

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



**PERENCANAAN DAN PEMBUATAN ALAT PENGAMAN
MOTOR INDUKSI 3 FASA TERHADAP GANGGUAN ARUS
DAN TEGANGAN BERBASIS MIKROKONTROLLER
AT89S8252**

SKRIPSI

Disusun oleh :
HERMAN SUSILO
0312125/P

SEPTEMBER 2006

LEMBAR PERSETUJUAN



PERENCANAAN DAN PEMBUATAN ALAT PENGAMAN MOTOR INDUKSI 3 FASA TERHADAP GANGGUAN ARUS DAN TEGANGAN BERBASIS MIKROKONTROLLER AT89S8252

SKRIPSI

*Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat
Guna Mencapai Gelar Sarjana Teknik Elektro Strata Satu (S-1)*

Disusun oleh :

HERMAN SUSILO
0312125/P

Mengetahui,
Ketua Jurusan Teknik Elektro S-1

Diperiksa dan Disetujui,
Dosen Pembimbing



Ir. Widodo Pudji Muljanto, MT
NIP.Y.1028700171

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

ABSTRAKSI

PERENCANAAN DAN PEMBUATAN ALAT PENGAMAN MOTOR INDUKSI 3 FASA TERHADAP GANGGUAN ARUS DAN TEGANGAN BERBASIS MIKROKONTROLLER AT89S8252

(Herman Susilo, 0312125/P, T. Elektro Energi Listrik S-1)
(Dosen Pembimbing : Ir. Widodo Pudji Muljanto, MT)

Kata kunci : Motor Induksi 3 Fasa, Gangguan Arus dan Tegangan, Mikrokontroller AT89S8252

Motor Induksi 3 Fasa banyak digunakan terutama pada industri berskala menengah ke atas. Pada penggunaannya sehari-hari sering dijumpai adanya gangguan-gangguan misalnya gangguan arus lebih, tegangan kurang, tegangan lebih atau tegangan putus satu fasa, yang dapat mengakibatkan kerusakan pada motor induksi. Untuk itu diperlukan suatu alat pengaman motor induksi 3 fasa terhadap gangguan-gangguan tersebut. Dalam hal ini dipilih yang bersifat elektronik digital, yaitu menggunakan mikrokontroller AT89S8252.

Dengan digunakannya alat pengaman tersebut, diharapkan keandalan sistem instalasi motor-motor listrik dapat lebih terjamin dan motor induksi terhindar dari kerusakan.

Penggunaan mikrokontroller AT89S8252 didukung oleh beberapa komponen yaitu sensor arus, sensor tegangan, ADC, keypad , LCD dan rangkaian driver kontaktor. Arus yang mengalir pada motor disensor oleh sensor arus, sedangkan besarnya tegangan yang masuk ke motor disensor oleh sensor tegangan. Setting arus maksimal, tegangan maksimal, tegangan minimal dan waktu tunda (*delay time*) dimasukkan melalui keypad. Dari hasil monitoring arus dan tegangan yang ditampilkan lewat LCD, apabila terjadi gangguan atau kondisi yang menyimpang dari setting yang telah ditetapkan sebelumnya, maka mikrokontroller memberi sinyal ke rangkaian driver kontaktor untuk meng-off-kan kontaktor sehingga motor berhenti beroperasi. Sehingga motor aman dari gangguan.

Pengujian dan analisis dilakukan untuk mengetahui unjuk kerja dari masing-masing komponen dalam merespon gangguan yang terjadi yakni gangguan arus lebih, gangguan tegangan lebih, gangguan tegangan kurang dan tegangan hilang satu fasa.

KATA PENGANTAR

Segala puji syukur penyusun panjatkan kehadirat Allah SWT, karena atas Pertolongan dan Ridho-Nyalah, penyusun dapat menyelesaikan Skripsi ini dengan judul :

PERENCANAAN DAN PEMBUATAN ALAT PENGAMAN MOTOR INDUKSI 3 FASA TERHADAP GANGGUAN ARUS DAN TEGANGAN BERBASIS MIKROKONTROLLER AT89S8252

Skripsi ini merupakan syarat untuk menyelesaikan program studi Strata satu (S-1) Jurusan Teknik Elektro S-1, Konsentrasi Teknik Energi Listrik, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Selama penyusunan Skripsi ini banyak masukan dan bantuan yang penyusun peroleh, sehingga pada kesempatan kali ini dengan segala kerendahan hati dan penuh rasa hormat penyusun menyampaikan terima kasih kepada yang terhormat :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku Rektor ITN Malang.
2. Bapak Ir. Mochtar Asroni, MSME selaku Dekan Fakultas Teknologi Industri ITN Malang.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Bapak Ir. Widodo Pudji Muljanto, MT selaku Dosen Pembimbing.
5. Segenap Dosen Teknik Elektro dan Karyawan ITN Malang.

6. Semua pihak yang tidak bisa disebutkan satu per satu yang telah memberikan dukungan hingga terselesaiannya penyusunan skripsi ini.

Selanjutnya penyusun sadar bahwa skripsi ini mungkin masih perlu disempurnakan, maka berbagai masukan, saran dan kritik yang sifatnya membangun sangat diharapkan untuk penyusunan skripsi berikutnya.

Akhirnya semoga skripsi ini bermanfaat bagi kita semua dan cukup membantu para pembaca, khususnya rekan-rekan mahasiswa.

Malang, September 2006

Penulis

Herman Susilo

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
DAFTAR GRAFIK	xiii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	1
1.3. Tujuan	2
1.4. Batasan Masalah.....	2
1.5. Metodologi	3
1.6. Sistematika Penulisan	4
BAB II LANDASAN TEORI.....	5
2.1. Motor Induksi Tiga Fasa	5
2.1.1. Bagian-bagian Motor Induksi	5
2.1.2. Prinsip Kerja Motor Induksi Tiga Fasa.....	6
2.2. Transformator.....	8
2.2.1. Definisi Transformator.....	8
2.2.2. Prinsip Kerja Transformator.....	8
2.2.3. Transformator Arus	9
2.3. Pengubah Analog ke Digital (ADC)	10
2.3.1. Teori Dasar.....	10
2.3.2. ADC 0809	11
2.4. Mikrokontroller AT89S8252	13
2.4.1. Fitur Mikrokontroller AT89S8252.....	13
2.4.2. Arsitektur AT89S8252	15

2.4.3. Konfigurasi Kaki-kaki MCU AT89S8252	16
2.4.4. Organisasi Memori AT89S8252	19
2.4.4.1. Memori Program	20
2.4.4.2. Memori Data	21
2.4.5. SFR Tambahan pada AT89S8252.....	25
2.4.5.1. SFR Untuk Timer 2.....	25
2.4.5.2. SFR Untuk Watchdog dan Memori.....	25
2.4.5.3. SFR Pengontrol SPI	26
2.4.6. Osilator.....	27
2.4.7. Timer dan Counter	28
2.4.8. Sistem Interrupt.....	30
2.4.9. Metode Pengalamatan	31
2.5. Bahasa Assembler MCS-51	32
2.6. LCD	33
2.7. <i>Keypad</i> (Papan Tombol).....	36
2.8. Transistor Sebagai Saklar.....	37
2.9. Konstanta Waktu RC.....	39
BAB III PERENCANAAN DAN PEMBUATAN ALAT	40
3.1. Blok Diagram Rangkaian.....	40
3.1.1. Fungsi Masing-masing Bagian.....	41
3.1.2. Prinsip Kerja.....	42
3.2. Perencanaan <i>Hardware</i> (Perangkat Keras)	43
3.2.1. Perencanaan Sensor Tegangan.....	43
3.2.2. Perencanaan Sensor Arus	45
3.2.3. ADC 0809	48
3.2.4. Keypad	50
3.2.5. LCD (<i>Liquid Crystal Display</i>).....	51
3.2.6. Mikrokontroller AT89S8252	53
3.2.7. Driver Kontaktor	55
3.3. Perencanaan <i>Software</i> (Perangkat Lunak).....	56

BAB IV PENGUJIAN ALAT DAN ANALISIS	58
4.1. Umum	58
4.2. Pengujian Sensor Arus dan ADC.....	58
4.2.1. Tujuan	58
4.2.2. Peralatan yang Digunakan.....	58
4.2.3. Prosedur Pengujian.....	59
4.2.4. Hasil Pengujian	60
4.2.5. Analisis Hasil Pengujian	60
4.3. Pengujian Sensor Tegangan dan ADC.....	63
4.3.1. Tujuan	63
4.3.2. Peralatan yang Digunakan.....	63
4.3.3. Prosedur Pengujian.....	63
4.3.4. Hasil Pengujian Sensor Tegangan dan ADC.....	65
4.3.5. Analisis Hasil Pengujian	65
4.4. Pengujian Mikrokontroller, LCD dan Keypad.....	68
4.4.1. Tujuan	68
4.4.2. Peralatan yang Digunakan.....	68
4.4.3. Prosedur Pengujian.....	68
4.4.4. Hasil Pengujian	68
4.4.5. Analisis Hasil Pengujian	69
4.5. Pengujian Rangkaian Driver Kontaktor	69
4.5.1. Tujuan	69
4.5.2. Peralatan yang Digunakan.....	69
4.5.3. Prosedur Pengujian.....	69
4.5.4. Hasil Pengujian	70
4.5.5. Analisis Hasil Pengujian	70
4.6. Pengujian Ketelitian Alat dalam Pengukuran Arus.....	71
4.6.1. Tujuan	71
4.6.2. Alat yang Digunakan.....	71
4.6.3. Prosedur Pengujian.....	71
4.6.4. Hasil Pengujian	72

4.6.5. Analisis Hasil Pengujian	72
4.7. Pengujian Ketelitian Alat dalam Pengukuran Tegangan.....	73
4.7.1. Tujuan	73
4.7.2. Alat yang Digunakan	73
4.7.3. Prosedur Pengujian.....	73
4.7.4. Hasil Pengujian	74
4.7.5. Analisis Hasil Pengujian	74
4.8. Pengujian Alat Terhadap Gangguan Arus Lebih	75
4.8.1. Tujuan	75
4.8.2. Alat yang Digunakan.....	75
4.8.3. Prosedur Pengujian.....	75
4.8.4. Hasil Pengujian	77
4.8.5. Analisis Hasil Pengujian	77
4.9. Pengujian Alat Terhadap Gangguan Tegangan Lebih	78
4.9.1. Tujuan	78
4.9.2. Alat yang Digunakan.....	78
4.9.3. Prosedur Pengujian.....	78
4.9.4. Hasil Pengujian	80
4.9.5. Analisis Hasil Pengujian	80
4.10. Pengujian Alat Terhadap Gangguan Tegangan Kurang.....	80
4.10.1. Tujuan	80
4.10.2. Alat yang Digunakan.....	80
4.10.3. Prosedur Pengujian.....	81
4.10.4. Hasil Pengujian	82
4.10.5. Analisis Hasil Pengujian	83
BAB V PENUTUP	84
5.1. Kesimpulan	84
5.2. Saran.....	84

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

Tabel 2.1	Fungsi Alternatif Port 1.....	17
Tabel 2.2	Pembagian Alamat pada SFR.....	24
Tabel 2.3	Mode Operasi Timer/Counter 0 dan 1.....	28
Tabel 2.4	Mode Operasi Timer 2	29
Tabel 4.1	Hasil Pengujian Rangkaian Sensor Arus dan ADC	60
Tabel 4.2	Hasil Pengujian Rangkaian Sensor Tegangan dan ADC	65
Tabel 4.3	Hasil Pengujian Mikrokontroller, LCD dan Keypad	68
Tabel 4.4	Hasil Pengujian Rangkaian Driver Kontaktor.....	70
Tabel 4.5	Hasil Pengujian Ketelitian Alat dalam Pengukuran Arus	72
Tabel 4.6	Hasil Pengujian Ketelitian Alat dalam Pengukuran Tegangan	74
Tabel 4.7	Hasil Pengujian Alat Terhadap Gangguan Arus Lebih.....	77
Tabel 4.8	Hasil Pengujian Alat Terhadap Gangguan Tegangan Lebih.....	80
Tabel 4.9	Hasil Pengujian Alat Terhadap Gangguan Tegangan Kurang	82

DAFTAR GAMBAR

Gambar 2.1	Bagian-bagian Motor Induksi Tiga Fasa.....	6
Gambar 2.2	Skema Prinsip Kerja Trafo.....	7
Gambar 2.3	Transformator Arus	8
Gambar 2.4	ADC 0809	12
Gambar 2.5	Blok Diagram AT89S8252	15
Gambar 2.6	Bentuk Fisik AT89S8252.....	16
Gambar 2.7	SPI Master-slave Interconnection	26
Gambar 2.8	Karakteristik Osilator	27
Gambar 2.9	Konfigurasi Kaki LCD	35
Gambar 2.10	<i>Keypad 4 x 3</i>	36
Gambar 2.11	Rangkaian Transistor sebagai Saklar	37
Gambar 2.12	Garis Beban DC	37
Gambar 3.1	Diagram Blok Rangkaian	40
Gambar 3.2	Rangkaian Sensor Tegangan	44
Gambar 3.3	Trafo Arus	46
Gambar 3.4	Rangkaian Sensor Arus	46
Gambar 3.5	ADC 0809	49
Gambar 3.6	Pembangkit Clock	50
Gambar 3.7	<i>Keypad 4 x 3</i>	51
Gambar 3.8	Rangkaian LCD	52
Gambar 3.9	Tampilan Layar LCD	53
Gambar 3.10	Rangkaian Mikrokontroller.....	55
Gambar 3.11	Rangkaian Driver Kontaktor	55
Gambar 3.12	<i>Flow Chart</i> Perancangan Perangkat Lunak.....	57
Gambar 4.1	Diagram Blok Pengujian Rangkaian Sensor Arus dan ADC	59
Gambar 4.2	Diagram Blok Pengujian Rangkaian Sensor Tegangan dan ADC64	
Gambar 4.3	Diagram Blok Pengujian Mikrokontroller, LCD dan <i>Keypad</i>	68

Gambar 4.4	Rangkaian Pengujian Driver Kontaktor	70
Gambar 4.5	Rangkaian Pengujian Ketelitian Pengukuran Arus	71
Gambar 4.6	Rangkaian Pengujian Ketelitian Pengukuran Tegangan	73
Gambar 4.7	Blok Diagram Pengujian Alat Pengaman Motor Terhadap Gangguan Arus Lebih	76
Gambar 4.8	Blok Diagram Pengujian Alat Pengaman Motor Terhadap Gangguan Tegangan Lebih	79
Gambar 4.9	Blok Diagram Pengujian Alat Pengaman Motor Terhadap Gangguan Tegangan Kurang.....	82

DAFTAR GRAFIK

Grafik	4.1	Hubungan Antara Arus Fasa dan Tegangan pada Sensor Arus ..	61
Grafik	4.2	Hubungan Antara Tegangan Sensor Arus dan Konversi ADC ...	62
Grafik	4.3	Hubungan Antara Tegangan Fasa dan Tegangan pada Sensor Tegangan	66
Grafik	4.4	Hubungan Antara Tegangan pada Sensor Tegangan dan Konversi ADC	67

BAB I

PENDAHULUAN

1.1. Latar Belakang

Motor Induksi 3 fasa merupakan motor arus bolak-balik yang banyak digunakan terutama pada industri berskala menengah ke atas. Hal ini karena konstruksinya yang sederhana dan harganya yang relatif lebih murah dibanding motor jenis lainnya.

Pada pengoperasian motor induksi 3 fasa sehari-hari, sering dijumpai adanya gangguan-gangguan terutama yang disebabkan oleh arus dan tegangan. Diantaranya yaitu gangguan arus lebih, tegangan turun/naik dan tegangan putus satu fasa.

Untuk mencegah kerusakan pada motor yang disebabkan oleh gangguan-gangguan tersebut maka mutlak diperlukan alat yang dapat berfungsi sebagai pengaman. Dalam hal ini dipilih yang bersifat elektronik digital dengan menggunakan mikrokontroller mengingat kelebihan-kelebihan yang dimilikinya yaitu lebih akurat dan lebih protektif dibanding yang bersifat magnetis dan termis.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, timbul masalah yang perlu dikaji penyelesaiannya yaitu :

1. Bagaimana merencanakan dan membuat rangkaian perangkat kerasnya (*hardware*) yang terdiri atas :

- Rangkaian sensor tegangan dan sensor arus.
 - Rangkaian pengkondisi sinyal.
 - ADC, LCD, keypad, mikrokontroller, rangkaian driver kontaktor.
2. Bagaimana merencanakan dan membuat rangkaian perangkat lunaknya (*software*).

1.3. Tujuan

Tujuan dari penulisan skripsi ini adalah :

1. Merencanakan dan membuat alat pengaman motor induksi 3 fasa terhadap gangguan arus dan tegangan dengan menggunakan mikrokontroller AT89S8252.
2. Meningkatkan keandalan sistem instalasi motor-motor listrik.

1.4. Batasan Masalah

Agar permasalahan tidak berkembang meluas maka pembahasan pada skripsi ini dibatasi pada hal-hal sebagai berikut :

1. Tidak membahas catu daya dan catu daya dianggap konstan.
2. Mikrokontroller yang digunakan AT89S8252.
3. Tidak membahas perubahan frekuensi, dan frekuensi dianggap 50 Hz.
4. Hanya membahas pengaman motor induksi 3 fasa, 380 V, 1 – 1,5 HP, 50 Hz terhadap gangguan arus lebih, tegangan naik/turun dan supplai tegangan putus satu fasa.

5. Tidak membahas kerusakan motor yang diakibatkan oleh gangguan-gangguan tersebut di atas.
6. Tidak membahas perubahan karakteristik motor pada saat terjadi gangguan tersebut di atas.

1.5. Metodologi

Skripsi ini lebih bersifat aplikatif yaitu perencanaan dan pembuatan alat.

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat yaitu :

1. Studi Literatur.

Yaitu penelitian yang dilakukan dengan jalan mengumpulkan data dan membaca buku-buku acuan yang ada hubungannya dengan masalah yang dibahas dalam skripsi ini.

2. Tahap Perencanaan.

- Perencanaan tiap-tiap blok diagram dengan komponen yang sesuai.
- Perencanaan sistem secara keseluruhan.

3. Tahap Pembuatan.

- Penggabungan tiap-tiap blok menjadi satu kesatuan rangkaian.
- Pembuatan perangkat lunak pada mikrokontroller AT89S8252.

4. Tahap Pengujian Alat.

Pengujian dilakukan intuk mengetahui unjuk kerja alat dengan menggunakan simulasi pada sistem tegangan, arus dan beban.

5. Menarik kesimpulan

1.6. Sistimatika Penulisan

BAB I Pendahuluan

Menguraikan secara singkat tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi dan sistimatika penulisan.

BAB II Dasar Teori

Menjelaskan secara garis besar teori tentang motor induksi 3 fasa, transformator, mikrokontroller AT89S8, ADC, keypad, LCD, transistor sebagai saklar dan konstanta RC.

BAB III Perencanaan dan Pembuatan Alat

Menguraikan tentang perencanaan serta proses pembuatan perangkat keras dan perangkat lunak.

BAB IV Pengujian Alat

Menguraikan tentang proses pengujian alat untuk mengetahui unjuk kerja alat yang dibuat.

BAB V Penutup

Menjelaskan tentang kesimpulan yang didapat dari keseluruhan skripsi ini dan saran-saran untuk penyempurnaan serta pengembangan.

BAB II

LANDASAN TEORI

2.1. Motor Induksi Tiga Fasa

2.1.1. Bagian-bagian Motor Induksi

a. Stator

Stator adalah bagian motor yang tidak bergerak (*statis*). Stator dibuat dengan kern inti besi yang berbentuk alur-alur sebagai tempat kawat-kawat kumparan. Kumparan stator berupa lilitan-lilitan dari kawat email yang memiliki kemampuan hantar arus (KHA) yang tergantung pada besarnya luas penampang kawat email yang digunakan. Semakin besar luas penampangnya, makin besar pula kemampuan hantar arusnya, demikian pula sebaliknya.

Apabila besarnya arus yang melewati kumparan stator melebihi KHA dari kawat email stator dan melebihi batas selang waktu yang diijinkan, hal ini akan mengakibatkan kerusakan/terbakarnya kumparan stator.

b. Rotor

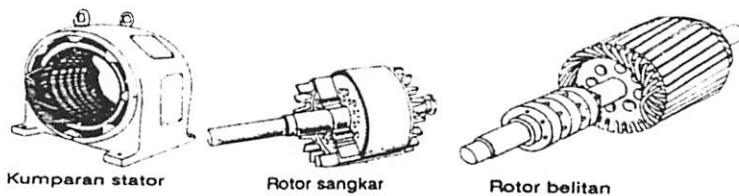
Rotor adalah bagian yang berputar dari motor. Adapun rotor motor induksi ada 2 macam :

1) Rotor sangkar tupai (*Squirrel Cage Rotor*)

Rotornya terdiri dari batang penghantar pejal yang disusun sedemikian rupa hingga menyerupai sangkar tupai . Konstruksi rotor ini paling sederhana dibandingkan dengan motor listrik lainnya.

2) Rotor Lilit (*Wound Rotor*)

Motor yang memakai rotor tipe ini disebut motor induksi rotor lilit atau motor induksi cicin geser (*slip ring*). Rotornya mempunyai lilitan yang dihubungkan ke *slip ring* dan umumnya disambung dengan tahanan luar .



Gambar 2.1 Bagian-bagian Motor Induksi 3 Fasa

Sumber : Zuhal, Dasar Tenaga Listrik, 1991, hal. 64

2.1.2. Prinsip Kerja Motor Induksi Tiga Fasa^[7]

Prinsip kerja motor induksi tiga fasa dapat dijelaskan sebagai berikut :

- Apabila sumber tegangan 3 fasa dipasang pada kumparan stator, timbulah medan putar dengan kecepatan :

$$n_s = \frac{120.f}{p} \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (2.1)$$

dimana :

n_s = kecepatan medan putar stator (rpm)

p = jumlah kutub

- Medan putar stator tersebut akan memotong batang konduktor pada rotor, akibatnya pada kumparan rotor akan timbul ggl (E).

$$E = 4,44.f.N.\Phi_m \dots\dots\dots\dots\dots\dots\dots\dots\dots (2.2)$$

2.2. Transformator

2.2.1. Definisi Transformator^[7]

Transformator adalah suatu alat listrik yang dapat memindahkan dan mengubah energi listrik dari satu atau lebih rangkaian listrik ke rangkaian listrik yang lain melalui suatu gandengan magnet berdasarkan prinsip induksi elektromagnet . Transformator digunakan secara luas, baik dalam bidang tenaga listrik maupun elektronika.

2.2.2. Prinsip Kerja Transformator

Prinsip kerja motor transformator dapat dijelaskan sebagai berikut :

- a. Apabila pada lilitan sisi primer diberi tegangan AC V_p , maka pada inti besi(kern) timbul fluks magnet.
- b. Fluks magnet tersebut berputar beriringan dengan arah putaran lilitan sekunder.
- c. Akibat induksi magnetik maka pada sisi sekunder timbul tegangan V_s .
- d. Apabila transformator dianggap ideal dengan mengabaikan rugi-rugi, maka berlaku persamaan :

$$\frac{N_p}{N_s} = \frac{V_p}{V_s} = \frac{I_s}{I_p} \quad \dots \dots \dots \dots \dots \dots \quad (2.4)$$

dimana : N_p = jumlah lilitan primer.

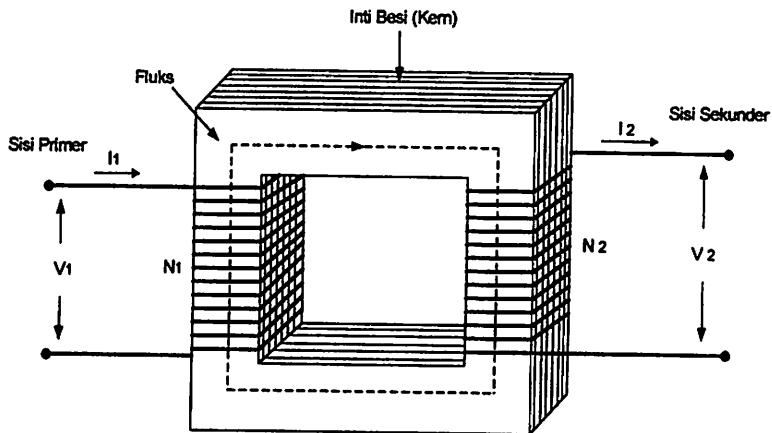
N_s = jumlah lilitan sekunder.

V_p = tegangan primer.

V_s = tegangan sekunder.

I_p = arus primer.

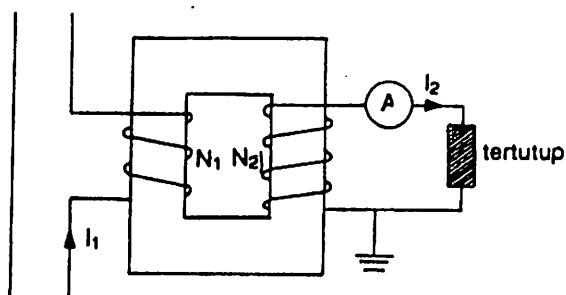
I_s = arus sekunder.



Gambar 2.2 Skema Prinsip Kerja Trafo

2.2.3. Transformator Arus [7]

Transformator arus digunakan untuk mengukur arus beban suatu rangkaian. Dengan menggunakan transformator arus maka arus beban yang besar dapat diukur hanya dengan menggunakan alat ukur (ammeter) yang tidak terlalu besar.



Gambar 2.3 Transformator Arus

Sumber : Zuhal, Dasar Tenaga Listrik, 1991, hal. 50

Beberapa parameter yang perlu diperhatikan dalam konverter analog ke digital, antara lain :

- **Resolusi**

Resolusi merupakan spesifikasi terpenting untuk konverter, dinyatakan dalam jumlah bit yang ada dalam setiap satu data digital. Resolusi konverter menjadi lebih baik bila jumlah bit semakin besar.

- **Akurasi/Ketelitian**

Akurasi merupakan perbedaan/selisih 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**

Yaitu selang waktu yang dibutuhkan untuk mengkonversikan setiap *sample* sinyal analog ke digital.

2.3.2. ADC 0809

ADC 0809 adalah komponen CMOS monolitis dengan sebuah *konverter* analog ke digital 8 bit, multiplekser 8 *input*, 8 bit *konverter A/D*, dan bekerja berdasarkan pendekatan bertingkat dari nilai yang diukur dan logika kontrol yang kompatibel dengan mikroprocessornya. ADC 0809 memiliki toleransi kesalahan $\pm \frac{1}{2}$ LSB dan ± 1 LSB, dapat konversi dalam waktu 100 μs .

- Kelebihan ADC 0809 adalah :

- a. mudah untuk aplikasi ke segala jenis mikrokontroller.
- b. tidak membutuhkan *zero* atau *full-scale*.
- c. *8-channel multiplexer* dengan *address logic*.
- d. range *input* 0V - 5V dengan *power supply* +5v.

- Pembacaan ADC 0809

- a. ADC 0809 punya 8 masukan *analog* (IN 0 sampai IN 7) yang dapat dipilih melalui alamat 3 bit (ADD-A sampai ADD-C).
- b. ADC akan memulai konversi dengan kecepatan sesuai dengan frekuensi/*clock* yang diberikan setelah sinyal *START* diaktifkan.
- c. Setelah konversi selesai maka sinyal *EOC* (*End Of Conversation*) akan aktif untuk menandai bahwa data pada kaki keluaran .
- d. *Output* dapat dibaca dengan bantuan sinyal *OE* (*Output Enable*), komponen ini mempunyai 2 kaki untuk $V_{ref}(+)$ dan $V_{ref}(-)$.
- e. Sedangkan kaki 13 ke GND dan kaki 11 ke Vcc.

IN3	1	28	IN2
IN4	2	27	IN1
IN5	3	26	IN0
IN6	4	25	ADD A
IN7	5	24	ADD B
START	6	23	ADD C
EOC	7	22	ALE
2-5	8	21	2-1 MSB
OUTPUT ENABLE	9	20	2-2
CLOCK	10	19	2-3
V_{cc}	11	18	2-4
$V_{ref}(+)$	12	17	2-8 LSB
GND	13	16	$V_{ref}(-)$
2-7	14	15	2-5

Gambar 2.4 ADC 0809

Sumber: Data sheet ADC 0809

2.4. Mikrokontroler AT89S8252

Mikrokontroler AT89S8252 produksi dari ATMEL merupakan pengembangan dari mikrokontroller standard MCS-51. Seri AT89S8252 adalah mikrokontroller yang membutuhkan daya rendah, memiliki kemampuan yang tinggi, dan merupakan mikrokomputer 8 bit yang dilengkapi 8K byte *Flash PEROM* (*Programmable and Erasable Read Only Memory*) yaitu ROM yang dapat ditulis ulang atau dihapus menggunakan sebuah perangkat programmer. Serta terdapat EEPROM *internal* sebesar 2K Byte.

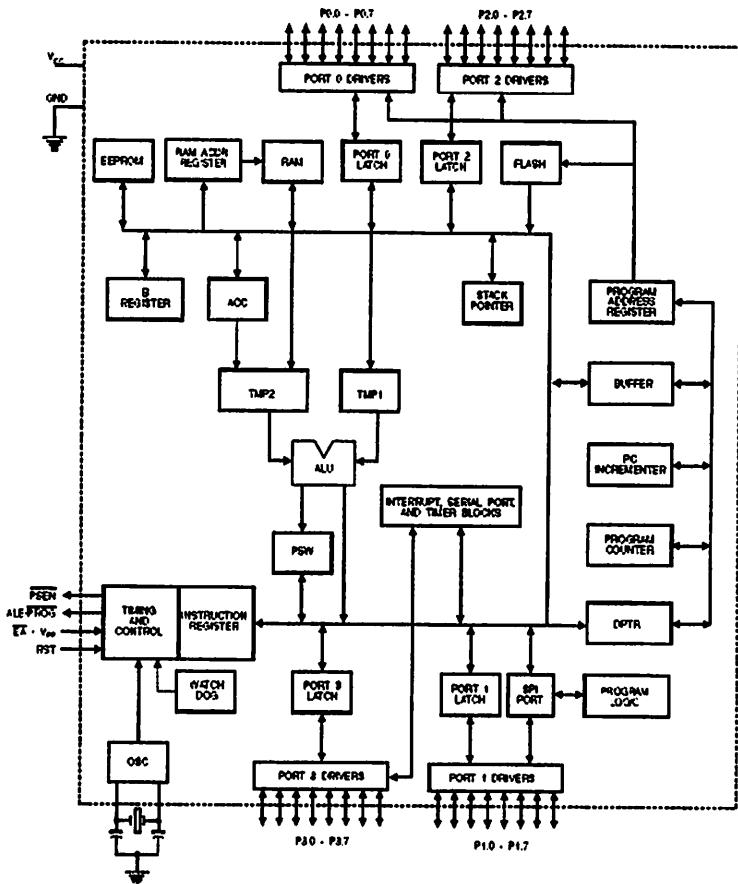
Flash PEROM dalam AT89S8252 menggunakan *Atmel's High-Density Non Volatile Technology* yang mempunyai kemampuan untuk ditulis ulang hingga 1000 kali dan berisikan perintah *standard* MCS-51. Selain itu juga dilengkapi RAM *internal* sebesar 256 byte. Dalam sistem Mikrokontroller terdapat dua hal yang mendasar, yaitu: perangkat keras dan perangkat lunak yang keduanya saling terkait dan mendukung. Mikrokontroller ini digunakan untuk beberapa keperluan mulai dari komersial, industri, otomotif, dan militer.

2.4.1. Fitur Mikrokontroller AT89S8252

Fitur Mikrokontroller AT89S8252 adalah sebagai berikut :

- Kompatibel dengan mikrokontroller MCS-51
- 8K byte In-System Programmable Downloadable Flash Memori
 - SPI Serial Interface for Program Downloading
 - Endurance: 1,000 Write/Erase Cycles

- 2K byte EEPROM
 - Edurance: 100,000 Write/Erase Cycles
- 3 Level program memori lock
- Memory 256 x 8 bit *Internal* RAM
- 32 Port I/O Lines yang dapat dipakai semua
- 3 buah Timer/Counter 16 bit
- Nine Interrupt Source
- Programmable UART (serial port)
- SPI Serial Interface
- Programmable Watchdog Timer
- Dual Data Pointer
- Frekuensi kerja 0 sampai 24 MHz
- Tegangan Operasi 4 Volt sampai 6 Volt



Gambar 2.5. Blok Diagram AT89S8252
Sumber : Datasheet AT89S8252

2.4.2. Arsitektur AT89S8252

Arsitektur Mikrokontroller AT89S8252 adalah sebagai berikut:

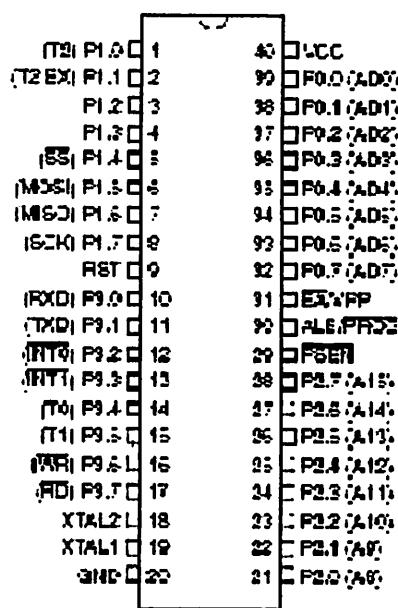
- 8 bit CPU (*Central Processing Unit*) dengan register A dan B.
- 16 bit *Program Counter* (PC) dengan *Data Pointer* (DPTR).
- 8 bit program status word (PSW).
- 8 bit stack pointer (SP).
- Internal EPROM dan ROM dari 0 sampai 4 Kb.
- 256 byte *Internal RAM*

- ⇒ 4 register bank masing-masing 8 register.
- ⇒ 16 byte yang dapat dialamatkan pada *bit level*.
- ⇒ 208 byte *memory general purpose memory data*.

- 32 pin *input/output* tersusun sebagai 4 port masing-masing 8 bit (P0 – P3).
- 3 buah Timer (T0 & T1) dengan masing-masing 16-bit timer/counter.
- Data serial *receiver/transmitter full duplex* yaitu SBUF.
- *Control register* antara lain TCON, TMOD, SCON, PCON, IP dan ME.
- 2 eksternal dan 3 internal sumber *interrupt*.
- Rangkaian *oscillator* dan *clock*.

2.4.3. Konfigurasi Kaki-kaki MCU AT89S8252

Berikut ini adalah bentuk fisik dari AT89S8252:



Gambar 2.6. Bentuk Fisik AT89S8252
Sumber : Datasheet AT89S8252

Fungsi-fungsi dari tiap-tiap pin sebagai berikut :

➤ **Pin 40:Vcc**

Dihubungkan dengan sumber tegangan + 5 V.

➤ **Pin 20: GND**

Dihubungkan dengan Ground rangkaian atau media pentahanan.

➤ **Pin 32-39 : Port 0 (P0.0 – P0.7)**

Port 0 merupakan *port I/O* 8 bit dua arah. *Port* ini digunakan sebagai multipleks bus alamat rendah dan bus data selama pengaksesan ke memori luar.

➤ **Pin 1-8 : Port 1 (P1.0 – P1.7)**

Port 1 dapat difungsikan sebagai masukan atau keluaran dan bekerja baik untuk operasi bit maupun *byte*, tergantung dari pengaturan program yang dibuat.

Tabel 2.1 Fungsi Alternatif Port 1^[9]

POR T PIN	FUNGSI
P1.0	T2 (masukan eksternal untuk Timer/Counter 2)
P1.1	T2EX (Timer/Counter 2 capture/reload trigger dan kontrol arah)
P1.2	-
P1.3	-
P1.4	SS (slave port select input)
P1.5	MOSI (master data output, slave data input untuk chanel SPI)
P1.6	MISO (maser data input, slave data output untuk chanel SPI)
P1.7	SCK (master clock output, slave clock input untuk chanel SPI)

➤ **Pin 21-28 : Port 2 (P2.0 – P2.7)**

Port 2 merupakan port input-output dengan internal pull-up. *Port ini* dapat digunakan sebagai alamat bus baik *byte* tinggi selama adanya akses ke

memori program atau memori data luar. Mengeluarkan address tinggi selama pengambilan (fetching) program memory external. Selama pengaksesan ke external data memory, port 2 mengeluarkan isi P2 SFR. Menerima address dan beberapa control selama pemrograman dan verifikasi.

➤ **Port 3 (P3.0 – P3.7)**

Port 3 mempunyai fungsi sebagai I/O juga mempunyai fungsi khusus sebagai berikut :

- RD (P3.7), sinyal pembacaan memori data luar.
- WR (P3.6), sinyal penulisan memori data luar.
- T1 (P3.5), masukan dari pewaktu/pencacah 1.
- T0 (P3.4), masukan dari pewaktu/pencacah 0.
- INT1 (P3.3), masukan interrupt 1.
- INT0 (P3.2), masukan interrupt 0.
- TXD (P3.1), keluaran pengiriman data untuk serial port (*asynchronous*) atau sebagai keluaran *clock (synchronous)*.
- RXD (P3.0), masukan data serial atau sebagai keluaran data.

➤ **Pin 9 : RST/VPD.**

Merupakan pin input yang aktif jika pin aktif tinggi selama dua siklus mesin maka ketika osilator bekerja akan mereset peralatan.

➤ **Pin 30 : ALE (*Address Latch Enable*).**

Pin ALE (aktif tinggi) mengeluarkan pulsa output untuk menyangga (*latch*) satu byte alamat rendah selama mengakses ke memori eksternal.

ALE dapat mengendalikan 8 beban TTL. Pin ini juga merupakan input pulsa program yang aktif rendah selama pemrograman EPROM. Pada operasi normal, ALE dikeluarkan pada suatu kecepatan yang konstan yaitu 1/6 dari frekuensi osilator dan dapat digunakan untuk *timing* eksternal atau untuk tujuan membuat *clock*.

➤ **Pin 29 : PSEN (*Program Strobe Enable*).**

Pin ini aktif rendah yang merupakan *strobe* pembacaan ke program memori eksternal.

➤ **Pin 18-19 : XTA1&XTAL2.**

Pin XTAL1 merupakan pin input ke penguat osilator pembalik dan XTAL2 merupakan pin output dari penguat osilator pembalik.

➤ **Pin 30 : EA,VPP (*External Access/Programming Supply Voltage*).**

Pin EA di Vcc agar 89S8252 dapat mengakses kode mesin dari program memori.

2.4.4. Organisasi Memori AT89S8252

Mikrokontroler AT89S8252 termasuk keluarga MCS-51 yang memiliki memori program dan memori data yang terpisah. Pemisahan ini dilakukan secara logika sehingga CPU dapat mengakses sampai 64 *Kbyte* memori program dan 64 *Kbyte* memori data. Lebar memori data internal adalah 8 bit dan 16 bit (register PC dan register DPTR).

2.4.4.1. Memori Program

Memori program menggunakan alamat sepanjang 64 *Kbyte* dengan 4 *Kbyte* (alamat \$0000 sampai dengan \$0FFF) yang merupakan memori internal sehingga 60 *Kbyte* merupakan memori eksternal. Dapat menggunakan 64 *Kbyte* memori eksternal sebagaimana yang ditunjukkan pada gambar memori data.

Memori program merupakan tempat penyimpanan data permanen. Memori program lebih dikenal dengan nama *Read Only Memory* (ROM). Data dalam ROM tidak akan terhapus meskipun catu daya dimatikan atau dikenal sebagai sifat *non-volatile*. Karena sifatnya yang demikian ROM dapat digunakan untuk menyimpan program.

Ada beberapa tipe ROM, antara lain :

- ROM (*Read Only Memory*)

Merupakan memori yang sudah diprogram oleh pabrik (ROM murni).

- PROM (*Programmable Read Only Memory*)

Merupakan memori yang dapat diprogram oleh pemakai tetapi tidak dapat diprogram ulang.

- EPROM (*Erasable Programmable Read Only Memory*)

Merupakan PROM yang dapat diulang. ROM ini juga terdapat pada mikrokontroler 8751, hal ini ditandai dengan adanya jendela kaca pada konstruksi IC 8751 yang digunakan untuk menghapus atau memperbaiki program yang sudah ada.

- EEPROM (*Electrical Erasable Programmable Read Only Memory*)

Pada prinsipnya hampir sama dengan EPROM, tetapi perbedaananya terletak pada pengosongan atau penghapusan program. Untuk EPROM dapat dihapus dengan menggunakan sinar *ultra violet*, sedangkan pada EEPROM pengisian program dapat dilakukan langsung atau menumpuk program lama dengan program yang baru. EEPROM lebih fleksibel dibandingkan EPROM.

2.4.4.2. Memori Data

Memori data merupakan tempat penyimpanan data yang bersifat sementara atau *volatile*. Dengan kata lain data akan hilang bila tidak dicatat. Memori data lebih dikenal dengan nama RAM (*Random Access Memory*), yaitu dapat dilakukan pembacaan dan penulisan data alamat yang tersedia.

Memori MCS-51 mempunyai 128 *bytes* - 256 *byte* RAM internal ditambah sejumlah register fungsi khusus atau *Special Function Register* (SFR). Selain mempunyai memori internal, MCU AT89S8252 mempunyai memori eksternal yang memiliki pengalamatan sampai 64 *Kbytes*.

Pada keluarga mikrokontroler MCS-51, ruang memori data eksternal terbagi menjadi 3 blok yang disebut *lower 128*, *upper 128* dan ruang SFR,

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

1. Register bank 0 – 3

Lokasi bank register dimulai dari alamat 00h – 1h 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 dialamat secara langsung yaitu dari \$00h sampai \$7fh.

3. Scratch Pad Area

Lokasi dari alamat \$30h sampai \$7fh atau sebanyak 80 *bytes* yang dapat digunakan sebagai alamat bagi RAM.

Pada 128 *bytes* atas (*upper* 128) ditempati oleh register yang mempunyai fungsi khusus yang disebut dengan *Special Function Register* (SFR). Ruang dari register fungsi khusus ini adalah dari 80h sampai FFh. Berikut ini adalah contoh isi vector alamat pada *Special Function Register*.

- Akumulator (Acc) atau register A dan register B.

Kedua register tersebut digunakan untuk operasi perkalian dan pembagian.

- *Program Status Word*

Register ini meliputi bit-bit : CY (*Carry*), AC (*Auxillary Carry*), FO sebagai flag, RS0 dan RS1 untuk pemilih register bank, OV (*Over Flow*), dan *parity flag*.

- *Stack Pointer* (SP)

SP merupakan register yang digunakan untuk penunjuk alamat. Register ini berguna apabila digunakan suatu *routine* pada program utama.

- *Data Pointer High* (DPH) dan *Data Pointer Lower* (DPL)

DPTR adalah register yang digunakan untuk pengalamatan tidak langsung. Register ini digunakan untuk mengakses memori program baik

internal maupun eksternal. DPTR dikontrol oleh 2 buah register 8 bit yaitu DPH dan DPL.

- *Port 0, Port 1, Port 2, Port 3*

Pada keluarga 8051 masing-masing *port* dapat dialamati langsung baik secara *byte* atau bit. Masing-masing *port* merupakan *port bi-directional (input/output)* :

1. *Port 0* digunakan sebagai pengalamatan memori dari luar.
2. *Port 1* digunakan sebagai I/O dari mikrokontroler.
3. *Port 2* digunakan sebagai pengalamatan memori dari luar.
4. *Port 3* berisi sinyal kontrol seperti *interrupt serial*, WR, dan RD.

- *Register Prioritas Interrupt (Interrupt Priority Register /IP).*

Merupakan register yang berisi bit-bit untuk mengaktifkan prioritas dari suatu *interrupt* yang ada pada mikrokontroler pada taraf yang diinginkan.

- *Interrupt Enable Register*

Merupakan register yang berisi bit-bit untuk menghidupkan atau mematikan sumber-sumber *interrupt*.

- *Timer/Counter Control Register*

TCON merupakan register yang berisi bit-bit memulai atau menghentikan pencacah atau pewaktu.

- *Serial Control Buffer*

Register ini digunakan untuk menampung data masukan (SBUF *in*) atau keluaran (SBUF *out*) dari *serial port*.

Tabel 2.2 Pembagian Alamat pada SFR^[9]

SYMBOL	NAME	ADDRESS
ACC	ACCUMULATOR	0E0H
B	B REGISTER	0F0H
PSW	PROGRAM STATUS WORD	0D0H
SP	STACK POINTER	81H
DPTR	DATA POINTER 2 BYTE	84 H
DPL	LOW BYTE	82H
DPH	HIGH BYTE	83H
P0	PORT 0	80H
P1	PORT 1	90H
P2	PORT 2	0A0H
P3	PORT 3	080H
IP	INTERRUPT PRIORITY CONTROL	088H
IE	INTERRUPT ENABLE CONTROL	0ABH
TMOD	TIMER/COUNTER MODE CONTROL	89H
TCON	TIMER/COUNTER CONTROL	88H
+TCON	TIMER/COUNTER 2 CONTROL	0CBH
TH0	TIMER/COUNTER 0 HIGH CONTROL	8CH
TL0	TIMER/COUNTER 0 LOW CONTROL	8DH
TH1	TIMER/COUNTER 1 HIGH CONTROL	8DH
TL1	TIMER/COUNTER 1 LOW CONTROL	8CH
TH2	TIMER/COUNTER 2 HIGH CONTROL	0CDH
TL2	TIMER/COUNTER 2 LOW CONTROL	0CCH
RCAP2H	T/C 2 CAPTURE REG. HIGH BYTE	0CBH
+RCAP2L	T/C 2 CAPTURE REG. LOW BYTE	0CAH
SCON	SERIAL CONTROL	98H
SBUF	SERIAL DATA BUFFER	99H
PCON	POWER CONTROL	87H

2.4.5. SFR Tambahan Pada AT89S8252

Selain memiliki SFR (*Special Function Register*) seperti halnya pada MCS-51, mikrokontroller AT89S8252 memiliki tambahan SFR. Hal ini tak lain karena adanya fitur tambahan pada mikrokontroller AT89S8252.

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

2.4.5.1. 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 SFR yang bernama T2CON (Timer 2 Control), T2MOD (Timer 2 Mode) dan sepasang register RCAP2H, RCAP2L yang merupakan register *capture/reload* untuk Timer 2 dalam 16-bit *capture mode/auto-reload mode*.

2.4.5.2. SFR Untuk Watchdog dan Memori

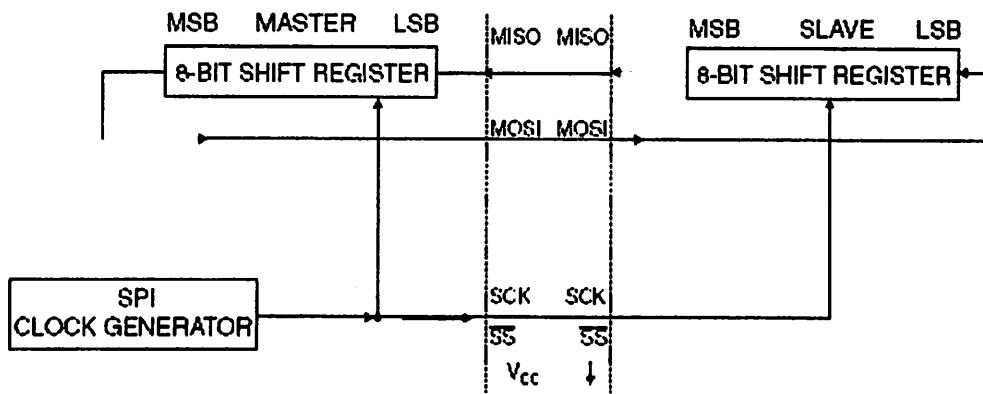
Untuk menggunakan Watchdog Timer/Memori, maka dapat dilakukan dengan mengatur SFR yang bernama WMCON dengan alamat 96H.

2.4.5.3. 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
- e. Sistem data transfer MSB dahulu atau LSB
- f. *Write Collision Flag Protection*

Gambar berikut menunjukkan hubungan antara CPU master dan slave.

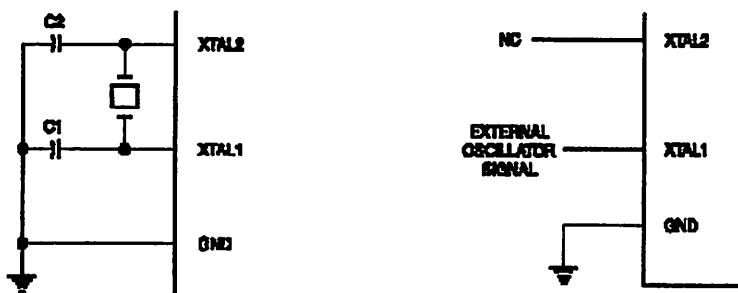


Gambar 2.7 SPI Master-slave Interconnection
Sumber : Datasheet AT89S8252

2.4.6. Osilator

Jantung dari AT89S8252 adalah rangkaian yang membangkitkan pulsa clock yang mensinkronkan semua operasi internal. Mikrokontroler AT89S8252 memiliki osilator internal (*on-chip oscillator*) yang dapat digunakan sebagai sumber waktu (clock) bagi CPU. Untuk menggunakan osilator internal diperlukan sebuah kristal atau resonator keramik antara pin XTAL₁ dan XTAL₂, dan sebuah kapasitor ke ground. XTAL₂ dan XTAL₁ secara berurutan merupakan input dan output dari sebuah inverting amplifier yang dapat dikonfigurasikan penggunaannya sebagai *on-chip oscillator* seperti yang ditunjukkan pada gambar 2.8a.

Untuk memberikan IC AT89S8252 sumber clock eksternal, maka pin XTAL₂ dibiarkan tidak berhubungan dengan sumber clock eksternal dan XTAL₁ dihubungkan dengan sumber clock eksternal seperti pada gambar 2.8b.



a). *Oscillator Connector* b). *External Clock Drive Configuration*

Gambar 2.8 Karakteristik Osilator
Sumber : Data Sheet AT89S8252

2.4.7. Timer dan Counter

Dalam mikrokontroler AT89S8252 terdapat 3 buah perekaki/pencacah (Timer/Counter) 16-bit yang dapat diatur melalui perangkat lunak, yaitu perekaki/pencacah 0 dan perekaki/pencacah 1. Timer/Counter ini diatur oleh SFR (*Special Function Register*) yaitu Timer/Counter Control (TCON dengan alamat 88_H) dan Timer/Counter Mode Control (TMOD dengan alamat 89_H). Selain itu nilai byte bawah dan byte atas dari Timer/Counter disimpan dalam register TL dan TH.

Jika difungsikan sebagai Timer, maka akan menggunakan sistem 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 T0 (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^[2]

MODE	TIMER/COUNTER 0	TIMER/COUNTER 1
0	13-bit Timer	13-bit Timer
1	16-bit Timer	16-bit Timer
2	8-bit auto-reload	8-bit auto-reload
3	Dua 8-bit Timer	Tidak bekerja

Pada mikrokontroler 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 pin 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^[2]

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

Keterangan :

RCLK = *Receive Clock Enable*. Jika dieset menyebabkan serial port menggunakan pulsa *overflow* Timer 2 sebagai detak penerimaan pada serial port. Jika RCLK = 0, maka Timer 1 yang digunakan

TCLK = *Transmit Clock Enable*. Jika diset menyebabkan serial port menggunakan pulsa *overflow* Timer 2 sebagai detak pengiriman. Jika TCLK = 0, maka 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*

2.4.8. Sistem Interrupt

AT89S8252 memiliki 6 buah sumber interupsi, 2 eksternal interupsi (INT0 dan INT1), 3 Timer interupsi (Timer 0,1 dan 2) dan satu serial port interupsi.

INT0 = interrupt pada P3.2 (kaki 12)

INT1 = interrupt pada P3.3 (kaki 13)

Timer 0 = Timer pada P3.4 (kaki 14)

Timer 1 = Timer pada P3.5 (kaki15)

Port serial = jika pengiriman/penerimaan suatu frame telah lengkap

Saat terjadinya interupsi, mikrokontroler secara otomatis akan menuju *subroutine* pada alamat tersebut. Setelah interupsi servis selesai dikerjakan, mikrokontroler akan mengerjakan program semula. Dua sumber eksternal adalah INT0 dan INT1, kedua interupsi eksternal akan aktif, transisi tergantung isi IT0 dan IT1 pada register TCON. Interrupt T0 dan T1 aktif pada saat Timer yang sesuai mengalami *roll over*. Interupsi serial akan dibangkitkan dengan melakukan operasi OR pada R1 dan T1 tiap-tiap sumber interupsi dapat *enable* atau *disable* secara *software*. Tingkat prioritas semua sumber interupsi dapat diprogram sendiri-sendiri dengan set atau *clear bit* pada SFR IP (*interrupt priority*). Register yang akan berperan dalam mengatur aktif tidaknya interupsi adalah Interrupt Enable Register.

2.4.9. Metode Pengalamatan

Metode pengalamatan yang digunakan pada MCS-51 terbagi menjadi dua jenis, yaitu pengalamatan langsung dan pengalamatan tidak langsung.

ϕ Pengalamatan Tak Langsung

Operasi pengalamatan tak langsung menunjukkan ke sebuah register yang berisi lokasi alamat memori yang akan digunakan dalam suatu operasi. Lokasi yang nyata tergantung dari isi register saat instruksi dijalankan. Untuk melakukan pengalamatan tak langsung digunakan simbol @. Misalnya :

- ADD A, @R0 : tambahkan isi R0 dengan Acc dan hasilnya di Acc
- DEC @R1 : kurangi isi dari alamat R1

ϕ Pengalamatan Langsung

Pengalamatan langsung dilakukan dengan memberikan nilai ke suatu register secara langsung. Untuk melakukan hal tersebut digunakan tanda #. Misalnya:

- MOV A,#01H : isi Acc dengan data 01H
- MOV DPTR,#19H : isi DPTR dengan data 19H

Pengalamatan data langsung dari 0 sampai 127 akan mengakses RAM internal, sedangkan pengalamatan dari 128 sampai 255 akan mengakses register perangkat keras. Misalnya :

- MOV P3,A : pindahkan isi Acc ke.alamat Port 3 (B0H)
- INC 50 : naikkan lokasi 50 (desimal) dalam memori

2.5. Bahasa Assembler MCS-51

Bahasa assembler digunakan dalam setiap operasi CPU dalam bentuk bahasa simbol yang disusun berurutan dalam pernyataannya. Masing-masing pernyataan akan diterjemahkan ke dalam instruksi bahasa mesin atau sering disebut *operation code/opcode*. Dalam penulisan bahasa mesin ini, terdapat berbagai macam kelompok instruksi, diantaranya :

- Perpindahan Data

Instruksi ini digunakan untuk memindahkan data antar register, memori, register-memori, antar muka register dan antar muka memori.

Contohnya: MOV A,R0 : memindahkan isi register R0 ke Acc.

MOV A,@R0 : memindahkan isi alamat R0 ke Acc.

- Operasi Aritmatika

Instruksi ini melaksanakan operasi aritmatika yang meliputi penjumlahan, pengurangan, perkalian, maupun pembagian.

Contohnya: ADD A,#data : menambah Acc dengan data.

ADC A,#data : menambah Acc dengan data dan carry.

INC R6 : menambah isi R6 dengan 1.

DEC R7 : mengurangi isi R5 dengan 1.

MUL AB : mengalikan isi Acc dengan isi register B.

DIV AB : membagi isi Acc dengan isi register B.

- Operasi Percabangan

Instruksi ini mengubah urutan normal pelaksanaan suatu program untuk melaksanakan pada lain tempat yang kita perlukan pada saat itu.

Contohnya:

1. CJNE (*Compare Jump Not Equal*)

Instruksi ini membandingkan isi lokasi memori tertentu dengan isi Acc, jika sama instruksi ini selanjutnya akan dieksekusi. Jika tidak sama eksekusi akan kembali ke alamat kode yang telah ditunjuk.

2. JB (*Jump if Bit Set*)

Instruksi ini akan menguji suatu alamat bit isi satu, eksekusi akan menuju ke alamat kode dan jika tidak instruksi akan dilanjutkan.

3. JNB (*Jump if Bit Not Set*)

Instruksi ini menguji suatu alamat bit. Jika berisi 0 maka eksekusi akan menuju ke alamat kode. Jika berisi 1 maka instruksi selanjutnya yang akan dieksekusi.

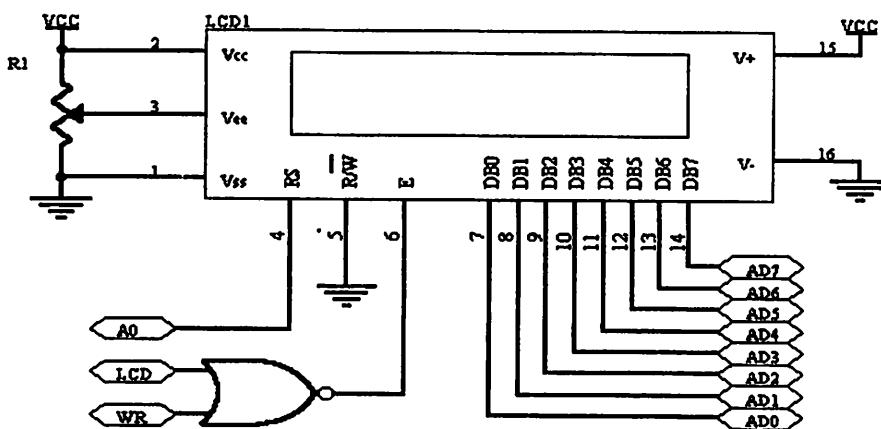
2.6. LCD

Liquid Crystal Display atau LCD merupakan komponen optoelektronik yaitu komponen yang bekerja atau dipengaruhi oleh sinar (optolistrik), komponen pembangkit cahaya (light emitting) dan komponen-komponen yang akan mengubah sinar. LCD terbuat dari bahan kristal cair yang merupakan suatu komponen organik dan mempunyai sifat optik seperti benda padat meskipun bahan tetap cair.

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

Karena sel-sel kristal cair merefleksikan cahaya dan bukan membangkitkan cahaya maka konsumsi daya yang dibutuhkan relatif rendah. Energi yang dipergunakan hanya untuk mengaktifkan kristal cair. Pada dasarnya LCD bekerja pada tegangan rendah (3 – 15 Vrms), frekuensi rendah (25 – 60 Hz) sinyal AC dan memakai arus listrik yang sangat kecil (25 - 300 μ A). LCD seringkali ditata sebagai tampilan *seven segment* untuk menampilkan angka tetapi juga memiliki keistimewaan lain, yaitu kemampuan untuk menampilkan karakter dan berbagai macam simbol.

Salah satu jenis LCD diantaranya adalah LCD M1632, suatu jenis piranti dengan konsumsi daya yang rendah, disusun dari dot matrik dan dikontrol oleh ROM atau RAM generator karakter dan RAM data display. Pengontrolan utamanya adalah pada ROM generator dan display data RAM yang menghasilkan kode ASCII jika padanya diberikan input ASCII. Untuk dapat difungsikan dengan baik maka perlu diperhatikan proses analisis yang telah ditentukan oleh pabrik pembuatnya. Timing penganalisaian sangat dipertimbangkan, karena jika meleset sampai ordo milli secon maka dapat dipastikan LCD tidak dapat berfungsi.



Gambar 2.9 Konfigurasi Kaki LCD

Adapun karakteristik dari LCD M1632 antara lain :

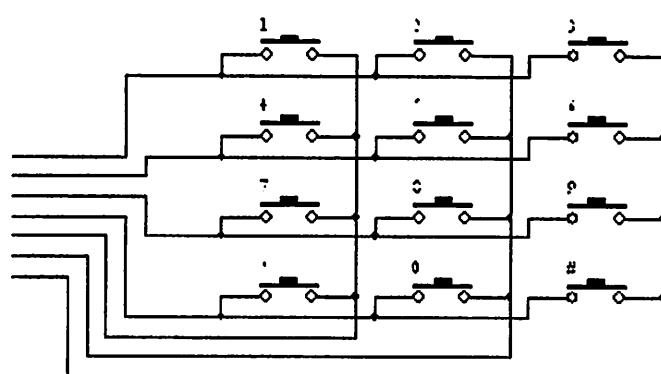
- Dengan 16 karakter – 2 baris dalam bentuk dotmatrik 5x7 dan cursor.
- *Duty ratio 1/16.*
- Memiliki ROM pembangkitan karakter untuk 192 jenis karakter.
- RAM untuk data display sebanyak 80x8 bit.
- Dapat dirangkai dengan MPU 8 bit atau 4 bit.
- RAM data display dan RAM pembangkit karakter dapat dibaca oleh MPU.
- Memiliki fungsi instruksi antara lain *display on/off*, *Cursor on/off*, *display karakter blink*, *cursor shift* dan *display shift*.
- Memiliki rangkaian osilator sendiri.
- Catu tegangan tunggal yaitu ± 5 V.
- Memiliki rangkaian reset otomatis .
- Temperatur operasi 0° C - 50° C.

2.7. Keypad (Papan Tombol)

Keypad digunakan untuk memasukkan data referensi dan mengubah data bila diinginkan. Penggunaan *keypad* pada mikrokontroller merupakan pemrograman input karena data diberikan untuk dibaca oleh mikrokontroller.

Bila kita menggunakan tombol yang jumlahnya cukup banyak, maka untuk menghindari pemakaian jumlah port yang besar, kita dapat menyusun tombol tersebut secara matrix. Di pasaran yang paling banyak digunakan yaitu *keypad* dengan matrix 4×4 dan 4×3 .

Pada skripsi ini digunakan *keypad* 4×3 sehingga dibutuhkan 7 kaki/pin port untuk dapat menterjemahkan sebanyak 12 tombol. Baris dijadikan sebagai keluaran dan kolom sebagai masukan. Pada suatu waktu hanya ada satu keluaran yang berlogika 1 yang lainnya berlogika 0, pada saat tersebut kita baca masukan tiap-tiap kolom, bila ada yang berlogika 1 maka terdapat tombol yang ditekan yang terletak pada matrix kolom dan baris tersebut.

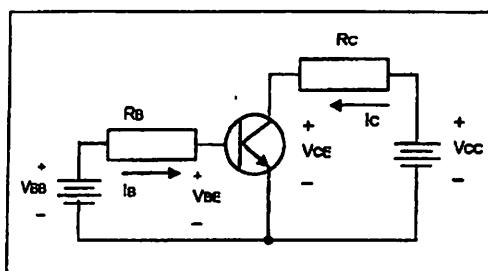


Gambar 2. 10 Keypad 4 x 3
Sumber : Perancangan

2.8. Transistor sebagai Saklar

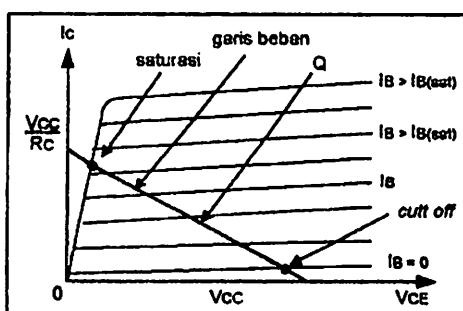
Transistor bekerja sebagai saklar maksudnya transistor dioperasikan pada salah satu dari saturasi atau *cut off*. Jika transistor berada pada titik saturasi sebagai akibat arus basis yang cukup besar, transistor tersebut seperti sebuah saklar tertutup. Jika arus basis kecil sekali maka transistor akan seperti sebuah saklar terbuka.

Rangkaian tipikal transistor sebagai saklar diperlihatkan dalam gambar 2.11. Jika arus basis lebih besar atau sama dengan $I_{B(Sat)}$, titik kerja Q berada pada ujung atas dari garis beban, transistor seperti sebuah saklar yang tertutup. Sebaliknya jika arus basis nol, transistor bekerja pada ujung bawah dari garis beban yang, transistor berperilaku seperti sebuah saklar yang terbuka.



Gambar 2.11 Rangkaian Transistor Sebagai Saklar

Sumber: Malvino, Prinsip-Prinsip Elektronika, 1985, hal 127



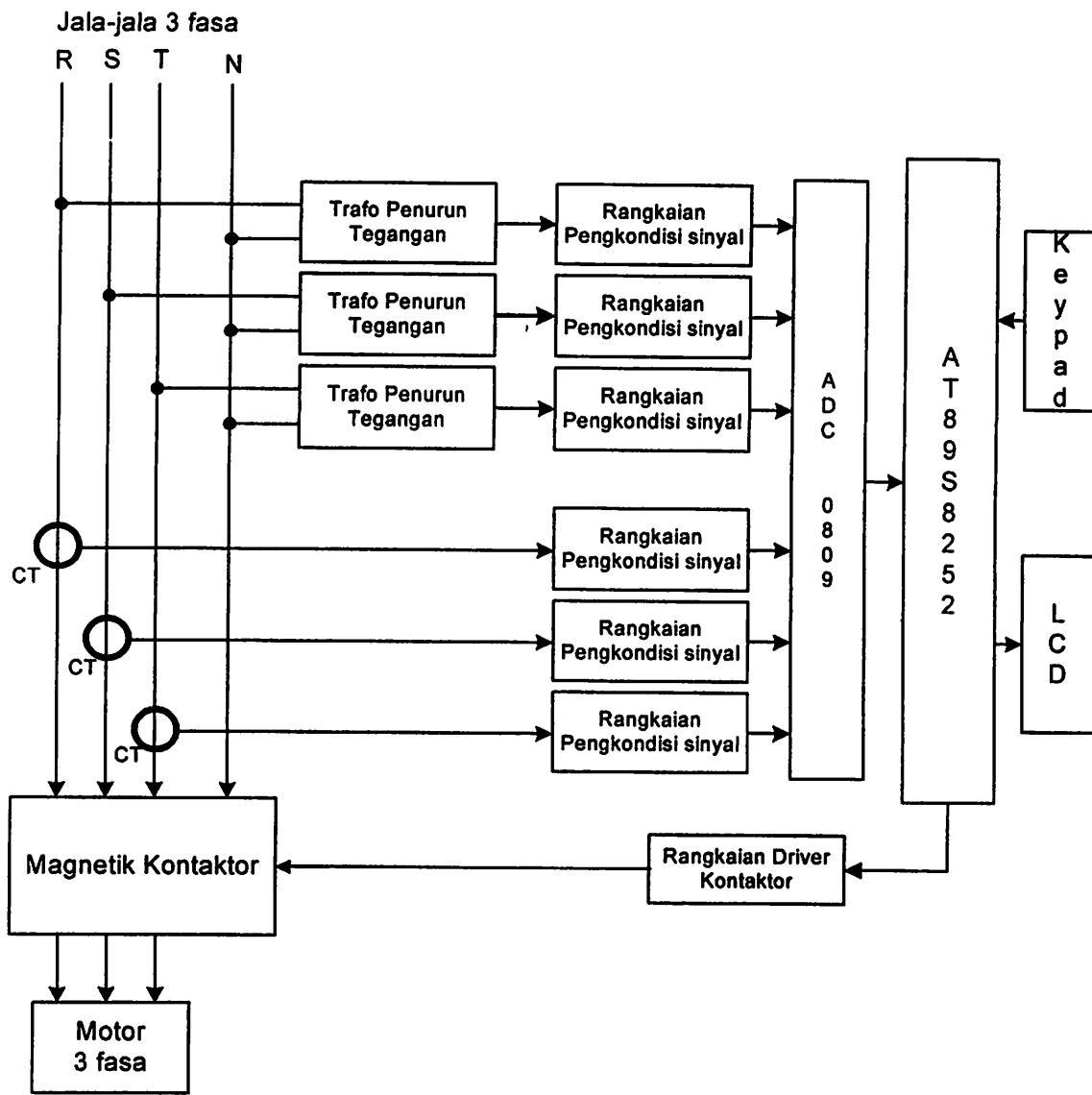
Gambar 2.12 Garis Beban DC

Sumber: Malvino, Prinsip-Prinsip Elektronika, 1985, hal 127

BAB III

PERENCANAAN DAN PEMBUATAN ALAT

3.1. Diagram Blok Rangkaian



Gambar 3.1 Diagram Blok Rangkaian
Sumber : Perancangan

3.1.1. Fungsi Masing-masing Bagian

1. Trafo penurun tegangan berfungsi sebagai sensor tegangan, menurunkan tegangan dari 0 – 240 V AC menjadi 0 – 6 V AC menyesuaikan besar tegangan yang diminta oleh ADC. Besarnya tegangan jala-jala disensor oleh trafo penurun tegangan. Apabila tegangan jala-jala naik maka tegangan pada sisi sekunder trafo penurun tegangan juga naik, demikian pula sebaliknya.
2. Trafo arus berfungsi sebagai sensor arus. Alat ini direncanakan untuk mengamankan motor 1 – 2 HP dan dapat di-setting besarnya arus pengamanannya maksimal 9,99 Amper. Sehingga trafo arus direncanakan dapat mengeluarkan tegangan 0 – 6 V AC pada sisi sekundernya apabila pada sisi primernya mengalir arus 0 – 9,99 Amper AC. Besarnya arus pada motor disensor oleh trafo arus. Apabila arus yang mengalir pada motor naik maka tegangan pada sisi sekunder trafo arus juga naik, demikian pula sebaliknya.
3. Rangkaian pengkondisi sinyal berfungsi untuk mengkondisikan sinyal tegangan yang keluar dari sensor tegangan maupun sensor arus agar dapat dibaca oleh ADC. Tegangan AC yang keluar dari sensor dirubah menjadi DC dan juga diatur besarnya agar sesuai dengan yang diminta ADC yaitu 0 – 5 V DC (sesuai datasheet).
4. ADC berfungsi mengkonversi data analog menjadi data digital sebagai masukan mikrokontroller.
5. Keypad berfungsi untuk memasukkan data settingan yang kita inginkan.
6. LCD berfungsi untuk menampilkan menu settingan dan hasil monitoring tegangan dan arus.

7. Mikrokontroller AT89S8252 berfungsi sebagai pengatur sistem secara keseluruhan memproses masukan dari ADC maupun dari keypad untuk diolah sesuai yang kita programkan pada *software*.
8. Rangkaian driver kontaktor berfungsi untuk meng-*on/off*-kan kontaktor apabila ada perintah dari mikrokontroller.

3.1.2. Prinsip Kerja

Prinsip kerja secara umum dapat dijelaskan sebagai berikut :

1. Sebelum masuk ke motor induksi 3 fasa, besarnya tegangan tiap fasa disensor oleh sensor tegangan. Sinyal dari sensor tegangan diolah lebih dulu oleh rangkaian pengkondisi sinyal sebelum dikonversikan ke data digital oleh ADC sebagai masukan mikrokontroller.
Apabila ada salah satu atau lebih tegangan fasa yang besarnya melebihi batasan setting tegangan minimal dan maksimal serta waktu tunda (*delay time*) yang kita tetapkan melalui keypad ke dalam mikrokontroller dengan tampilan LCD, maka relay pada rangkaian driver kontaktor membuka untuk memutus arus yang akan masuk ke magnetik kontaktor, sehingga kontaktor tidak akan dapat bekerja meskipun tombol *push-button* ON pada rangkaian kontaktor ditekan. Motor baru dapat beroperasi apabila persyaratan besarnya tegangan dipenuhi oleh jaringan jala-jala 3 fasa. Setelah motor beroperasi apabila terjadi gangguan tegangan maka rangkaian akan bekerja seperti hal di atas.
2. Pada saat motor induksi 3 fasa beroperasi, arus yang mengalir pada tiap-tiap fasa pada motor disensor oleh sensor arus. Sinyal dari sensor arus diolah lebih

dulu oleh rangkaian pengkondisi sinyal yang kemudian baru dikonversikan ke data digital oleh ADC sebagai masukan mikrokontroller.

Apabila ada salah satu atau lebih, arus motor yang melebihi batasan setting arus dan setting waktu tunda (*delay time*) yang kita tetapkan ke dalam mikrokontroller melalui keypad dengan tampilan pada LCD, maka mikrokontroller memberi sinyal ke rangkaian driver kontaktor untuk meng-off-kan magnetik kontaktor , sehingga motor berhenti beroperasi.

3.2. Perencanaan *Hardware* (Perangkat Keras)

3.2.1. Perencanaan Sensor Tegangan

Sensor tegangan merupakan rangkaian yang terdiri atas trafo penurun tegangan yang kemudian dirangkai dengan rangkaian pengkondisi sinyal.

Tahapan perencanaan :

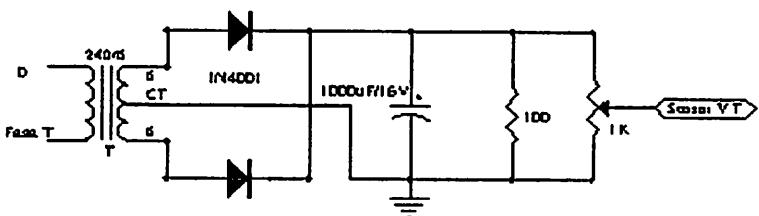
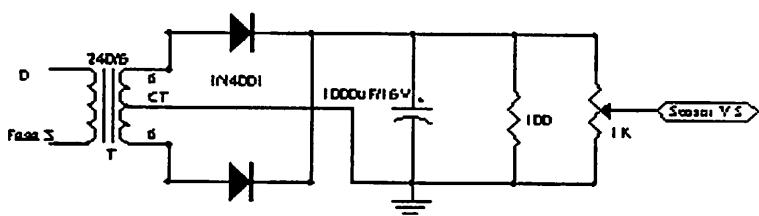
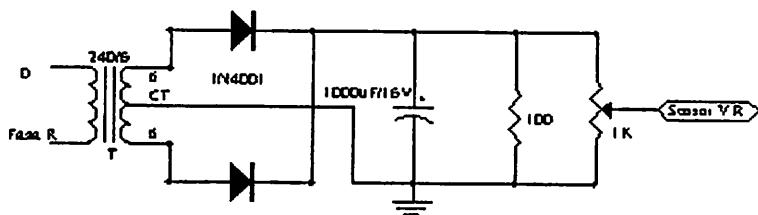
1. Karena besarnya tegangan masukan yang diminta/dibaca ADC 0809 berkisar 0 – 5 V DC (sesuai data sheet), maka digunakan sensor tegangan yang berupa trafo penurun tegangan 240V/6V CT/ 350 mA.
2. Tegangan dari trafo penurun tegangan kemudian disearahkan oleh 2 buah deoda IN4001 karena arus yang mengalir tidak melebihi dari 1 A.
3. Sebagai filter untuk memperkecil ripple dan menstabilkan tampilan LCD digunakan kapasitor $C = 1000 \mu F/16 V$.
4. Untuk mengatur dan mengkalibrasi tegangan yang masuk ke ADC dipakai $VR = 1 K\Omega$ sebagai pembagi tegangan.

5. Sebagai beban dipakai Resistor = 100Ω , $\frac{3}{4}$ Watt yang diparalel dengan VR dan kapasitor, untuk mengatur waktu pengosongan muatan kapasitor, agar respon tampilan LCD dapat dengan cepat mengikuti fluktuasi tegangan jala-jala.

Kapasitor akan terbuang habis muatannya setelah $5.RC$ detik [6].

dimana ; R dalam satuan $M\Omega$, sedangkan C dalam satuan μF .

$$\text{Sehingga waktu pengosongan muatan} = 5 \times \frac{100 \times 1000}{1100} E^{-6} \times 1000 \\ = 0,45 \text{ detik}$$



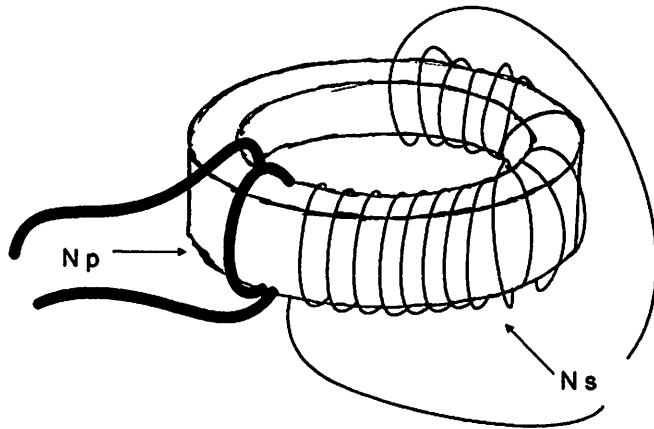
Gambar 3.2 Rangkaian Sensor Tegangan
Sumber : Perancangan

3.2.2. Perencanaan Sensor Arus

Sensor arus yang digunakan berupa trafo arus yang kemudian dirangkai dengan rangkaian pengkondisi sinyal berupa rangkaian penyearah dan pengatur tegangan. Dalam skripsi ini alat pengaman direncanakan untuk di-*setting* pada rentang pengamanan arus 0 – 9,99 amper AC.

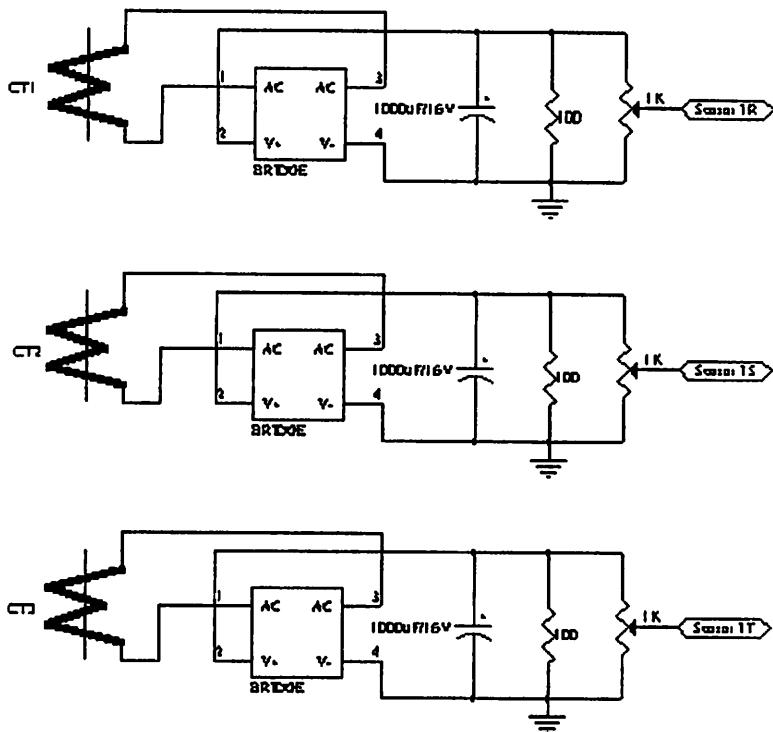
Karena tegangan masukan yang diminta ADC besarnya 0 – 5 volt DC, sehingga dengan besar arus 0 – 9,99 amper AC yang mengalir pada sisi primer trafo arus, direncanakan/diharapkan akan keluar tegangan 0 – 6 volt AC pada sisi sekundernya. Selanjutnya akan keluar tegangan 0 – 5 volt DC setelah melalui rangkaian pengkondisi sinyal kemudian masuk ke ADC.

Melalui beberapa percobaan didapatkan trafo arus dengan jumlah lilitan primer $N_p = 2$ dan jumlah lilitan sekunder $N_s = 50$, dengan ukuran inti besi/kern \emptyset luar = 6 cm, \emptyset dalam = 4,2 cm dan tinggi = 2,5 cm, dapat menghasilkan tegangan sekunder 0 – 6 volt AC untuk arus primer 0 – 9,99 amper AC.



Gambar 3.3 Trafo Arus

Sumber : Perancangan



Gambar 3.4 Rangkaian Sensor Arus

Sumber : Perancangan

Tahapan perencanaan :

1. Karena besarnya tegangan masukan yang diminta/dibaca ADC 0809 berkisar 0 – 5 V DC (sesuai data sheet), maka digunakan sensor arus yang berupa trafo arus dengan tegangan keluaran maksimal 6 volt AC pada sisi sekundernya jika pada sisi primernya dialiri arus 9,99 A AC.
2. Tegangan dari trafo penurun tegangan kemudian disearahkan oleh kiprok/deoda jembatan.
3. Sebagai filter untuk memperkecil ripple dan menstabilkan tampilan LCD digunakan kapasitor $C = 1000 \mu F / 16 V$.
4. Untuk mengatur dan mengkalibrasi tegangan yang masuk ke ADC dipakai $VR = 1 K\Omega$ sebagai pembagi tegangan.
5. Sebagai beban dipakai Resistor = 100Ω , $\frac{1}{4}$ Watt yang dipararel dengan VR dan kapasitor untuk mengatur waktu pengosongan muatan kapasitor, agar respon tampilan LCD dapat dengan cepat mengikuti fluktuasi arus fasa.

Kapasitor akan terbuang habis muatannya setelah $5.RC$ detik ^[6].

dimana ; R dalam satuan $M\Omega$, sedangkan C dalam satuan μF .

$$\text{Sehingga waktu pengosongan muatan} = 5 \times \frac{100 \times 1000}{1100} E^{-6} \times 1000 \\ = 0,45 \text{ detik}$$

3.2.3. ADC 0809

Data tegangan yang diperoleh dari sensor adalah berupa besaran tegangan analog, maka data analog tersebut harus diubah ke bentuk data digital 8 bit agar dapat dibaca oleh mikrokontroler. Untuk itu digunakan konverter analog ke digital (ADC). ADC yang dipakai dalam perancangan ini adalah ADC 0809 yang merupakan ADC dengan 8 masukan analog yang dimultipleks menjadi data digital 8 bit. Dalam perancangan alat ini dipakai 6 buah masukan yaitu IN-0 sampai IN-5, 3 masukan untuk sensor tegangan dan 3 masukan lagi untuk sensor arus. Untuk menyeleksi keenam *input* tersebut maka seluruh alamat yang ada pada ADC 0809 dihubungkan pada mikrokontroler AT 89S8252.

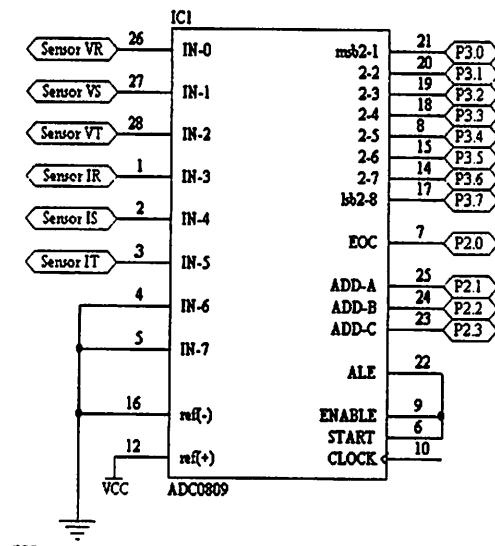
Dikarenakan tegangan yang terukur cukup kecil maka tingkat resolusi dari ADC 0809 diharapkan cukup kecil, sehingga digunakan $V_{ref} = 5$ volt, dengan tingkat resolusi ADC 0809 adalah :

$$\text{Resolusi} = V_{\text{referensi}} / (2^8 - 1) \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots(3.1)$$

$$\text{Resolusi} = 5/(2^8 - 1)$$

$$= 0,0196 \text{ V}$$

Jadi besarnya resolusi adalah sebesar $0,0196 \text{ V} \approx 0,02 \text{ V}$



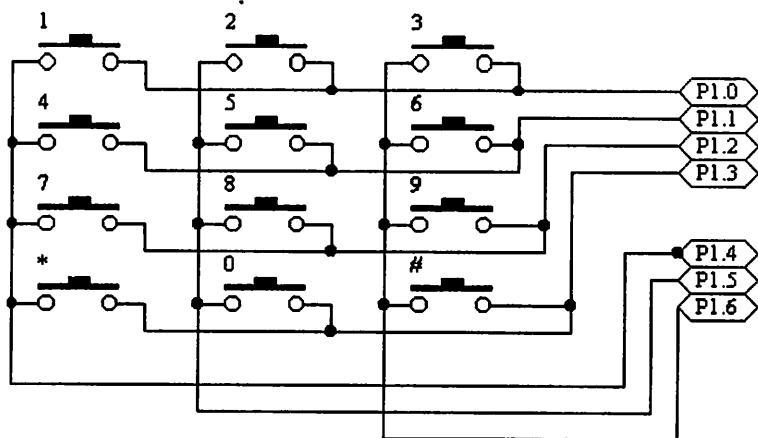
Gambar 3.5 ADC 0809

Sumber : Perancangan

Pin-pin lain yang dihubungkan dengan mikrokontroler ialah *End of Conversion* (EOC), data disalurkan melalui pin 2-1 sampai 2-8 ke port 3, sedangkan alamat A, B, C dihubungkan ke port 2 mikrokontroller. Pin ALE, ENABLE dan START dihubungkan jadi satu agar ADC bekerja secara auto trigger/running.

Untuk membuat ADC 0809 dapat bekerja, maka diperlukan sebuah clock. Pada datasheet tertulis bahwa frekwensi clock berkisar antara 10 KHz sampai 1280 KHz. Pada perancangan ini digunakan sumber pulsa dari rangkaian pembangkit clock/pewaktu 555.

Pada perancangan alat ini digunakan *keypad* 4x3 sehingga dibutuhkan 7 kaki/pin port untuk dapat menterjemahkan sebanyak 12 tombol. Deretan baris dihubungkan dengan port P1.0 – P1.3 pada mikrokontroller, sedangkan deretan kolom dihubungkan dengan port P1.4 – P1.6 pada mikrokontroller.



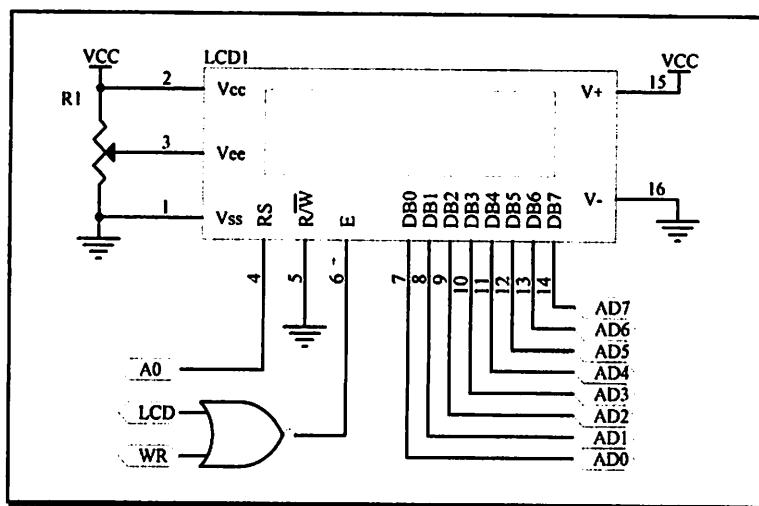
Gambar 3.7 Keypad 4 x 3
Sumber : Perancangan

3.2.5. LCD (*Liquid Crystal Display*)

Pada sistem yang direncanakan digunakan LCD jenis TM162ABC yang mempunyai persamaan dengan jenis TM1632 merupakan LCD dua baris dengan tiap barisnya terdiri dari 16 karakter.

LCD ini membutuhkan 3 sinyal kontrol, R/W (*read/write*) untuk membaca data kemudian menulisnya, E (*Enable*) yang merupakan sinyal untuk meng-enable-kan dan RS (*Register Select*) untuk memilih register yang diakses. LCD TM162ABC memiliki 2 register yaitu register data dan register instruksi.

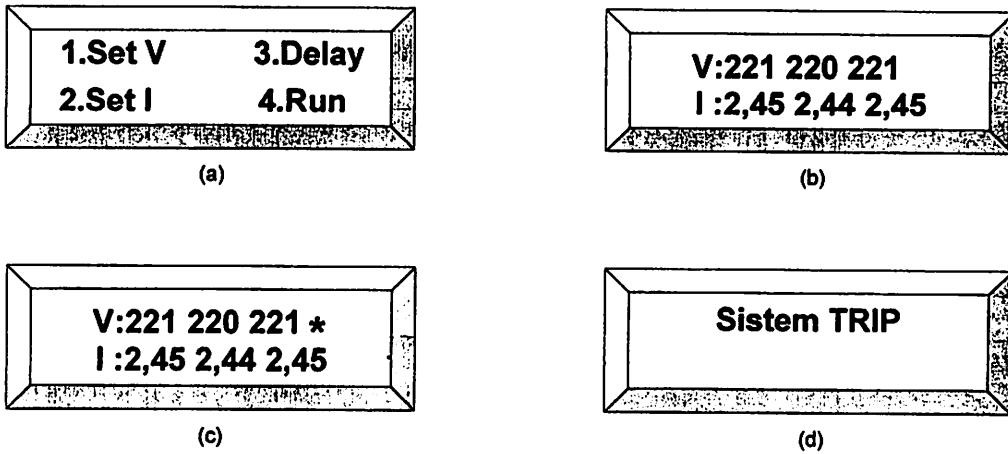
Pin R/W dihubungkan ke *ground* atau selalu berlogika 0 karena dalam perancangan LCD ini hanya selalu dalam operasi tulis dan pin RS dihubungkan ke pin A0 sistem mikrokontroler. Pengaktifan LCD ini selanjutnya tergantung pada pin E. Dimana pin E ini tergantung dari adres dekoder dan perintah write mikrokontroler. Rangkaian LCD seperti terlihat dalam Gambar 3.8.



Gambar 3.8 Rangkaian LCD

LCD direncanakan untuk menampilkan :

- Menu settingan tegangan, arus, delay dan run.
- Hasil monitoring tegangan dan arus tiap-tiap fasa setiap saat.
- Status sistem apakah normal atau ada gangguan. Apabila muncul tanda * yang berkedip berarti sistem terjadi gangguan, dengan setiap kedipan berarti gangguan sudah berlangsung 1 detik. Apabila sistem normal maka tanda * hilang dari tampilan.
- Tanda “ SISTEM TRIP” apabila gangguan berlangsung melebihi setting delay time.



Gambar 3.9 Tampilan Layar LCD

Sumber : Perancangan

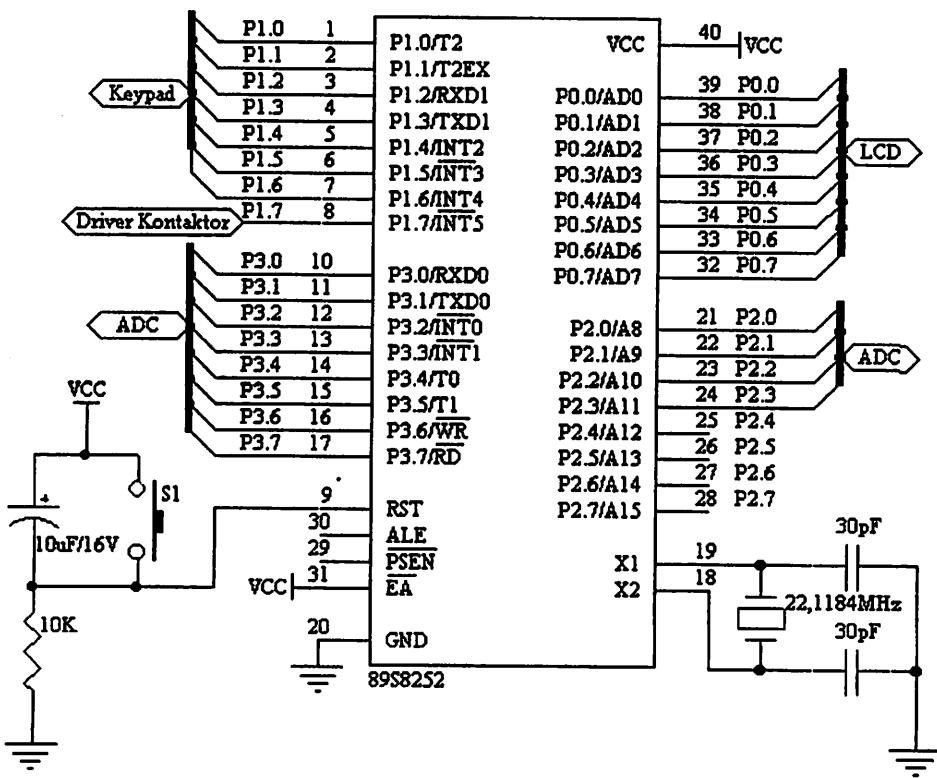
3.2.6. Mikrokontroller AT89S8252

Pembagian port-port mikrokontroller direncanakan seperti pada gambar 3.10 sebagai berikut :

- Port 0 (P0.0 – P0.7) difungsikan untuk mengirim sinyal ke LCD.
- Port 1 (P1.0 – P1.6) difungsikan untuk menerima input dari keypad.
- Port 1 (P1.7) difungsikan untuk mengeluarkan trigger ke rangkaian driver kontaktor apabila terjadi gangguan dengan waktu melebihi *time delay*.
- Port 2 (P2.0 – P2.3) difungsikan untuk pengalamatan ADC.
- Port 3 (P3.0 – P3.7) difungsikan untuk membaca/menerima sampling dari ADC.
- Pin 9 sebagai Reset jika sistem mengalami *trouble / hang* atau mereset jika sistem mengalami trip setelah terjadi gangguan. Kapasitor 10 uF dan resistor 10 KΩ dipakai untuk membentuk rangkaian *reset*, dengan adanya rangkaian

reset ini AT89S8252 otomatis di-*reset* begitu rangkaian menerima catu daya atau dapat juga di-*reset* secara manual dengan jalan menekan tombol Reset.

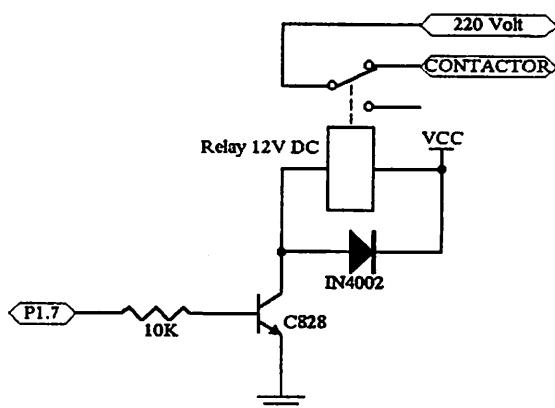
- Pin 18 dan 19 untuk menerima clock dari osilator X-tal.. Xtal dengan frekuensi 22,1184 MHz dan 2 kapasitor 30 pF dipakai untuk membentuk rangkaian oscilator /clock .
- Pin 20 dihubungkan ke ground.
- Pin 31 dan 40 dihubungkan ke VCC + 5 Volt.
- Pin ALE (Address Latch Enable) dan pin PSEN (Program Store Enable) yang berfungsi untuk membaca program memori luar tidak dipakai karena untuk menjalankan program yang dibuat ini tidak diperlukan memori luar cukup menggunakan memori internal yang tersedia pada AT89S8252.
- Pin EA (External Access Enable) dihubungkan dengan VCC karena menggunakan memori internal.



Gambar 3.10 Rangkaian Mikrokontroller
Sumber : Perancangan

3.2.7. Driver Kontaktor

Rangkaian driver kontaktor digunakan untuk meng-off-kan kontaktor jika terjadi gangguan dengan lama gangguan yang melewati *time delay*.



Gambar 3.11 Rangkaian Driver Kontaktor
Sumber : Perancangan

Perhitungan untuk menentukan resistor beban adalah sebagai berikut :

Data data yang diketahui (datasheet) $h_{fe} = 130$, $V_{BB} = 5$ volt, $I_C = 50\text{mA}$,

$V_{BE} = 0,8$ volt

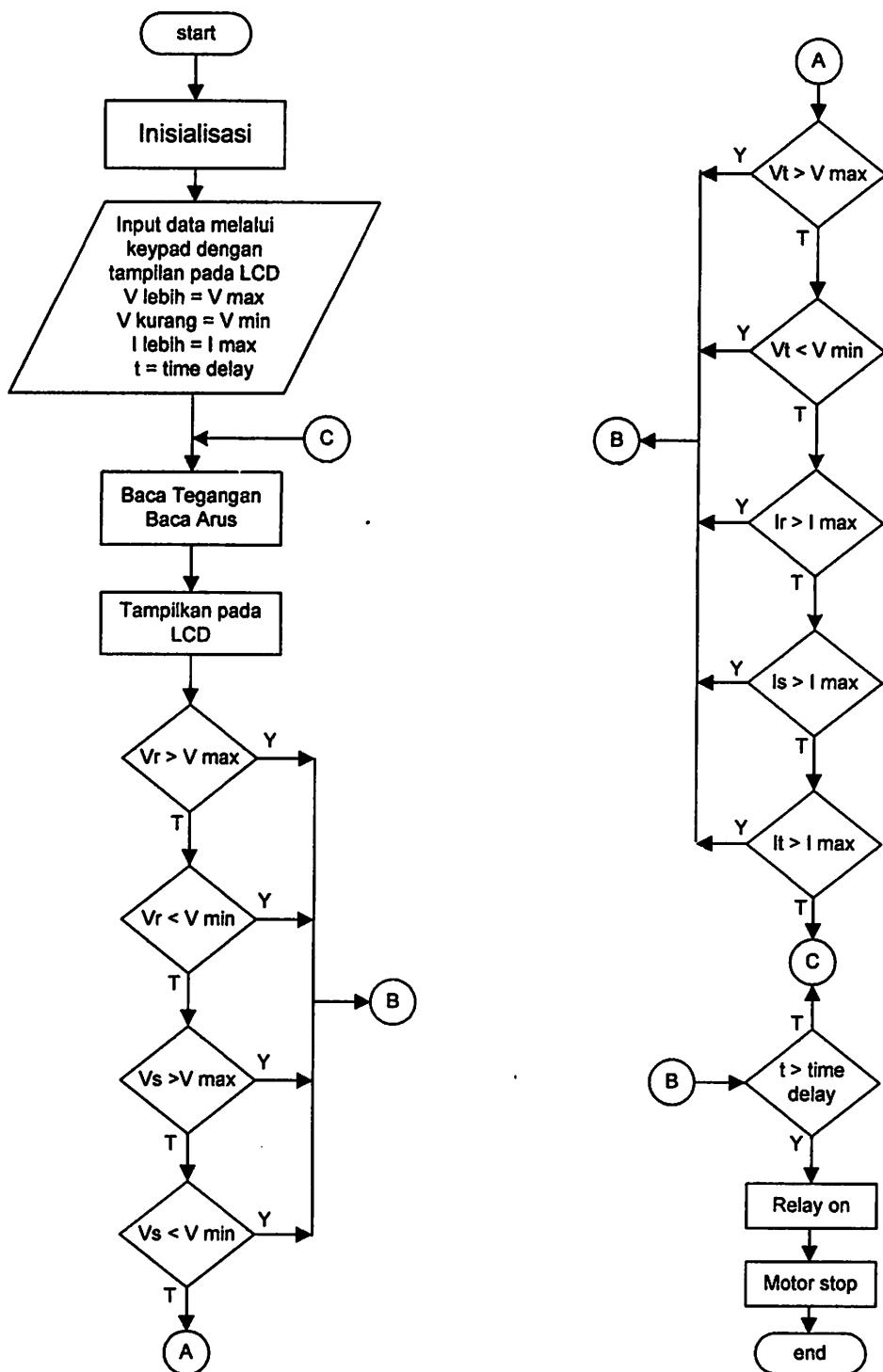
$$I_B = \frac{I_C}{h_{fe}} = \frac{50 \cdot 10^{-3}}{130} = 3,846 \cdot 10^{-4} \text{ A}$$

$$R_B = \frac{V_{BB} - V_{BE}}{I_B} = \frac{5 - 0,8}{3,846 \cdot 10^{-4}} = 10920 \Omega$$

Maka dipilih resistor $10 \text{ K}\Omega$

3.3. Perencanaan *Software* (Perangkat Lunak)

Perangkat lunak dirancang agar alat dapat bekerja sesuai dengan perencanaan sistem secara keseluruhan. Bahasa pemrograman yang digunakan adalah bahasa *Assembly Mikrokontroller Standard MCS-51*. *Software* program terdapat pada lampiran, sedangkan *flowchart* untuk pembuatan *software* dapat dilihat pada gambar dibawah ini :



Gambar 3.12 Flow Chart Perancangan Perangkat Lunak
Sumber : Perancangan

BAB IV

PENGUJIAN ALAT DAN ANALISIS

4.1. Umum

Secara umum, pengujian ini bertujuan untuk mengetahui apakah piranti dapat bekerja sesuai dengan spesifikasi perencanaan yang telah ditetapkan. Pengujian piranti ini dilakukan dalam dua tahap. Pertama, dilakukan pengujian terhadap perangkat keras pada masing-masing blok rangkaian penyusun sistem antara lain rangkaian sensor-ADC, keypad, LCD, mikrokontroller, rangkaian driver kontaktor. Pengujian kedua dilakukan pada sistem secara keseluruhan dengan mengintegrasikan perangkat keras dan perangkat lunak untuk mengetahui unjuk kerja alat yang dibuat.

4.2. Pengujian Sensor Arus dan ADC

4.2.1. Tujuan

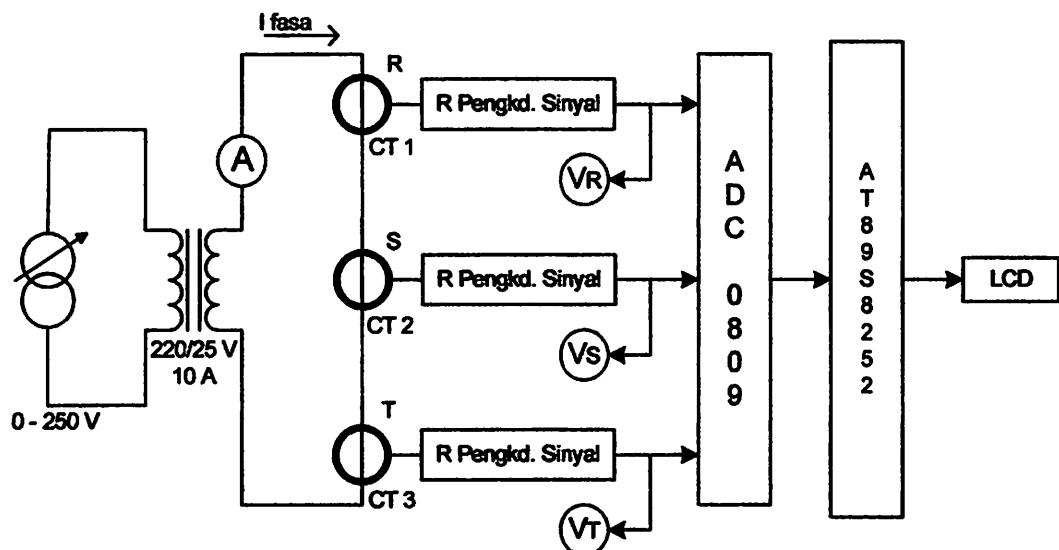
Tujuan dari pengujian rangkaian sensor arus dan ADC adalah untuk mengetahui level tegangan keluaran sensor arus dan hasil konversi ADC apabila arus fasa yang disensor mengalami perubahan.

4.2.2. Peralatan yang Digunakan

1. Beban AC variabel dalam hal ini menggunakan simulasi dengan autotrafo dan trafo 220/25 V 10 A.
2. Voltmeter digital merk UniVolt DT-860 Japan.
3. Tang amper digital merk CONSTANT Australia.
4. Software untuk menampilkan hasil konversi ADC pada LCD.

4.2.3 Prosedur Pengujian

1. Rangkaian dibuat seperti Gambar 4.1.
2. Memasang simulasi beban variabel.
3. Mengamati perubahan arus fasa dengan menggunakan tang amper.
4. Mengamati tegangan keluaran rangkaian pada sensor arus (pada rangkaian pengkondisi sinyal) dengan voltmeter, dan mengamati hasil konversi ADC pada LCD.
5. Mengulangi langkah 3 untuk simulasi beban yang berbeda.



Gambar 4.1 Diagram Blok Pengujian Rangkaian Sensor Arus dan ADC

4.2.4. Hasil Pengujian

Tabel 4.1 Hasil Pengujian Rangkaian Sensor Arus dan ADC

I FASA (Amp AC)	VR (Volt DC)	V _S (Volt DC)	V _T (Volt DC)	Konversi ADC (desimal)		
				R	S	T
0.35	0	0	0	0	0	0
0.75	0.16	0.16	0.16	8	8	8
0.84	0.26	0.26	0.26	12	12	12
1	0.39	0.39	0.39	20	20	20
1.2	0.59	0.59	0.59	30	30	30
1.35	0.73	0.73	0.73	37	37	37
2.03	1.34	1.3	1.34	68	67	68
2.53	1.73	1.69	1.73	88	87	88
3.05	2.1	1.95	2.1	107	106	107
3.4	2.27	2.2	2.27	116	115	116
4.01	2.55	2.4	2.55	130	129	130
4.51	2.81	2.74	2.81	143	142	143
5	3.1	2.95	3.1	158	157	158
5.58	3.25	3.25	3.25	166	166	166
6.13	3.43	3.4	3.43	175	175	175
6.6	3.67	3.67	3.67	187	187	187
7.16	3.92	3.92	3.92	200	200	200
7.69	4	4	4	204	204	204
8.08	4.14	4.14	4.14	211	211	211
8.6	4.22	4.22	4.22	215	215	215
9.07	4.47	4.47	4.47	228	228	228
9.54	4.5	4.5	4.5	230	230	230
10	4.7	4.7	4.7	240	240	240

4.2.5. Analisis Hasil Pengujian

Dapat dilihat pada tabel di atas bahwa sensor arus menghasilkan tegangan DC 0 – 4,7 Volt mendekati dengan perencanaan semula 0 – 5 volt, sisa 0,3 volt dimaksudkan untuk kalibrasi. Berdasarkan tabel di atas dapat disimpulkan bahwa semakin besar arus fasa yang melewati trafo arus maka makin besar pula tegangan

Tab. 4.3.4: Einfluss der Anzahl der ARA

Wertesatzung der Betriebsergebnisse für die Anwendung des Volumenverfahrens

Kontrolliert (SAC) (gezähmelt)			T	S	R	V	AV	RV	Avg. AV
			Wertesatzung						
0	0	0	0	0	0	0	0	0	0
8	8	8	0,18	0,18	0,18	0,18	0,18	0,18	0,18
12	12	12	0,28	0,28	0,28	0,28	0,28	0,28	0,28
20	20	20	0,30	0,30	0,30	0,30	0,30	0,30	0,30
30	30	30	0,30	0,30	0,30	0,30	0,30	0,30	0,30
32	32	32	0,31	0,31	0,31	0,31	0,31	0,31	0,31
38	38	38	0,31	0,31	0,31	0,31	0,31	0,31	0,31
48	48	48	0,31	0,31	0,31	0,31	0,31	0,31	0,31
58	58	58	0,31	0,31	0,31	0,31	0,31	0,31	0,31
68	68	68	0,31	0,31	0,31	0,31	0,31	0,31	0,31
78	78	78	0,31	0,31	0,31	0,31	0,31	0,31	0,31
102	102	102	1,08	1,08	1,08	1,08	1,08	1,08	1,08
116	116	116	1,18	1,18	1,18	1,18	1,18	1,18	1,18
130	130	130	1,28	1,28	1,28	1,28	1,28	1,28	1,28
143	143	143	1,45	1,45	1,45	1,45	1,45	1,45	1,45
153	153	153	1,45	1,45	1,45	1,45	1,45	1,45	1,45
168	168	168	1,48	1,48	1,48	1,48	1,48	1,48	1,48
188	188	188	1,48	1,48	1,48	1,48	1,48	1,48	1,48
208	208	208	1,52	1,52	1,52	1,52	1,52	1,52	1,52
228	228	228	1,52	1,52	1,52	1,52	1,52	1,52	1,52
248	248	248	1,52	1,52	1,52	1,52	1,52	1,52	1,52
268	268	268	1,52	1,52	1,52	1,52	1,52	1,52	1,52
288	288	288	1,52	1,52	1,52	1,52	1,52	1,52	1,52
308	308	308	1,52	1,52	1,52	1,52	1,52	1,52	1,52
328	328	328	1,52	1,52	1,52	1,52	1,52	1,52	1,52
348	348	348	1,52	1,52	1,52	1,52	1,52	1,52	1,52
368	368	368	1,52	1,52	1,52	1,52	1,52	1,52	1,52
388	388	388	1,52	1,52	1,52	1,52	1,52	1,52	1,52
408	408	408	1,52	1,52	1,52	1,52	1,52	1,52	1,52

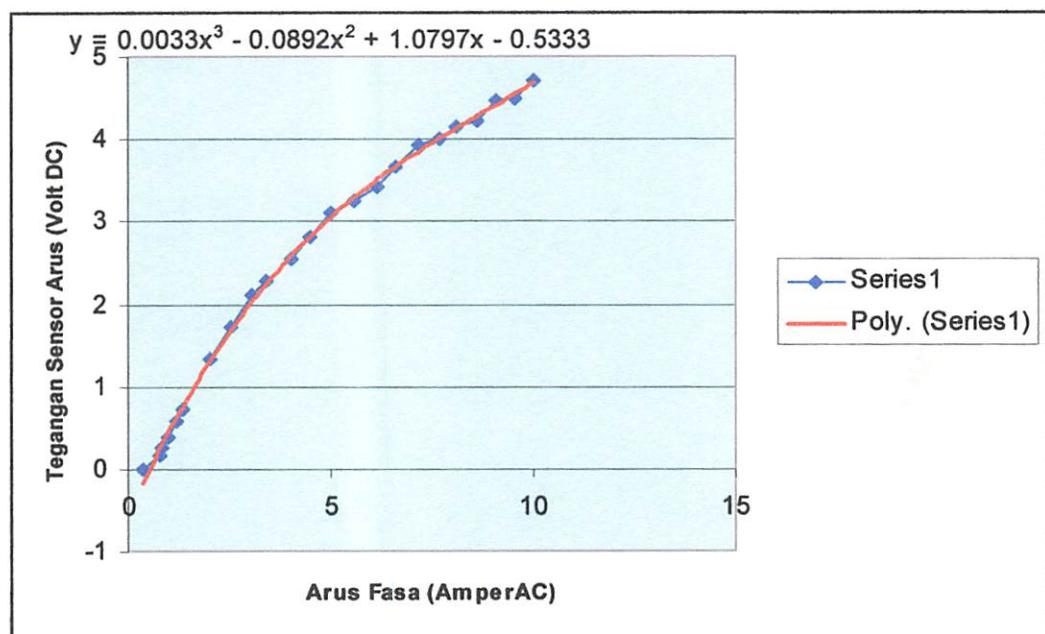
Tab. 4.3.5: Einfluss der Anzahl der ARA

Diese Tabelle zeigt jedoch die direkte Abhängigkeit zwischen dem Volumenverfahren und dem Ergebnis der Kostenabschätzung. Die Kostenabschätzung ist auf die Anzahl der ARA abgestimmt. Da die Kostenabschätzung nur die Kosten für die Produktion berücksichtigt, führt dies zu einer Unterabschätzung der Kosten. Dies ist jedoch nicht der Fall, wenn die Kostenabschätzung die Kosten für die Produktion sowie die Kosten für die Verwaltung berücksichtigt. In diesem Fall führt die Anzahl der ARA zu einem höheren Ergebnis der Kostenabschätzung.

yang timbul pada sensor arus dan mengakibatkan ADC menghasilkan konversi yang lebih tinggi stepnya.

Dengan mengambil data perubahan arus fasa yang diikuti dengan perubahan tegangan pada sensor arus (diwakili oleh fasa R) maka didapatkan grafik 4.1.

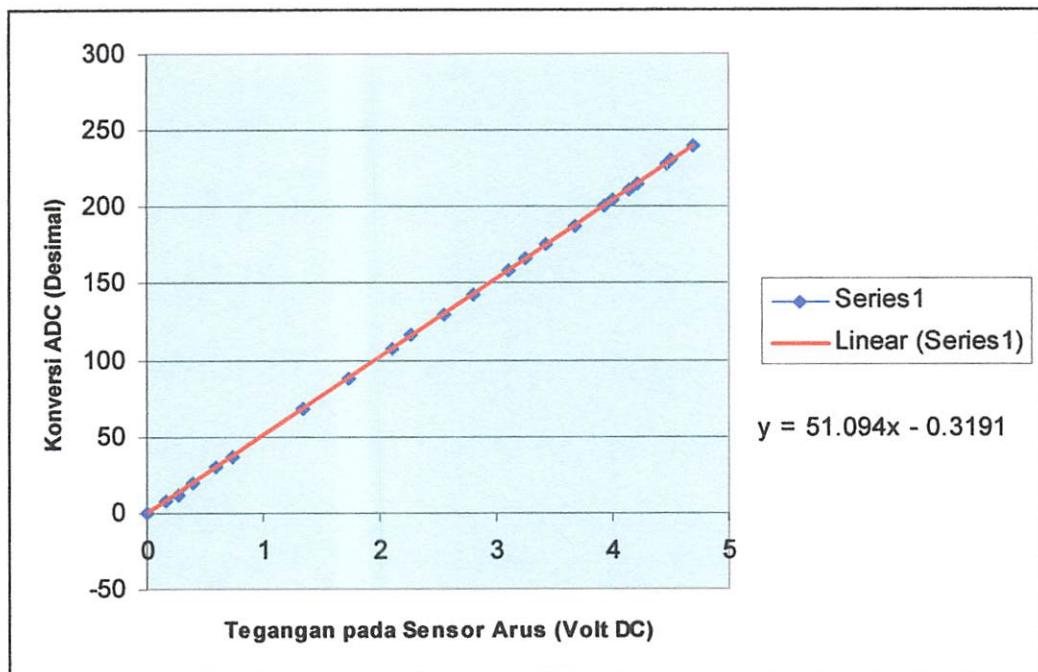
Grafik 4.1 Hubungan Antara Arus Fasa dan Tegangan pada Sensor Arus



Dari grafik di atas, dengan menggunakan fungsi grafik pada MS Excel melalui pendekatan polinomial orde 3 maka didapat persamaan grafik $y = 0,0033x^3 - 0,0892x^2 + 1,0797x - 0,5333$, dimana y = tegangan keluaran pada sensor arus, dan x = arus fasa yang disensor.

Dengan mengambil data perubahan tegangan pada sensor arus (diwakili fasa R) yang diikuti dengan perubahan konversi ADC (diwakili oleh fasa R) maka didapatkan grafik 4.2.

Grafik 4.2 Hubungan Antara Tegangan Sensor Arus dan Konversi ADC



Dari grafik di atas, dengan menggunakan fungsi grafik pada MS Excel melalui pendekatan linier maka didapat persamaan grafik $y = 51,094x - 0,3191$ dimana y = hasil konversi ADC, dan x = tegangan pada sensor arus.

4.3. Pengujian Sensor Tegangan dan ADC

4.3.1. Tujuan

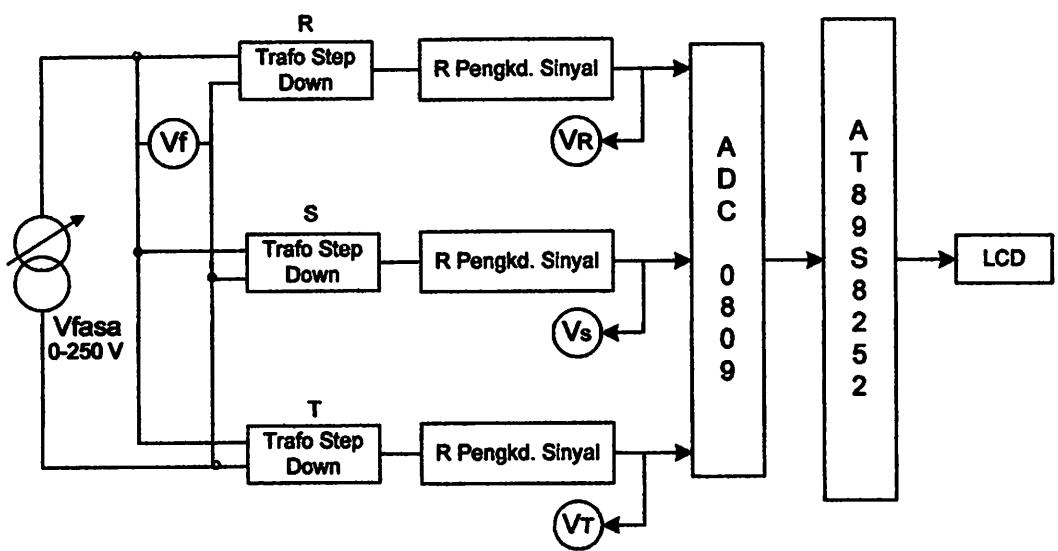
Tujuan dari pengujian rangkaian sensor tegangan adalah untuk mengetahui level tegangan keluaran pada sensor tegangan dan hasil konversi ADC apabila tegangan fasa yang disensor mengalami perubahan.

4.3.2. Peralatan yang Digunakan

1. Autotrafo 0 – 250 V.
2. Dua buah Voltmeter digital, merk UniVolt DT-860 Japan dan merk Kaiser M890D.
3. Software untuk menampilkan hasil konversi ADC pada LCD.

4.3.3. Prosedur Pengujian

1. Rangkaian dibuat seperti Gambar 4.2.
2. Memasang simulasi beban variabel dengan menggunakan autotrafo.
3. Mengamati perubahan tegangan fasa dengan menggunakan voltmeter.
4. Mengamati tegangan keluaran rangkaian sensor tegangan dengan volt meter, dan perubahan konversi ADC pada LCD.
5. Mengulangi langkah 3 untuk tegangan fasa yang berbeda.



Gambar 4.2 Diagram Blok Pengujian Rangkaian Sensor Tegangan dan ADC

4.3.4. Hasil Pengujian Sensor Tegangan dan ADC

Tabel 4.2 Hasil Pengujian Rangkaian Sensor Tegangan dan ADC

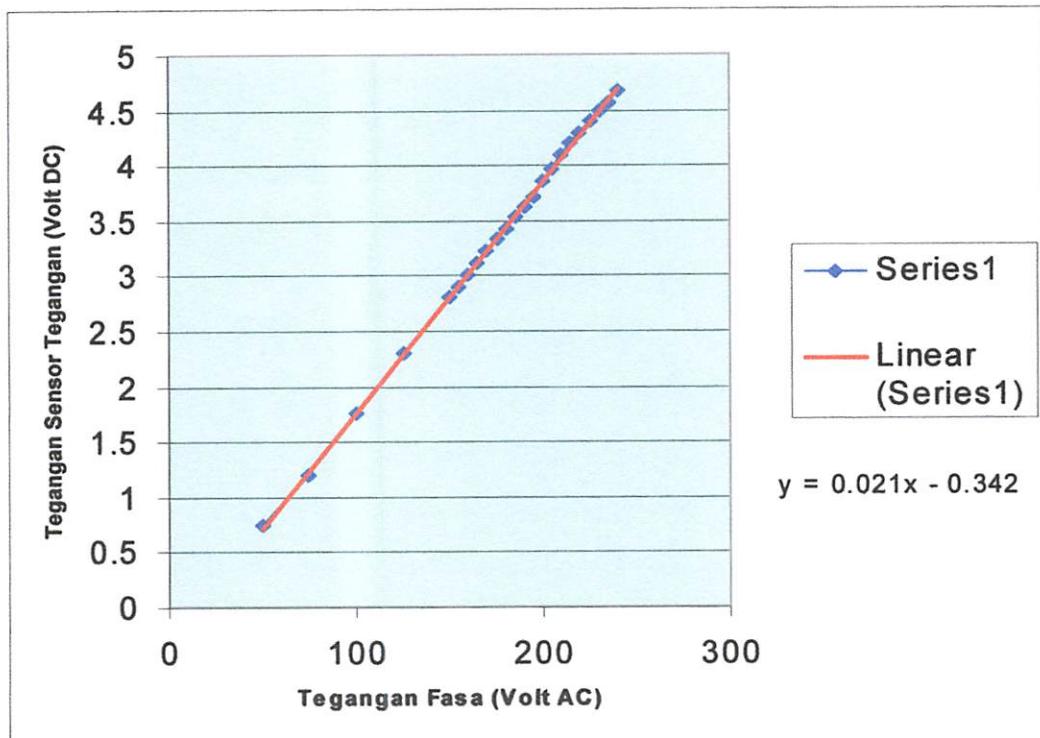
V fasa (volt AC)	V _R (volt DC)	V _S (volt DC)	V _T (volt DC)	Konversi ADC (Desimal)		
				R	S	T
50	0.74	0.74	0.78	37	37	38
75	1.2	1.2	1.22	61	61	62
100	1.76	1.76	1.78	89	89	90
125	2.3	2.3	2.34	117	117	118
150	2.8	2.8	2.83	143	143	144
155	2.9	2.9	2.94	147	147	149
160	3	3	3.04	153	153	155
165	3.12	3.12	3.15	159	159	159
170	3.22	3.22	3.25	164	164	165
175	3.33	3.33	3.35	169	169	170
180	3.43	3.43	3.45	174	174	175
185	3.53	3.53	3.55	180	180	181
190	3.62	3.62	3.65	184	184	186
195	3.71	3.71	3.74	189	189	190
200	3.86	3.86	3.89	196	196	198
205	3.97	3.97	3.99	202	202	203
210	4.1	4.1	4.12	209	209	210
215	4.2	4.2	4.22	214	214	215
220	4.29	4.29	4.31	218	218	219
225	4.4	4.4	4.42	224	224	225
230	4.5	4.5	4.52	229	229	230
235	4.57	4.57	4.59	233	233	234
240	4.67	4.67	4.69	238	238	239

4.3.5. Analisis Hasil Pengujian

Dapat dilihat pada tabel di atas bahwa sensor tegangan menghasilkan tegangan DC 0 – 4,67 Volt, untuk tegangan fasa 0 – 240 Volt AC, mendekati perencanaan semula yaitu 0 – 5 Volt. Berdasarkan tabel di atas dapat disimpulkan bahwa semakin besar tegangan fasa yang disensor maka makin besar pula tegangan yang timbul pada sensor tegangan dan mengakibatkan ADC menghasilkan konversi yang lebih tinggi stepnya.

Dengan mengambil data perubahan tegangan fasa yang diikuti dengan perubahan tegangan sensor (diwakili oleh fasa R) maka didapatkan grafik 4.3.

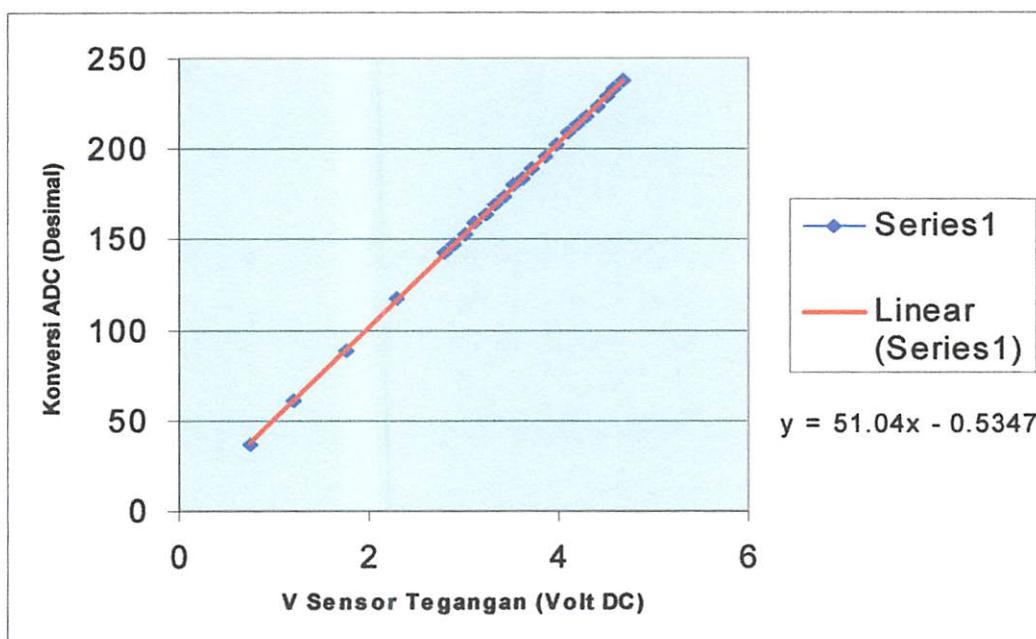
Grafik 4.3 Hubungan Antara Tegangan Fasa dan Tegangan pada Sensor Tegangan



Dari grafik di atas, dengan menggunakan fungsi grafik pada MS Excel melalui pendekatan linier maka didapat persamaan grafik $y = 0,021x - 0,342$ dimana y = tegangan keluaran pada sensor tegangan, dan x = tegangan fasa yang disensor.

Dengan mengambil data perubahan tegangan sensor (diwakili fasa R) yang diikuti dengan perubahan konversi ADC (diwakili oleh fasa R) maka didapatkan grafik 4.4.

Grafik 4.4 Hubungan Antara Tegangan pada Sensor Tegangan dan Konversi ADC



Dari grafik di atas, dengan menggunakan fungsi grafik pada MS Excel melalui pendekatan linier maka didapat persamaan grafik $y = 51,04x - 0,5347$ dimana y = hasil konversi ADC dan x = tegangan keluaran pada sensor tegangan

4.4. Pengujian Sistem Mikrokontroler, LCD dan Keypad

4.4.1. Tujuan

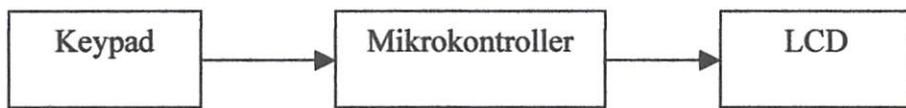
Untuk mengetahui unjuk kerja dari mikrokontroler, LCD dan keypad .

4.4.2. Peralatan yang Digunakan

1. Mikrokontroller.
2. LCD.
3. Keypad 4x3.

4.4.3. Prosedur Pengujian

1. Membuat rangkaian seperti pada gambar 4.3.
2. Menjalankan program yang telah dibuat.
3. Mengamati tampilan LCD saat salah satu tombol keypad ditekan.



Gambar 4.3 Blok Diagram Pengujian Mikrokontroler, Keypad dan LCD

4.4.4. Hasil Pengujian

Tabel 4.3 Hasil Pengujian Mikrokontroler, LCD dan Keypad

Tombol yg ditekan	Tampilan LCD
0	0
1	1
2	2
3	3
4	4
5	5
6	6

7	7
8	8
9	9
*	*
#	#

4.4.5. Analisis Hasil Pengujian

Dari hasil pengujian dapat disimpulkan bahwa yang ditampilkan LCD sesuai dengan tombol pada keypad yang ditekan.

4.5. Pengujian Rangkaian Driver Kontaktor

4.5.1. Tujuan

Mengetahui unjuk kerja rangkaian driver kontaktor saat sebelum atau sesudah diberi pemicuan tegangan 5 volt DC.

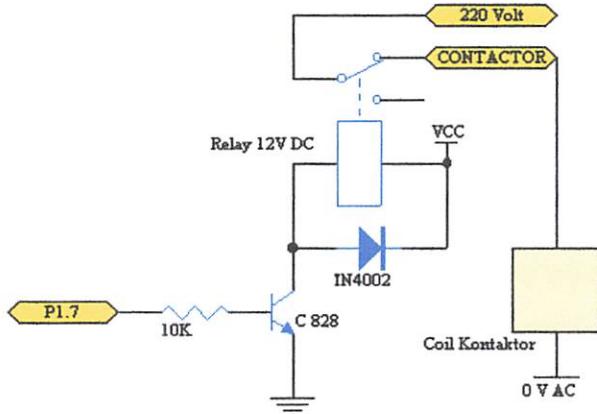
4.5.2. Peralatan yang digunakan

1. Magnetik Kontaktor.
2. Catu daya 12 volt dan 5 volt DC.
3. Catu daya 220 volt AC.

4.5.3. Prosedur Pengujian

1. Menghubungkan catu daya 12 V pada rangkaian relay.
2. Menghubungkan *common* relay dengan tegangan 220 volt AC.
3. Menghubungkan kaki NC (Normally Close) relay dengan magnetik kontaktor
4. Mengaktifkan catu daya 12 volt dan memicu rangkaian driver kontaktor dengan tegangan 5 volt.

5. Mengamati rangkaian sebelum dan sesudah pemicuan.



Gambar 4.4 Pengujian Rangkaian Driver Kontaktor

4.5.4. Hasil Pengujian

Tabel 4.4 Hasil Pengujian Rangkaian Driver Kontaktor

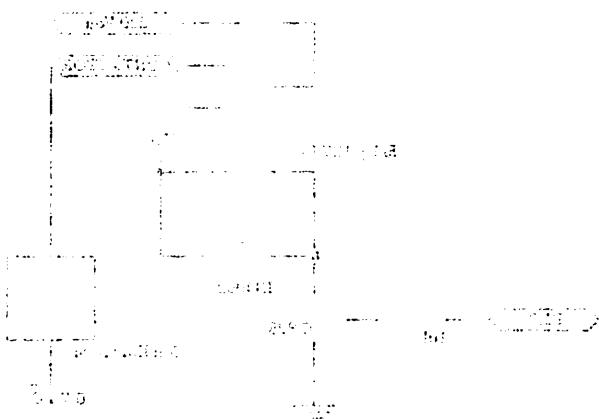
Sebelum Pemicuan	Sesudah Pemicuan
Relay off (NC menutup)	Relay on (NC membuka)
Kontaktor on	Kontaktor off

4.5.5. Analisis Hasil Pengujian

Berdasarkan hasil pengujian rangkaian driver kontaktor, maka dapat diambil kesimpulan bahwa pada saat terjadi pemicuan, transistor akan bekerja sebagai saklar yang tertutup sehingga relay bekerja karena mendapat tegangan 12 volt.

Jika pemicuan dihilangkan maka transistor bekerja sebagai saklar yang terbuka, sehingga relay off.

5. Mordavianskiy tatarskii zemelnyi soviet na sovremennoy posadke



Однодневная выдача земли татарам в селе Красногорье

5.5.1. Характеристика

Типы земельных участков для земельной выдачи

Земельный участок	Земельное право
Комплект	Район (МС, макары)
Комплект	Район (МС, макары)

5.5.2. Материалы

Изучение земельных участков для земельной выдачи показало, что земельные участки для земельной выдачи в основном расположены в селах и деревнях, где земельные участки для земельной выдачи находятся в частной собственности. Важно отметить, что земельные участки для земельной выдачи в основном расположены в селах и деревнях, где земельные участки для земельной выдачи находятся в частной собственности.

На земельных участках для земельной выдачи

4.6. Pengujian Ketelitian Alat dalam Pengukuran Arus

4.6.1. Tujuan

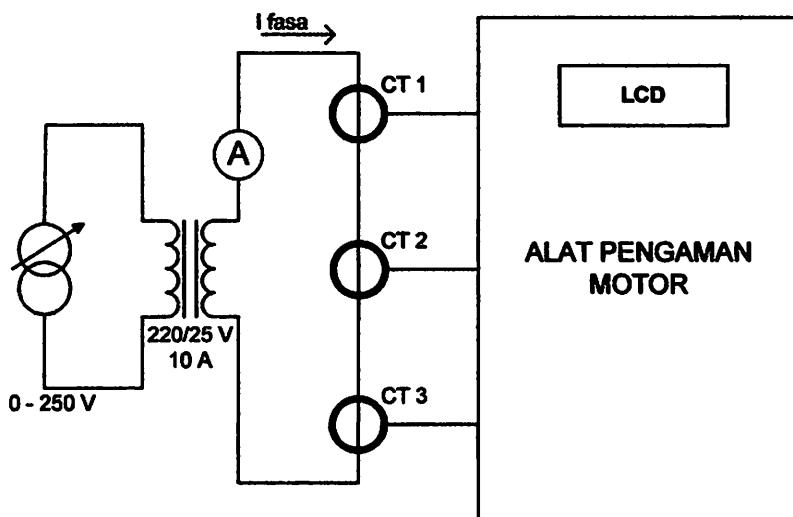
Untuk mengetahui tingkat ketelitian alat dalam pengukuran arus.

4.6.2. Alat yang Digunakan

1. Beban AC variabel dalam hal ini menggunakan simulasi dengan autotrafo dan trafo 220/25 V 10 A.
2. Tang Amper digital merk CONSTANT Australia.

4.6.3. Prosedur Pengujian

1. Membuat Rangkaian seperti gambar 4.5.
2. Mengatur beban Variabel.
3. Mengamati perubahan arus fasa dengan tang amper .
4. Mengamati hasil tampilan LCD.
5. Mengulangi langkah 3 untuk besar arus yang berbeda.



Gambar 4.5 Rangkaian Pengujian Ketelitian Pengukuran Arus

4.6.4. Hasil Pengujian

Tabel 4.5 Pengujian Ketelitian Alat dalam Pengukuran Arus

Pembacaan Alat Ukur Arus (Amper)			Tampilan LCD (Amper)			Error (%)
I _R	I _S	I _T	I _R	I _S	I _T	
1,00	1,00	1,00	0,87	0,87	0,87	13
1,51	1,51	1,51	1,41	1,41	1,41	10
2,03	2,03	2,03	2,00	2,00	2,00	1,4
2,51	2,51	2,51	2,51	2,51	2,51	0
3,02	3,02	3,02	3,02	3,02	3,02	0
3,51	3,51	3,51	3,52	3,52	3,52	0,28
4,01	4,01	4,01	3,99	3,99	3,99	0,49
4,48	4,48	4,48	4,47	4,47	4,47	0,22
5,01	5,01	5,01	4,99	4,99	4,99	0,4
5,51	5,51	5,51	5,50	5,50	5,50	0,18
6,06	6,06	6,06	6,04	6,04	6,04	0,33
6,52	6,52	6,52	6,52	6,52	6,52	0
7,00	7,00	7,00	7,03	7,03	7,03	0,43
7,53	7,53	7,53	7,57	7,57	7,57	0,53
8,00	8,00	8,00	8,08	8,08	8,08	0,99
9,05	9,05	9,05	9,16	9,16	9,16	1,2
Rata – Rata Error						1,84

4.6.5. Analisis Hasil Pengujian

Berdasarkan tabel di atas dapat disimpulkan bahwa kesalahan pengukuran arus alat pengaman adalah 1,84 %. Error tertinggi terletak pada pengukuran arus 1,00 dan 1,51 amper, hal ini karena arus tersebut masih cukup kuat untuk membangkitkan tegangan pada belitan sekunder trafo arus.

/

Tabelle 2.4: Bedeutung der Leistungsfähigkeit verschiedener Antriebe

Antrieb Typ (A)	Technische Daten (A)				Performance (A)		
	A	B	C	D	E	F	G
1	58,0	58,0	58,0	60,1	60,1	60,1	60,1
01	44,1	44,1	44,1	45,1	45,1	45,1	46,1
4,1	90,2	90,2	90,2	90,2	90,2	90,2	90,2
0	12,5	12,5	12,5	12,5	12,5	12,5	12,5
0	30,5	30,5	30,5	30,5	30,5	30,5	30,5
82,0	55,5	55,5	55,5	56,5	56,5	56,5	56,5
84,0	60,3	60,3	60,3	60,3	60,3	60,3	60,3
52,0	74,4	74,4	74,4	74,4	74,4	74,4	74,4
4,0	99,4	99,4	99,4	99,4	10,2	10,2	10,2
81,0	92,2	92,2	92,2	92,2	12,2	12,2	12,2
63,0	40,0	40,0	40,0	40,0	60,0	60,0	60,0
0	52,0	52,0	52,0	52,0	62,0	62,0	62,0
64,0	60,5	60,5	60,5	60,5	60,5	60,5	60,5
65,0	72,5	72,5	72,5	72,5	62,5	62,5	62,5
99,0	80,8	80,8	80,8	80,8	90,8	90,8	90,8
5,1	8,8	8,8	8,8	8,8	20,0	20,0	20,0
48,1	Rahmengewicht				Antriebsleistung (A)		

Tabelle 2.5: Antriebsleistung (A)

Beobachtungstafel der Antriebsleistung (A) für verschiedene Antriebstechniken und deren Leistungsfähigkeit. Die Tabelle zeigt die Antriebsleistung (A) in W für verschiedene Antriebstechniken (A) und verschiedene Betriebsbedingungen (B). Die Antriebsleistung (A) ist proportional zur Antriebstechnik (A) und abhängig von den Betriebsbedingungen (B).

Antriebstechnik (A)	Betriebsbedingung (B)	Antriebsleistung (A)
1	1	58,0
1	2	60,1
01	1	44,1
01	2	45,1
4,1	1	90,2
4,1	2	90,2
0	1	12,5
0	2	12,5
82,0	1	55,5
82,0	2	56,5
84,0	1	60,3
84,0	2	60,3
52,0	1	74,4
52,0	2	74,4
4,0	1	99,4
4,0	2	99,4
81,0	1	92,2
81,0	2	92,2
63,0	1	40,0
63,0	2	60,0
0	1	52,0
0	2	62,0
64,0	1	60,5
64,0	2	60,5
65,0	1	72,5
65,0	2	62,5
99,0	1	80,8
99,0	2	90,8
5,1	1	8,8
5,1	2	20,0

4.7. Pengujian Ketelitian Alat dalam Pengukuran Tegangan

4.7.1. Tujuan

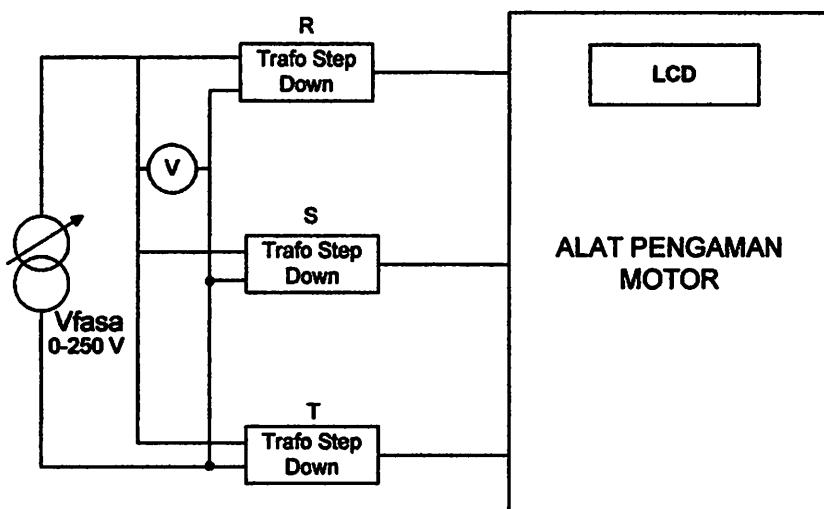
Untuk mengetahui tingkat ketelitian alat dalam pengukuran tegangan

4.7.2. Alat yang Digunakan

1. Sumber tegangan variabel/autotrafo.
2. Voltmeter digital merk UniVolt Japan DT-860.

4.7.3. Prosedur Pengujian

1. Membuat Rangkaian seperti gambar 4.6.
2. Mengatur tegangan fasa variabel.
3. Mengamati perubahan tegangan fasa dengan voltmeter .
4. Mengamati hasil tampilan LCD.
5. Mengulangi langkah 3 untuk besar tegangan fasa yang berbeda.



Gambar 4.6 Pengujian Ketelitian Pengukuran Tegangan

4.2. Beobachtungskategorie A für die eine Funktionsweise festzustellen

4.2.1. Turbinen

Unter Beobachtungskategorie A für die eine Funktionsweise festzustellen ist die im Beobachtungsbereich festgestellte

4.2.2. Abluft und Drosselung

1. Stomper (einschließlich Schaltstellung)

2. Automatisches Drosselventil (VAV) mit ID 800

4.2.3. Prosessturmspeisung

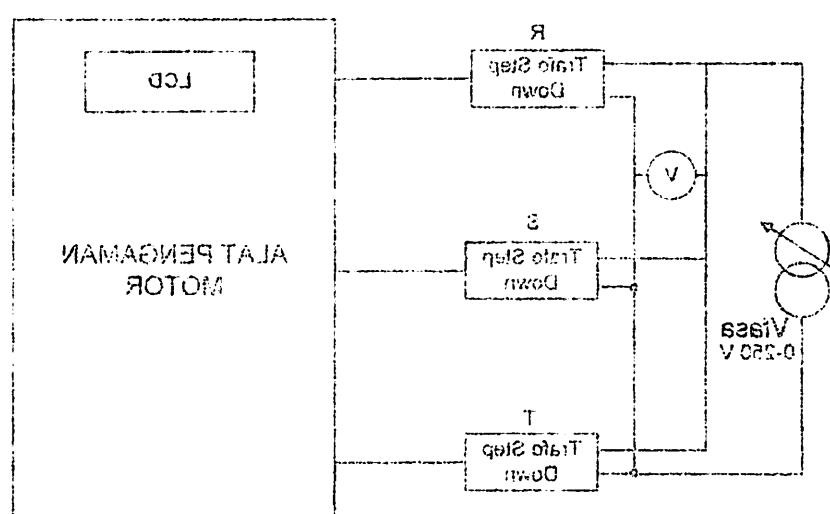
1. Wärmeraum Randspeisung schaltbare Stomper-F9

2. Wärmeraum Randspeisung feste Anzapfer

3. Wärmeraum Randspeisung feste Jalousien Anzapfer

4. Wärmeraum passiv (nicht FCI)

5. Wärmeraum passiv (nicht FCI)



Geispiel-F9 Beobachtungskategorie Funktionsweise festzustellen

4.7.4. Hasil Pengujian

Tabel 4. 6 Pengujian Ketelitian Alat dalam Pengukuran Tegangan

Pembacaan Alat Ukur Tegangan (Volt)			Tampilan LCD (Volt)			Error (%)
V _R	V _S	V _T	V _R	V _S	V _T	
150	150	150	146	146	146	2,66
160	160	160	158	158	158	1,26
170	170	170	167	167	167	1,76
180	180	180	178	178	178	1,12
190	190	190	188	188	188	1,06
200	200	200	202	202	202	1,00
220	220	220	224	224	224	1,82
230	230	230	235	235	235	2,17
235	235	235	240	240	240	2,12
Rata – Rata Error						1,66

4.7.5. Analisis Hasil Pengujian

Berdasarkan tabel di atas dapat disimpulkan bahwa kesalahan pengukuran arus alat pengaman dalam pengukuran tegangan adalah 1,66 % masih dalam batas toleransi.

A.5.4. Tabelle für Pauschalierungskoeffizienten bei der Planung von Tiefbohrungen

Tiefe (m)	Tiefbohrung Tief						Pauschalierung Acht Ufer					
	V	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁
120	120	120	120	120	120	120	120	120	120	120	120	120
130	128	128	128	128	128	128	128	128	128	128	128	128
140	136	136	136	136	136	136	136	136	136	136	136	136
150	144	144	144	144	144	144	144	144	144	144	144	144
160	152	152	152	152	152	152	152	152	152	152	152	152
170	160	160	160	160	160	160	160	160	160	160	160	160
180	168	168	168	168	168	168	168	168	168	168	168	168
190	176	176	176	176	176	176	176	176	176	176	176	176
200	184	184	184	184	184	184	184	184	184	184	184	184
210	192	192	192	192	192	192	192	192	192	192	192	192
220	200	200	200	200	200	200	200	200	200	200	200	200
230	208	208	208	208	208	208	208	208	208	208	208	208
240	216	216	216	216	216	216	216	216	216	216	216	216
250	224	224	224	224	224	224	224	224	224	224	224	224
260	232	232	232	232	232	232	232	232	232	232	232	232
270	240	240	240	240	240	240	240	240	240	240	240	240
280	248	248	248	248	248	248	248	248	248	248	248	248
290	256	256	256	256	256	256	256	256	256	256	256	256
300	264	264	264	264	264	264	264	264	264	264	264	264
310	272	272	272	272	272	272	272	272	272	272	272	272
320	280	280	280	280	280	280	280	280	280	280	280	280
330	288	288	288	288	288	288	288	288	288	288	288	288
340	296	296	296	296	296	296	296	296	296	296	296	296
350	304	304	304	304	304	304	304	304	304	304	304	304
360	312	312	312	312	312	312	312	312	312	312	312	312
370	320	320	320	320	320	320	320	320	320	320	320	320
380	328	328	328	328	328	328	328	328	328	328	328	328
390	336	336	336	336	336	336	336	336	336	336	336	336
400	344	344	344	344	344	344	344	344	344	344	344	344
Rück - Rücken Faktor												

A.5.5. Anlagen für Tiefbohrungen

Berechnungswert für die doppelte Distanzabteilung zwischen Kesselschiffen bei einer Tiefbohrung mit einer Tiefenlänge von 100 m und einer Tiefenbreite von 100 m ist gleichzeitig der Abstand zwischen den Schiffen.

folgendermaßen:

4.8. Pengujian Alat Terhadap Gangguan Arus Lebih

4.8.1. Tujuan

Untuk mengetahui respon alat pengaman saat terjadi gangguan arus lebih pada motor 3 fasa.

4.8.2. Alat yang Digunakan

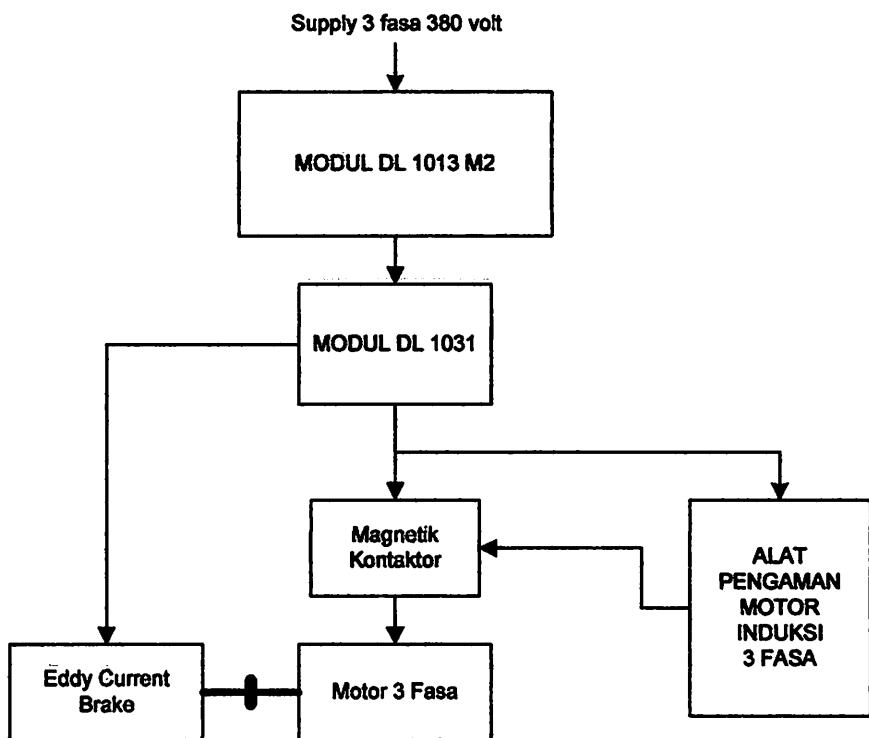
1. Motor 3 fasa 1,1 KW, 220/380 V, 2820 Rpm, 4,3 / 2,5 A, 50 Hz
Merk DELORENZO Italy.
2. Eddy Current Brake sebagai beban motor.
3. Satu set panel modul percobaan merk DELORENZO Italy
DL 1013 M2 / DL 1031.

4.8.3. Prosedur Pengujian

1. Membuat rangkaian seperti pada gambar 4.7.
2. Memasukkan setting arus lebih maksimal dan setting delay time pada mikrokontroller melalui keypad dengan tampilan LCD.
3. Menghidupkan dan mengatur besarnya tegangan power supply pada modul DL 1013 M2.
4. Menjalankan motor 3 fasa dengan menekan tombol ON pada kontaktor.
5. Mengatur arus medan pada Eddy Current Brake sehingga beban pada motor bertambah.
6. Mengamati kenaikan arus motor pada LCD dan mengamati munculnya tanda * yang berkedip pada LCD yang menandakan terjadi gangguan

dan menghitung lamanya gangguan dengan asumsi 1 kedipan * berarti gangguan sudah berlangsung selama 1 detik.

7. Menambah arus medan pada Eddy Current Brake dengan jalan memutar kenop supplay arus medan ke kanan yang terdapat pada modul DL 10113 M2, sampai arus motor melebihi setting arus maksimal dan akhirnya kontaktor trip.
8. Mengulangi mulai langkah ke-2 untuk setting arus lebih dan *delay time* yang berbeda .



Gambar 4.7 Blok Diagram Pengujian Alat Pengaman Motor Terhadap Gangguan Arus Lebih

4.8.4. Hasil Pengujian

Tabel 4.7 Hasil Pengujian Alat Terhadap Gangguan Arus Lebih

Setting Arus Max (A)	Arus Beban (A)			Setting Delay Time (detik)	Lama Gangguan (detik)	Status Kontaktor	Waktu Trip (detik)
	I_R	I_S	I_T				
2,30	2,32	2,32	2,32	3	4	Trip	4
2,27	2,30	2,30	2,30	3	4	Trip	4
2,27	2,30	2,30	2,30	4	3	Tidak Trip	-
2,20	2,22	2,22	2,22	4	5	Trip	5
2,05	2,10	2,10	2,10	4	5	Trip	5

4.8.5. Analisis Hasil Pengujian

Dari tabel di atas dapat disimpulkan bahwa apabila arus motor yang mengalir melebihi setting arus maksimal dan setting *delay time*, maka kontaktor akan trip, tetapi jika lama gangguan tidak melebihi setting *delay time*, maka sistem akan berjalan normal/tidak trip.

4.9. Pengujian Alat Terhadap Gangguan Tegangan Lebih

4.9.1. Tujuan

Untuk mengetahui respon alat pengaman saat terjadi tegangan lebih pada sistem atau pada motor.

4.9.2. Alat yang Digunakan

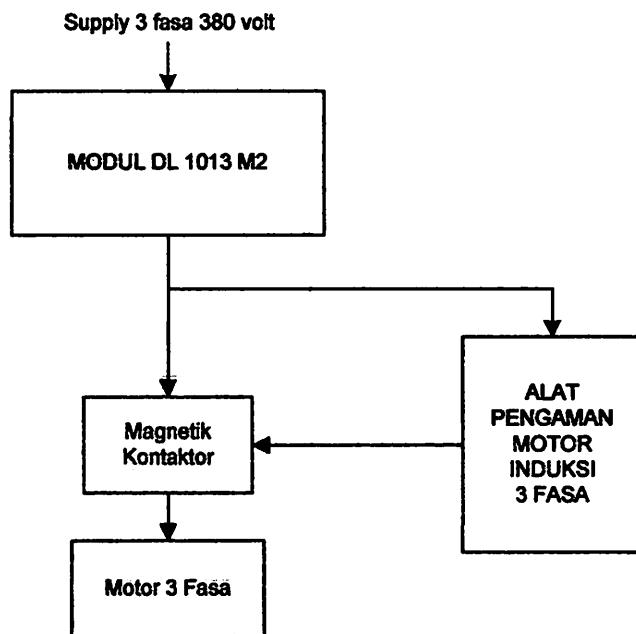
1. Satu set panel modul percobaan merk DELORENZO Italy DL 1013 M2.
2. Voltmeter Digital merk UniVolt Japan.
3. Motor 3 fasa 1,1 KW, 220/380 V, 2820 Rpm, 4,3 / 2,5 A, 50 Hz Merk DELORENZO Italy.

4.9.3. Prosedur Pengujian

1. Membuat rangkaian seperti gambar 4.8.
2. Memasukkan setting tegangan lebih maksimal dan setting delay time pada mikrokontroller melalui keypad dengan tampilan LCD.
3. Menghidupkan dan mengatur besarnya tegangan power supply pada modul DL 1013 M2 sebesar 220 Volt (tegangan fasa).
4. Menghidupkan motor dengan jalan menekan tombol ON pada magnetik kontaktor.
5. Menambah tegangan fasa dengan jalan memutar kenop pengatur tegangan ke kanan pada modul DL 1013 M2.
6. Mengamati kenaikan tegangan fasa pada LCD dan mengamati munculnya tanda * yang berkedip pada LCD yang menandakan terjadi

gangguan dan menghitung lamanya gangguan dengan asumsi 1 kedipan * berarti gangguan sudah berlangsung selama 1 detik.

7. Menambah tegangan fasa sampai melampaui setting tegangan maksimal sampai akhirnya kontaktor trip.
8. Mengulangi mulai langkah ke-2 untuk setting tegangan lebih dan *delay time* yang berbeda



Gambar 4.8 Blok Diagram Pengujian Alat Pengaman Motor Terhadap Gangguan Tegangan Lebih

4.9.4. Hasil Pengujian

Tabel 4.8 Hasil Pengujian Alat Terhadap Gangguan Tegangan Lebih

Setting Tegangan Max (Volt)	Tegangan Fasa (Volt)			Setting Delay Time (detik)	Lama Gangguan (detik)	Status Kontaktor	Waktu Trip
	V _R	V _S	V _T				
220	221	221	221	2	3	Trip	3
225	226	226	226	2	3	Trip	3
230	231	231	231	3	4	Trip	4
235	236	236	236	3	4	Trip	4
235	236	236	236	4	3	Tidak Trip	-

4.9.5. Analisis Hasil Pengujian

Dari hasil pengujian di atas dapat disimpulkan bahwa apabila tegangan fasa melebihi setting tegangan maksimal dan setting *delay time*, maka kontaktor akan trip, tetapi jika lama gangguan tidak melebihi setting *delay time*, maka sistem akan berjalan normal/tidak trip.

4.10. Pengujian Alat Terhadap Gangguan Tegangan Kurang

4.10.1. Tujuan

Untuk mengetahui respon alat pengaman saat terjadi gangguan tegangan kurang pada sistem atau pada motor.

4.10.2. Alat yang Digunakan

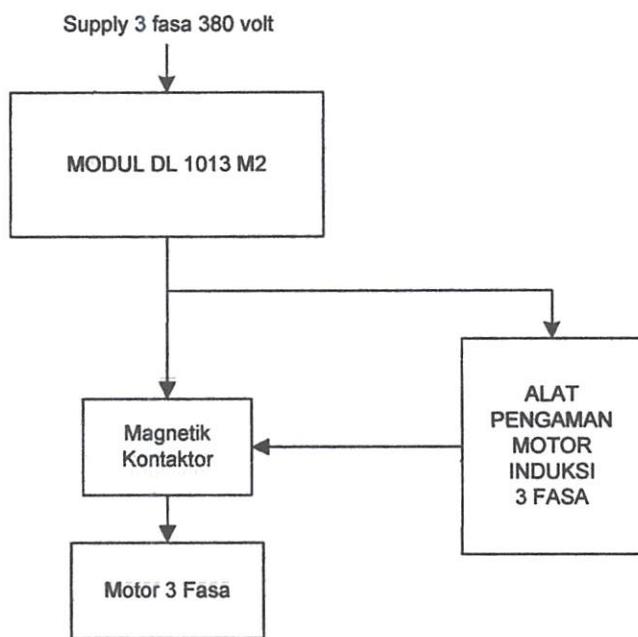
1. Satu set panel modul percobaan merk DELORENZO Italy DL 1013 M2.
2. Voltmeter Digital merk UniVolt Japan.

3. Motor 3 fasa 1,1 KW, 220/380 V, 2820 Rpm, 4,3 / 2,5 A, 50 Hz

Merk DELORENZO Italy.

4.10.3. Prosedur Pengujian

1. Membuat rangkaian seperti gambar 4.9.
2. Memasukkan setting tegangan kurang minimal dan setting delay time pada mikrokontroller melalui keypad dengan tampilan LCD.
3. Menghidupkan dan mengatur besarnya tegangan power supply pada modul DL 1013 M2 sebesar 220 Volt (tegangan fasa).
4. Menghidupkan motor dengan jalan menekan tombol ON pada magnetik kontaktor.
5. Mengurangi tegangan fasa dengan jalan memutar kenop pengatur tegangan ke kiri pada modul DL 1013 M2.
6. Mengamati penurunan tegangan fasa pada LCD dan mengamati munculnya tanda * yang berkedip pada LCD yang menandakan terjadi gangguan dan menghitung lamanya gangguan dengan asumsi 1 kedipan * berarti gangguan sudah berlangsung selama 1 detik.
7. Mengurangi tegangan fasa sampai melampaui setting tegangan minimal sampai akhirnya kontaktor trip.
8. Mengulangi mulai langkah ke-2 untuk setting tegangan kurang dan *delay time* yang berbeda.



Gambar 4.9 Blok Diagram Pengujian Alat Pengaman Motor Terhadap Gangguan Tegangan Kurang

4.10.4. Hasil Pengujian

Tabel 4.9 Hasil Pengujian Alat Terhadap Gangguan Tegangan Kurang

Setting Tegangan Min (Volt)	Tegangan Fasa (Volt)			Setting Delay Time (detik)	Lama Gangguan (detik)	Status Kontaktor	Waktu Trip
	V _R	V _S	V _T				
210	209	209	209	2	3	Trip	3
205	204	204	204	2	3	Trip	3
205	204	204	204	4	2	Tidak Trip	-
200	210	0	210	3	4	Trip	4
200	0	210	210	3	4	Trip	4

4.10.5. Analisis Hasil Pengujian

Dari tabel di atas dapat disimpulkan bahwa apabila tegangan fasa kurang dari setting tegangan minimal atau bahkan hilang/putus dan lama gangguan melebihi setting *delay time*, maka kontaktor akan trip, tetapi jika lama gangguan tidak melebihi setting *delay time*, maka sistem akan berjalan normal/tidak trip.

BAB V

P E N U T U P

5.1. Kesimpulan

Dari hasil perancangan dan pembuatan serta pengukuran alat, dapat disimpulkan sebagai berikut :

1. Tegangan keluaran pada sensor arus (y) sebanding dengan besarnya arus fasa yang disensor (x) dengan persamaan $y = 0,0033x^3 - 0,0892x^2 + 1,0797x - 0,5333$.
2. Hasil konversi ADC (y) sebanding dengan tegangan masukan ADC (x) dari sensor arus dengan persamaan $y = 51,094x - 0,3191$.
3. Tegangan keluaran pada sensor tegangan (y) sebanding dengan besar tegangan fasa yang disensor (x) dengan persamaan $y = 0,021x - 0,342$.
4. Hasil konversi ADC (y) sebanding dengan tegangan masukan ADC (x) dari sensor tegangan dengan persamaan $y = 51,04x - 0,5347$.
5. Setelah diuji coba dengan menggunakan beban simulasi maupun beban motor induksi 3 fasa, alat dapat bekerja dengan baik sesuai perencanaan.
6. Magnetik kontaktor akan trip apabila arus atau tegangan pada motor menyimpang dari setting yang telah ditetapkan dan melebihi setting *delay time*.

5.2. Saran

Masukan ADC yang masih tersisa 2 fasilitas untuk sensor yang lain dapat digunakan untuk pengembangan lebih lanjut.

DAFTAR PUSTAKA

- [1] Coughlin, Robert F., Frederick F. Driscoll, “**Penguatan Operasional dan Rangkaian Terpadu Linier**”, Penerbit Erlangga, edisi kedua.
- [2] Eko Putra, Agfianto, “**Belajar Mikrokontroller AT89c51/52/55 Teori dan Aplikasi**” , Penerbit Gava Media, edisi 2, 2005.
- [3] Ibnu Malik, Moh, “**Belajar Mikrokontroller AT89S8252**” , Penerbit Gava Media, 2003.
- [4] Malvino, “**Prinsip-Prinsip Elektronika**”, Penerbit Erlangga, edisi ketiga.
- [5] Sutrisno, “**Elektronika Teori dan Penerapannya**”, Penerbit ITB, jilid 2, 1987.
- [6] Wasito S, “**Vademekum Elektronika**”, Penerbit Gramedia Pustaka Utama, edisi kedua, 2004.
- [7] Zuhal, “**Dasar Tenaga Listrik**”, Penerbit ITB Bandung, 1991.
- [8] **Buku Panduan Pelatihan Mikrokontroller**, Lab. Pemeliharaan dan Perbaikan Perancangan Elektronika, ITN Malang, 2006.
- [9] Data Sheets AT89S8252, ATMEL.
- [10] Data Sheets ADC0808/ADC0809, www.national.com, 1999.
- [11] Liquid Crystal Display Module M 1632, User Manual, Seiko Instruments Inc., 1987.
- [12] Data Sheets 2 SC 828, www.DatasheetCatalog.com

Lampiran 1

- BERITA ACARA UJIAN SKRIPSI
- LEMBAR PERBAIKAN SKRIPSI
- FORMULIR PERBAIKAN SKRIPSI
- LEMBAR BIMBINGAN SKRIPSI
- FORMULIR BIMBINGAN SKRIPSI



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama : Herman Susilo
NIM : 0312125/P
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : Perencanaan dan Pembuatan Alat Pengaman Motor Induksi 3
Fasa Terhadap Gangguan Arus dan Tegangan Berbasis
Mikrokontroller AT89S8252

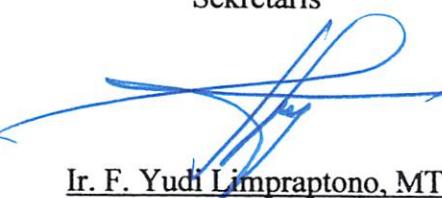
Dipertahankan di hadapan Majelis Penguji Skripsi jenjang Strata satu (S-1)
pada :

Hari : Senin
Tanggal : 25 September 2006
Dengan Nilai : 76,25 (B+) 

Panitia Ujian Skripsi

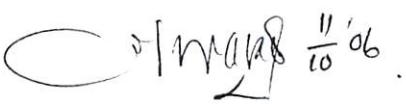
Ketua

Ir. Mochtar Asroni, MSME
NIP.Y.1018100036

Sekretaris

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

Anggota Penguji

Pengaji I


Ir. I Made Wartana, MT
NIP.131991182

Pengaji II


Ir. M. Abdul Hamid, MT
NIP.Y.1018800188



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK

LEMBAR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata satu (S-1) Jurusan Teknik Elektro Konsentrasi T. Energi Listrik pada :

Hari : Senin

Tanggal : 25 September 2006

telah dilakukan perbaikan skripsi untuk mahasiswa :

Nama : Herman Susilo

NIM : 0312125/P

Judul : Perencanaan dan Pembuatan Alat Pengaman Motor Induksi 3 Fasa Terhadap Gangguan Arus dan Tegangan Berbasis Mikrokontroller AT89S8252.

Perbaikan meliputi :

Uraian	Paraf
Sebaiknya dicoba dengan motor induksi sehingga dapat diuji keandalan atau proteksinya	

Penguji I

Diperiksa/disetujui,
Anggota Penguji

Penguji II

Ir. I Made Wartana, MT
NIP.131991182

Ir. M. Abdul Hamid, MT
NIP.Y.1018800188

Mengetahui,
Dosen Pembimbing

Ir. Widodo Pudji Muljanto, MT
NIP.Y.1028700171



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

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

NAMA : HERMAN S
NIM :
Perbaikan meliputi : 0312125 /P

Sabtu siang di coban & motor ini dibangun
sik syit di uji ke aman dan ada proteksi yang

Malang,

()



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK

LEMBAR BIMBINGAN SKRIPSI

Nama : Herman Susilo
NIM : 0312125/P
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : Perencanaan dan Pembuatan Alat Pengaman Motor Induksi 3 Fasa Terhadap Gangguan Arus dan Tegangan Berbasis Mikrokontroller AT89S8252
Tanggal Pengajuan Skripsi : 29 Juli 2006
Selesai Penulisan Skripsi : 22 September 2006
Dosen Pembimbing : Ir. Widodo Pudji Muljanto, MT
Telah Dievaluasi Dengan Nilai : 80 (Delapan Puluh)

Mengetahui,
Ketua Jurusan Teknik Elektro S-1



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

Diperiksa dan Disetujui,
Dosen Pembimbing



Ir. Widodo Pudji Muljanto, MT
NIP.Y.1028700171



FORMULIR BIMBINGAN SKRIPSI

Nama : HERMAN SUSILO
Nim : 0312125/P
Masa Bimbingan : Mulai tanggal 29 Juli 2006 s/d 29 Januari 2007
Judul : PERENCANAAN DAN PEMBUATAN ALAT PENGAMAN MOTOR INDUKSI 3 FASA TERHADAP GANGGUAN ARUS DAN TEGANGAN BERBASIS MIKROKONTROLLER AT89S8252

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	3/8-06	Konsultasi Bab I + II	/
2	7/8-06	Revisi Bab I + II	/
3	9/8-06	Acc Bab I + II, Konsultasi Bab III	/
4	16/8-06	Revisi Bab III	/
5	28/8-06	Acc Bab III	/
6	1/9-06	Konsultasi Bab IV + V	/
7	5/9-06	Revisi Bab IV + V	/
8	7/9-06	Konsultasi Makalah Seminar	/
9	8/9-06	Acc Seminar	/
10	22/9-06	Konsultasi Buku Skripsi Acc Kompre	/

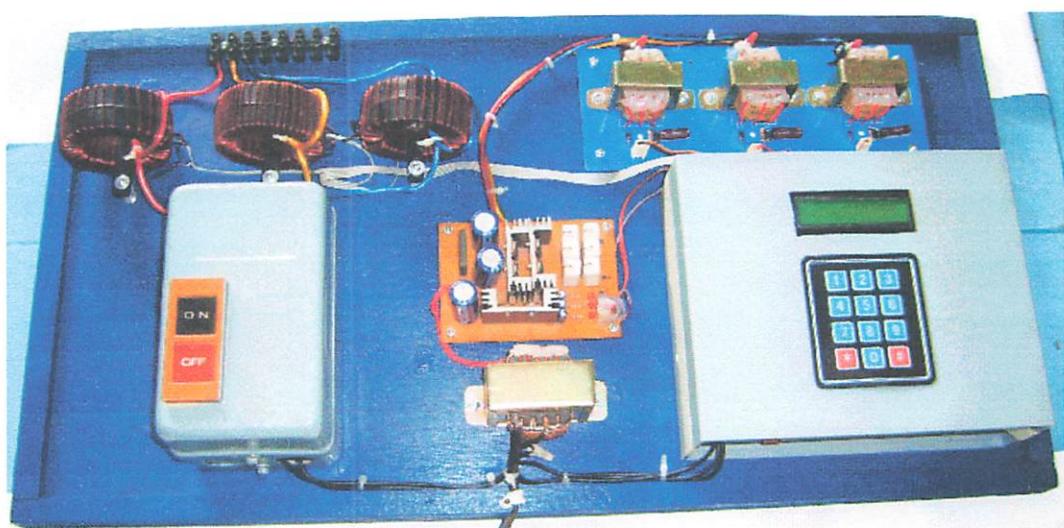
Malang, 22-Sext - 2006...

Dosen Pembimbing

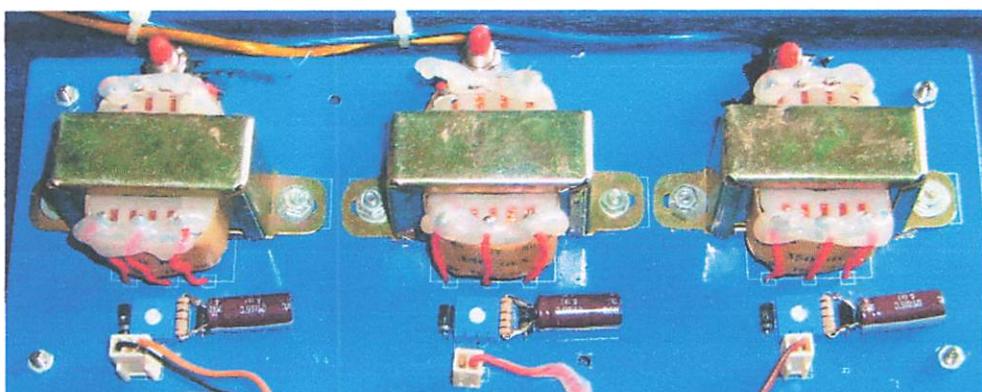
(Ir. WIDODO PUDJI M., MT)

Lampiran 2

- **FOTO ALAT DAN PENGUJIAN**
- **GAMBAR SKEMA RANGKAIAN**
- **LISTING PROGRAM**
- **DATASHEET AT89S8252**
- **DATASHEET ADC 0808 / 0809**
- **DATASHEET TR C 828**



Gambar Alat Keseluruhan



Sensor Tegangan



Sensor Arus



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK
LABORATORIUM KONVERSI ENERGI

Perihal : Permohonan Penggunaan Fasilitas Laboratorium Konversi Energi
Untuk Pengujian Alat Skripsi

Kepada : Yth. Bapak Ir. M. Abdul Hamid, MT
Ka. Lab. Konversi Energi
Institut Teknologi Nasional
Malang

Dengan hormat,

Saya yang bertanda-tangan di bawah ini :

N a m a : Herman Susilo
N I M : 0312125/P
Jurusan : Teknik Elektro S-1 Energi Listrik

Dengan ini mengajukan permohonan untuk dapat diijinkan menggunakan fasilitas laboratorium Konversi Energi guna pengujian alat Skripsi dengan judul :

Perencanaan dan Pembuatan Alat Pengaman Motor Induksi 3 fasa Terhadap Gangguan Arus dan Tegangan Berbasis Mikrokontroller AT89S8252

Demikian permohonan saya, atas izin serta perhatian Bapak, saya sampaikan terimakasih.



(Ir. M. Abdul Hamid, MT)

Malang, 27 September 2006
Hormat saya,

(Herman Susilo)



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK
LABORATORIUM KONVERSI ENERGI ELEKTRIK

Tabel Hasil Pengujian Alat Terhadap Gangguan Arus Lebih
pada Motor Induksi 3 Fasa

Setting Arus Max (A)	Arus Motor (A)			Setting Delay Time (detik)	Lama Gangguan (detik)	Status Kontaktor	Waktu Trip (detik)
	I_R	I_S	I_T				
2,30	2,32	2,32	2,32	3	4	Trip	4
2,27	2,30	2,30	2,30	3	4	Trip	4
2,27	2,30	2,30	2,30	4	3	Tidak	-
2,20	2,22	2,22	2,22	4	5	Trip	5
2,05	2,10	2,10	2,10	4	5	Trip	5

Data Motor 3 Fasa :

Merk : DELORENZO

20 Viate Romagna 20089 ROZZANO (Milano) ITALY

Daya : 1,1 KW

Tegangan : 220 / 380 Volt Δ / Y

Arus : 4,3 / 2,5 Amper

Rpm : 2820 Rpm

Frekuensi : 50 Hz

$\cos \phi$: 0,83



ACC Data
Asisten Laboratorium
Konversi Energi Elektrik



(Cosmas) (Anthy)



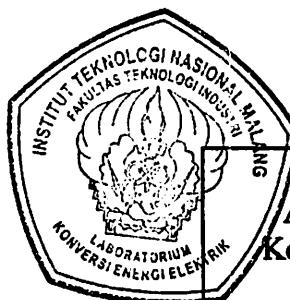
INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK
LABORATORIUM KONVERSI ENERGI ELEKTRIK

Tabel Hasil Pengujian Alat Terhadap Gangguan Tegangan Lebih

Setting Tegangan Max (Volt)	Tegangan Fasa (Volt)			Setting Delay Time (detik)	Lama Gangguan (detik)	Status Kontaktor	Waktu Trip
	V _R	V _S	V _T				
220	221	221	221	2	3	Trip	3
225	226	226	226	2	3	Trip	3
230	231	231	231	3	9	Trip	4
235	236	236	236	3	9	Trip	9
235	236	236	236	9	3	Tidak Trip	-

Tabel Hasil Pengujian Alat Terhadap Gangguan Tegangan Kurang

Setting Tegangan Min (Volt)	Tegangan Fasa (Volt)			Setting Delay Time (detik)	Lama Gangguan (detik)	Status Kontaktor	Waktu Trip
	V _R	V _S	V _T				
210	209	209	209	2	3	Trip	3
205	209	204	204	2	3	Trip	3
205	204	204	204	4	2	Tidak Trip	-
200	210	0	210	3	9	Trip	4
200	0	210	210	3	9	Trip	4



ACC Data
Asisten Laboratorium
Konversi Energi Elektrik

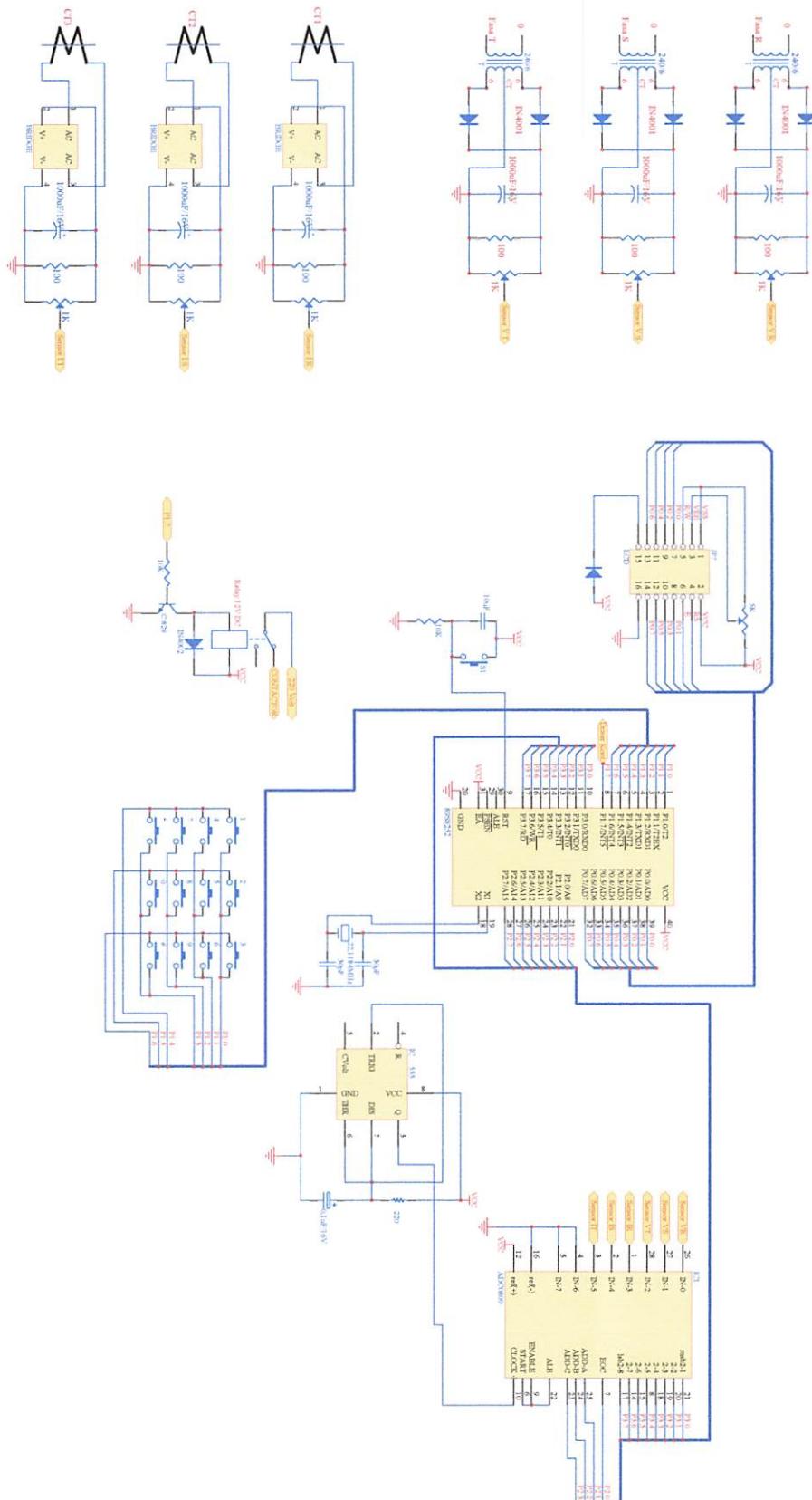
(Cosmas D Anthy)



Pengujian Alat Terhadap Gangguan Arus Lebih



Pengujian Alat Terhadap Gangguan Tegangan



Einführung in die Technik (Programm-Arbeit der TFG Informatik AT1992/23)	
Autoren	Nicole Schäfer
C	1. Präsentation
Autoren	Elisabeth Schäfer
Autoren	Katharina Schäfer

Listing Program

```
ADCToleransi    Equ     00h
LCDBaris1      Equ     80h
LCDBaris2      Equ     0C0h
cKeypadShow     Equ     0

Sensor1         Data    30h
Sensor2         Data    31h
Sensor3         Data    32h
Sensor4         Data    33h
Sensor5         Data    34h
Sensor6         Data    35h
Sensor7         Data    36h
Sensor8         Data    37h
IsError         Data    38h
CntTimeError    Data    39h
```

```
WMCON           Data    96h
```

```
;===== KEYPAD =====
kolom1          Bit     P1.4    ; kiri (1,4,7,redial)
kolom2          Bit     P1.5    ; (2,5,8,0)
kolom3          Bit     P1.6    ; (3,3,9,#)
kolom4          Bit     P1.7

baris1          Bit     P1.0    ; atas (1,2,3)
baris2          Bit     P1.1    ; (4,5,6)
baris3          Bit     P1.2    ; (7,8,9)
baris4          Bit     P1.3    ; (*,0,#)
;
keyport         Data    P1
```

```
Data_LCD        Equ     P0
lcdrs           Bit     P0.4      ;LCD rs pin connected to P6
lcde            Bit     P0.5      ;LCD e pin connected to PA4
```

```
-----
; format:
; Flag          -> 1
; Nama          -> 10
; Score         -> (3 * 10) = 30
; Total         -> 1
; ----- + 42 bytes
```

```
DataVOver       Equ     0000h
DataVUnder      Equ     0001h
DataI           Equ     0002h
DataTime        Equ     0003h
```

```
DphhX           Data    36h
DpllX           Data    37h
```

Dphh	Data	38h
Dpll	Data	39h
CntKey	Data	3Ah
keydata	Data	3Bh
FlagKey	Data	3Ch
Tmp1	Data	10h
Tmp2	Data	11h
Tmp3	Data	12h
Tmp4	Data	13h
tmp5	Data	14h
tmp6	Data	15h
Tmp1a	Data	16h
Tmp2a	Data	17h
Tmp3a	Data	18h
Tmp4a	Data	19h
tmp5a	Data	1Ah
tmp6a	Data	1Bh
VOver	Data	40h
VUnder	Data	41h
I	Data	42h
TDelay	Data	43h

Booting:

Mov	SP	#80h
Clr	EA	
Call	LCD_Inisialisasi	
Call	LCD_Clear	
Call	LCD_Clear	
Call	LCD_Blink_Off	
Jmp	StartGetADC	
Jmp	MenuUtama	
;	0123456789abcdef	
txt_menu1:	Db	'1.Set V 3.Delay',0
txt_menu2:	Db	'2.Set I 4.Run',0
txt_V1:	Db	'V Max:xxx V',0
txt_V2:	Db	'V Min:xxx V',0
txt_I1:	Db	'I Max:x,xx A',0
txt_Delay1:	Db	'Delay:xx s',0
;	0123456789abcdef	
txt_Yakin1:	Db	' Anda yakin?',0
txt_Yakin2:	Db	' (1-Y/2-T)',0
txt_Simpan:	Db	' Data disimpan!',0

```

txt_Trip:      Db      ' Sistem TRIP',0

MenuUtama:
    Call    LCD_Clear
    Call    lcd_line_1
    Mov     DPTR, #txt_menu1
    Call    LCD_String

    Call    LCD_Line_2
    Mov     DPTR, #txt_menu2
    Call    LCD_String

GetMenuUtama:
    Mov     R0, #1
    Call    Keypad_Get
    Cjne   A, #'1', CekMenu2
    Jmp    MenuSetingV

CekMenu2:
    Cjne   A, #'2', CekMenu3
    Jmp    MenuSetingI

CekMenu3:
    Cjne   A, #'3', CekMenu4
    Jmp    MenuSetingT

CekMenu4:
    Cjne   A, #'4', GetMenuUtama
    Jmp    Running

;-----
MenuSetingV:
    Call    LCD_Clear
    Call    lcd_line_1
    Mov     DPTR, #txt_V1
    Call    LCD_String

    Call    LCD_Line_2
    Mov     DPTR, #txt_V2
    Call    LCD_String

MenuSetingOvH:
    Mov     R0, #16h
    Call    LCD_Position

    Mov     A, #LCDBaris1+6
    Call    LCD_Command
    Call    LCD_Blink_On

    Mov     R0, #cKeypadShow
    Call    Keypad_Get
    Anl    A, #0Fh
    Mov     tmp1a, A

    Mov     R0, #cKeypadShow

```

```

Call      Keypad_Get
Anl      A, #0Fh
Swap     A
Mov      Tmp1,A
;---
Mov      R0, #cKeypadShow
Call      Keypad_Get
Anl      A, #0Fh
Orl      A, Tmp1
Mov      Tmp1,A

MenuSettingOvL:
Mov      A, #LCDBaris2+6
Call      LCD_Command

Mov      R0, #cKeypadShow
Call      Keypad_Get
Anl      A, #0Fh
Mov      tmp2a,A

Mov      R0, #cKeypadShow
Call      Keypad_Get
Anl      A, #0Fh
Swap     A
Mov      Tmp2,A
;---
Mov      R0, #cKeypadShow
Call      Keypad_Get
Anl      A, #0Fh
Orl      A, Tmp2
Mov      Tmp2,A

Call      LCD_Blink_Off

ConversionVI:
;-----
;Over
Mov      6, Tmp1a
Mov      7, Tmp1
Call      BCD_Hex
Mov      VOver, 7

;-----
;Under
Mov      6, Tmp2a
Mov      7, Tmp2
Call      BCD_Hex
Mov      VUnder, 7

Call      LCD_Clear
Call      lcd_line_1
Mov      DPTR, #txt_Yakin1
Call      LCD_String

Call      LCD_Line_2
Mov      DPTR, #txt_Yakin2

```

```

    Call    LCD_String

GetYakinV1:
    Mov     R0, #1
    Call    Keypad_Get
    Cjne   A, #'1', GetYakinV2

    Mov     DPTR, #DataVOver
    Mov     A, VOver
    Call    EEPROM_Write

    Mov     DPTR, #DataVUnder
    Mov     A, VUnder
    Call    EEPROM_Write

    Call    LCD_Clear
    Call    lcd_line_1
    Mov    DPTR, #txt_Simpan
    Call    LCD_String

    Call    Delay_Fix_1s
    Call    Delay_Fix_1s
    Jmp    MenuUtama

GetYakinV2:
    Cjne   A, #'2', GetYakinV1
    Jmp    MenuUtama

;-----
MenuSettingI:
    Call    LCD_Clear
    Call    lcd_line_1
    Mov    DPTR, #txt_I1
    Call    LCD_String

MenuSettingOvI:
    Mov     R0, #16h
    Call    LCD_Position

    Call    LCD_Command
    Call    LCD_Blink_On

    Mov     R0, #cKeypadShow
    Call    Keypad_Get
    Anl    A, #0Fh
    Swap   A
    Mov    Tmp1, A

    Mov     A, '#'
    Call    LCD_Char
;---
    Mov     R0, #cKeypadShow
    Call    Keypad_Get
    Anl    A, #0Fh
    Orl    A, Tmp1
    Mov    Tmp1, A

```

```

Mov      R0, #cKeypadShow
Call     Keypad_Get

Call     LCD_Blink_Off

ConversionI:
;-----
;Over
Mov      6, #0
Mov      7, Tmp1
Call     BCD_Hex
Mov      A, 7
Clr      C
Subb    A, #5
Call     Get_DPTR_ArusSeting
Clr      A
Movc    A, @A+DPTR
Mov      I, A

Call     LCD_Clear
Call     lcd_line_1
Mov      DPTR, #txt_Yakin1
Call     LCD_String

Call     LCD_Line_2
Mov      DPTR, #txt_Yakin2
Call     LCD_String

GetYakinI1:
Mov      R0, #1
Call     Keypad_Get
Cjne   A, #'1', GetYakinI2

Mov      DPTR, #DataI
Mov      A, I
Call     EEPROM_Write

Call     LCD_Clear
Call     lcd_line_1
Mov      DPTR, #txt_Simpan
Call     LCD_String

Call     Delay_Fix_1s
Call     Delay_Fix_1s
Jmp     MenuUtama

GetYakinI2:
Cjne   A, #'2', GetYakinI1
Jmp     MenuUtama

;-----
MenuSetingT:
Call     LCD_Clear
Call     lcd_line_1
Mov      DPTR, #txt_Delay1

```

```

Call      LCD_String

MenuSetingDelay:
    Mov      R0, #16h
    Call     LCD_Position

    Mov      A, #LCDBaris1+6
    Call     LCD_Command
    Call     LCD_Blink_On

    Mov      R0, #cKeypadShow
    Call     Keypad_Get
    Anl      A, #0Fh
    Swap    A
    Mov      Tmp1, A
    ;---
    Mov      R0, #cKeypadShow
    Call     Keypad_Get
    Anl      A, #0Fh
    Orl      A, Tmp1
    Mov      Tmp1, A

    Call     LCD_Blink_Off

ConversionT:
;-----
;Over
    Mov      6, #0
    Mov      7, Tmp1
    Call    BCD_Hex
    Mov      TDelay, 7

    Call     LCD_Clear
    Call     lcd_line_1
    Mov      DPTR, #txt_Yakin1
    Call     LCD_String

    Call     LCD_Line_2
    Mov      DPTR, #txt_Yakin2
    Call     LCD_String

GetYakinT1:
    Mov      R0, #1
    Call     Keypad_Get
    Cjne   A, #'1', GetYakinT2

    Mov      DPTR, #DateTime
    Mov      A, TDelay
    Call     EEPROM_Write

    Call     LCD_Clear
    Call     lcd_line_1
    Mov      DPTR, #txt_Simpan
    Call     LCD_String

    Call     Delay_Fix_1s

```

```

Call      Delay_Fix_1s
Jmp      MenuUtama

GetYakinT2:
Cjne    A, #'2', GetYakinT1
Jmp      MenuUtama

```

Running:

```

Call      LCD_Clear
Call      lcd_line_1

Mov      Sensor1, #10
Mov      Sensor2, #10
Mov      Sensor3, #10

Mov      Sensor4, #9
Mov      Sensor5, #11
Mov      Sensor6, #10

Mov      VOver, #10
Mov      VUnder, #10
Mov      I, #10

Mov      CntTimeError, #5
Mov      TDelay, CntTimeError

```

Simulasi:

```

Call      Delay_Fix_1s

Call      IsSensorError
Cjne    A, #0FFh, SClearError
Mov      A, IsError
Cjne    A, #00, SMenuUtamaXX
Mov      IsError, #0FFh
Jmp      SMenuUtamaXX

```

SClearError:

```

Mov      R0, #1Fh
Call      LCD_Position
Mov      A, '_'
Call      LCD_Char

Mov      IsError, #0
Mov      TDelay, CntTimeError

```

SMenuUtamaZZ:

```

Jmp      Simulasi

```

SMenuUtamaXX:

```

Mov      R0, #1Fh
Call      LCD_Position
Mov      A, '*'
Call      LCD_Char
Djnz      TDelay, Simulasi

```

```

Call    LCD_Clear
Call    lcd_line_1
Mov     DPTR, #txt_Trip
Call    LCD_String

Sjmp   $
Call    LCD_Hexa
Sjmp   $

Mov     DPTR, #DataVOver
Call    EEPROM_Read
Call    LCD_Hexa
Mov     A, #'-
Call    LCD_Char

Mov     DPTR, #DataVUnder
Call    EEPROM_Read
Call    LCD_Hexa
Mov     A, #'-
Call    LCD_Char

Mov     DPTR, #DataI
Call    EEPROM_Read
Call    LCD_Hexa
Mov     A, #'-
Call    LCD_Char

Mov     DPTR, #DataTime
Call    EEPROM_Read
Call    LCD_Hexa
Mov     A, #'-
Call    LCD_Char

Sjmp   $

LoopX:
;Running:

Mov     IsError, #0

Mov     DPTR, #DataVOver
Call    EEPROM_Read
Mov     VOver, A

Mov     DPTR, #DataVUnder
Call    EEPROM_Read
Mov     VUnder, A

Mov     DPTR, #DataI
Call    EEPROM_Read
Mov     I, A

Mov     DPTR, #DataTime
Call    EEPROM_Read

```

```

Mov      CntTimeError,A
Mov      TDelay,CntTimeError

StartGetADC:
    Call    ADCGet1
    Call    Delay_Fix_100ms

    Call    ADCGet2
    Call    Delay_Fix_100ms

    Call    ADCGet3
    Call    Delay_Fix_100ms

    Call    ADCGet4
    Call    Delay_Fix_100ms

    Call    ADCGet5
    Call    Delay_Fix_100ms

    Call    ADCGet6
    Call    Delay_Fix_100ms

    Call    Delay_Fix_100ms
    Call    Delay_Fix_100ms
    Call    Delay_Fix_100ms

;<-----
```

```

Call    ShowADCHex

Jmp    StartGetADC

Call    IsSensorError
Cjne  A,#0FFh,ClearError
Mov   A,IsError
Cjne  A,#00,MenuUtamaXX
Mov   IsError,#0FFh

ClearError:
    Mov   IsError,#0
    Mov   TDelay,CntTimeError
MenuUtamaZZ:
    Jmp   MenuUtama

MenuUtamaXX:
    Djnz  TDelay,MenuUtamaZZ

    Call    LCD_Clear
    Call    lcd_line_1
    Mov   DPTR,#txt_Trip
    Call    LCD_String

    Sjmp  $
```

```

IsSensorError:

;cek Over Voltage
IsOverVoltageR:
    Mov     A, Sensor1
    Cjne   A, VOver, IsOverVoltageRBesar
    Jmp    IsOverVoltageS

IsOverVoltageRBesar:
    Jc      IsOverVoltageS
    Mov    A, #0FFh
    Ret

;-----
IsOverVoltageS:
    Mov     A, Sensor2
    Cjne   A, VOver, IsOverVoltageSBesar
    Jmp    IsOverVoltageT

IsOverVoltageSBesar:
    Jc      IsOverVoltageT
    Mov    A, #0FFh
    Ret

;-----
IsOverVoltageT:
    Mov     A, Sensor3
    Cjne   A, VOver, IsOverVoltageTBesar
    Jmp    IsUnderVoltageR

IsOverVoltageTBesar:
    Jc      IsUnderVoltageR
    Mov    A, #0FFh
    Ret

;=====
;cek Under Voltage
IsUnderVoltageR:
    Mov     A, Sensor1
    Cjne   A, VUnder, IsUnderVoltageRBesar
    Jmp    IsUnderVoltageS

IsUnderVoltageRBesar:
    Jnc    IsUnderVoltageS
    Mov    A, #0FFh
    Ret

;-----
IsUnderVoltageS:
    Mov     A, Sensor2
    Cjne   A, VUnder, IsUnderVoltageSBesar
    Jmp    IsUnderVoltageT

IsUnderVoltageSBesar:
    Mov    A, #0FFh
    Ret

;-----
IsUnderVoltageT:
    Mov     A, Sensor3
    Cjne   A, VUnder, IsUnderVoltageTBesar
    Jmp    IsOverCurrentR

IsUnderVoltageTBesar:
    Mov    A, #0FFh
    Ret

```

```

;=====
;cek Over Current
IsOverCurrentR:
    Mov     A, Sensor4
    Cjne   A, I, IsOverCurrentRBesar
    Jmp    IsOverCurrentS
IsOverCurrentRBesar:
    Jc     IsOverCurrentS
    Mov    A, #0FFh
    Ret
;-----
IsOverCurrentS:
    Mov     A, Sensor5
    Cjne   A, I, IsOverCurrentSBesar
    Jmp    IsOverCurrentT
IsOverCurrentSBesar:
    Jc     IsOverCurrentT
    Mov    A, #0FFh
    Ret
;-----
IsOverCurrentT:
    Mov     A, Sensor6
    Cjne   A, I, IsOverCurrentTBesar
    Jmp    SensorNormal
IsOverCurrentTBesar:
    Jc     SensorNormal
    Mov    A, #0FFh
    Ret
SensorNormal:
    Mov    A, #0
    Ret

EEPROM_Write:
RAM_Write:
    Push   01
    Mov    01, A
    Mov    A, WMCON
    Clr    ACC.0
    Setb   ACC.3
    Setb   ACC.4
    Mov    WMCON, A

cek_lagi:
    Mov    A, WMCON
    Jnb    ACC.1, cek_lagi
    Mov    A, 01
save_03:
    Movx   @DPTR, A
save_01:
    Movx   A, @DPTR
    Mov    WMCON, A
    Pop
    01

```

```

    Ret

EEPROM_Read:
RAM_Read:
    Call    Set_EEProm
    Movx   A, @DPTR
    Ret

Set_EEProm:
    Push   ACC

cek_lagia:
    ;Mov    A, WMCN
    ;Jnb    ACC.1, cek_lagia

    Mov    A, WMCN
    Clr    ACC.0
    Setb   ACC.3
    Setb   ACC.4
    Mov    WMCN, A

    Pop    ACC
    Ret

BCD_Hex:
    Push   ACC
    Push   B

    Mov    A, 07
    Swap   A
    Anl    A, #0Fh           ; ambil xx9x
    Mov    B, #0Ah
    Mul    AB

    Mov    A, 06
    Anl    A, #0Fh           ; ambil x9xx
    Mov    B, #64h
    Mul    AB
    Add    A, 07
    Mov    07, A
    Mov    06, B

    Pop    B
    Pop    ACC
    Ret

Hex_BCD:
    Push   B

    Mov    B, #100
    Div    AB
    Mov    06, A
    Mov    A, #10
    Xch    A, B
    Div    AB
    Swap   A
    Add    A, B

```

```

Mov      07,A
Pop      B
Ret
ShowADCHex:
Call    LCD_Clear
Call    lcd_line_1

Mov      A,'V'
Call    LCD_Char
Mov      A,':'
Call    LCD_Char

Mov      A,Sensor1
Anl    A,#0Fh
Orl    A,#30h

Mov      A,' '
Call    LCD_Char

Mov      A,Sensor2
Call    Conversion_Hexa_BCD
Mov      A,6
Anl    A,#0Fh
Orl    A,#30h
Call    LCD_Char
Mov      A,7
Call    LCD_Hexa

Mov      A,' '
Call    LCD_Char

Mov      A,Sensor3
Call    Conversion_Hexa_BCD
Mov      A,6
Anl    A,#0Fh
Orl    A,#30h
Call    LCD_Char
Mov      A,7
Call    LCD_Hexa

;---- ARUS ----
Call    LCD_Line_2

Mov      A,'I'
Call    LCD_Char
Mov      A,':'
Call    LCD_Char

Mov      A,Sensor4
Call    Get_DPTR_Arus
Call    LCD_String
Mov      A,' '
Call    LCD_Char

Mov      A,Sensor5

```

```

Call    Get_DPTR_Arus
Call    LCD_String
Mov     A, #' '
Call    LCD_Char

Mov     A, Sensor6
Call    Get_DPTR_Arus
Call    LCD_String
Mov     A, #' '
Call    LCD_Char

Ret
Conversion_Hexa_BCD:
Push   B
Mov    B, #100
Div    AB
Mov    6, A
Mov    A, #10
Xch    A, B
Div    AB
Swap   A
Add    A, B
Mov    7, A
Pop    B
Ret
Ret
ADCGet1:
Mov    P2, #00000000b ;alamat
Nop
Mov    P2, #00001000b ;ale
Nop
Mov    P2, #00110000b ;ena
Nop

Nop
Nop
Nop
Mov    A, P3
Clr    C
Subb  A, #ADCToleransi
Mov    Sensor6, A
Mov    P2, #00000000b ;ena+alamat
Nop

Nop

```

```

Nop

Ret
ADCGet2:
    Mov      P2,#00000001b    ;alamat
    ;Call    PPI_Send_Port_A
    Nop
    Nop
    Nop
    Nop
    Nop
    Mov      P2,#00001001b    ;ale
    ;Call    PPI_Send_Port_A
    Nop
    Nop
    Nop
    Nop
    Nop
    Mov      P2,#00100001b    ;alamat+ale
    ;Call    PPI_Send_Port_A
    Nop
    Mov      P2,#00110001b    ;ena
    ;Call    PPI_Send_Port_A
    Nop

    Nop
    Nop
    Nop
    Mov      A,P3
    Clr      C
    Subb   A,#ADCToleransi
    Add     A,#1
    Mov     Sensor1,A
    Mov      P2,#00000001b    ;ena+alamat
    ;Call    PPI_Send_Port_A
    Nop

    Nop
    Nop
    Ret
ADCGet3:
    Mov      P2,#00000010b    ;alamat
    ;Call    PPI_Send_Port_A
    Nop
    Nop
    Nop

```

```

Nop
Mov      P2,#00001010b    ;ale
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Mov      P2,#00100010b    ;alamat+ale
;Call    PPI_Send_Port_A
Nop
Mov      P2,#00110010b    ;ena
;Call    PPI_Send_Port_A
Nop

Nop
Nop
Nop
Mov      A,P3
Clr      C
Subb   A,#ADCToleransi
Add     A,#1+2
Mov     Sensor2,A
Mov      P2,#00000010b    ;ena+alamat
;Call    PPI_Send_Port_A
Nop

Nop
Nop
Ret
ADCGet4:
Mov      P2,#00000011b    ;alamat
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Nop
Mov      P2,#00001011b    ;ale
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Mov      P2,#00100011b    ;alamat+ale
;Call    PPI_Send_Port_A
Nop

```

```

Nop
Mov      P2,#00110011b    ;ena
Nop

Nop
Nop
Nop
Mov      A,P3
Clr      C
Subb    A,#6
Add     A,#1
Mov     Sensor3,A
Mov      P2,#00000011b    ;ena+alamat
;      Call    PPI_Send_Port_A
Nop

Nop
Nop
Ret

ADCGet5:
Mov      P2,#00000100b    ;alamat
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Nop
Mov      P2,#00001100b    ;ale
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Mov      P2,#00100100b    ;alamat+ale
;Call    PPI_Send_Port_A
Nop

```

```

Nop
Nop
Nop
Nop
Mov      P2,#00110100b    ;ena
;Call    PPI_Send_Port_A
Nop

Nop
Nop
Nop
Mov      A,P3
Clr      C
Subb   A,#ADCToleransi
Add     A,#4
Mov     Sensor4,A
Mov      P2,#00000100b    ;ena+alamat
;      Call    PPI_Send_Port_A
Nop

Nop
Nop
Ret

ADCGet6:
Mov      P2,#00000101b    ;alamat
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Nop
Mov      P2,#00001101b    ;ale
;Call    PPI_Send_Port_A
Nop
Nop
Nop
Mov      P2,#00100101b    ;alamat+ale
;Call    PPI_Send_Port_A
Nop
Mov      P2,#00110101b    ;ena
;Call    PPI_Send_Port_A
Nop

```

```

Nop
Nop
Nop
Mov    A, P3
Clr    C
Subb   A, #ADCToleransi
Mov    Sensor5,A
Mov    P2,#00000101b ;ena+alamat
; Call   PPI_Send_Port_A
Nop

Nop
Nop
Ret
ADCGet7:
Mov    P2,#00000110b ;alamat
;Call   PPI_Send_Port_A
Nop
Nop
Nop
Nop
Mov    P2,#00001110b ;ale
;Call   PPI_Send_Port_A
Nop
Nop
Nop
Nop
Mov    P2,#00100110b ;alamat+ale
;Call   PPI_Send_Port_A
Nop
Mov    P2,#00110110b ;ena
;Call   PPI_Send_Port_A
Nop

Nop
Nop
Nop
;Lcall  PPI_Get_Port_B
Mov    Sensor7,P3
Mov    P2,#00000110b ;ena+alamat
; Call   PPI_Send_Port_A
Nop

Nop

```

```

        Nop
    Ret
;===== DELAY =====
Delay_Var_1ms:
    Call    Delay_Fix_1ms
    Djnz    R0,Delay_Var_1ms
    Ret
Delay_Var_10ms:
    Call    Delay_Fix_10ms
    Djnz    R0,Delay_Var_10ms
    Ret
Delay_Var_100ms:
    Call    Delay_Fix_100ms
    Djnz    R0,Delay_Var_100ms
    Ret
Delay_Var_1s:
    Call    Delay_Fix_1s
    Djnz    R0,Delay_Var_1s
    Ret
Delay_Var_10s:
    Call    Delay_Fix_10s
    Djnz    R0,Delay_Var_10s
    Ret
Delay_Var_10us:
    Call    Delay_Fix_10us
    Djnz    R0,Delay_Var_10us
    Ret
Delay_Fix_10us:
    Push   1
    Mov    1,#20
    Djnz   1,$
    Pop    1
    Ret
Delay_Fix_10s:
    Push   1
    Mov    1,#100
delay_fix_10s_1:
    Call    Delay_Fix_100ms
    Djnz    1,delay_fix_10s_1
    Pop    1
    Ret
Delay_Fix_1s:
    Push   1
    Mov    1,#100
delay_fix_1000ms_1:
    Call    Delay_Fix_10ms
    Djnz    1,delay_fix_1000ms_1
    Pop    1
    Ret
Delay_Fix_100ms:
    Push   1
    Mov    1,#10
delay_fix_100ms_1:
    Call    Delay_Fix_10ms
    Djnz    1,delay_fix_100ms_1
    Pop    1

```

```

        Ret
Delay_Fix_10ms:
    Mov     TMOD,#00000001b ; Timer 1 bekerja pada mode 1
    Mov     TLO,#3Dh   ; siapkan waktu tunda 50 mili-detik
    Mov     TH0,#0B0h
    Clr     TFO          ; me-nol-kan bit limpahan
    Setb    TR0          ; timer mulai bekerja
    Jnb    TFO,$         ; tunggu di sini sampai melimpah
    Clr     TR0          ; timer berhenti kerja
    Ret
        Ret
Delay_Fix_1ms:
    Mov     TMOD,#00000001b ; Timer 1 bekerja pada mode 1
    Mov     TLO,#0EDh   ; siapkan waktu tunda 50 mili-detik
    Mov     TH0,#78h
    Clr     TFO          ; me-nol-kan bit limpahan
    Setb    TR0          ; timer mulai bekerja
    Jnb    TFO,$         ; tunggu di sini sampai melimpah
    Clr     TR0          ; timer berhenti kerja
    Ret
        Ret

Keypad_Get:
    Push   B           ; amankan register B
KeyInGet1:
    Call   Keypad3x4      ;scan keypad
    Mov    A,keydata       ;isi keydata = data di
rutin keypad
    Cjne  A,#0FFh,KeyInGet0 ;jk isi a # Offh ==>lompat
    Jmp   KeyInGet1
KeyInGet0:
    Mov    B,A           ;simpan isi a to b
KeyInGet:
    Call   Keypad3x4      ;jk isi A # B lompat
    Cjne  A,B,KeyInOut
    Jmp   KeyInGet
KeyInOut:
    Mov    A,R0
    Cjne  A,#0,KeyInOut1
    Mov    A,B
    Call   LCD_Char
KeyInOut1:
    Call   Delay_Fix_100ms
    Call   Delay_Fix_100ms
    Call   Delay_Fix_100ms
    Call   Delay_Fix_100ms
    Mov    A,B
    Pop    B           ;idem
    Ret

=====
; routine u/ baca keypad 3x4
; output pd keydata(0-9,E=redial,F=#)
=====
Keypad3x4:

```

```

        Mov      keyport,#0FFh
        Clr      kolom1
;-----
;----- ull:
        Jb      baris1,key1
        Mov      keydata,#'1'
        Ret
key1:
        Jb      baris2,key2
        Mov      keydata,#'4'
        Ret
key2:
        Jb      baris3,key3
        Mov      keydata,#'7'
        Ret
key3:
        Jb      baris4,key4
        Mov      keydata,#'*'
        Ret
;-----
key4:
        Setb    kolom1
        Clr     kolom2
;-----
        Jb      baris1,key5
        Mov      keydata,#'2'
        Ret
key5:
        Jb      baris2,key6
        Mov      keydata,#'5'
        Ret
key6:
        Jb      baris3,key7
        Mov      keydata,#'8'
        Ret
key7:
        Jb      baris4,key8
        Mov      keydata,#'0'
        Ret
;-----
key8:
        Setb    kolom2
        Clr     kolom3
;-----
        Jb      baris1,key9
        Mov      keydata,#'3'
        Ret
key9:
        Jb      baris2,key10
        Mov     keydata,#'6'
        Ret
key10:
        Jb      baris3,key11
        Mov     keydata,#'9'
        Ret

```

```

key11:
    Jb      baris4,key12
    Mov     keydata,'#"'
    Ret
key12:
    Mov     keydata,#0FFh

    Ret
LCD_Clear1:
    Call   lcd_line_1
    Mov    DPTR,#txt_blank
    Call   LCD_String

    Call   LCD_Line_2
    Mov    DPTR,#txt_blank
    Call   LCD_String

    Ret
txt_blank:    Db      '           ',0

LCD_Clear:
    Mov    A,#01h          ;Clear LCD command
    Call   LCD_Command
    Ret

LCD_Blink_Off:
    Push   ACC
    Mov    A,#00001100b    ;
    Lcall  LCD_Command
    Pop    ACC
    Ret
LCD_Blink_On:
    Push   ACC
    Mov    A,#00001101b    ;
    Lcall  LCD_Command
    Pop    ACC
    Ret
lcd_line_1:
    Push   ACC
    Push   AR0
    Mov    A,#80h
    Call   LCD_Command
    Pop    AR0
    Pop    ACC
    Ret
LCD_Line_2:
    Push   ACC
    Push   AR0
    Mov    A,#0C0h
    Call   LCD_Command
    Pop    AR0
    Pop    ACC
    Ret
LCD_Line_3:

```

```

        Push    ACC
        Push    ARO
        Mov     A, #94h
        Call    LCD_Command
        Pop     ARO
        Pop     ACC
        Ret
LCD_Clear_Line2:
        Call    LCD_Line_2
        Mov    DPTR, #txt_blank
        Call    LCD_String
        Call    LCD_Line_2
        Ret
LCD_Line_4:
        Push    ACC
        Push    ARO
        Mov     A, #0D4h
        Call    LCD_Command
        Pop     ARO
        Pop     ACC
        Ret
LCD_Position:
        Push    ACC
        Mov    A, R0      ;15      26
        Anl    A, #0F0h   ;10      20
        Cjne   A, #10h, lcd_cursor_position1
        Mov    A, R0      ;15
        Anl    A, #0Fh    ;05
        Orl    A, #80h    ;85
        Call   LCD_Command
        Jmp   lcd_cursor_position_end
lcd_cursor_position1:
        Cjne   A, #20h, lcd_cursor_position_end
        Mov    A, R0      ; 26
        Anl    A, #0Fh    ;06
        Orl    A, #0C0h    ; c6
        Call   LCD_Command
lcd_cursor_position_end:
        Pop    ACC
        Ret
=====
=====
;           Initialize LCD module
=====
=====
LCD_Inisialisasi:
        Mov    Data_LCD, #0
        Call   Delay_Fix_100ms
        Mov    Data_LCD, #3
        Call   Clk_LCD

        Call   Delay_Fix_10ms
        Mov    Data_LCD, #3

```

```

        Call    Clk_LCD

        Call    Delay_Fix_1ms

        Mov     Data_LCD,#3
        Call    Clk_LCD

        Mov     Data_LCD,#2
        Call    Clk_LCD

font      Mov     A,#00101000b      ;Function set, 4 wire, 2 line, 5x7
        Call    LCD_Command

off       Mov     A,#00001100b      ;Display on, cursor off, blink
        Call    LCD_Command

        Mov     A,#00000110b      ;Address increment, no scrolling
        Call    LCD_Command

        Ret

Clk_LCD:
        Setb   lcde      ;Toggle enable line
        Call    Delay_Fix_10us
        Clr    lcde
        Ret

LCD_Command:
        Push   B
        Mov    B,A

        Clr    lcdrs

        Mov    A,B
        Swap   A
        Anl    A,#0Fh
        Mov    Data_LCD,A

        Call    Clk_LCD

        Mov    A,B
        Anl   A,#0Fh      ;Strip off upper bits
        Mov    Data_LCD,A      ;Put on port

        Call    Clk_LCD
        Call    delayL

        Pop    B

        Ret

LCD_Char:
        Push   B
        Push   0
        Push   ACC

```

```

Mov      B,A      ;Save character

Setb    lcdrs

Mov      A,B      ;Get character back
Swap    A          ;Get upper nibble
Anl    A,#0Fh    ;Strip off upper bits
Mov      Data_LCD,A ;Put on port

Setb    lcdrs    ;Register select set for data
Call    Clk_LCD

Mov      A,B      ;Recall character
Anl    A,#0Fh    ;Strip off upper bits
Mov      Data_LCD,A ;Put on port

Setb    lcdrs    ;Register select set for data
Call    Clk_LCD

Call    delayL

Pop    ACC
Pop    0
Pop    B

Ret

LCD_String:
Push  ACC
Push  DPL
Push  DPH

TampilKata1:
Clr   A
Movc  A,@A+DPTR
Cjne A,#0,TampilKata2           ;intinya
Jmp   out

TampilKata2:
Inc   DPTR
Call  LCD_Char
Jmp   TampilKata1

out:
Pop   DPH
Pop   DPL
Pop   ACC

Ret

LCD_Hexa:
Push  07h
Push  ACC

Mov   7,A
Anl  A,#0F0h
Swap A
Orl  A,#30h
Lcall tes_huruf_
Call  LCD_Char

```

```

Mov      A, 7
Anl      A, #0Fh
Orl      A, #30h
Lcall    tes_huruf_
Lcall    LCD_Char

Pop     ACC
Pop     07h
Ret

tes_huruf_:
Cjne   A, #3Ah, tes_huruf_1
Mov    A, #'A'
Ret

tes_huruf_1:
Cjne   A, #3Bh, tes_huruf_2
Mov    A, #'B'
Ret

tes_huruf_2:
Cjne   A, #3Ch, tes_huruf_3
Mov    A, #'C'
Ret

tes_huruf_3:
Cjne   A, #3Dh, tes_huruf_4
Mov    A, #'D'
Ret

tes_huruf_4:
Cjne   A, #3Eh, tes_huruf_5
Mov    A, #'E'
Ret

tes_huruf_5:
Cjne   A, #3Fh, tes_huruf_6
Mov    A, #'F'
Ret

tes_huruf_6:
Ret
Ret

```

delayL:

```

Push   0
Push   1
Push   2

Mov    2, #80
tundalqqqa:
Mov    0, #10
Djnz  0, $;tundalqqqa
Mov    1, #30
Djnz  1, $
Mov    1, #1
Djnz  1, $
Djnz  2, tundalqqqa

Pop    2
Pop    1

```

```

        Pop      0
        Ret

Get_DPTR_Arus:
        Push     B
        Mov      DPTR,#txt_arus0
        Mov      B,#5           ; jumlah antar lokasi nama
variabel = 12
        Mul     AB            ; kalikan a dengan b
        Add     A,DPL          ; jumlahkan a dengan dpl
        Mov     DPL,A          ; simpan ke dpl
        Mov     A,B            ; ambil b
        Addc   A,DPH
        Mov     DPH,A
        Pop     B
        Ret

Get_DPTR_ArusSetting:
        Push    B
        Mov     DPTR,#txt_arusSetting5
        Mov     B,#1           ; jumlah antar lokasi nama
variabel = 12
        Mul     AB            ; kalikan a dengan b
        Add     A,DPL          ; jumlahkan a dengan dpl
        Mov     DPL,A          ; simpan ke dpl
        Mov     A,B            ; ambil b
        Addc   A,DPH
        Mov     DPH,A
        Pop     B
        Ret

txt_arus0:  Db     '0,53',0
txt_arus1:  Db     '0,55',0
txt_arus2:  Db     '0,57',0
txt_arus3:  Db     '0,59',0
txt_arus4:  Db     '0,61',0
txt_arus5:  Db     '0,63',0
txt_arus6:  Db     '0,65',0
txt_arus7:  Db     '0,67',0
txt_arus8:  Db     '0,69',0
txt_arus9:  Db     '0,71',0
txt_arus10:  Db    '0,73',0
txt_arus11:  Db    '0,75',0
txt_arus12:  Db    '0,77',0
txt_arus13:  Db    '0,79',0
txt_arus14:  Db    '0,81',0
txt_arus15:  Db    '0,83',0
txt_arus16:  Db    '0,85',0
txt_arus17:  Db    '0,87',0
txt_arus18:  Db    '0,89',0
txt_arus19:  Db    '0,91',0
txt_arus20:  Db    '0,93',0
txt_arus21:  Db    '0,95',0
txt_arus22:  Db    '0,98',0
txt_arus23:  Db    '1,00',0
txt_arus24:  Db    '1,02',0
txt_arus25:  Db    '1,04',0

```

txt_arus26:	Db	'1,06',0
txt_arus27:	Db	'1,08',0
txt_arus28:	Db	'1,10',0
txt_arus29:	Db	'1,12',0
txt_arus30:	Db	'1,14',0
txt_arus31:	Db	'1,17',0
txt_arus32:	Db	'1,19',0
txt_arus33:	Db	'1,21',0
txt_arus34:	Db	'1,23',0
txt_arus35:	Db	'1,25',0
txt_arus36:	Db	'1,27',0
txt_arus37:	Db	'1,30',0
txt_arus38:	Db	'1,32',0
txt_arus39:	Db	'1,34',0
txt_arus40:	Db	'1,36',0
txt_arus41:	Db	'1,39',0
txt_arus42:	Db	'1,41',0
txt_arus43:	Db	'1,43',0
txt_arus44:	Db	'1,45',0
txt_arus45:	Db	'1,48',0
txt_arus46:	Db	'1,50',0
txt_arus47:	Db	'1,52',0
txt_arus48:	Db	'1,54',0
txt_arus49:	Db	'1,57',0
txt_arus50:	Db	'1,59',0
txt_arus51:	Db	'1,61',0
txt_arus52:	Db	'1,64',0
txt_arus53:	Db	'1,66',0
txt_arus54:	Db	'1,68',0
txt_arus55:	Db	'1,71',0
txt_arus56:	Db	'1,73',0
txt_arus57:	Db	'1,76',0
txt_arus58:	Db	'1,78',0
txt_arus59:	Db	'1,80',0
txt_arus60:	Db	'1,83',0
txt_arus61:	Db	'1,85',0
txt_arus62:	Db	'1,88',0
txt_arus63:	Db	'1,90',0
txt_arus64:	Db	'1,93',0
txt_arus65:	Db	'1,95',0
txt_arus66:	Db	'1,98',0
txt_arus67:	Db	'2,00',0
txt_arus68:	Db	'2,03',0
txt_arus69:	Db	'2,05',0
txt_arus70:	Db	'2,08',0
txt_arus71:	Db	'2,11',0
txt_arus72:	Db	'2,13',0
txt_arus73:	Db	'2,16',0
txt_arus74:	Db	'2,18',0
txt_arus75:	Db	'2,21',0
txt_arus76:	Db	'2,24',0
txt_arus77:	Db	'2,26',0
txt_arus78:	Db	'2,29',0
txt_arus79:	Db	'2,32',0
txt_arus80:	Db	'2,35',0
txt_arus81:	Db	'2,37',0

txt_arus82:	Db	'2,40',0
txt_arus83:	Db	'2,43',0
txt_arus84:	Db	'2,46',0
txt_arus85:	Db	'2,48',0
txt_arus86:	Db	'2,51',0
txt_arus87:	Db	'2,54',0
txt_arus88:	Db	'2,57',0
txt_arus89:	Db	'2,60',0
txt_arus90:	Db	'2,63',0
txt_arus91:	Db	'2,66',0
txt_arus92:	Db	'2,69',0
txt_arus93:	Db	'2,72',0
txt_arus94:	Db	'2,75',0
txt_arus95:	Db	'2,78',0
txt_arus96:	Db	'2,81',0
txt_arus97:	Db	'2,84',0
txt_arus98:	Db	'2,87',0
txt_arus99:	Db	'2,90',0
txt_arus100:	Db	'2,93',0
txt_arus101:	Db	'2,96',0
txt_arus102:	Db	'2,99',0
txt_arus103:	Db	'3,02',0
txt_arus104:	Db	'3,05',0
txt_arus105:	Db	'3,09',0
txt_arus106:	Db	'3,12',0
txt_arus107:	Db	'3,15',0
txt_arus108:	Db	'3,18',0
txt_arus109:	Db	'3,21',0
txt_arus110:	Db	'3,25',0
txt_arus111:	Db	'3,28',0
txt_arus112:	Db	'3,31',0
txt_arus113:	Db	'3,35',0
txt_arus114:	Db	'3,38',0
txt_arus115:	Db	'3,42',0
txt_arus116:	Db	'3,45',0
txt_arus117:	Db	'3,48',0
txt_arus118:	Db	'3,52',0
txt_arus119:	Db	'3,55',0
txt_arus120:	Db	'3,59',0
txt_arus121:	Db	'3,63',0
txt_arus122:	Db	'3,66',0
txt_arus123:	Db	'3,70',0
txt_arus124:	Db	'3,73',0
txt_arus125:	Db	'3,77',0
txt_arus126:	Db	'3,81',0
txt_arus127:	Db	'3,84',0
txt_arus128:	Db	'3,88',0
txt_arus129:	Db	'3,92',0
txt_arus130:	Db	'3,96',0
txt_arus131:	Db	'3,99',0
txt_arus132:	Db	'4,03',0
txt_arus133:	Db	'4,07',0
txt_arus134:	Db	'4,11',0
txt_arus135:	Db	'4,15',0
txt_arus136:	Db	'4,19',0
txt_arus137:	Db	'4,23',0

txt_arus138:	Db	'4,27',0
txt_arus139:	Db	'4,31',0
txt_arus140:	Db	'4,35',0
txt_arus141:	Db	'4,39',0
txt_arus142:	Db	'4,43',0
txt_arus143:	Db	'4,47',0
txt_arus144:	Db	'4,51',0
txt_arus145:	Db	'4,55',0
txt_arus146:	Db	'4,59',0
txt_arus147:	Db	'4,64',0
txt_arus148:	Db	'4,68',0
txt_arus149:	Db	'4,72',0
txt_arus150:	Db	'4,77',0
txt_arus151:	Db	'4,81',0
txt_arus152:	Db	'4,85',0
txt_arus153:	Db	'4,90',0
txt_arus154:	Db	'4,94',0
txt_arus155:	Db	'4,99',0
txt_arus156:	Db	'5,03',0
txt_arus157:	Db	'5,08',0
txt_arus158:	Db	'5,12',0
txt_arus159:	Db	'5,17',0
txt_arus160:	Db	'5,21',0
txt_arus161:	Db	'5,26',0
txt_arus162:	Db	'5,31',0
txt_arus163:	Db	'5,35',0
txt_arus164:	Db	'5,40',0
txt_arus165:	Db	'5,45',0
txt_arus166:	Db	'5,50',0
txt_arus167:	Db	'5,54',0
txt_arus168:	Db	'5,59',0
txt_arus169:	Db	'5,64',0
txt_arus170:	Db	'5,69',0
txt_arus171:	Db	'5,74',0
txt_arus172:	Db	'5,79',0
txt_arus173:	Db	'5,84',0
txt_arus174:	Db	'5,89',0
txt_arus175:	Db	'5,94',0
txt_arus176:	Db	'5,99',0
txt_arus177:	Db	'6,04',0
txt_arus178:	Db	'6,10',0
txt_arus179:	Db	'6,15',0
txt_arus180:	Db	'6,20',0
txt_arus181:	Db	'6,25',0
txt_arus182:	Db	'6,31',0
txt_arus183:	Db	'6,36',0
txt_arus184:	Db	'6,42',0
txt_arus185:	Db	'6,47',0
txt_arus186:	Db	'6,52',0
txt_arus187:	Db	'6,58',0
txt_arus188:	Db	'6,63',0
txt_arus189:	Db	'6,69',0
txt_arus190:	Db	'6,75',0
txt_arus191:	Db	'6,80',0
txt_arus192:	Db	'6,86',0
txt_arus193:	Db	'6,92',0

txt_arus194:	Db	'6,97',0
txt_arus195:	Db	'7,03',0
txt_arus196:	Db	'7,09',0
txt_arus197:	Db	'7,15',0
txt_arus198:	Db	'7,21',0
txt_arus199:	Db	'7,27',0
txt_arus200:	Db	'7,33',0
txt_arus201:	Db	'7,39',0
txt_arus202:	Db	'7,45',0
txt_arus203:	Db	'7,51',0
txt_arus204:	Db	'7,57',0
txt_arus205:	Db	'7,63',0
txt_arus206:	Db	'7,69',0
txt_arus207:	Db	'7,76',0
txt_arus208:	Db	'7,82',0
txt_arus209:	Db	'7,88',0
txt_arus210:	Db	'7,95',0
txt_arus211:	Db	'8,01',0
txt_arus212:	Db	'8,08',0
txt_arus213:	Db	'8,14',0
txt_arus214:	Db	'8,21',0
txt_arus215:	Db	'8,27',0
txt_arus216:	Db	'8,34',0
txt_arus217:	Db	'8,40',0
txt_arus218:	Db	'8,47',0
txt_arus219:	Db	'8,54',0
txt_arus220:	Db	'8,61',0
txt_arus221:	Db	'8,67',0
txt_arus222:	Db	'8,74',0
txt_arus223:	Db	'8,81',0
txt_arus224:	Db	'8,88',0
txt_arus225:	Db	'8,95',0
txt_arus226:	Db	'9,02',0
txt_arus227:	Db	'9,09',0
txt_arus228:	Db	'9,16',0
txt_arus229:	Db	'9,23',0
txt_arus230:	Db	'9,31',0
txt_arus231:	Db	'9,38',0
txt_arus232:	Db	'9,45',0
txt_arus233:	Db	'9,53',0
txt_arus234:	Db	'9,60',0
txt_arus235:	Db	'9,67',0
txt_arus236:	Db	'9,75',0
txt_arus237:	Db	'9,82',0
txt_arus238:	Db	'9,90',0
txt_arus239:	Db	'9,97',0
txt_arus240:	Db	'10,0',0
txt_arus241:	Db	'10,1',0
txt_arus242:	Db	'10,2',0
txt_arus243:	Db	'10,2',0
txt_arus244:	Db	'10,3',0
txt_arus245:	Db	'10,4',0
txt_arus246:	Db	'10,5',0
txt_arus247:	Db	'10,6',0
txt_arus248:	Db	'10,6',0
txt_arus249:	Db	'10,7',0

txt_arus250:	Db	'10,8',0
txt_arus251:	Db	'10,9',0
txt_arus252:	Db	'11,0',0
txt_arus253:	Db	'11,0',0
txt_arus254:	Db	'11,1',0
txt_arus255:	Db	'11,2',0
txt_arusSeting5:	Db	0
txt_arusSeting6:	Db	4
txt_arusSeting7:	Db	9
txt_arusSeting8:	Db	14
txt_arusSeting9:	Db	19
txt_arusSeting10:	Db	23
txt_arusSeting11:	Db	28
txt_arusSeting12:	Db	33
txt_arusSeting13:	Db	37
txt_arusSeting14:	Db	42
txt_arusSeting15:	Db	46
txt_arusSeting16:	Db	51
txt_arusSeting17:	Db	55
txt_arusSeting18:	Db	59
txt_arusSeting19:	Db	63
txt_arusSeting20:	Db	67
txt_arusSeting21:	Db	71
txt_arusSeting22:	Db	75
txt_arusSeting23:	Db	79
txt_arusSeting24:	Db	82
txt_arusSeting25:	Db	86
txt_arusSeting26:	Db	89
txt_arusSeting27:	Db	93
txt_arusSeting28:	Db	96
txt_arusSeting29:	Db	99
txt_arusSeting30:	Db	103
txt_arusSeting31:	Db	106
txt_arusSeting32:	Db	109
txt_arusSeting33:	Db	112
txt_arusSeting34:	Db	115
txt_arusSeting35:	Db	118
txt_arusSeting36:	Db	121
txt_arusSeting37:	Db	123
txt_arusSeting38:	Db	126
txt_arusSeting39:	Db	129
txt_arusSeting40:	Db	132
txt_arusSeting41:	Db	134
txt_arusSeting42:	Db	137
txt_arusSeting43:	Db	139
txt_arusSeting44:	Db	142
txt_arusSeting45:	Db	144
txt_arusSeting46:	Db	147
txt_arusSeting47:	Db	149
txt_arusSeting48:	Db	151
txt_arusSeting49:	Db	153
txt_arusSeting50:	Db	156
txt_arusSeting51:	Db	158
txt_arusSeting52:	Db	160
txt_arusSeting53:	Db	162
txt_arusSeting54:	Db	164

txt_arusSeting55:	Db	166
txt_arusSeting56:	Db	169
txt_arusSeting57:	Db	171
txt_arusSeting58:	Db	173
txt_arusSeting59:	Db	175
txt_arusSeting60:	Db	177
txt_arusSeting61:	Db	178
txt_arusSeting62:	Db	180
txt_arusSeting63:	Db	182
txt_arusSeting64:	Db	184
txt_arusSeting65:	Db	186
txt_arusSeting66:	Db	188
txt_arusSeting67:	Db	190
txt_arusSeting68:	Db	191
txt_arusSeting69:	Db	193
txt_arusSeting70:	Db	195
txt_arusSeting71:	Db	197
txt_arusSeting72:	Db	198
txt_arusSeting73:	Db	200
txt_arusSeting74:	Db	202
txt_arusSeting75:	Db	203
txt_arusSeting76:	Db	205
txt_arusSeting77:	Db	207
txt_arusSeting78:	Db	208
txt_arusSeting79:	Db	210
txt_arusSeting80:	Db	211
txt_arusSeting81:	Db	213
txt_arusSeting82:	Db	214
txt_arusSeting83:	Db	216
txt_arusSeting84:	Db	217
txt_arusSeting85:	Db	219
txt_arusSeting86:	Db	220
txt_arusSeting87:	Db	222
txt_arusSeting88:	Db	223
txt_arusSeting89:	Db	225
txt_arusSeting90:	Db	226
txt_arusSeting91:	Db	228
txt_arusSeting92:	Db	229
txt_arusSeting93:	Db	230
txt_arusSeting94:	Db	232
txt_arusSeting95:	Db	233
txt_arusSeting96:	Db	234
txt_arusSeting97:	Db	236
txt_arusSeting98:	Db	237
txt_arusSeting99:	Db	238
txt_arusSeting100:	Db	240
txt_arusSeting101:	Db	241
txt_arusSeting102:	Db	242
txt_arusSeting103:	Db	244
txt_arusSeting104:	Db	245
txt_arusSeting105:	Db	246
txt_arusSeting106:	Db	247
txt_arusSeting107:	Db	249
txt_arusSeting108:	Db	250
txt_arusSeting109:	Db	251
txt_arusSeting110:	Db	252

txt_arusSeting111: Db 254
txt_arusSeting112: Db 255
i

ii

Features

Compatible with MCS-51™ Products

8K Bytes of In-System Reprogrammable Downloadable Flash Memory

- SPI Serial Interface for Program Downloading

- Endurance: 1,000 Write/Erase Cycles

256 Bytes EEPROM

- Endurance: 100,000 Write/Erase Cycles

2.7V to 6V Operating Range

Full Static Operation: 0 Hz to 24 MHz

Three-level Program Memory Lock

16 x 8-bit Internal RAM

32 Programmable I/O Lines

Three 16-bit Timer/Counters

One Interrupt Sources

Programmable UART Serial Channel

PI Serial Interface

Low-power Idle and Power-down Modes

Interrupt Recovery From Power-down

Programmable Watchdog Timer

Two Data Pointers

Power-off Flag

Description

AT89S8252 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of downloadable Flash programmable and erasable read only memory and 256 bytes of EEPROM. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip downloadable Flash allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU and downloadable Flash on a monolithic chip, the Atmel AT89S8252 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

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 disables the oscillator, disabling all other chip functions until the next 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. Flash Lock Bit 2 has been activated.



8-bit Microcontroller with 8K Bytes Flash

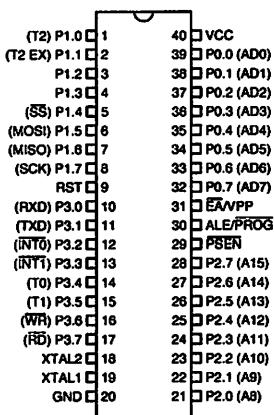
AT89S8252



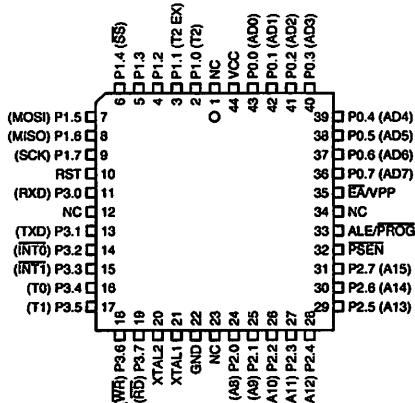


Pin Configurations

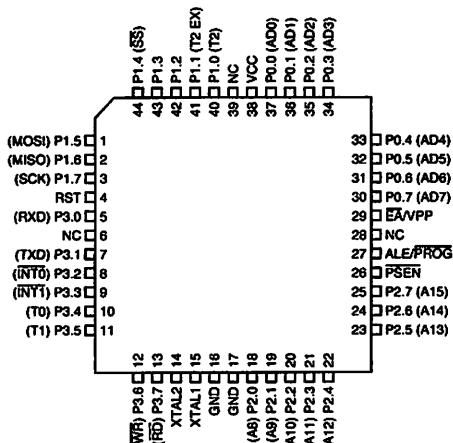
PDIP



PLCC



PQFP/TQFP



Pin Description

C
Supply voltage.

D
und.

t 0

t 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.

t 0 can also be configured to be the multiplexed lower address/data bus during accesses to external

program and data memory. In this mode, P0 has internal pullups.

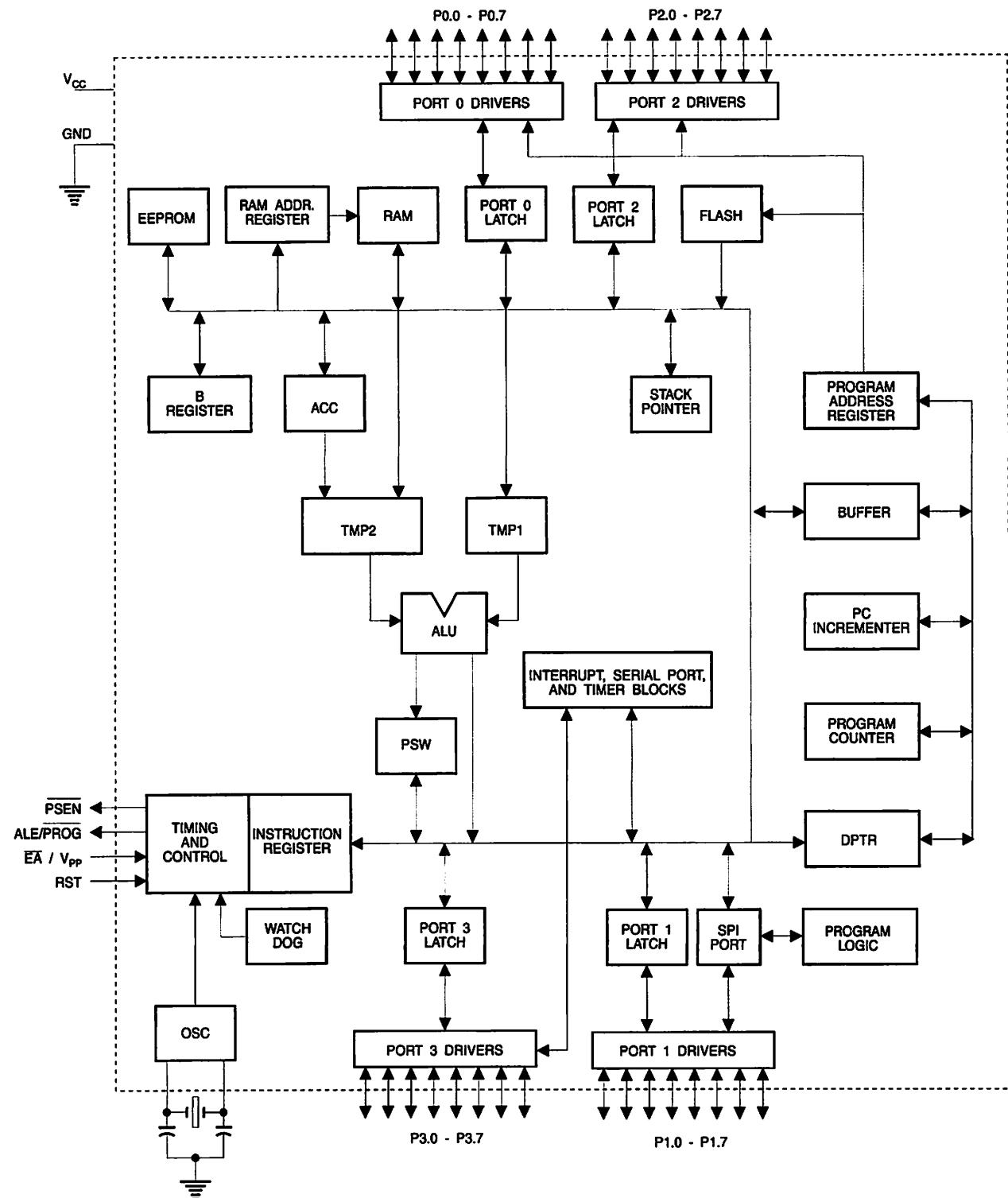
Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

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

AT89S8252

Block Diagram





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

n Description

thermore, P1.4, P1.5, P1.6, and P1.7 can be configured the SPI slave port select, data input/output and shift ck 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	\overline{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)

rt 1 also receives the low-order address bytes during sh programming and verification.

t 2

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

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

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

t 3

rt 3 is an 8 bit bi-directional I/O port with internal pullups. e Port 3 output buffers can sink/source four TTL inputs. en 1s are written to Port 3 pins, they are pulled high by internal pullups and can be used as inputs. As inputs,

Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

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

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

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

RST

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

ALE/PROG

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

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

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

PSEN

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

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

EA/VPP

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external pro

AT89S8252

in memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset.

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

Table 1. AT89S8252 SFR Map and Reset Values

F8H								0FFH
F0H	B 00000000							0F7H
E8H								0EFH
E0H	ACC 00000000							0E7H
D8H								0DFH
D0H	PSW 00000000					SPCR 000001XX		0D7H
C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000		0CFH
C0H								0C7H
B8H	IP XX000000							0BFH
B0H	P3 11111111							0B7H
A8H	IE 0X000000		SPSR 00XXXXXX					0AFH
A0H	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	DPCL 00000000	DPOH 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000
								87H



Special Function Registers

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.

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 2. T2CON—Timer/Counter 2 Control Register

T2CON Address = 0C8H								Reset Value = 0000 0000B
Bit Addressable								
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
PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN	
7	6	5	4	3	2	1	0	
Symbol	Function							
S2	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.							
S1								
S0								
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.							
PS	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1							
WDTRST	Watchdog Timer Reset and EEPROM Ready/Busy Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The WDTRST bit is then automatically reset to "0" in the next instruction cycle. The RDY/BSY 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.							
RDY/BSY								
WDTEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.							

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 WCOL, in the SPSR register. The SPDR is double buffered for writing and the values in SPDR are not changed by set.

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 SPCR register. Two priorities can be set for each of the 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.



Ie 4. SPCR—SPI Control Register

'CR Address = D5H Reset Value = 0000 01XXB

SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
it 7	6	5	4	3	2	1	0

mbol	Function												
'IE	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.												
'E	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: SPR1SPR0 SCK = F_{osc} divided by <table> <tr> <td>0</td> <td>0</td> <td>4</td> </tr> <tr> <td>0</td> <td>1</td> <td>16</td> </tr> <tr> <td>1</td> <td>0</td> <td>64</td> </tr> <tr> <td>1</td> <td>1</td> <td>128</td> </tr> </table>	0	0	4	0	1	16	1	0	64	1	1	128
0	0	4											
0	1	16											
1	0	64											
1	1	128											

Ie 5. SPSR – SPI Status Register

'SR Address = AAH Reset Value = 00XX XXXXB

SPIF	WCOL	-	-	-	-	-	-
it 7	6	5	4	3	2	1	0

mbol	Function
'IF	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 accessing 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.

Ie 6. SPDR – SPI Data Register

'DR Address = 86H Reset Value = unchanged

SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
it 7	6	5	4	3	2	1	0

AT89S8252

Data Memory – EEPROM and RAM

The AT89S8252 implements 2K bytes of on-chip EEPROM 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 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.

On-chip EEPROM data memory is selected by setting the EEMEN bit in the WMCON register at SFR address 96H. The EEPROM address range is from 000H to 7FH. 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. The software should reset EEMWE bit to "0" if no further EEPROM write is required. EEPROM write cycles in the final programming mode are self-timed and typically take 10 ms. The progress of EEPROM write can be monitored by reading the RDY/BSY bit (read-only) in SFR WMCON. RDY/BSY = 0 means programming is still in progress and RDY/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 in 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 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 ±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, AT89C52 and AT89C55. For further information, see the October 1995 Microcontroller Data Book, page 2-45, section titled, "Timer/Counters."

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



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.

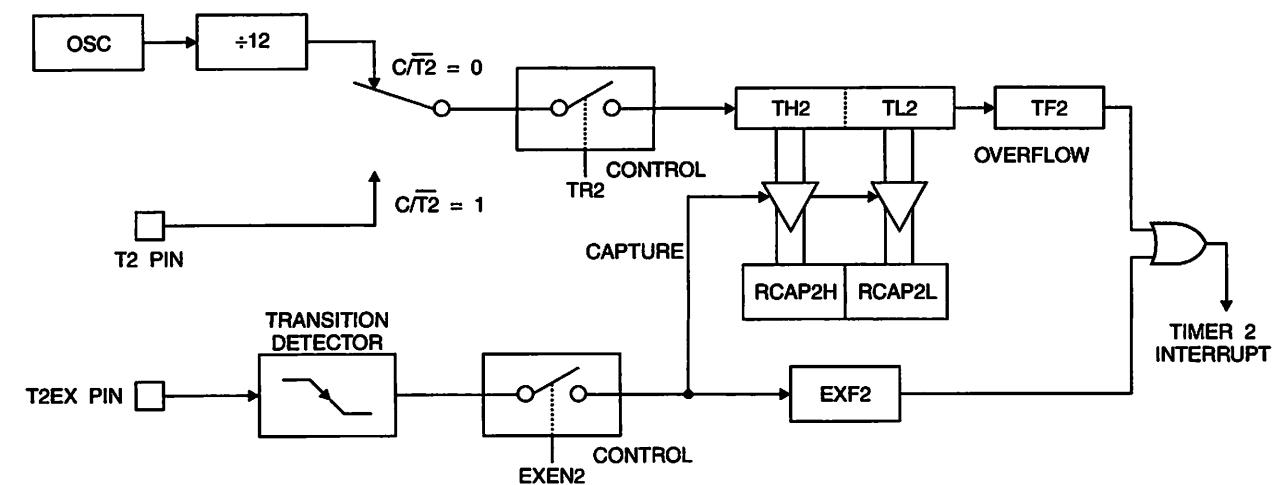
Table 8. Timer 2 Operating Modes

ICLK + 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 selected by the DCEN (Down Counter Enable) bit located in 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 EXEN2 = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with a 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)

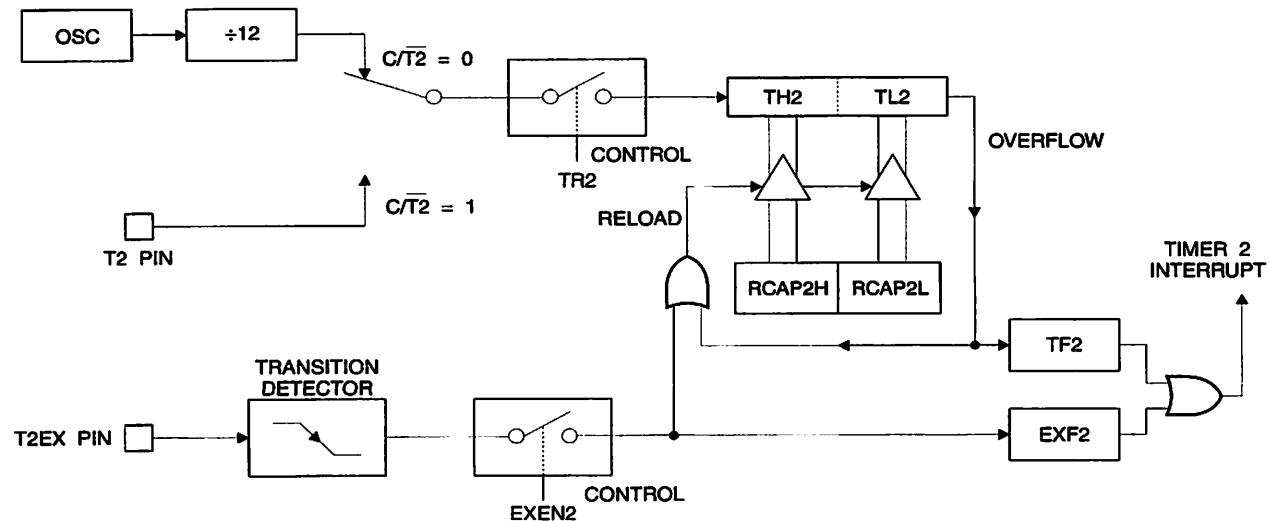


Table 9. T2MOD – Timer 2 Mode Control Register

MOD Address = 0C9H								Reset Value = XXXX XX00B	
Bit Addressable									
Bit	-	-	-	-	-	-	T2OE	DCEN	
7	6	5	4	3	2	1	1	0	
Symbol									
Function									
Not implemented, reserved for future use.									
OE	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)

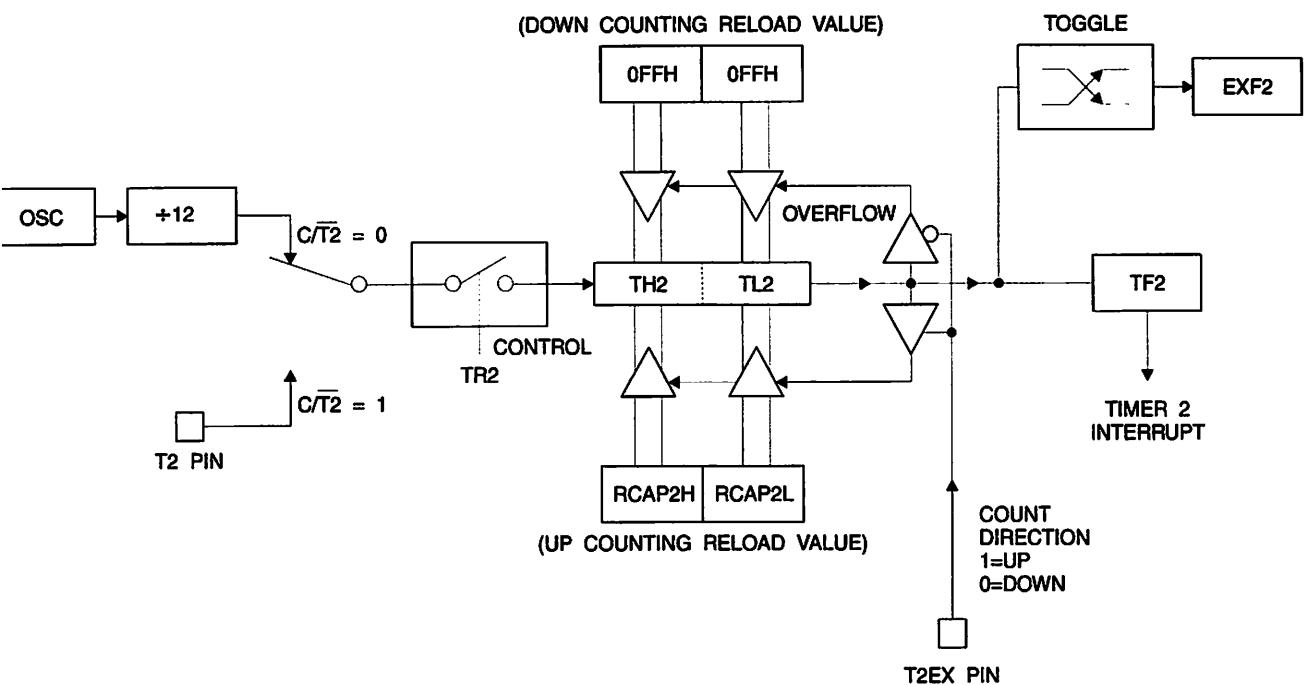
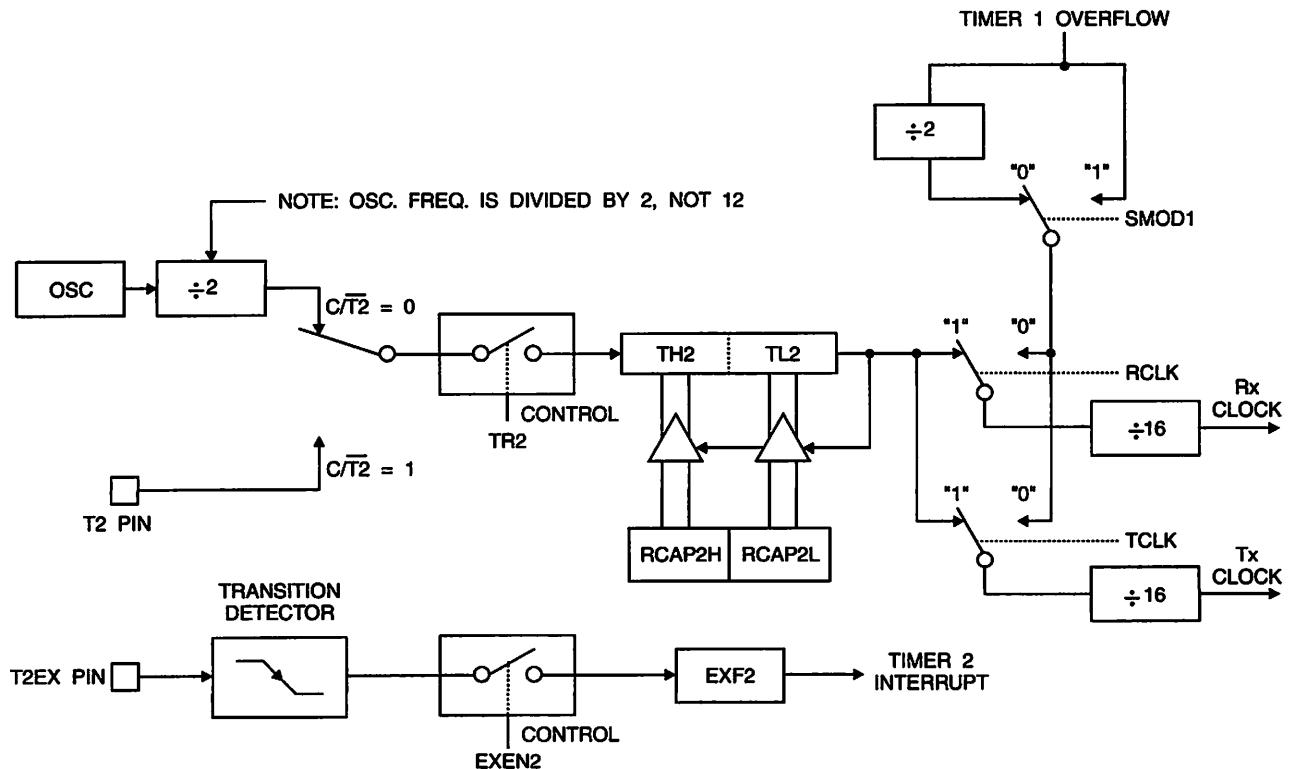


Figure 4. Timer 2 in Baud Rate Generator Mode



Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting T2K and/or RCLK in T2CON (Table 2). Note that the baud rates for transmit and receive can be different if Timer 2 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.

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

Baud rates in Modes 1 and 3 are determined by Timer 2 overflow rate according to the following equation.

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

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

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

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

Timer 2 as a baud rate generator is shown in Figure 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 ($\text{TR2} = 1$) 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 at a 16 MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/T2 (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.

AT&T

Figure 5. Timer 2 in Clock-out Mode

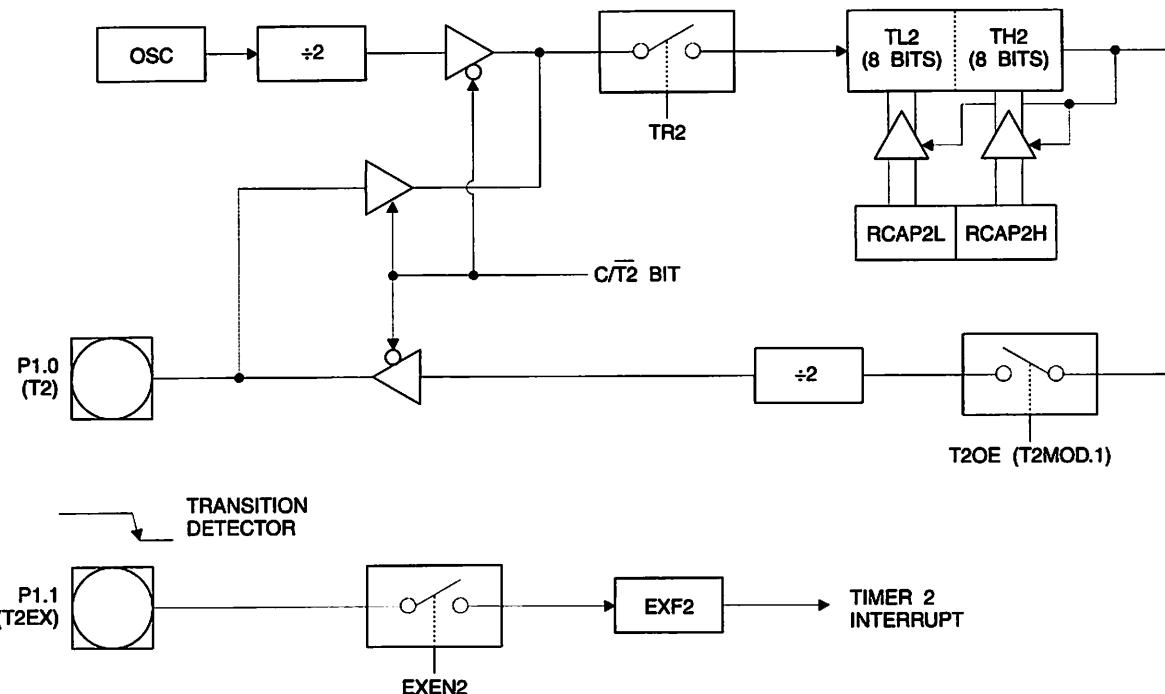
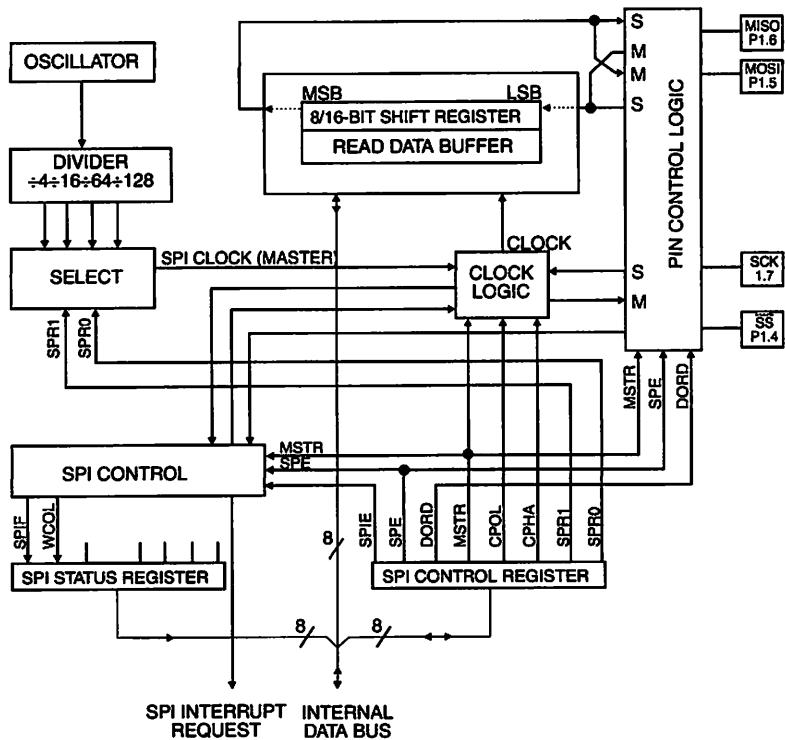


Figure 6. SPI Block Diagram



AT89S8252

UART

The UART in the AT89S8252 operates the same way as the UART in the AT89C51, AT89C52 and AT89C55. For further information, see the October 1995 Microcontroller Application Book, page 2-49, section titled, "Serial Interface."

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

- .5 MHz Bit Frequency (max.)

- MSB First or LSB 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, SS/P1.4, is set low to select an individual SPI device as a slave. When 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

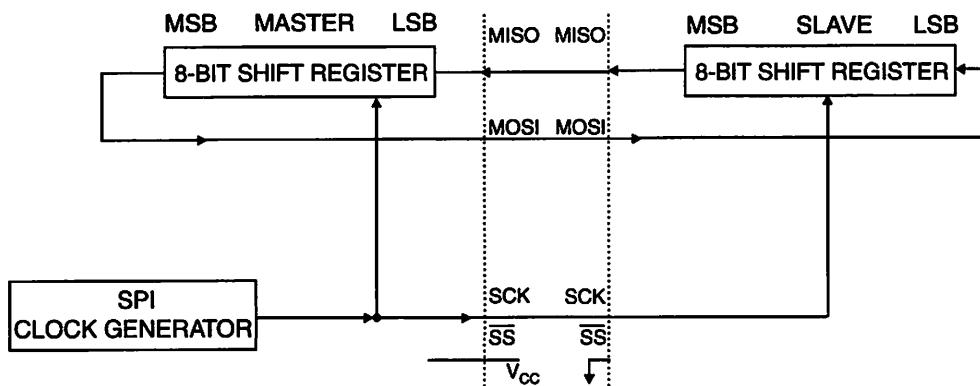
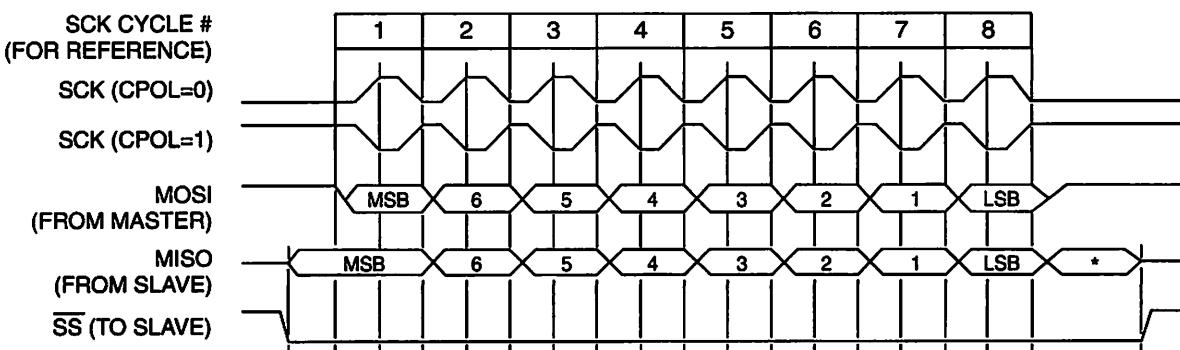


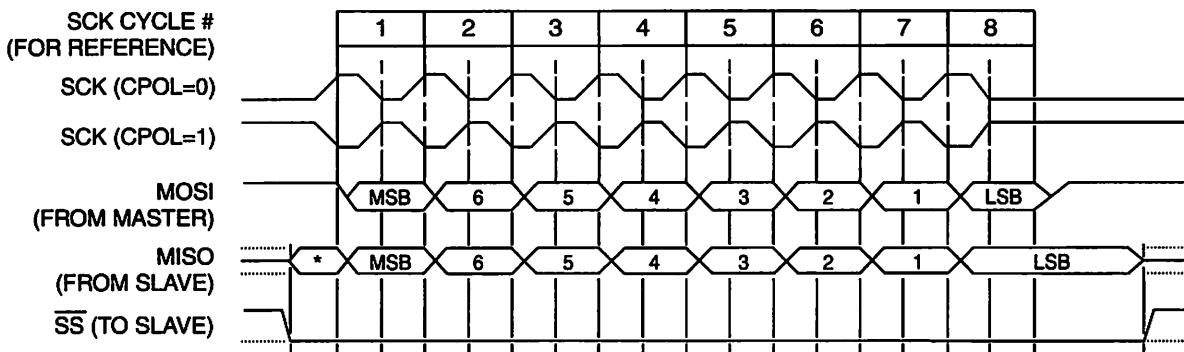
Figure 8. SPI transfer Format with CPHA = 0



*not defined but normally MSB of character just received



Figure 9. SPI Transfer Format with CPHA = 1



* undefined but normally LSB of previously transmitted character

Interrupts

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

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE. IE also contains a global disable bit, EA, which enables 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 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. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, so that bit will have to be cleared in software.

Timer 0 and Timer 1 flags, TF0 and TF1, are set at S2P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle. However, 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

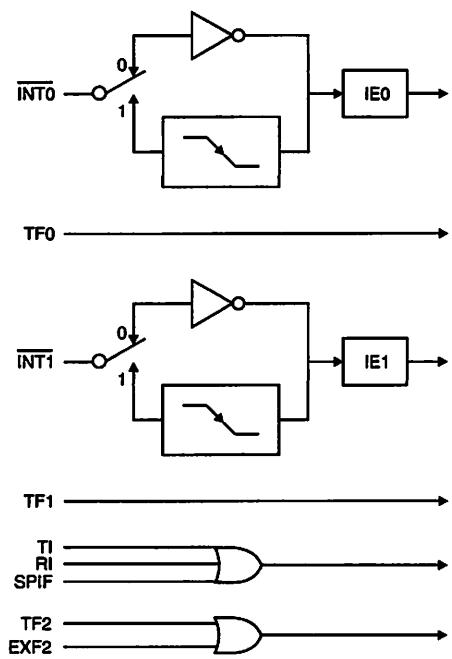
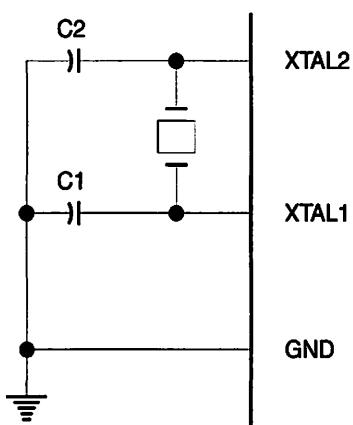
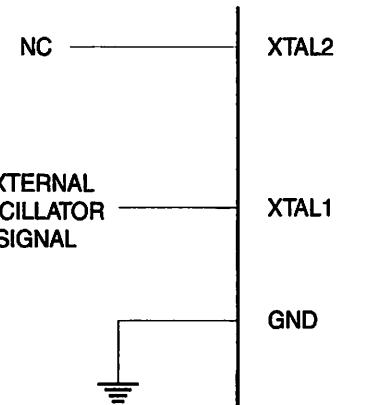


Figure 11. Oscillator Connections



Note: Note: $C_1, C_2 = 30 \text{ pF} \pm 10 \text{ pF}$ for Crystals
 $= 40 \text{ pF} \pm 10 \text{ pF}$ for Ceramic Resonators

Figure 12. External Clock Drive Configuration



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 passes through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.



Idle Mode

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

Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution

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

Status of External Pins During Idle and Power-down Modes

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

Power-down Mode

In power-down mode, the oscillator is stopped and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. Exit from power-down can be initiated either by hardware reset or by an enabled external interrupt. Reset defines the SFRs but does not change the on-chip RAM. A reset should not be activated before V_{CC} is restored to 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 at 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⁽¹⁾⁽²⁾

Program Lock Bits			Protection Type
LB1	LB2	LB3	
1	U	U	U
2	P	U	U
3	P	P	U
4	P	P	P

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

AT89S8252

Programming the Flash and EEPROM

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

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

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

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

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

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

Power-up sequence:

Apply power between V_{CC} and GND pins.

Set RST pin to "H".

Apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.

Set PSEN pin to "L"

ALE pin to "H"

EA pin to "H" and all other pins to "H".

Apply the appropriate combination of "H" or "L" logic levels to pins P2.6, P2.7, P3.6, P3.7 to select one of the programming operations shown in the Flash Programming Modes table.

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/PROG once to program a byte in the Code memory array, the Data memory array or the lock bits. The byte-write cycle is self-timed and typically takes 1.5 ms.
7. To verify the byte just programmed, bring pin P2.7 to "L" and read the programmed data at pins P0.0 to P0.7.
8. Repeat steps 3 through 7 changing the address and data for the entire 2K or 8K bytes array or until the end of the object file is reached.
9. Power-off sequence:
Set XTAL1 to "L".
Set RST and \overline{EA} pins to "L".
Turn V_{CC} power off.

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

Data Polling: The AT89S8252 features DATA Polling to indicate the end of a write cycle. During a write cycle in the parallel or serial programming mode, an attempted read of the last byte written will result in the complement of the written datum on P0.7 (parallel mode), and on the MSB of the serial output byte on MISO (serial mode). Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. 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 issuing the Chip Erase instruction. In this mode, chip erase is self-timed and takes about 16 ms.

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

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

The AT89S8252 is shipped with the Serial Programming mode enabled.

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

(030H) = 1EH indicates manufactured by Atmel

(031H) = 72H indicates 89S8252

Programming Interface

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

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

Serial Downloading

Both the Code and Data memory arrays can be programmed using the serial SPI bus while RST is pulled to ground. 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 7FFH 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.

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:

struction Set

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

te: 1. DATA polling is used to indicate the end of a write cycle which typically takes less than 2.5 ms at 5V.

2. "aaaaa" = high order address.
3. "x" = don't care.



Flash and EEPROM Parallel Programming Modes

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

Notes: 1. "h" = weakly pulled "High" internally.

2. Chip Erase and Serial Programming Fuse require a 10 ms PROG pulse. Chip Erase needs to be performed first before reprogramming any byte with a content other than FFH.

3. P3.4 is pulled Low during programming to indicate RDY/BSY.

4. "X" = don't care

AT89S8252

Figure 13. Programming the Flash/EEPROM Memory

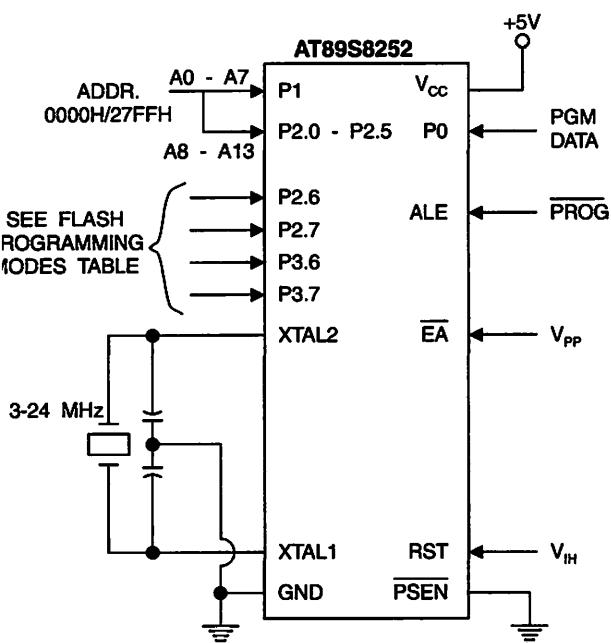


Figure 15. Flash/EEPROM Serial Downloading

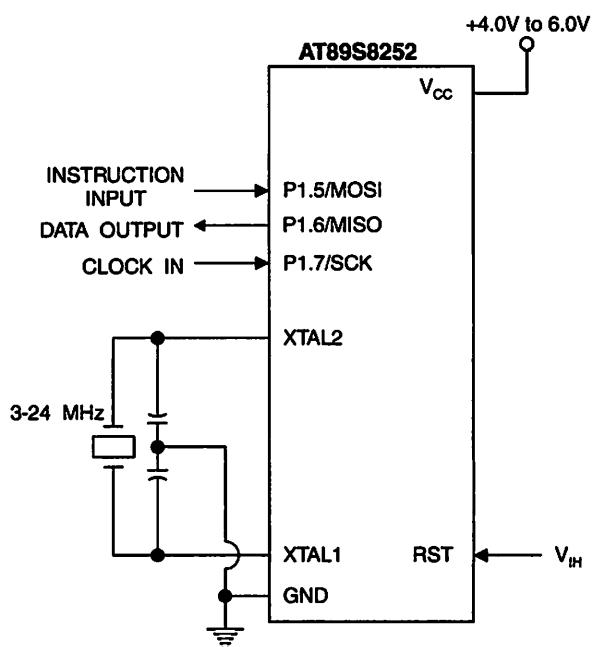
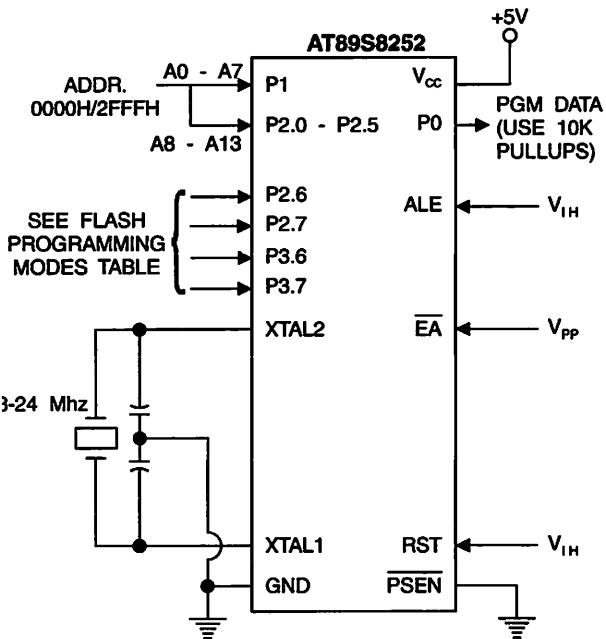


Figure 14. Verifying the Flash/EEPROM Memory





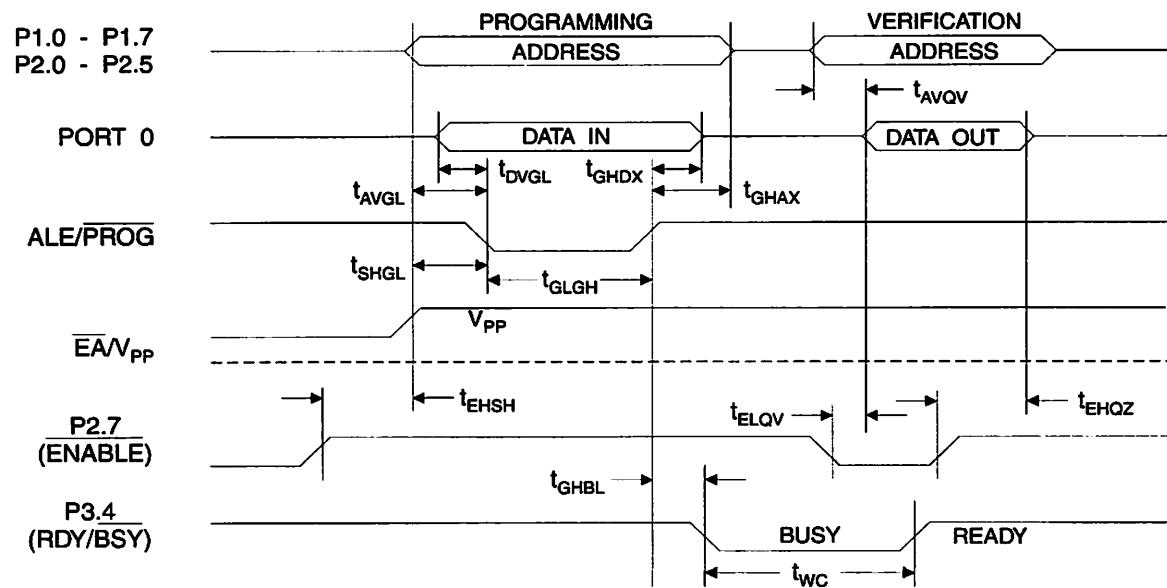
Flash Programming and Verification Characteristics – Parallel Mode

= 0°C to 70°C, V_{CC} = 5.0V ± 10%

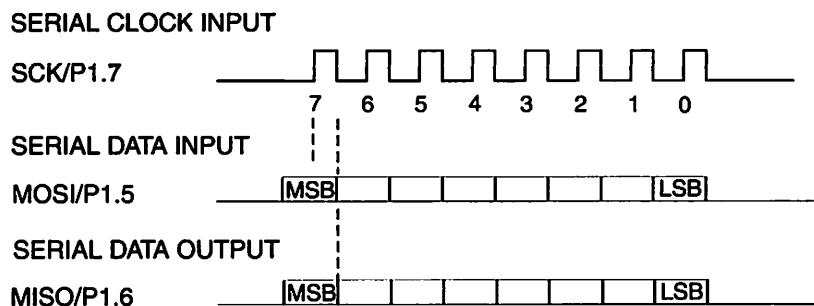
Symbol	Parameter	Min	Max	Units
V _P	Programming Enable Voltage	11.5	12.5	V
I _P	Programming Enable Current		1.0	mA
t _{CLCL}	Oscillator Frequency	3	24	MHz
t _{ASL}	Address Setup to PROG Low	48t _{CLCL}		
t _{AH}	Address Hold after PROG	48t _{CLCL}		
t _{DGL}	Data Setup to PROG Low	48t _{CLCL}		
t _{DHD}	Data Hold after PROG	48t _{CLCL}		
t _{SH}	P2.7 (ENABLE) High to V _{PP}	48t _{CLCL}		
t _{VPL}	V _{PP} Setup to PROG Low	10		μs
t _{LGH}	PROG Width	1	110	μs
t _{AV}	Address to Data Valid		48t _{CLCL}	
t _{QV}	ENABLE Low to Data Valid		48t _{CLCL}	
t _{QZ}	Data Float after ENABLE	0	48t _{CLCL}	
t _{HBL}	PROG High to BUSY Low		1.0	μs
t _c	Byte Write Cycle Time		2.0	ms

AT89S8252

Flash/EEPROM Programming and Verification Waveforms – Parallel Mode



Serial Downloading Waveforms





Absolute Maximum Ratings*

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

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

Characteristics

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

Symbol	Parameter	Condition	Min	Max	Units	
I_L	Input Low-voltage	(Except $\overline{\text{EA}}$)	-0.5	$0.2 V_{CC} - 0.1$	V	
I_{L1}	Input Low-voltage ($\overline{\text{EA}}$)		-0.5	$0.2 V_{CC} - 0.3$	V	
I_H	Input High-voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V	
I_{H1}	Input High-voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V	
I_{OL}	Output Low-voltage ⁽¹⁾ (Ports 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.5	V	
I_{OL1}	Output Low-voltage ⁽¹⁾ (Port 0, ALE, $\overline{\text{PSEN}}$)	$I_{OL} = 3.2 \text{ mA}$		0.5	V	
I_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{\text{PSEN}}$)	$I_{OH} = -60 \mu\text{A}, V_{CC} = 5\text{V} \pm 10\%$	2.4		V	
		$I_{OH} = -25 \mu\text{A}$	$0.75 V_{CC}$		V	
		$I_{OH} = -10 \mu\text{A}$	$0.9 V_{CC}$		V	
I_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	$I_{OH} = -800 \mu\text{A}, V_{CC} = 5\text{V} \pm 10\%$	2.4		V	
		$I_{OH} = -300 \mu\text{A}$	$0.75 V_{CC}$		V	
		$I_{OH} = -80 \mu\text{A}$	$0.9 V_{CC}$		V	
Logical 0 Input Current (Ports 1,2,3)		$V_{IN} = 0.45\text{V}$		-50	μA	
Logical 1 to 0 Transition Current (Ports 1,2,3)		$V_{IN} = 2\text{V}, V_{CC} = 5\text{V} \pm 10\%$		-650	μA	
Input Leakage Current (Port 0, $\overline{\text{EA}}$)		$0.45 < V_{IN} < V_{CC}$		± 10	μA	
RST	Reset Pull-down Resistor		50	300	$\text{k}\Omega$	
o	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF	
C	Power Supply Current	Active Mode, 12 MHz		25	mA	
		Idle Mode, 12 MHz		6.5	mA	
	Power-down Mode ⁽²⁾	$V_{CC} = 6\text{V}$		100	μA	
		$V_{CC} = 3\text{V}$		40	μA	

Notes: 1. Under steady state (non-transient) conditions, I_{OL}

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

must be externally limited as follows:

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.

Maximum I_{OL} per port pin: 10 mA

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

Maximum I_{OL} per 8-bit port:

Port 0: 26 mA

Ports 1, 2, 3: 15 mA

AT89S8252

C Characteristics

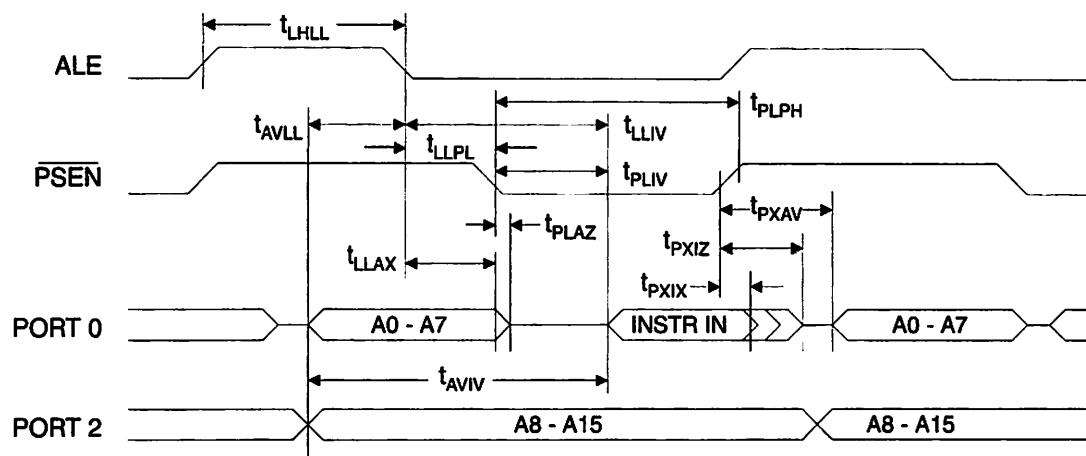
Under operating conditions, load capacitance for Port 0, ALE/PROG, and PSEN = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

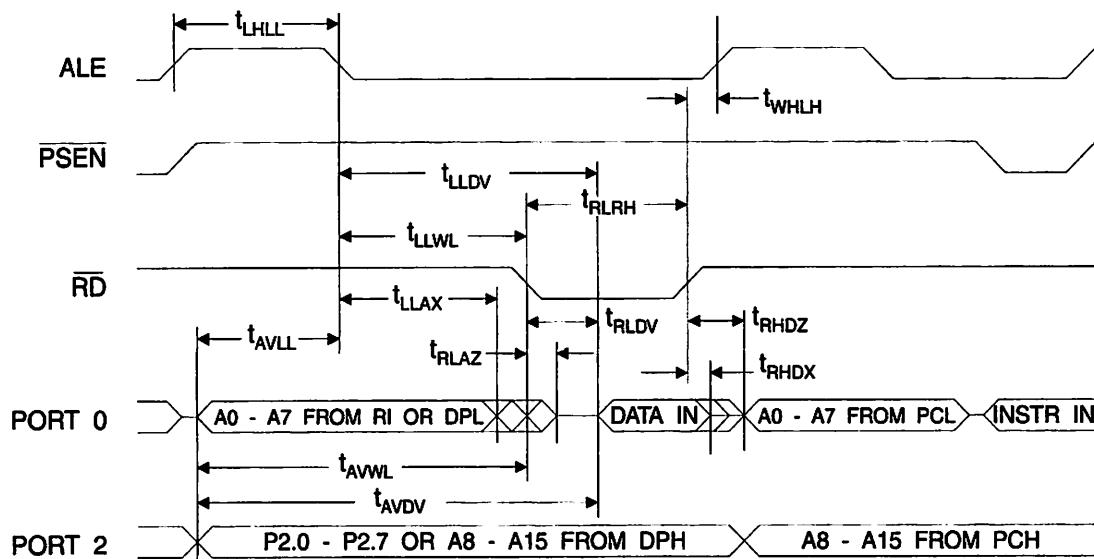
Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
t_{CLCL}	Oscillator Frequency	0	24	MHz
t_{ILL}	ALE Pulse Width	$2t_{CLCL} - 40$		ns
t_{VLL}	Address Valid to ALE Low	$t_{CLCL} - 13$		ns
t_{AX}	Address Hold after ALE Low	$t_{CLCL} - 20$		ns
t_{IV}	ALE Low to Valid Instruction In		$4t_{CLCL} - 65$	ns
t_{PL}	ALE Low to PSEN Low	$t_{CLCL} - 13$		ns
t_{LPH}	PSEN Pulse Width	$3t_{CLCL} - 20$		ns
t_{LIV}	PSEN Low to Valid Instruction In		$3t_{CLCL} - 45$	ns
t_{IX}	Input Instruction Hold after PSEN	0		ns
t_{IZ}	Input Instruction Float after PSEN		$t_{CLCL} - 10$	ns
t_{AV}	PSEN to Address Valid	$t_{CLCL} - 8$		ns
t_{IV}	Address to Valid Instruction In		$5t_{CLCL} - 55$	ns
t_{LAZ}	PSEN Low to Address Float		10	ns
t_{LRH}	RD Pulse Width	$6t_{CLCL} - 100$		ns
t_{LWH}	WR Pulse Width	$6t_{CLCL} - 100$		ns
t_{LDV}	RD Low to Valid Data In		$5t_{CLCL} - 90$	ns
t_{HDX}	Data Hold after RD	0		ns
t_{HDZ}	Data Float after RD		$2t_{CLCL} - 28$	ns
t_{DV}	ALE Low to Valid Data In		$8t_{CLCL} - 150$	ns
t_{DV}	Address to Valid Data In		$9t_{CLCL} - 165$	ns
t_{WL}	ALE Low to RD or WR Low	$3t_{CLCL} - 50$	$3t_{CLCL} + 50$	ns
t_{WL}	Address to RD or WR Low	$4t_{CLCL} - 75$		ns
t_{VWX}	Data Valid to WR Transition	$t_{CLCL} - 20$		ns
t_{VWH}	Data Valid to WR High	$7t_{CLCL} - 120$		ns
t_{HQX}	Data Hold after WR	$t_{CLCL} - 20$		ns
t_{LAZ}	RD Low to Address Float		0	ns
t_{HLH}	RD or WR High to ALE High	$t_{CLCL} - 20$	$t_{CLCL} + 25$	ns



External Program Memory Read Cycle

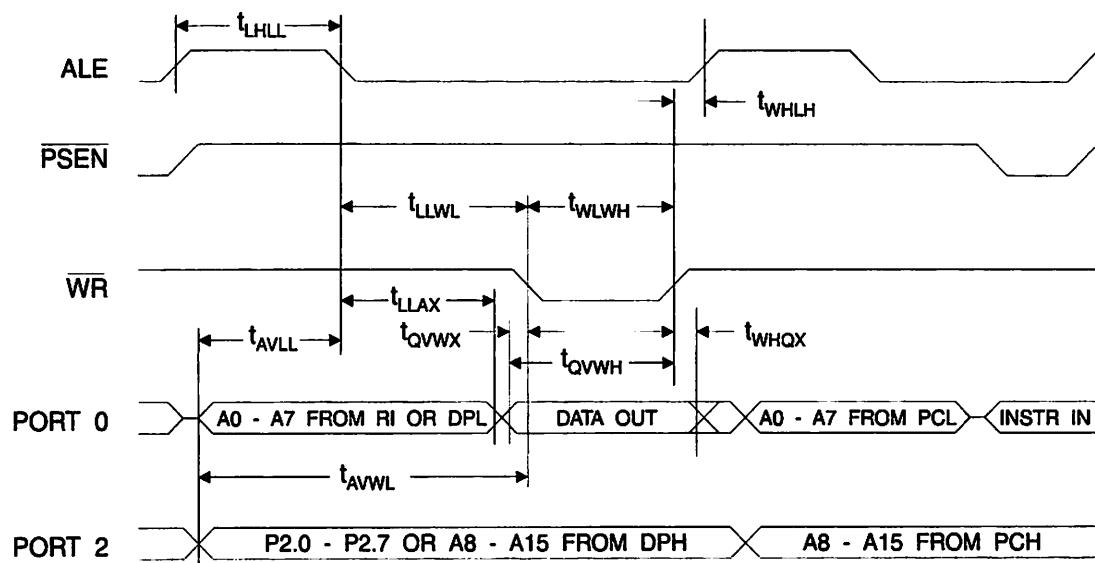


External Data Memory Read Cycle

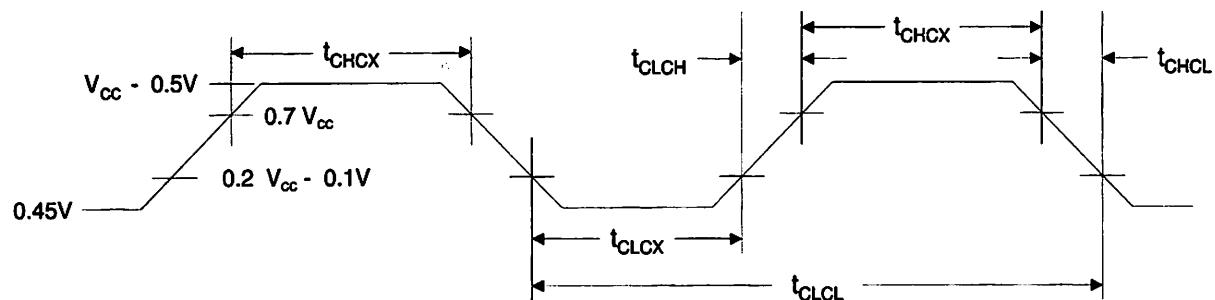


AT89S8252

External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

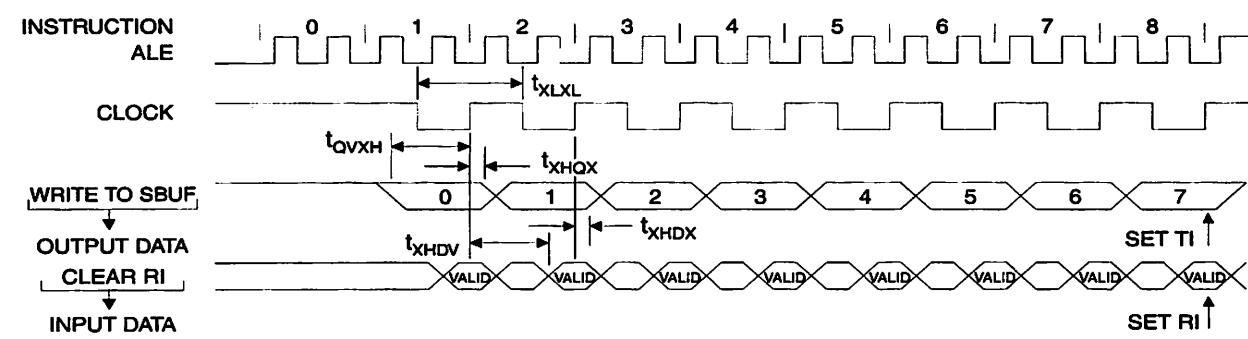
Symbol	Parameter	$V_{cc} = 4.0V \text{ to } 6.0V$		Units
		Min	Max	
t_{CLCL}	Oscillator Frequency	0	24	MHz
t_{CL}	Clock Period	41.6		ns
t_{HCX}	High Time	15		ns
t_{LCX}	Low Time	15		ns
t_{LCH}	Rise Time		20	ns
t_{HCL}	Fall Time		20	ns

Serial Port Timing: Shift Register Mode Test Conditions

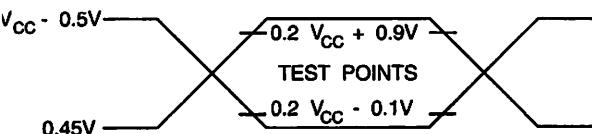
The values in this table are valid for $V_{CC} = 4.0V$ to $6V$ and Load Capacitance = 80 pF .

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

Shift Register Mode Timing Waveforms

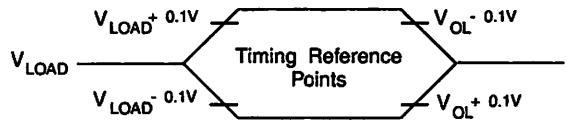


C Testing Input/Output Waveforms⁽¹⁾



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

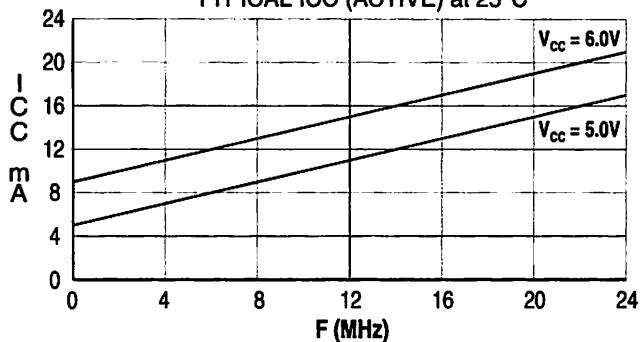
Float Waveforms⁽¹⁾



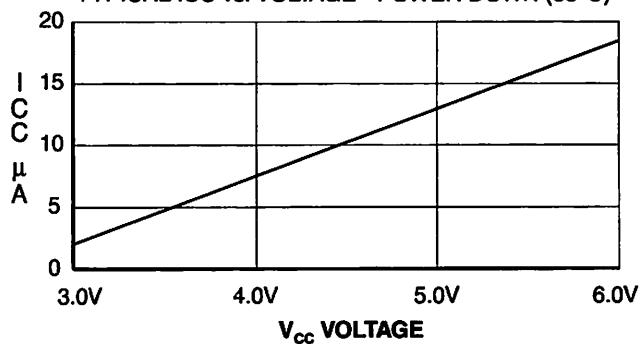
- Notes:
- For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OL}/V_{OL} level occurs.

AT89S8252

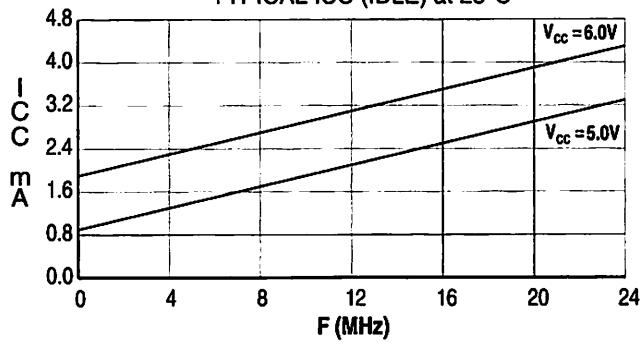
TYPICAL ICC (ACTIVE) at 25°C

**AT89S8252**

TYPICAL ICC vs. VOLTAGE - POWER DOWN (85°C)

**AT89S8252**

TYPICAL ICC (IDLE) at 25°C



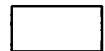
Notes:

1. XTAL1 tied to GND for Icc (power-down)
2. Lock 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	
		AT89S8252-24QC	44Q	
	4.0V to 6.0V	AT89S8252-24AI	44A	Industrial (-40°C to 85°C)
		AT89S8252-24JI	44J	
		AT89S8252-24PI	40P6	
		AT89S8252-24QI	44Q	
33	4.5V to 5.5V	AT89S8252-33AC	44A	Commercial (0°C to 70°C)
		AT89S8252-33JC	44J	
		AT89S8252-33PC	40P6	
		AT89S8252-33QC	44Q	



= Preliminary Information

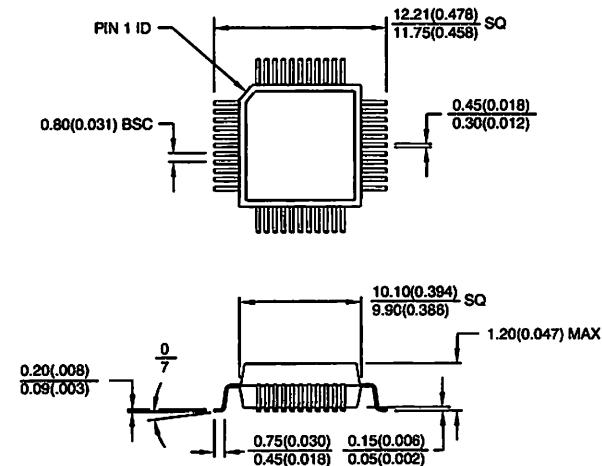
Package Type

4A	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
4J	44-lead, Plastic J-leaded Chip Carrier (PLCC)
0P6	40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
4Q	44-lead, Plastic Gull Wing Quad Flatpack (PQFP)

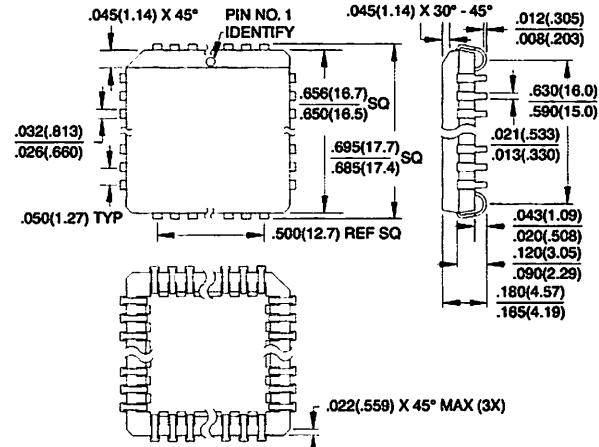
AT89S8252

Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flatpack (TQFP)
 Dimensions in Millimeters and (Inches)*
 JEDEC STANDARD MS-026 ACB

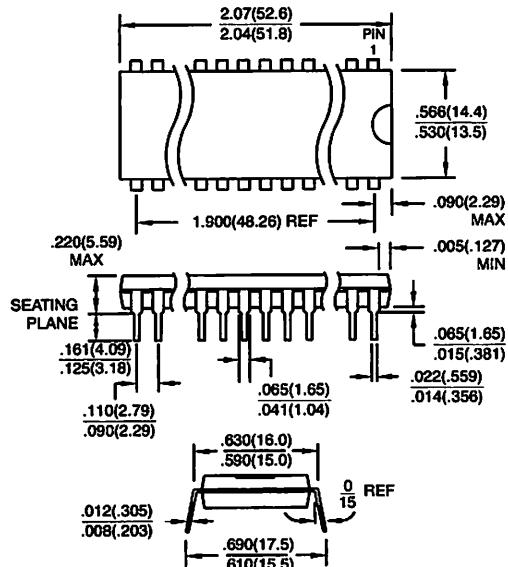


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
 Dimensions in Inches and (Millimeters)
 JEDEC STANDARD MS-018 AC

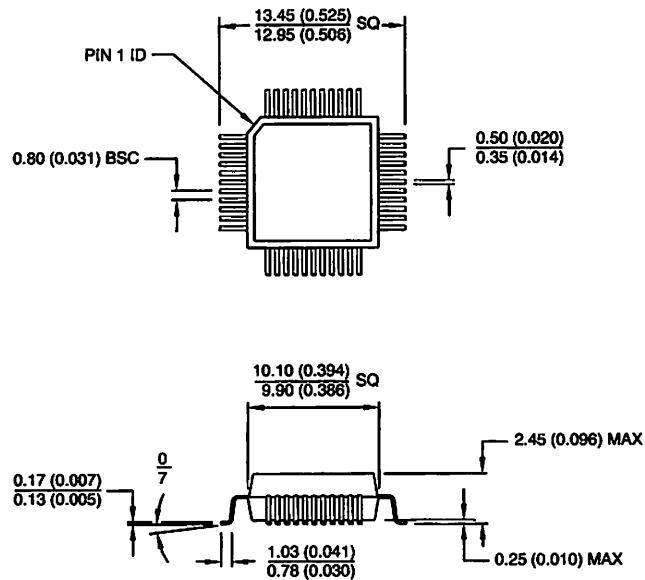


Controlling dimension: millimeters

40P6, 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
 Dimensions in Inches and (Millimeters)



44Q, 44-lead, Plastic Quad Flat Package (PQFP)
 Dimensions in Millimeters and (Inches)*
 JEDEC STANDARD MS-022 AB



Controlling dimension: millimeters



Atmel Headquarters

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

Europe

Atmel U.K., Ltd.
Coliseum Business Centre
Riverside Way
Camberley, Surrey GU15 3YL
England
TEL (44) 1276-686-677
FAX (44) 1276-686-697

Asia

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

Japan

Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

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

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

Fax-on-Demand

North America:
1-(800) 292-8635

International:
1-(408) 441-0732

e-mail
literature@atmel.com

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

BBS
1-(408) 436-4309

Atmel Corporation 2000.

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

Works bearing ® and/or ™ are registered trademarks and trademarks of Atmel Corporation.

Names and product names in this document may be trademarks of others.



Printed on recycled paper.

0401E-02/00/xM



National Semiconductor

October 1999

ADC0808/ADC0809 8-Bit μP Compatible A/D Converters with 8-Channel Multiplexer

ADC0808/ADC0809

8-Bit μP 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® 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.)

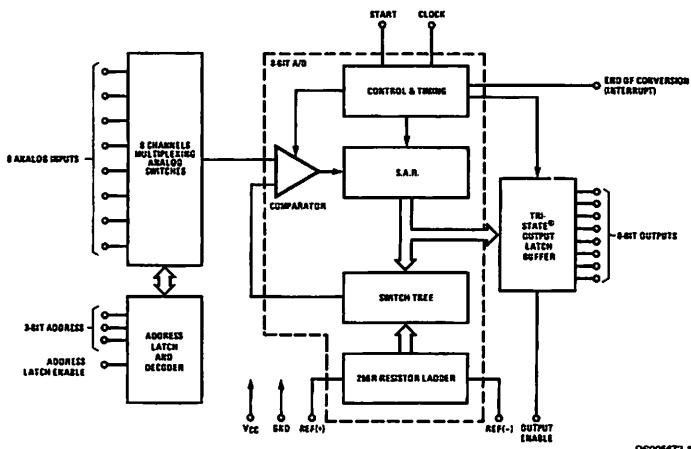
Features

- Easy interface to all microprocessors
- Operates ratiometrically or with 5 V_{DC} 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 MM74C949
- ADC0809 equivalent to MM74C949-1

Key Specifications

■ Resolution	8 Bits
■ Total Unadjusted Error	±½ LSB and ±1 LSB
■ Single Supply	5 V _{DC}
■ Low Power	15 mW
■ Conversion Time	100 µs

Block Diagram



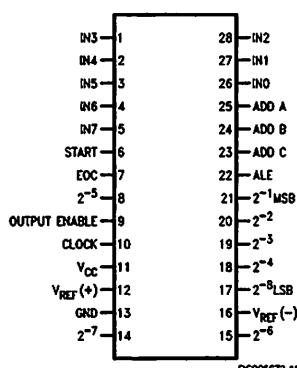
See Ordering
Information

DS005672-1

TRI-STATE® is a registered trademark of National Semiconductor Corp.

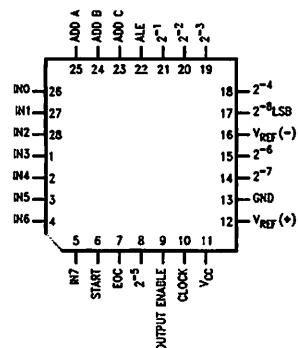
Connection Diagrams

Dual-In-Line Package



Order Number 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 $+85^{\circ}\text{C}$			-55°C to $+125^{\circ}\text{C}$
Error	$\pm\frac{1}{2}$ 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 (START, OE, CLOCK, ALE, ADD A, ADD B, ADD C)	-0.3V to +15V
Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A=25^\circ C$	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)	$T_{MIN} \leq T_A \leq T_{MAX}$
ADC0808CCN, ADC0809CCN	-40°C $\leq T_A \leq +85^\circ C$
ADC0808CCV, ADC0809CCV	-40°C $\leq T_A \leq +85^\circ C$

Range of V_{CC} (Note 1) 4.5 V_{DC} to 6.0 V_{DC}

Electrical Characteristics

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

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	ADC0808					
	Total Unadjusted Error (Note 5)	25°C T_{MIN} to T_{MAX}			$\pm\frac{1}{2}$ $\pm\frac{3}{4}$	LSB LSB
	ADC0809					
	Total Unadjusted Error (Note 5)	0°C to 70°C T_{MIN} to T_{MAX}			± 1 $\pm 1\frac{1}{4}$	LSB 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
$V_{REF(+)} + V_{REF(-)}$ 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 $f_c=640$ kHz, (Note 6)		-2	± 0.5	2	μA

Electrical Characteristics

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

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

Electrical Characteristics (Continued)

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

Symbol	Parameter	Conditions	Min	Typ	Max	Units
DATA OUTPUTS AND EOC (INTERRUPT)						
$V_{OUT(1)}$	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	$I_O = 1.6 \text{ mA}$			0.45	V
$V_{OUT(0)}$	Logical "0" Output Voltage EOC	$I_O = 1.2 \text{ mA}$			0.45	V
I_{OUT}	TRI-STATE Output Current	$V_O = 5V$ $V_O = 0$	-3		3	μA μA

Electrical Characteristics

Timing Specifications $V_{CC} = V_{REF(+)} = 5V$, $V_{REF(-)} = GND$, $t_s = t_l = 20 \text{ ns}$ and $T_A = 25^\circ C$ 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	μs
t_{H1}, t_{H0}	OE Control to Q Logic State	$C_L = 50 \text{ pF}$, $R_L = 10k$ (Figure 8)		125	250	ns
t_{H1}, t_{H0}	OE Control to Hi-Z	$C_L = 10 \text{ pF}$, $R_L = 10k$ (Figure 8)		125	250	ns
t_c	Conversion Time	$f_c = 640 \text{ kHz}$, (Figure 5) (Note 7)	90	100	116	μs
f_c	Clock Frequency		10	640	1280	kHz
t_{EOC}	EOC Delay Time	(Figure 5)	0		8±2 μs	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 7 V_{DC} .

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 0 V_{DC} to 5 V_{DC} input voltage range will therefore require a minimum supply voltage of 4.900 V_{DC} 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 13.

Note 6: Comparator input current is a bias current into or out of the chopper stabilized comparator. The 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 Ω 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 $\pm \frac{1}{2}$ 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)

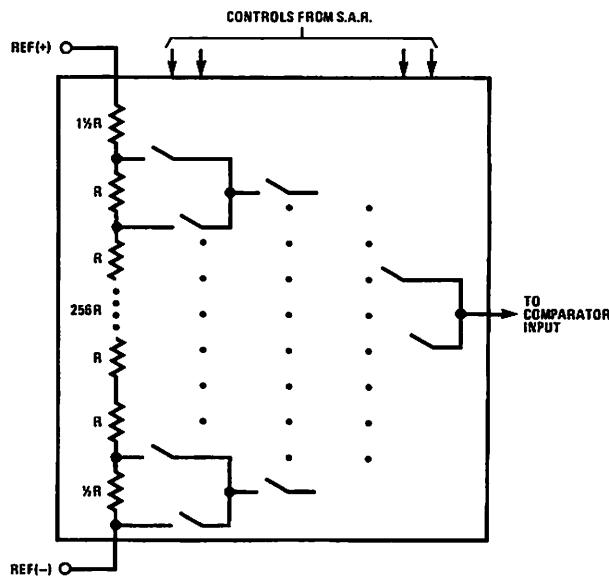


FIGURE 1. Resistor Ladder and Switch Tree

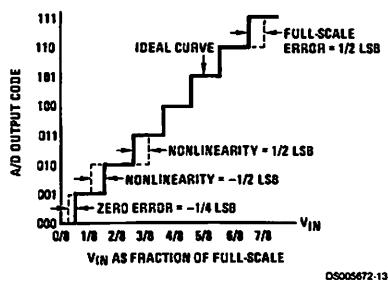


FIGURE 2. 3-Bit A/D Transfer Curve

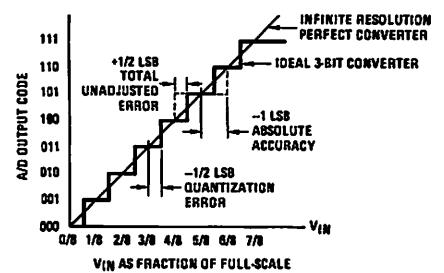


FIGURE 3. 3-Bit A/D Absolute Accuracy Curve

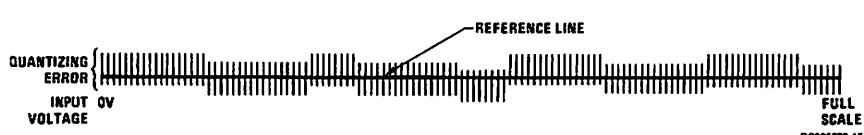
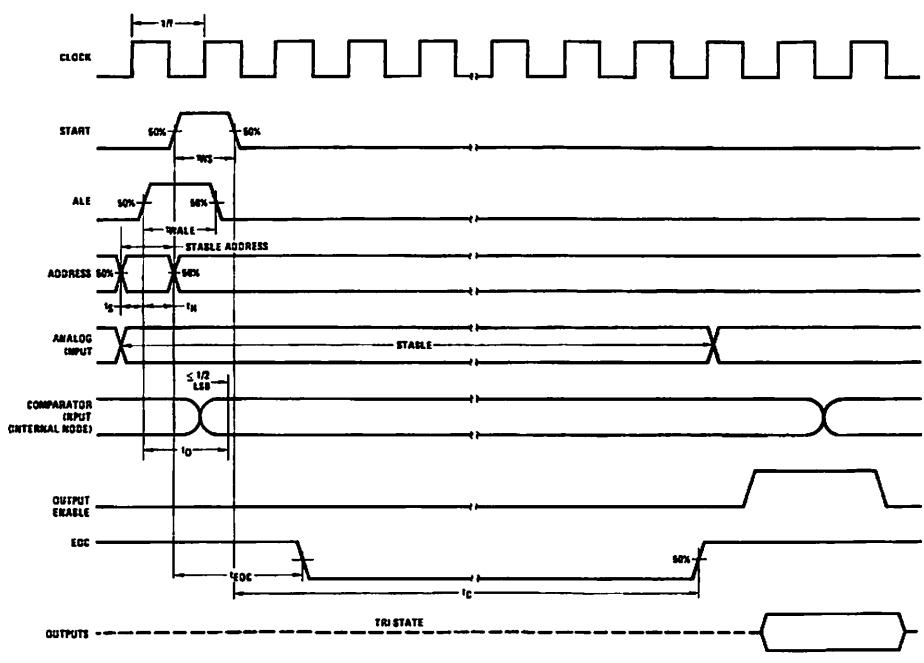


FIGURE 4. Typical Error Curve

Timing Diagram**FIGURE 5.**

Typical Performance Characteristics

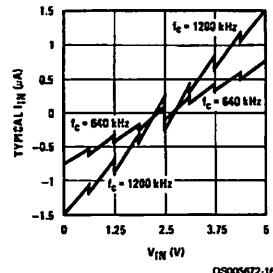


FIGURE 6. Comparator I_{IN} vs V_{IN}
($V_{CC} = V_{REF} = 5V$)

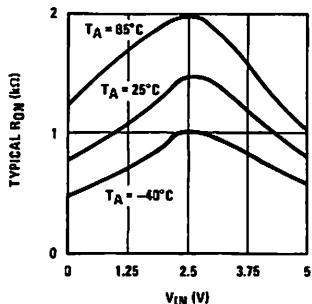


FIGURE 7. Multiplexer R_{ON} vs V_{IN}
($V_{CC} = V_{REF} = 5V$)

TRI-STATE Test Circuits and Timing Diagrams

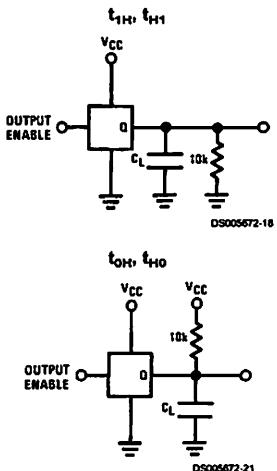


FIGURE 8.

Applications Information

OPERATION

1.0 RATIO METRIC CONVERSION

The ADC0808, ADC0809 is designed as a complete Data Acquisition System (DAS) for ratiometric conversion systems. In ratiometric 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

$$\frac{V_{IN}}{V_{fs} - V_z} = \frac{D_x}{D_{MAX} - D_{MIN}} \quad (1)$$

V_{IN} =Input voltage into the ADC0808

V_{fs} =Full-scale voltage

V_z =Zero voltage

D_x =Data point being measured

D_{MAX} =Maximum data limit

D_{MIN} =Minimum data limit

A good example of a ratiometric 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).

Ratiometric 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 times 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.

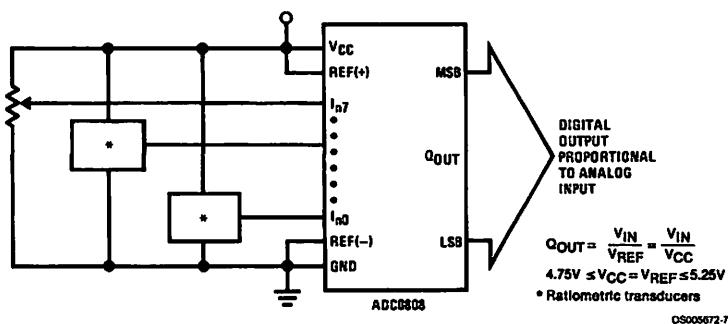
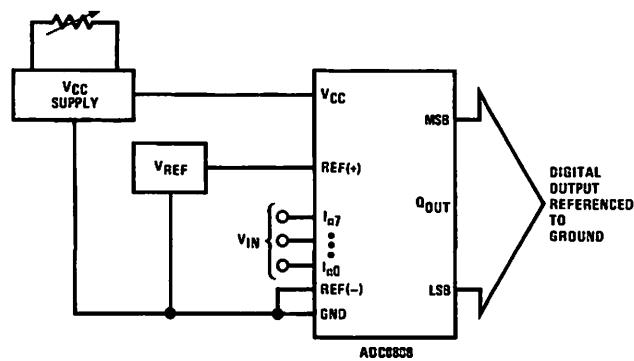


FIGURE 9. Ratiometric Conversion System

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 μ F output capacitor.

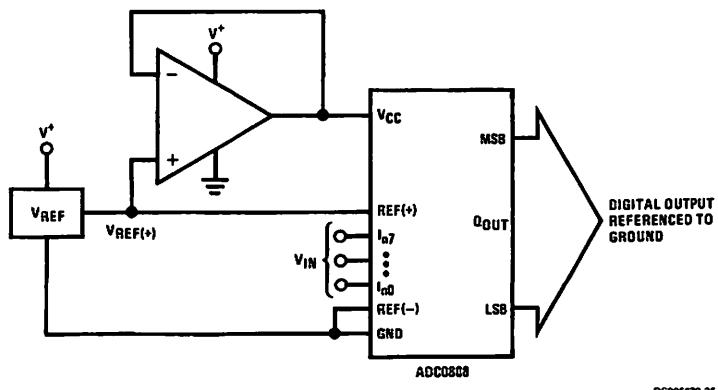
The top and bottom ladder voltages cannot exceed V_{CC} and ground, respectively, but they can be symmetrically less than V_{CC} and greater than ground. The center of the ladder voltage should always be near the center of the supply. The sensitivity of the converter can be increased, (i.e., size of 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)

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

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

FIGURE 10. Ground Referenced Conversion System Using Trimmed Supply



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

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

FIGURE 11. Ground Referenced Conversion System with Reference Generating V_{CC} Supply

Applications Information (Continued)

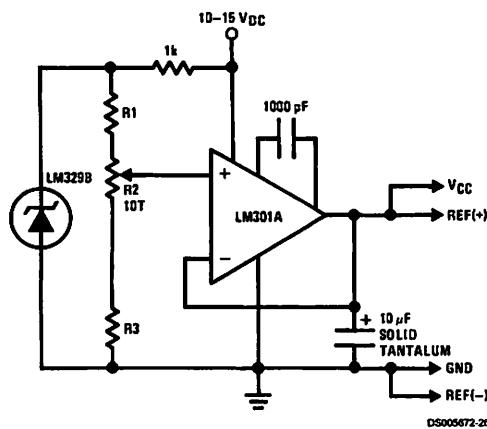
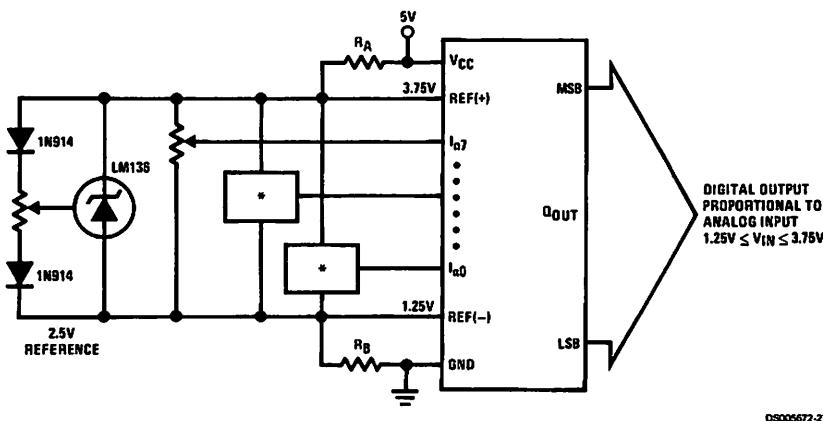


FIGURE 12. Typical Reference and Supply Circuit



$$R_A = R_B$$

*Ratiometric transducers

FIGURE 13. Symmetrically Centered Reference

3.0 CONVERTER EQUATIONS

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

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

The center of an output code N is given by:

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

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)

$V_{REF(+)} + 512$

Applications Information (Continued)

4.0 ANALOG COMPARATOR INPUTS

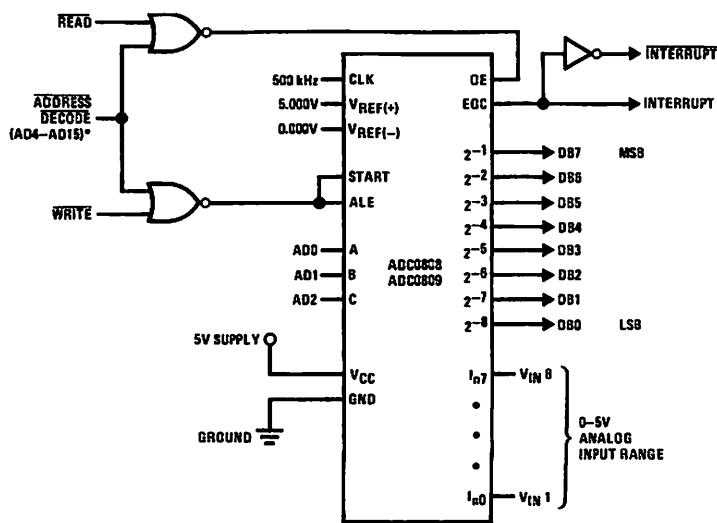
The dynamic comparator input current is caused by the periodic switching of on-chip stray capacitances. These are connected alternately to the output of the resistor ladder/switch tree network and to the comparator input as part of the operation of the chopper stabilized comparator.

The average value of the comparator input current varies directly with clock frequency and with V_{IN} 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



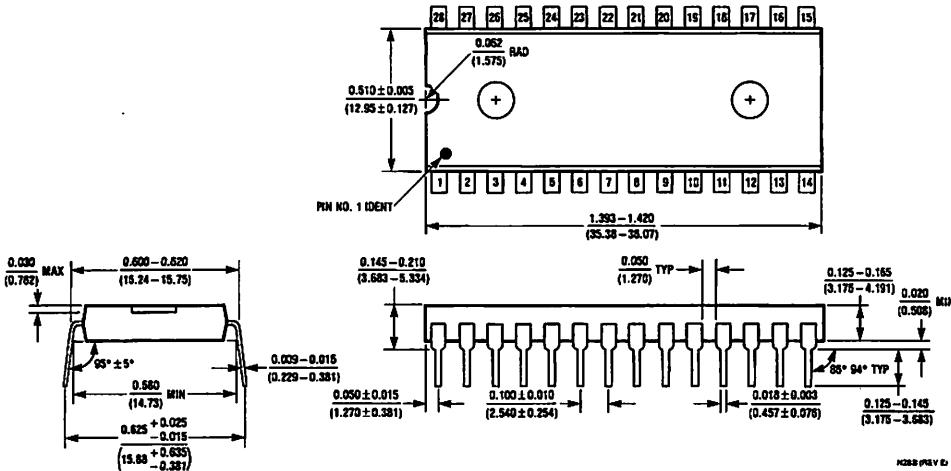
DS005072-10

*Address latches needed for 8085 and SC/MP interfacing the ADC0808 to a microprocessor

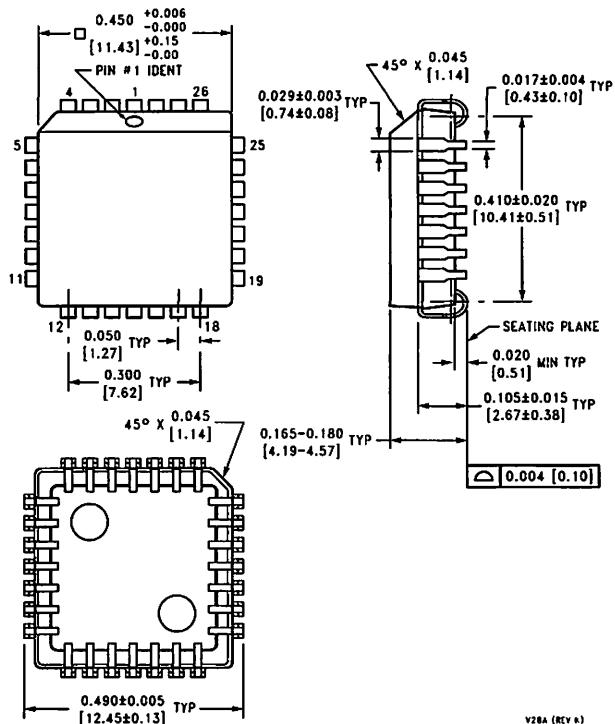
TABLE 2. Microprocessor Interface Table

PROCESSOR	READ	WRITE	INTERRUPT (COMMENT)
8080	MEMR	MEMW	INTR (Thru RST Circuit)
8085	RD	WR	INTR (Thru RST Circuit)
Z-80	RD	WR	INT (Thru RST Circuit, Mode 0)
SC/MP	NRDS	NWDS	SA (Thru Sense A)
6800	VMA- ϕ 2-R/W	VMA- ϕ -RW	IRQA or IRQB (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: 1-800-272-9959
Fax: 1-800-737-7018
Email: support@nsc.com
www.national.com

National Semiconductor
Europe
Fax: +49 (0) 1 80-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 1 80-530 85 85
English Tel: +49 (0) 1 80-532 78 32
Français Tel: +49 (0) 1 80-532 93 58
Italiano Tel: +49 (0) 1 80-534 16 80

National Semiconductor
Asia Pacific Customer
Response Group
Tel: 65-2544466
Fax: 65-2504466
Email: sea.support@nsc.com

National Semiconductor
Japan Ltd.
Tel: 81-3-5639-7560
Fax: 81-3-5639-7507

2SC828, 2SC828A

シリコン NPN エピタキシャルプレーナ形 / Si NPN Epitaxial Planar

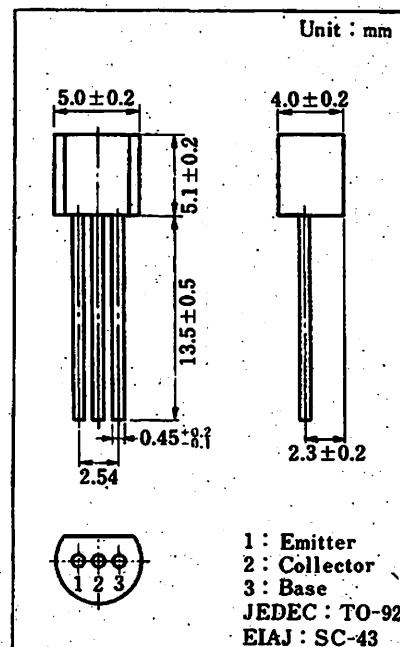
小信号増幅用 / Small Signal Amplifier

■ 特徴 / Feature

- 直流電流増幅率 h_{FE} が高い。/ High h_{FE}

■ 絶対最大定格 / Absolute Maximum Ratings ($T_a=25^\circ\text{C}$)

Item	Symbol	Value	Unit
コレクタ・ベース電圧 2SC828	V_{CEO}	30	V
2SC828A	V_{CEO}	45	V
コレクタ・エミッタ電圧 2SC828	V_{CEO}	25	V
2SC828A	V_{CEO}	45	V
エミッタ・ベース電圧	V_{EBO}	7	V
せん頭コレクタ電流	I_{CP}	100	mA
コレクタ電流	I_C	50	mA
コレクタ損失	P_c	400	mW
接合部温度	T_j	150	°C
保存温度	T_{sig}	-55 ~ +150	°C



■ 電気的特性 / Electrical Characteristics ($T_a=25^\circ\text{C}$)

Item	Symbol	Condition	min.	typ.	max.	Unit
コレクタ・ベース電圧 2SC828	V_{CEO}	$I_C = 10 \mu\text{A}, I_E = 0$	30			V
2SC828A	V_{CEO}	$I_C = 10 \mu\text{A}, I_E = 0$	45			V
コレクタ・エミッタ電圧 2SC828	V_{CEO}	$I_C = 2 \text{ mA}, I_E = 0$	25			V
2SC828A	V_{CEO}	$I_C = 2 \text{ mA}, I_E = 0$	45			V
エミッタ・ベース電圧	V_{EBO}	$I_E = 10 \mu\text{A}, I_C = 0$	7			V
直流電流増幅率	h_{FE} *	$V_{CE} = 5 \text{ V}, I_C = 2 \text{ mA}$	130		520	
トランジション周波数	f_T	$V_{CB} = 10 \text{ V}, I_E = -2 \text{ mA}$		220		MHz
ベース・エミッタ電圧	V_{BE}	$V_{CB} = 5 \text{ V}, I_C = 10 \text{ mA}$			0.8	V
コレクタ・エミッタ飽和電圧	$V_{CES(sat)}$	$I_C = 50 \text{ mA}, I_E = 5 \text{ mA}$		0.14		V
雑音指数	NF	$V_{CE} = 5 \text{ V}, I_C = 0.2 \text{ mA}, R_g = 2 \text{ k}\Omega, f = 1 \text{ kHz}$		6		dB

* h_{FE} ランク分類 / h_{FE} Classifications

Class	Q	R	S
h_{FE}	130 ~ 260	180 ~ 360	260 ~ 520