oleh kontroler, dan  $Ktn$  adalah konstata motor. Karena torsi pada sendi akan menghasilkan gerakan maka output (dinamik) robot dapat dinyatakan memiliki 3 komponen yang menyatu dalam fenomena gerak rotasi tiap lengan sendi, yaitu sudut yang menyatu dalam fenomena gerak rotasi tiap lengan sendi, yaitu sudut  $\theta$ , kecepatan sudut  $\theta'$ , dan percepatan sudut  $\theta''$ .



Gambaro 2-18 Transformasi Kinematik Maju dan Kinematik Invers

Jika jari-jari  $r$  dan  $\theta$  dari suatu struktur robot  $n$  DOF (*Degree of Freedom* atau derajat kebebasan) di ketahui maka posisi  $P(x, y, z)$  dapat di hitung. Jika  $\theta$  merupakan sebuah fungsi berdasar sebuah waktu,  $\theta(t)$ , maka posisi dari orientasi  $P(t)$  dapat di hitung secara pasti juga. Transformasi koordinat ini dikenal sebagai kinematik maju/langsung.

Sebaliknya, jika posisi dan orientasi  $P(t)$  di ketahui maka  $\theta(t)$  tidak dapat langsung di hitung tanpa mendefinisikan beberapa DOF setrutur robot itu. Jumlah sendi  $n$  dari  $n$ -DOF yang bisa di buat untuk melaksanakan tugas sesuai dengan posisi dan orientasi  $P(t)$  itu dapat bernilai  $n = (m, m + 1, m + 2, \dots, m + p)$  dimana  $m$  adalah jumlah sendi minimum dan  $p$  adalah jumlah sendi yang dapat ditambahkan, jadi bila  $(m + p) - n$  disebut sebagai robot redundan. Transformasi ini dikenal sebagai kinematik invers.

Kinematik dalam robotik adalah suatu bentuk pernyataan yang berisi tentang deskripsi matematik geometri dari suatu struktur robot. Dari persamaan kinematik dapat diperoleh hubungan antara konsep geometri ruang sendi pada robot dengan konsep koordinat yang biasa dipakai untuk menentukan kedudukan dari suatu obyek. Dengan model kinematik, programmer dapat menentukan



konfigurasi referensi input yang harus di umpankan ke tiap aktuator agar robot dapat melakukan gerak simultan (seluruh sendi) untuk mencapai posisi yang dikehendaki. Sebaliknya informasi kedudukan (sudut) yang dinyatakan oleh tiap sendi ketika robot sedang melakukan gerakan, dengan analisa kinematik programmer dapat menentukan dimana posisi ujung lengan atau bagian robot yang bergerak itu dalam koordinat ruang.

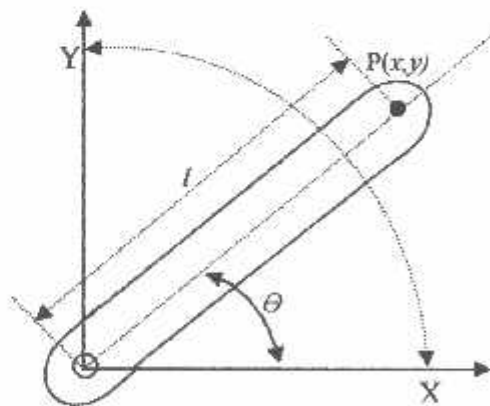
Secara umum persamaan kinematik maju untuk setiap sendi 1lengan dapat dinyatakan sebagai,

$$P(x,y) = f(r, \theta) \dots\dots\dots (2.6)$$

Dengan  $r$  adalah jari-jari lengan (link) dan  $\theta$  adalah sudut sendi. Dalam hal ini  $P$  adalah koordinat  $(x,y)$  yang relatof terhadap koordinat tetap/acuan  $(0, 0)$  pada titik sendi. Jika  $r$  adalah tetap dengan asumsi lengan bergerak secara rotasi maka  $r$  diagap konstan. Dengan demikian perubahan  $P$  hanya di pengaruhi oleh perubahan  $\theta$ . Persamaan kenematik invers-nya dapat dinyatakan sebagai,

$$(r, \theta) = f(P) \dots\dots\dots (2.7)$$

**2.7.1. Kinematik dan Dinamik Robot**



Gambar 3-19 Konfigurasi Robot Tangan Satu Sendi

Kedudukan ujung lengan  $P(x,y)$  dapat di peroleh dengan cara kinematik maju sebagai berikut

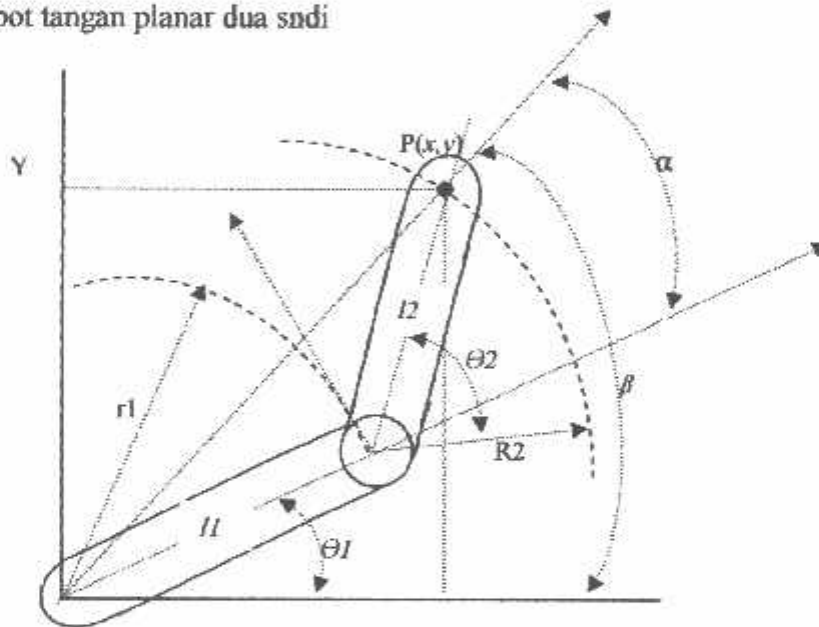
$$x = l \cdot \cos (\theta) \text{ dan } \dots\dots\dots (2.8)$$

$$y = l \cdot \sin (\theta) \dots\dots\dots (2.9)$$

Jika  $(x,y)$  diketahui maka  $\theta$  dapat dihitung dengan cara,

$$\theta = \arctan\left(\frac{y}{x}\right) \text{ atau } \theta = \tan^{-1}\left(\frac{y}{x}\right) \dots\dots\dots (2.10)$$

Kinematik robot tangan planar dua sнди



Gambar 2-20 Knfigurasi Robot Tangan Planar 2 Sendi (2DOF)

Kedudukan ujung lengan dinyatakan sebagai  $P(x,y)$

$$P(x,y) = f(\theta_1, \theta_2) \dots\dots\dots (2.11)$$

dengan  $\theta_i$  = sudut sendi robot ke -i

Jika P diasumsikan sebagai vector penjumlahan yang terdiri dari vector  $r_1$  lengan-1 dan  $r_2$  lengan-2.

$$r_1 = [l_1 \cos \theta_1, l_1 \sin \theta_1] \dots\dots\dots (2.12)$$

$$r_2 = [l_2 \cos (\theta_1 + \theta_2), l_2 \sin (\theta_1 + \theta_2)] \dots\dots\dots (2.13)$$

maka

$$x = l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2) \dots\dots\dots (2.14)$$

$$y = l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2) \dots\dots\dots (2.15)$$

Persamaan di atas adalah persamaan kinematik maju dari robot tangan 2 sendi.

Kinematik invers robot dapat dijabarkan sebagai berikut. Dengan menggunakan hukum identitas trigonometri.

$$\cos (a + b) = \cos (a) \cos (b) - \sin (a) \sin (b) \dots\dots\dots (2.16)$$

$$\sin (a + b) = \sin (a) \cos (b) + \sin (b) \cos (a) \dots\dots\dots (2.17)$$

Persamaan diatas dapat di tulis kembali,

$$x = l_1 \cos \theta_1 + l_2 \cos \theta_1 \cos \theta_2 - l_2 \sin \theta_1 \sin \theta_2 \dots\dots\dots (2.18)$$

$$y = l_1 \sin \theta_1 + l_2 \sin \theta_1 \cos \theta_2 + l_2 \cos \theta_1 \sin \theta_2 \dots\dots\dots (2.19)$$

## BAB III

### PERANCANGAN DAN PEMBUATAN ALAT

#### 3.1. Pendahuluan

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

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

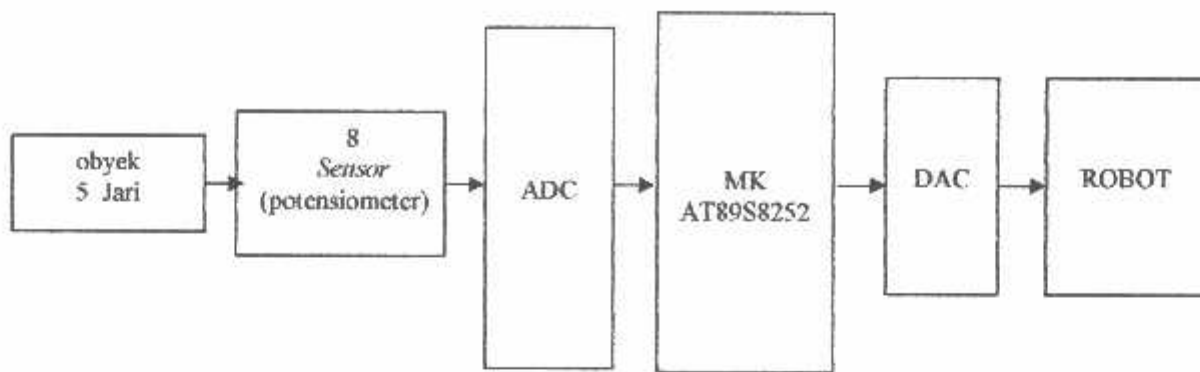
Pada perancangan perangkat keras akan meliputi seluruh *peripheral* yang digunakan pada sistem ini. Pada perancangan perangkat lunak akan meliputi diagram alir dan *software* secara umum. Akan tetapi kedua perangkat ini dalam kerjanya akan saling menunjang satu sama lain.

#### 3.2. Perancangan dan pembuatan perangkat keras

Pada perancangan dan pembuatan perangkat keras ini dapat dijelaskan melalui blok diagram dibawah ini :

##### 3.2.1. Diagram Blok

Di bawah dapat dilihat diagram blok dari alat yang akan dirancang.



Gambar 3-1. Diagram Blok Sistem Prototype robot yang menirukan gerakan jari tangan

Fungsi dari masing-masing blok rangkaian sistem adalah sebagai berikut :

- **Obyek**  
Obyeknya adalah 5 jari tangan manusia yang berfungsi untuk memberi perubahan nilai pada transduser.
  - Jari telunjuk, jari tengah, jari manis, jari yang masing-masing dibagi menjadi 2 persendian.
  - Ibu jari dan jari kelingking masing-masing 1 sendi.
- **Sensor**  
*Sensor* menggunakan 8 potensiometer jenis geser yang dilekatkan di bawah persendian jari tangan.  
Rangkaian *sensor* berfungsi sebagai penerima sinyal gerak dari Objek yang akan di kirim ke ADC.
- **Rangkaian ADC**  
Menggunakan ADC 0808  
Rangkaian ADC berfungsi mendekodekan sinyal analog ke biner dari *sensor* potensio yang akan diterima dan diproses oleh Mikrokontroler.
- **Mikrokontroler**  
Pada mikrokontroler ini port yang digunakan sebanyak 32 port yang terdiri dari P0.0 – P0.7 sebagai inputan P1.0 – P1.7, P2.0 – P2.7, P3.0 – P3.7 Digunakan untuk *Driver* motor DC.

- *ROBOT*

- a. *Driver*

- Driver* berfungsi sebagai penggerak motor DC.

- b. *Motor DC*

- Berfungsi sebagai penggerak jari-jari robot yang terkontrol dan terkendali oleh gerak jari tangan.

- c. *Mekanik*

- Bagian-bagian yang bergerak dan bentuk bangun dari robot.

### 3.2.2. Mikrokontroler Sebagai Minimum Sistem

Rangkaian minimum sistem dari mikrokontroler AT89S8252 terdiri dari rangkaian clock dan reset. Rangkaian tersebut tersusun dari komponen-komponen 3 buah kapasitor, 1 buah IC mikrokontroler, sebuah resistor dan sebuah kristal atau resonator keramik. Rangkaian kapasitor dan kristal atau resonator keramik digunakan sebagai rangkaian pembangkit internal clock generator yang terdapat pada AT89S8252. Nilai kapasitansi ditentukan sesuai dengan jenis oscilator yang digunakan, yaitu:

$C1$  dan  $C2 = 20\text{pF} - 40\text{pF}$  untuk kristal

$C1$  dan  $C2 = 30\text{pF} - 50\text{pF}$  untuk resonator keramik.

Karena dalam rancangan digunakan oscilator kristal maka harga kapasitor yang penulis gunakan adalah sebesar 30pF.

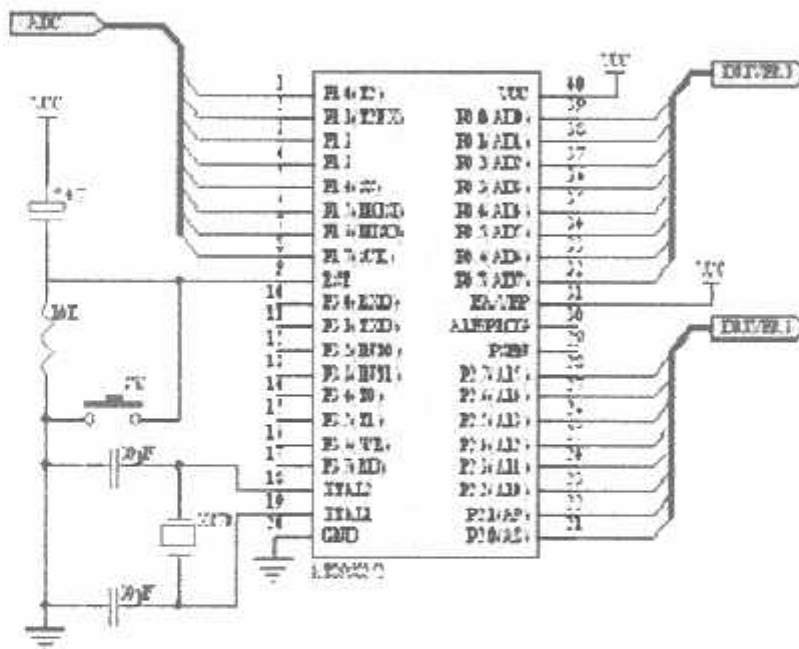
Mikrokontroler AT89S51 mempunyai frekwensi maksimal 12 MHz, dimana 1 siklus mesin = 12 clock. Dalam rangkaian digunakan kristal dengan harga 11,0592 MHz, maka program akan dijalankan pada setiap langkahnya selama 1,085  $\mu\text{s}$ . Siklus tersebut diambil berdasarkan ketentuan mikrokontroler

AT89S8252 yaitu 12 clock = 1 siklus mesin, sedangkan frekwensi yang digunakan 12 MHz, maka waktu yang dipakai dalam setiap 1 siklus mesin adalah 1  $\mu$ s. Dengan demikian perhitungannya dapat dilihat sebagai berikut:

$$f = 11,0592 \text{ MHz} \qquad T = \frac{1}{f} \qquad T = \frac{1}{11,0592 \times 10^6}$$

Karena 1 siklus mesin = 12T maka,

$$1 \text{ siklus mesin} = 12 \times \frac{1}{11.0592 \times 10^6} = 1,085 \mu\text{s}. \qquad \dots \dots \dots (3.1)$$

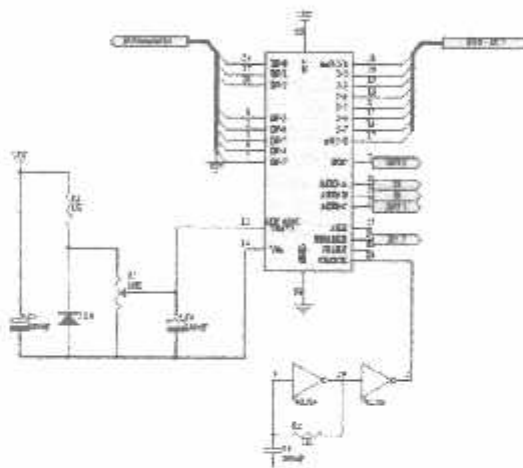


Gambar 3-2  
Rangkaian Minimum Sistem Mikrokontroler AT89S8252<sup>110]</sup>

### 3.2.3 Perencanaan ADC

Dalam perencanaan alat ini penulis menggunakan ADC 0808 yang merupakan ADC dengan sistem *Successive Aproximation*. ADC 0808 memiliki 8 buah *input*, tetapi pada alat ini hanya dipakai tiga buah *input* yaitu dari rangkaian

sensor suhu yang kemudian masuk pada pin IN-0, dari rangkaian sensor asap yang kemudian masuk pada pin IN-1, terakhir dari rangkaian rotari selektor yang masuk pada pin IN-2. ADC ini akan mengeluarkan data ke port 1 mikrokontroler. Untuk pemilihan saluran *input* (ADD-A, ADD-B, ADD-C), ADD-A dihubungkan pada port T1 mikrokontroler, ADD-B dihubungkan pada port T0 mikrokontroler, dan ADD-C dihubungkan pada port INT-0 mikrokontroler.



Gambar 3-3 Rangkaian ADC 0808

Dalam perancangan ini tegangan referensi yang diberikan pada ADC adalah 5 Volt. Dimana tingkat resolusi ADC 0808 adalah :

$$\text{Resolusi} = \frac{V_{ref}}{(2^8 - 1)} \quad \dots \quad (3.2)$$

$$\text{Resolusi} = \frac{5V}{255} = 19,6mV$$

Jadi besarnya resolusi adalah sebagai berikut : 19,6 mV.



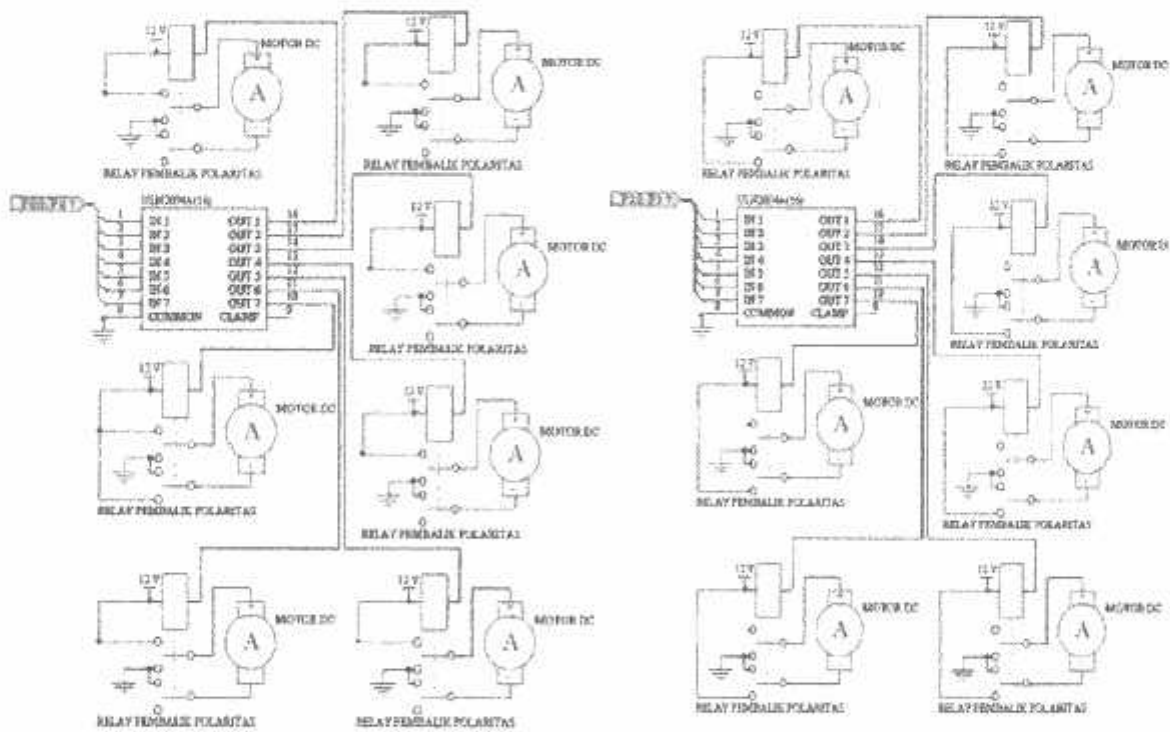
Untuk mengetahui data keluaran dari ADC, dapat diketahui dengan rumus :

$$V_o = \frac{V_{in}}{V_{ref}} \times 255 \dots\dots\dots (3.3)$$

Jika dalam pengujian ADC menggunakan LED, maka data yang telah dihitung diubah dalam bentuk *hiner* agar lebih mudah dalam pengecekan kesalahan (*error*).

### 3.2.4. Driver Motor DC

Rangkaian ini berfungsi untuk menyambung dan memutuskan rangkaian yang ada diluar mikrokontroller. yaitu motor *conveyor*, motor kotak penampung apel, pembalik polaritas kotak penampung apel dan *central lock* digambarkan sebagai berikut :



Gambar 3-3. Rangkaian Driver Relay

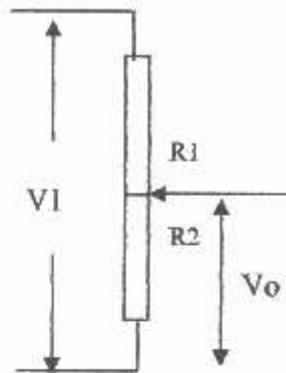
Meski IC ULN 2004A menggerakkan 7 buah relay, IC ULN 2004A tetap dapat menggerakkan relay dengan baik karena IC ULN 2004A mampu mengoperasikan sampai dengan arus 500mA.

$$\begin{aligned} R_{\text{relay}} &= 400\Omega \\ V &= 12 \text{ Volt} \\ I &= \frac{V}{R_{\text{relay}}} \\ &= \frac{12}{400} \\ &= 30 \text{ mA} \end{aligned}$$

### 3.2.5. Sensor Gerak (Potensiometer)

sensor gerak ini menggunakan potensiometer yang dipasang pada sarung tangan dengan sedemikian rupa sehingga dapat mendeteksi gerakan jari yang berada di dalam sarung tangan tersebut.

Pada rangkaian ini nilai potensio yang di gunakan sebesar 10K $\Omega$  dan dihubungkan dengan VCC 5 Volt. Sehingga perhitungannya sebagai berikut :



Gambar 3-4 Potensiometer

Untuk mencari tegangan keluaran dari potensio di atas :

Untuk mencari tegangan keluaran dari potensio di atas :

$$V_o = \frac{R_2}{R_1 + R_2} \times V_i \quad \dots\dots\dots (3.14)$$

Sebagai contoh perhitungan apabila potensio pada posisi senter atau tengah, berarti  $R_1 = 5K\Omega$  dan  $R_2 = 5K\Omega$ , maka:

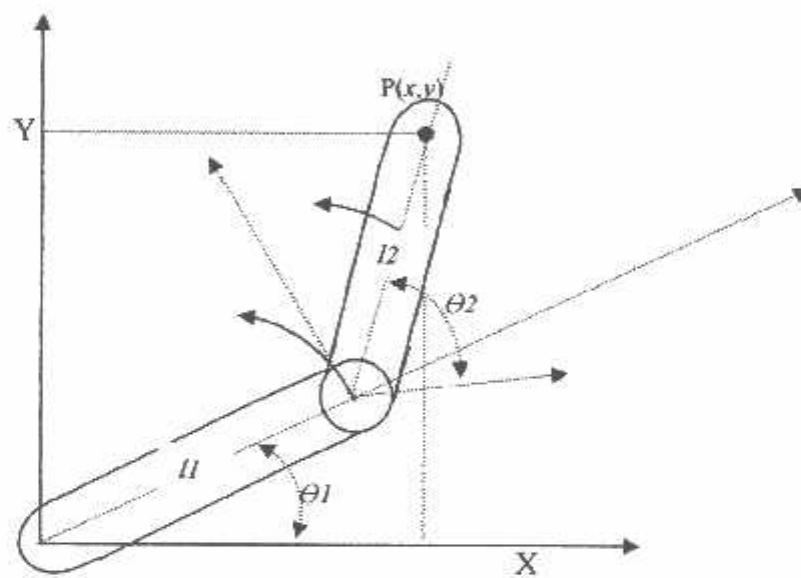
$$\begin{aligned} V_o &= \frac{5 \cdot 10^3}{5 \cdot 10^3 + 5 \cdot 10^3} \times 5 \\ &= \frac{5 \cdot 10^3}{10 \cdot 10^3} \times 5 \\ &= 2,5 \text{ volt} \end{aligned}$$

### 3.2.6. Perancangan Robot

Pada dasarnya robot jari ini adalah robot tangan dua sendi, robot ini sebagai interpretasi tangan manusia yang mempunyai derajat kebebasan yang paling sederhana adalah 2 DOF. Lebih sederhana lagi struktur 2DOF ini memiliki konfigurasi planar atau sejajar dengan permukaan Bumi sehingga kajian dinamik yang berkaitan dengan factor gravitasi dapat diabaikan. Dapat disebutkan bahwa

robot-robot seperti Bipet Robot atau Walking Robot, Mobil Manipulator, ataupun dari keluarga *Biologically Inspired Robot*, (Robot yang diinspirasi dari makhluk biologis) adalah menggunakan kajian Robot Tangan Dua Sendi (RTDS) sebagai pembangun system yang paling kecil. Pergerakan antara 2(dua) lengan dan 2(dua) sendi menjadi dasar bagi persendian robot ini.

Model matematik untuk kinematik dan dinamik robot tangan dua sendi ini mengacu pada gambar 3-5



Gambar 3-5 Foto Robot Tangan Dua Sendi

Putaran robot dan jarak tempuh sudu dapat dicari dengan perbandingan jumlah gir pada motor ( $N_1$ ) dan gir pada mekanik robot ( $N_2$ ). Dimana  $N_1$  11 gigik dan  $N_2$  121. dan mengatur kecepatan putar dengan memberikan sistem pengereman yang diatur dengan sedemikian rupa. Dan putaran output dapat ditentukan sebagai berikut.

$$\text{Putaran}_{\text{out}} = \frac{N_1}{N_2} \text{Putaran}_{\text{Motor}} \dots\dots\dots (3.15)$$

Maka dengan diketahuinya atau di tentukanya sudut perpindahan lengan maka dapat ditentukan titik (x,y) dengan menggunakan persamaan kinematik. Prsamaan kinematiknya diabil dari persamaan (2.8),(2.9),(2.14) dan (2.15) di muka, yaitu :

Kinematik 1 sendi

$$x = l \cdot \text{Cos } \theta$$

$$y = l \cdot \text{Sin } \theta$$

Kinematik 2 Sendi

$$x = l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2)$$

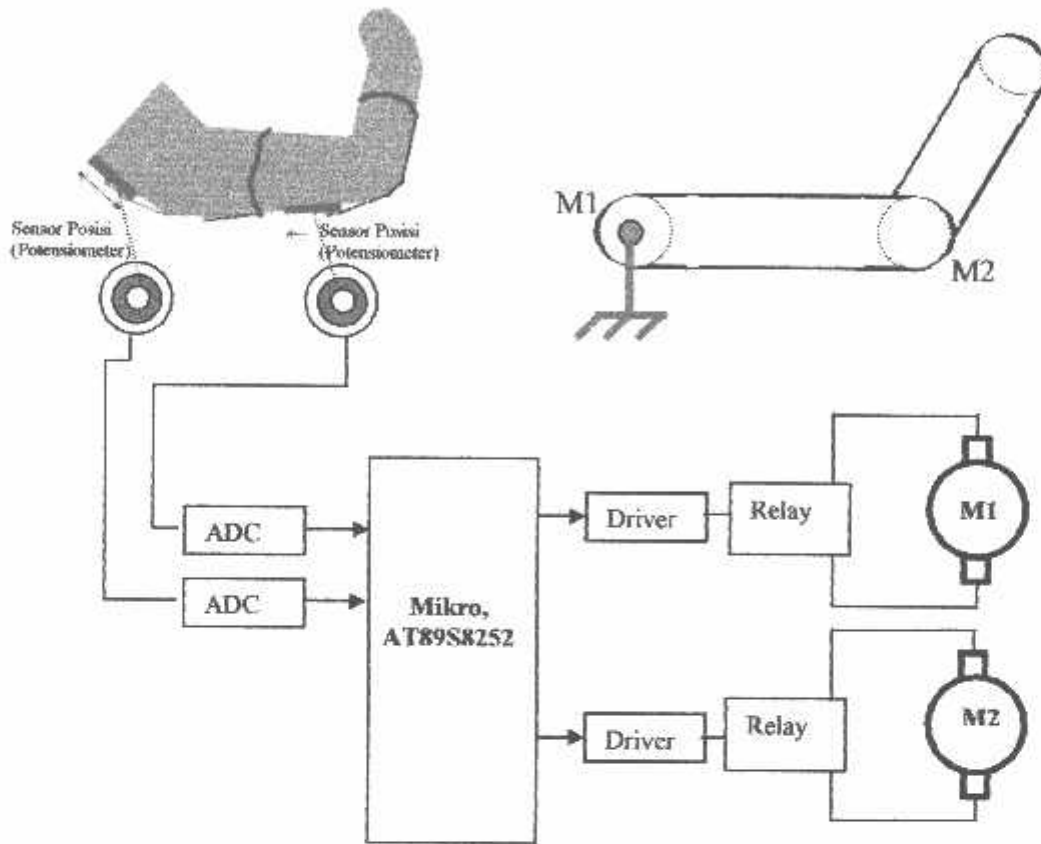
$$y = l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2)$$

Persamaan-persamaan kinematik dan dinamik di atas akan kita terapkan dalam simulasi .

### 3.2.7. Kontrol Robot

Pada dasarnya robot ini adala robot tangan 2(dua) lengan 2(dua) sendi. Perancangan RTDS (Robot Tangan Dua Sendi) ini mengunkan rangkaian control berbasis mikrokontroler AT89S8252. Mikrokontroler tersebut Harus Menyediakan seminimnya 2 kanal ADC dan 2 kanal DAC untuk robot tangan dua sendi tanpa umpan balik.Dari gerakan robot 2 inputan ADC masing-masing akan di hubungkan

ke sensor posisi (potensiometer) pada ke dua sendi jari tangan untuk mengemudikan motor di sendi-1 dan sendi-2 pada robot.



Gambar 3.6 skema Rangkaian Kontrol Berbasis Mikrokontroler AT89S8252

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

Untuk menunjang kinerja sistem keseluruhan diperlukan suatu perangkat lunak (*software*). Perangkat lunak yang digunakan AT89S8252 disini megunakan bahasa assembler, langkah langkah pembuatan program ini adalah sebagai berikut:

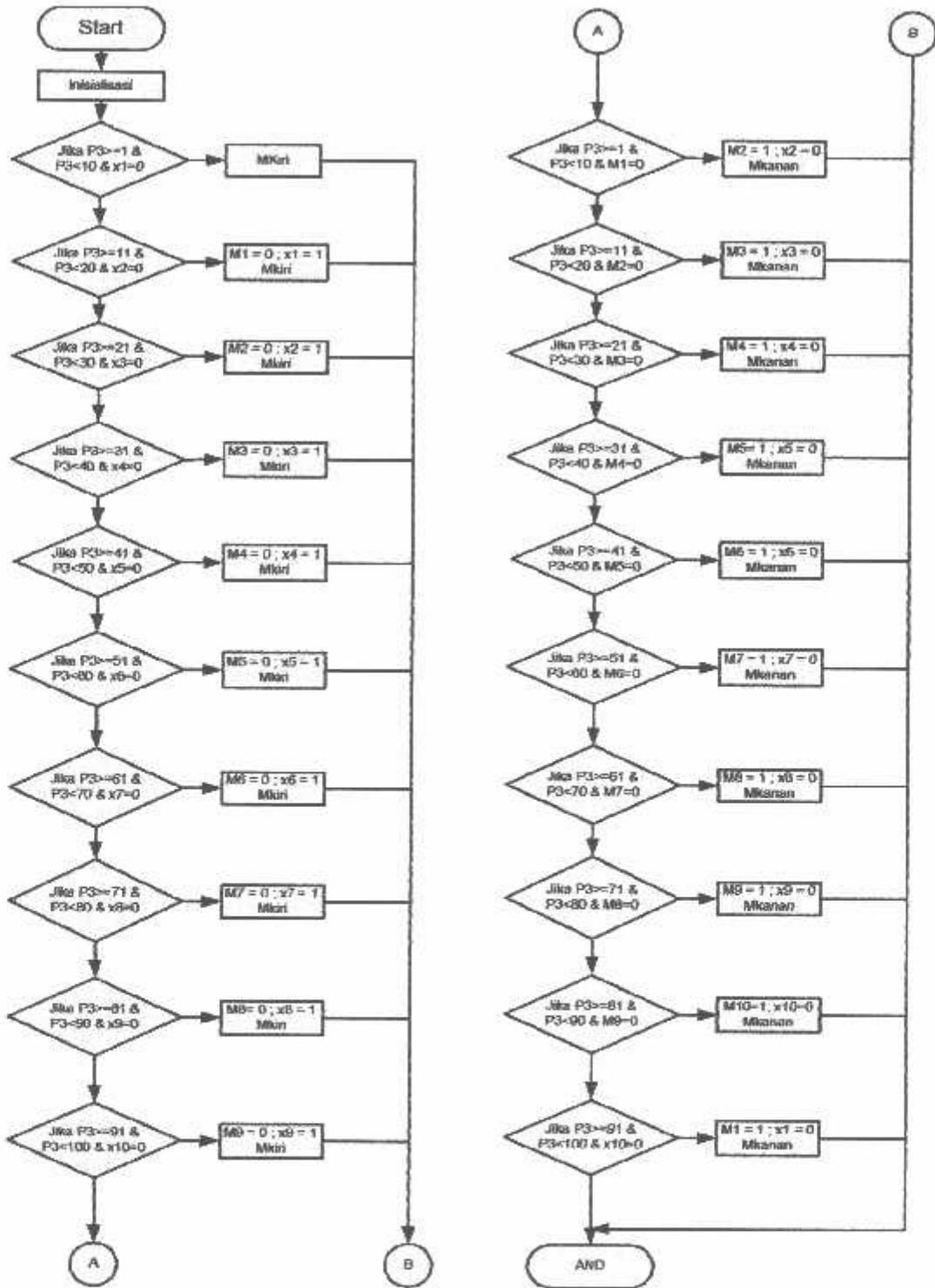
1. Membuat diagram alir (*flowchart*) program yang akan dibuat.
2. Mengubah aliran diagram tersebut ke dalam bahasa program.

3. Membuat diagram alir (*flowchart*) program yang akan dibuat.
4. Mengubah aliran diagram tersebut ke dalam bahasa program.
5. Penulisan program dengan menggunakan teks editor dan disimpan dengan ekstensi \*.H51 dan bahasa C.
6. Meng-kompilasi program dengan menggunakan *Under DOS* sehingga didapat file dengan ekstensi \*.Hex
7. Men-download file berekstensi Hex ke dalam EPROM mikrokontroler AT89S8252 menggunakan *Software Ic-Program SPI-Flash Programmer* yang dapat di download secara gratis di [WWW.IC-Program.com](http://WWW.IC-Program.com)

Selain itu juga dibutuhkan *software* untuk mengolah *database* pada PC dan di komunikasikan dengan mikrokontroler. Dalam hal ini megukan bahasa pemrograman *Mikrosop Vissual Basic 6.0*

*Software* dari alat tersebut terdapat dibagian dilampiran dan diagram alir (*flowchart*) adalah sebagai berikut,

### 3.3.1. Flowchat



Gambar 3-7 Flowchart Program



## BAB IV PENGUJIAN ALAT

### 4.1. Pendahuluan

Untuk mengetahui keberhasilan dari perencanaan dan pembuatan alat ini maka diperlukan pengujian terhadap alat tersebut. Dalam rangka pengujian diuraikan sejumlah pengukuran dan penghitungan melalui percobaan yang dilakukan untuk mengetahui sistem kerja dari alat ini secara keseluruhan.

### 4.2 Pengujian Rangkaian Sensor

#### 4.2.1 Tujuan

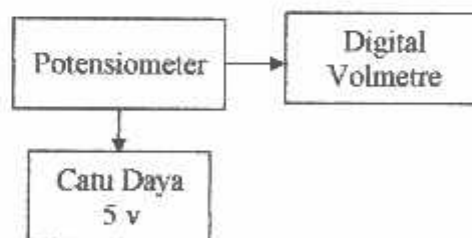
Tujuan dari pengujian rangkaian sensor adalah untuk mengetahui tingkat keberhasilan dalam perencanaan dan perancangan alat.

#### 4.2.2. Peralatan Yang Digunakan

Digital Voltmeter

Catu Daya 5 Volt (Vi)

Potensiometer Sebagai Sensor Gerak

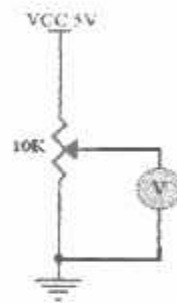


Gambar 4-1 Diagram Blok Pengukuran Sensor Gerak (Potensiometer)

### 4.2.3. Prosedur Pengujian

Prosedur pengujian rangkaian potensiometer sebagai pengukur gerak jari adalah sebagai berikut :

1. Membuat rangkaian pengujian potensiometer seperti Gambar 4.1.
2. Sumber tegangan DC diaktifkan.
3. Mencatat nilai tegangan yang tertera pada multimeter.



Gambar 4-2. Rangkaian Pengujian Potensiometer

Dengan perhitungan :

Pada saat jari bergerak , maka nilai resistansi pada potensiometer berubah, maka tegangan  $V_o$ -nya adalah seperti pada persamaan 3.14 :

$$V_o = \frac{R_2}{R_1 + R_2} \times V_i$$

Data yang kita gunakan adalah dari nilai  $R_{max}$ . ( $10K\Omega$ ) pada potensio geser kita bagi 10 bagian, maka didapat data sebagai berikut :

Tabel 4-1  
 Hasil Pengukuran Dan Perhitungan Potensiometer Sebagai Sensor Gerak

No.	Tegangan Sumber (Volt)	Pengukuran R Pada Potensiometer (K $\Omega$ )	Hasil Output Tegangan (Volt)	
			Perhitungan	Pengukuran
1		0	0	0
2		1	0.5	0.49
3		2	1	0.99
4		3	1.5	1.47
5		4	2	1.98
6	5	5	2.5	2.5
7		6	3	3.05
8		7	3.5	3.48
9		8	4	3.99
10		9	4.5	4.48
11		10	5	4.98

Dari tabel diatas, dapat dicari persentase kesalahan (*% error*) antara perhitungan dari potensiometer. Penyimpangan dari pengambilan data diperoleh dari besarnya selisih antara hasil penunjukan alat ukur yang diuji dengan hasil perhitungan. Jadi besarnya penyimpangan adalah :

$$E = F_u - F_s \dots\dots\dots (4.1)$$

Dimana :

E = Hasil Penyimpangan (*Error*)

Fu = Hasil Perhitungan

Fs = Hasil Pengukuran

Sedangkan penyimpangan dalam bentuk persen diperoleh dari perhitungan sebagai berikut :

$$\sum \%Penyimpangan = \frac{\sum Fu - \sum Fs}{\sum Fs} \cdot 100\% \dots\dots\dots (4.2)$$

Sedangkan untuk prosentase ketelitian dalam persen diperoleh perhitungan sebagai berikut :

$$\sum \%Ketelitian = 100\% - \sum \%Penyimpangan \dots\dots\dots (4.3)$$

Contoh : Diketahui hasil tegangan *output* hasil perhitungan dari potensiometer sebesar 2 V dan tegangan hasil pengukuran sebesar 1,98 V, maka presentase kesalahannya (*error*) adalah :

Penyelesaian :

$$\begin{aligned} E &= \frac{[2 - 1,98]}{1,98} \times 100\% \\ &= 1,01\% \end{aligned}$$

Dengan rumus diatas dapat diketahui presentase kesalahan (*error*) dari sensor gerak, dalam hal ini potensiometer seperti yang ditunjukkan pada tabel berikut ini :

Tabel 4.2 Presentase Kesalahan (% error) Antara Tegangan Hasil Pengukuran Dan Perhitungan pada Potensiometer

No.	Pengukuran R Pada Potensiometer (K $\Omega$ )	Hasil Output Tegangan (V)		Kesalahan / error (%)
		Perhitungan	Pengukuran	
1.	0	0	0	0
2	1	0.5	0.49	2,04
3	2	1	0.99	1,01
4	3	1,5	1.47	2,04
5	4	2	1.98	1,01
6	5	2,5	2,5	0
7	6	3	3.05	1,06
8	7	3.5	3.48	0,57
9	8	4	3,99	0,24
10	9	4,5	4.48	0,48
11	10	5	4.98	0,40

$$\text{Error rata-rata} = \frac{\text{JumlahError}}{\text{JumlahPerhitungan}} \times 100\%$$

Jika dari tabel diketahui:

$$\sum \text{Perhitungan (volt)} = 27.5$$

$$\sum \text{Error} = 6.7$$

Sehingga kesalahan rata-ratanya:

$$\begin{aligned}\text{Error rata-rata} &= \frac{\text{JumlahError}}{\text{JumlahPerhitungan}} \times 100\% \\ &= \frac{6.7}{27.5} \times 100\% \\ &= 0,246\%\end{aligned}$$

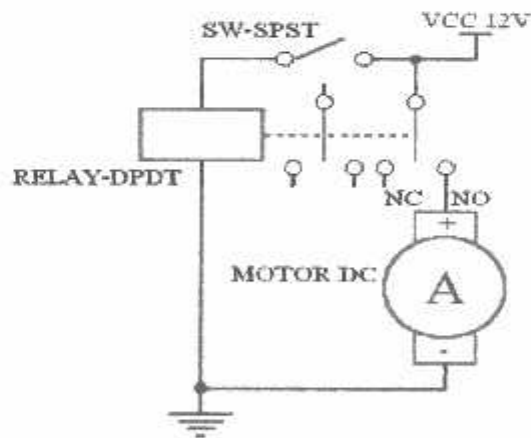
### 4.3. Pengujian Relay Penggerak Motor

#### 4.3.1. Tujuan

Tujuan dari pengujian relay penggerak motor adalah untuk mengetahui apakah relay dapat berfungsi dengan baik dan dapat digunakan pada pembuatan sistem.

#### 3.3.2. Peralatan yang Digunakan :

1. Catu daya 12 volt.
2. Relay DPDT.
3. Motor DC.
4. *Switch* SPST.



Gambar 4.3. Rangkaian Pengujian Relay

### 4.3.3. Prosedur Pengujian

Prosedur pengujian rangkaian relay sebagai penggerak motor adalah sebagai berikut :

1. Membuat rangkaian pengujian relay seperti gambar 4.3.
2. Sumber tegangan DC diaktifkan.
3. Mengamati apa yang terjadi pada motor.

- Hasil Pengujian :

Tabel 4-3.  
Hasil Pengujian Untuk Mengetahui Respon Driver Relay

Vcc (Volt)	Switch	Relay	Kondisi Motor DC
12	On	Aktif	Hidup
12	Off	Mati	Mati

## 4.4. Pengujian Rangkaian ADC 0808

### 4.4.1. Tujuan

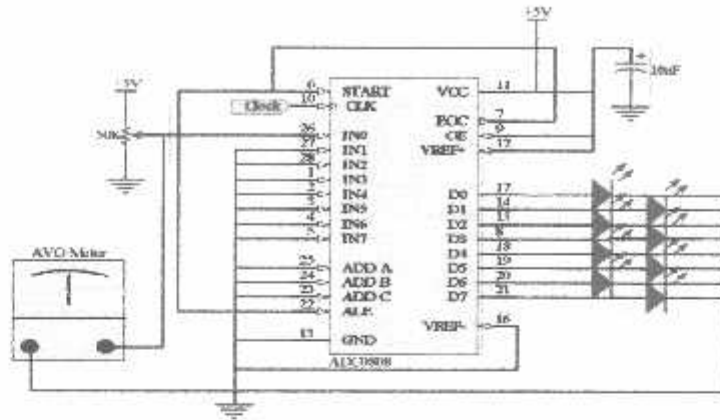
Untuk mengetahui hasil konversi tegangan *analog* menjadi bentuk tegangan *digital* ( high '1' dan low '0' ) dari tiap volt *input* tegangan *analog*.

### 4.4.2. Peralatan yang digunakan

- Rangkaian pengujian ADC 0808
- Voltmeter digital ( DT9205B )
- Sumber tegangan +5 Volt

### 4.4.3. Langkah Pengujian

1. Merangkai rangkaian seperti pada gambar dibawah ini:



Gambar 4-6 Rangkaian Pengujian ADC 0808

2. Menghubungkan pin – pin keluaran dari ADC 0808 dengan rangkaian LED.
3. Menghubungkan  $V_R = 50K$  ke pin 26 pada ADC 0808 (  $V_{in-0}$  ).
4. Menghubungkan rangkaian pengujian ADC 0808 dengan sumber tegangan 5 Volt.
5. Menghubungkan kutub positif dan negatif multimeter digital (multimeter diset pada skala 20 Volt DC) pada  $V_R$  sebagai  $V_{in-0}$ , kemudian memutar  $V_R$  (tentukan nilai  $V_{in-0}$  yang diinginkan) dan mencatatnya pada tabel 4-6.
6. Mengamati LED yang menyala dan mati sebagai indikator keluaran 8 bit dari ADC 0808. Mencatat hasilnya pada tabel 4-6.

#### b. Analisa

Untuk perhitungan Beda tegangan/bit dari output ADC 0808 dapat dicari dengan menggunakan rumus berikut:



$$\text{Beda tegangan / bit} = \frac{V_{ref}^{(+)} - V_{ref}^{(-)}}{2^8 - 1} \dots\dots\dots (4.4)$$

Dimana:  $V_{ref}^{(+)} = 5 \text{ Volt}$

$V_{ref}^{(-)} = 0 \text{ Volt}$

Sehingga:

$$\begin{aligned} \text{Beda tegangan / bit} &= \frac{V_{ref}^{(+)} - V_{ref}^{(-)}}{2^8 - 1} \\ &= \frac{5 - 0}{256 - 1} \\ &= \frac{5}{255} \\ &= 0,0196 \text{ Volt} \end{aligned}$$

Dan untuk keluaran ADC 0808 dapat dicari dengan menggunakan rumus berikut:

$$Out_{ADC(\text{desimal})} = \frac{V_{in}}{\text{BedaTegangan / Bit}} \dots\dots\dots (4.5)$$

Dimana:

$V_{in}$  = tegangan input ADC 0808 ( $V_R = 50 \text{ K}$  ke pin 26)

Beda Tagangan/Bit = 0,0196 Volt

Contoh: Jika diketahui  $V_{in}$  0,25 Volt. Maka berapakah keluaran ADC 0808 ?

Penyelesaian:

$$\begin{aligned} Out_{ADC(\text{desimal})} &= \frac{V_{in}}{\text{BedaTegangan / Bit}} \\ Out_{ADC(\text{desimal})} &= \frac{0,25}{0,0196} \\ &= 12,75 \approx 13_{(10)} \end{aligned}$$

$$Out_{ADC(\text{biner})} = 00001101_{(2)}$$

$$Out_{ADC(\text{hexa})} = 0D_{(16)}$$

Berikut merupakan tabel perbandingan tegangan output antara hasil pengukuran dan hasil perhitungan dari keluaran ADC 0808:

Tabel 4-4 Data Hasil Pengukuran Rangkaian Konversi ADC 0808

Vin (Volt)	Keluaran ADC 0808					
	Perhitungan			Pengukuran		
	D7-D0	Desimal	Hexadesimal	D7-D0	Desimal	Hexadesimal
0	00000000	0	00h	00000000	0	00h
0,25	00001101	13	0Dh	00001100	12	0Ch
0,5	00011010	26	1Ah	00011010	26	1Ah
0,75	00100110	38	26h	00100111	39	27h
1,01	00110100	52	34h	00110100	52	34h
1,26	01000000	64	40h	01000000	64	40h
1,5	01001101	77	4Dh	01001100	76	4Ch
1,75	01011001	89	59h	01011000	88	58h
2,02	01100111	103	67h	01101000	104	68h
2,25	01110011	115	73h	01110100	116	74h
2,5	10000000	128	80h	10000000	128	80h
2,75	10001100	140	8Ch	10001110	142	8Eh
3	10011001	153	99h	10011000	152	98h
3,25	10100110	166	A6h	10101000	168	A8h
3,5	10110011	179	B3h	10110100	180	B4h
3,75	10111111	191	BFh	11000010	194	C2h
4	11001100	204	CCh	11001110	206	Ceh
4,25	11011001	217	D9h	11011000	216	D8h
4,5	11100110	230	E6h	11101000	232	E8h
4,75	11110010	242	F2h	11110101	245	F5h
4,99	11111111	255	FFh	11111111	255	FFh

Dari tabel tersebut dapat dicari selisih antara hasil perhitungan dengan hasil pengukuran keluaran ADC 0808. Selisih dapat dicari dengan rumus sebagai berikut:

$$\text{Selisih} = |\text{Pengukuran} - \text{Perhitungan}|$$

Contoh: Jika diketahui hasil keluaran ADC 0808 perhitungan sebesar  $13_{(10)}$  dan pengukuran sebesar  $12_{(10)}$ , maka selisihnya adalah:

Penyelesaian:

$$\begin{aligned}\text{Selisih} &= |\text{Pengukuran} - \text{Perhitungan}| \\ &= |12 - 13| \\ &= 1_{(10)} = 01_{(16)}\end{aligned}$$

Dengan rumus diatas dapat diketahui selisih antara hasil perhitungan dengan hasil pengukuran keluaran ADC 0808 ditunjukkan pada tabel berikut:

Tabel 4-5  
Data Hasil Pengukuran Rangkaian Konversi ADC 0808

Keluaran ADC 0808						Selisih (Hexa)
Perhitungan			Pengukuran			
D7-D0	Desimal	Hexadesimal	D7-D0	Desimal	Hexadesimal	
00000000	0	00h	00000000	0	00h	00h
00001101	13	0Dh	00001100	12	0Ch	01h
00011010	26	1Ah	00011010	26	1Ah	00h
00100110	38	26h	00100111	39	27h	01h
00110100	52	34h	00110100	52	34h	00h
01000000	64	40h	01000000	64	40h	00h
01001101	77	4Dh	01001100	76	4Ch	01h
01011001	89	59h	01011000	88	58h	01h
01100111	103	67h	01101000	104	68h	01h
01110011	115	73h	01110100	116	74h	01h
10000000	128	80h	10000000	128	80h	00h
10001100	140	8Ch	10001110	142	8Eh	02h
10011001	153	99h	10011000	152	98h	01h
10100110	166	A6h	10101000	168	A8h	02h
10110011	179	B3h	10110100	180	B4h	01h
10111111	191	BFh	11000010	194	C2h	03h
11001100	204	CCh	11001110	206	Ceh	02h
11011001	217	D9h	11011000	216	D8h	01h
11100110	230	E6h	11101000	232	E8h	02h
11110010	242	F2h	11110101	245	F5h	03h
11111111	255	FFh	11111111	255	FFh	00h

Dari tabel diatas dapat dicari persentase kesalahan (%error) dari rangkaian pengkondisi sinyal. Untuk persentase kesalahan dapat dicari dengan menggunakan persamaan sebagai berikut:

$$\text{Error rata-rata} = \frac{\text{JumlahError}}{\text{JumlahPerhitungan}} \times 100\%$$

Jika dari tabel diketahui:

$$\sum \text{Perhitungan (desimal)} = 2682$$

$$\sum \text{Error} = 17_{(16)} = 23_{(10)}$$

Sehingga kesalahan rata-ratanya:

$$\begin{aligned} \text{Error rata-rata} &= \frac{\text{JumlahError}}{\text{JumlahPerhitungan}} \times 100\% \\ &= \frac{23}{2682} \times 100\% \\ &= 0,856 \% \end{aligned}$$

#### 4.5. Pengujian Mekanik

Berdasarkan bentuk, panjang berat robot jari ini memiliki ukuran mekanik yang berbeda, dan terpasang pada tempat yang berbeda.

Perbedaan tersebut sangat mempengaruhi kerja dan kemampuan gerak robot. Dan perbedaan tersebut dapat dihitung sebagai berikut.

Jumlah gigi pada poros motor  $N_1$  121 Jumlah gigi pada poros lengan  $N_2$  11 Dengan menggunakan persamaan 3.15 maka dapat diketahui perbandingan putaran motor dan lengan.

$$\text{Putaran}_{\text{out}} = \frac{N_1}{N_2} \text{Putaran}_{\text{motor}}$$

$$\text{Putaran}_{\text{out}} = \frac{11}{121} 5,5$$

$$= 0,5 \text{ putaran}$$

Kedudukan ujung lengan  $P(x,y)$  dapat di peroleh menggunakan persamaan 2,8 2,9 2,14 dan 2,15 maka didapat :

Tabel 4-6  
 Hasil Pengukuran Jarak Tempuh Lengan

Jenis lengan	Sudut		Panjang Lengan (cm)		Kinematik (x,y)	
	$\theta_1$	$\theta_2$	$L_1$	$L_2$	1 sendi	2 sendi
Ibu jari	$90^0$	-	16	-	(0,16)	-
	0	-	16	-	(16,0)	-
Jari telunjuk	$90^0$	$90^0$	7	10	-	(-10,7)
	$90^0$	0	7	10	-	(7,17)
	0	$90^0$	7	10	-	(7,10)
	0	0	7	10	-	(17,0)
Jari tengah	$90^0$	$90^0$	10	15	-	(-15,10)
	$90^0$	0	10	15	-	(0,25)
	0	$90^0$	10	15	-	(10,15)
	0	0	10	15	-	(25,0)
Jari manis	$90^0$	$90^0$	7	10	-	(-10,7)
	$90^0$	0	7	10	-	(7,17)
	0	$90^0$	7	10	-	(7,10)
	0	0	7	10	-	(17,0)
Jari kelingking	$90^0$	-	16	-	(0,16)	-
	0	-	16	-	(16,0)	-

Tabel di atas adalah batas kondisi lengan saat mencengkram dan kondisi lengan jari lurus.

Untuk mengetahui konversi tegangan output potensiometer ke besaran sudut.

Sebagai misal,  $V_{out}$  mempunyai jangkauan (0-5 )V sedang sudut yang di ukur adalah  $(0-180)^{\circ}$ , maka perputaran  $\theta^{\circ}$  adalah setara dengan perubahan tegangan output sebesar,

$$V_{out} = (\theta / 180) \cdot V_{in} \text{ dan}$$

$$\theta = (V_{out} / V_{in}) \cdot 180^{\circ}$$

Tabel 4-7  
Presentase Kesalahan (% *error*) Konversi Tegangan Output Sensor Gerak  
Rerhadap Perubahan Sudut Pada Robot

No	Vin (Volt)	Vout (Volt)	Sudut $\theta$		Error (%)
			Perhitungan	Pengukuran	
1	5	0	0	0	0
2	5	0.27	9,72 <sup>0</sup>	12 <sup>0</sup>	28
3	5	0.55	19,8 <sup>0</sup>	20 <sup>0</sup>	1
4	5	0,8	28,8 <sup>0</sup>	30 <sup>0</sup>	4
5	5	1,1	39,6 <sup>0</sup>	40 <sup>0</sup>	1
6	5	1,4	50,4 <sup>0</sup>	50 <sup>0</sup>	0,8
7	5	1,65	59,4 <sup>0</sup>	60 <sup>0</sup>	1
8	5	1,9	68,4 <sup>0</sup>	70 <sup>0</sup>	2,28
9	5	2,20	79,2 <sup>0</sup>	80 <sup>0</sup>	1
10	5	2,5	90 <sup>0</sup>	90 <sup>0</sup>	0

Dari tabel diatas, didapat persentase kesalahan (% *error*) antara perhitungan dari konversi tegangan ke sudut. Penyimpangan dari pengambilan data diperoleh dari besarnya selisih antara hasil penunjukan alat ukur yang diuji dengan hasil perhitungan. Jadi besarnya *error* rata-rata adalah :

$$\text{Error rata-rata} = \frac{\text{JumlahError}}{\text{JumlahPerhitungan}} \times 100\%$$

Jika dari tabel diketahui:

$$\sum \text{Perhitungan } (\theta^b) = 450$$

$$\sum \text{Error} = 49_{(10)}$$

Sehingga kesalahan rata-ratanya:

$$\begin{aligned} \text{Error rata-rata} &= \frac{\text{JumlahError}}{\text{JumlahPerhitungan}} \times 100\% \\ &= \frac{49}{450} \times 100\% \\ &= 9,856 \% \end{aligned}$$

#### **4.5. Pengujian Keseluruhan Alat**

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

Pengujian ini dilakukan untuk mengetahui apakah alat sudah bekerja sesuai dengan yang telah direncanakan.

##### **4.5.1. Peralatan yang Digunakan**

- Tangan Munusia.
- Sarungtangan.

##### **4.5.3. Langkah pengujian**

- Merangkai alat sesuai dengan perancangan.
- Memasukkan tangan pada sarungtangan.



- Menghubungkan catu daya ke rangkaian keseluruhan.
- Analisis pengujian :
  1. Pada saat sensor posisi mengalami perubahan maka motor akan bergerak.
  2. Robot akan bergerak mengikuti arah gerak jari tangan.

## BAB V

### PENUTUP

#### 5.1. Kesimpulan

Berdasar perencanaan, pembuatan dan pengujian robot jari-jari yang bergerak seperti halnya manusia yang menggunakan mikrokontroler AT89S8252 adalah sebagai berikut :

1. Arah gerak robot sama dengan arah gerak jari dan *Error* rata-rata sudut gerak antara jari manusia dengan robot 3,11%
2. *Error* rata-rata tegangan output dari potnsiometer sebagai sensor gerak 0,246%
3. Jumlah perbandingan gigi pada poros motor dan pada poros mekanik berpengaruh pada gerak robot.

#### 5.2. Saran-saran

Tujuan utama dari penulisan adalah bagaimana membuat suatu robot yang mampu bergerak seperti halnya manusia. Saran-saran yang bisa digunakan untuk pengembangan alat ini secara lanjut antara lain:

1. Karena sistem ini masih menggunakan Mikrokontroler AT89S8252, untuk perkembangan berikutnya diharapkan dapat digunakan mikrokontroler versi lain .
2. Diharapkan menggunakan motor *stepper* yang memiliki torsi yang kuat.

3. Untuk selanjutnya diharapkan pengontrolan gerak robot dapat dilakukan dengan menggunakan sistem jaringan atau memanfaatkan frekuensi sehingga robot dapat di Kontrol dengan jarak yang lebih jauh, atau beda ruangan dan beda tempat.

## DAFTAR PUSTAKA

- [1]. Endra Pitawarno, 2006. "*Robotika Desain, Kontrol, dan Kecerdasan Buatan*". ANDI Yogyakarta.
- [2]. Kristanto, Andi, "*Teknik Antar Muka Dan pemrograman Mikrokontroler AT89S8252*".
- [3]. S.Pd. Agus Sugiharto. 2002. "*Penerapan Dasar Transduser Dan Sensor*". Kanisius.
- [4]. Robert F. Caughlin and Frederik F. Driskal. 1997. "*Pengant Operasional Dan Rangkain Terpadu Lintier*". Erlangga.
- [5]. Data Sheet ADC 0808
- [6]. Data Sheet ULN 2004A
- [7]. <http://www.ASIMO.com>
- [8]. <http://www.AKIBO.com>
- [9]. <http://www.TECEL.com>
- [10]. <http://www.atmel.com>

---



# LAMPIRAN

---

---

---

## **DAFTAR LAMPIRAN**

**1. LEMBAR PERSETUJUAN PERBAIKAN**

**SKRIPSI**

**2. SKEMATIK RANGKAIAN**

**3. PROGRAM LISTINGT**

**4. DATA SHEET**

---

---



```

#include <stdio.h>
#include <reg51.h>
#include <math.h>

sbit OUT7 = P2 ^ 7;
sbit OUT6 = P2 ^ 6;
sbit OUT5 = P2 ^ 5;
sbit OUT4 = P2 ^ 4;
sbit OUT3 = P2 ^ 3;
sbit OUT2 = P2 ^ 2;
sbit OUT1 = P2 ^ 1;
sbit OUT0 = P2 ^ 0;

sbit OUTX7 = P0 ^ 7;
sbit OUTX6 = P0 ^ 6;
sbit OUTX5 = P0 ^ 5;
sbit OUTX4 = P0 ^ 4;
sbit OUTX3 = P0 ^ 3;
sbit OUTX2 = P0 ^ 2;
sbit OUTX1 = P0 ^ 1;
sbit OUTX0 = P0 ^ 0;

sbit AX = P0 ^ 0;
sbit BX = P0 ^ 1;
sbit CX = P0 ^ 2;

void delay(unsigned int n)
{
    unsigned int i,j;
    for (i=0;i<n;i++) for (j=0;j<1000;j++);
}

void main()

{

int    M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,
        M11,M12,M13,M14,M15,M16,M17,M18,M19,M20,
        M21,M22,M23,M24,M25,
        X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,
        X11,X12,X13,X14,X15,X16,X17,X18,X19,X20,
        X21,X22,X23,X24,X25;

X1 = 0; X10 = 0; X19 = 0;
X2 = 0; X11 = 0; X20 = 0;
X3 = 0; X12 = 0; X21 = 0;
X4 = 0; X13 = 0; X22 = 0;
X5 = 0; X14 = 0; X23 = 0;
X6 = 0; X15 = 0; X24 = 0;
X7 = 0; X16 = 0; X25 = 0;
X8 = 0; X17 = 0;
X9 = 0; X18 = 0;
Mulai:

```



```
if (P3 >= 1 && P3 <= 10 && X1 == 0)
{
    goto MKiri;
}
else
if (P3 >= 11 && P3 <= 20 && X2 == 0)
{
    M1 = 0;
    X1 = 1;
    goto MKiri;
}
else
if (P3 >= 21 && P3 <= 30 && X3 == 0)
{
    M2 = 0;
    X2 = 1;
    goto MKiri;
}
else
if (P3 >= 31 && P3 <= 40 && X4 == 0)
{
    M3 = 0;
    X3 = 1;
    goto MKiri;
}
else
if (P3 >= 41 && P3 <= 50 && X5 == 0)
{
    M4 = 0;
    X4 = 1;
    goto MKiri;
}
else
if (P3 >= 51 && P3 <= 60 && X6 == 0)
{
    M5 = 0;
    X5 = 1;
    goto MKiri;
}
else
if (P3 >= 61 && P3 <= 70 && X7 == 0)
{
    M6 = 0;
    X6 = 1;
    goto MKiri;
}
else
if (P3 >= 71 && P3 <= 80 && X8 == 0)
{
    M7 = 0;
    X7 = 1;
    goto MKiri;
}
else
if (P3 >= 91 && P3 <= 100 && X9 == 0)
{
    M8 = 0;
    X8 = 1;
```

---

```
    goto MKiri;
}
else
if (P3 >= 101 && P3 <= 110 && X10 == 0)
{
    M9 = 0;
    X9 = 1;
    goto MKiri;
}
else
if (P3 >= 111 && P3 <= 120 && X11 == 0)
{
    M10 = 0;
    X10 = 1;
    goto MKiri;
}
else
if (P3 >= 121 && P3 <= 130 && X12 == 0)
{
    M11 = 0;
    X11 = 1;
    goto MKiri;
}
else
if (P3 >= 131 && P3 <= 140 && X13 == 0)
{
    M12 = 0;
    X12 = 1;
    goto MKiri;
}
else
if (P3 >= 141 && P3 <= 150 && X14 == 0)
{
    M13 = 0;
    X13 = 1;
    goto MKiri;
}
else
if (P3 >= 151 && P3 <= 160 && X15 == 0)
{
    M14 = 0;
    X14 = 1;
    goto MKiri;
}
if (P3 >= 161 && P3 <= 170 && X16 == 0)
{
    M15 = 0;
    X15 = 1;
    goto MKiri;
}
else
if (P3 >= 171 && P3 <= 180 && X17 == 0)
{
    M16 = 0;
    X16 = 1;
    goto MKiri;
}
else
```

```
if (P3 >= 181 && P3 <= 190 && X18 == 0)
{
    M17 = 0;
    X17 = 1;
    goto MKiri;
}
else
if (P3 >= 191 && P3 <= 200 && X19 == 0)
{
    M18 = 0;
    X18 = 1;
    goto MKiri;
}
else
if (P3 >= 201 && P3 <= 210 && X20 == 0)
{
    M19 = 0;
    X19 = 1;
    goto MKiri;
}
if (P3 >= 211 && P3 <= 220 && X21 == 0)
{
    M20 = 0;
    X20 = 1;
    goto MKiri;
}
else
if (P3 >= 221 && P3 <= 230 && X22 == 0)
{
    M21 = 0;
    X21 = 1;
    goto MKiri;
}
else
if (P3 >= 231 && P3 <= 240 && X23 == 0)
{
    M22 = 0;
    X22 = 1;
    goto MKiri;
}
else
if (P3 >= 241 && P3 <= 250 && X24 == 0)
{
    M23 = 0;
    X23 = 1;
    goto MKiri;
}
else
if (P3 >= 251 && P3 <= 254 && X25 == 0)
{
    M24 = 0;
    X24 = 1;
    goto MKiri;
}
}
```

```
if (P3 >= 1 && P3 <= 10 && M1 == 0)
{
    M2 = 1;
    X2 = 0;
    goto MKanan;
}
else
if (P3 >= 11 && P3 <= 20 && M2 == 0)
{
    M3 = 1;
    X3 = 0;
    goto MKanan;
}
else
if (P3 >= 21 && P3 <= 30 && M3 == 0)
{
    M4 = 1;
    X4 = 0;
    goto MKanan;
}
else
if (P3 >= 31 && P3 <= 40 && M4 == 0)
{
    M5 = 1;
    X5 = 0;
    goto MKanan;
}
else
if (P3 >= 41 && P3 <= 50 && M5 == 0)
{
    M6 = 1;
    X6 = 0;
    goto MKanan;
}
else
if (P3 >= 51 && P3 <= 60 && M6 == 0)
{
    M7 = 1;
    X7 = 0;
    goto MKanan;
}
else
if (P3 >= 61 && P3 <= 70 && M7 == 0)
{
    M8 = 1;
    X8 = 0;
    goto MKanan;
}
else
if (P3 >= 71 && P3 <= 80 && M8 == 0)
```

```
{
  M9 = 1;
  X9 = 0;
  goto MKanan;
}
else
if (P3 >= 91 && P3 <= 100 && M9 == 0)
{
  M10 = 1;
  X10 = 0;
  goto MKanan;
}
else
if (P3 >= 101 && P3 <= 110 && M10 == 0)
{
  M11 = 1;
  X11 = 0;
  goto MKanan;
}
else
if (P3 >= 111 && P3 <= 120 && M11 == 0)
{
  M12 = 1;
  X12 = 0;
  goto MKanan;
}
else
if (P3 >= 121 && P3 <= 130 && M12 == 0)
{
  M13 = 1;
  X13 = 0;
  goto MKanan;
}
else
if (P3 >= 131 && P3 <= 140 && M13 == 0)
{
  M14 = 1;
  X14 = 0;
  goto MKanan;
}
else
if (P3 >= 141 && P3 <= 150 && M14 == 0)
{
  M15 = 1;
  X15 = 0;
  goto MKanan;
}
else
if (P3 >= 151 && P3 <= 160 && M15 == 0)
{
  M16 = 1;
  X16 = 0;
  goto MKanan;
}
if (P3 >= 161 && P3 <= 170 && M16 == 0)
{
  M17 = 1;
  X17 = 0;
```

```
    goto MKanan;
}
else
if (P3 >= 171 && P3 <= 180 && M17 == 0)
{
    M18 = 1;
    X18 = 0;
    goto MKanan;
}
else
if (P3 >= 181 && P3 <= 190 && M18 == 0)
{
    M19 = 1;
    X19 = 0;
    goto MKanan;
}
else
if (P3 >= 191 && P3 <= 200 && M19 == 0)
{
    M20 = 1;
    X20 = 0;
    goto MKanan;
}
else
if (P3 >= 201 && P3 <= 210 && M20 == 0)
{
    M21 = 1;
    X21 = 0;
    goto MKanan;
}
if (P3 >= 211 && P3 <= 220 && M21 == 0)
{
    M22 = 1;
    X22 = 0;
    goto MKanan;
}
else
if (P3 >= 221 && P3 <= 230 && M22 == 0)
{
    M23 = 1;
    X23 = 0;
    goto MKanan;
}
else
if (P3 >= 231 && P3 <= 240 && M23 == 0)
{
    M24 = 1;
    X24 = 0;
    goto MKanan;
}
else
if (P3 >= 241 && P3 <= 250 && M24 == 0)
{
    M25 = 1;
    X25 = 0;
    goto MKanan;
}
else
```

```
if (P3 >= 251 && P3 <= 254 && M25 == 0)
{
    M1 = 1;
    X1 = 0;
    goto MKanan;
}

else
{
    goto Mulai;
}
```

```
MKiri:
{
    delay(6);
    OUT7 = 0;
    OUT6 = 1;
    OUT5 = 0;
    OUT4 = 1;
    OUT3 = 0;
    OUT2 = 1;
    OUT1 = 0;
    OUT0 = 1;
    OUTX7 = 0;
    OUTX6 = 1;
    OUTX5 = 0;
    OUTX4 = 1;
    OUTX3 = 0;
    OUTX2 = 1;
    OUTX1 = 0;
    OUTX0 = 1;

    delay(1);
    OUT7 = 1;
    OUT6 = 0;
    OUT5 = 1;
    OUT4 = 0;
    OUT3 = 1;
    OUT2 = 0;
    OUT1 = 1;
    OUT0 = 0;
    OUTX7 = 1;
    OUTX6 = 0;
    OUTX5 = 1;
    OUTX4 = 0;
    OUTX3 = 1;
    OUTX2 = 0;
    OUTX1 = 1;
    OUTX0 = 0;

    goto Mulai;
}

MKanan:
{
    delay(1);
```



Motor3.c

```
OUT7 = 0;  
OUT6 = 1;  
OUT5 = 0;  
OUT4 = 1;  
OUT3 = 0;  
OUT2 = 1;  
OUT1 = 0;  
OUT0 = 1;  
OUTX7 = 0;  
OUTX6 = 1;  
OUTX5 = 0;  
OUTX4 = 1;  
OUTX3 = 0;  
OUTX2 = 1;  
OUTX1 = 0;  
OUTX0 = 1;
```

```
delay(6);  
OUT7 = 1;  
OUT6 = 0;  
OUT5 = 1;  
OUT4 = 0;  
OUT3 = 1;  
OUT2 = 0;  
OUT1 = 1;  
OUT0 = 0;  
OUTX7 = 1;  
OUTX6 = 0;  
OUTX5 = 1;  
OUTX4 = 0;  
OUTX3 = 1;  
OUTX2 = 0;  
OUTX1 = 1;  
OUTX0 = 0;
```

```
goto Mulai;
```

```
}  
}
```

## Features

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

## Description

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

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

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



8-bit  
Microcontroller  
with 8K Bytes  
Flash

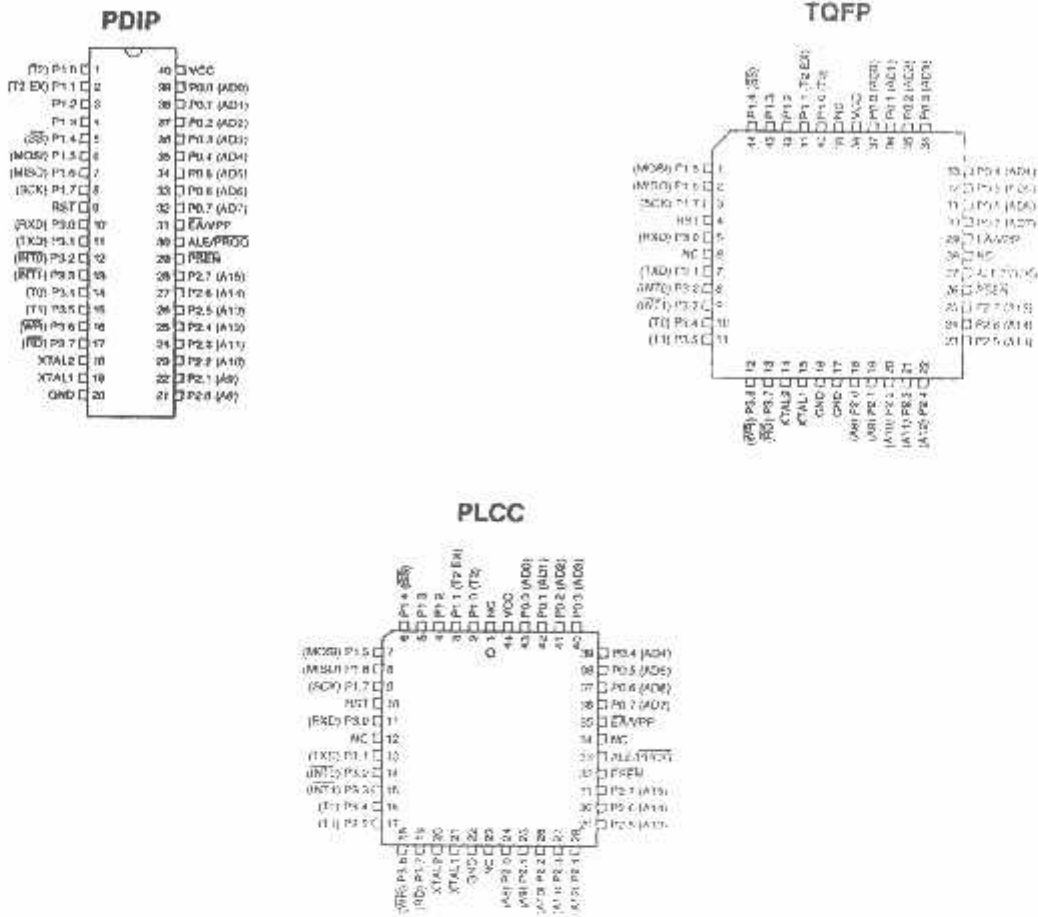
AT89S8252

Not Recommended  
for New Designs.  
Use AT89S8253.

DA113, AUG-90 (01B)



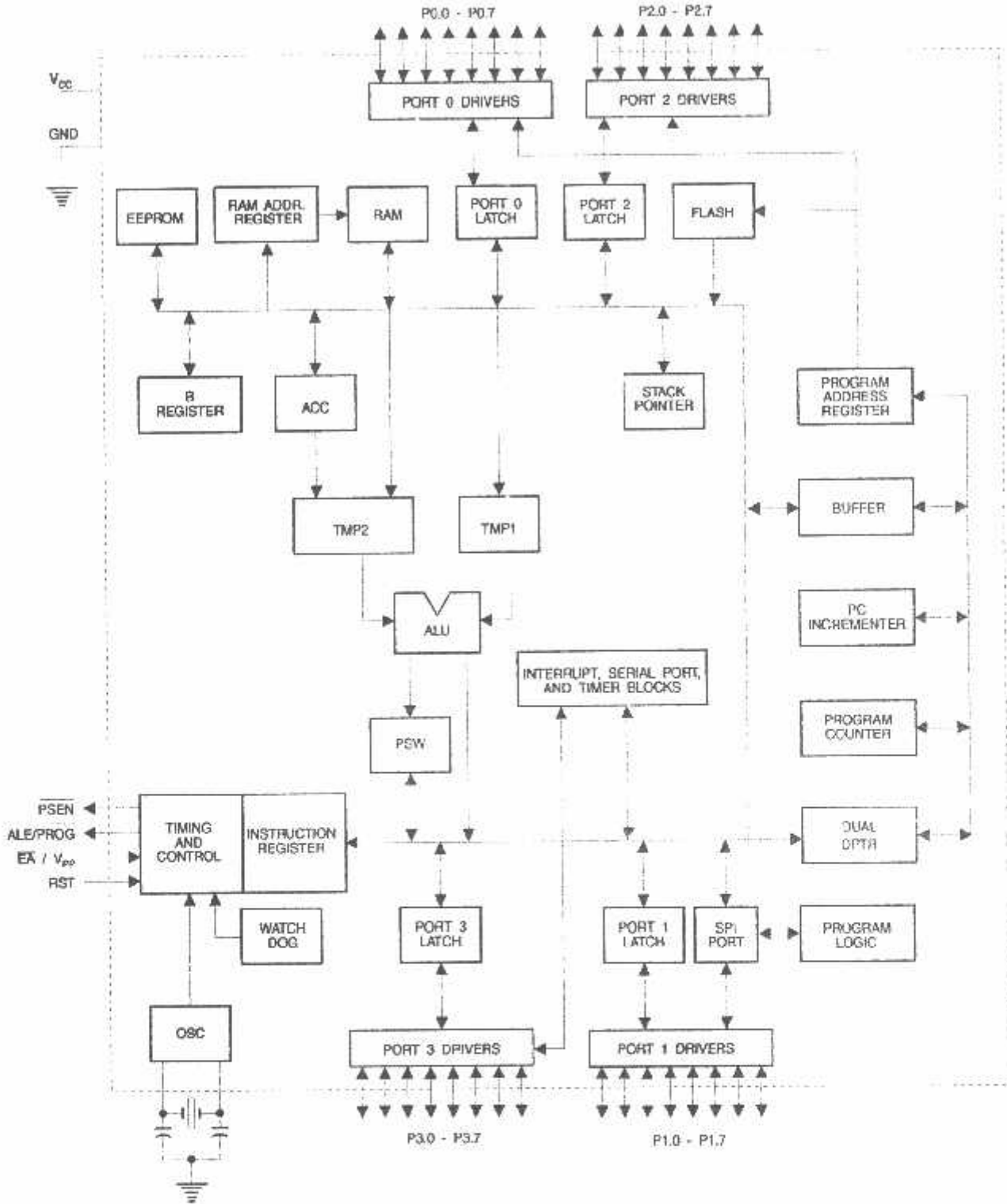
## In Configurations



## n Description

- VCC** Supply voltage.
- ID** Ground.
- Port 0** Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.  
Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups.  
Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification.
- Port 1** Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current ( $I_{OL}$ ) because of the internal pull-ups.

Block Diagram



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

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

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

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

## Port 2

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

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

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

## Port 3

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

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

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

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

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

**$\overline{\text{LE/PROG}}$**  Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ( $\overline{\text{PROG}}$ ) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

**$\overline{\text{SEN}}$**  Program Store Enable is the read strobe to external program memory.

When the AT89S8252 is executing code from external program memory,  $\overline{\text{PSEN}}$  is activated twice each machine cycle, except that two  $\overline{\text{PSEN}}$  activations are skipped during each access to external data memory.

**$\overline{\text{EA/VPP}}$**  External Access Enable.  $\overline{\text{EA}}$  must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed,  $\overline{\text{EA}}$  will be internally latched on reset.

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

**TAL1** Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

**TAL2** Output from the inverting oscillator amplifier.

## Special Function Registers

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

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

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

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

Table 1. AT89S8252 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000					SPCR 000001XX			0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX						0AFH
0A0H	P2 11111111								0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111						WMCON 00000010		97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPCR XXXXXXXX	PCON 0XXX0000	87H

Table 2. T2CON – Timer/Counter 2 Control Register

T2CON Address = 0C8H							Reset Value = 0000 0000B	
Bit Addressable								
	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
Bit	7	6	5	4	3	2	1	0

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





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

**Table 3. WMCON—Watchdog and Memory Control Register**

WMCON Address = 96H		Reset Value = 0000 0010B						
Bit	PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDRST	WDTEN
Bit	7	6	5	4	3	2	1	0
Symbol	Function							
PS2 PS1 PS0	Prescaler Bits for the Watchdog Timer. When all three bits are set to "0", the watchdog timer has a nominal period of 16 ms. When all three bits are set to "1", the nominal period is 2048 ms.							
EEMWE	EEPROM Data Memory Write Enable Bit. Set this bit to "1" before initiating byte write to on-chip EEPROM with the MOVX instruction. User software should set this bit to "0" after EEPROM write is completed.							
EEMEN	Internal EEPROM Access Enable. When EEMEN = 1, the MOVX instruction with DPTR will access on-chip EEPROM instead of external data memory. When EEMEN = 0, MOVX with DPTR accesses external data memory.							
DPS	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1							
WDRST RDY/BSY	Watchdog Timer Reset and EEPROM Ready/Busy Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The WDRST bit is then automatically reset to "0" in the next instruction cycle. The WDRST bit is Write-Only. This bit also serves as the RDY/BSY flag in a Read-Only mode during EEPROM write. RDY/BSY = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/BSY bit equals "0" and is automatically reset to "1" when programming is completed.							
WDTEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.							

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

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

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

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

Table 4. SPCR – SPI Control Register

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

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

**Table 5. SPSR – SPI Status Register**

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

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

**Table 6. SPDR – SPI Data Register**

SPDR Address = 86H				Reset Value = unchanged				
Bit	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
7	6	5	4	3	2	1	0	0

## Data Memory – EEPROM and RAM

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

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

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

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

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

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

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

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

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

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

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

## Programmable Watchdog Timer

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

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

**Table 7. Watchdog Timer Period Selection**

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

## Timer 0 and 1

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

## Timer 2

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

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

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

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

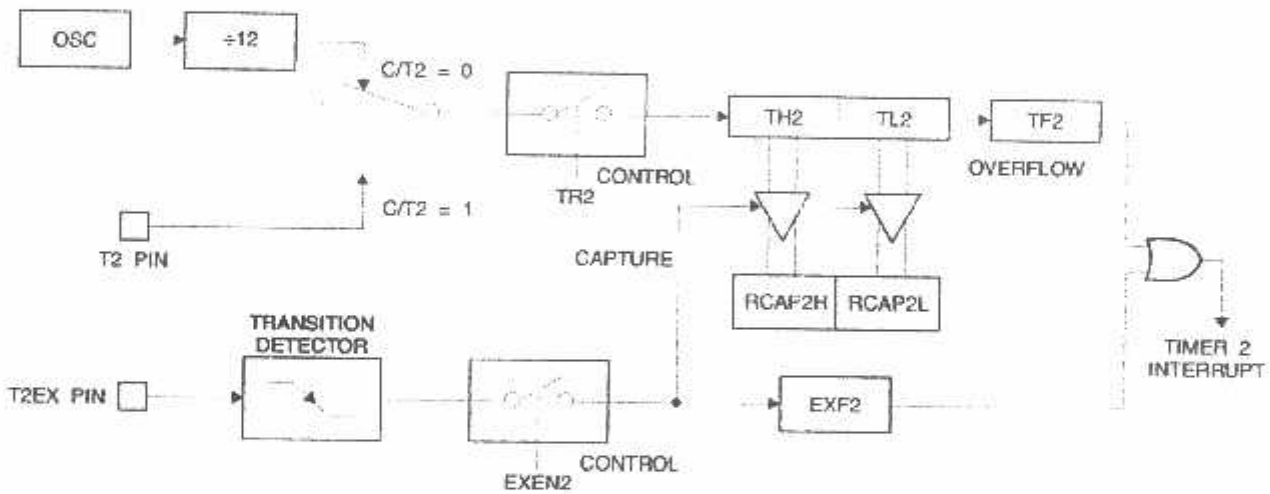
**Table 8. Timer 2 Operating Modes**

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

### Capture Mode

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

**Figure 1. Timer 2 in Capture Mode**



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

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

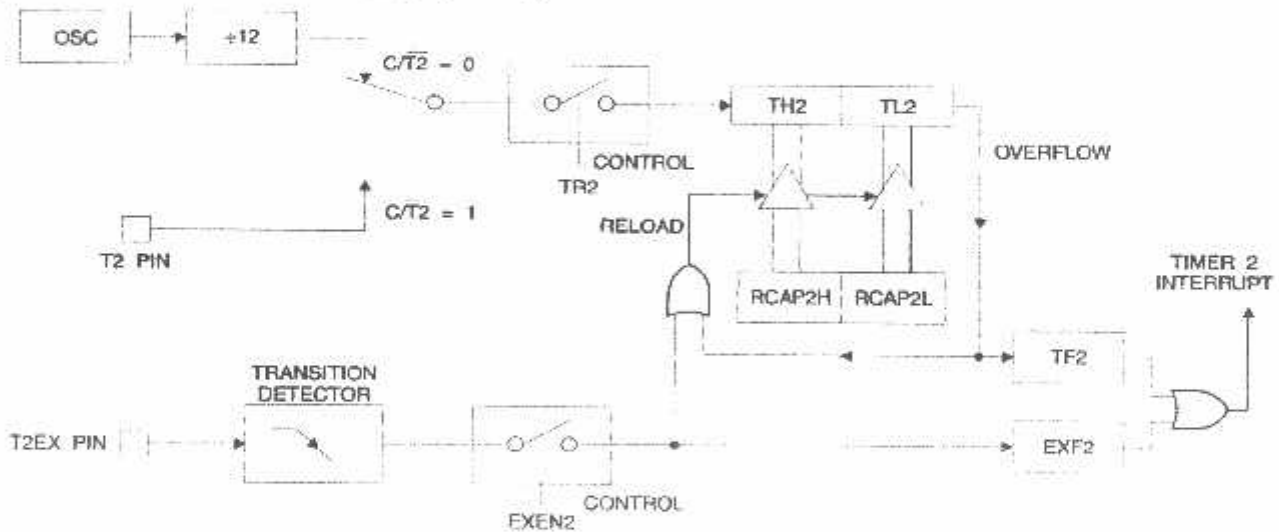
Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H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

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

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

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

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



**Table 9.** T2MOD – Timer 2 Mode Control Register

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

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

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

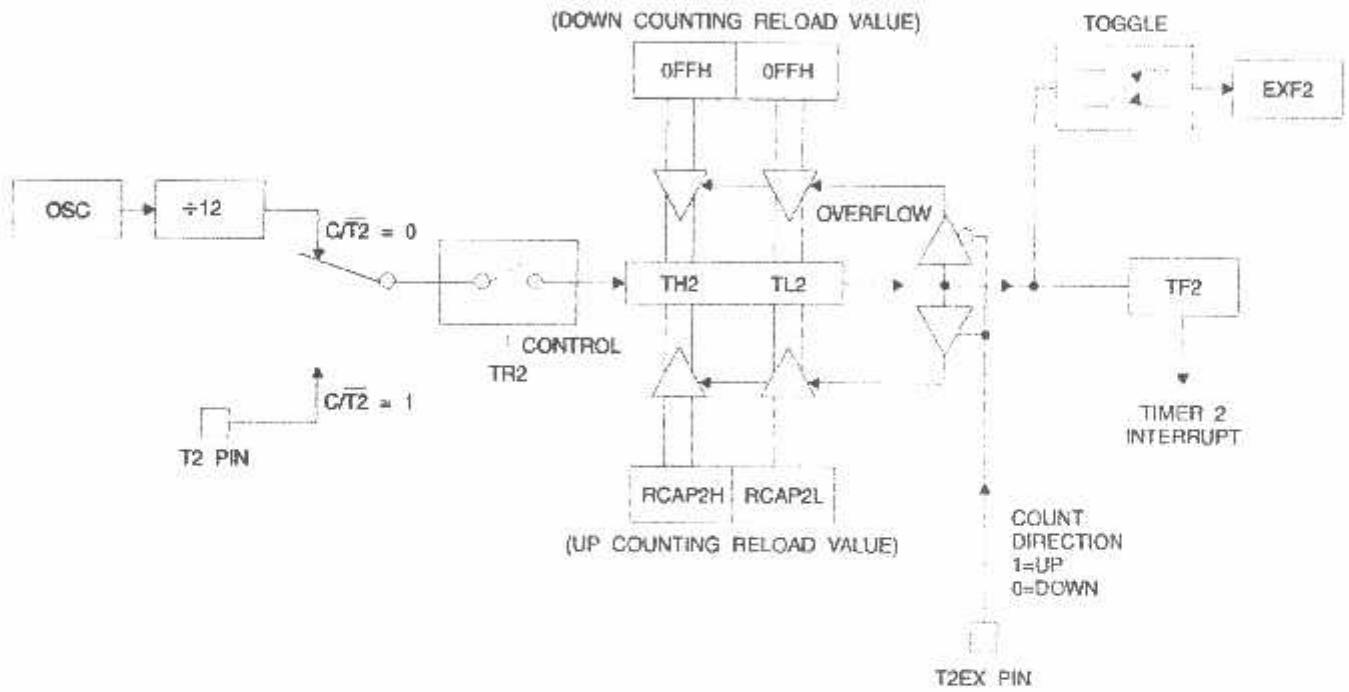
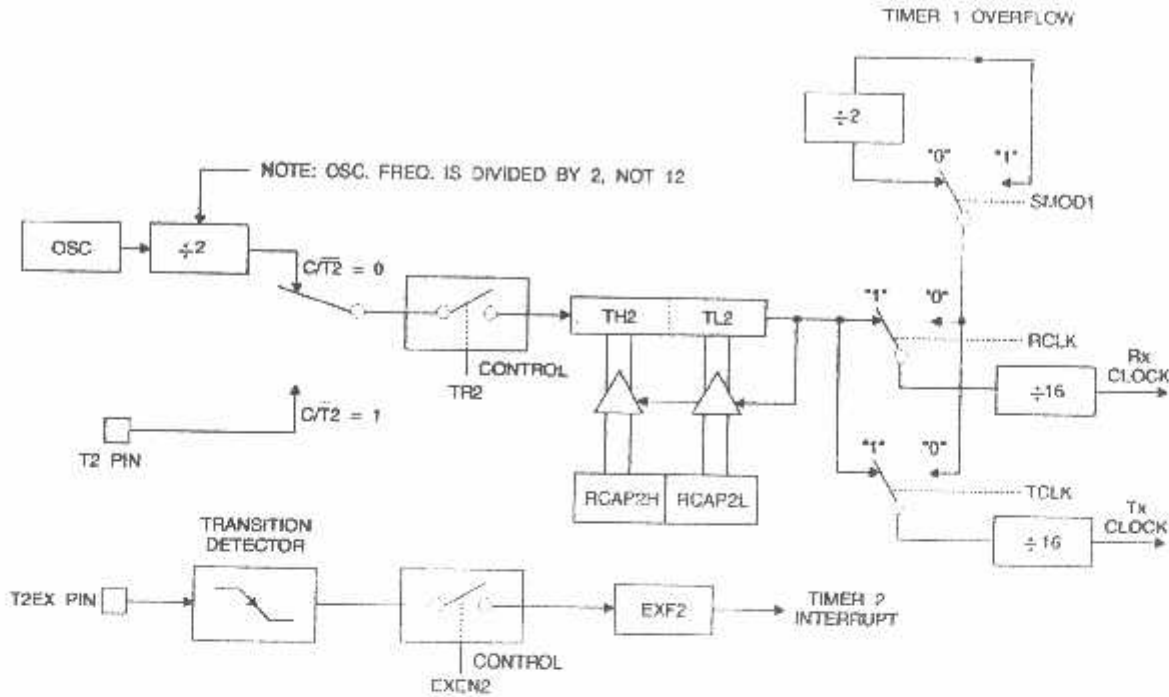


Figure 4. Timer 2 in Baud Rate Generator Mode



### Baud Rate Generator

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

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

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

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

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

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

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



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

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

## Programmable Clock Out

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

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

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

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

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

Figure 5. Timer 2 in Clock-out Mode

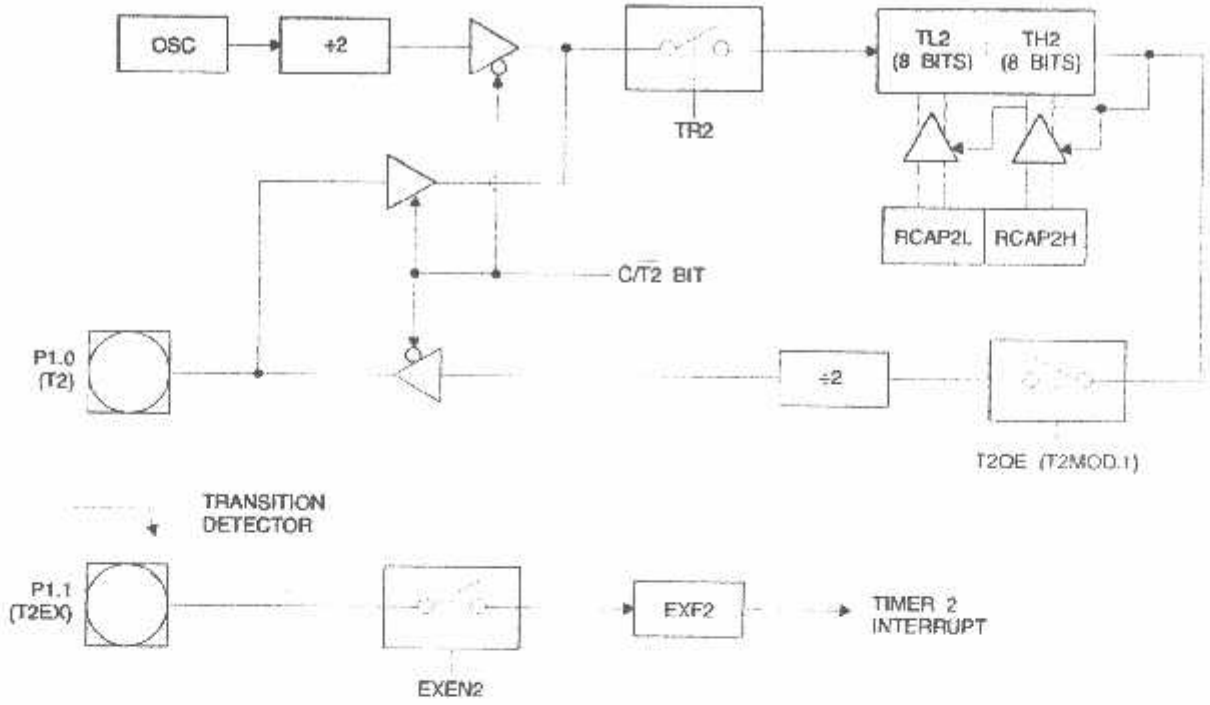
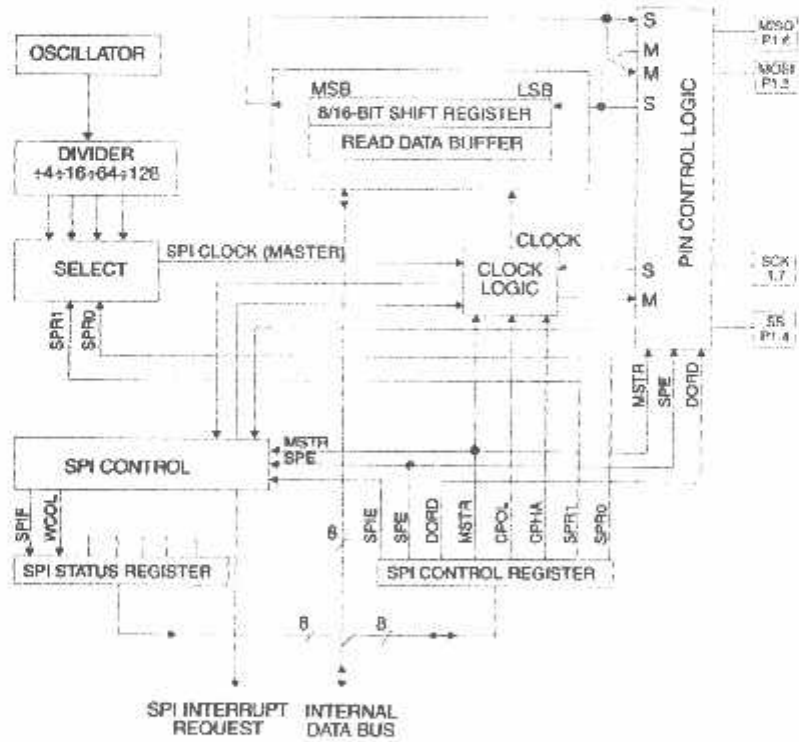


Figure 6. SPI Block Diagram



## UART

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

## Serial Peripheral Interface

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

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

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

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

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

Figure 7. SPI Master-slave Interconnection

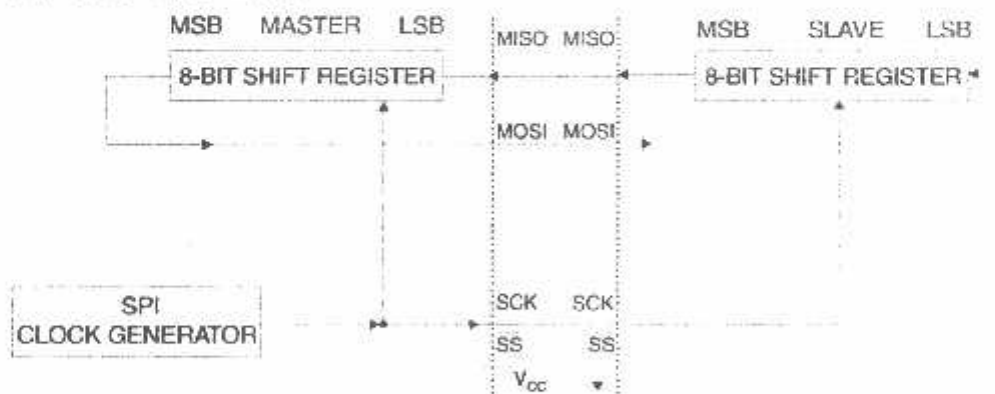
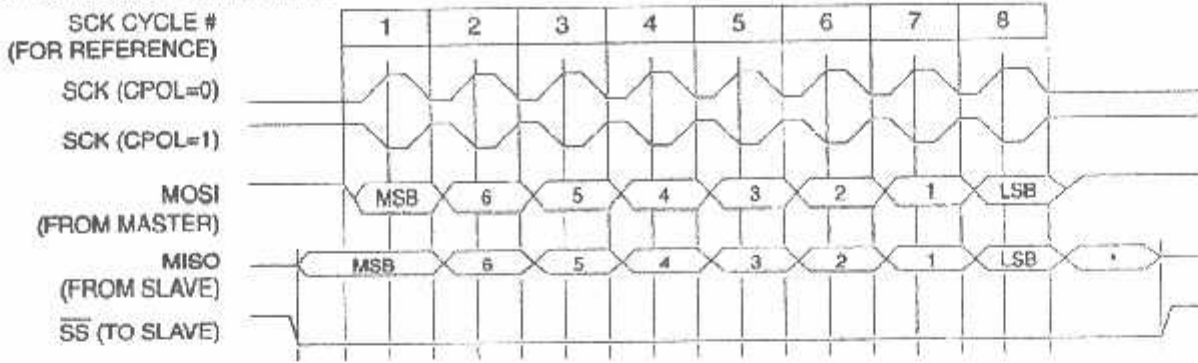
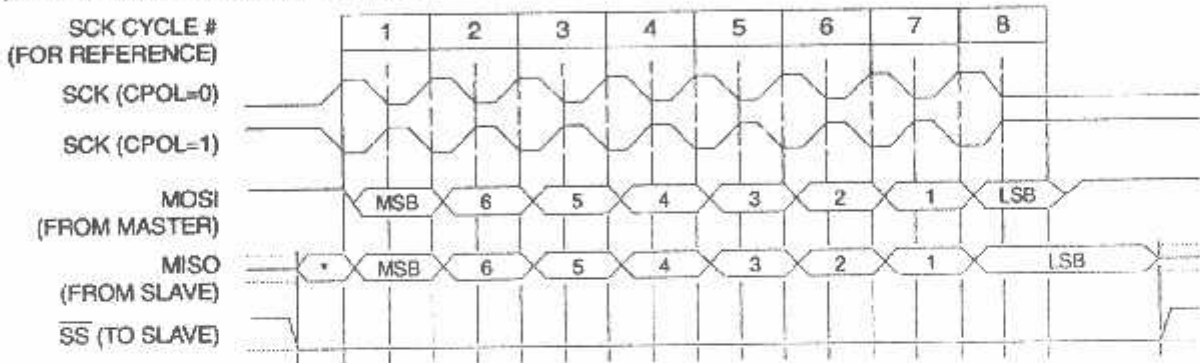


Figure 8. SPI transfer Format with CPHA = 0



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

Figure 9. SPI Transfer Format with CPHA = 1



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

## Interrupts

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

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

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

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

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

**Table 10. Interrupt Enable (IE) Register**

MSB(LSB)

EA	-	ET2	ES	ET1	EX1	ET0	EX0
----	---	-----	----	-----	-----	-----	-----

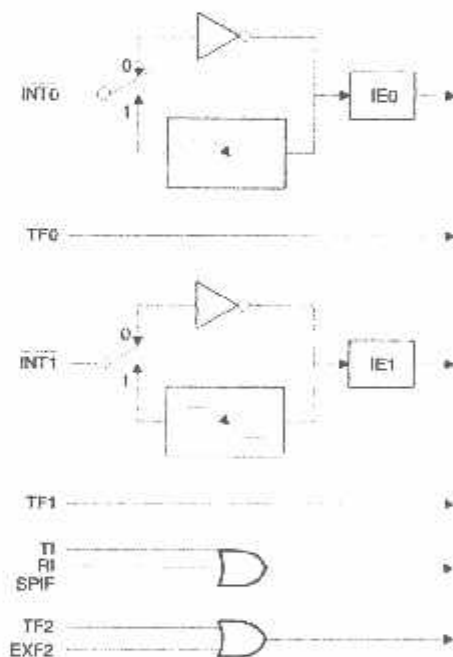
Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

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

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

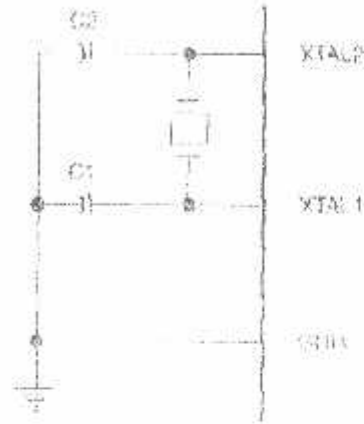
**Figure 10. Interrupt Sources**



## Oscillator Characteristics

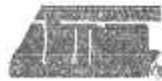
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected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

Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration





## Idle Mode

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

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

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

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Tri-state	Data	Data	Data
Idle	External	1	1	Floating	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Floating	Data	Data	Data

## Power-down Mode

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

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

## Program Memory Lock Bits

The AT89S8252 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

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

Once programmed, the lock bits can only be unprogrammed with the Chip Erase operations in either the parallel or serial modes.

### Lock Bit Protection Modes<sup>(1)(2)</sup>

	Program Lock Bits			Protection type
	LB1	LB2	LB3	
1	U	U	U	No external memory lock feature
2	P	U	U	MCUC instructions executed from external program memory are disabled but instructions from internal memory. EA is sampled and latched on reset, and further programming of the Flash memory (parallel or serial mode) is disabled.
3	P	P	U	Same as Mode 2, but parallel or serial verify are also disabled.
4	P	P	P	Same as Mode 3, but external execution is also disabled.

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

## AT89S8252

## Programming the Flash and EEPROM

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

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

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

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

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

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

1. Power-up sequence:
  - Apply power between  $V_{CC}$  and GND pins.
  - Set RST pin to "H".
  - Apply a 3 MHz to 21 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Set  $\overline{PSEN}$  pin to "L"
  - ALE pin to "H"
  - $\overline{EA}$  pin to "H" and all other pins to "H".
3. Apply the appropriate combination of "H" or "L" logic levels to pins P2.0, P2.7, P3.6, P3.7 to select one of the programming operations shown in the Flash Programming Modes table.
4. Apply the desired byte address to pins P1.0 to P1.7 and P2.0 to P2.5.
  - Apply data to pins P0.0 to P0.7 for Write Code operation.
5. Raise  $\overline{EA}/V_{PP}$  to 12V to enable Flash programming, erase or verification.
6. Pulse ALE ( $\overline{PROMG}$ ) once to program a byte in the Code memory array, the Data memory array or the lock bits. The byte-write cycle is self-timed and typically takes 1.5 ms.
7. To verify the byte just programmed, bring pin P2.7 to "1" and read the programmed data at pins P0.0 to P0.7.
8. Repeat steps 3 through 7 changing the address and data for the entire 2K or 8K bytes array or until the end of the object file is reached.
9. Power-off sequence:
  - Set XTAL1 to "1".
  - Set RST and  $\overline{EA}$  pins to "L".
  - Turn  $V_{CC}$  power off.





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

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

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

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

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

In the serial programming mode, a chip erase operation is initiated by using the Chip Erase instruction. In this mode, chip erase is self-timed and takes about 10 ms.

During chip erase, a serial read from any address local to P0/P1/P2/00H at the data outputs.

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

*The AT89S8252 is shipped with the Serial Programming Mode enabled.*

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

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

## Programming Interface

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

Most well-known microprogramming vendors offer support for the Atmel AT89 microcontroller series. Please contact your local programming vendor for the appropriate software version.

## Serial Downloading

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

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

The Code and Data memory arrays have separate address spaces:

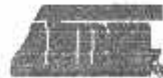
0000H to 1FFFH for Code memory and 000H to 7FFFH for Data memory.

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

## Serial Programming Algorithm

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

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





## Serial Programming Instruction






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

### Instruction Set

Instruction	Input Format			Operation
	Byte 1	Byte 2	Byte 3	
Programming Enable	1010 1100	0101 00*1	xxxx xxxx	Enable serial programming interface after RST goes high.
Chip Erase	1010 1100	xxxx x100	xxxx xxxx	Chip erase both 8K & 2K memory arrays.
Read Code Memory	aaaa a00*	low addr	xxxx xxxx	Read data from Code memory array at selected addresses. The 5 MSBs of the first byte are the high order address bits. The low order address bits are in the second byte. Data are available at pin MISO during the third byte.
Write Code Memory	aaaa a010	low addr	data in	Write data to Code memory location at selected address. The address bits are the 5 MSBs of the first byte together with the second byte.
Read Data Memory	00aa a10*	low addr	xxxx xxxx	Read data from Data memory array at selected address. Data are available at pin MISO during the third byte.
Write Data Memory	00aa a110	low addr	data in	Write data to Data memory location at selected address.
Write Lock Bits	1010 1100	0000 x111 0001	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

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

### Flash and EEPROM Parallel Programming Modes

Mode	RST	PSEN	ALE/PROG	EA/V <sub>PP</sub>	P2.6	P2.7	P3.6	P3.7	Data I/O P0.7:0	Address P2.5:0 P3.7:0
Serial Prog. Modes	H	H <sup>(1)</sup>	H <sup>(1)</sup>	X						
Chip Erase	H	L		12V	H	L	L	L	X	X
Write (10K bytes) Memory	H	L		12V	L	H	H	H	0xV	ADDR
Read (10K bytes) Memory	H	L	L	12V	L	L	H	-	00LT	ADDR
Write Lock Bits	H	L		12V	H	L	H	-	00H	X
B0-1									00L	X
B1-2									00L	X
B2-3									00L	X
B3-4									00L	X
B5-6									00L	X
B7-8									00L	X
Read Lock Bits	L		H	12V	-	H	L	L	00LT	X
B0-1									00LT	X
B2-3									00LT	X
B4-5									00LT	X
B6-7									00LT	X
B8-9									00LT	X
Read Atnel Code	H	L	L	12V	L	L	L	L	00LT	30H
Read Device Code	H	L	L	12V	L	L	L	L	00LT	31H
Read Prog. Enable	H	L		12V	L	H	L	H	00L	X
Read Prog. Disable	H	L		12V	L	H	L	H	00L	X
Read Serial Prog. Fuse	H	L	L	12V	H	H	L	L	00V	X

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





Figure 13. Programming the Flash/EEPROM Memory

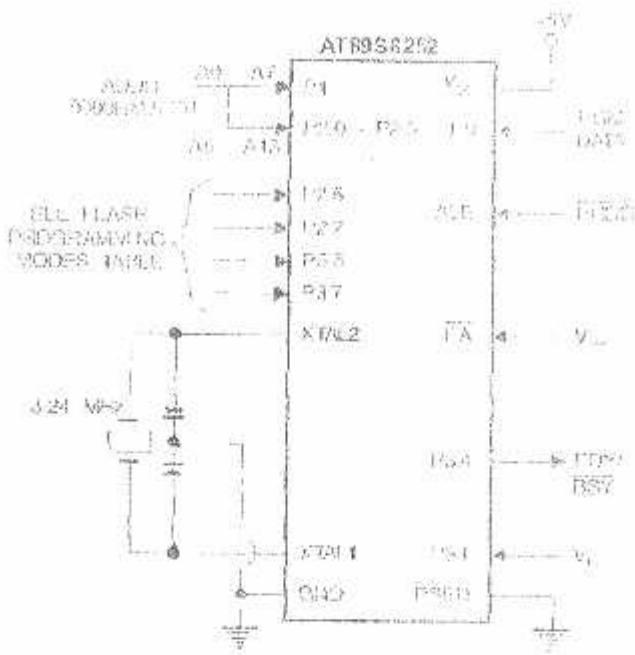


Figure 15. Flash/EEPROM Serial Downloading

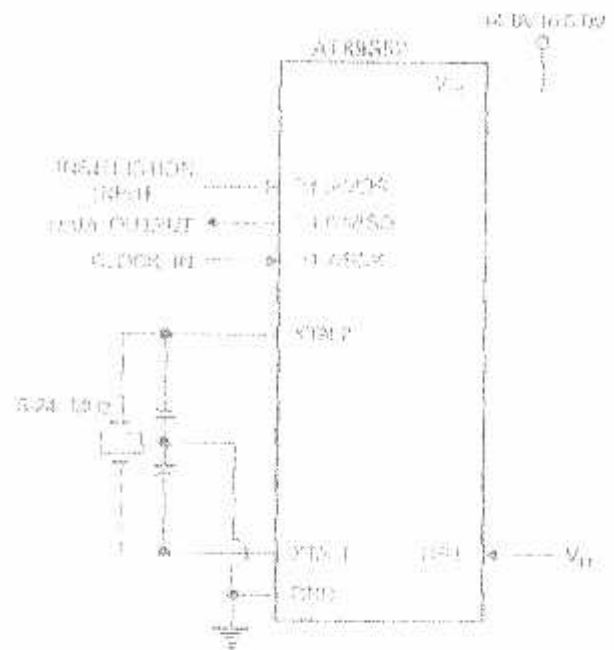
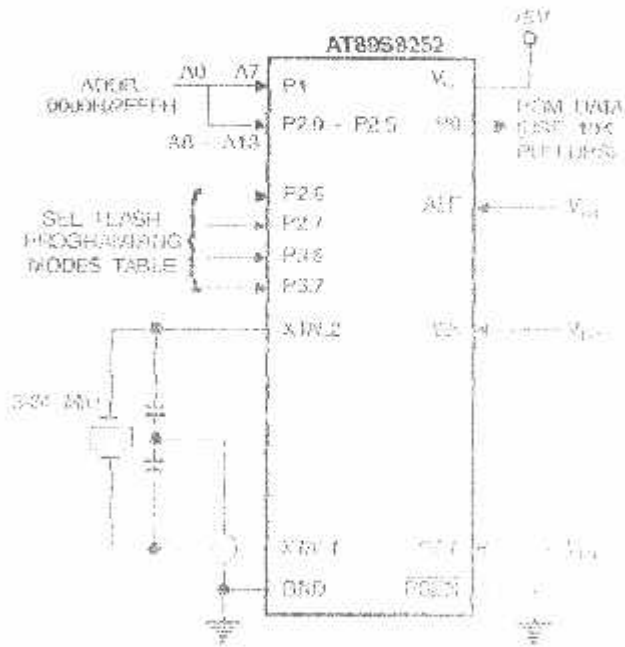


Figure 14. Verifying the Flash/EEPROM Memory

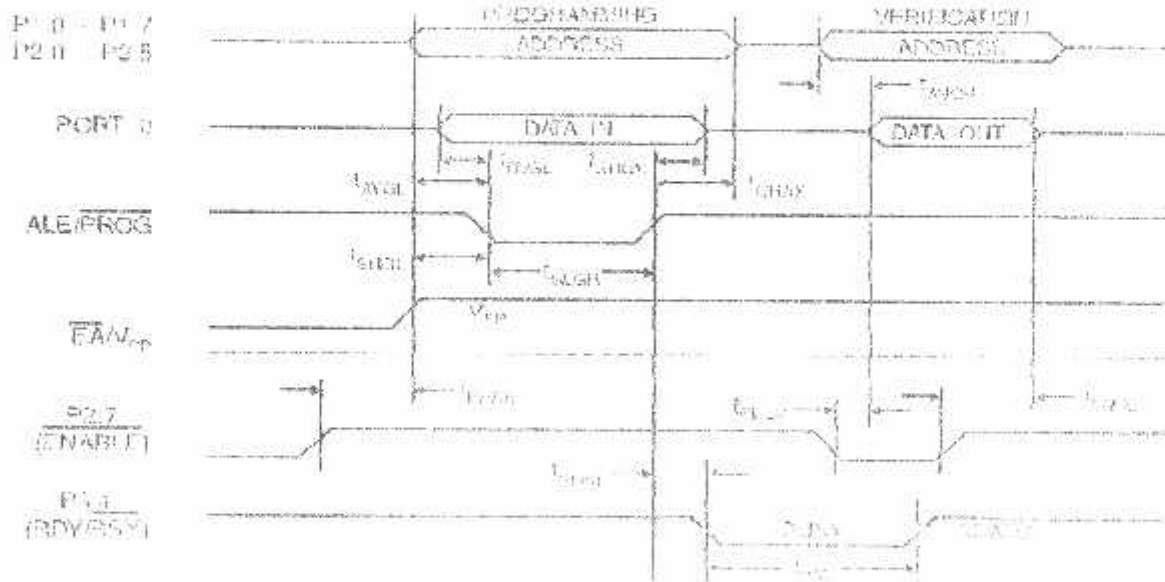


### Flash Programming and Verification Characteristics – Parallel Mode

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

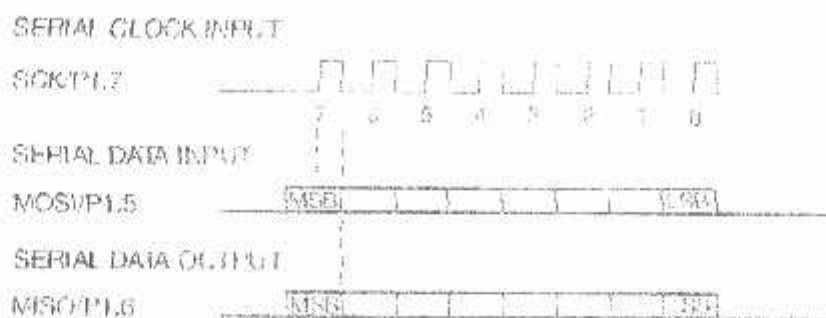
Symbol	Parameter	Min	Max	Units
$V_{PP}$	Programming Enable Voltage	11.5	12.5	V
$I_{PP}$	Programming Enable Current		5	mA
$f_{OSC}$	Oscillator Frequency	5	26	MHz
$t_{ADV}$	Address Setup to $\overline{\text{PROG}}$ Low	$40t_{CLK}$		
$t_{AHZ}$	Address Hold after $\overline{\text{PROG}}$	$48t_{CLK}$		
$t_{DVS}$	Data Setup to $\overline{\text{PROG}}$ Low	$48t_{CLK}$		
$t_{DHS}$	Data Hold after $\overline{\text{PROG}}$	$48t_{CLK}$		
$t_{E2H}$	P2.7 (ENABLE) High to $V_{PP}$	$48t_{CLK}$		
$t_{SPL}$	$V_{PP}$ Setup to $\overline{\text{PROG}}$ Low	10		ns
$t_{PROG}$	$\overline{\text{PROG}}$ Width	1	100	ns
$t_{WAV}$	Address to Data Valid		$60t_{CLK}$	
$t_{ENAV}$	ENABLE Low to Data Valid		$48t_{CLK}$	
$t_{DFLZ}$	Data Float after ENABLE	0	$48t_{CLK}$	
$t_{G+R}$	$\overline{\text{PROG}}$ High to $\overline{\text{BUSY}}$ Low		1.0	ns
$t_{WC}$	Byte Write Cycle Time		2.0	ms

### Flash/EEPROM Programming and Verification Waveforms – Parallel Mode





## Serial Downloading Waveforms



## Serial Programming Characteristics

Figure 16. Serial Programming Timing

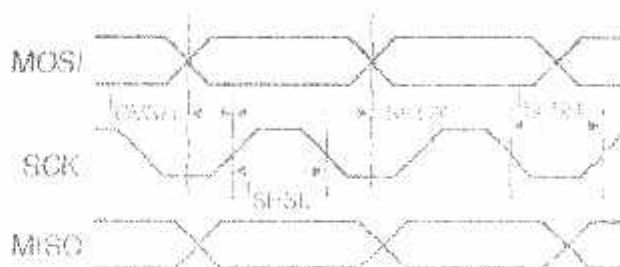


Table 11. Serial Programming Characteristics:  $T_A = -40^\circ\text{C}$  to  $85^\circ\text{C}$ ,  $V_{CC} = 4.0$  to  $5.0\text{V}$  (Unless Otherwise Noted)

Symbol	Parameter	Min	Typ	Max	Units
$f_{\text{OPLL}}$	Oscillator Frequency	0		24	MHz
$t_{\text{OPL}}$	Oscillator Period	41.5			ns
$t_{\text{HSL}}$	SCK Pulse Width High	$2 \times t_{\text{OPL}}$			ns
$t_{\text{LSL}}$	SCK Pulse Width Low	$24 t_{\text{OPL}}$			ns
$t_{\text{MOSI}}$	MOSI Setup to SCK High	$t_{\text{OPL}}$			ns
$t_{\text{HOSI}}$	MOSI Hold after SCK High	$2 t_{\text{OPL}}$			ns

**Absolute Maximum Ratings\***

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.6V
Maximum Operating Voltage	5.5V
DC Output Current	15.0 mA

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

**DC Characteristics**

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

Symbol	Parameter	Condition	Min	Max	Units
$V_{IL}$	Input Low voltage	(Except EA)	0.1	$0.7 V_{CC} - 0.1$	V
$V_{IH}$	Input High voltage (EA)		1.5	$0.8 V_{CC} + 0.5$	V
$V_{OL}$	Output Low voltage	(Except XTAL1, PSEN)	$0.2 V_{CC} - 0.9$	$V_{CC} - 0.5$	V
$V_{OH}$	Output High voltage	(XTAL1, PSEN)	$0.7 V_{CC}$	$V_{CC} + 0.2$	V
$V_{OL1}$	Output Low voltage <sup>(1)</sup> (Ports 1, 2, 3)	$I_{OL} = 1.5\text{ mA}$		0.5	V
$V_{OL2}$	Output Low voltage <sup>(2)</sup> (Port 0, ALE, PSEN)	$I_{OL} = 3.2\text{ mA}$		0.5	V
$V_{OH1}$	Output High voltage (Ports 1, 2, 3, ALE, PSEN)	$I_{OH} = -50\ \mu\text{A}$ , $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -25\ \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -10\ \mu\text{A}$	$0.9 V_{CC}$		V
$V_{OH2}$	Output High voltage (Port 0) external diode Model	$I_{OH} = -800\ \mu\text{A}$ , $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -300\ \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -80\ \mu\text{A}$	$0.9 V_{CC}$		V
$I_{L1}$	Logical '0' Input Current (Ports 1, 2, 3)	$V_{IH} = 0.45\text{V}$		-50	$\mu\text{A}$
$I_{L2}$	Logical '1' Input Transition Current (Ports 1, 2, 3)	$V_{IH} = 2\text{V}$ , $V_{CC} = 5\text{V} \pm 10\%$		-250	$\mu\text{A}$
$I_{L3}$	Input Leakage Current (Port 0 EA)	$0.45 < V_{IN} < V_{CC}$		±10	$\mu\text{A}$
RRST	Reset Pull-down Resistor		50	100	k $\Omega$
$C_{in}$	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		14	pF
$I_{CC}$	Power Supply Current	Active Mode, 12 MHz		5	mA
		Idle Mode, 12 MHz		0.5	mA
	Power-down Mode <sup>(3)</sup>	$V_{CC} = 6\text{V}$		100	$\mu\text{A}$
		$V_{CC} = 3\text{V}$		60	$\mu\text{A}$

- Notes: 1. Under steady state (non-transient) conditions,  $I_{OL}$  must be externally limited as follows:  
 Maximum  $I_{OL}$  per port pin: 10 mA  
 Maximum  $I_{OL}$  per 8-bit port, Port 0: 20 mA; Ports 1, 2, 3: 15 mA  
 Maximum total  $I_{OL}$  for all output pins: 70 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  $V_{OH}$  for Power-down is 2V.







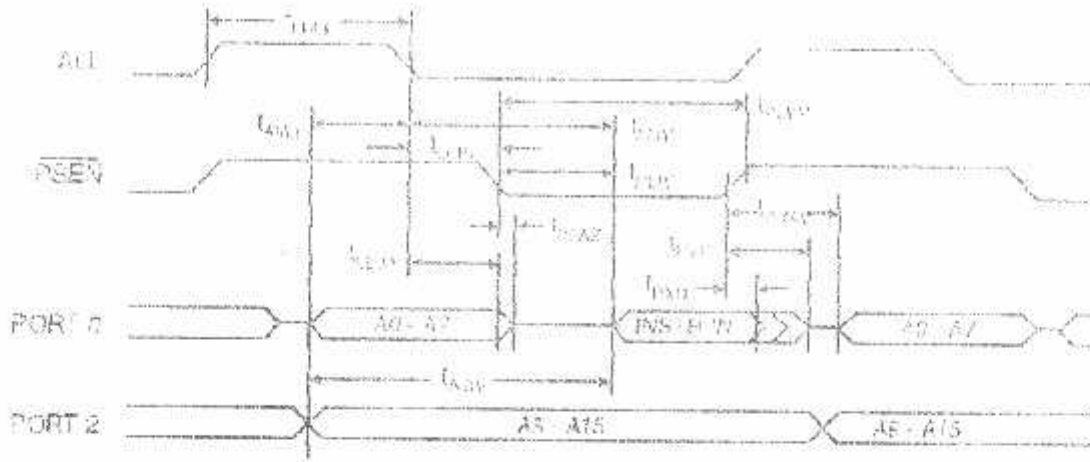
## AC Characteristics

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

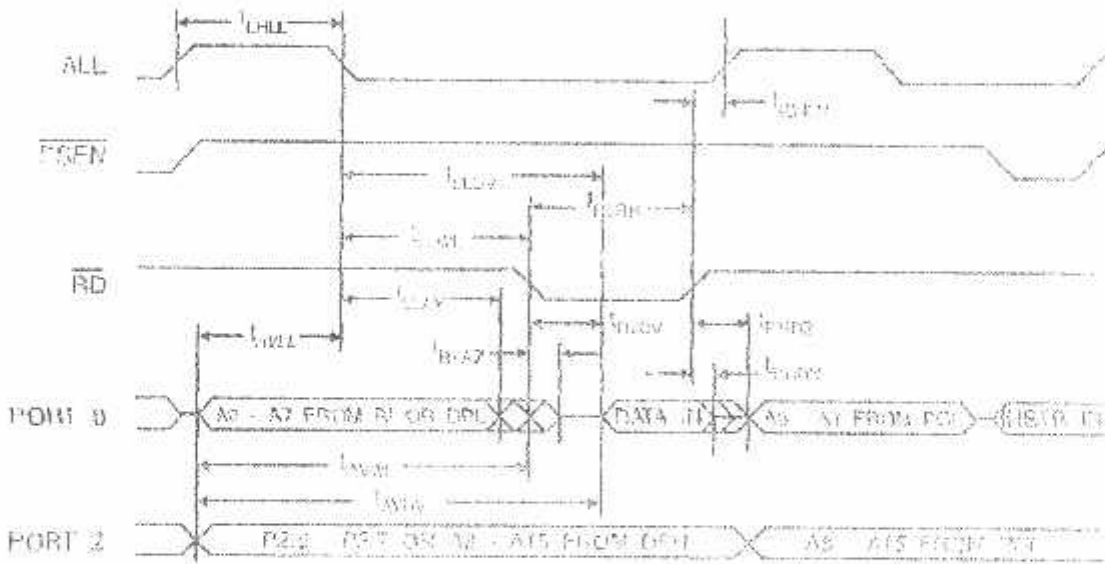
### External Program and Data Memory Characteristics

Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
$f_{osc}$	Oscillator Frequency	0	24	MHz
$t_{ALE}$	ALE Pulse Width	$2t_{osc} + 40$		ns
$t_{AVL}$	Address Valid to ALE Low	$t_{osc} + 10$		ns
$t_{AH}$	Address Hold after ALE Low	$t_{osc} + 20$		ns
$t_{ALEI}$	ALE Low to Valid Instruction In		$4t_{osc} + 60$	ns
$t_{ALEP}$	ALE Low to PSEN Low	$t_{osc} + 10$		ns
$t_{PSEW}$	PSEN Pulse Width	$3t_{osc} + 20$		ns
$t_{PSEI}$	PSEN Low to Valid Instruction In		$3t_{osc} + 45$	ns
$t_{PIH}$	Input Instruction Hold after PSEN	0		ns
$t_{PIZ}$	Input Instruction Float after PSEN		$t_{osc} + 10$	ns
$t_{P2AV}$	PSEN to Address Valid	$t_{osc} + 8$		ns
$t_{AVP}$	Address to Valid Instruction In		$5t_{osc} + 65$	ns
$t_{P2AZ}$	PSEN Low to Address Float		10	ns
$t_{RDW}$	RD Pulse Width	$5t_{osc} + 100$		ns
$t_{WRW}$	WR Pulse Width	$5t_{osc} + 100$		ns
$t_{RDV}$	RD Low to Valid Data In		$5t_{osc} + 100$	ns
$t_{RDH}$	Data Hold after RD	0		ns
$t_{RDZ}$	Data Float after RD		$2t_{osc} + 80$	ns
$t_{RDV2}$	ALE Low to Valid Data In		$8t_{osc} + 120$	ns
$t_{RDV1}$	Address to Valid Data In		$3t_{osc} + 100$	ns
$t_{RDV1}$	ALE Low to RD or WR Low	$3t_{osc} + 60$	$3t_{osc} + 60$	ns
$t_{RDV2}$	Address to RD or WR Low	$4t_{osc} + 70$		ns
$t_{DVAH}$	Data Valid to WR Transition	$t_{osc} + 20$		ns
$t_{DVWH}$	Data Valid to WR High	$7t_{osc} + 120$		ns
$t_{WDH}$	Data Hold after WR	$t_{osc} + 20$		ns
$t_{RDZ1}$	RD Low to Address Float		0	ns
$t_{RDZ2}$	RD or WR High to ALE High	$t_{osc} + 20$	$t_{osc} + 25$	ns

External Program Memory Read Cycle

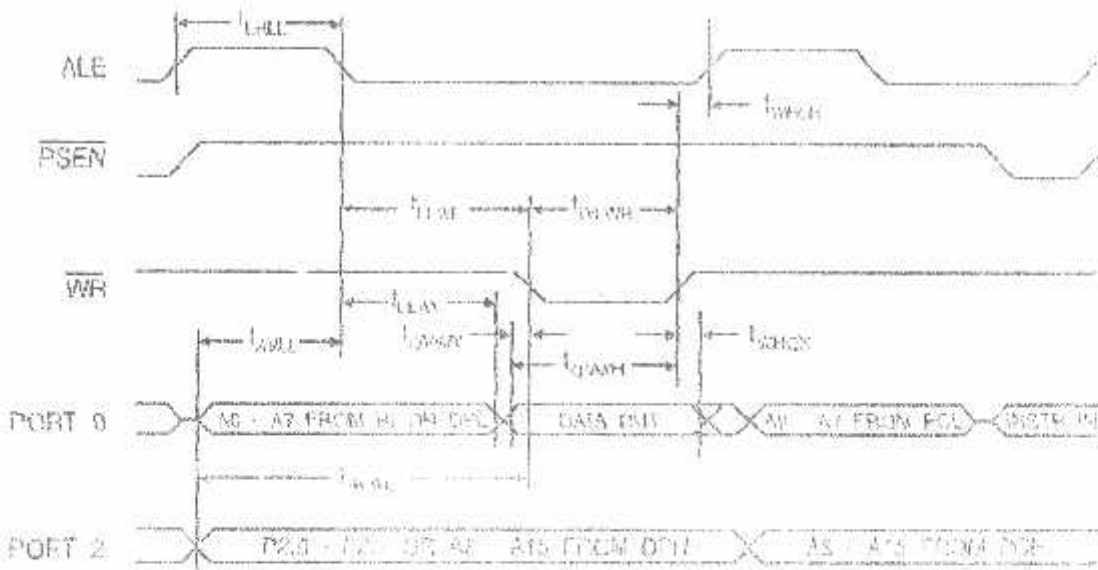


External Data Memory Read Cycle





## External Data Memory Write Cycle



## External Clock Drive Waveforms



## External Clock Drive

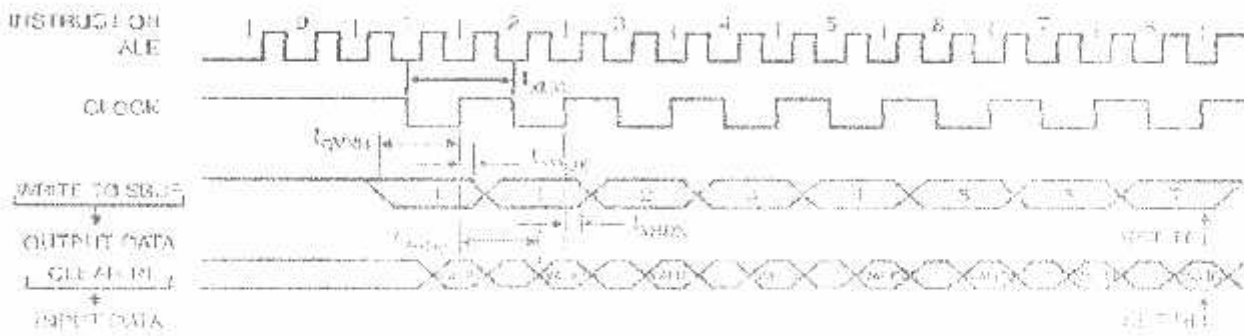
Symbol	Parameter	$V_{CC} = 4.0V \text{ to } 6.0V$		Units
		Min	Max	
$f_{CLK}$	Oscillator Frequency	0	24	MHz
$T_{CLK}$	Clock Period	41.6		ns
$t_{RCLK}$	High Time	15		ns
$t_{LCLK}$	Low Time	15		ns
$t_{SUCLK}$	Rise Time		20	ns
$t_{FALLCLK}$	Fall Time		20	ns

### Serial Port Timing: Shift Register Mode Test Conditions

no values in this table are valid for  $V_{CC} = 4.0V$  to  $5V$  and Load Capacitance =  $80 pF$ .

Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
$t_{CLK}$	Serial Port Clock Cycle Time	$12\%_{V_{CC}}$		$\mu s$
$t_{OVSE}$	Output Data Setup to Clock Rising Edge	$10t_{CLK} + 133$		ns
$t_{OHSE}$	Output Data Hold after Clock Rising Edge	$2t_{CLK} + 17$		ns
$t_{IHSE}$	Input Data Hold after Clock Rising Edge	0		ns
$t_{HSDV}$	Clock Rising Edge to Input Data Valid		$10t_{CLK} + 133$	ns

### Shift Register Mode Timing Waveforms



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



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

### Float Waveforms<sup>(1)</sup>

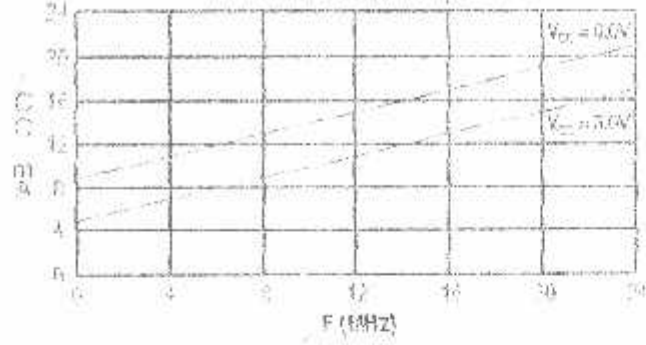


Note: 1. For timing purposes, a data pin is "floating" if it is having a  $>100 mV$  change from its relaxed (output) condition or is not held at that level when a  $>100 mV$  change from the level of  $V_{OH}$  or  $V_{OL}$  level occurs.



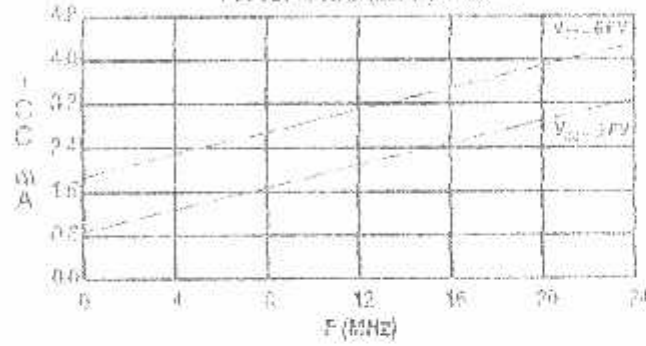
AT89S8252

TYPICAL  $I_{CC1}$  (ACTIVE) AT 25°C



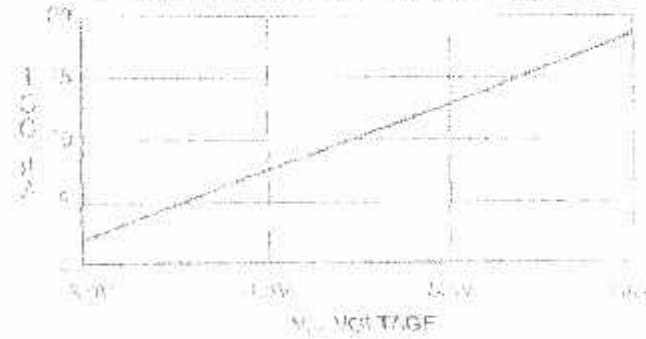
AT89S8252

TYPICAL  $I_{CC1}$  (IDLE) AT 25°C



AT89S8252

TYPICAL  $I_{CC1}$  & VOLTAGE POWER (POWER) AT 25°C



- Notes: 1. X At 1 level to GND for test (power shown)  
 2. Load bits programmed

Ordering Information

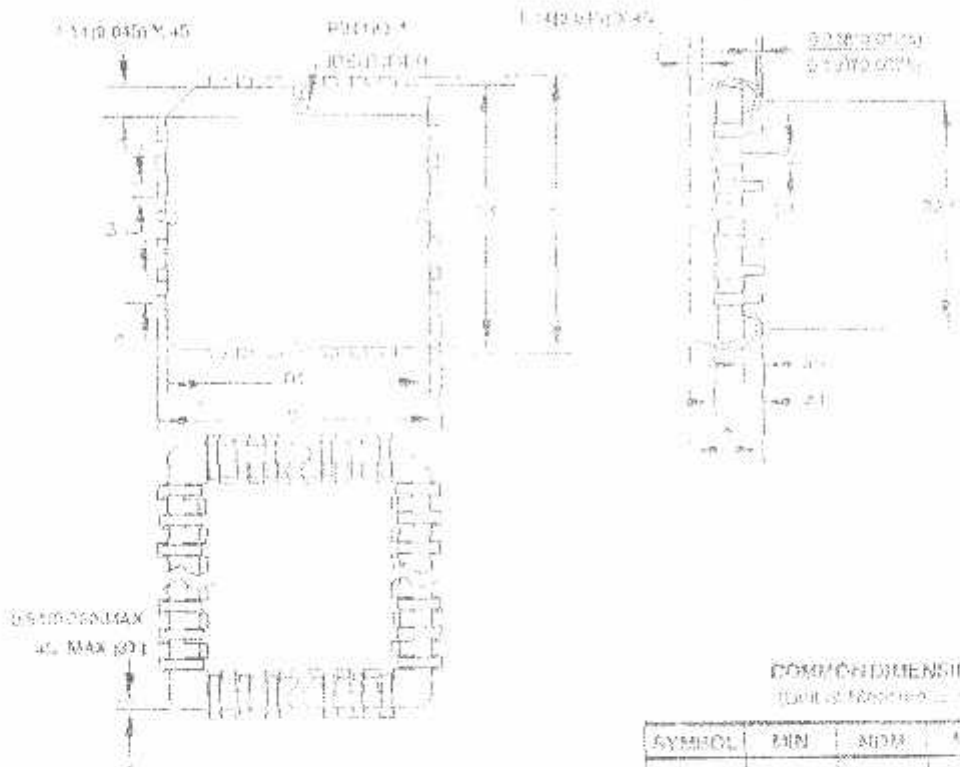
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 6.0V	AT89S8252-24AC	44A	Commercial (0°C to 70°C)
		AT89S8252-24JC	44J	
		AT89S8252-24PC	40P6	
	4.0V to 5.0V	AT89S8252-24AI	44A	Industrial (-40°C to 85°C)
		AT89S8252-24JI	44J	
		AT89S8252-24PI	40P6	

Package Type	
44A	44-lead, Thin Plastic Gull Wing 0.4mm Flatpack (TQFP)
44J	44-lead, Plastic J-Header Chip Carrier (PICO)
40P6	40-lead, 0.600" Wide, Plastic Dual In-Line Package (PDIP)





4J - PLCC



COMMON DIMENSIONS  
(Units: Millimeters - mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	4.121		4.778	
AA	2.260		2.540	
AC	1.508			
AD	17.530		17.781	
AE	16.540		17.017	Dim 2
AF	17.540		17.791	
AG	16.540		17.017	Dim 2
D1-D2	14.950		15.240	
B	0.500		0.610	
BF	0.330		0.440	
Ø		1.675 TYP		

- Notes:
1. This package conforms to JEDEC reference MS-016, Variation A2.
  2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.10 to 0.20 mm on side. Dimensions D1 and E1 include mold flash and protrusion and are measured at the extreme external condition at the top of the package leads.
  3. Lead coplanarity is 0.08 mm (0.0031 inch) maximum.

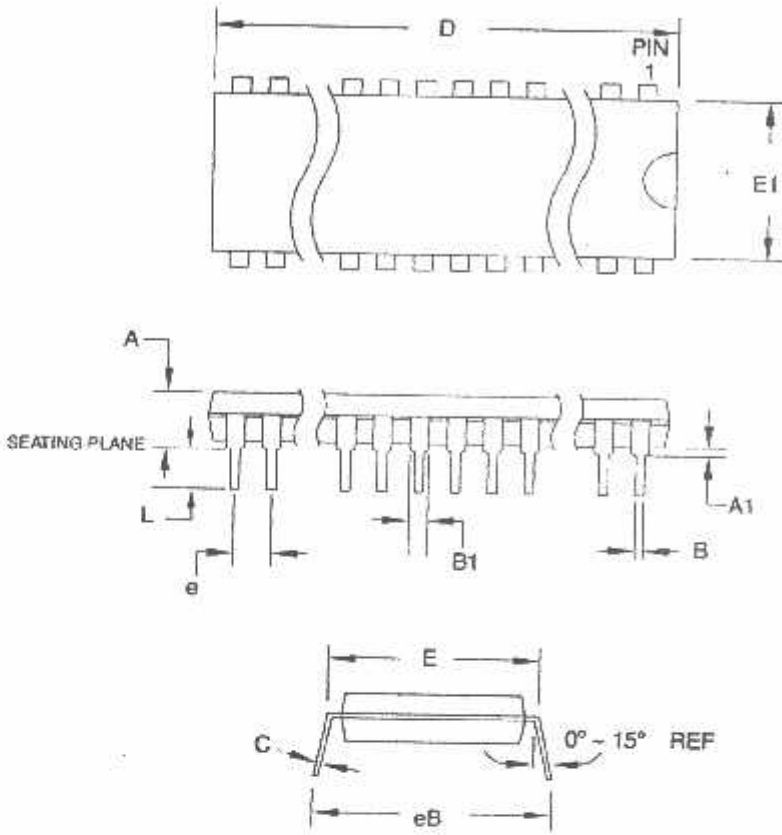
120403

2005 Orchard Parkway San Jose, CA 95131	TITLE	DRAWING NO.	REV.
	44J-44-pin, Parallel-Loaded Chip Carrier (PLCC)	44J	1/8





40P6 – PDIP



COMMON DIMENSIONS  
(Unit of Measure = mm)

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

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

09/28/01



2325 Orchard Parkway  
San Jose, CA 95131

TITLE

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

DRAWING NO.

40P6

REV.

B



## Atmel Corporation

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

## Regional Headquarters

### Europe

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

### Asia

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

### Japan

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

## Atmel Operations

### Memory

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

### Microcontrollers

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

### La Chantrerie

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

### ASIC/ASSP/Smart Cards

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

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

### Scottish Enterprise Technology Park

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

### RF/Automotive

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

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

### Biometrics/Imaging/Hi-Rel MPU/ High-Speed Converters/RF Datacom

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

## Literature Requests

[www.atmel.com/literature](http://www.atmel.com/literature)

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

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



Printed on recycled paper.

0101C-MICRO-3/06

xM

## ADC0808/ADC0809 8-Bit $\mu$ P Compatible A/D Converters with 8-Channel Multiplexer

### General Description

The ADC0808, ADC0809 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 8-channel multiplexer and microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 8-channel multiplexer can directly access any of 8 single-ended analog signals.

The device eliminates the need for external zero and full-scale adjustments. Easy interfacing to microprocessors is provided by the latched and decoded multiplexer address inputs and latched TTL TRI-STATE<sup>®</sup> outputs.

The design of the ADC0808, ADC0809 has been optimized by incorporating the most desirable aspects of several A/D conversion techniques. The ADC0808, ADC0809 offers high speed, high accuracy, minimal temperature dependence, excellent long-term accuracy and repeatability, and consumes minimal power. These features make this device ideally suited to applications from process and machine control to consumer and automotive applications. For 16-channel multiplexer with common output (sample/hold port) see ADC0816 data sheet. (See AN-247 for more information.)

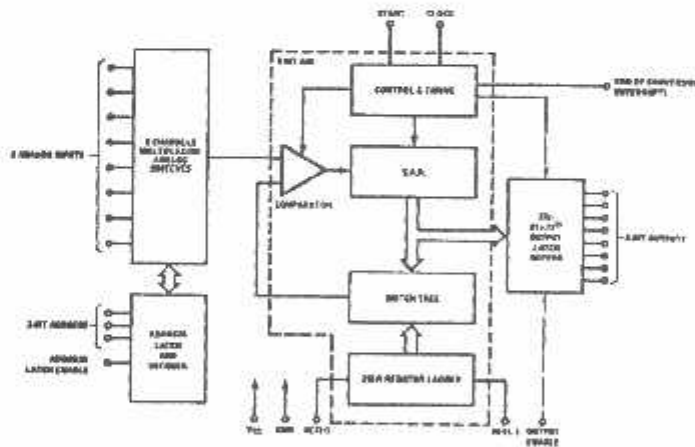
### Features

- Easy interface to all microprocessors
- Operates ratiometrically or with 5 V<sub>DC</sub> or analog span adjusted voltage reference
- No zero or full-scale adjust required
- 8-channel multiplexer with address logic
- 0V to 5V input range with single 5V power supply
- Outputs meet TTL voltage level specifications
- Standard hermetic or molded 28-pin DIP package
- 28-pin molded chip carrier package
- ADC0808 equivalent to MM74C09-9
- ADC0809 equivalent to MM74C09-1

### Key Specifications

- |                          |                               |
|--------------------------|-------------------------------|
| ■ Resolution             | 8 Bits                        |
| ■ Total Unadjusted Error | $\pm 1/2$ LSB and $\pm 1$ LSB |
| ■ Single Supply          | 5 V <sub>DC</sub>             |
| ■ Low Power              | 15 mW                         |
| ■ Conversion Time        | 100 $\mu$ s                   |

### Block Diagram



TRI-STATE<sup>®</sup> is a registered trademark of National Semiconductor Corp.

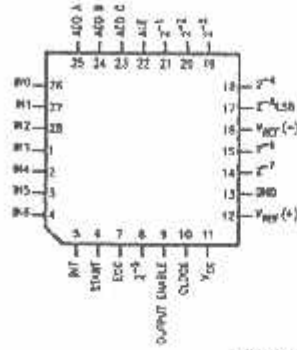
### Connection Diagrams

Dual-In-Line Package



Order Numbers ADC0808CCN or ADC0809CCN  
See NS Package J28A or N28A

Molded Chip Carrier Package



Order Number ADC0808CCV or ADC0809CCV  
See NS Package V28A

### Ordering Information

TEMPERATURE RANGE		-40°C to +85°C			-55°C to +125°C
Error	± 1/2 LSB Unadjusted	ADC0808CCN	ADC0808CCV	ADC0808CCJ	ADC0808CJ
	±1 LSB Unadjusted	ADC0809CCN	ADC0809CCV		
Package Outline		N28A Molded DIP	V28A Molded Chip Carrier	J28A Ceramic DIP	J28A Ceramic DIP

### Absolute Maximum Ratings (Notes 2, 1)

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

Supply Voltage ( $V_{CC}$ ) (Note 3)	6.5V
Voltage at Any Pin	-0.3V to ( $V_{CC}+0.3V$ )
Except Control Inputs	
Voltage at Control Inputs	-0.3V to +15V
(START, OE, CLOCK, ALE, ADD A, ADD B, ADD C)	
Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A=25^\circ\text{C}$	875 mW
Lead Temp. (Soldering, 10 seconds)	
Dual-In-Line Package (plastic)	260°C

Dual-In-Line Package (ceramic)	300°C
Molded Chip Carrier Package	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C
ESD Susceptibility (Note 8)	400V

### Operating Conditions (Notes 1, 2)

Temperature Range (Note 1)	$T_{MIN}$ to $T_A$ to $T_{MAX}$
ADC0808CCN, ADC0809CCN	-40°C to $T_A$ to +85°C
ADC0808CCV, ADC0809CCV	-40°C to $T_A$ to +85°C
Range of $V_{CC}$ (Note 1)	4.5 $V_{DC}$ to 6.0 $V_{DC}$

### Electrical Characteristics

Converter Specifications:  $V_{CC} \approx 5 V_{CC} = V_{REF(+)}$ ,  $V_{REF(-)} = \text{GND}$ ,  $T_{MIN} \leq T_A \leq T_{MAX}$  and  $f_{CLK} = 640 \text{ kHz}$  unless otherwise stated.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	ADC0808					
	Total Unadjusted Error	25°C			±1/2	LSB
	(Note 5)	$T_{MIN}$ to $T_{MAX}$			±3/4	LSB
	ADC0809					
	Total Unadjusted Error	0°C to 70°C			±1	LSB
	(Note 5)	$T_{MIN}$ to $T_{MAX}$			±1 1/4	LSB
	Input Resistance	From Ref(+) to Ref(-)	1.0	2.5		kΩ
	Analog Input Voltage Range	(Note 4) V(+) or V(-)	GND-0.10		$V_{CC}+0.10$	$V_{DC}$
$V_{REF(+)}$	Voltage, Top of Ladder	Measured at Ref(+)		$V_{CC}$	$V_{CC}+0.1$	V
$\frac{V_{REF(+)} + V_{REF(-)}}{2}$	Voltage, Center of Ladder		$V_{CC}/2-0.1$	$V_{CC}/2$	$V_{CC}/2+0.1$	V
$V_{REF(-)}$	Voltage, Bottom of Ladder	Measured at Ref(-)	-0.1	0		V
$I_{IN}$	Comparator Input Current	$f_c = 640 \text{ kHz}$ , (Note 6)	-2	±0.5	2	μA

### Electrical Characteristics

Digital Levels and DC Specifications: ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV,  $4.75 < V_{CC} < 5.25V$ ,  $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$  unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>ANALOG MULTIPLEXER</b>						
$I_{OFF(+)}$	OFF Channel Leakage Current	$V_{CC}=5V$ , $V_{IN}=5V$ , $T_A=25^\circ\text{C}$ $T_{MIN}$ to $T_{MAX}$		10	200	nA μA
$I_{OFF(-)}$	OFF Channel Leakage Current	$V_{CC}=5V$ , $V_{IN}=0$ , $T_A=25^\circ\text{C}$ $T_{MIN}$ to $T_{MAX}$	-200 -1.0	-10		nA μA
<b>CONTROL INPUTS</b>						
$V_{REF(+)}$	Logical "1" Input Voltage		$V_{CC}-1.5$			V
$V_{REF(-)}$	Logical "0" Input Voltage				1.5	V
$I_{IN(+)}$	Logical "1" Input Current (The Control Inputs)	$V_{IN}=15V$			1.0	μA
$I_{IN(0)}$	Logical "0" Input Current (The Control Inputs)	$V_{IN}=0$	-1.0			μA
$I_{CC}$	Supply Current	$f_{CLK}=640 \text{ kHz}$		0.3	3.0	mA

### Electrical Characteristics (Continued)

Digital Levels and DC Specifications: ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV,  $4.75 < V_{CC} < 5.25V$ ,  $-40 < T_A < 85 < C$  unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>DATA OUTPUTS AND EOC (INTERRUPT)</b>						
$V_{OUT(1)}$	Logical "1" Output Voltage	$V_{CC} = 4.75V$ $I_{OUT} = -360\mu A$ $I_{OUT} = -10\mu A$		2.4 4.5		$V_{(min)}$ $V_{(min)}$
$V_{OUT(0)}$	Logical "0" Output Voltage	$I_O = 1.6 mA$			0.45	V
$V_{OUT(EOC)}$	Logical "0" Output Voltage EOC	$I_O = 1.2 mA$			0.45	V
$I_{OUT}$	TRI-STATE Output Current	$V_O = 5V$ $V_O = 0$	-3		3	$\mu A$ $\mu A$

### Electrical Characteristics

Timing Specifications  $V_{CC} = V_{REF(1)} = 5V$ ,  $V_{REF(1)} = GND$ ,  $t_r = t_f = 20 ns$  and  $T_A = 25 < C$  unless otherwise noted.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
$t_{WS}$	Minimum Start Pulse Width	(Figure 5)		100	200	ns
$t_{WALE}$	Minimum ALE Pulse Width	(Figure 5)		100	200	ns
$t_s$	Minimum Address Set-Up Time	(Figure 5)		25	50	ns
$t_H$	Minimum Address Hold Time	(Figure 5)		25	50	ns
$t_D$	Analog MUX Delay Time From ALE	$R_S = 0\Omega$ (Figure 5)		1	2.5	$\mu s$
$t_{H1}, t_{H0}$	OE Control to Q Logic State	$C_L = 50 pF$ , $R_L = 10k$ (Figure 8)		125	250	ns
$t_{H0}, t_{H1}$	OE Control to Hi-Z	$C_L = 10 pF$ , $R_L = 10k$ (Figure 8)		125	250	ns
$t_c$	Conversion Time	$f_c = 640 kHz$ , (Figure 5) (Note 7)		90	100	$\mu s$
$f_c$	Clock Frequency		10	640	1280	kHz
$t_{EOC}$	EOC Delay Time	(Figure 5)	0		$8 \pm 2 \mu s$	Clock Periods
$C_{IN}$	Input Capacitance	At Control Inputs		10	15	pF
$C_{OUT}$	TRI-STATE Output Capacitance	At TRI-STATE Outputs		10	15	pF

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to GND, unless otherwise specified.

Note 3: A zener diode exists internally from  $V_{CC}$  to GND and has a typical breakdown voltage of  $1 V_{CC}$ .

Note 4: Two on-chip diodes are tied to each analog input which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the  $V_{CC}$  supply. The spec allows 100 mV forward bias of either diode. This means that as long as the analog  $V_{IN}$  does not exceed the supply voltage by more than 100 mV, the output code will be correct. To achieve an absolute  $0V_{DC}$  to  $5V_{DC}$  input voltage range will therefore require a minimum supply voltage of  $4.900 V_{CC}$  over temperature variations, initial tolerance and loading.

Note 5: Total unadjusted error includes offset, full-scale, linearity, and multiplexer errors. See Figure 3. None of these A/Ds requires a zero or full scale adjust. However, if an all zero code is desired for an analog input other than 0.0V, or if a narrow full-scale span exists (for example, 0.5V to 4.5V full-scale) the reference voltages can be adjusted to achieve this. See Figure 1.

Note 6: Comparator input current is a bias current into or out of the chopper stabilized comparator. This bias current varies directly with clock frequency and has little temperature dependence (Figure 6). See paragraph 4.0.

Note 7: The outputs of the data register are updated one clock cycle before the rising edge of EOC.

Note 8: Human body model, 100 pF discharged through a 1.5 k $\Omega$  resistor.

## Functional Description

**Multiplexer.** The device contains an 8-channel single-ended analog signal multiplexer. A particular input channel is selected by using the address decoder. Table 1 shows the input states for the address lines to select any channel. The address is latched into the decoder on the low-to-high transition of the address latch enable signal.

TABLE 1.

SELECTED ANALOG CHANNEL	ADDRESS LINE		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

## CONVERTER CHARACTERISTICS

### The Converter

The heart of this single chip data acquisition system is its 8-bit analog-to-digital converter. The converter is designed to give fast, accurate, and repeatable conversions over a wide range of temperatures. The converter is partitioned into 3 major sections: the 256R ladder network, the successive approximation register, and the comparator. The converter's digital outputs are positive true.

The 256R ladder network approach (Figure 1) was chosen over the conventional R/2R ladder because of its inherent monotonicity, which guarantees no missing digital codes. Monotonicity is particularly important in closed loop feedback control systems. A non-monotonic relationship can cause oscillations that will be catastrophic for the system. Additionally, the 256R network does not cause load variations on the reference voltage.

The bottom resistor and the top resistor of the ladder network in Figure 1 are not the same value as the remainder of the network. The difference in these resistors causes the output characteristic to be symmetrical with the zero and full-scale points of the transfer curve. The first output transition occurs when the analog signal has reached  $+1/2$  LSB and succeeding output transitions occur every 1 LSB later up to full-scale.

The successive approximation register (SAR) performs 8 iterations to approximate the input voltage. For any SAR type converter,  $n$  iterations are required for an  $n$ -bit converter. Figure 2 shows a typical example of a 3-bit converter. In the ADC0808, ADC0809, the approximation technique is extended to 8 bits using the 256R network.

The A/D converter's successive approximation register (SAR) is reset on the positive edge of the start conversion (SC) pulse. The conversion is begun on the falling edge of the start conversion pulse. A conversion in process will be interrupted by receipt of a new start conversion pulse. Continuous conversion may be accomplished by tying the end-of-conversion (EOC) output to the SC input. If used in this mode, an external start conversion pulse should be applied after power up. End-of-conversion will go low between 0 and 8 clock pulses after the rising edge of start conversion.

The most important section of the A/D converter is the comparator. It is this section which is responsible for the ultimate accuracy of the entire converter. It is also the comparator drift which has the greatest influence on the repeatability of the device. A chopper-stabilized comparator provides the most effective method of satisfying all the converter requirements.

The chopper-stabilized comparator converts the DC input signal into an AC signal. This signal is then fed through a high gain AC amplifier and has the DC level restored. This technique limits the drift component of the amplifier since the drift is a DC component which is not passed by the AC amplifier. This makes the entire A/D converter extremely insensitive to temperature, long term drift and input offset errors.

Figure 4 shows a typical error curve for the ADC0808 as measured using the procedures outlined in AN-179.

Functional Description (Continued)

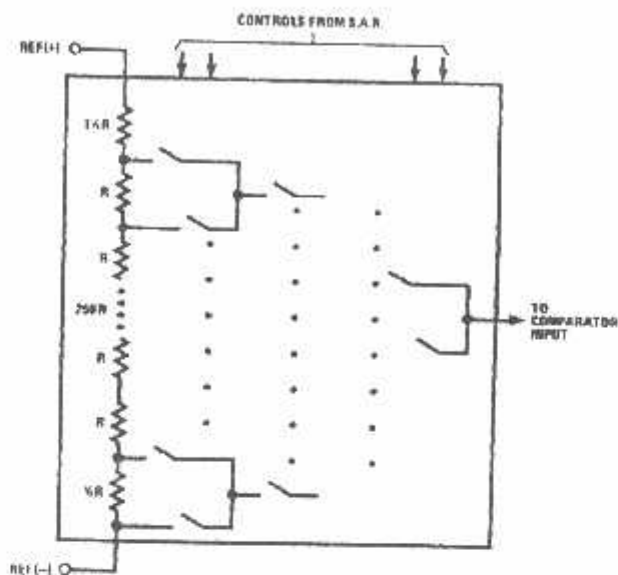


FIGURE 1. Resistor Ladder and Switch Tree

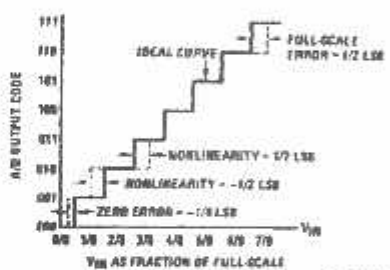


FIGURE 2. 3-Bit A/D Transfer Curve

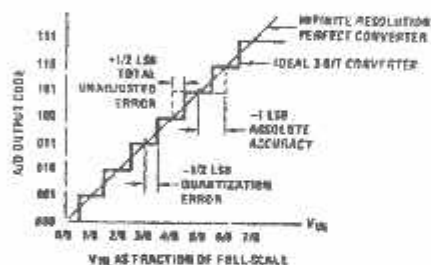


FIGURE 3. 3-Bit A/D Absolute Accuracy Curve



FIGURE 4. Typical Error Curve



Timing Diagram

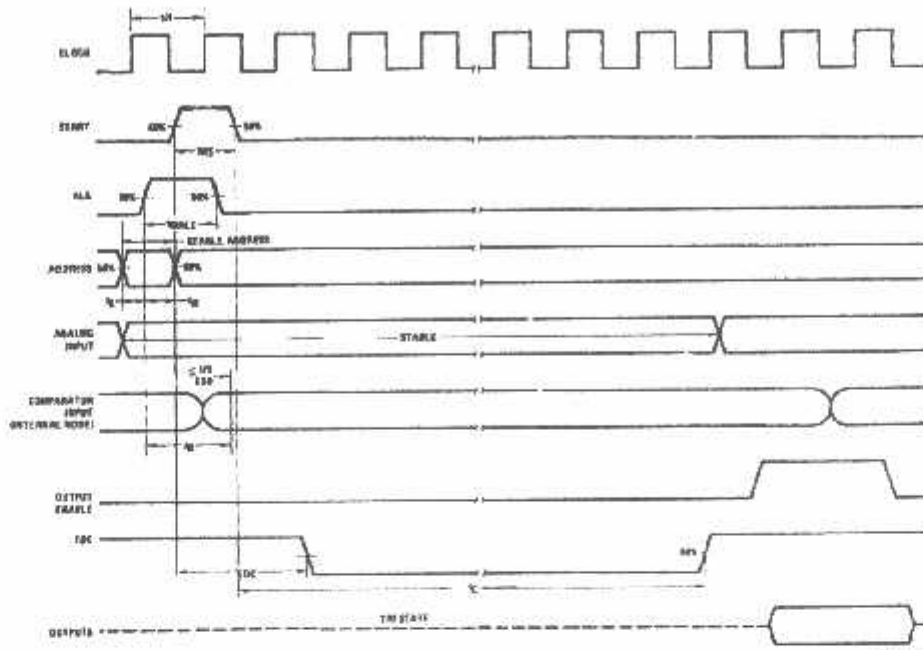


FIGURE 5.

20000004

Typical Performance Characteristics

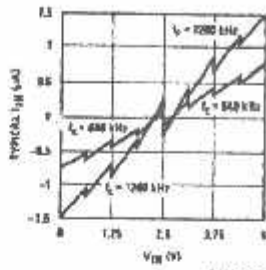


FIGURE 6. Comparator  $I_{IN}$  vs  $V_{IN}$  ( $V_{CC}=V_{REF}=5V$ )

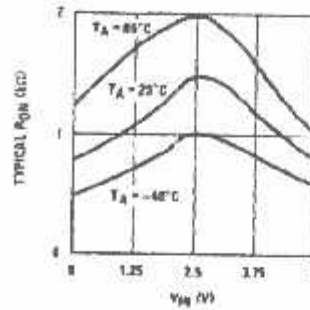


FIGURE 7. Multiplexer  $R_{ON}$  vs  $V_{IN}$  ( $V_{CC}=V_{REF}=5V$ )

TRI-STATE Test Circuits and Timing Diagrams

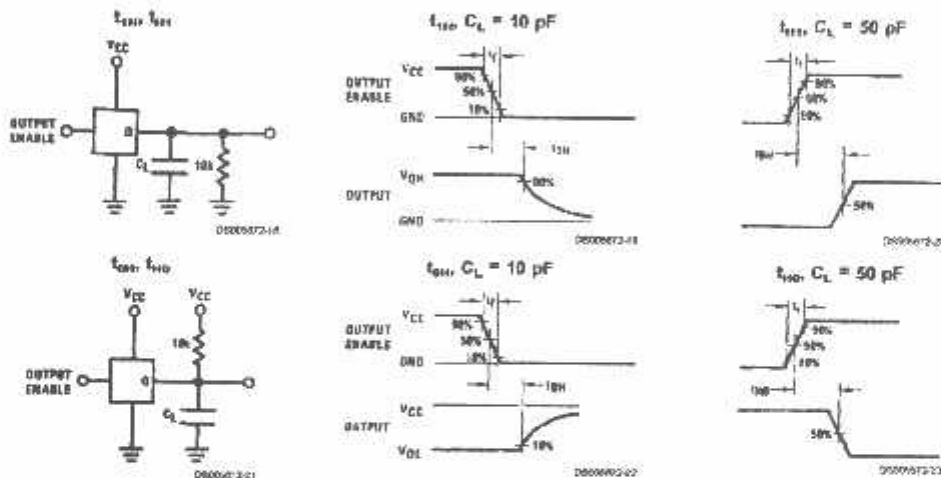


FIGURE 8.

Applications Information

OPERATION

1.0 RATIO-METRIC CONVERSION

The ADC0808, ADC0809 is designed as a complete Data Acquisition System (DAS) for ratio-metric conversion systems. In ratio-metric systems, the physical variable being measured is expressed as a percentage of full-scale which is not necessarily related to an absolute standard. The voltage input to the ADC0808 is expressed by the equation

$$\frac{V_{IN}}{V_{FS}-V_2} = \frac{D_X}{D_{MAX}-D_{MIN}} \quad (1)$$

$V_{IN}$  = Input voltage into the ADC0808  
 $V_{FS}$  = Full-scale voltage  
 $V_2$  = Zero voltage

$D_X$  = Data point being measured  
 $D_{MAX}$  = Maximum data limit  
 $D_{MIN}$  = Minimum data limit

A good example of a ratio-metric transducer is a potentiometer used as a position sensor. The position of the wiper is directly proportional to the output voltage which is a ratio of the full-scale voltage across it. Since the data is represented as a proportion of full-scale, reference requirements are greatly reduced, eliminating a large source of error and cost for many applications. A major advantage of the ADC0808, ADC0809 is that the input voltage range is equal to the supply range so the transducers can be connected directly across the supply and their outputs connected directly into the multiplexer inputs, (Figure 9).

Ratio-metric transducers such as potentiometers, strain gauges, thermistor bridges, pressure transducers, etc., are suitable for measuring proportional relationships; however, many types of measurements must be referred to an absolute standard such as voltage or current. This means a sys-

## Applications Information (Continued)

tern reference must be used which relates the full-scale voltage to the standard volt. For example, if  $V_{CC} = V_{REF} = 5.12V$ , then the full-scale range is divided into 256 standard steps. The smallest standard step is 1 LSB which is then 20 mV.

### 2.0 RESISTOR LADDER LIMITATIONS

The voltages from the resistor ladder are compared to the selected into 8 lines in a conversion. These voltages are coupled to the comparator via an analog switch tree which is referenced to the supply. The voltages at the top, center and bottom of the ladder must be controlled to maintain proper operation.

The top of the ladder, Ref(+), should not be more positive than the supply, and the bottom of the ladder, Ref(-), should not be more negative than ground. The center of the ladder voltage must also be near the center of the supply because the analog switch tree changes from N-channel switches to P-channel switches. These limitations are automatically satisfied in ratiometric systems and can be easily met in ground referenced systems.

Figure 10 shows a ground referenced system with a separate supply and reference. In this system, the supply must be trimmed to match the reference voltage. For instance, if a 5.12V is used, the supply should be adjusted to the same voltage within 0.1V.

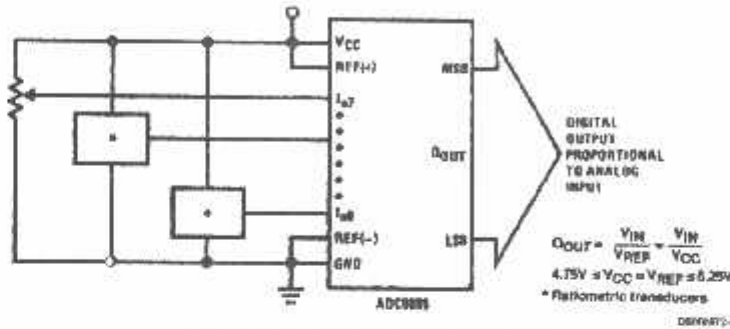
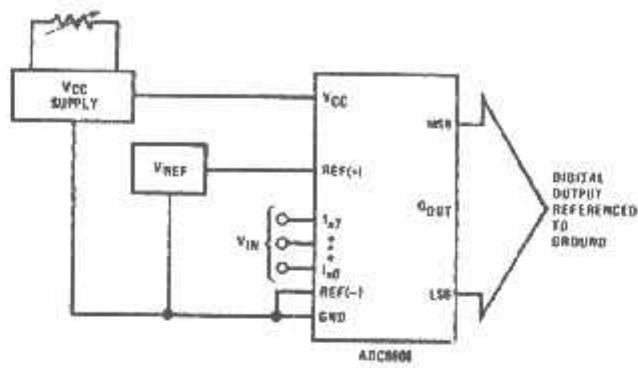


FIGURE 9. Ratiometric Conversion System

The ADC0808 needs less than a milliamp of supply current so developing the supply from the reference is readily accomplished. In Figure 11 a ground referenced system is shown which generates the supply from the reference. The buffer shown can be an op amp of sufficient drive to supply the milliamp of supply current and the desired bus drive, or if a capacitive bus is driven by the outputs a large capacitor will supply the transient supply current as seen in Figure 12. The LM301 is overcompensated to insure stability when loaded by the 10  $\mu F$  output capacitor.

The top and bottom ladder voltages cannot exceed  $V_{CC}$  and ground, respectively, but they can be symmetrically less than  $V_{CC}$  and greater than ground. The center of the ladder voltage should always be near the center of the supply. The sensitivity of the converter can be increased (i.e., size of the LSB steps decreased) by using a symmetrical reference system. In Figure 13, a 2.5V reference is symmetrically centered about  $V_{CC}/2$  since the same current flows in identical resistors. This system with a 2.5V reference allows the LSB bit to be half the size of a 5V reference system.

Applications Information (Continued)

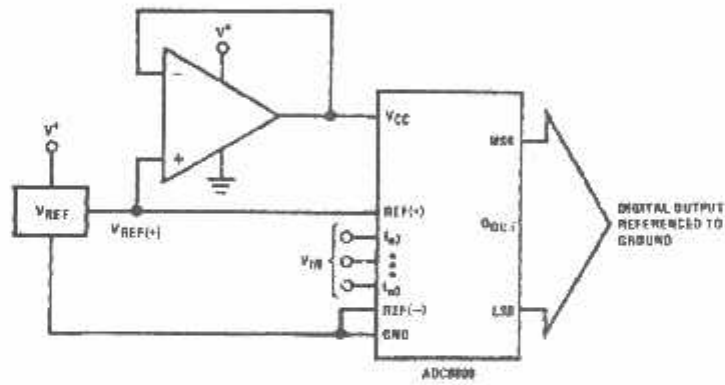


DS0808-124

$$Q_{OUT} = \frac{V_{IN}}{V_{REF}}$$

$$4.75V \leq V_{CC} = V_{REF} \leq 5.25V$$

FIGURE 10. Ground Referenced Conversion System Using Trimmed Supply



DS0809-325

$$Q_{OUT} = \frac{V_{IN}}{V_{REF}}$$

$$4.75V \leq V_{CC} = V_{REF} \leq 5.25V$$

FIGURE 11. Ground Referenced Conversion System with Reference Generating V<sub>CC</sub> Supply

Applications Information (Continued)

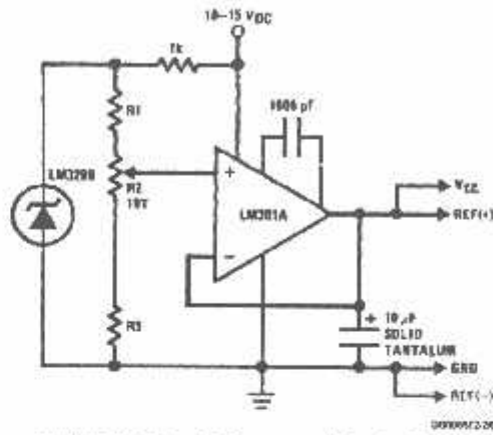
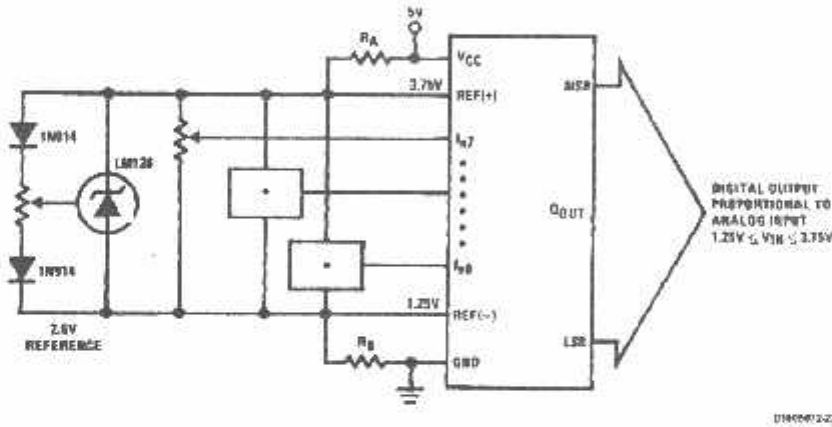


FIGURE 12. Typical Reference and Supply Circuit



D180002-01

$R_A = R_B$

\*Ratiometric transducers

FIGURE 13. Symmetrically Centered Reference

3.0 CONVERTER EQUATIONS

The transition between adjacent codes  $N$  and  $N+1$  is given by:

$$V_{IN} = \left( V_{REF(+)} - V_{REF(-)} \right) \left[ \frac{N}{256} \pm \frac{1}{512} \right] \pm V_{TUE} + V_{REF(-)} \quad (2)$$

The center of an output code  $N$  is given by:

$$V_{IN} = \left( V_{REF(+)} - V_{REF(-)} \right) \left[ \frac{N}{256} \right] \pm V_{TUE} + V_{REF(-)} \quad (3)$$

The output code  $N$  for an arbitrary input are the integers within the range:

$$N = \frac{V_{IN} - V_{REF(-)}}{V_{REF(+)} - V_{REF(-)}} \times 256 \pm \text{Absolute Accuracy} \quad (4)$$

Where:  $V_{IN}$  = Voltage at comparator input  
 $V_{REF(+)}$  = Voltage at Ref(+)  
 $V_{REF(-)}$  = Voltage at Ref(-)  
 $V_{TUE}$  = Total unadjusted error voltage (typically  $V_{REF(+)} \times 512$ )

## Applications Information (Continued)

### 4.0 ANALOG COMPARATOR INPUTS

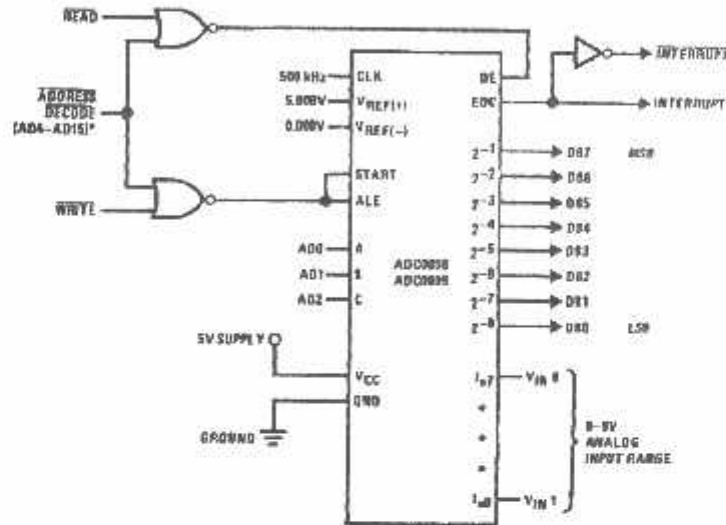
The dynamic comparator input current is caused by the periodic switching of on-chip stray capacitances. These are connected alternately to the output of the resistor ladder/switch tree network and to the comparator input as part of the operation of the chopper stabilized comparator.

The average value of the comparator input current varies directly with clock frequency and with  $V_{CC}$  as shown in Figure 6.

If no filter capacitors are used at the analog inputs and the signal source impedances are low, the comparator input current should not introduce converter errors, as the transient created by the capacitance discharge will die out before the comparator output is strobed.

If input filter capacitors are desired for noise reduction and signal conditioning they will tend to average out the dynamic comparator input current. It will then take on the characteristics of a DC bias current whose effect can be predicted conventionally.

### Typical Application



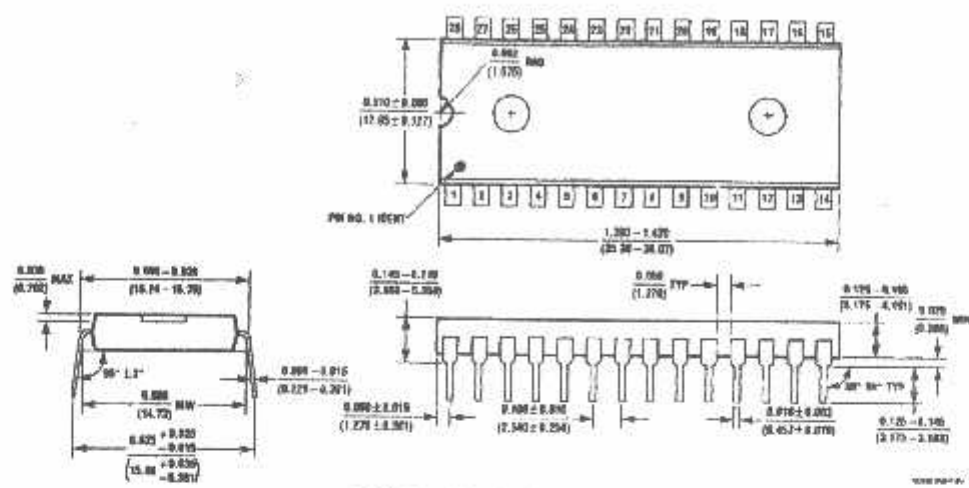
\*Address latches needed for 8085 and SCMP interfacing the ADC0808 to a microprocessor.

0500042-10

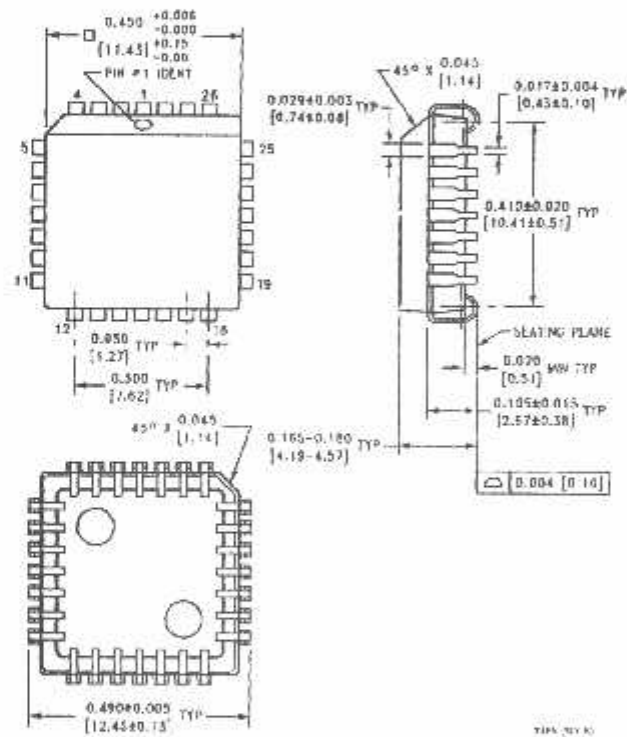
TABLE 2. Microprocessor Interface Table

PROCESSOR	READ	WRITE	INTERRUPT (COMMENT)
8080	MEMR	MEMW	INTR (Thru RST Circuit)
8085	$\overline{RD}$	$\overline{WR}$	INTR (Thru RST Circuit)
Z-80	$\overline{RD}$	$\overline{WR}$	$\overline{INT}$ (Thru RST Circuit, Mode 0)
SCMP	NRDS	NWDS	SA (Thru Sense A)
6800	VMA+ $\phi$ 2+RW	VMA+ $\phi$ + $\overline{RW}$	IRQA or IROB (Thru PIA)

**Physical Dimensions** inches (millimeters) unless otherwise noted.



**Molded Dual-In-Line Package (N)**  
 Order Number ADC0808CCN or ADC0809CCN  
 NS Package Number N28B



**Molded Chip Carrier (V)**  
 Order Number ADC0808CCV or ADC0809CCV  
 NS Package Number V28A

Notes

**LIFE SUPPORT POLICY**

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

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

 National Semiconductor Corporation  
Americas  
Tel: 1-800-272-0050  
Fax: 1-800-737-7018  
Email: [asupport@nsc.com](mailto:asupport@nsc.com)

[www.national.com](http://www.national.com)

National Semiconductor  
Europe  
Fic: +49 (0) 1 80-530 65 88  
Email: [europesupport@nsc.com](mailto:europesupport@nsc.com)  
Deutsch: Tel: +49 (0) 1 80-530 65 88  
English: Tel: +49 (0) 1 80-532 78 32  
Français: Tel: +49 (0) 1 80-532 93 55  
Italian: Tel: +49 (0) 1 80-534 16 86

National Semiconductor  
Asia Pacific Customer  
Response Group  
Tel: 65-2544495  
Fax: 65-2534466  
Email: [soa.support@nsc.com](mailto:soa.support@nsc.com)

National Semiconductor  
Japan Ltd  
Tel: 81-3-5639-7590  
Fax: 81-3-5639-7507





# ULN2001A-ULN2002A ULN2003A-ULN2004A

## SEVEN DARLINGTON ARRAYS

- SEVEN DARLINGTONS PER PACKAGE
- OUTPUT CURRENT 500mA PER DRIVER (800mA PEAK)
- OUTPUT VOLTAGE 50V
- INTEGRATED SUPPRESSION DIODES FOR INDUCTIVE LOADS
- OUTPUTS CAN BE PARALLELED FOR HIGHER CURRENT
- TTL/CMOS/PMOS/DTL COMPATIBLE INPUTS
- INPUTS PINNED OPPOSITE OUTPUTS TO SIMPLIFY LAYOUT

### DESCRIPTION

The ULN2001A, ULN2002A, ULN2003 and ULN2004A are high voltage, high current darlington arrays each containing seven open collector darlington pairs with common emitters. Each channel rated at 500mA and can withstand peak currents of 600mA. Suppression diodes are included for inductive load driving and the inputs are pinned opposite the outputs to simplify board layout.

The four versions interface to all common logic families :

ULN2001A	General Purpose, DTL, TTL, PMOS, CMOS
ULN2002A	14-25V PMOS
ULN2003A	5V TTL, CMOS
ULN2004A	6-15V CMOS, PMOS

These versatile devices are useful for driving a wide range of loads including solenoids, relays DC motors, LED displays filament lamps, thermal print-heads and high power buffers.

The ULN2001A/2002A/2003A and 2004A are supplied in 16 pin plastic DIP packages with a copper leadframe to reduce thermal resistance. They are available also in small outline package (SO-16) as ULN2001D/2002D/2003D/2004D.



DIP16

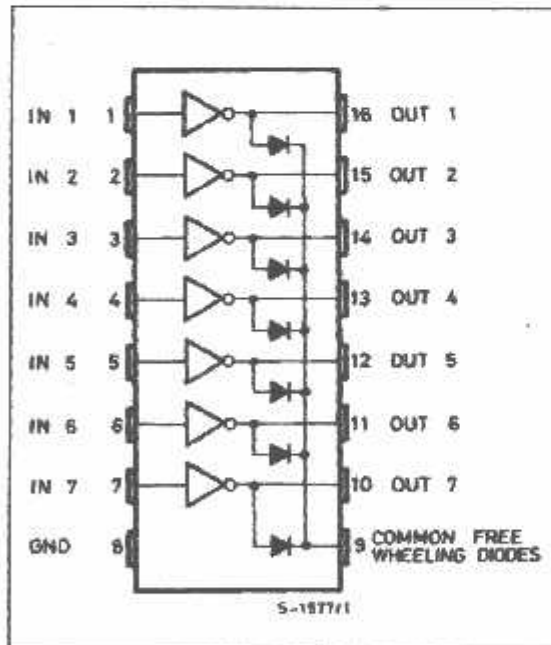
ORDERING NUMBERS: ULN2001A/2A/3A/4A



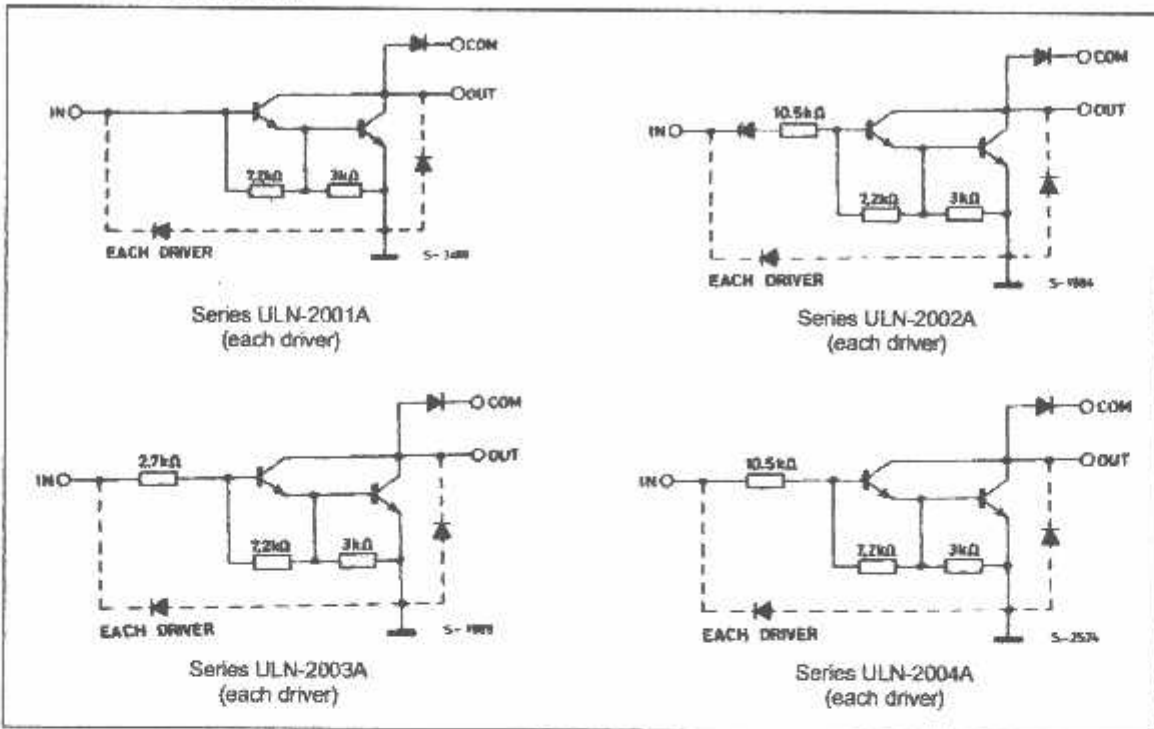
SO16

ORDERING NUMBERS: ULN2001D/2D/3D/4D

### PIN CONNECTION



**SCHEMATIC DIAGRAM**



**ABSOLUTE MAXIMUM RATINGS**

Symbol	Parameter	Value	Unit
$V_o$	Output Voltage	50	V
$V_{in}$	Input Voltage (for ULN2002A/D - 2003A/D - 2004A/D)	30	V
$I_c$	Continuous Collector Current	500	mA
$I_b$	Continuous Base Current	25	mA
$T_{amb}$	Operating Ambient Temperature Range	-20 to 85	°C
$T_{stg}$	Storage Temperature Range	-55 to 150	°C
$T_j$	Junction Temperature	150	°C

**THERMAL DATA**

Symbol	Parameter		DIP16	SO16	Unit
$R_{th(j-amb)}$	Thermal Resistance Junction-ambient	Max.	70	120	°C/W

**ELECTRICAL CHARACTERISTICS** ( $T_{amb} = 25^{\circ}\text{C}$  unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit	Fig.	
$I_{CEX}$	Output Leakage Current	$V_{CE} = 50\text{V}$ $T_{amb} = 70^{\circ}\text{C}$ , $V_{CE} = 50\text{V}$			50 100	$\mu\text{A}$ $\mu\text{A}$	1a 1a	
		$T_{amb} = 70^{\circ}\text{C}$ for ULN2002A $V_{CE} = 50\text{V}$ , $V_i = 6\text{V}$			500	$\mu\text{A}$	1b	
		for ULN2004A $V_{CE} = 50\text{V}$ , $V_i = 1\text{V}$			500	$\mu\text{A}$	1b	
$V_{CE(sat)}$	Collector-emitter Saturation Voltage	$I_C = 100\text{mA}$ , $I_B = 250\mu\text{A}$		0.9	1.1	V	2	
		$I_C = 200\text{mA}$ , $I_B = 350\mu\text{A}$		1.1	1.3	V	2	
		$I_C = 350\text{mA}$ , $I_B = 500\mu\text{A}$		1.3	1.6	V	2	
$I_{(on)}$	Input Current	for ULN2002A, $V_i = 17\text{V}$		0.82	1.25	mA	3	
		for ULN2003A, $V_i = 3.85\text{V}$		0.93	1.35	mA	3	
		for ULN2004A, $V_i = 5\text{V}$		0.35	0.5	mA	3	
		$V_i = 12\text{V}$		1	1.45	mA	3	
$I_{(off)}$	Input Current	$T_{amb} = 70^{\circ}\text{C}$ , $I_C = 500\mu\text{A}$	50	65		$\mu\text{A}$	4	
$V_{(on)}$	Input Voltage	$V_{CE} = 2\text{V}$ for ULN2002A $I_C = 300\text{mA}$			13		V	5
		for ULN2003A $I_C = 200\text{mA}$			2.4			
		$I_C = 250\text{mA}$			2.7			
		$I_C = 300\text{mA}$			3			
		for ULN2004A $I_C = 125\text{mA}$			5			
		$I_C = 200\text{mA}$			6			
		$I_C = 275\text{mA}$ $I_C = 350\text{mA}$			7 8			
$h_{FE}$	DC Forward Current Gain	for ULN2001A $V_{CE} = 2\text{V}$ , $I_C = 350\text{mA}$	1000					2
$C_i$	Input Capacitance			15	25	pF		
$t_{PLH}$	Turn-on Delay Time	$0.5 V_i$ to $0.5 V_o$		0.25	1	$\mu\text{s}$		
$t_{PHL}$	Turn-off Delay Time	$0.5 V_i$ to $0.5 V_o$		0.25	1	$\mu\text{s}$		
$I_R$	Clamp Diode Leakage Current	$V_R = 50\text{V}$ $T_{amb} = 70^{\circ}\text{C}$ , $V_R = 50\text{V}$			50 100	$\mu\text{A}$ $\mu\text{A}$	6 6	
$V_C$	Clamp Diode Forward Voltage	$I_F = 350\text{mA}$		1.7	2	V	7	

TEST CIRCUITS

Figure 1a.

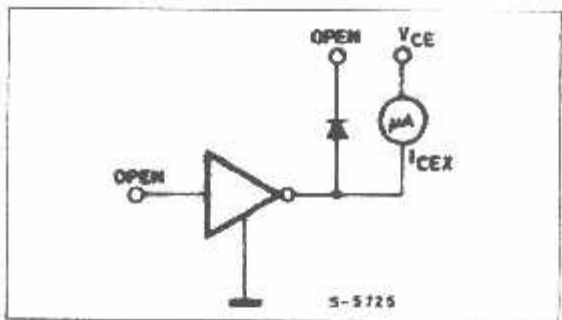


Figure 1b.

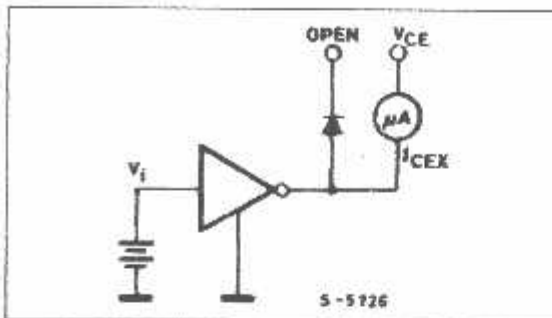


Figure 2.

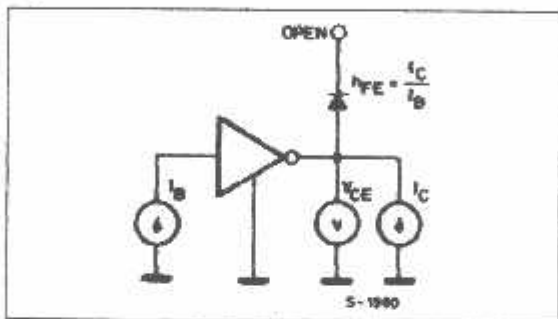


Figure 3.

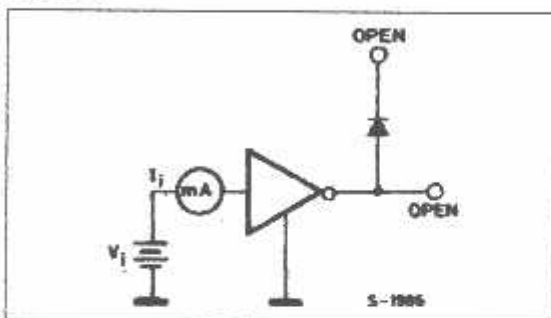


Figure 4.

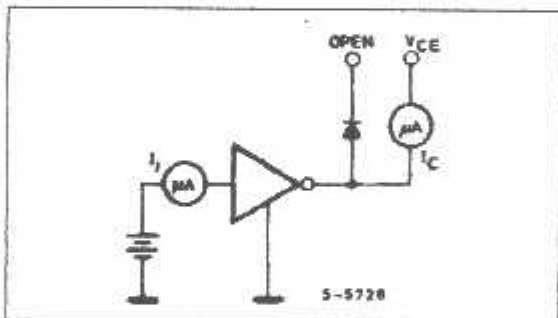


Figure 5.

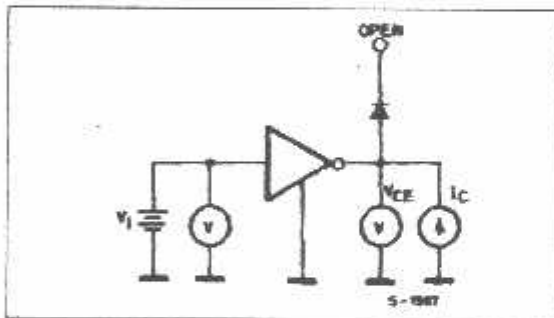


Figure 6.

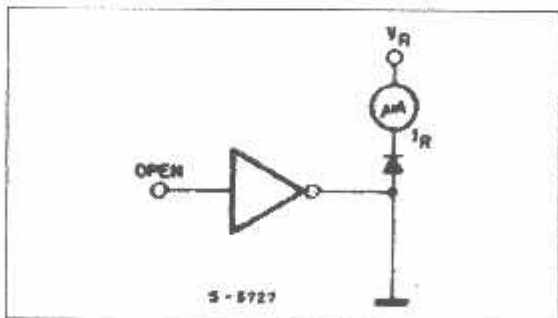


Figure 7.

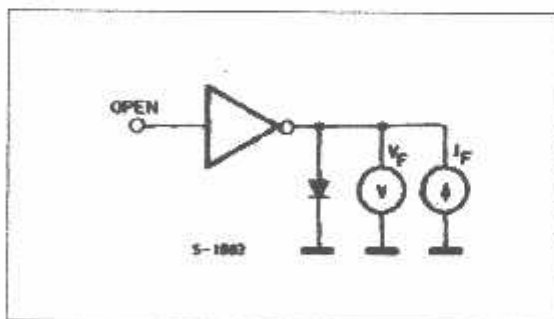


Figure 8: Collector Current versus Input Current

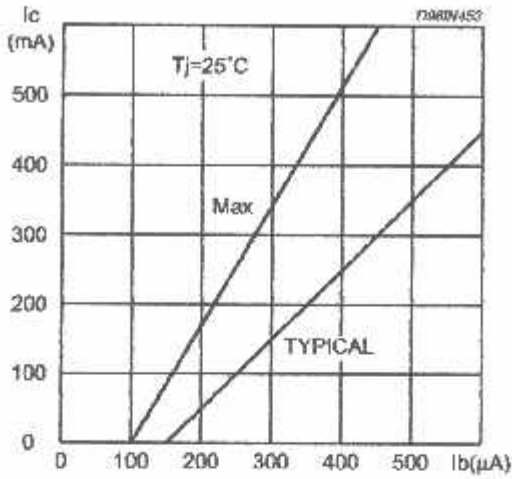


Figure 9: Collector Current versus Saturation Voltage

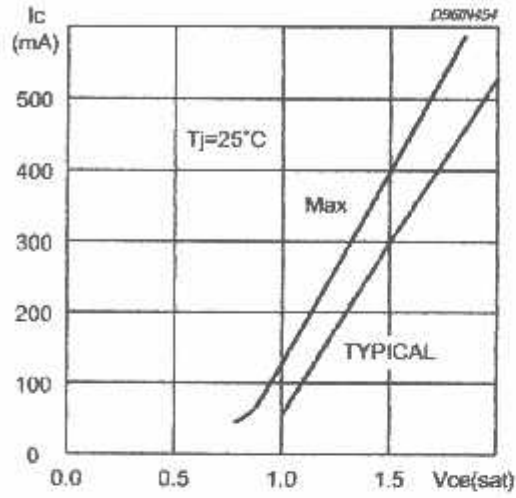


Figure 10: Peak Collector Current versus Duty Cycle

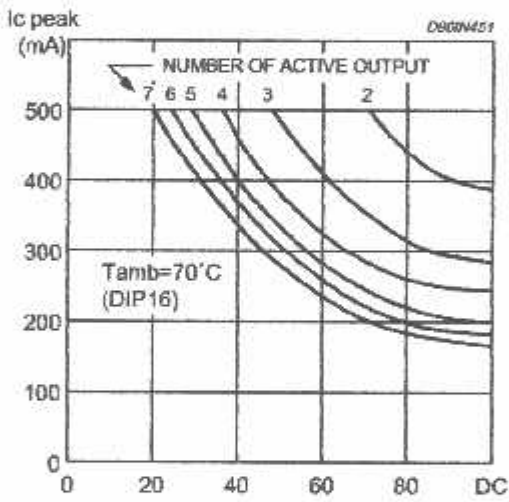
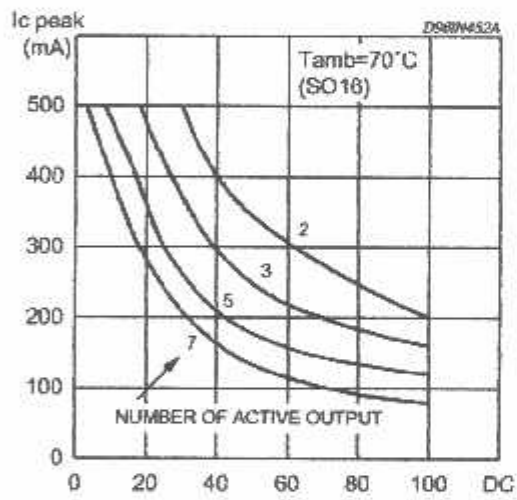
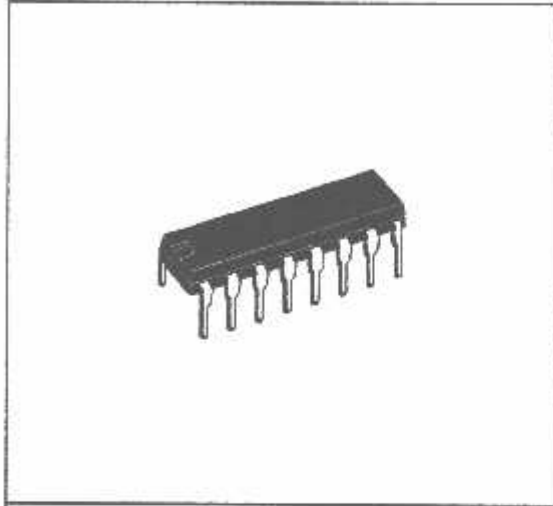


Figure 11: Peak Collector Current versus Duty Cycle

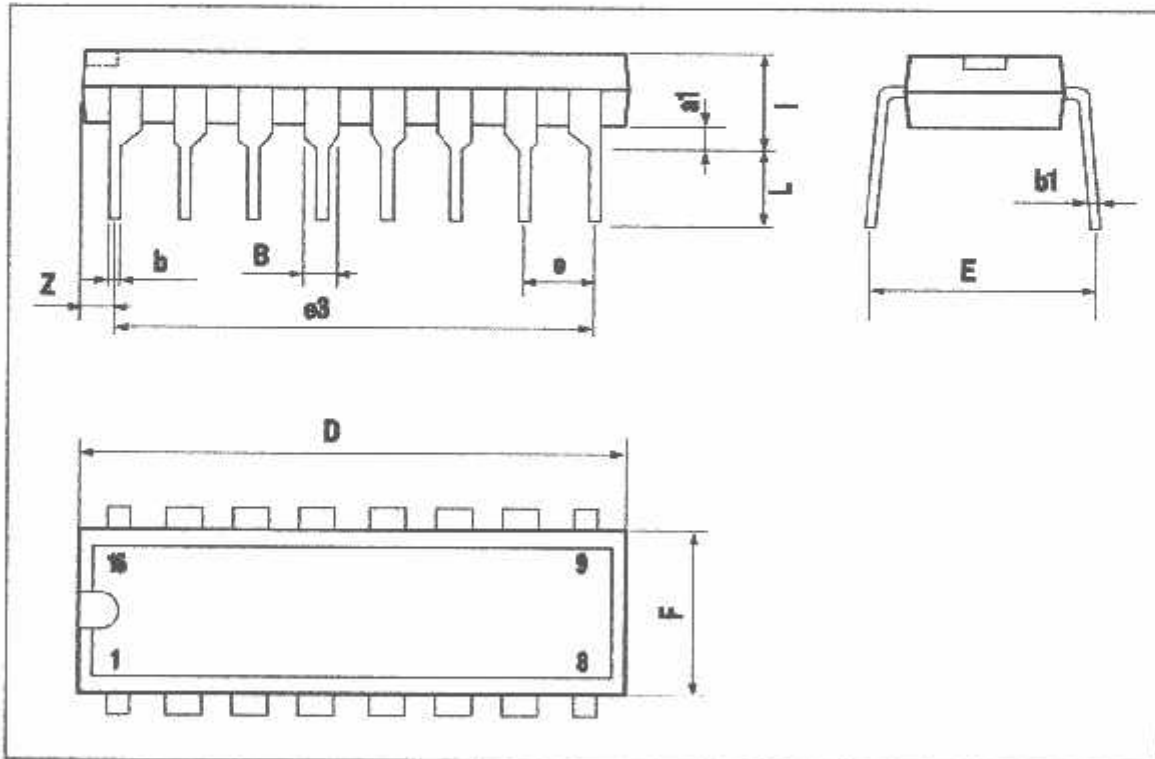


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
a1	0.51			0.020		
B	0.77		1.65	0.030		0.065
b		0.5			0.020	
b1		0.25			0.010	
D			20			0.787
E		8.5			0.335	
e		2.54			0.100	
e3		17.78			0.700	
F			7.1			0.280
I			5.1			0.201
L		3.3			0.130	
Z			1.27			0.050

OUTLINE AND MECHANICAL DATA



DIP16



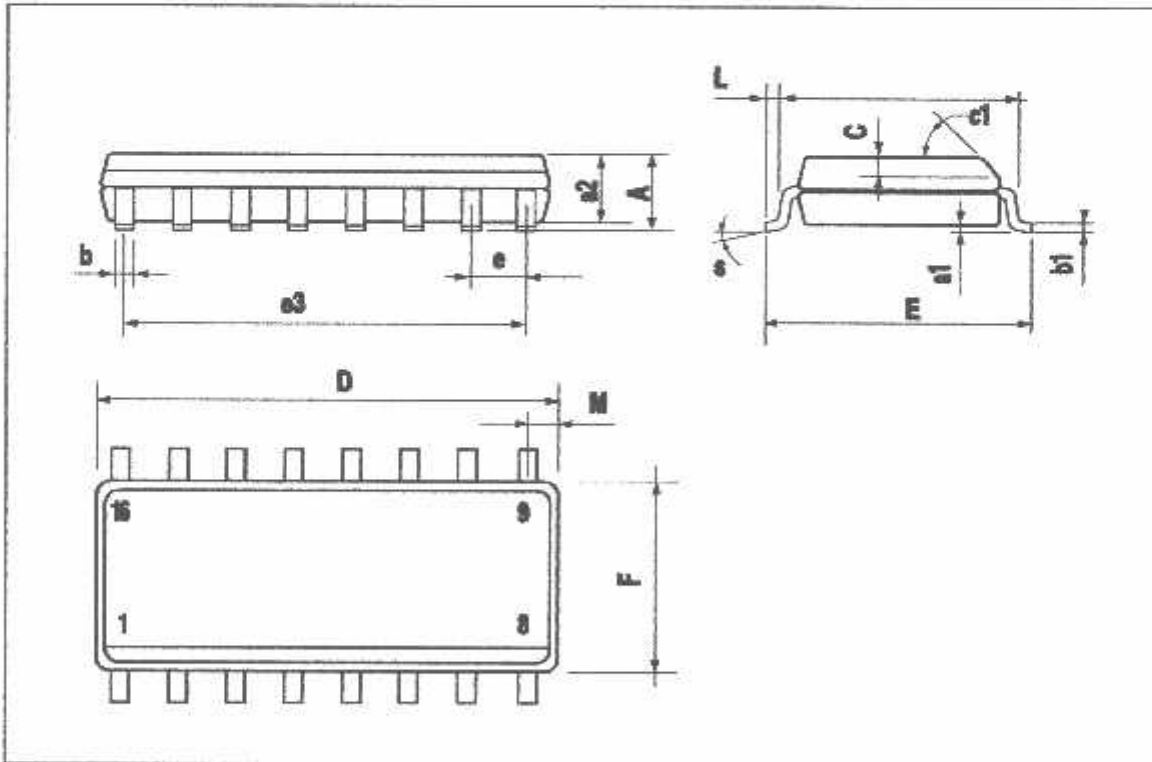
DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			1.75			0.069
a1	0.1		0.25	0.004		0.009
a2			1.6			0.063
b	0.35		0.48	0.014		0.018
b1	0.19		0.25	0.007		0.010
C		0.5			0.020	
c1	45° (typ.)					
D (1)	9.8		10	0.386		0.394
E	5.8		6.2	0.228		0.244
e		1.27			0.050	
e3		8.89			0.350	
F (1)	3.8		4	0.150		0.157
G	4.6		5.3	0.181		0.209
L	0.4		1.27	0.016		0.050
M			0.82			0.024
S	8° (max.)					

OUTLINE AND MECHANICAL DATA



SO16 Narrow

(1) D and F do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15mm (.006inch).



Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specification mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics  
© 2002 STMicroelectronics - Printed in Italy - All Rights Reserved  
STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco -  
Singapore - Spain - Sweden - Switzerland - United Kingdom - United States.  
<http://www.st.com>