

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



SKRIPSI

**PERANCANGAN DAN PEMBUATAN ALAT PENGENDALI KEMUDI
ARAH PADA EKOR PESAWAT TERBANG DENGAN
MENGUNAKAN MIKROKONTROLLER AT89S51**

Disusun Oleh :

**SILVESTER YOSEPH K TADUBUN
NIM: 98 17 059**

MARET 2005

LEMBAR PERSETUJUAN



**PERANCANGAN DAN PEMBUATAN ALAT PENGENDALI KEMUDI
ARAH PADA EKOR PESAWAT TERBANG DENGAN
MENGUNAKAN MIKROKONTROLLER AT89S51**

SKRIPSI

*Disusun Untuk Melengkapi Dan Memenuhi Syarat
Guna Mencapai Gelar Sarjana Teknik*

**Disusun Oleh:
SILVESTER YOSEPH KAROLUS TADUBUN
NIM : 98.17.059**

Diperiksa dan Disetujui

Dosen Pembimbing I

Ir. Usman Djuanda, MM
NIP.P. 1018700143

Dosen Pembimbing II

M. Ibrahim Ashari, ST
NIP.P. 1030100358



**Mengetahui,
Ketua Jurusan T. Elektro S-1**

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

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



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

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : Silvester Yoseph Karolus Tadubun
N I M : 98.17.059
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : Perancangan Dan Pembuatan Alat Pengendali kemudi
Arah Pada Ekor Pesawat Terbang Dengan
Menggunakan Mikrokontroller AT89S51.

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

Hari : Selasa
Tanggal : 29 Maret 2005
Dengan Nilai : 81 (A) *Sm*



**(Ir. Mochtar Asroni, MSME)
Ketua**

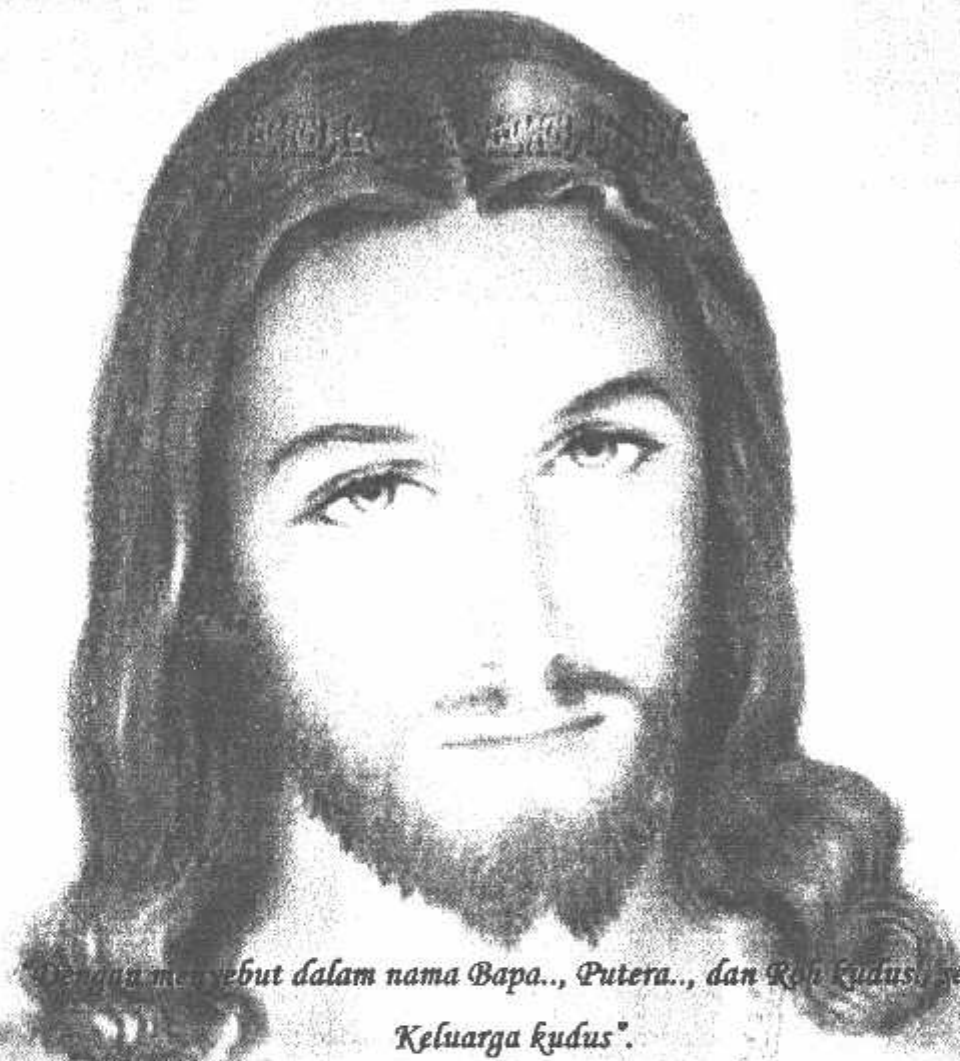
Panitia Ujian Skripsi

**(Ir. F. Yudi Limpraptono, MT)
Sekretaris**

Anggota Penguji

**(Ir. Sidik Noerjahjono, MT)
Penguji I**

**(Joseph Dedy Irawan, ST, MT)
Penguji II**



"Dengan menyebut dalam nama Bapa., Putera., dan Roh Kudus, serta Keluarga Kudus".

Ya Allahku., Ya TuhanKu., Ya Yesus Putra Bapa... Tuntunlah HambaMu yang Hina dengan FirmanMu ini :

*"Bukan Kamu yang memilih Aku, tetapi Akulah yang memilih Kamu.
Yoh 15:16."*

"Marilah kepadaKu, semua yang letih sesu dan berbeban berat, Aku akan Memberi Kelegahan MepadaMu. Mat 11:28."

"Sekalipun Aku Berjalan dalam lembah kekelaman, Aku tidak takut bahaya sebab Engkau besertaKu. Mzm 23:4"

<http://www.jeevandham.org/>

Gelar :

*Nyanda dan Ibunda, Via. T dan Vika. T, Kakakku berdua Hence. T dan Bill. T
sekeluarga, adikku berdua Peter. T dan Yully. T* yang tersayang, terimakasih
atas semua kasih sayang, doa, perhatian, dan nasehat serta
pengorbanan dukungannya moril maupun spritual yang tak henti-tentiNya
tercurahkan kepada Ku. Begitu besar jasa yang engkau berikan sehingga tak
mungkin tuk bisa sepenuh membalasnya.

Ucapan Terimakasih Ku Kepada
Ucapan Terimakasih Ku Kepada

*Bapa Nus. T Sekeluarga, Bapa Yos. T Sekeluarga, dan Keluarga Besar
Tadubun dan Rumere* di Merauke-Kepi, Bial. Terimakasih atas nasehat,
pengorbanan, dukungannya moril maupun spritual yang begitu besar kepadaku
sehingga dapat menyelesaikan gelar ini

Bapak *Jhon G. gebze* selaku Bupati Merauke, Bapak *Jhon Rumlus* selaku Bupati
Mappi dan *Delsas-Keuskupan* Merauke serta *Yasanto* Merauke yang sudah
memberikan dukungannya moril kepadaku.

Buat *Keluarga Bapak Paimo (Bapak, Ibu, Mas Cahar Sekeluarga, Saipul dan
Porwati, Liva, Kiptia dan Ilah)* Di Blubokh, Makasih atas semua pengorbana dan
dukungan yang begitu besar.

Buat *Persimer Malang (Kees, Fajar, Kris, Tian, Philip, Frein, Juliian, Hendrik,
Mario, Steven, Nando, Petrus, Paskal, Roy, Jhon, iwan, Jefri, Findi)* terimakasih
atas kepercayaan kepadaku yang melatih dan capten tim, teruskanlah persatuan
dan kekompakan dalam tim serta lebih giat latihan agar bisa jadi yang terbaik .

Go my Valentine

Buat *Sporter Persimer (Pa Willy, Pa Romanus, Mas Robert dan Wiwik, Enjel, Gres, Yuni, Eti, Mila, Emei, Titin)* terimakasih atas Suara tiga dan konsumsinya serta kesetiaan yang selalu bersama Persimer.

Buat *Hendrik dan Diks Ipar*, terimakasih atas jasanya mauh meminjamkan motor yang sudah menemaniku selama mengerjakan skripsi sampai selesai.

Buat *PS. Black Star (PUCEN) Malang (Mario, Tito, Agus, Zulfansen, Tobi, Bakri, Simon, Boni, Bule, Harry, Yosis, Sam, Yudi, Julman, Agus, Ido, Samsul, Maks, Ogen, Boni, Gafur, Deni dan Official Netson, K'Lent, Jefri, Willy, Om'Dolfin)*. Terimakasih atas kepercayaan mengurus dan menjadi captan tim dalam liga kelumitu Desember 2004.

Buat *Are Kost Darma 3b (Dedi, Ali dan Ani, Toleh, Tenel, Kiky, Pedro, Daut, Rahman, Hengky, Achon, Alfret, Bari dan Ika, Mondir, Koni, Ota, Adipapa)* yang selalu enjoi. Ku ucapkan moturson atas semua kekompakannya. Dan selalu menjaga persahabatan walau nanjauh.

Buat *Elctro '89 (Elka 1,2)*. Makasih atas kekompakan dan dukungan selama ini dalam memperjuangkan sarjana S. Semoga persabatan kita selalu terjaga.

Tiada kam ya ag dapat kuncarkan. Selain hari kepadamu.

Tiada rasa tercurahkan. Selain sayang dan rindih.

Tiada yang dapat kabalas hanyalah doa.

Kau yang kucinta dan kusayang

Kan kujadikan kekuatan dalam perjalanan hidupku

Cedkawasih

S.V. Charlo's Tadubun

ABSTRAKSI

PERANCANGAN DAN PEMBUATAN ALAT PENGENDALI KEMUDI ARAH PADA EKOR PESAWAT TERBANG DENGAN MENGUNAKAN MIKROKONTROLLER AT98S51

(Silvester Yoseph K. Tadubun, Jurusan Teknik ElektroS-1/Elektronika)
(Dosen Pembimbing I : Ir. Usman Djuanda, MM)
(Dosen Pembimbing II : M. Ibrahim Ashari, ST)

Pada zaman sekarang manusia selalu menuntut kemudahan-kemudahan dalam bekerja dengan tanpa meninggalkan hasil yang maksimal, oleh sebab itu perlu dibuat alat yang sederhana dan memberi pilihan yang dapat membantu dan mempermudah kerja manusia misalkan saja alat kontrol kemudi arah pada pesawat terbang, kontrol kemudi arah difungsikan untuk pembelokan arah dan mendeteksi sudut pembelokan.

System ini menggunakan Mikrokontroller AT89S51 sebagai basisnya. Alat yang dibuat meliputi perencanaan perangkat keras dan perangkat lunak. Perencanaan perangkat keras meliputi : rangkaian sensor potensiometer, rangkaian ADC rangkaian driver motor DC, rangkaian keypad dan minimum system mikrokontroller AT89S51. Perencanaan perangkat lunak meliputi : *flowchart* cara kerja system dan software. Pada dasarnya prinsip kerjanya pengendali (kontrol) pada kemudi arah dimulai dari instruksi perintah, pengambilan data dari pergerakan Motor DC dan Sensor potensiometer untuk diolah dan seterusnya ditampilkan untuk L C D. Dimana ketika instruksi pembelokan arah diberikan maka driver motor DC akan berputar sehingga menggerakkan kemudi arah dan potensiometer, perubahan-perubahan nilai resistansi inilah oleh transduser akan diubah menjadi besaran listrik analog yang kemudian dari besaran analog oleh enter ADC diubah menjadi pulsa digital yang kemudian diumpankan ke MCU AT89S51, dari MCU ini data yang berupa pulsa diolah dan ditampilkan kedalam LCD.

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa yang telah memberikan Rahmat kepada kita semua dan khususnya penulis sehingga dapat menyelesaikan laporan Skripsi ini dengan judul :

**PERANCANGAN DAN PEMBUATAN ALAT PENGENDALI KEMUDI
ARAH PADA EKOR PESAWAT TERBANG DENGAN MENGGUNAKAN
MIKROKONTROLLER AT98S51**

Pembuatan skripsi ini guna memenuhi syarat akhir kelulusan pendidikan jenjang strata-1 di Program Studi Elektronika Jurusan Teknik Elektro Institut Teknologi Nasional Malang.

Dengan segala kerendahan hati atas keberhasilan penyelesaian laporan Skripsi ini, Penyusun mengucapkan terima kasih kepada :

1. Kedua Orang Tua-ku dan Kakak-ku serta adik-ku atas segala doa dan dukungannya.
2. Bapak Dr. Ir. Abraham Lomi, MSFE, selaku Rektor Institut Teknologi Nasional Malang
3. Bapak Ir. Mochtar Asroni, MSME, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Malang.
4. Bapak Ir. F. Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro Institut Teknologi Malang.

5. Ibu Ir. Mimien Mustikawati, selaku Sekretaris Jurusan Teknik Elektro Institut Teknologi Malang.
6. Bapak Ir. Usman Djuanda, MM, selaku Dosen Pembimbing I yang telah memberikan bimbingan, saran dan pemikiran dalam menyelesaikan laporan skripsi ini.
7. Bapak M. Ibrahim Ashari, ST, selaku Dosen Pembimbing II yang telah memberikan bimbingan, saran dan pemikiran dalam menyelesaikan laporan skripsi ini.
8. Bapak Ir. Sidik Noerjahjono, MT, selaku Dosen Penguji ujian Skripsi yang memberikan kritik dan saran.
9. Bapak Joseph Dedy Irawan, ST, MT, selaku Dosen Penguji ujian Skripsi yang memberikan kritik dan saran.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang bersifat membangun sangat diharapkan dari pembaca khususnya mahasiswa Teknik Jurusan Elektro ITN Malang, agar di masa datang akan lebih baik lagi.

Harapan dari penulis semoga skripsi ini dapat bermanfaat.

Malang, Maret 2005

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAK	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xii
DAFTAR TABEL	xv
BAB I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan	2
1.4. Batasan Masalah	3
1.5. Metodologi	4
1.6. Sistematika Pembahasan	4

BAB II. LANDASAN TEORI	6
2.1. Pesawat Terbang	6
2.1.1. Kemudi Arah	7
2.1.2. Sistem Kontrol Kemudi Arah (<i>Rudder</i>)	8
2.2. Keypad 4 x 4	9
2.3. Transistor	10
2.3.1. Tiga Daerah Transistor	12
2.3.2. Pembiasan Transistor Bipolar	12
2.3.2. Transistor Sebagai Saklar	16
2.4. Relay	18
2.5. Motor DC	22
2.5.1. Prinsip Kerja Motor DC	22
2.5.2. Pengendali Arah Putar Motor DC	25
2.6. Sensor	26
2.6.1. Sensor Sudut	27
2.7. <i>Analog to Digital Converter (ADC)</i>	28
2.8. Mikrokontroler AT89S51	32
2.8.1. Pendahuluan	32
2.8.2. Perangkat Keras Mikrokontroler	33
2.8.3. Konfigurasi Pena-Pena Mikrokontroler AT89S51	34
2.8.4. Organisasi Memory	40

2.8.4.1. Pemisahan Memory Program dan Data	40
2.8.4.2. Memory Program	41
2.8.4.3. Memory Data	42
2.8.5. Pewaktu	45
2.8.6. Sistem Interupsi	46
2.8.7. SFR (<i>Special Function Register</i>)	48
2.8.8. Time / Counter	51
2.8.9. Port Serial	53
2.8.10. Program Status Word	53
2.8.11. Power Control Word	55
2.8.12. Metode Pengalamatan	56
2.9. <i>Liquid Crystal Display</i> (LCD)	58
2.9.1. Beberapa Bagian-bagian Pendukung Modul M1632 Dalam Pengoperasian	59
2.9.2. Pena-Pena LCD M1632	60
2.9.3. Struktur Memory LCD M1632	61
2.9.4. Register-Register LCD M1632	62
2.9.5. Perintah-Perintah LCD M1632	64
BAB III. PERENCANAAN ALAT	68
3.1. Pendahuluan	68

3.2.	Alat dan Bahan	68
3.3.	Perancangan Diagram Blok	69
3.4.	Prinsip Kerja Rangkaian	70
3.5.	Rangkaian Keypad 4 x 4	71
3.6.	Rangkaian <i>Driver</i> Motor DC	73
3.7.	Rangkaian Sensor Sudut	74
3.8.	Rangkaian ADC (<i>Analog to Digital Converter</i>)	76
3.9.	Perencanaan Minimum Sistem AT89S51	79
3.9.1.	Rangkaian Clock Minimum System	81
3.9.2.	Rangkaian Reset	83
3.10.	<i>Liquid Crystal Display</i> (LCD)	84
3.11.	Perencanaan Konstruksi Kemudi Arah Pada Pesawat Terbang	86
3.12.	Perancangan Perangkat Lunak	87
BAB IV.	PENGUJIAN DAN ANALISA ALAT	91
4.1.	Pendahuluan	91
4.2.	Pengujian Perangkat Keras	91
4.2.1.	Pengujian Rangkaian Sensor Sudut	92
4.2.2.	Pengujian Rangkaian ADC 0804	98
4.2.3.	Pengujian Rangkaian <i>Driver</i> Motor DC	101

4.2.4. Pengujian Rangkaian Keypad 4 x 4	102
4.2.5. Pengujian Rangkaian Alat Keseluruhan	104
4.3. Pengujian Perangkat Lunak	105
4.3.1. Pengujian Program Assembly	102
BAB V. PENUTUP	106
5.1. Kesimpulan	107
5.2. Saran	104

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar	Halaman
2.1. Pesawat Terbang CASA-212	7
2.2. Kemudi Arah (Rudder)	8
2.3. Konfigurasi Tombol Keypad	10
2.4. Simbol Transistor Bipolar	11
2.5. Garis Beban Transistor	11
2.6. Rangkaian Bias Forward – Forward	13
2.7. Rangkaian Bias Referse – Referse	13
2.8. Rangkaian Bias Forward – Referse	14
2.9. Transistor Dalam Keadaan Saturasi	17
2.10. Transistor Dalam Keadaan Cut Off	18
2.11. Cara Kerja Relay	19
2.12. Jenis Relay SPST	20
2.13. Jenis Relay SPDT	20
2.14. Jenis Relay DPST	21
2.15. Jenis Relay DPDT	21
2.16. Kaidah Tangan Kanan	23
2.17. Cara Kerja Motor DC	24
2.18. Sebuah Motor DC	25

Gambar	Halaman
2.19. Pengendali Arah Putar Motor DC	26
2.20. Konfigurasi Pena-Pena ADC 0804	29
2.21. Blok Diagram Mikrokontroller	34
2.22. Konfigurasi Pena-Pena AT89S51	36
2.23. Alamat RAM	40
2.24. Struktur Memory	41
2.25. Memory Program	41
2.26. Organisasi RAM Internal	44
2.27. Menggunakan Oscilator Internal	45
2.28. Menggunakan Sumber Clock Eksternal	46
2.29. Modul M1632	59
2.30. Susunan Data Status HD 44780	64
2.31. Hubungan antar Mikrokontroller Dengan LCD M1632	67
3.1. Blok Diagram Rangkaian	70
3.2. Rangkaian Keypad 4 x 4	72
3.3. Rangkaian <i>Driver</i> Motor DC	73
3.4. Rangkaian Sensor Sudut	75
3.5. Rangkaian ADC (<i>Analog to Digital Converter</i>) 0804	79
3.6. Perencanaan Rangkaian Mikrokontroller AT89S51	81
3.7. Rangkaian Clock	82



BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi dibidang elektronika telah membawa perubahan dan kemajuan yang begitu cepat dan canggih. Hal ini dibuktikan dengan adanya penemuan-penemuan baru disegala bidang teknologi. Pesatnya perkembangan teknologi ini menghasilkan berbagai penemuan dan terobosan dibidang mikroprosesor dan mikrokontroller yang semakin banyak membantu manusia.

Pada zaman sekarang manusia selalu menuntut kemudahan-kemudahan dalam bekerja dengan tanpa meninggalkan hasil yang maksimal, oleh sebab itu perlu dibuat alat yang sederhana dan memberi pilihan yang dapat membantu dan mempermudah kerja manusia misalkan saja alat kontrol dari kemudi arah (rudder) pada pesawat terbang, alat kontrol dari kemudi arah difungsikan untuk pentrolan pembelokan secara program dan dapat mengetahui atau melihat arah sudut secara horizontal yakni arah sudut kanan dan arah sudut kiri, atau untuk mengetahui arah horizontal bidang dari pesawat terbang. Didalam dunia penerbangan sehari-hari alat kontrol dari kemudi arah pada pesawat terbang biasa hanya mampu mengukur arah sudut horizontal kanan dan sudut horizontal kiri secara manual,

Adapun kontroller yang digunakan sebagai pengolah otomatisasi adalah Mikrokontroller, karena mikrokontroller dapat dengan mudah

digunakan untuk memperoleh data dan efisiensi peralatan dibidang menggunakan PC (Personal Computer).

1.2. Rumusan Masalah

Dari kenyataan tersebut maka dicari pemecahan masalah agar bagaimana kita mampu mengontrol dan mengetahui arah sudut bidang secara horisontal yang akurat dan mudah dalam membacanya, maka timbullah suatu ide dan pemikiran sehingga dibuatlah pemecahan masalah tersebut, yaitu pembuatan suatu alat dimana alat tersebut mampu untuk mengontrol dan membaca serta mengetahui arah sudut bidang horisontal yang akurat dan mudah dalam pengontrolan.

Bagaimana memanfaatkan mikrokontroller sebagai suatu alat untuk mengatur kendalih pada kemudi arah dan mendeteksi sudut pembelokan dari pesawat terbang.

Karena alasan tersebut dalam Tugas Akhir ini dipilih judul :

**PERANCANGAN DAN PEMBUATAN ALAT PENGENDALI
KEMUDI ARAH PADA EKOR PESAWAT TERBANG DENGAN
MENGUNAKAN MIKROKONTROLLER AT89S51**

1.3. Tujuan

Tujuan dari penyusunan Tugas Akhir ini adalah, Untuk merencanakan dan membuat alat yang dapat mengontrol dan mengetahui sudut dalam derajat dari kemudi arah pesawat terbang dengan tampilan LCD

berbasiskan Mikrokontroller AT89S51, dan mempermudah dalam pengoperasian system pada pesawat terbang.

1.4. Batasan Masalah

Dalam perencanaan dan pembuatan alat kontrol kemudi arah ini perlu diberikan suatu ruang lingkup pembahasan.

Ruang lingkup pembahasan penulisan Tugas Akhir ini dibahas masalah-masalah sebagai berikut :

1. Sensor yang dipergunakan adalah potensiometer.
2. Tidak digunakan untuk mengukur sudut yang melebihi dari kemampuan alat ukur yakni 50 derajat (ring ukur sudut dari -25° sampai $+25^{\circ}$)
3. Membuat program dengan bahasa assembly MCS 51 untuk mengolah data input dan menghasilkan data output yang disimpan kedalam IC Mikrokontroler AT89S51 yang digunakan untuk memberikan masukan ke LCD
4. Tidak membahas power supply, karena dianggap sudah stabil dimana kita tinggal menyesuaikan tegangannya.
5. Tidak membahas pengaruh kecepatan pesawat, grafitasi bumi dan pengaruh angin terhadap pesawat perbang
6. Pembatasan Pembahasan pesawat terbang hanya khusus pada pengendali (kontrol) dari kemudi arah.

1.5. Metodologi

Guna merealisasikan Tugas Akhir ini, penulis menggunakan beberapa metode, yaitu :

1. Library Research

Yaitu memperoleh data dengan cara membaca dan mempelajari buku literature yang ada hubungannya dengan penyusunan Tugas Akhir ini.

2. Field Research

Yaitu memperoleh data dengan jalan membuat analisa secara aktual dan dengan menarik kesimpulan serta pengujian data yang ada.

3. Pengolahan Data

Yaitu mengolah data dengan jalan membuat analisa secara aktual dan dengan menarik kesimpulan serta pengujian data yang ada.

1.6. Sistematika Pembahasan

Tugas akhir ini disusun berdasarkan beberapa teori penunjang serta bagian-bagian dari perencanaan piranti system yang dibagi menjadi 5 (lima) bab dan beberapa sub bab. Inti pembahasan penulisan dapat diuraikan sebagai berikut :

BAB I : PENDAHULUAN

Merupakan pendahuluan yang berisikan tentang latar belakang, rumusan masalah, tujuan, batasan masalah dan metodologi penulisan serta sistematika pembahasan.

BAB II : LANDASAN TEORI

Membahas tentang teori dasar yang berisikan teori-teori yang didapat selama mengikuti perkuliahan dan memiliki penunjang sebagai dasar dalam perencanaan dan pembuatan alat yang diajukan pada Tugas Akhir ini.

BAB III: PERENCANAAN ALAT

Menerangkan dan membahas masalah perencanaan dan pembuatan alat.

BAB IV: PENGUJIAN ALAT

Membahas tentang pengujian alat secara keseluruhan.

BAB V : PENUTUP

Berisikan kesimpulan dan saran yang didapat selama perencanaan dan pembuatan alat.

BAB II

TEORI DASAR

Untuk dapat memahami alat yang akan dirancang maka dalam bab II ini dijelaskan mengenai teori dasar yang berkaitan dengan sistem, diantaranya meliputi: Pesawat Terbang, kemudi arah, Keypad, Driver Motor DC, Sensor potensiometer, Analog To Digital Converter (ADC), Mikrokontroler AT89S51, dan Liquid Crystal Display (LCD).

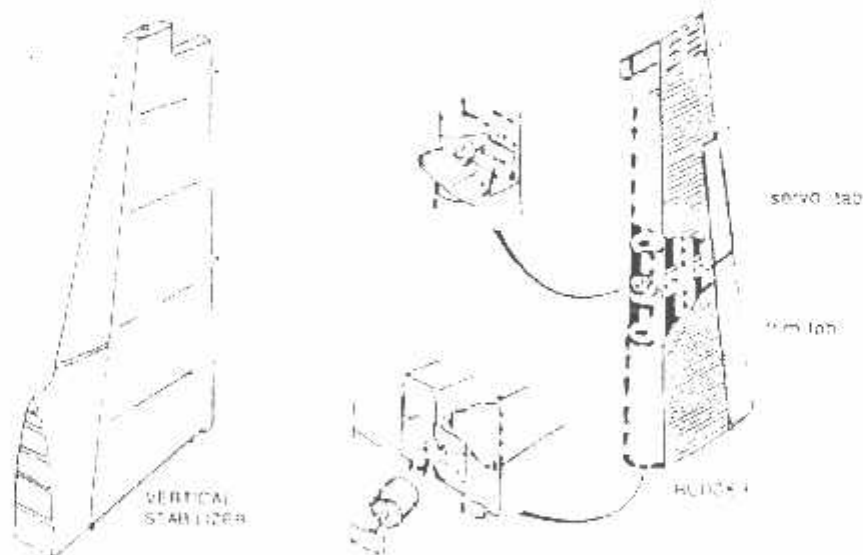
2.1. Pesawat Terbang

Perkembangan teknologi di bidang penerbangan saat ini sangatlah cepat dan canggih baik dalam desain dan pembuatan perangkat keras (hardware) maupun perangkat lunak (software) yang digunakan untuk pengendalian dalam sistem penerbangan.

Secara garis besar, setiap pesawat terbang memiliki beberapa sistem dan karakteristik yang sama, yaitu memiliki kemudi dan beberapa sistem serta karakteristik lain yang mempunyai sistem dan fungsi masing-masing. Dan juga memiliki perbedaan-perbedaan karakteristik antar pesawat terbang, misalnya kecepatan terbang, ketinggian terbang serta kegunaan dan fungsi dari setiap pesawat terbang. Referensi pesawat terbang yang dipakai dalam perancangan dan pembuatan tugas akhir ini adalah pesawat terbang CASA 212 dan OV-10 BRONCO.

2.1.2. Sistem Pengendali Kemudi Arah pada Ekor

Pada pesawat terbang casa 212 dan 1V-10 bronco, System control kemudi arah yang terpasang pada ekor (fin) dan dicontrol dari flight deck oleh dua pasang pedal yang dapat disetel. System kemudi arah (rudder) dilengkapi dengan trim tab dan servo tab (sebelah atas). Pergerakan kemudi arah (rudder) ini pada posisi sebelah kanan maksimal 35° dan pada posisi sebelah kiri maksimal 35° sedangkan tegangan cablenya 198 lbs pada temperatur 15° c dengan diameter cable $5/32$ inch. Komponen-pomponen kemudi arah ini adalah Kemudi (rudder) control ini adalah pedal, control rot, cable sigment, pulley dan quadrant shaft.



Gambar 2.2 Kemudi arah (Rudder)

2.2. Teori Keypad 4 x 4

Untuk mempermudah penggunaan mikrokontroller sebagai alat proses, maka diperlukan sarana yang dapat menjadi penghubung penggunaan dengan alat kontrol, yaitu sebagai sarana input data yang nantinya akan diolah oleh mikrokontroller.

Peralatan input data yang dapat menunjang mikrokontroller adalah beberapa saklar tekan yang menyatakan angka dan karakter yang disusun berbentuk matrik 4 kolom dan 4 baris dengan total tombol 16 buah. Keypad ini berfungsi untuk memberi masukan nilai derajat langsung ke minimum system. kemudian data ditampilkan pada ke LCD.

Rangkaian ini dapat dianalogikan dengan empat buah kabel terbuka yang disilang dengan empat buah kabel yang lain (diletakan diatasnya). Perlakuan ini akan menyebabkan perolehan 16 titik persilangan. Bila pada suatu titik kabel persilangan itu bersentuhan (salah satu ditekan hingga menyentuh kabel yang lain dibawahnya), maka diasumsikan bahwa tombol keypad pada posisi bersilangan tersebut ditekan. Berikut ini adalah gambar konfigurasi tombol Keypad :

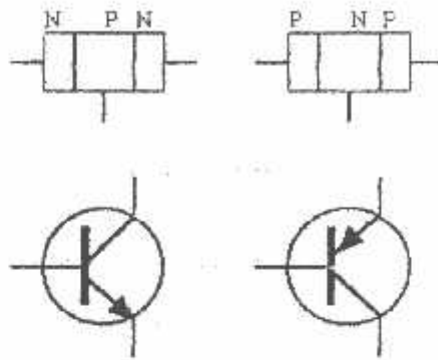
0	1	2	3
4	5	6	7
8	9	A	B
C	D	E	F

Gambar 2.3 Konfigurasi Tombol Keypad

2.3. Transistor

Transistor adalah suatu komponen aktif yang dibuat dari bahan semikonduktor. Apabila kita mendoping semikonduktor untuk mendapatkan kristal NPN atau PNP, maka kristal seperti ini disebut Transistor *Junction*. Daerah tipe N mempunyai banyak sekali *electron* pita konduksi dan daerah tipe P mempunyai banyak *Hole*. Jadi transistor *junction* mempunyai dua macam pembawa muatan yaitu *elektron bebas* pada daerah tipe N dan *hole* pada daerah tipe P. Oleh karena itu transistor *junction* sering disebut juga transistor dua kutub (*bipolar*).

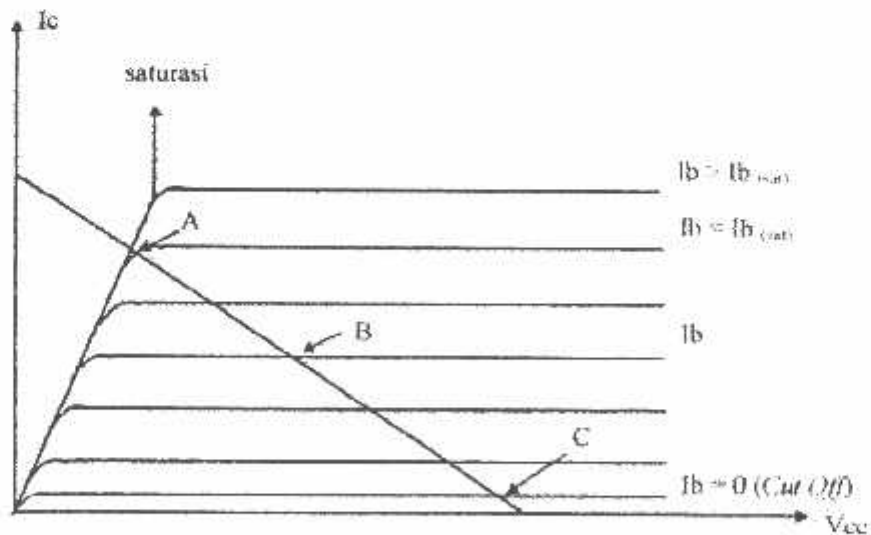
Transistor *bipolar* ada dua macam yaitu transistor jenis NPN dan PNP. Adapun symbol dari kedua transistor seperti terlihat pada gambar 2.3 dibawah ini :



Gambar 2.4 Simbol Transistor Bipolar

Daerah kerja transistor terbagi menjadi tiga yaitu : daerah kerja *cut off*, daerah kerja *saturasi* dan daerah kerja *aktif*.

Untuk lebih jelasnya mengenai daerah kerja transistor dapat dilihat pada gambar dibawah ini :



A = Penjenuhan B = Titik Kerja C = Titik Sumbat (*Cut Off*)

Gambar 2.5 Garis Beban Transistor

2.3.1. Tiga Daerah Transistor

Berdasarkan gambar 2.5 diatas dapat dilihat bahwa transistor mempunyai tiga daerah yaitu : Basis, Kolektor dan Emitor. Pada transistor NPN mempunyai dua *junction*, yaitu *junction* antara emitor dan basis serta *junction* antara basis dan kolektor. Oleh karena itu transistor bersifat seperti dua buah diode.

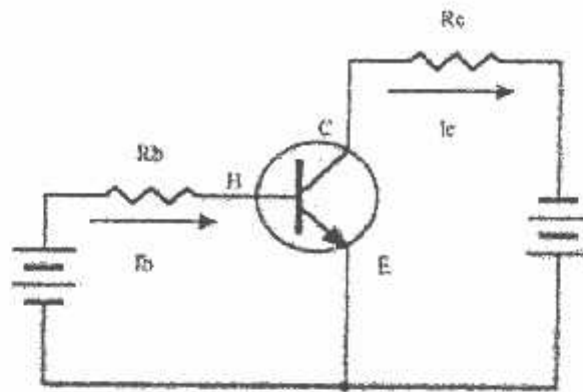
Sedangkan pada transistor PNP adalah kebalikan dari transistor NPN khususnya untuk karakteristik arus dan tegangannya.

2.3.2. Pembiasan Transistor Bipolar

Adapun beberapa cara pembiasan transistor antara lain :

1. Bias *forward-forward*

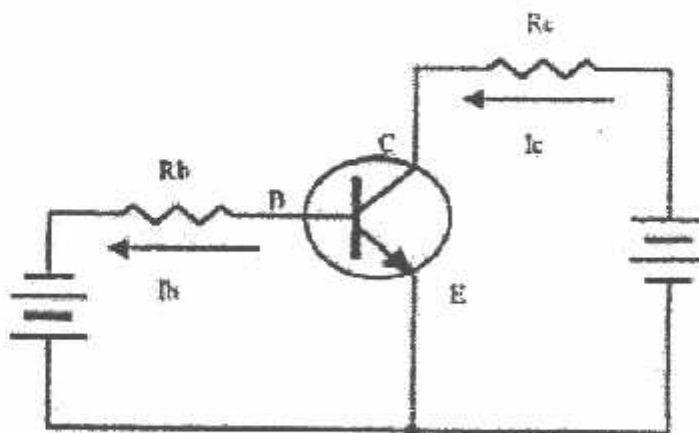
Gambar bias *forward-forward* seperti yang terlihat pada gambar 2.6 dibawah. Pada gambar tersebut menggunakan bias *forward-forward*, karena diode emitor dan diode kolektor bias *forward*. Pembawa electron melewati *junction* dan mengalir melaluhi basis. Tegangan Emitor-Basis (V_{BE}) membias *forward* diode emitor dan menghasilkan arus I_E . Tegangan Basis Kolektor (V_{BC}) membias *forward* diode kolektor yang akan menyebabkan arus I_C .



Gambar 2.6 Rangkaian Bias Forward-forward

2. Bias *Reverse-reverse*

Gambar rangkaian untuk bias *reverse-reverse* seperti terlihat pada gambar 2.7 dibawah ini :



Gambar 2.7 Rangkaian Bias Reverse-reverse

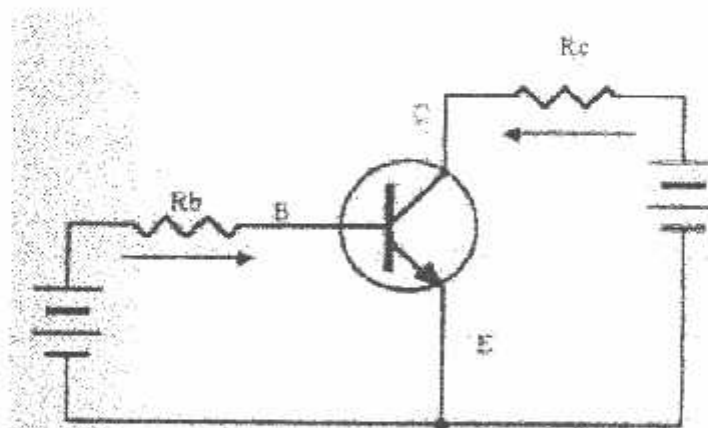
Pada rangkaian diatas, diode emitor dan diode kolektor membias *reverse*.

Untuk keadaan ini, maka hanya arus kecil yang mengalir terdiri dari arus

saturasi yang dihasilkan secara termal dan arus bocor permukaan. Komponen yang dihasilkan secara termal tergantung pada temperatur dan akan naik dua kali setiap kenaikan 10°C . komponen bocor permukaan akan bertambah besar dengan bertambahnya tegangan. Arus reverse ini kecil sekali sehingga bisa diabaikan. Sehingga pada konfigurasi bias ini, kedua diode bertindak sebagai saklar yang terbuka dengan arusnya sangat kecil maka dapat diabaikan.

3. Bias *forward-reverse*

Gambar bias transistor *forward-reverse* seperti terlihat pada gambar 2.8 dibawah ini :



Gambar 2.8 Rangkaian Bias Forward-reverse

Pada gambar diatas diode emitor bias *forward*, diode kolektor bias *reverse* sehingga terjadi hal yang tidak diharapkan. Dimana seharusnya arus emitor yang besar karena diode emitor dibias *forward*, tetapi kita tidak mengharapkan arus kolektor yang besar karena diode kolektor

dibias *reverse*. Tetapi hal inilah yang terjadi, sehingga transistor merupakan suatu penemuan yang besar.

Pada saat bias *forward* diberikan pada diode emitor, electron-elektron bias dalam emitor belum memasuki daerah basis. Jika V_{BE} lebih besar dari pada *potensial barrier*, maka banyak electron bebas emitor memasuki daerah basis. Electron –elektron bebas ini dalam basis dapat mengalir kedua arah, kebawah basis yang tipis menuju kaki basis atau melewati daerah kolektor.

Agar electron bebas mengalir kebawah basis, mereka harus terlebih dahulu masuk atau jatuh kedalam *hole* yaitu rekombinasi dengan *hole* dalam basis. Kemudian sebagai *electron valensi* mereka dapat mengalir kebawah melalui *hole* basis yang berdekatan dan kedalam kaki basis. Maka komponen ke bawah dari arus basis ini disebut arus rekombinasi, dan arus ini kecil karena basis *didrop* sedikit sehingga hanya memiliki *hole* yang sedikit.

Karena lapisan basis sangat tipis, maka setelah basis sedang penuh dengan electron-elektron bebas yang dimasuk, akan menyebabkan *difusi* kedalam lapisan pengosongan. Sekali memsuki lapisan ini mereka didorong oleh medan lapisan pengosongan kedalam lapisan kolektor. Elektron-elektron ini kemudian mengalir melaluhi kaki kolektor.

Pada diode emitor, aliran electron bebas dari terminal negatif sumber munuju daerah emitor dan karena operasinya bias *forward* maka electron bebas emitor akan masuk kdacrah basis. Basis yang *didrop* sedikit akan

memberikan waktu aktif yang cukup pada hampir semua electron bebas untuk berdifusi kedalam lapisan pengosongan kolektor. Elektron bebas ini meninggalkan kolektor dan memasuki terminal positif sumber tegangan kolektor. Pada umumnya untuk transistor bipolar lebih besar sama dengan 95% electron bebas yang disuntikan dari emitor akan mengalir ke kolektor dan kurang dari sama dengan 5% electron bebas jatuh kedalam hole basis dan mengalir keluar ke kaki basis.

2.3.3 Transistor Sebagai Saklar

Adapun transistor yang digunakan sebagai saklar pada skripsi ini adalah transistor C9013 dan BD139. transistor ini merupakan transistor bipolar dengan tipe NPN.

Transistor yang digunakan sebagai saklar harus dikondisikan dalam dua kondisi yaitu :

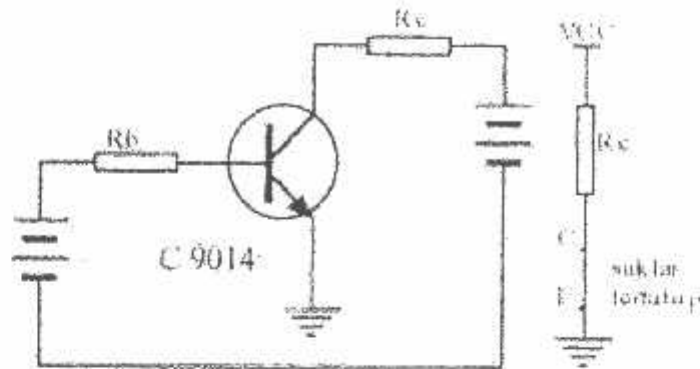
1. Transistor dalam keadaan *jenuh* (saturasi)

@ Kuat arus (I_C) mencapai maksimum

@ Nilai V_{cc} sama dengan 0 volt

@ Tegangan pada beban sama dengan tegangan sumber ($V_{cc} = V_{Re}$)

Untuk lebih jelas dapat dilihat pada gambar 2.9 dibawah ini :



Gambar 2.9 Transistor Dalam Keadaan Saturasi

Untuk menghitung resistansi pada basis menggunakan rumus :

$$V_{cc} - I_c \cdot R_c - V_{ce} = 0$$

Karena keadaan saturasi $V_{ce} = 0$ sehingga rumus menjadi :

$$V_{cc} - I_c \cdot R_c = 0$$

Maka arus kolektor dan arus basis :

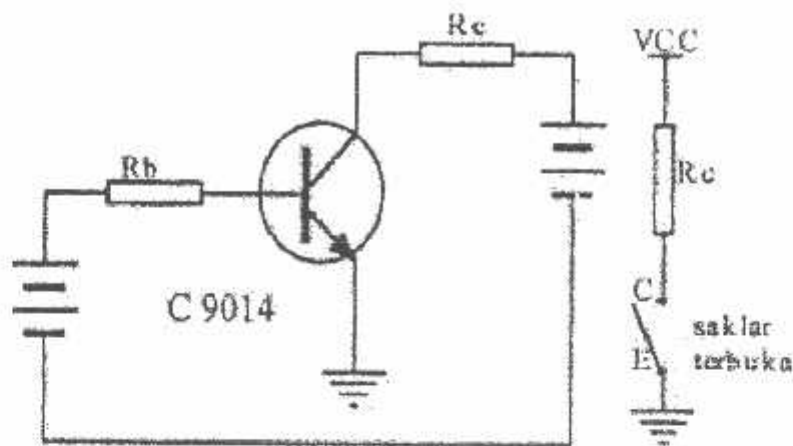
$$I_c = \beta_{dc} \cdot I_b$$

$$I_b = \frac{V_{cc} - V_{be}}{R_b}$$

2. Transistor Dalam Keadaan *Cut Off* (Sumbat/Open sirkuit)

Transistor dalam keadaan cut off ini berlaku hal-hal sebagai berikut :

- @ Nilai arus I_b sama dengan 0 volt
- @ Nilai arus I_c sangat kecil sekali sehingga dapat diabaikan.
- @ Besarnya V_{cc} sama dengan V_{ce}

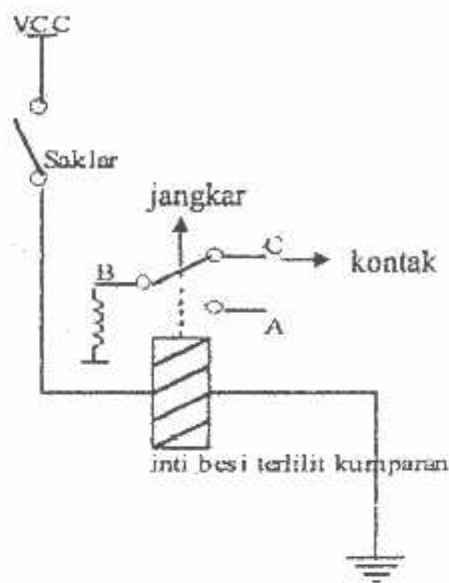


Gambar 2.10 Transistor Dalam Keadaan Cut Off

2.4. Relay

Relay adalah komponen elektronik yang terdiri dari sebuah lilitan kawat (kumparan/koil) yang terlilitkan pada sebuah besi lunak. Jika kumparan dialiri arus listrik maka inti besi akan menjadi medan magnet dan menarik pegas sehingga kontak AB terhubung dan BC terputus, begitupun juga sebaliknya jika kumparan tidak dialiri arus listrik maka inti besi kehilangan medan magnet sehingga kontak AB terputus dan kontak BC terhubung.

Relay merupakan suatu alat untuk menghubungkan atau memerlukan kontak antara komponen yang satu dengan yang lain. Dalam memutus, menghubungkan atau membalik fase kontak digerakan oleh fluksi yang ditimbulkan dari adanya medan magnet listrik yang dihasilkan oleh kumparan yang melilit pada besi lunak.



Gambar 2.11 Cara Kerja Relay

Relay yang digunakan untuk pembalik tegangan dc dalam skripsi ini adalah DPDT (*Dual Pin Dual Terminal*) dengan tegangan input 12 Volt dc, seperti yang tertera pada gambar 2.15 dibawah.

Ada beberapa jenis susunan kontak relay dimana semuanya terisolasi terhadap arus listrik yang ada didalam kumparan. Jenis susunan kotak relay adalah :

@ Normally Open (*Normal terbuka*)

yaitu kontak-kontak tertutup pada saat kumparan relay dialiri arus.

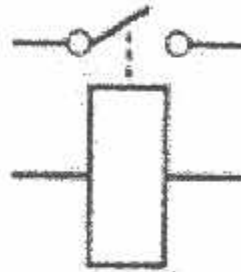
@ Normally Close (*Normal tutup*)

yaitu kontak-kontak terbuka pada saat kumparan relay dialiri arus

Macam-macam relay yaitu :

1. SPST (*Single Pin Single Terminal*)

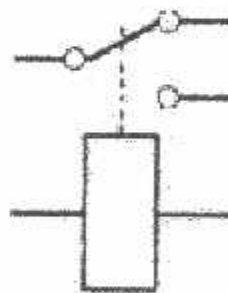
Simbol relay SPST



Gambar 2.12 Jenis Relay SPST

2. SPDT (*Single Pin Dual Terminal*)

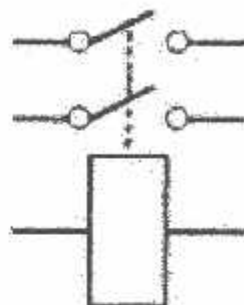
Simbol relay SPDT



Gambar 2.13 Jenis Relay SPDT

3. DPST (*Dual Pin Single Terminal*)

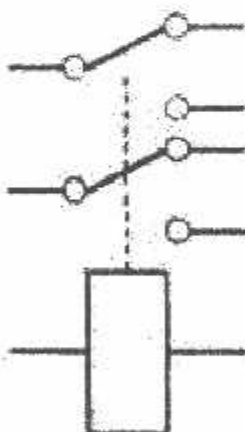
Simbol relay DPST



Gambar 2.14 Jenis Relay DPST

4. DPDT (*Dual Pin Dual Terminal*)

Simbol relay DPDT



Gambar 2.15 Jenis Relay DPDT

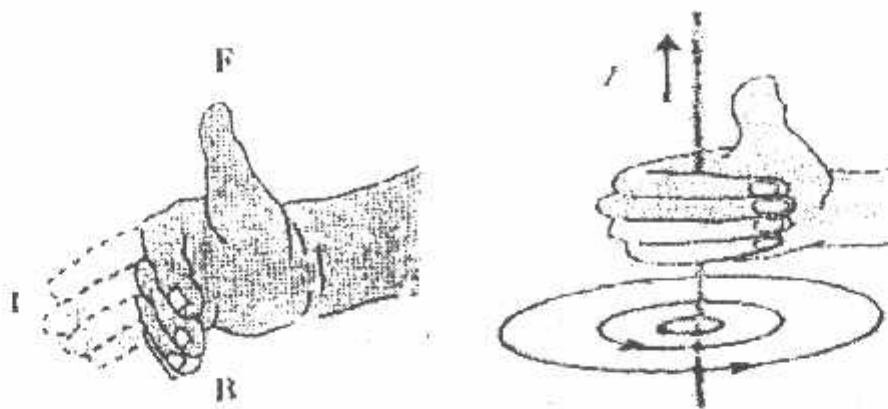
2.5. Motor DC

Motor dc adalah suatu motor penggerak yang dikemudikan dengan arus searah (DC), dan merupakan salah satu bentuk mesin arus searah yang akan berfungsi bila ;

- # Kumparan medan, untuk menghasilkan medan magnet.
- # Kumparan jangkar, untuk mengimbaskan ggl pada konduktor-konduktor yang terletak pada alur-alur jangkar.
- # Celah udarah, yang memungkinkan berputarnya jangkar dalam medan magnet.

2.5.1 Prinsip Kerja Motor DC

Apabila kumparan jangkar dari mesin arus searah dialiri arus dan kumparan medan diberi penguatan, maka akan timbul gaya *lorentz* pada setiap sisi kumparan jangkar. Arah medan magnet dapat ditentukan dengan kaidah tangan kanan dengan ibu jari menunjukkan arah gaya putar dari arus listrik yang mengalir dalam sebuah kumparan jangkar yang berada dalam medan magnet, jari tengah menunjukkan arah arus listrik yang mengalir pada konduktor, dan jari telunjuk menunjukkan arah medan magnet.



Gambar 2.16 Kaidah Tangan kanan

Adapun besarnya *gaya lorentz* yang bekerja pada kumparan jangkar tersebut adalah :

$$F = B \cdot I \cdot L$$

Gaya lorentz tersebut menimbulkan kopel sehingga jangkar berputar.

Besarnya kopel tersebut adalah :

$$M = F \cdot r \quad \text{dyne} \cdot \text{cm}$$

$$M = B \cdot L \cdot r \quad \text{dyne} \cdot \text{cm}$$

$$M = C \cdot \Phi \cdot La \quad \text{dyne} \cdot \text{cm}$$

Dimana :

F = *Gaya lorentz*

B = Kerapatan fluks magnet (weber)

L = Panjang konduktor (meter)

I = Arus listrik (ampere)

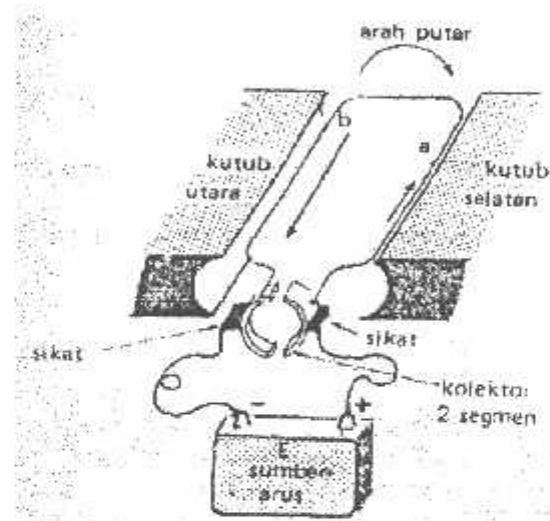
R = Jari-jari jangkar (cm)

Φ = Fluk tiap kutub (Maxwell)

C = Constanta

B = Kerapatan medan magnet (gauss)

I_a = Arus jangkar (ampere)

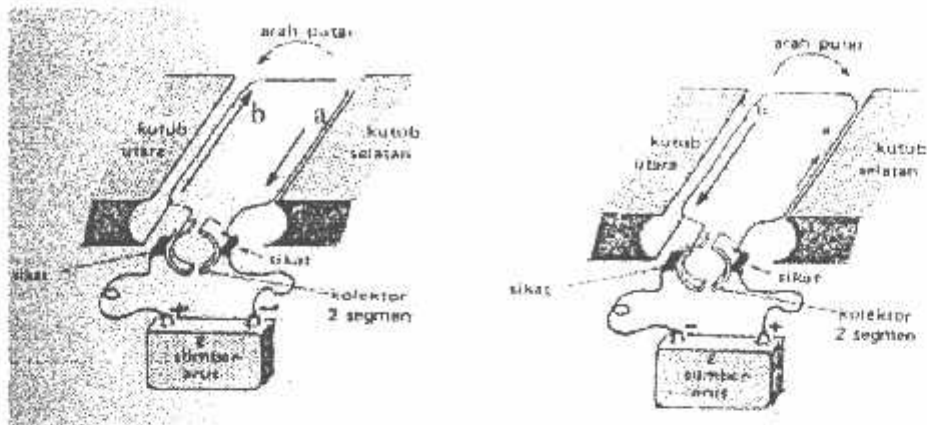


Gambar 2.17 Cara Kerja Motor DC

Perputaran kumparan jangkar dalam medan magnet dan konduktor jangkar yang dialiri arus menimbulkan kopel yang memotong medan magnet tersebut sehingga pada konduktor jangkar akan timbul tegangan induksi (ggl). Ggl ini arahnya berlawanan dengan tegangan jepit jangkar, sehingga disebut G.G.L lawan yang besarnya :

$$E_l ; C \cdot n \cdot \Phi \quad \text{Volt}$$

Rumus-rumus kerja motor arus searah adalah sebagai berikut :



Gambar 2.19 Pengendali Arah Putaran Motor DC

Pada gambar 2.19 diatas, agar arah putar motor dc berubah, maka polaritas tegangan pada catu daya harus dibalik.

2.6. Sensor

Besaran masukan pada kebanyakan sistem instrumentasi relatif bukan besaran listrik sehingga digunakan cara manipulasi untuk mengontrolnya, untuk memanipulasinya diperlukan sebuah transduser.

Definisi transduser adalah suatu alat yang digerakan oleh suatu energi didalam sebuah transmisi dan akan menyalurkan energi dalam bentuk yang sama atau dalam bentuk yang berlainan kesistim transmisi yang kedua (William D. Cooper, 199:384).

Berdasarkan prinsip listrik yang bersangkutan, transduser di bedakan menjadi :

1. Transduser pasif, yaitu transduser yang memerlukan daya dari luar, contohnya adalah potensiometer, LVDT dan Strain gage.
2. Transduser aktif, yaitu transduser yang tidak memerlukan daya luar karena mempunyai pembangkit sendiri, contohnya adalah sel foto tegangan, piezoelektronik.

Dalam penyusunan skripsi ini akan dipakai transduser pasif yaitu jenis potensiometer.

2.6.1. Sensor Sudut

Sensor sudut yang dipakai adalah Potensiometer yang merupakan sebuah alat elektromekanik yang mengandung elemen tahanan yang dihubungkan dengan sebuah kontak geser yang dapat bergerak. Gerakan dari kontak geser ini menghasilkan suatu perubahan tahanan yang linier, logaritmik, eksponensial dan sebagainya. Pada skripsi ini potensiometer berfungsi untuk mengukur perubahan pergeseran sudut pada pengendali kemudi arah. Rangkaian dalam teori potensiometer terdiri dari beberapa resistor yang dihubungkan seri dan output dari potensiometer tersebut berupa tegangan.

Tegangan output dari potensiometer akan berubah-ubah sesuai dengan perubahan dari sudut yang diukur yang berarti adalah konversi dari posisi mekanik menjadi sesuatu besar tegangan. Pada dasarnya suatu presisi potensiometer terdiri dari elemen resistif dengan angker yang dapat bergerak atau silinder yang dihubungkan dengan elemen. Sebagai angker

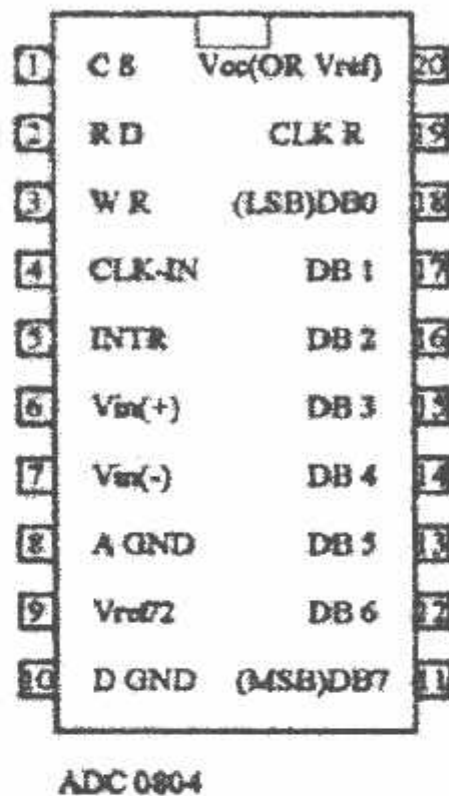
yang berputar, variasi-variasi tahanan diantara dua yaitu akhir dari elemen tahanan dan silinder, menunjukkan posisi sumbu.

Kelebihan dari transduser jenis ini adalah memiliki efisiensi listrik yang sangat baik dan dapat memberikan suatu keluaran yang cukup untuk memperbolehkan suatu operasi pengontrolan tanpa harus ada penguatan. Disamping itu jenis transduser ini bisa mendapat input tegangan dari AC maupun dari DC sehingga dapat melayani cakupan pemakaian yang cukup luas. Sedangkan kelemahan dari transduser ini memiliki umur pemakaian yang terbatas dan cepat menjadi derau, sebagai akibat dari gesekan mekanis dari kontak geser terhadap elemen tahanan.

2.7. Analog To Digital Converter

Apa bila di dalam suatu sistem elektronik hanya dapat mengolah data dalam bentuk biner saja atau disebut prosesan besaran digital, maka setiap data analog yang akan diproses oleh sistem tersebut harus diubah terlebih dahulu ke dalam bentuk biner (digital).

Piranti dan sistem elektronika hanya mengenal isyarat digital. Tetapi kebanyakan isyarat seperti isyarat yang dihubungkan dengan transduser mempunyai bentuk analog. Sebelum diumpankan ke sistem digital, isyarat-isyarat yang harus diubah ke dalam bentuk digital oleh pengubah analog ke digital (ADC).



Gambar 2.20 Konfigurasi Pena-pena ADC 0804

Jadi untuk menghubungkan sistem digital suatu rangkaian sistem analog dari dunia nyata memerlukan suatu pengubah (*converter*) sistem analog ke sistem digital, yang biasanya dikenal dengan nama ADC (*Analog to digital Converter*). Fungsi dasar dari *converter A/D* adalah mengubah tegangan analog ke dalam kode-kode biner (*Digit*) sehingga dapat diolah oleh system digital.

Tegangan analog merupakan masukan bagi ADC yang berasal dari rangkaian sensor atau rangkaian penguatan dimana berubah-ubah yang

terjadi pada ADC sesuai dengan perubahan dari pergeseran yang pada sensor potensiometer. Inputan tegangan analog ini di ubah oleh ADC menjadi bentuk digital yang sebanding dengan tegangan analognya. Kode biner hasil konversi inilah yang kemudian dipakai sebagai data yang diolah oleh rangkaian digital yang salah satunya adalah mikrokontroller AT89S51 melalui data busnya. Pengubahan analog ke digital merupakan jantung dari sistem data yang berfungsi untuk mengubah data kedalam bentuk digital sehingga cocok untuk diproses oleh minimum sistem mikrokontroller AT89S51.

Dalam tugas akhir ini digunakan ADC 0804 dari national semi konduktor. ADC jenis ini adalah jenis SAR (*Successive Aproximation Register*) ini memiliki rangkaian yang lebih kompleks dibandingkan jenis ADC yang lainnya. Dikatakan demikian karena jenis ini peralatan konversinya tidak menggunakan counter untuk memberikan input blok konverter digital ke analognya sebagai ganti konverter ini menggunakan register yang disebut *Successive Aproximation Register* (SAR), serta menggunakan sebuah konverter digital ke analognya yang mengontrol kerjanya sendiri.

Keuntungan utama dari pemakaian SAR ini adalah sebagai resolusi bit hanya membutuhkan selang waktu n buah pulsa clock untuk konversi. rangkaian umumnya terletak pada register yang mempunyai keluaran digital.

Pada awal start konversi akan aktif yang mana akan meng-clear data yang sebelumnya, pada saat ini MSB (*most significant bit*) '1', apa bila

kontrol register ini menyatakan keluaran kurang besar, maka bit yang diset tetap dan bit yang berikutnya akan diset. Sebaliknya apa bila kontrol yang menyatakan keluaran yang terlalu besar, maka bit awal yang diset akan mereset kembali dan bit berikutnya akan diset. Urutan ini akan dikerjakan terus sampai urutan bit terakhir, sehingga data yang paling sebanding dengan tegangan analognya.

ADC ini termasuk dalam tipe 08xx dengan satu buah masukan analog dan delapan buah output digital. ADC ini mempunyai pendekatan berturut-turut seperti yang diterangkan diatas sehingga ADC ini mempunyai kecepatan tinggi dalam konversi. ADC jenis ini tersedia secara komersial sebagai rangkaian terpadu IC dengan resolusi 8 bit sampai 16 bit, dan mudah di interfacekan dengan mikrokontroler.

ADC 0804 mempunyai pembangkit pulsa internal (on chip) bekerja dengan 2,5 pada $V_{ref}/2$. ADC ini mempunyai range antara 0V sampai 5V dengan menggunakan pencatu daya + 5Volt dan mempunyai waktu konversi optimum sekitar 100us. Frekuensi clock dari konverter harus terletak dalam daerah frekuensi 100-800 KHz sedangkan harga idealnya adalah 640KHz. ADC 0804 mempunyai clock internal yang dipasang pada free running dengan menggunakan komponen resistor dan kapasitor untuk mendapatkan sinyal clock. Frekuensi yang dihasilkan oleh pembangkit pulsa atau clock ini sesuai dengan rumus :

$$f = \frac{1}{1.1.RC}$$

2.8. Mikrokontroler AT89S51

2.8.1. Pendahuluan

Perbedaan mendasar antara mikrokontroler dan mikroprosesor adalah bahwa mikrokontroler selain memiliki CPU juga dilengkapi memori dan input output yang merupakan kelengkapan sebagai sistem minimum mikrokomputer sehingga sebuah mikrokontroler dapat dikatakan sebagai mikrokomputer dalam keping tunggal (Single Chip Microcomputer) yang dapat berdiri sendiri.

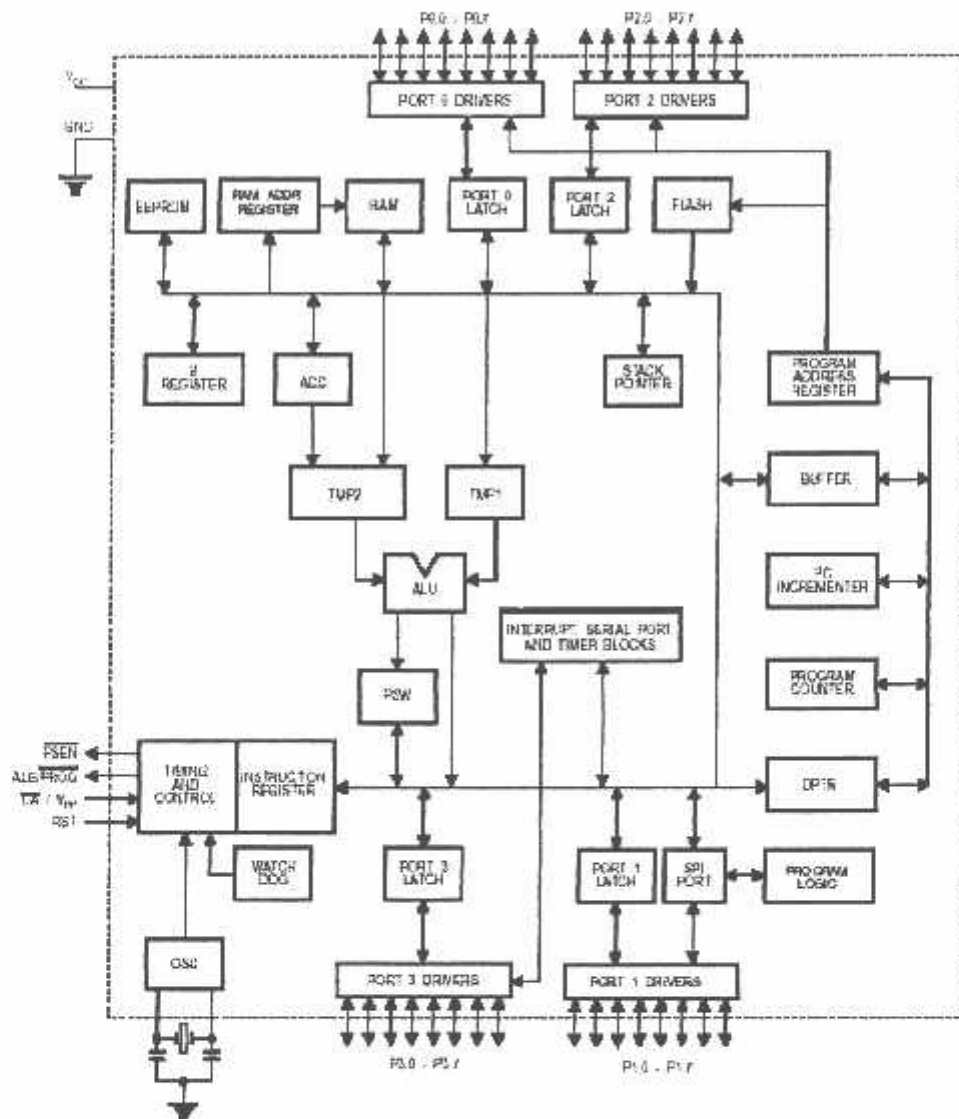
Mikrokontroler AT89S51 adalah mikrokontroler ATMEL yang kompatibel penuh dengan mikrokontroler keluarga MCS - 51, membutuhkan daya rendah, memiliki performance yang tinggi dan merupakan mikrokomputer 8 bit yang dilengkapi 4 Kbyte PEROM (*Programmable Erasable Read Only Memory*) dan 128 Byte RAM internal. Program memori yang dapat dihapus atau diprogram ulang dalam sistem atau menggunakan programmer Nonvolatile Memory konvensional.

Dalam sistem mikrokontroler terdapat dua hal yang mendasar, yaitu perangkat lunak dan perangkat keras yang keduanya saling berkaitan dan mendukung.

2.8.2. Perangkat keras mikrokontroler AT89S51

Secara umum Mikrokontroler AT89S51 memiliki :

- @ Cpu 8 bit termasuk keluarga MCS – 51
- @ 16 bit program counter (PC) dan data pointer (DPTR)
- @ 8 bit program status word (PSW)
- @ 8 bit stack pointer (SP)
- @ 4 Bank register
- @ 128 byte Internal RAM
- @ 4 buah Port I/O, masing-masing terdiri atas 8 jalur I/O
- @ 2 Timer / counter 16 bit
- @ 1 Serial Port Full Duplex
- @ Control register, yaitu : TCON< TMOD< CSON< PCON< IP< dan IE
- @ 5 buah sumber interupt (2 buah sumber interupt external dan 3 buah sumber interupt internal)
- @ Osilator dan Clock Internal

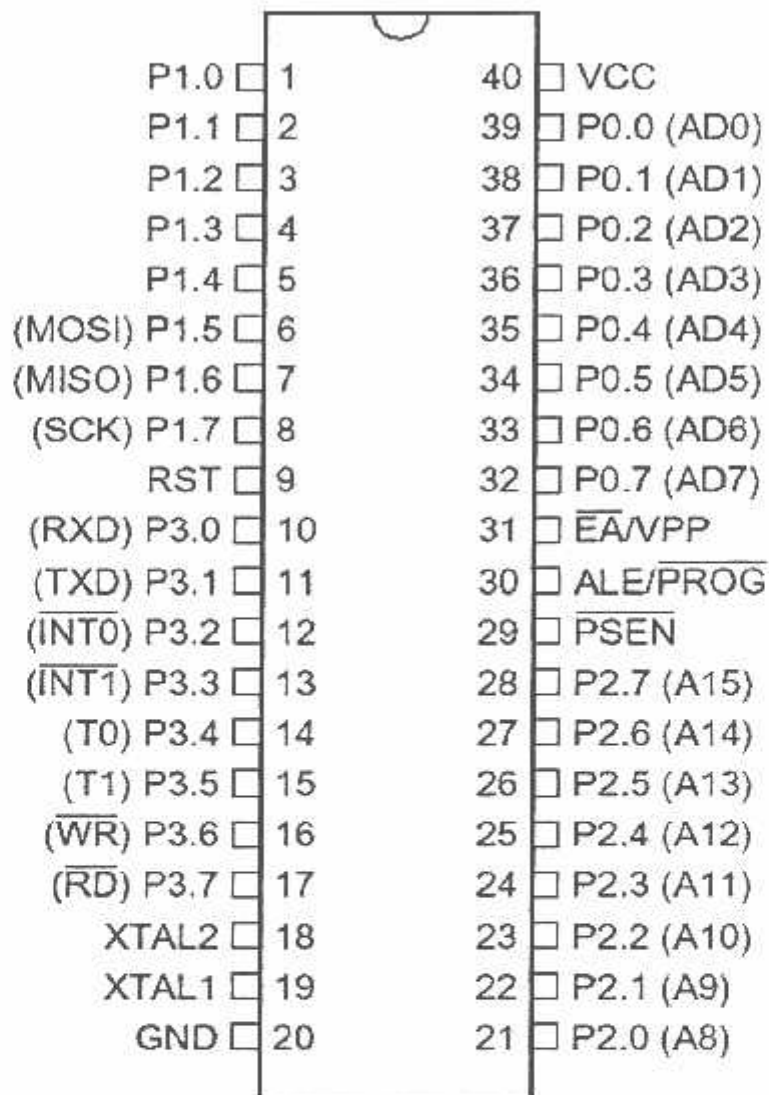


Gambar 2.21 Block Diagram Mikrokontroler AT89S51

2.8.3. Konfigurasi Pena-Pena Mikrokontroler AT89S51

Mikrokontroler AT89S51 terdiri dari 40 pin dengan konfigurasi sebagai berikut:

PDIP



Gambar 2.22 Konfigurasi Pena-pena AT89S51 [4]

Fungsi tiap-tiap pin-nya adalah sebagai berikut :

- @ VCC (Supply tegangan), pin 40
- @ GND (Ground), pin 20

@ Port 0, pina 32 – 39

merupakan port I/O dua arah dan dikonfigurasi sebagai multiplex dua bus alamat rendah (A0-A7) dan data selama pengaksesan program memori dan internal

@ Port 1, pin 1 – 8

Merupakan port I/O dengan dua arah dan dengan internal Pull-Up. Ketika diberikan logika 1 pin ini akan di Pull Up secara internal sehingga dapat digunakan sebagai input. Sebagai inputan pin – pin ini di hubungkan ke ground maka masing-masing pin ini dapat menghantarkan arus karena di Pull High secara internal. Port 1 juga menerima Low Order adder bytes selama melakukan verifikasi program.

Pada port 1 di AT89S51 pin ini mempunyai alternatif seperti pada tabel berikut ini ;

Tabel 2.1 Fungsi Khusus pada Port 1

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

@ Port 2, pin 21 – 28

Merupakan port I/O Bi-directional yang dilengkapi dengan internal Pull-Up. Mengeluarkan alamat tinggi selama pengambilan program memory external. Ini menggunakan perintah dengan alamat 16 bit (misalnya MOVX @DPTR).

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

@ Port 3, pin 10 – 17

Merupakan port I/O Bi-directional yang dilengkapi dengan Pull – Up. Jika diberikan logika “1” pada pin-pin port 3 maka masing-masing Pull Hig oleh Pull Up internal sehingga dapat digunakan sebagai inputan, jika pin-pin port 3 dihubungkan ke gound, maka masing-masing pin akan memberikan arus karena di Pull High secara internal

Dan port 3 juga memiliki fungsi khusus dan dapat dilihat pada tabel berikut ini :

Tabel 2.2 Fungsi Khusus pada Port 3

Nama Penyemat	Fungsi khusus
Port 3.0	RxD (Port masukan serial)
Port 3.1	TxD (Port keluaran serial)
Port 3.2	/INT0 (Masukan Interupsi Eksternal 0)
Port 3.3	/INT1 (Masukan Interupsi Eksternal 1)
Port 3.4	T0 (masukan pewaktu eksternal 0)
Port 3.5	T1 (masukan pewaktu eksternal 1)
Port 3.6	/WR (sinyal tulis memori data eksternal)
Port 3.7	/RD (sinyal baca memori data eksternal)

@ RST Reset, pin 9

Input Reset merupakan reset master untuk AT89S5. Inputan reset akan memberikan logika high "1" pada pin ini dengan jangka waktu yang ditentukan oleh lamanya pengosongan data muatan kapasitor. Jangka waktu minimal adalah 2 (dua) siklus mesin (24 priode frekuensi clock).

@ ALE / Prog (Address Lacth Enable), pin 30

Pulsa output ALE digunakan untuk proses-proses *'latching' byte address* rendah selama pengaksesan ke memori eksternal.

Digunakan untuk menahan alamat memori eksternal selama pelaksanaan instruksi.

@ PSEN (Program Strobe Enable), pin 29

Merupakan sinyal baca ke program memory eksternal

@ EA / VPP (External Access Enable), pin 31

Dapat diberikan logika rendah (Ground) atau logika tinggi (+5V).

Jika diberikan logika tinggi maka mikrokontroller akan mengakses program dari ROM internal (EEPROM/Flash Memory), dan jika diberikan logika rendah maka mikrokontroller akan mengakses program dari memory eksternal.

@ X – TAL 1 dan X – TAL 2, pin 19, 18

Pin ini dihubungkan dengan kristal / sumber osilator dari luar, dan bila menggunakan penguatan osilator internal. X-TAL 1 merupakan masukan ke penguatan osilator internal berpenguatan tinggi. Sedangkan X-TAL 2 adalah keluaran dari rangkaian penguatan osilator internal. Untuk keperluan ini diperlukan kapasitor penstabil sebesar 30pF. Dan nilai dari X- TAL tersebut antara 4 – 24 Mhz.

Metode pengamatan

Byte Address	Alamat BIT Bit Address	Byte Address	Alamat BIT Bit Address
7F	<div style="border: 1px solid black; padding: 5px; text-align: center;"> GENERAL PURPOSE RAM (RAM UNTUK SEGALA KEPERLUAN) </div>	FF	
		F7 F6 F5 F4 F3 F2 F1 F0	B
		E7 E6 E5 E4 E3 E2 E1 E0	ACC
		D7 D6 D5 D4 D3 D2 - D0	PSW
		- - - BC BB BA B9 B8	IP
		B7 B6 B5 B4 B3 B2 B1 B0	P3
		AF - - AC AB AA A9 A8	IE
		A7 A6 A5 A4 A3 A2 A1 A0	P2
		not bit addressable	SBUF
	9F 9E 9D 9C 9B 9A 99 98	SCON	
	97 96 95 94 93 92 91 90	P1	
	not bit addressable	TH1	
	not bit addressable	TH0	
	not bit addressable	TL1	
	not bit addressable	TL0	
	not bit addressable	TMOD	
	8F 8E 8D 8C 8B 8A 89 88	TCON	
	not bit addressable	PCON	
	not bit addressable	DPH	
	not bit addressable	DPL	
	not bit addressable	SP	
	87 86 85 84 83 82 81 80	P0	

RAM

Spesial Function Register

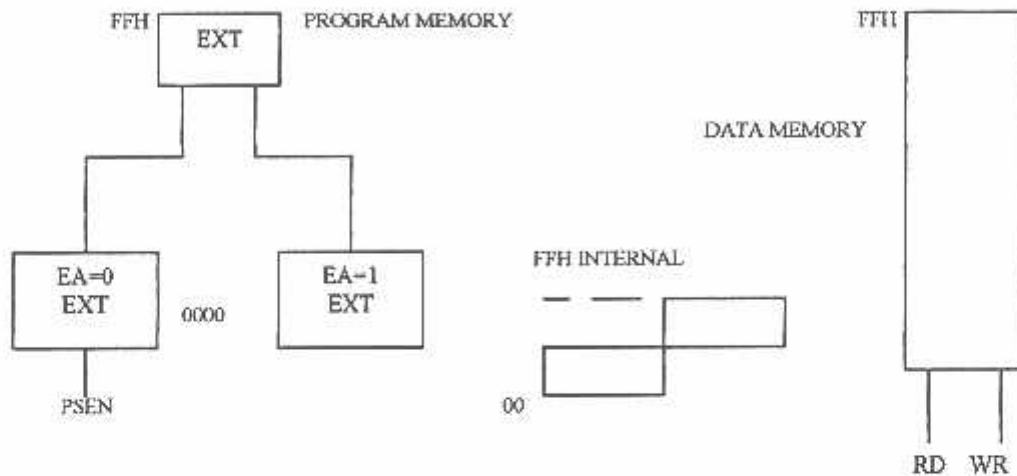
Gambar 2.23 Alamat RAM

2.8.4. Organisasi Memory

2.8.4.1. Pemisahan Memory Program dan Data

Semua produk mikrokontroller Flash AT89S51 Atmel memiliki ruang alamat memory data dan program yang terpisah, sebagaimana ditunjukkan pada gambar 2.24. Pemisahan memory program dan data

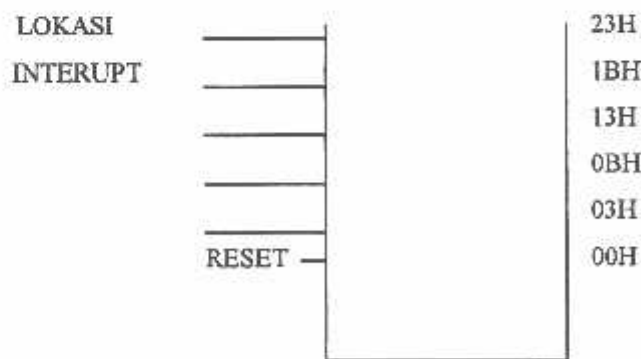
tersebut membolehkan memori data diakses dengan alamat 8 bit. Namun demikian, alamat memori data 16 bit biasa juga dihasilkan melalui register DPTR.



Gambar 2.24 Struktur Memori [1]

2.8.4.2. Memory Program

Pada gambar 2.25 ditunjukkan pemetaan bagian bawah dari memory. Setelah reset, CPU segera mengerjakan program mulai dari lokasi 0000h.



Gambar 2.25 Memori Program [1]

Sebagai mana ditunjukkan pada gambar 2.25, masing-masing interupsi disimpan pada lokasi yang sudah tetap dalam memory program. Sebuah interupsi menyebabkan CPU melompat ke lokasi interupsi yang bersangkutan. Yaitu letak subrutin layanan interupsi tersebut. Misalnya, *External Interrupt 0* disimpan pada lokasi 0003h, jika interupsi ini digunakan maka layanan rutin interupsi ini harus dituliskan pada lokasi ini, jika tidak maka lokasi tersebut dipakai sebagai memory program serbaguna (untuk keperluan lain).

Lokasi-lokasi layanan interupsi tersebut menempati lokasi-lokasi dengan jarak 8 byte. 0003h untuk *External interrupt 0*, 000Bh untuk *timer 0*, 0013h untuk *External Interrupt 1*, dan seterusnya. Jika seluruh rutin layanan interupsi sangat pendek (kurang dari 8 byte) maka harus digunakan suatu perintah lompat ke lokasi rutin interupsi yang sebenarnya (di lokasi dalam memori program).

Alamat-alamat yang paling bawah dari memory program dapat berada dalam *flash on – chip* maupun eksternal. Untuk melakukan hal ini dilakukan pengkabelan pada pin EA atau *External Access* ke Vcc atau ke GND (akses eksternal) sesuai dengan kebutuhan.

2.8.4.3. Memory Data (RAM) Internal

Ruangan alamat bahwa memory data (RAM) internal dengan kapasitas 128 byte yaitu :00H-7FH yang terbagi atas 3 daerah, yaitu:

@ Empat bank register

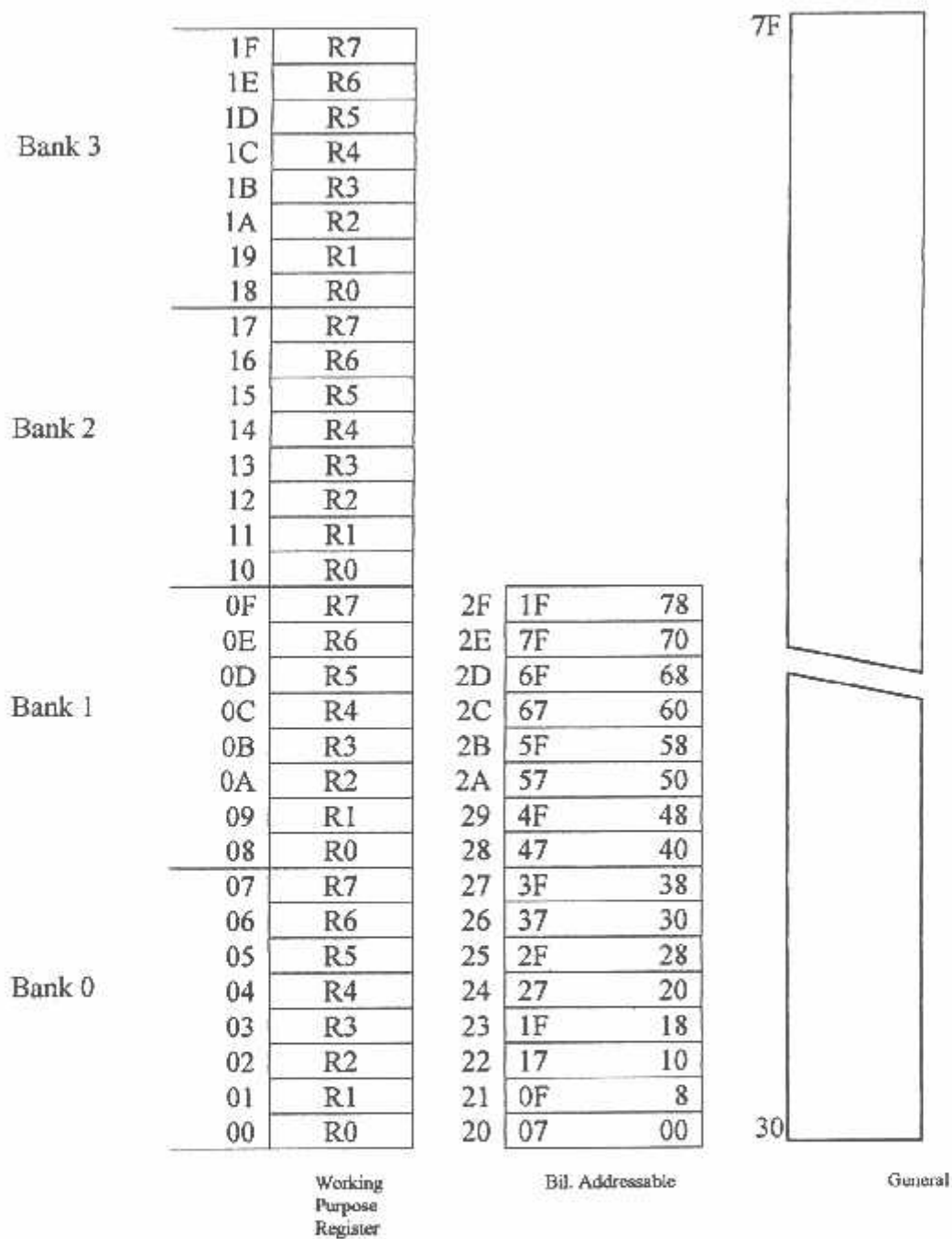
Setiap bank terdiri dari 8 register (R0-R7) sehingga jumlah register untuk keempat bank register menjadi 32 buah register yang menempati ruang alamat 00H-1FH. Mengaktifkan salah satu bank register dapat dilakukan dengan mengatur RS0-RS1 pada program *status word* (PSW).

@ Bit Addressable

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

@ General Purpose Area

Terdiri atas 80 byte yang menempati alamat 30H-7FH. Yang dapat dialamati secara langsung dan digunakan untuk keperluan umum (*general purpose*) misalnya digunakan untuk lokasi stack.



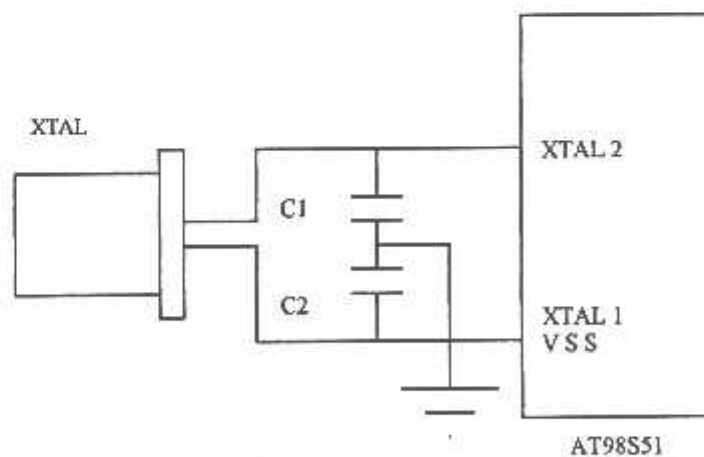
Gambar 2.26 Organisasi RAM Internal

Tabel 2.3 Pengaturan RS1-RS0 Select Register Bank

RS 1	RS 0	Select Register Bank
0	0	Bank 1
0	1	Bank 2
1	0	Bank 3
1	1	Bank 4

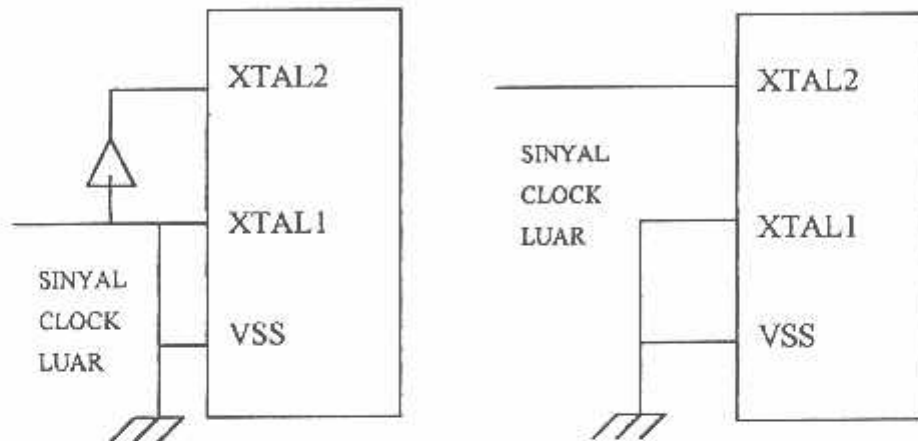
2.8.5. Waktu CPU

Mikrokontroler AT89S51 memiliki osilator internal (on chip oscillator) yang dapat digunakan sebagai sumber clock bagi CPU. Dalam penggunaan seperti terlihat pada gambar 2.27 ini :



Gambar 2.27 Menggunakan Osilator internal[2]

Bila menggunakan clock eksternal maka menggunakan rangkaian seperti pada gambar 2.28.



Gambar 2.28 Menggunakan Sumber Clock Eksternal[2]

2.8.6. Sistem Interupsi

Mikrokontroller AT89S51 mempunyai 5 buah sumber interupsi yang dapat membangkitkan permintaan interupsi, yaitu INT0, INT1, T1, T2 dan Port Serial.

Saat terjadinya interupsi mikrokontroller secara otomatis akan menuju ke subrutin pada alamat tersebut. Setelah interupsi selesai dikerjakan, mikrokontroller akan mengerjakan program semula. Tiap-tiap sumber interupsi dapat enable atau disable secara software.

Tingkat prioritas semua sumber *interrupt* dapat diprogramkan sendiri-sendiri dengan set atau clear bit pada (Interrupt Priority). Jika dua permintaan

interupsi dengan tingkat prioritas yang berbeda diterima secara bersamaan, maka permintaan interupsi dengan prioritas tertinggi yang akan dilayani. Jika permintaan interupsi dengan prioritas yang sama diterima bersamaan, maka akan dilakukan polling untuk menentukan mana yang akan dilayani. Bit-bit pada IP adalah sebagai berikut:

MSB		LSB
-	-	-
-	PS	PT1
-	PX1	PT0
-	PX0	-

Keterangan :

- @ Priority bit = 1 menandakan prioritas tinggi
- @ Priority bit = 0 menandakan prioritas rendah

Simbol	Posisi	Fungsi
-	IP.7	Kosong
-	IP.6	Kosong
-	IP.5	Kosong
PS	IP.4	Bit prioritas interupsi port serial
PT1	IP.3	Bit prioritas interupsi Timer 1
PX1	IP.2	Bit prioritas interupsi $\overline{INT1}$
PT0	IP.1	Bit prioritas interupsi Timer 0
PX0	IP.0	Bit prioritas interupsi

Tabel 2.4 Alamat Sumber Intrupsi[2]

Nama	Lokasi	Alat Interupsi
Reset	0000H	Power on reset
/Int 0	0003H	INT0
Timer 0	000BH	Timer 0
/Int 1	0013H	INT 1
Timer 1	001BH	Timer 1
Sint	0023H	Port I/O serial

2.8.7. SFR (Special Function Register)

Register Fungsi Khusus (*Special Function Register*) terletak pada 128 byte bagian atas memory data internal dan berisi *register-register* untuk pelayanan *latch port, timer, program status words, control peripheral* dan sebagainya. Alamat register fungsi khusus ditunjukkan pada tabel 2.5.

Tabel 2.5 Special Function Register [2]

SIMBOL	NAMA REGISTER	ALAMAT
ACC	Accumulator	E0H
B	Register B	F0H
PSW	Program Status Word	D0H

SP	Stack Pointer	81H
DPTR	Data Pointer 2 Byte	
DP0L	Data pointer Register 0 Bit Rendah	82H
DP0H	Data Pointer Register 0 Bit Tinggi	83H
DP1L	Data Pointer Register 1 Bit Rendah	84H
DP1H	Data Pointer Register 1 Bit Tinggi	85H
P0	Port 0	80H
P1	Port 1	90H
P2	Port 2	A0H
P3	Port 3	B0H
IP	Interrupt Periority Control	B8H
IE	Interrupt Enable Control	A8H
TMOD	Time/Counter mode Control	89H
TCON	Timer/Counter Control	88H
TH0	Timer/Counter 0 High Control	8CH
TL0	Timer/Counter 0 Low Control	8AH
TH1	Timer/Counter 1 High Control	8DH
TL1	Serial Control 1 Low Control	8BH
SCON	Serial Control	98H
SBUF	Serial Data Buffer	99H
PCON	Power Control	87H
AUXR	Auxiliary Register 0	87H

AUXR1	Auxiliary Register 1	A2H
WDTRST	Watchdog Timer Reset	AEH

Beberapa macam register fungsi khusus yang sering digunakan sebagai berikut ini :

- @ *Accumulator (ACC)* merupakan register untuk penambahan dan pengurangan. Perintah mnemonic untuk mengakses akumulolator disederhanakan sebagai A.
- @ *Register B* merupakan register khusus yang berfungsi melayani operasi perkalian dan pembagian.
- @ *Register R* merupakan delapan set *register* yang dinamakan R0, R1, R2, R3, R4, R5, R6, dan R7, fungsi dari register-register ini adalah merupakan register serbaguna yang membantu menyimpan data yang menggunakan banyak operasi. Lokasinya pada awal 32 byte RAM internal yang memiliki alamat dari 00H sampai 1FH.
- @ *Stack pointer (SP)* merupakan register 8 bit yang dapat diletakan di alamat manapun pada RAM *internal*.
- @ *Data Pointer (DPTR)* terdiri dari empat *register*, yaitu *data pointer 0 rendah (DP0L)*, pada alamat 82H, *data pointer 0 tinggi (DP0H)* pada alamat 83H, *data pointer 1 rendah (DP1L)* pada alamat 84H, dan *data pointer 1 tinggi (DP1H)* pada alamat 85H.

yang berfungsi untuk mengakses source code ataupun data yang terletak dimemory eksternal.

@ *Port 0* sampai *Port 3* merupakan register yang berfungsi untuk membaca dan mengeluarkan data pada port 0, 1, 2, 3. Masing-masing register ini dapat dialamati per-bit.

@ *Control Register* terdiri dari register yang mempunyai fungsi kontrol. Untuk mengontrol sistem interupsi, terdapat dua register khusus, yaitu *register IP* (Interupt Priority) dan *register IE* (Interupt Enable). Untuk mengontrol pelayanan timer/counter terdapat register khusus, yaitu register *TCON* (timer/counter control) serta pelayanan port serial menggunakan register *SCON* (Serial Port Control).

2.8.8. Timer / Counter

Pada Mikrokontroller AT89S51 terdapat dua buah *timer / counter 16 bit* yakni TL0 pada alamat 8AH, TH0 pada alamat 8CH, TL1 pada alamat 8BH, dan TH1 pada alamat 8DH. Yang dapat diatur melalui perangkat lunak, yaitu *timer / counter 1*, Periode waktu timer / counter secara umum ditentukan oleh persamaan berikut :

@ Sebagai timer / counter 8 bit

$$T = 9255 - TLx) * us$$

Dimana ;

TLx adalah isi register TL0 atau TL1

@ Sebagai timer / counter 16 bit

$$T = (65535 - THx - TLx) * 1 \mu s$$

THx = isi register TH0 atau TH1

TLx = isi register TL0 atau TL1

Pengontrolan kerja timer/counter adalah *register timer/counter control* (TCON) yang terletak pada alamat 88H. Definisi dari bit-bit timer control adalah sebagai berikut :

MSB				LSB			
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Pengontrolan pemilihan model operasi *timer/counter* adalah register *timer/counter Mode Control* (TMOD) yang terletak pada alamat 89H. Didefinisikan sebagai berikut :

MSB				LSB			
GATE	C/T	M1	M0	GATE	C/T	M1	M0

Keterangan :

@ GATE : Saat TRx dalam TCON diset 1 dan GATE = 1, *Timer / Counter* akan berjalan ketika TRx = 1 (timer dikontrol software)

@ C / T : Pemilih fungsi *timer* atau *counter*. Clear untuk operasi timer dengan masukan dari system clock internal. Set

(1) untuk operasi counter dengan masukan dari pena T0 dan T1.

@ M1 & M0: Bit pemilih *mode* [2]

2.8.9. Port Serial

Mikrokontroller AT89S51 dilengkapi dengan *port serial*. Port serial memungkinkan kita mengirim data dalam format serial.

Port serial dalam mikrokontroller AT89S51 memiliki sifat *full duplex*, yang berarti dapat mengirim dan menerima data secara bersamaan. Register penerima dan pengirim pada port serial diakses pada SBUF (*Serial buffer/Penyagga*) yaitu pada alamat 99H. Register pengontrol kerja port serial ini adalah SCON (*Serial Control*) yang terletak pada alamat 98H. Bit – bit SCON ini didefinisikan sebagai berikut :

MSB				LSB			
SM0	SM1	SM2	REN	TB8	SB8	T1	R1

2.8.10. Program Status Word

Program Status Word (PSW) terletak pada alamat D0H yang berisi beberapa bit status yang mencerminkan keadaan mikrokontroller. Definisi dari bit-bit dalam PSW yang dijelaskan dibawah ini.

CY	AC	F0	RS1	RS+0	OV	-	P
----	----	----	-----	------	----	---	---

Keterangan :

@ P (*bit paritas*)

Bit paritas akan berada pada kondisi *set* jika jumlah bit 1 dalam acumolator adalah *ganjil* dan akan berada pada *kondisi clear* jika jumlah bit 1 dalam acumolator adalah *genap*. Misalnya data yang tersimpan pada acumolator adalah 10101110b atau AEH maka *parity bit* akan berda dalam kondisi *set*. Data AEH mempunyai lima bit yang berkondisi 1 atau dapat disebut mempunyai bit 1 dalam jumlah yang ganjil.

@ 0 V (*Flog Overflow*)

Akan berada pada kondisi *set* jika pada operasi aritmatik menghasilkan bilangan yang lebih besar dari pada 128 atau lebih kecil dari -128.

@ RS0 dan RS1 (*Register Selektor*)

Bit pemilih *bank register (registr bank select bits)* RS0 dan RS1 digunakan untuk menentukan lokasi dari *bank register (R0-R7)* pada memori. RS0 dan RS1 selalu bernilai 0 setiap kali system ini di reset sehingga lokasih dari *register R0 hingga R7* akan berada pada alamat 00H hingga 07H.

@ F 0 (*Flag 0*)

Flag 0 Dapat digunakan untuk tujuan umum tergantung pada kebutuhan pemakaian.

@ AC (*flag Auxiliary carry*)

Ini akan selalu dalam kondisi *set* apabila pada saat proses penjumlahan terjadi *carry* dari bit ketiga hingga bit ke empat.

@ CY (*flag Carry*)

Ini terletak pada alamat D7H, berfungsi sebagai pendeteksi terjadinya kelebihan pada operasi penjumlahan, atau terjadi peminjaman (*borrow*) pada operasi pengurangan. Misalnya, jika data pada accumulator adalah FFH dan dijumlahkan dengan bilangan satu atau lebih, maka akan terjadi kelebihan sehingga akan membuat *carry* menjadi *set*. Demikian juga apabila data pada accumulator adalah 00H dan dikurangkan dengan bilangan satu atau lebih, akan terjadi peminjaman sehingga membuat *carry* juga menjadi *set*.

2.8.11. Power Control Register

Definisi bit dalam power control *register* (PCON) yang terletak pada lokasi 87H dan dijelaskan berikut ini. Perlu diingat bahwa register PCON ini dapat dialamatkan per bit. Bit-bit dalam *register* PCON adalah :

SMOD	-	-	-	GF1	GF0	PD	IDL
------	---	---	---	-----	-----	----	-----

Keterangan :

- @ SMOD Bila *timer* 1 digunakan untuk menghasilkan boud rate dan SMOD =1, Boud rate dari port serial yang akan

dikalikan dua ketika port serial digunakan dalam mode 1, 2, atau 3.

Tidak dipakai, untuk pengembangan lebih lanjut.

- @ GF1 *Bit flag serbaguna*
- @ GF0 *Bit flag serbaguna*
- @ PD *Bit power down.* Bila bit ini diset (1), mode power down akan aktif. Hanya berlaku untuk tipe CHMOS.
- @ IDL *Idle mode bit.* Bila bit ini di set, akan diperoleh *model idle.*
- @ TCON *Time Counter,* yang berisi bit – bit untuk memulai atau menghentikan pencacahan / pewaktu.
- @ SBUFF *Serial Counter Buffer.* Dugunakan untuk keluaran data dari port.

2.8.12. Metode Pengalamatan

Metode pengalamatan pada AT89S51 adalah sebagai berikut :

A. Pengalamatan tak langsung

Operand pengalamatan tak langsung menunjuk kearah sebuah register yang berisi lokasi alamat memori yang akan digunakan dalam operasi lokasi yang nyata tergantung pada isi register saat instruksi dijalankan.

Untuk melaksanakan pengalamatan tak langsung digunakan simbol @

ADD a,@R0 ; Tambahkan isi RAM yang lokasinya
; ditunjukkan oleh register R0 ke

; akumulator.
DEC @ R1 ; Kurangi satu isi RAM yang alamatnya
 ; ditunjukkan oleh register R1.
 MOVX @ DPTR,A ; Pindahkan isi dari akumulator memory
 ; luar yang lokasinya ditunjukkan oleh
 ; data pointer (DPTR).

B. Pengalamatan Langsung

Pengalamatan langsung dilakukan dengan memberikan nilai ke suatu register secara langsung. Untuk melaksanakan hal tersebut digunakan Tanda #.

MOV A, #01H ; Isi akumulator dengan bilangan 01H
 MOV DPTR, #19ABH ; Isi register DPTR dengan bilangan
 ; 19ABH.

C. Pengalamatan data langsung dari 0 sampai 127 akan mengakses RAM Internal, sedang pengalamatan dari 128 sampai 255 akan mengakses register perangkat keras.

MOV P3, #A ; Pindahkan isi akumulator ke alamat
 ; data B0H (B0H adalah alamat Port 3)
 Inc 50 ; Naikan lokasi 50 (desimal) dalam
 ; memori.

D. Pengalamatan Bit

Pengalamatan bit adalah menunjukkan alamat lokasi bit baik dalam RAM internal (byte 32 sampai 47) atau bit perangkat keras. Untuk melakukan pengalamatan bit digunakan simbol titik (.), misalnya :

FLASG.3,40.5,21H.5 dan ACC.7.

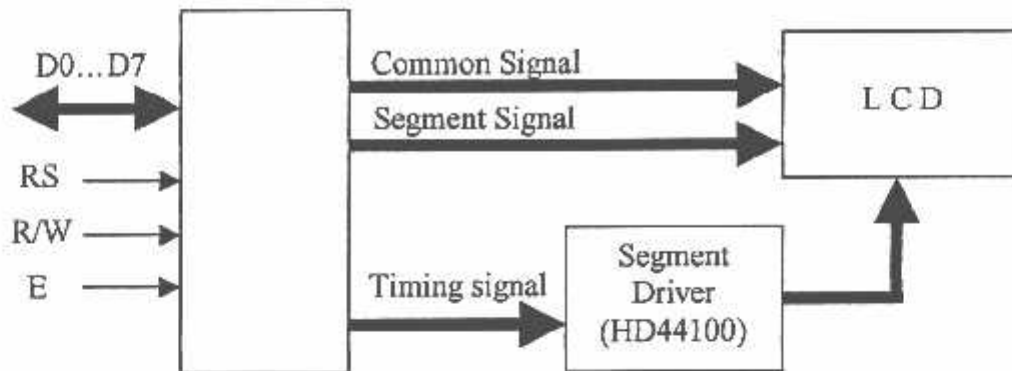
E. Pengalamatan Kode

Ada tiga macam instruksi yang dibutuhkan dalam pengalamatan kode, yaitu *relative jump*, *in-blok jump* atau *call*, dan *jump long* atau *call*. [2]

2.9. Liquid Crystal Display (LCD)

M1632 merupakan modul LCD *matrix* dengan konfigurasi 16 karakter dan 2 baris dengan tiap karakter dibentuk oleh 8 baris pixel dan 5 kolom pixel (*1 baris pixel terakhir adalah kursor*). Yang terbentuk oleh hubungan antara layar LCD dengan HD44780 yang merupakan mikrokontroler kendali LCD. HD44780 ini sudah tersedia dalam modul M1632.

HD44780 sebetulnya merupakan mikrokontroler yang dirancang khusus untuk mengendalikan LCD dan mempunyai kemampuan untuk mengatur proses scanning pada layar LCD yang terbentuk oleh 16 COM dan 40 SEG sehingga mikrokontroler/perangkat yang mengakses modul LCD ini tidak perlu lagi mengatur proses scanning pada layar LCD. Mikrokontroler/perangkat tersebut hanya mengirimkan data-data yang merupakan karakter yang akan ditampilkan pada LCD atau perintah yang mengatur proses tampilan pada LCD saja.



Gambar 2.29 Modul M1632

2.9.1. Beberapa Bagian Pendukung M1632 Dalam Pengoperasiannya

1. Bagian penguat sinyal

Bagian ini berfungsi menguatkan sinyal sebelum sinyal tersebut masuk ke layar LCD dan terdiri dari bagian penguat segment (*segment signal driver*) adalah sinyal yang berasal dari sinyal data 40 bit shift register yang ditahan oleh 40 bit rangkaian latch berfungsi untuk mengatur tampilan kolom pixel layar LCD. Dan bagian penguat sinyal pin bersama (*common signal driver*) yang berasal dari 16 bit shift register untuk mengatur scanning baris pixel dari layar LCD. Sedangkan driver voltage selector berfungsi untuk mengatur penguatan sinyal yang dikirimkan ke layar LCD sehingga kontras dari LCD dapat diatur.

2. Bagian *memory*

Bagian memori terdiri dari 9.920 bit CGROM, 64 byte CGRAM, dan 80 x 8 bit DDRAM yang diatur pengamatan oleh address counter dan akses datanya dilakukan oleh register data.

3. Bagian *register*

Bagian ini terdiri dari register data dan register perintah. Proses akses data ke atau dari register data akan mengakses ke CGRAM , DDRAM atau CGROM, tergantung pada kondisi addres counter sedangkan proses akses data ke atau dari register perintah akan mengakses ke Intruksion decoder yang akan menentukan perintah-perintah yang akan dilakukan oleh LCD.

4. Bagian *antar muka* dengan *mikrokontroller*

Bagian ini berhubungan dengan kaki-kaki modul M1632 yang akan berhubungan langsung dengan mikrokontroler.

2.9.2. Pena-pena LCD M1632

LCD modul M1632 mempunyai 16 pena/penyemat dengan fungsi sebagai berikut:

1. DDRAM

Adalah merupakan memory tempat karakter yang ditampilkan berada. Contohnya, karakter "A" atau 41h yang ditulis pada alamat 00 akan ditampilkan pada baris pertama dan kolom pertama dari LCD. Apabila ditulis di alamat 40h, karakter tersebut akan tampil pada baris kedua kolom pertama dari LDC

2. CGRAM

adalah memory untuk menggambarkan pola sebuah karakter dan bentuk karakter dapat diubah-ubah sesuai keinginan. Akan tetapi isi memory ini akan hilang saat power supply tidak aktif sehingga pola karakter akan hilang.

3. CGROM

adalah memory untuk menggambarkan sebuah karakter dan pola tersebut sudah ditentukan secara permanen dari HD44780 sehingga tidak dapat diubah lagi. Oleh karena ROM bersifat permanen, pola karakter tersebut tidak akan hilang walaupun power supply tidak aktif. misalnya saat HD44780 akan menampilkan data 41h yang tersimpan pada DDRAM, akan mengambil data di alamat 41h (0100 0001) yang ada pada CGROM, yaitu pola karakter A.

2.9.4. Register-register LCD

HD44780 yang terdapat pada modul M1632 mempunyai dua buah register yang akscsnyan diatur dengan kaki RS. Saat RS berlogika 0, register

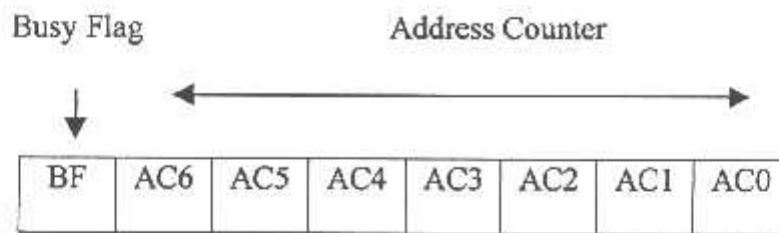
yang diakses adalah Register Perintah dan saat RS berlogika 1, register yang diakses adalah register data.

1. Register *Data*

adalah register dimana mikrokontroler dapat menuliskan atau membaca data ke atau dari DDRAM maupun CGRAM. Akses data ke DDRAM, baik penulisan maupun pembacaan, merupakan akses ke bagian memory tampilan pada layar LCD, sedangkan akses ke CGRAM merupakan proses untuk mengedit pola karakter yang ada pada lokasi CGRAM tersebut.

2. Register *Perintah*

adalah register dimana perintah-perintah mikrokontroler HD44780 selaku pengendali modul M1632 diberikan. Perintah-perintah tersebut berfungsi untuk mengatur tampilan pada LCD atau alamat dari DDRAM dan CGRAM. Selain itu, register ini juga merupakan tempat dimana status HD44780 dapat dibaca. Bit ke-7 data status yang terbaca adalah *busy flag* (tanda sibuk), yaitu tanda yang mengindikasikan bahwa HD44780 masih dalam kondisi sibuk sehingga proses akses data lebih lanjut dari mikrokontroler yang terhubung pada modul M1632 harus menunggu hingga tanda sibuk ini selesai. Bit ke-6 hingga bit ke-0 adalah *address Counter (Penghitung Alamat)* dari DDRAM. Address counter ini menunjukkan lokasi dari DDRAM yang sedang ditunjukkan saat itu.



Gambar 2.30 Susunan Data Status HD44780

2.9.5. Perintah-perintah LCD M1632

Untuk mengatur tampilan pada layar LCD, alamat DDRAM atau CGRAM mikrokontroler yang terhubung dengan modul M1632 harus mengirimkan data-data tertentu ke register perintah sesuai tabel 2.5 (perintah-perintah M1632).

Tabel 2.7 Perintah-perintah M1632

Perintah	D7	D6	D5	D4	D3	D2	D1	D0	Deskripsi
Hapus Display	0	0	0	0	0	0	0	1	Hapus Display dan DDRAM
Posisi Awal	0	0	0	0	0	0	1	X	Set Alamat DDRAM di 0
Set Mode	0	0	0	0	0	1	I/D	S	Atur arah Pergeseran

									Kursor dan Display
Display On/Off	0	0	0	0	1	D	C	B	Atur display (D) On/Off, kursor (C) On/Off, Blinking (B)
Geser kursor display	0	0	0	1	S/C	R/L	X	X	Geser kursor atau display tanpa mengubah alamat DDRAM
Set Fungsi	0	0	1	DL	N	F	X	X	Atur panjang Data, jumlah baris yang tampil, dan font karakter
Set alamat CGRAM	0	1	Acg	Acg	Acg	acg	acg	acg	Data dapat di baca atau di tulis setelah alamat diatur

BAB III

PERENCANAAN ALAT

3.1. PENDAHULUAN

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

Pada perancangan perangkat keras ini, sistem kontrol yang dapat dibuat pada tugas akhir ini adalah sistem yang di gunakan untuk mengendalikan dan mengukur sudut pembelokan dalam derajat. pembelukan akan dideteksi menggunakan sensor Sudut (Potensiometer) yang digerakan oleh motor dc dengan inputan dari keypad dan data dari perubahan diolah oleh mikrokontroller AT89S51 dan ditampilkam pada liquid crystal display (LCD). Sedangkan perangkat lunak digunakan untuk menjalankan system.

3.2. Alat dan Bahan

Dalam perancangan dan pembuatan alat ini diperlukan komponen-komponen sebagai berikut :

1. 1 buah Mikrokontroller AT89S51
2. 1 buah Liquid Crystal Display (LCD) 16 x 2
3. 1 buah Keypad 4 x 4
4. 1 buah Motor dc
5. 2 buah transistor C9013

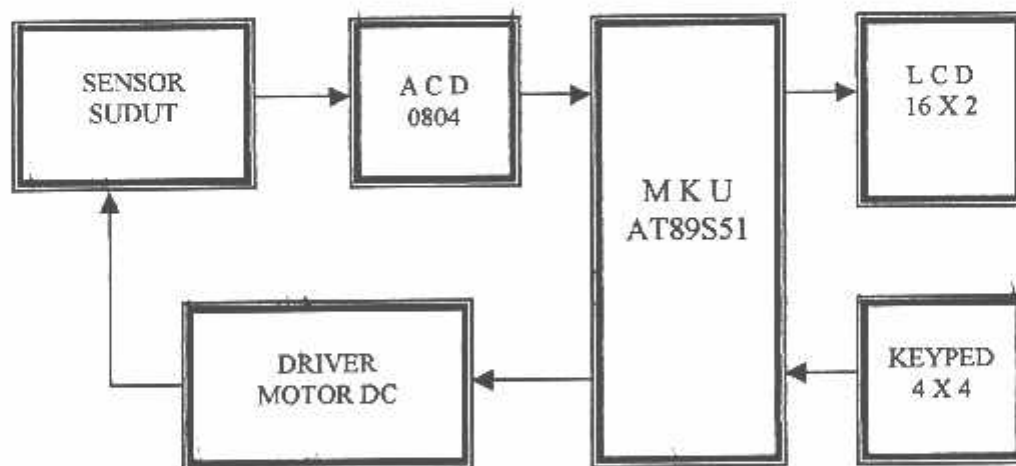
@ Blok rangkaian Minimum Mikrokontroller AT89S51

Digunakan untuk mengolah data yang diterima, mengontrol dan mengendalikan rangkaian-rangkaian yang dihubungkan dengannya.

@ Blok Liquid Crystal Display (LCD) 16 x 2.

Digunakan untuk menampilkan nilai input dan nilai output.

Untuk merancang alat ini perlu dipahami terlebih dahulu digram blok system, perhitungan-perhitungan dan perancangan hardware. Prinsip kerja secara umum dari alat ini dapat dijelaskan dalam blok diagram rangkaian dibawah pada gambar 3.1 :



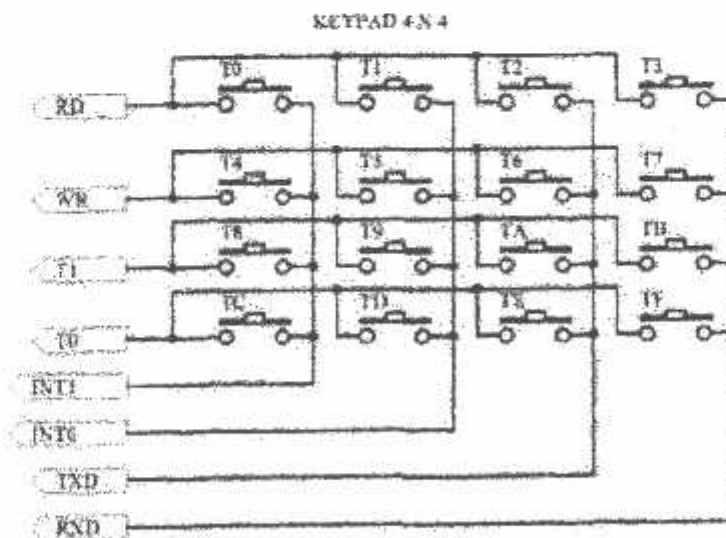
Gambar 3.1 Blok Diagram Rangkaian

3.4. Prinsip Kerja Rangkaian

Prinsip kerja dari rangkaian diatas dapat dijelaskan sebagai berikut. Pada dasarnya inti dari pengambilan data untuk diolah yang seterusnya

- @ Tombol 0 sampai 9 untuk nilai derajat
- @ Tombol C untuk pembelokan ke arah kiri
- @ Tombol D untuk pembelokan ke arah kanan
- @ Tombol E untuk Cancel
- @ Tombol F untuk Enter

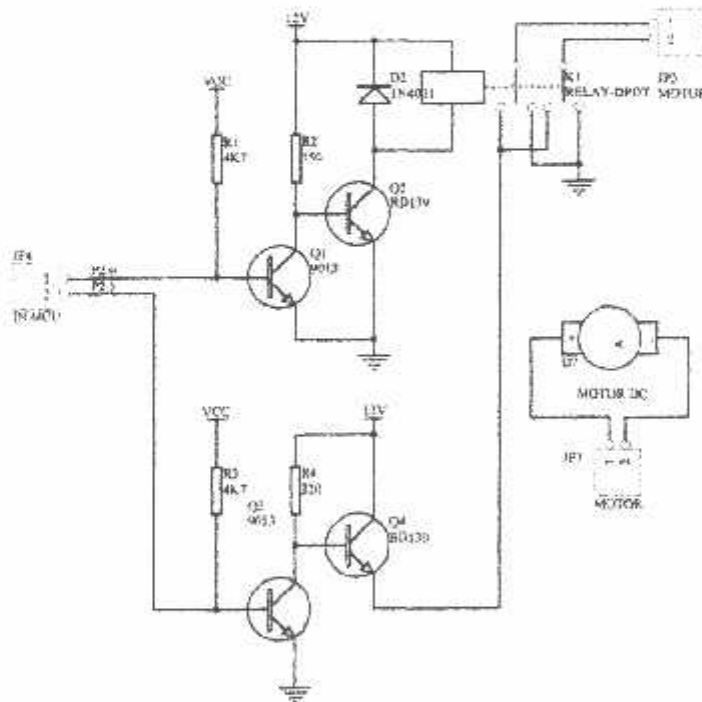
Keypad ini dihubungkan langsung ke mikrokontroller. Pada pin 1, 2, 3 dan 4 sebagai scening, sedangkan pin 5, 6, 7 dan 8 di pergunakan sebagai data hasil penekanan keypad. Kemudian data ditampilkan pada ke LCD pada baris dua. Dan juga mikrokontroller mengoutputkan data ke driver motor dc untuk menggerakkan sensor potensiometer dan kemudi arah, ini disesuaikan dengan nilai inputan yang diinputkan lewat keypad. Adapun gambar rangkaian keypad ditunjukkan pada gambar 3.2 :



Gambar 3.2 Rangkaian Keypad 4 x 4

3.6. Rangkaian Driver Motor DC

Driver motor yang digunakan untuk menggerakan motor dc adalah menggunakan rangkaian driver relay yang dapat menggerakan motor dc bergerak dua arah (maju dan balik) seperti yang sudah dijelaskan pada bab II. Rangkaian relay DPDT yang terhubung pada AT89S51 adalah terlihat pada gambar 3 – 3 rangkaian tersebut selaian relay juga terdiri dari 2 transistor C9013, 2 transistor BD139, 2 tahanan 4K7, 220 ohm,150 ohm dan diode IN4001 serta catu daya 12Vdc



Gambar 3.3 Rangkaian Driver Motor DC

Cara kerja rangkaian driver relay pada gambar diatas menggunakan transistor sebagai saklar statis yaitu untuk memberikan arus pada relay,

transistor C9013 akan aktif bila mendapatkan bias dari basis (arus basis) maka transistor C9013 akan closes sirkuit arus akan mengalir dari kolektor emitor dan keadaan ini disebut *saturasi atau jenuh*. Sebaliknya jika tidak terdapat arus basis maka kolektor emitor akan open sirkuit sehingga tegangan antara keduanya adalah minimum yang disebut sebagai kondisi *cut off atau sumbat*.

Transistor BD139 digunakan sebagai not inputan relay, jika basis mendapatkan trigger maka kolektor emitor akan open sirkuit atau kondisi cut off, begitu pula sebaliknya jika basis tidak mendapatkan trigger maka kolektor emitor akan closes sirkuit atau kondisi satu rasi.

Relay sebagai saklar dinamis yang menghubungkan motor dc dengan tegangan 12Vdc, relay yang digunakan adalah relay mekanik DPDT dengan tegangan 12Vdc. Sinyal penggerak relay ini adalah keluaran dari transistor. Karena relay adalah beban induktif, maka pada saat saklar berguling ke Off, akan masih terdapat tegangan induksi, sehingga pada rangkaian driver relay ini ditambahkan diode IN4001 untuk menghubungkan-singkat tegangan induksi tersebut, sehingga transistor dapat diselamatkan.

3.7. Rangkaian Sensor Sudut

Sensor sudut yang digunakan pada perencanaan alat pengukur sudut kemiringan bidang dalam derajat adalah menggunakan potensiometer. Pada bab II telah dijelaskan bahwa untuk mengetahui sudut kemiringan bidang

sebagai deviasi (dalam prosentasi dari total tahanan) dari tahanan nyata pada beberapa titik tahanan yang di harapkan.

Suatu nilai standar dari linier adalah 0,15 dengan 0,01% bentuk yang terpakai. Dalam tugas akhir ini potensiometer dibangkitkan dengan arus searah yang diambil dari tegangan Vcc 5 Volt DC, dan potensiometer ini memiliki sudut putar sebesar 0° sampai 270° sedangkan derajat yang kita butuhkan hanya sebesar 70° (-35° sampai 0° dan sampai $+35^{\circ}$) yang berarti kita tidak menggunakan potensiometer satu putaran penuh atau 270° .

3.8. Rangkaian ADC (Analog to Digital Converter)

Rangkaian ADC ini berfungsi mengkonversikan besaran analog menjadi besaran digital agar nantinya dapat diolah oleh mikrokontroler sebagai data masukan. Karena keluaran dari potensiometer adalah sinyal analog yang harus diubah menjadi sinyal digital agar dapat diinterfacekan sehingga dapat di baca oleh program komputer.

Untuk rangkaian pengkonversi data analog ke digital (ADC) digunakan IC 0804 buatan nasional semikonduktor. Dengan sebuah masukan yaitu Vin+ (pin6) dan delapan buah keluaran yaitu DB0-DB7 (pin 11-pin 19) pada port1. Bekerja dengan tegangan referensi sebesar 2,5 Volt pada Vref/2 atau pin 9 yaitu didapat dari resistor pembagi tegangan (sesuai dengan data sheed). Fungsi dari rangkaian referensi ini adalah untuk mendapatkan resolusi satu bit yang di inginkan.

ADC ini memiliki range antara 0 sampai 255 dengan menggunakan pencatu daya +5 Volt. ADC 0804 telah dilengkapi dengan internal clock yang di pasang secara running dan dapat diaktifkan dengan menghubungkan komponen resistor eksternal (R) dan komponen kapasitor eksternal (C) pada clock R dan clock IN untuk menghasilkan sinyal clock. Nilai R di tentukan sebesar 10 K Ω dan nilai C sebesar 100 pF, sehingga memberikan frekuensi clock sebesar :

$$F_{clk} = \frac{1}{1,1RC} = \frac{1}{1,1 \times 10K \times 100 pF}$$

$$F_{clk} = \frac{1 \times 10^{12}}{1,1 \times 10 \times 100 \times 10^3} = \frac{1 \times 10^{12} \times 10^{-6}}{1,1 \times 1 \times 1}$$

$$F_{clk} = \frac{1000 \times 10^3}{1,1}$$

$$F_{clk} = 90,909 KHz$$

Dari perhitungan di atas dapat dilihat bahwa harga frekuensi yang di dapatkan sudah memenuhi harga yang di minta ACD 0804 berdasarkan data sheetnya.

ACD di rancang untuk dapat menerima tegangan masukan (V_{in+}) 0Volt sampai 5Volt sehingga satu bit berbobot :

$$\frac{V_{in}}{225} = \frac{5}{225} = 19,6 mV.$$

Sedangkan tegangan Frekuensi (V_{ref}) sama dengan:

$$V_{ref} = \frac{V_{in}}{2} = 2,5 Volt$$

Sesuai dengan data sheet, maka untuk perhitungan setiap satu 1 bit berbobot:

$$\begin{aligned} \text{Resolusi 1 bit} &= \frac{V_{ref}}{255} \text{ banyak data perbit } 2^8 = 255 \\ &= \frac{2,5}{255} = 9,8 \text{ mV atau } = 10 \text{ mV} \end{aligned}$$

Dan untuk perhitungan pada sudut maksimal dari 35 derajat maka keluaran tegangan yang di keluarkan oleh sensor potensiometer adalah :

$$\begin{aligned} P_{out} &= (\text{Resolusi 1 bit}) \times (\text{Sudut maksimal}) \\ &= 10 \text{ mV} \times 35 \\ &= 250 \text{ mV} \end{aligned}$$

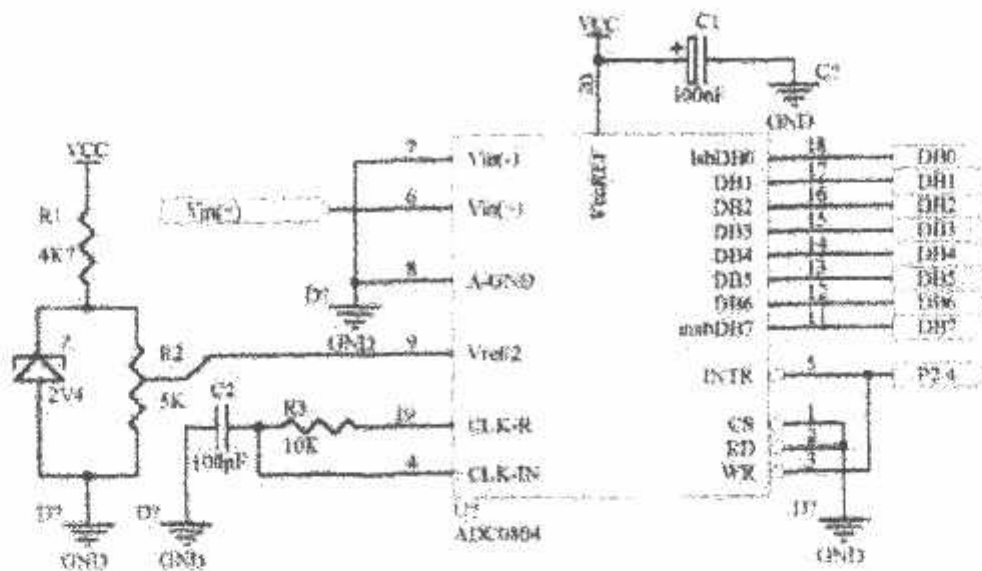
Sehingga output digital bilangan biner ADC adalah :

$$\begin{aligned} \text{ADC Output} &= \frac{V_{outsensor}}{\text{resolusi}} \\ &= \frac{250mV}{10mV} = 35 \text{ D} \end{aligned}$$

maka untuk :

$$\begin{aligned} \text{bilangan biner adalah} &= 100011 \text{ B} \\ \text{bilangan hexsa adalah} &= 23 \text{ H} \end{aligned}$$

Rangkaian ADC 0804 ini dapat dilihat pada gambar berikut ini :



Gambar 3.5 Rangkaian ADC (Analog To Digital Converter)

3.9. Perencanaan Minimum Sistem AT89S51

Mikrokontroler AT89S51 dirancang untuk berdiri sendiri, karena sudah terdapat EPROM, RAM, serta Port I/O internal. Untuk berhubungan dengan peralatan luar dan chip terbagi atas 3 bus, yaitu

@ Data Bus

Yaitu jalur untuk input-output data yang lebarnya sesuai lebar data yang di peroleh oleh mikrokontroler, yaitu 8 bit.

@ Address Bus

Yaitu jalur alamat dari input-output atau memori yang di hubungkan sehingga suatu saat hanya ada satu device yang

berhubungan dengan CPU, lebar address bus mikrokontroler adalah 16 bit (A0-A15).

@ Kontrol Bus

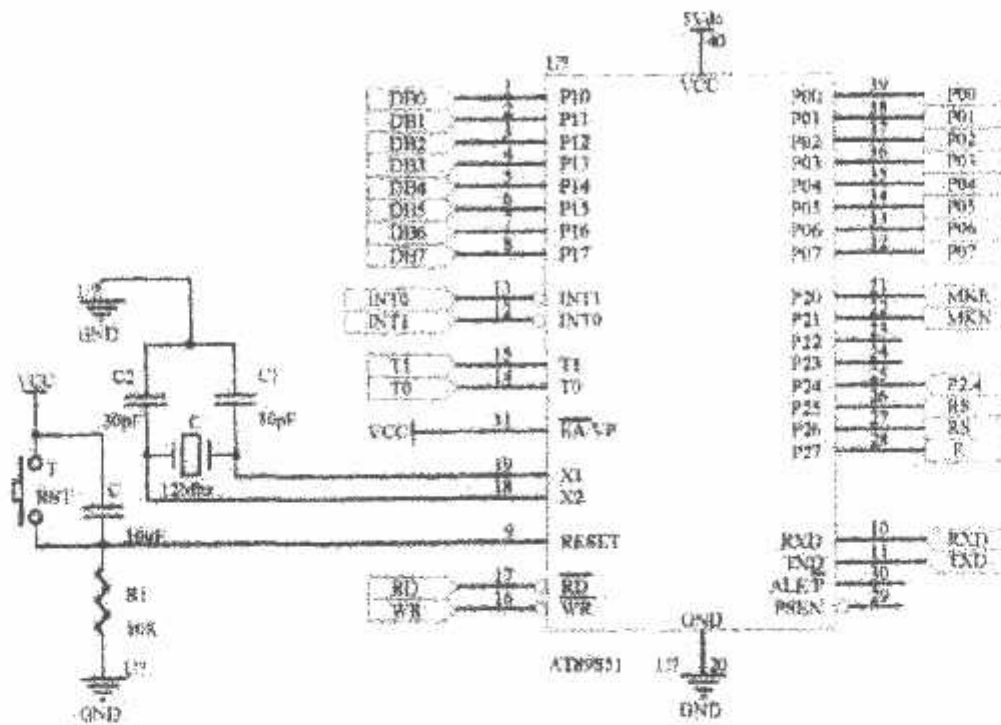
kontrol bus berfungsi untuk mengatur sinkronisasi hubungan antara MCU dengan luar.

Untuk nilai arus yang melalulahi mikrokontroler AT89S51 telah memiliki ketetapan yang sama dengan ADC yakni arus input pada port 1,2,3 saat $V_{in} = 0,45 \text{ V}$ adalah sebesar -50 micro Ampere . Sedangkan arus output maksimum per bit port adalah sebesar :

Port0 = 36 mA

Port 1,2,3 = 15 mA

Total arus maksimum untuk semua keluaran pin adalah sebesar 81 mA dan V_{cc} Minimum untuk power-down adalah 2V.



Gambar 3.6 Perencanaan Rangkaian Mikrokontroler AT89S51

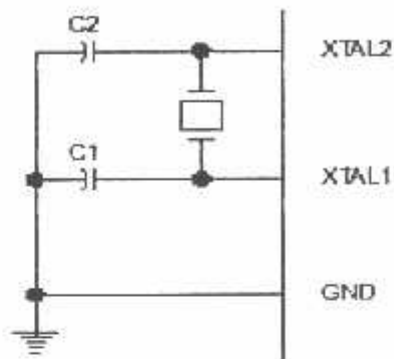
3.9.1. Rangkaian Clock Minimum Sistem

Mikrokontroler AT89S51 ini memiliki internal clock generator yang berfungsi sebagai sumber clock yang di perlukan.

Rangkaian ini terdiri dari dua buah kapasitor dan sebuah kristal, dengan ketentuan :

- C1 dan C2 = 30 pF ± 10 pF untuk kristal
- = 40 pF ± 10 pF untuk kiramik resonators

Dalam minimum sistem ini menggunakan kristal 12Mhz dan C1 = C2 sebesar 30pF. Dengan rangkaian sebagai berikut :



Gambar 3.7 Rangkaian Clock

Untuk rangkaian clock digunakan sebuah kristal 12Mhz dan 2 buah kapasitor 30pF. Dengan menggunakan nilai kristal diatas maka dapat dihitung waktu yang diperlukan untuk satu siklus mesin.

Diketahui : $F = 12\text{Mhz}$

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

Sehingga :

$$T = \frac{1}{12\text{Mhz}} = \frac{1}{12} \mu\text{s}$$

Maka untuk satu siklus mesin dari mikrokontroller AT89S51 adalah sebesar :

$$T_{me} = 12 \times T$$

$$T_{me} = 12 \times \frac{1}{12} \mu\text{s} = 1 \mu\text{s}$$

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

3.9.2. Rangkaian Reset

Rangkaian reset dalam mikrokontroller AT89S51 akan melakukan reset setelah catu daya dihidupkan. Pada saat kondisi reset maka vektor reset pada alamat 0000H akan ditujuh oleh mikrokontroller AT89S51 (dalam hal ini program counter) agar program yang terdapat didalam mikrokontroller kembali ke kondisi semula atau dengan kata lain mikrokontroller mengakses awal dari program yang telah diisi didalamnya. Didalam reset ini akan menggunakan beberapa macam cara untuk mereset miktrokontroller AT89S51. Cara pertama menggunakan Switch (manual), dimana user yang akan mengoperasikan switch ini. Cara ke dua menggunakan kapasitor 10uF, dimana kapasitor tersebut akan berkondisi aktif high selama beberapa detik.

Besarnya nilai tahanan dan kapasitor pada rangkaian reset akan menentukan lamanya waktu pulsa reset.

Dengan rumus :

$$t = R \times C$$

Agar reset dapat terjadi secara normal maka nilai 't' harus jauh lebih besar dari waktu satu kali siklus mesin.

$$t \gg 30 \times T_{me}$$

Karena nilai T_{me} diatas adalah 1uS maka :

$$t \gg 30 \times 1\mu S$$

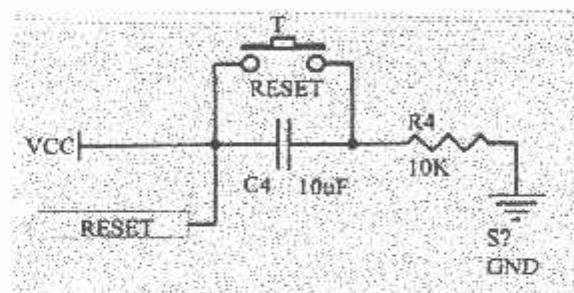
$$t \gg 30\mu S$$

Dengan mengambil nilai R dan C sebesar 10K Ω dan 10uF maka besarnya 't' dapat dicari sebagai berikut :

$$\begin{aligned}
 t &= R \times C \\
 &= 10\text{K}\Omega \times 10\mu\text{F} \\
 &= (10 \times 10^3) \times (10 \times 10^{-6}) = 100 \times 10^{-3} \\
 t &= 0,1 \text{ second}
 \end{aligned}$$

Maka dengan demikian ;

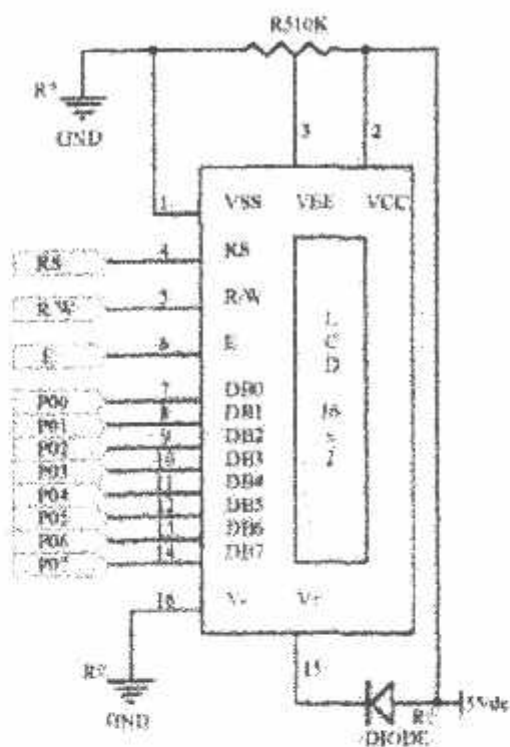
Nilai 't' jauh lebih besar dibandingkan dengan nilai minimalnya.



Gambar 3.8 Rangkaian Reset

3.10. Liquid Cristal Display (LCD)

LCD diperlukan untuk menampilkan nilai karakter input yang akan diproses dan data karakter output dari hasil pengukuran supaya hasil proses dan pengukuran bisa dipahami oleh manusia. LCD M1632 pada tugas akhir ini LCD digunakan hanya sebagai tampilan data karakter input output saja. Hal ini dilakukan dengan cara memberi pada R/W LCD dihubungkan pada R/W Mikrokontroller.



Gambar 3.9 Modul LCD M1632

Untuk nilai arus yang ada pada V_{in} LCD adalah sebesar 0,33mA yang di dapat dari :

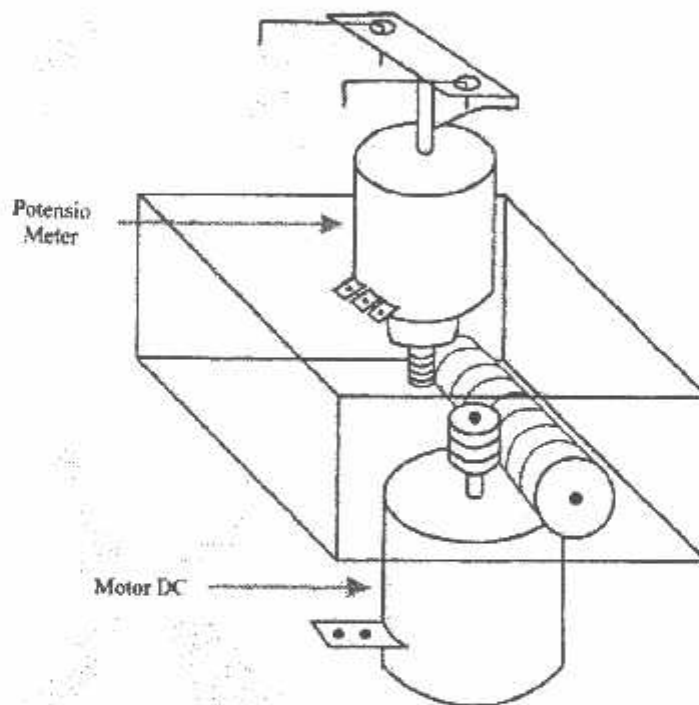
Dik : $V_{in} = 5 \text{ Volt}$

$R = 10 \text{ Kohm}$

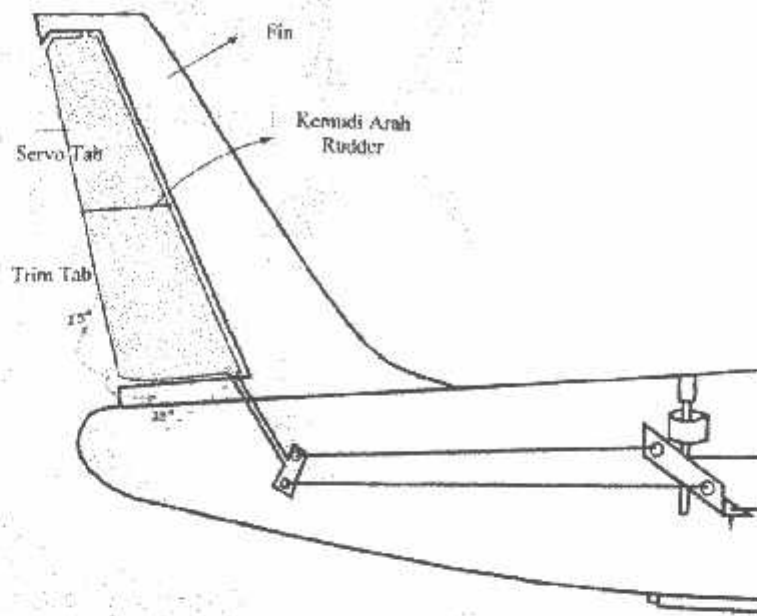
$$\begin{aligned}
 I_{LCD} &= \frac{V_{in}}{R} \\
 &= \frac{5\text{Volt}}{10 \times 10^3 \text{ ohm}} = \frac{5 \times 10^{-3} \text{ v}}{10 \text{ ohm}} \\
 &= 0,5 \text{ mA}
 \end{aligned}$$

3.11. Perancangan Konstruksi Kumudi Arah pada Pesawat Terbang

Perancangan konstruksi kemudi arah pada pesawat terbang dimaksudkan agar penempatan dari sensor potensiometer dan motor dc yang disesuaikan dengan kondisi body dan fin miniatur pesawat terbang, sehingga pergerakan kemudi arah dengan sudut 35° dapat bekerja secara optimal.



Gambar 3.10 Perancangan Konstruksi Sensor dan Motor DC



Gambar 3.11 Perancangan Konstruksi Kemudi Arah

3.12. Perancangan Perangkat Lunak

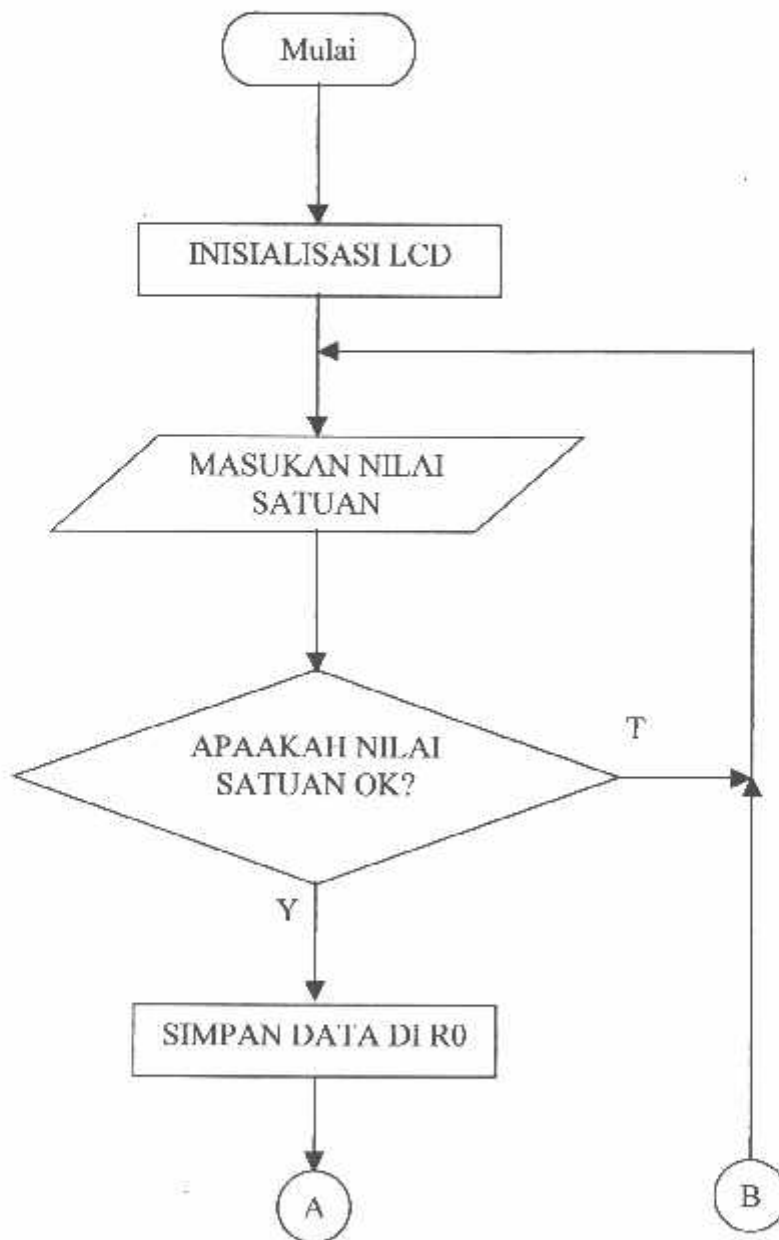
Program adalah kumpulan dari instruksi-instruksi untuk mengendalikan operasi sistem perangkat keras (hardware). Adapun langkah-langkah pembuatan program adalah sebagai berikut :

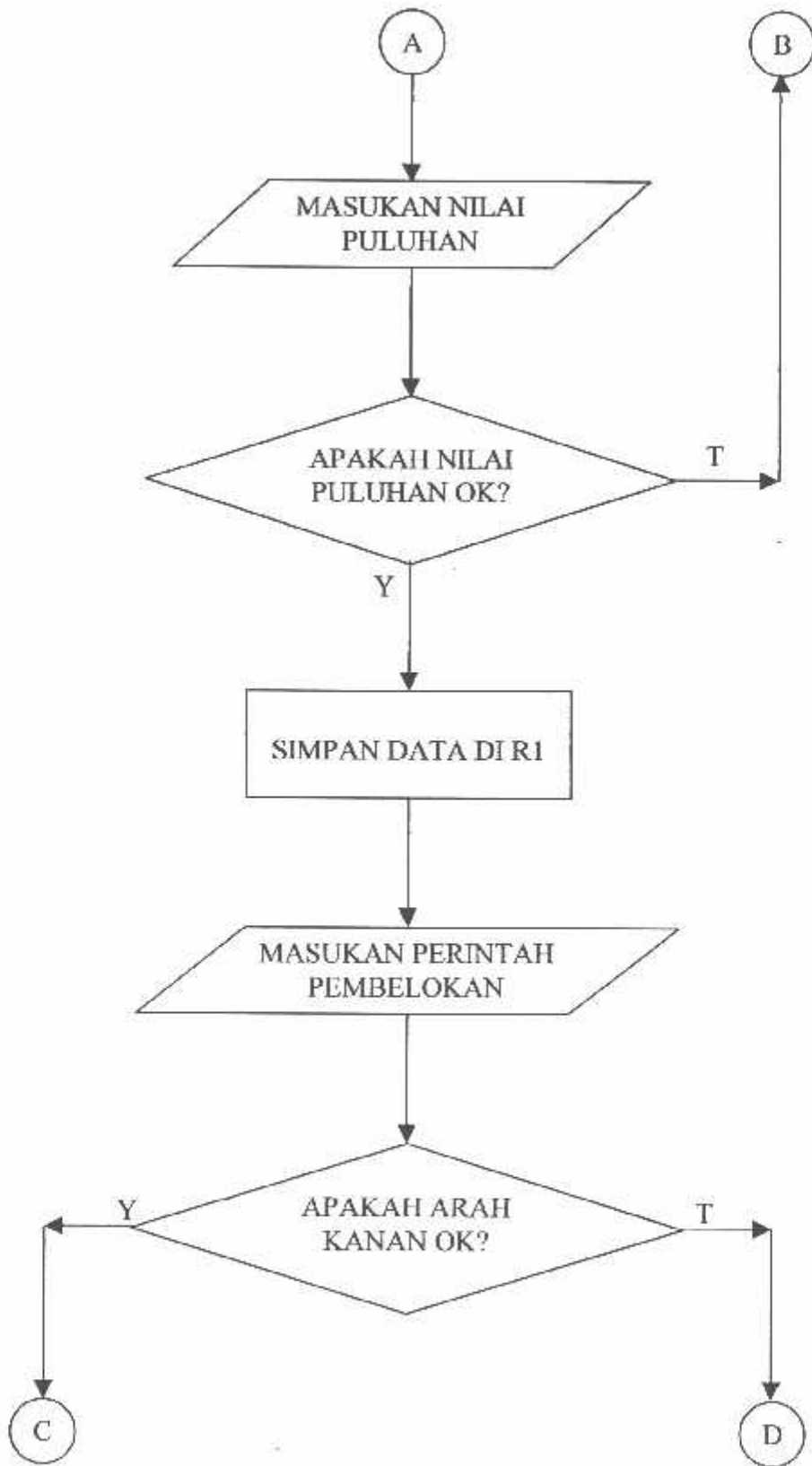
1. Membuat diagram alir (Flow chart) dari program yang akan dibuat.
2. Mengubah diagram alir kedalam bahasa pemrograman Assembly.
3. Mengkompilasi program yang di buat ke memory sampai menghasilkan program yang paling sesuai
4. Memasukan program yang sudah sesuai kedalam mikrokontroller AT89S51 dengan menggunakan EEPROM Programmer

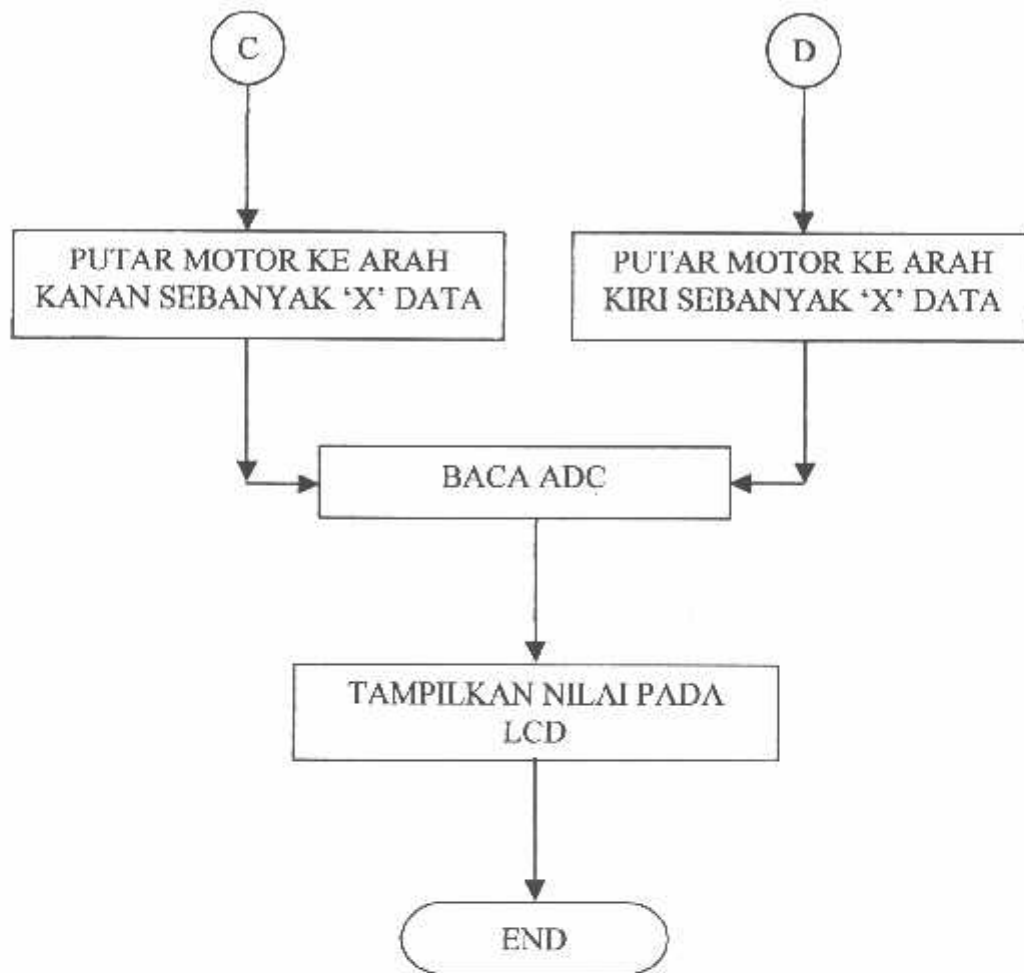
Sistem ini akan bekerja dengan baik jika perencanaan perangkat lunak (software) sesuai dengan perangkat keras (hardware).

Berikut ini merupakan flow chart dari sistem alat kontrol dari kemudi arah pada pesawat terbang yang direncanakan.

Flowchart







BAB IV

PENGUJIAN DAN ANALISA ALAT

4.1. Pendahuluan

Untuk mendapatkan hasil yang maksimal setelah melakukan perencanaan dan pembuatan alat, maka perlu dilakukan suatu pengujian terhadap alat yang dibuat. Pengujian tersebut bertujuan untuk mengetahui apakah system yang telah dibuat dapat bekerja sesuai dengan apa yang diinginkan dalam perancangan.

4.2. Pengujian Perangkat Keras

Dalam pengujian alat dibagi dalam beberapa sub system dari instrument dan peralatan, diantaranya adalah pengujian :

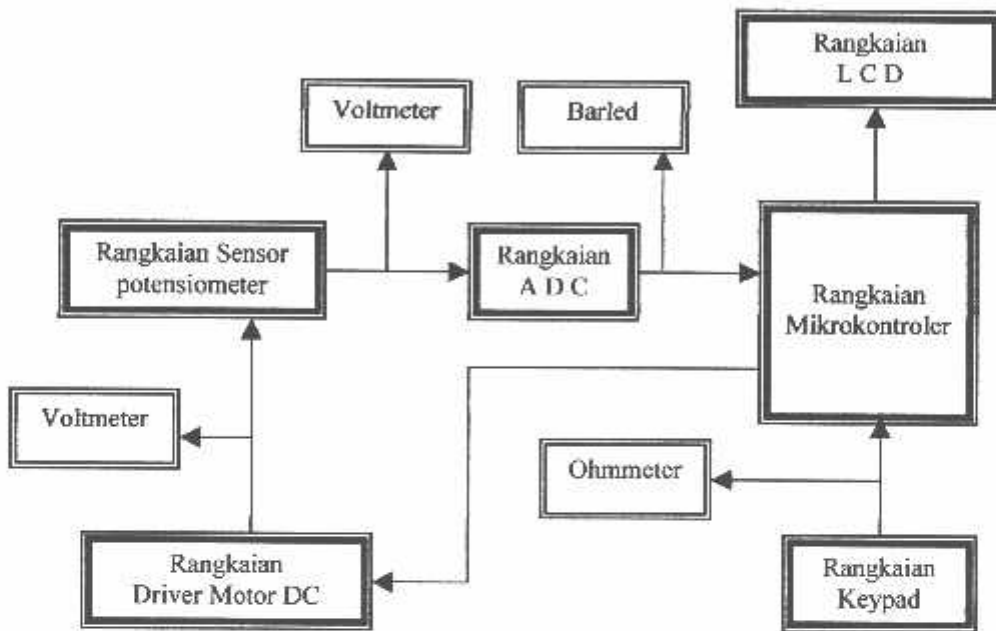
- @ Rangkaian Sensor Sudut (potensiometer)
- @ Rangkaian ADC
- @ Rangkaian Driver Motor DC
- @ Rangkaian Keypad
- @ Rangkaian Keseluruhan

peralatan utama yang dibutuhkan untuk menunjang dalam pengujian alat adalah :

- @ Multimeter digital
- @ Power supply 5 Volt dc dan 12 Volt dc
- @ Modul downlodder AT89S51

@ Mistar busur derajat

Sub system yang diuji diatas dapat digambarkan untuk diagram blok dari pengujian alat secara lengkap adalah :



Gambar 4.1 Diagram Blok Pengujian Alat

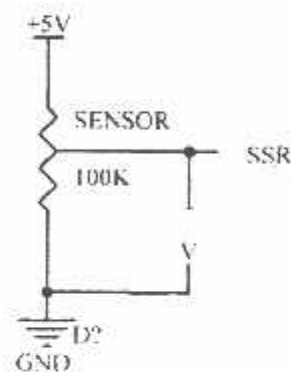
4.2.1. Pengujian Rangkaian Sensor Sudut

Sensor sudut yang digunakan adalah potensiometer, Bagian yang diukur pada potensiometer adalah tegangan output (V_{out}) yang dihasilkannya. Tegangan output selalu berubah – ubah sesuai dengan nilai yang dihasilkan dari perubahan pada nilai tahanan potensiometer.

1. Tujuan

Dari pengujian potensiometer adalah untuk mengetahui tahanan output (V_{out}) dengan sudut yang dibentuk oleh kemudi arah dan linieritas dari potensiometer tersebut.

Peralatan pengujian yang dibutuhkan adalah Voltmeter, catu daya 5Vdc dan 12V dc serta penggaris busur derajat.



Gambar 4.2 Pengujian Rangkaian Sensor Sudut

Tabel 4.1

Data Hasil Pengujian Rangkaian Sensor Sudut arah Kanan

No	Sudut ($^{\circ}$)	Voutp (Volt)
	Pembelokan	Pembelokan Kanan
1	0	3,40
2	1	3,39
3	2	3,38

4	3	3,36
5	4	3,35
6	5	3,345
7	6	3,34
8	7	3,335
9	8	3,33
10	9	3,32
11	10	3,31
12	11	3,30
13	12	3,29
14	13	3,28
15	14	3,27
16	15	3,26
17	16	3,25
18	17	3,24
19	18	3,23
20	19	3,22
21	20	3,21
22	21	3,20
23	22	3,19
24	23	3,18
25	24	3,17

26	25	3,17
27	26	3,16
28	27	3,15
29	28	3,14
30	29	3,13
31	30	3,12
32	31	3,09
33	32	3,07
34	33	3,07
35	34	3,06
36	35	3,05

Tabel 4.4

Data Hasil Pengujian Rangkaian Sensor Sudut Arah Kiri

No	Sudut ($^{\circ}$)	Voutp (volt)
	Pembelokan	Pembelokan Kiri
1	0	3,40
2	1	3,41
3	2	3,42
4	3	3,425
5	4	3,43
6	5	3,43

7	6	2,435
8	7	3,44
9	8	3,445
10	9	3,45
11	10	3,46
12	11	3,47
13	12	3,48
14	13	3,49
15	14	3,52
16	15	3,53
17	16	3,54
18	17	3,56
19	18	3,57
20	19	3,58
21	20	3,60
22	21	3,61
23	22	3,63
24	23	3,65
25	24	3,66
26	25	3,68
27	26	3,71
28	27	3,73

29	28	3,75
30	29	3,76
31	30	3,79
32	31	3,82
33	32	3,84
34	33	3,86
35	34	3,87
36	35	3,89

2. Analisis pengujian dan kesimpulan

- Dari hasil pengujian pada tabel diatas dapat disimpulkan bahwa, setiap perubahan sudut per satu derajat maka sensor potensiometer juga akan ikut bergeser per satu derjad. Namun perubahan nilai tahanan pada potensiometer tidak berubah-ubah secara linier, hal ini mengakibatkan perubahan nilai tegangan output (V_{out}) tidak linier pula.
- Dari data hasil pengujian untuk sudut pembelokan arah kanan, dapat disimpulkan bahwa bila nilai sudut yang diinginkan semakin besar maka tegangan output yang dihasilkan akan semakin kecil. Begitu pula sebaliknya pada sudut pembelokan arah kiri, bila nilai sudut semakin besar maka tegangan output yang dihasilkan akan semakin besar pula.

Tabel 4.2 Data Hasil Pengujian Rangkaian ADC

No	ADCBiner (Kanan)	ADCBiner (Kiri)
1	11001110	11001110
2	11001101	11001111
3	11001100	11011000
4	11001011	11010001
5	11001010	11010010
6	11001001	11010010
7	11001001	11010011
8	11001000	11010100
9	11000111	11010101
10	11000110	11010110
11	11000101	11010110
12	11000100	11010111
13	11000011	11011000
14	11000001	11011001
15	10111111	11011010
16	10111110	11011011
17	10111101	11011011
18	10111100	11011100
19	10111011	11011101

20	10111001	11011110
21	10111000	11011111
22	10111000	11100000
23	10110111	11100010
24	10110110	11100011
25	10110110	11100100
26	10110101	11100101
27	10110011	11100110
28	10110010	11100111
29	10110001	11101001
30	10110000	11101011
31	10110001	11101100
32	10110010	11101101
33	10101111	11101110
34	10101110	11110001
35	10101101	11110010
36	10101100	11110011

2. Analisa pengujian dan kesimpulan

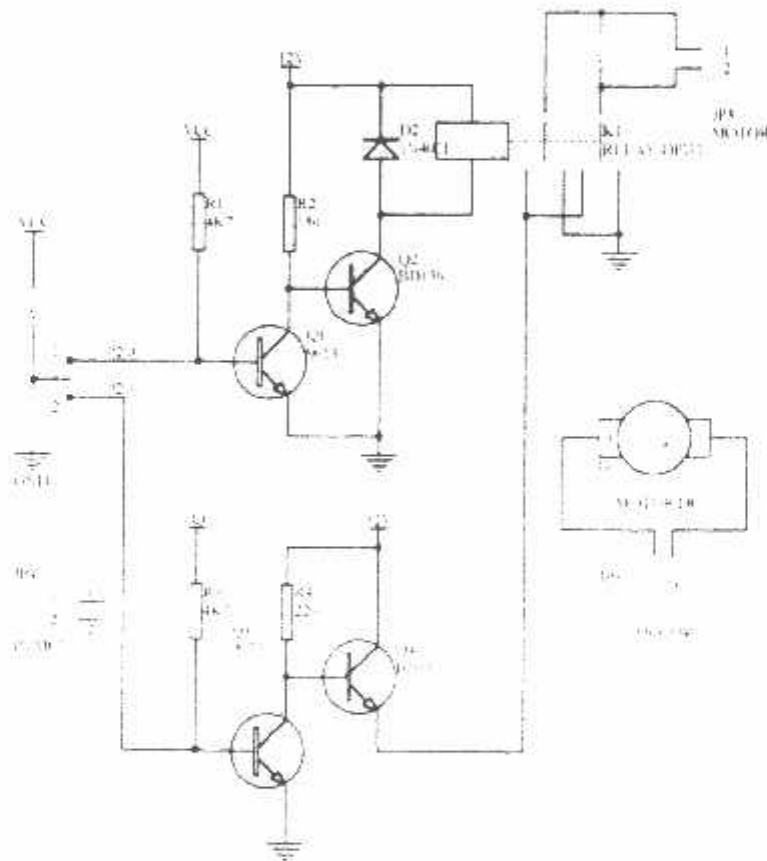
Dari rangkaian diatas didapat analisa sebagai berikut :

ADC 0808 mengkonversikan tegangan analog menjadi kombinasi biner tiap kenaikan (0,01) Volt akan naik satu biner

4.2.3. Pengujian Rangkaian Driver Motor DC

1. Tujuan

Dalam pengujian rangkaian driver motor dc ini, untuk mengetahui nilai tegangan dan membuktikan cara kerja driver motor DC.



Gambar 4.4 Pengujian Rangkaian Driver Motor DC

Tabel 4.3 Data Hasil Pengujian Rangkaian Driver Motor DC

Input		Voutp (Volt)	Kondisi Motor
P2.1	P2.0	Driver Motor	DC
0	0	4,30	Putar Kanan
0	1	0	Mati
1	0	-4,35	Putar Kiri

2. Analisis pengujian dan kesimpulan

Dari hasil pengujian driver motor dc diatas, driver berfungsi sebagai saklar pembalik polaritas, dimana : Transistor BD139 (Q2,Q4) berfungsi sebagai not yaitu jika input basis mendapatkan logika 1 maka Q2 dan Q4 berkondisi cut off atau pada kolektor-emitor tidak ada arus yang mengalir, begitu pula sebaliknya. Jika basis mendapatkan logika 0 maka Q2 dan Q4 akan saturasi atau pada kolektor-emitor terdapat arus yang mengalir.

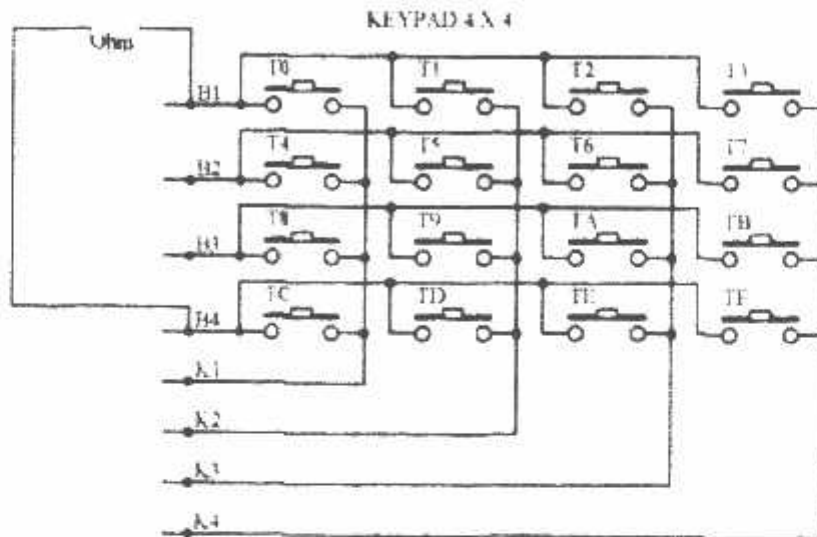
4.2.4. Pengujian Rangkaian Keypad 4x4

1. Tujuan

Pengujian rangkaian keypad dilakukan agar keypad dapat dipastikan bekerja dengan baik sesuai dengan perencanaan. Dalam pelaksanaan pengukuran ini menggunakan multimeter dalam posisi ohm dengan cara menghubungkan seperti pada gambar dibawah ini, juga pada kolom dan baris yang lain.

Peralatan yang dibutuhkan dalam pengujian keypad ini adalah Ohmmeter.

Atau pengujian menggunakan diode led dan batterey 1,5Volt.



Gambar 4.5 Rangkaian Pengujian Keypad 4x4

Tabel 4.4 Hasil Pengujian Keypad 4x4

Tombol	D a t a							
Angka	B1	B2	B3	B4	K1	K2	K3	K4
0	0	1	1	1	0	1	1	1
1	0	1	1	1	1	0	1	1
2	0	1	1	1	1	1	0	1
3	0	1	1	1	1	1	1	0
4	1	0	1	1	0	1	1	1
5	1	0	1	1	1	0	1	1

6	1	0	1	1	1	1	0	1
7	1	0	1	1	1	1	1	0
8	1	1	0	1	0	1	1	1
9	1	1	0	1	1	0	1	1
A	1	1	0	1	1	1	0	1
B	1	1	0	1	1	1	1	0
C	1	1	1	0	0	1	1	1
D	1	1	1	0	1	0	1	1
E	1	1	1	0	1	1	0	1
F	1	1	1	0	1	1	1	0

2. Analisa pengujian dan kesimpulan

Dalam hasil pengujian pada saat tombol angka '1' ditekan yang terhubung adalah antara kolom 1 (K1) dan baris 1 (B1). Untuk lebih jelas dapat dilihat hasil pengujian seperti pada tabel. Nilai '0' pada tabel baris dan kolom menunjukkan bahwa antara kolom dan baris tersebut terhubung sesuai dengan penekanan pada keypad.

4.2.5. Pengujian Rangkaian Keseluruhan

Pengujian rangkaian secara keseluruhan dilakukan dengan menghubungkan masing-masing rangkaian atau blok dan menjalankan perangkat lunak yang dibuat. Pengujian ini dimaksudkan untuk mengetahui apakah peralatan yang dibuat telah sesuai dengan perencanaan. Pengujian

menjalankan dan mengukur sudut pada kemudi arah dengan menggunakan busur derajat,

4.3. Pengujian Perangkat Lunak

4.3.1. Pengujian Program Assembly

Pengujian program assembly ini dilakukan pada saat mengkompile program sumber assembly, yang merupakan kumpulan baris-baris perintah yang telah disimpan dengan *extention* ASM. Program ini dapat ditulis menggunakan *Shoftware text editor* seperti *Notepad* atau *editor* DOS. Pada bagian proses assembly, program. ASM dikompile menjadi dua bagian, yaitu *listing assembly* * .LST dan *program objek* * .HEX yang berisikan kode-kode yang hanya dikenali oleh mikrokontroller. Program inilah yang akan didownload ke mikrokontroller AT89S51.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan dari perancangan dan pembuatan pengendali kemudi arah pada pesawat terbang dengan menggunakan teknologi mikrokontroler, maka dapat diambil kesimpulan sebagai berikut :

- Referensi pesawat terbang yang digunakan adalah pesawat terbang CASA 212 dan OV-10 BRONCO dengan sudut pembelokan maksimal 35 derajat.
- Dari hasil pengujian sensor potensiometer logaritmis dalam pergeseran persatu derajat digunakan cara pendekatan linier, didapatkan selisih nilai output yang sangat kecil dan tidak konstan sehingga ADC tidak dapat menghasilkan nilai output perbit secara presisi.
- Nilai output ADC akan mengikuti perubahan dari nilai resistansi pada sensor potensiometer.
- Rangkaian driver dapat aktif atau saturasi dengan output tegangan tidak akan lebih dari 5 volt Dc karena time/waktu yang di dibutuhkan oleh relay untuk menyaklar sangat pendek dan cepat.
- Dalam pengujian sudut pembelokan pada kemudi arah dari 0^0 sampai 35^0 ini menggunakan busur derajat dengan selisih pergeseran antara kekiri dan kekanan sangatlah kecil.

- Berdasarkan keseluruhan hasil pengujian alat ini maka dapat dikatakan system bekerja dengan baik sesuai dengan apa yang direncanakan.

5.2. Saran

- Alat pengendali ini juga dapat dipakai pada Kendaraan Transportasi lainnya misalnya pada kapal laut.
- Pada pesawat terbang motor yang digunakan sebagai penggerak adalah motor servo. Dan pada alat ini menggunakan motor DC sebagai motor penggerak dengan sudut pembelukan yang dapat diatur dengan meminimal pembelukan per satu derajat.

LAMPIRAN



FORMULIR BIMBINGAN SKRIPSI

Nama : Silvester Yoseph K. T.
Nim : 9817059
Masa Bimbingan : 27 Mar-2004 s/d 27-Sep-2004
Judul Skripsi : Perancangan dan pembuatan alat pengendali (Kontrol) dari kemudi arah pada pesawat terbang dengan menggunakan Mikrokontroller AT89S51

No	Tanggal	Uraian	Paraf Pembimbing
1.	31-05-04	Revisi proposal dan Diagram blok	
2.	25-09-04	Cari Referensi Pesawat terbang	
3.	10-10-04	Revisi Gambar keseluruhan dan tambah skema rangkaian Op Amp	
4.	13-12-04	Revisi Lengkapi Gambar Rangkaian UCN2003 dan Flowchart bab III	
5.	15-03-05	ACC Seminar hasil	
6.	23-07-05	Ref Anatomi Pesawat dan gambar	
7.	24-03-05	Acc Bab IV, V	
8.	28-03-05	Acc Makalah skripsi	
9.			
10.			

Malang, 200
Dosen Pembimbing

Ir. Usman Djuanda, MM



INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No. 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Silvester Yoseph K. T.
Nim : 9817059
Masa Bimbingan : 27-Mar-2004 s/d 27-Sep-2004
Judul Skripsi : Perancangan dan pembuatan alat pendeteksi derajat kemiringan pada pesawat terbang menggunakan Mikrokontroler AT89S51

No	Tanggal	Uraian	Paraf Pembimbing
1.	2 des'04	acc bab I	<i>[Signature]</i>
2.	10 des'04	acc bab II	<i>[Signature]</i>
3.	16 des'04	Revisi OP amp.	<i>[Signature]</i>
4.	18 des'04	flow chart, program.	<i>[Signature]</i>
5.	2 maret'05	acc bab III	<i>[Signature]</i>
6.	3 maret'05	acc makalah seminar	<i>[Signature]</i>
7.	29-03-05	acc bab IV, V	
8.	28-03-05	Acc Makalah Skripsi	
9.			
10.			

Malang, 200
Dosen Pembimbing

M. Ibrahim Ashari, ST




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

LEMBAR BIMBINGAN SKRIPSI

1. Nama : Silvester Yoseph Karolus Tadubun
2. NIM : 98.17.059
3. Jurusan : Teknik Elektro S-1
4. Konsentrasi : Teknik Elektronika
5. Judul Skripsi : Perancangan Dan Pembuatan Alat Pengendali Kemudi Arah Pada Ekor Pesawat Terbang Dengan Menggunakan Mikrokontroller AT89S51.
6. Tanggal Pengujian Skripsi : 29 Maret 2005
7. Selesai Pengujian Skripsi : 29 Maret 2005
8. Pembimbing : 1. Ir. Usman Djuanda, MM
2. M. Ibrahim Ashari, ST
9. Telah Dievaluasi Dengan Nilai : 81 (Delapan Puluh satu) *81*

Diperiksa Dan Disetujui


Dosen Pembimbing I


(Ir. Usman Djuanda, MM)
NIP.P. 1018700143

Dosen Pembimbing II


(M. Ibrahim Ashari, ST)
NIP.P. 1030100358

**Mengetahui,
Ketua Jurusan T. Elektro S-1**


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



Formulir Perbaikan Ujian Skripsi

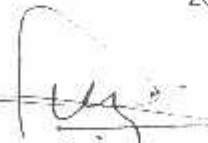
Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : SILVESTER JOSEPH KT
NIM : 9817059
Perbaikan meliputi :

Judul dikembalikan.

Malang,

200


(Silvester J, KT)



Formulir Perbaikan Ujian Skripsi

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

NAMA : SILVESTER YKI
NIM :
Perbaikan meliputi : 9817059

1) FLOW CHART

2) JUDUL

3) TABEL² PENBUJARAN

TERANCIANGAN DAN PEMBUATAN ALAT
PENGENDALI KEYUDI ARAHI PADA
ELKOR PESAWAT TERBANG DENGAN
MENGGUNAKAN MIKROKONTROLER AT89S51

Malang, 29-3-2005



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

FORMOLIR PERBAIKAN SKRIPSI

Nama : Silvester Yoseph Karolus Tadubun
NIM : 98.17.059
Masa Bimbingan : 27 Maret 2004 – 27 Maret 2005
Judul Skripsi : Perancangan Dan Pembuatan Alat Pengendali
Kemudi Arah Pada Ekor Pesawat Terbang
Dengan Menggunakan Mikrokontroller
AT89S51.

No	Tanggal	Materi Perbaikan	Paraf Pembimbing
1	04 April 2005	Judul disesuaikan (perancangan dan pembuatan alat pengendali kemudi arah pada ekor pesawat terbang dengan menggunakan mikrokontroller AT89S51)	
2	04 April 2005	Flow chart	
3	04 April 2005	Tabel-tabel pengujian	

Disetujui :

**(Ir. Sidik Noerjahjono, MT)
Penguji I**

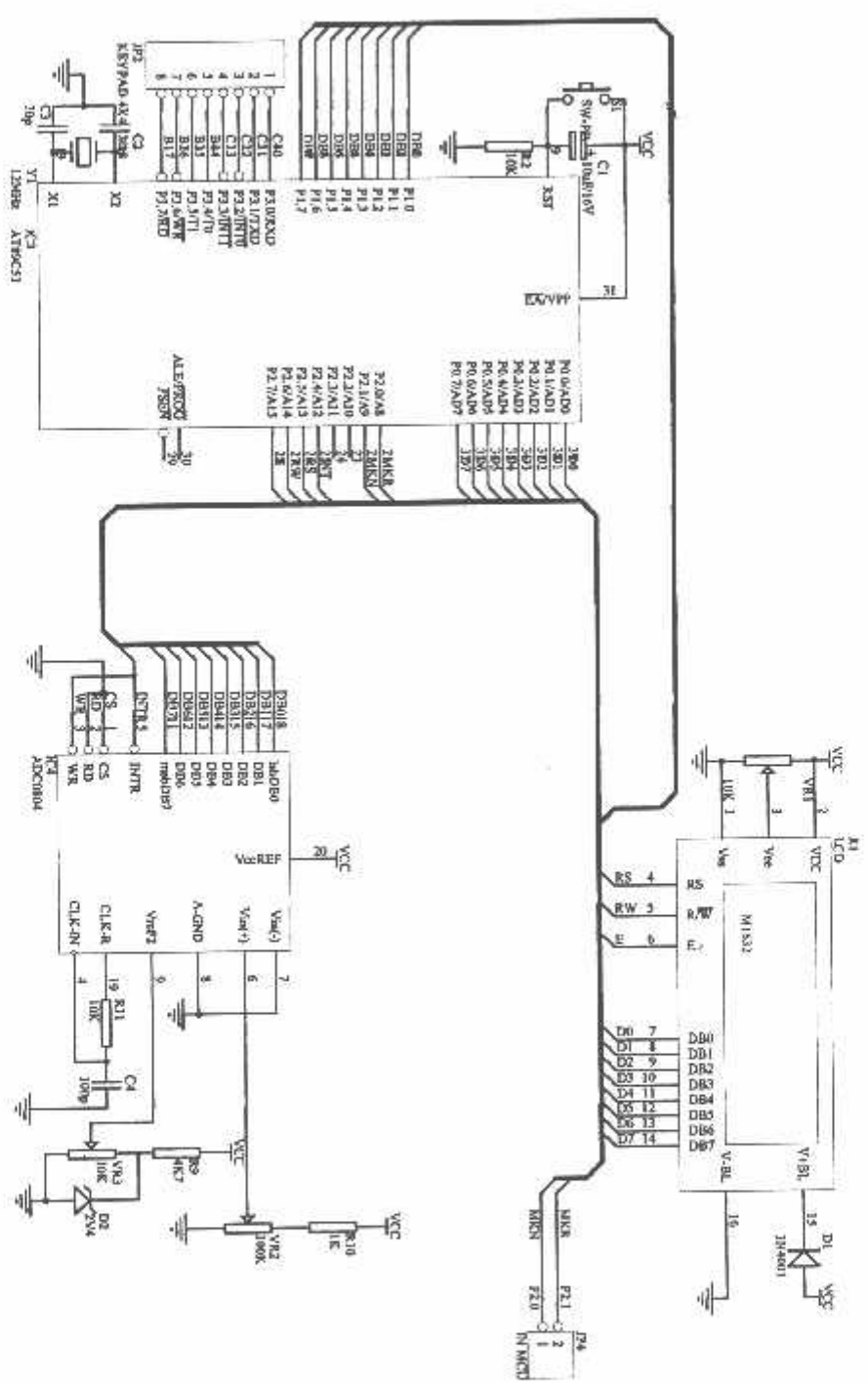
Dosen Pembimbing I

**(Ir. Usman Djuanda, MM)
NIP.P. 1018700143**

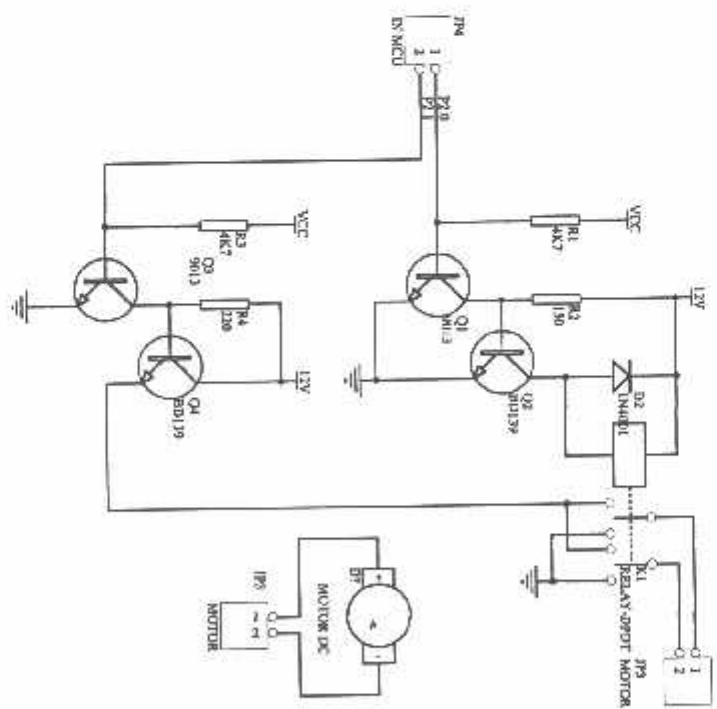
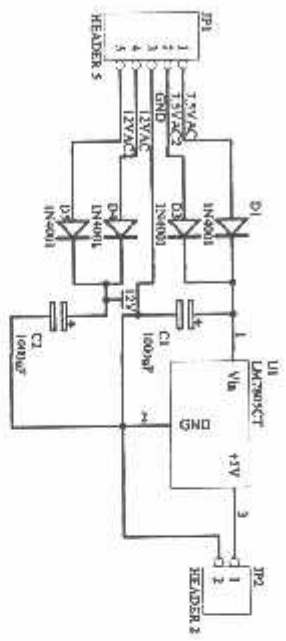
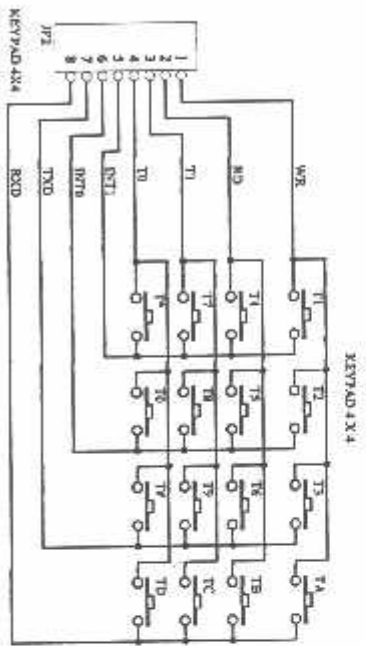
**(Joseph Dedy Irawan, ST, MT)
Penguji II**

Dosen Pembimbing II

**(M. Ibrahim Ashari, ST)
NIP.P. 1030100358**

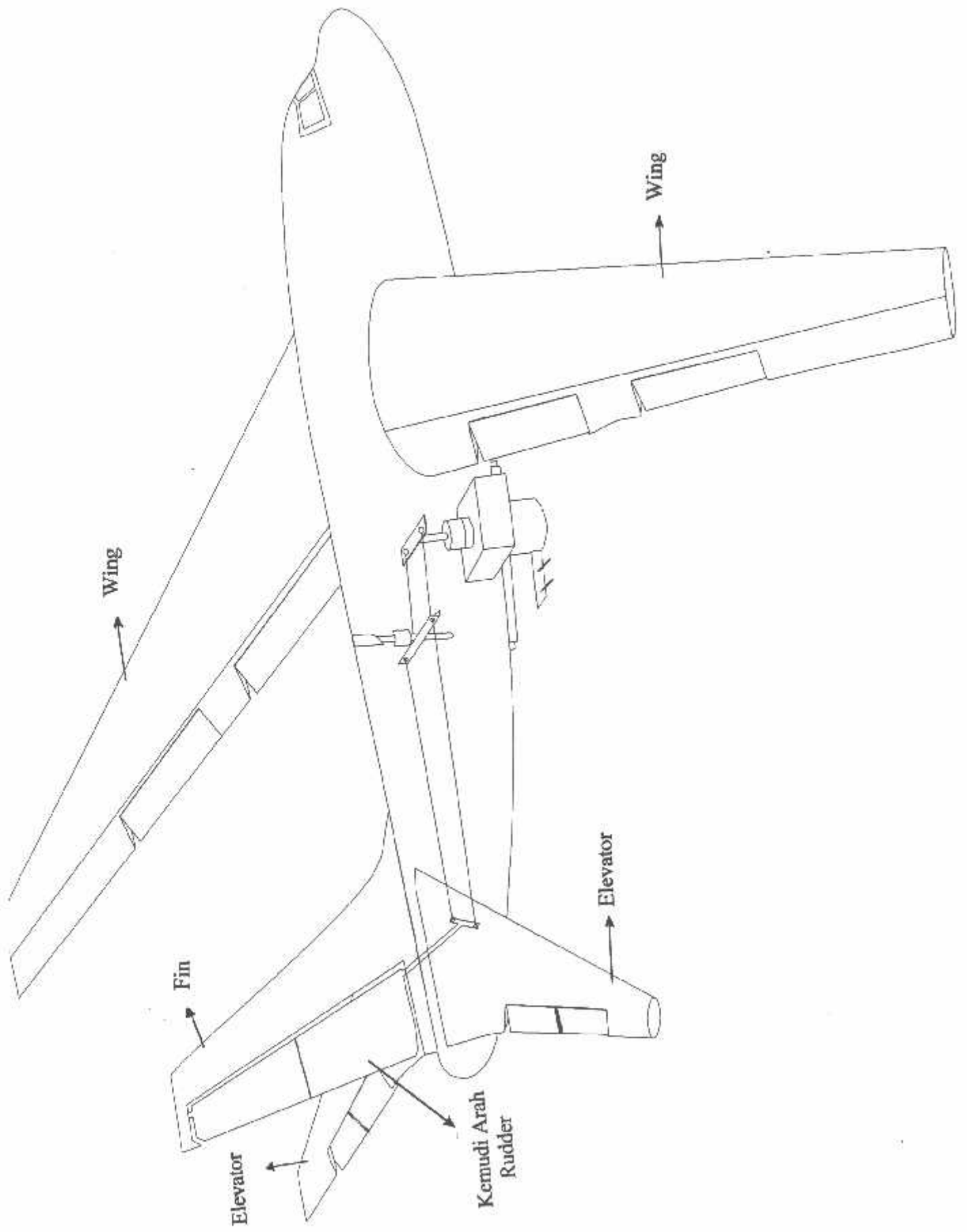


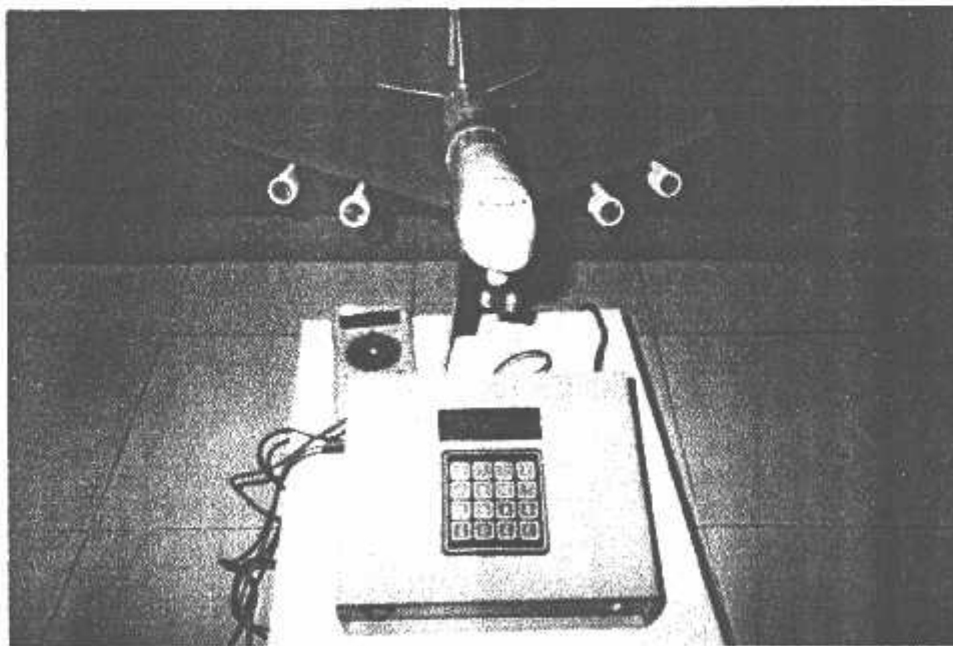
Title: KEMUDI EKOR PESAWAT BERBASIS AT89C84
 Sim: M4
 Number: 1
 Revision: 1
 Date: 5-Eph-2003
 File: C:\epw\src\lib\book\of\pewaw\DOB Sheet of: SIM.YVSTER

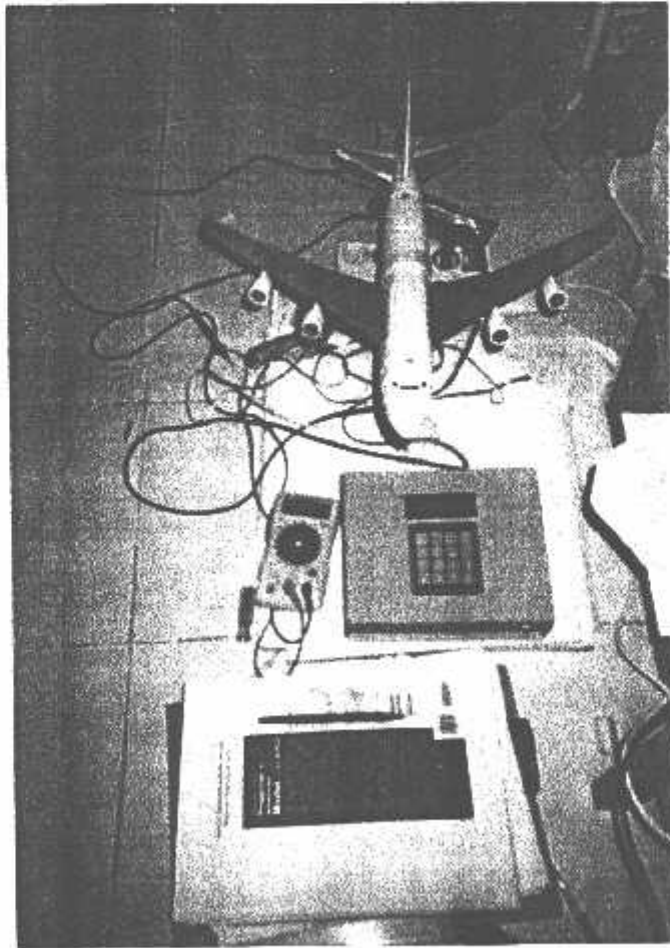
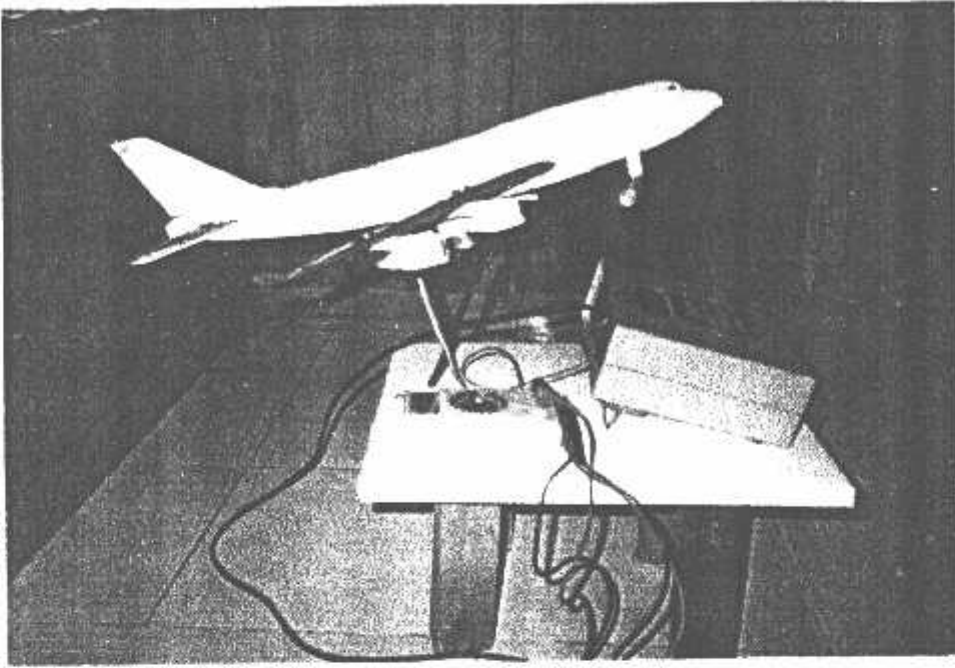


TUR	Number	Revision
AA	3-Relay-10A	Sheet 1 of 1

Date: 3-Relay-10A
 File: C:\3-Relay-10A\3-Relay-10A.dwg
 Drawn By:







```

;*****
;JUDUL: KEMUDI ARAH PESAWAT USE AT89S51
;OLEH : SILVESTER
;NIM  : 98.17.059
;ON MARCH 2005
;*****

;*****
;port definition
;*****
rs          bit    P2.5    ;instruksi/write
data
rw          bit    P2.6    ;read/write
cs          bit    P2.7    ;chip select LCD
adc_int     bit    P2.4    ;adc_intr
wait_me     bit    2fh.0   ;bit konfirmasi OK
m_arah      bit    P2.0    ;Arah motor ke
kiri/kanan
m_active    bit    P2.1    ;start/stop motor
last_arah   equ    50h     ;data arah terakhir
keydata     equ    70h     ;data keydata
digit1      equ    71h     ;puluhan derajat
digit2      equ    72h     ;satuan derajat
digit11     equ    73h     ;ratusan adc
digit22     equ    74h     ;puluhan adc
digit33     equ    75h     ;satuan adc
arah        equ    76h     ;arah kiri/kanan
arah_data   equ    77h     ;data xx step
derajat     equ    78h     ;data derajat
derajat2    equ    79h     ;data derajat
display

```



```

c4          bit    P3.0    ;column 4
c3          bit    P3.1    ;column 3
c2          bit    P3.2    ;column 2
c1          bit    P3.3    ;column 1
b4          bit    P3.4    ;baris 4
b3          bit    P3.5    ;baris 3
b2          bit    P3.6    ;baris 2
b1          bit    P3.7    ;baris 1

```

```

;*****

```

```

;main program here

```

```

;*****

```

```

    org 0

```

```

    call init_LCD

```

```

        setb    m_arah        ;reset arah motor

```

```

        setb    m_active      ;reset motor

```

```

        setb    wait_me       ;kondisi awal bit

```

```

konfirmasi

```

```

        mov     last_arah,#0

```

```

        call    my_id

```

```

        call    detect

```

```

;*****

```

```

;identitas information

```

```

;*****

```

```

my_id:  mov     DPTR,#ID1

```

```

        call    line1

```

```

            call    wr_lcd

```

```

        mov     DPTR,#ID2

```

```

        call    line2

```

```

        call    wr_lcd

```

```

    call dl_id100
    mov  DPTR,#ID3
    call line1
    call wr_lcd
    mov  DPTR,#ID4
    call line2
    call wr_lcd

    call dl_id100
    mov  DPTR,#ID5
    call line1
    call wr_lcd
    mov  DPTR,#ID6
    call line2
    call wr_lcd
    call dl_id100

    mov  DPTR,#MENU
    call line1
    call wr_lcd
        call  inputxx          ;** deg
    ret

;*****
;sub routine detect penekanan tombol
;*****
detect: call  keypad
        mov   A,keydata
        cjne  A,#0FFH,tekan1 ;wait penekanan
tombol

```

```

        sjmp detect
tekan1:

;*****
;sub routine detect puluhan keypad
;*****
dg0:  cjne A,#0,dg1
        call line2
        mov  A,#30H
        call wr_data
        sjmp puluhan
dg1:  cjne A,#1,dg2
        call line2
        mov  A,#31H
        call wr_data
        sjmp puluhan
dg2:  cjne A,#2,dg3
        call line2
        mov  A,#32H
        call wr_data
        sjmp puluhan
dg3:  cjne  A,#3,detect
        call line2
        mov  A,#33H
        call wr_data
puluhan:
        anl  A,#0FH
        mov  digit1,A

```

```

;*****
;sub routine detect satuan keypad
;*****
tekan2:  call keypad

        mov  A,keydata
        cjne A,#0FFH,dg00
        sjmp tekan2
dg00:   cjne A,#0,dg11
        mov  A,#30H
        call wr_data
        jmp  satuan
dg11:   cjne A,#1,dg22
        mov  A,#31H
        call wr_data
        jmp  satuan
dg22:   cjne A,#2,dg33
        mov  A,#32H
        call wr_data
        jmp  satuan
dg33:   cjne A,#3,dg44
        mov  A,#33H
        call wr_data
        jmp  satuan
dg44:   cjne A,#4,dg55
        mov  A,#34H
        call wr_data
        jmp  satuan
dg55:   cjne A,#5,dg66
        mov  A,#35H
        call wr_data

```

```

        jmp satuan
dg66:   cjne A,#6,dg77
        mov  A,#36H
        call wr_data
        jmp  satuan
dg77:   cjne A,#7,dg88
        mov  A,#37H
        call wr_data
        jmp  satuan
dg88:   cjne A,#8,dg99
        mov  A,#38H
        call wr_data
        jmp  satuan
dg99:   cjne  A,#9,dg10
        mov  A,#39H
        call wr_data
        sjmp  satuan
dg10:   cjne  A,#0EH,tekan2
        jmp   ok_batal1

```

```

;*****
;routine write data keypad to lcd
;*****

```

```

satuan:  anl  A,#0FH
        mov  digit2,A
        mov  A,#0DH
        call wr_inst
        mov  A,#0C7H
        call wr_inst
        mov  A,#7FH
        call wr_data

```

```

        mov     A, #'*'
        call   wr_data
        mov     A, #7EH
        call   wr_data
        mov     A, #0C8H
        call   wr_inst
ok2:    lcall   keypad
        mov     A, keydata
        cjne   A, #0FFH, ok_kiri
        sjmp   ok2

;*****
;sub routine tombol kiri
;*****
ok_kiri:
        cjne   A, #0CH, ok_kanan
        clr     wait_me           ;tanda bit OK
        mov     arah, #00H       ;tanda arah kiri
        mov     A, #0C7H
        call   wr_inst
        mov     DPTR, #kiri
        call   wr_lcd
        jmp    ok_proses

;*****
;sub routine tombol kanan
;*****
ok_kanan:
        cjne   A, #0DH, ok_batal2
        clr     wait_me           ;tanda bit OK
        mov     arah, #01H       ;tanda arah kanan

```

```

mov  A,#0C7H
call wr_inst
mov  DPTR,#KANAN
call wr_lcd
jmp  ok_proses

;*****
;sub routine tombol batal
;*****
ok_batal2:
    cjne A,#0EH,ok_proses
ok_batal1:
    setb  wait_me      ;clear bit OK
    mov   arah,#0FFH   ;clear arah
kiri/kanan
    mov   A,last_arah
    cjne  A,#0,ok_batal3
    call  line1
    mov   DPTR,#MENU
    call  wr_lcd
    sjmp  ok_batal4
ok_batal3:
    call  line1
    mov   A,last_arah
    mov   B,#10
    div   AB
    add   A,#30H
    call  wr_data
    mov   A,B
    add   A,#30H
    call  wr_data

```

```

ok_batal4:
    call    inputxx
    jmp     detect

;*****
;sub routine tombol OK
;*****
ok_proses:
    cjne A,#0FH,ok2
    jnb   wait_me,ok2
        setb   wait_me
    mov   A,#0CH
    call  wr_inst
        mov   A,arah
arah0:  cjne  A,#00H,arah1      ;jika A=0 maka kiri
dipilih
        call  xx_step
        jmp   l_step

arah1:  cjne  A,#01H,arah0      ;jika A=1 maka
kanan dipilih
        call  xx_step
        jmp   r_step

;*****
;sub routine ambil data derajat
;*****
xx_step:mov   A,digit1
        mov   B,#10
        mul   AB
        add   A,digit2

```



```

        cjne    A,#0,ok_step
        call   inputxx
        jmp    detect
ok_step:mov    derajat,A
        mov    arah_data,A
        mov    derajat2,#0
        ret

```

```

;*****

```

```

;sub routine keypad

```

```

;*****

```

```

keypad:

```

```

key1:  mov    P3,#0F7H
        jb    B1,key2

```

```

k1:    jnb   B1,$
        mov   keydata,#0
        ret

```

```

key2:  jb    B2,key3

```

```

k2:    jnb   B2,$
        mov   keydata,#4
        ret

```

```

key3:  jb    B3,key4

```

```

k3:    jnb   B3,$
        mov   keydata,#8
        ret

```

```

key4:  jb    B4,key5

```

```

k4:    jnb   B4,$
        mov   keydata,#0CH
        ret

```

```

key5:  mov    P3,#0FBH
        jb    B1,key6

```

```

k5:  jnb  B1,$
      mov  keydata,#1
      ret
key6: jb   B2,key7
k6:  jnb  B2,$
      mov  keydata,#5
      ret
key7: jb   B3,key8
k7:  jnb  B3,$
      mov  keydata,#9
      ret
key8: jb   B4,key9
k8:  jnb  B4,$
      mov  keydata,#0DH
      ret
key9: mov  P3,#0FDH
      jb   B1,key10
k9:  jnb  B1,$
      mov  keydata,#2
      ret
key10: jb  B2,key11
k10: jnb  B2,$
      mov  keydata,#6
      ret
key11: jb  B3,key12
k11: jnb  B3,$
      mov  keydata,#0AH
      ret
key12: jb  B4,key13
k12: jnb  B4,$
      mov  keydata,#0EH

```

```

        ret
key13:  mov     P3,#0FEH
        jb     B1,key14
k13:   jnb    B1,$
        mov     keydata,#3
        ret
key14:  jb     B2,key15
k14:   jnb    B2,$
        mov     keydata,#7
        ret
key15:  jb     B3,key16
k15:   jnb    B3,$
        mov     keydata,#0BH
        ret
key16:  jb     B4,nokey
k16:   jnb    B4,$
        mov     keydata,#0FH
        ret
nokey:  mov     keydata,#0FFH
        ret

;*****
;subroutine initial LCD
;*****
init_LCD:
        nop
        mov     A,#3FH ; clear display n
        call    wr_inst
        mov     A,#01H ; function set 8 bit
        call    wr_inst

```

```

        mov     A,#06H ; entry mode kursor ke kanan
data tctap
        call    wr_inst
        mov     A,#0DH ; display ON, kursor not
display, kursor tidak kedip
        call    wr_inst
        ret

```

```

;*****
;sub routine write data to lcd use DPTR
;*****

```

```

wr_lcd:  clr  A
         movc A,@A+DPTR
         call wr_data
         inc  DPTR
         cjne A,#0,wr_lcd
         ret

```

```

;*****
;sub routine baris 1
;*****
line1:   mov  A,#80H
         call wr_inst
         ret

```

```

;*****
;sub routine baris 2
;*****
line2:   mov  A,#0C0H
         call wr_inst
         ret

```

```

;*****
;sub routine write instruksi to LCD
;*****
wr_inst:
        clr     rw     ;mode write to LCD
        clr     rs     ;untuk menuliskan instruksi
write:
        mov    P0,A
        setb   cs
        clr    cs
        call  dl_lcd
        setb   cs
        ret

;*****
;sub routine write data to LCD
;*****
wr_data:
        clr     rw     ;mode write to LCD
        setb    rs     ;untuk menuliskan data /
character
        jmp    write

;*****
;sub routine delay LCD
;*****
dl_lcd:  mov    R0,#0
dl1:    mov     R1,#20
        djnz  R1,$
        djnz  R0,dl1

```

```

ret

;*****
;sub routine delay ID
;*****
dl_id:   mov  R0,#255
dl2:    mov  R1,#255
        djnz R1,$
        djnz R0,dl2
        ret
dl_id100:
        mov  R2,#20
dl3:    call dl_id
        djnz R2,dl3
        ret

;*****
;sub routine ambil data adc
;*****
adc_data:
        mov  A,#0CH
        call wr_inst
        mov  A,#8DH
        call wr_inst
        call adc_input
        mov  A,#30H
        add  A,digit11
        call wr_data
        mov  A,#30H
        add  A,digit22
        call wr_data

```

```

        mov     A,#30H
        add     A,digit33
        call   wr_data
        ret

adc_input:
        clr     adc_int ; start konversi
        setb    adc_int ; stop konversi
EOC:    jb     ADC_INT,EOC
        mov     A,P1     ; baca data melalui P1
        call   dl_adc
        call   convert
        ret

dl_adc:  mov     R0,#50
dl_adcl:mov     R1,#30
        djnz   R1,$
        djnz   R0,dl_adcl
        ret

;*****
;subroutine konversi nilai Hex adc menjadi ASCII LCD
;*****
Convert:
        mov     R0,A
        mov     B,#100
        div    AB
        mov     digit11,A
        mov     A,B
        mov     B,#10
        div    AB
        mov     digit22,A
        mov     digit33,B

```

```

ret

;*****
;sub routine step ke kiri
;*****
l_step: mov     A,last_arah
        cjne   A,#0,l_steppp ;jika Last_arah<>0
decrement
        clr    2FH.7         ;jika last_arah=0
increment

        sjmp  l_stepp
l_steppp:
        jb    2FH.6,ppp
        setb  2FH.7
        sjmp  l_stepp
ppp:    clr   2FH.7
l_stepp:
        nop
        call  adc_data
        call  degree
sss0:   jb    2FH.7,sss
call    max
        call  r_posisi
        mov   A,#87H
        call  wr_inst
        mov   A,#7FH
        call  wr_data
        setb  2FH.6
        sjmp  sss2
sss:    call  l_posisi

```



```

        mov     A,last_arah
        cjne   A,#0,sss2
        clr    2FH.7
        mov    last_arah,#0FFH
        mov    A,#87H
        call   wr_inst
        mov    A,#7FH
        call   wr_data
        sjmp   sss0
sss2:
        nop
        setb   m_arah           ;arah kiri
        call   dl_mtr
        clr    m_active         ;putar motor
        call   dl_step
        setb   m_active         ;stop motor
        djnz   arah_data,l_stepp
        setb   m_active         ;motor off
        setb   m_arah           ;arah off
        call   input0
        call   dl_id100
        call   inputxx
        jmp    detect

;*****
;sub routine step ke kanan
;*****
r_step:
        nop
        call   adc_data
        call   degree

```

```

        mov     A,last_arah
        cjne   A,#0,oo00
        setb   2FH.7
oo00:   jb     2FH.7,oo11
        call   l_posisi
        clr   2FH.7
        sjmp  oo22

oo11:   call   max
        call   r_posisi
        clr   2FH.6
        mov   A,#87H
        call   wr_inst
        mov   A,#7EH
        call   wr_data
        sjmp  oo33

oo22:   mov   A,last_arah
        cjne   A,#0,oo33
        setb   2FH.7
        setb   2FH.6
        mov   last_arah,#0FFH
        mov   A,#87H
        call   wr_inst
        mov   A,#7EH
        call   wr_data
        sjmp  oo00

oo33:   nop
        clr   m_arah           ;arah kanan
        call  dl_mtr
        clr   m_active        ;putar motor
        call  dl_step

```

```

        setb    m_active        ;stop motor
        djnz   arah_data,r_step
        setb    m_active
setb    m_arah
        call   input0
        call   dl_id100
        call   inputxx
        jmp    detect

input0: mov     A,last_arah
        cjne   A,#0,inputxx
        mov    DPTR,#MENU
        call   line1
        call   wr_lcd

inputxx: ret
;*****
;sub routine untuk menulis ** deg line 2
;*****
inputxx:mov    DPTR,#MENU2
        call   line2
        call   wr_lcd
        mov    A,#0DH
        call   wr_inst
        call   line2
        ret
;*****
;sub routine jika sudut melebihi batas max 35 degree
;*****
max:    mov     A,last_arah
        cjne   A,#35,ok_right
        call   line2

```

```

        mov     DPTR,#MAXIMAL
        call   wr_lcd
        call   dl_id100
        call   inputxx
        jmp    detect
ok_right:
        ret
r_posisi:
        inc     last_arah
        call   posisi_ok
        ret
l_posisi:
        dec     last_arah
        call   posisi_ok
        ret
posisi_ok:
        lcall  line1
        mov    A,last_arah
        mov    B,#10
        div   AB
        add   A,#30H
        call  wr_data
        mov   A,B
        add   A,#30H
        call  wr_data
        mov   A,#' '
        call  wr_data
        mov   A,'#d'
        call  wr_data
        mov   A,'#e'
        call  wr_data

```

```

        mov     A,#'g'
        call   wr_data
        mov     A,#' '
        call   wr_data
        ret

dl_mtr: mov     R0,#10
dl_mtrl:mov     R1,#100
        djnz   R1,$
        djnz   R0,dl_mtrl
        ret

;*****
;sub routine counter derajat
;*****
degree: call    line2
        inc    derajat2
        mov    A,derajat2
        mov    B,#10
        mov    A,derajat2
        div   AB
        add   A,#30H
        call  wr_data
        mov   A,B
        add   A,#30H
        call  wr_data
        ret

```

```

;*****
;sub routine delay step
;*****
dl_step:mov     R7,#12    ;setting derajat/step
dl_step1:
        mov     TMOD,#01    ;mode 16 bit
        mov     TH0,#0FCH   ;D8
        mov     TL0,#17H    ;EF
        setb   TR0
dl_step2:
        nop
        jbc    TF0,counter
        sjmp   dl_step2
counter:
        djnz   R7,dl_step1
        ret
;*****
;Data Pointer
;*****
ID1: DB ' KONTROL KEMUDI ',0
ID2: DB '  ARAH PESAWAT  ',0
ID3: DB 'SILVESTER Y.K.T.',0
ID4: DB '  98.17.059    ',0
ID5:  DB ' ELEKTRONIKA S1 ',0
ID6:  DB ' ITN  MALANG   ',0
MENU:  DB 'INPUT :      ',0
MENU2: DB '** deg      ',0
KIRI:  DB '□-          ',0
KANAN: DB ' --          ',0
MAXIMAL:DB 'SORRY MAX 35 deg',0
        End

```

REFERENSI

Features

- Compatible with MCS-51[®] Products
- 4K Bytes of In-System Programmable (ISP) Flash Memory
 - Endurance: 1000 Write/Erase Cycles
- 4.0V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 128 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-bit Timer/Counters
- Six Interrupt Sources
- Full Duplex UART Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Fast Programming Time
- Flexible ISP Programming (Byte and Page Mode)

Description

The AT89S51 is a low-power, high-performance CMOS 8-bit microcontroller with 4K bytes of in-system programmable Flash memory. 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 Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S51 provides the following standard features: 4K bytes of Flash, 128 bytes of SRAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.



8-bit Microcontroller with 4K Bytes In-System Programmable Flash

AT89S51

Rev. 2487A-10/01

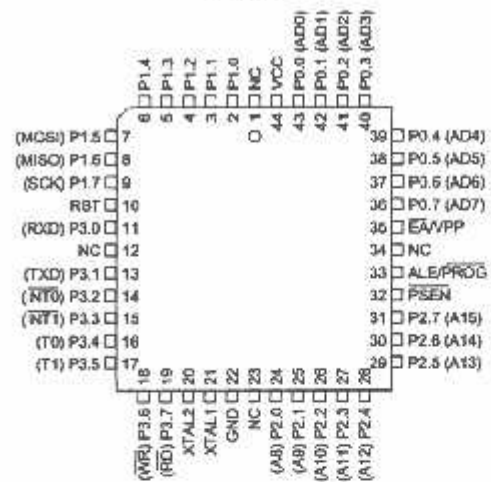


in Configurations

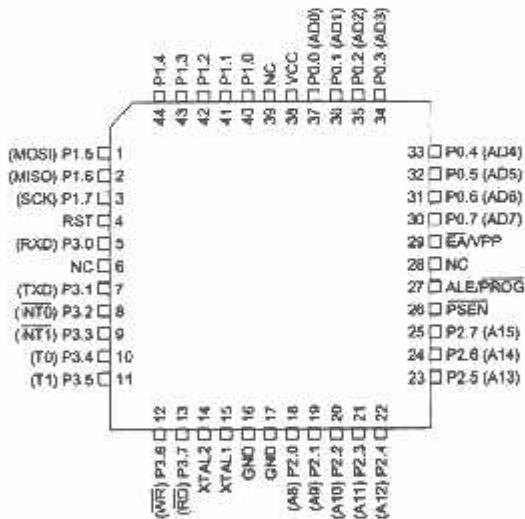
PDIP



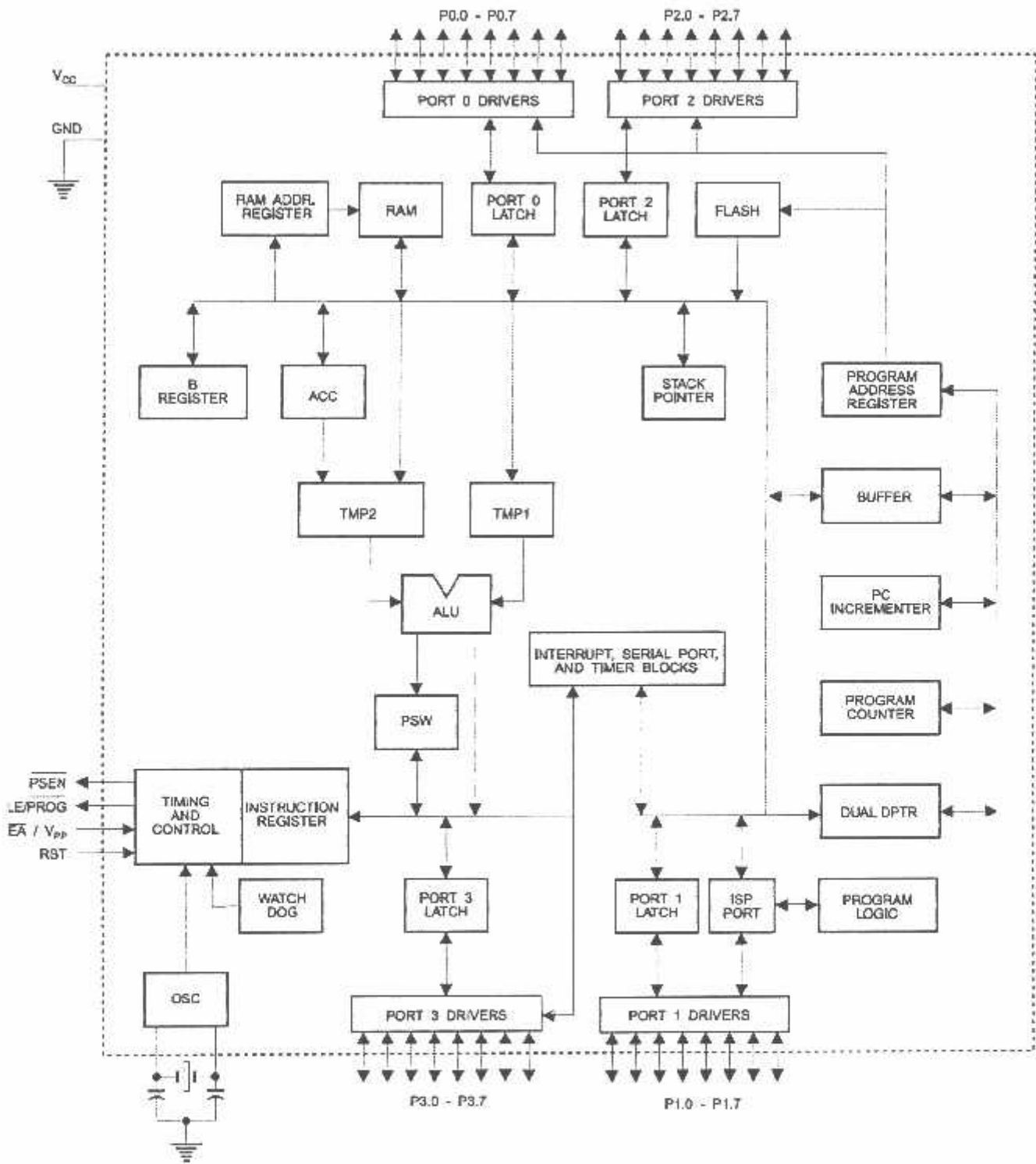
PLCC



TQFP



Block Diagram





Pin Description

V_{CC} Supply voltage.

V_{IND} Ground.

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

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

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. **External pull-ups are required during program verification.**

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

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

Port Pin	Alternate Functions
P1.5	MOSI (used for In-System Programming)
P1.6	MISO (used for In-System Programming)
P1.7	SCK (used for In-System Programming)

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

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

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

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

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

Port 3 also serves the functions of various special features of the AT89S51, as shown in the following table.

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

ST Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives High for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

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

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

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

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

When the AT89S51 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

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

$\overline{\text{EA}}$ should be strapped to V_{CC} for internal program executions.

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

AL1 Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

AL2 Output from the inverting oscillator amplifier



Special Function Registers

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

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

Table 1. AT89S51 SFR Map and Reset Values

0F8H								0FFH
0F0H	B 00000000							0F7H
0E8H								0EFH
0E0H	ACC 00000000							0E7H
0D8H								0DFH
0D0H	PSW 00000000							0D7H
0C8H								0CFH
0C0H								0C7H
0B8H	IP XX000000							0BFH
0B0H	P3 11111111							0B7H
0A8H	IE 0X000000							0AFH
0A0H	P2 11111111		AUXR1 XXXXXXXX0				WDRST XXXXXXXXX	0A7H
98H	SCON 00000000	SBUF XXXXXXXX						9FH
90H	P1 11111111							97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR XXX00XX0	8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	PCON 0XXX0000	87H

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

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

Table 2. AUXR: Auxiliary Register

AUXR		Address = 8EH				Reset Value = XXX00XX0B				
Not Bit Addressable										
Bit	7	6	5	WDIDLE	DISRTO	3	2	1	DISALE	0
-	Reserved for future expansion									
DISALE	Disable/Enable ALE									
	DISALE									
	Operating Mode									
	0	ALE is emitted at a constant rate of 1/6 the oscillator frequency								
	1	ALE is active only during a MOVX or MOVC instruction								
DISRTO	Disable/Enable Reset out									
	DISRTO									
	0	Reset pin is driven High after WDT times out								
	1	Reset pin is input only								
WDIDLE	Disable/Enable WDT in IDLE mode									
	WDIDLE									
	0	WDT continues to count in IDLE mode								
	1	WDT halts counting in IDLE mode								

Dual Data Pointer Registers: To facilitate accessing both internal and external data memory, two banks of 16-bit Data Pointer Registers are provided: DP0 at SFR address locations 82H-83H and DP1 at 84H-85H. Bit DPS = 0 in SFR AUXR1 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 rest under software control and is not affected by reset.

Table 3. AUXR1: Auxiliary Register 1

AUXR1								
Address = A2H								
Reset Value = XXXXXXX0B								
Not Bit Addressable								
Bit	7	6	5	4	3	2	1	DPS
	-	-	-	-	-	-	-	0
								1
-	Reserved for future expansion							
DPS	Data Pointer Register Select							
	DPS							
	0	Selects DPTR Registers DP0L, DP0H						
	1	Selects DPTR Registers DP1L, DP1H						

Memory Organization

MCS-51 devices have a separate address space for Program and Data Memory. Up to 64K bytes each of external Program and Data Memory can be addressed.

Program Memory

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

On the AT89S51, if \overline{EA} is connected to V_{CC} , program fetches to addresses 0000H through FFFH are directed to internal memory and fetches to addresses 1000H through FFFFH are directed to external memory.

Data Memory

The AT89S51 implements 128 bytes of on-chip RAM. The 128 bytes are accessible via direct and indirect addressing modes. Stack operations are examples of indirect addressing, so the 128 bytes of data RAM are available as stack space.

Watchdog Timer (Enabled with Reset-out)

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upsets. The WDT consists of a 14-bit counter and the Watchdog Timer Reset (WDTRST) SFR. The WDT is defaulted to disable from exiting reset. To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDTRST register (SFR location 0A6H). When the WDT is enabled, it will increment every machine cycle while the oscillator is running. The WDT timeout period is dependent on the external clock frequency. There is no way to disable the WDT except through reset (either hardware reset or WDT overflow reset). When WDT overflows, it will drive an output RESET HIGH pulse at the RST pin.

Using the WDT

To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDTRST register (SFR location 0A6H). When the WDT is enabled, the user needs to service it by writing 01EH and 0E1H to WDTRST to avoid a WDT overflow. The 14-bit counter overflows when it reaches 16383 (3FFFH), and this will reset the device. When the WDT is enabled, it will increment every machine cycle while the oscillator is running. This means the user must reset the WDT at least every 16383 machine cycles. To reset the WDT the user must write 01EH and 0E1H to WDTRST. WDTRST is a write-only register. The WDT counter cannot be read or written. When WDT overflows, it will generate an output RESET pulse at the RST pin. The RESET pulse duration is 98xTOSC, where TOSC=1/FOSC. To make the best use of the WDT, it

should be serviced in those sections of code that will periodically be executed within the time required to prevent a WDT reset.

WDT During Power-down and Idle

In Power-down mode the oscillator stops, which means the WDT also stops. While in Power-down mode, the user does not need to service the WDT. There are two methods of exiting Power-down mode: by a hardware reset or via a level-activated external interrupt, which is enabled prior to entering Power-down mode. When Power-down is exited with hardware reset, servicing the WDT should occur as it normally does whenever the AT89S51 is reset. Exiting Power-down with an interrupt is significantly different. The interrupt is held low long enough for the oscillator to stabilize. When the interrupt is brought high, the interrupt is serviced. To prevent the WDT from resetting the device while the interrupt pin is held low, the WDT is not started until the interrupt is pulled high. It is suggested that the WDT be reset during the interrupt service for the interrupt used to exit Power-down mode.

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

Before going into the IDLE mode, the WDIDLE bit in SFR AUXR is used to determine whether the WDT continues to count if enabled. The WDT keeps counting during IDLE (WDIDLE bit = 0) as the default state. To prevent the WDT from resetting the AT89S51 while in IDLE mode, the user should always set up a timer that will periodically exit IDLE, service the WDT, and reenter IDLE mode.

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

UART

The UART in the AT89S51 operates the same way as the UART in the AT89C51. For further information on the UART operation, refer to the ATMEL Web site (<http://www.atmel.com>). From the home page, select 'Products', then '8051-Architecture Flash Microcontroller', then 'Product Overview'.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S51 operate the same way as Timer 0 and Timer 1 in the AT89C51. For further information on the timers' operation, refer to the ATMEL Web site (<http://www.atmel.com>). From the home page, select 'Products', then '8051-Architecture Flash Microcontroller', then 'Product Overview'.

Interrupts

The AT89S51 has a total of five interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), two timer interrupts (Timers 0 and 1), and the serial port interrupt. These interrupts are all shown in Figure 1.

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

Note that Table 4 shows that bit position IE.6 is unimplemented. In the AT89S51, bit position IE.5 is also unimplemented. User software should not write 1s to these bit positions, since they may be used in future AT89 products.

The Timer 0 and Timer 1 flags, TF0 and TF1, are set at S5P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle.

Table 4. Interrupt Enable (IE) Register

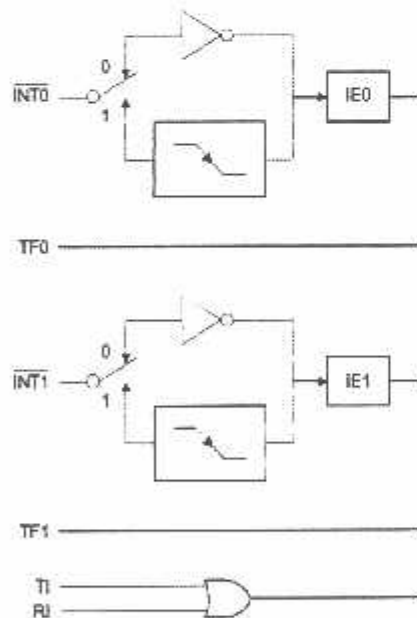
(MSB)				(LSB)			
EA	-	-	ES	ET1	EX1	ET0	EX0

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
-	IE.5	Reserved
ES	IE.4	Serial Port interrupt enable bit
ET1	IE.3	Timer 1 interrupt enable bit
EX1	IE.2	External interrupt 1 enable bit
ET0	IE.1	Timer 0 interrupt enable bit
EX0	IE.0	External interrupt 0 enable bit

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

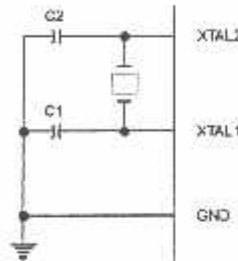
Figure 1. Interrupt Sources



Oscillator Characteristics

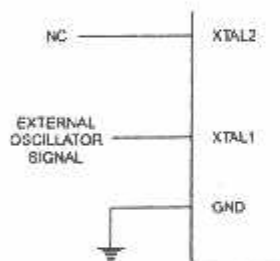
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 3. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Figure 2. Oscillator Connections



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

Figure 3. External Clock Drive Configuration



Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special 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.

Power-down Mode

In the Power-down mode, the oscillator is stopped, and the instruction that invokes Power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power-down mode is terminated. Exit from Power-down mode can be initiated either by a hardware reset or by activation of an enabled external interrupt into INT0 or INT1. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Table 5. 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

Program Memory Lock Bits

The AT89S51 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.

Table 6. Lock Bit Protection Modes

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

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

Programming the Flash – Parallel Mode

The AT89S51 is shipped with the on-chip Flash memory array ready to be programmed. The programming interface needs a high-voltage (12-volt) program enable signal and is compatible with conventional third-party Flash or EPROM programmers.

The AT89S51 code memory array is programmed byte-by-byte.

Programming Algorithm: Before programming the AT89S51, the address, data, and control signals should be set up according to the Flash programming mode table and Figures 13 and 14. To program the AT89S51, take the following steps:

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise \overline{EA}/V_{PP} to 12V.
5. Pulse ALE/ \overline{PROG} once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 50 μ s. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

Data Polling: The AT89S51 features Data Polling to indicate the end of a byte write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on P0.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

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

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

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

(000H) = 1EH indicates manufactured by Atmel
 (100H) = 51H indicates 89S51
 (200H) = 06H

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

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

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

Programming the Flash – Serial Mode

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

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

Either an external system clock can be supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK) frequency should be less than 1/16 of the crystal frequency. With a 33 MHz oscillator clock, the maximum SCK frequency is 2 MHz.

Serial Programming Algorithm

To program and verify the AT89S51 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 33 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 16.
3. The Code array is programmed one byte at a time in either the Byte or Page mode. The write cycle is self-timed and typically takes less than 0.5 ms at 5V.
4. Any memory location can be verified by using the Read instruction that 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 device 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.

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

Serial Programming Instruction Set

The Instruction Set for Serial Programming follows a 4-byte protocol and is shown in Table 8 on page 18.

Programming Interface – Parallel Mode

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

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

Table 7. Flash Programming Modes

Mode	V _{CC}	RST	PSEN	ALE/ PROG	EA/ V _{PP}	P2.6	P2.7	P3.3	P3.6	P3.7	P0.7-0 Data	Address	
												P2.3-0	P1.7-0
Write Code Data	5V	H	L		12V	L	H	H	H	H	D _{IN}	A11-8	A7-0
Read Code Data	5V	H	L	H	H	L	L	L	H	H	D _{OUT}	A11-8	A7-0
Write Lock Bit 1	5V	H	L		12V	H	H	H	H	H	X	X	X
Write Lock Bit 2	5V	H	L		12V	H	H	H	L	L	X	X	X
Write Lock Bit 3	5V	H	L		12V	H	L	H	H	L	X	X	X
Read Lock Bits 1, 2, 3	5V	H	L	H	H	H	H	L	H	L	P0.2, P0.3, P0.4	X	X
Chip Erase	5V	H	L		12V	H	L	H	L	L	X	X	X
Read Atmel ID	5V	H	L	H	H	L	L	L	L	L	1EH	0000	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	51H	0001	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	06H	0010	00H

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

Figure 4. Programming the Flash Memory (Parallel Mode)

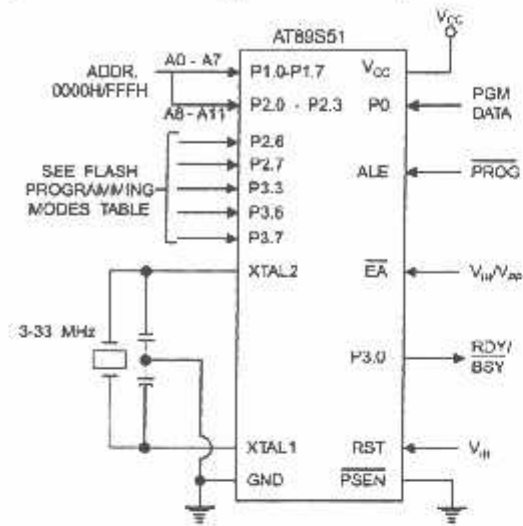
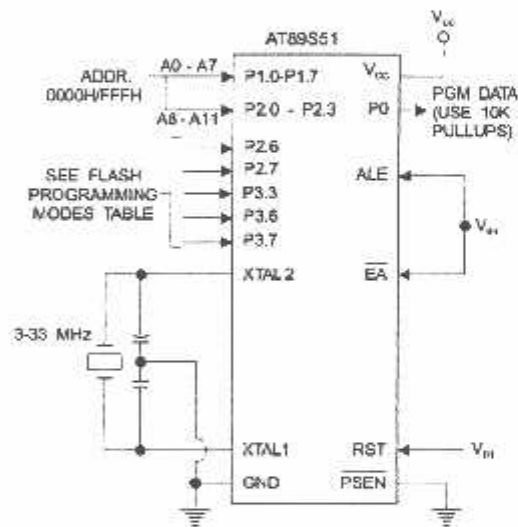


Figure 5. Verifying the Flash Memory (Parallel Mode)



Flash Programming and Verification Characteristics (Parallel Mode)

$T_A = 20^\circ\text{C}$ to 30°C , $V_{CC} = 4.5$ to 5.5V

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Supply Voltage	11.5	12.5	V
I_{PP}	Programming Supply Current		10	mA
I_{CC}	V_{CC} Supply Current		30	mA
$1/t_{CLCL}$	Oscillator Frequency	3	33	MHz
t_{AVGL}	Address Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold After $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold After $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{EHS}	P2.7 ($\overline{\text{ENABLE}}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{\text{PROG}}$ Low	10		μs
t_{GHSL}	V_{PP} Hold After $\overline{\text{PROG}}$	10		μs
t_{GLGH}	$\overline{\text{PROG}}$ Width	0.2	1	μs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{\text{ENABLE}}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	Data Float After $\overline{\text{ENABLE}}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{\text{PROG}}$ High to $\overline{\text{BUSY}}$ Low		1.0	μs
t_{WC}	Byte Write Cycle Time		50	μs

Figure 6. Flash Programming and Verification Waveforms – Parallel Mode

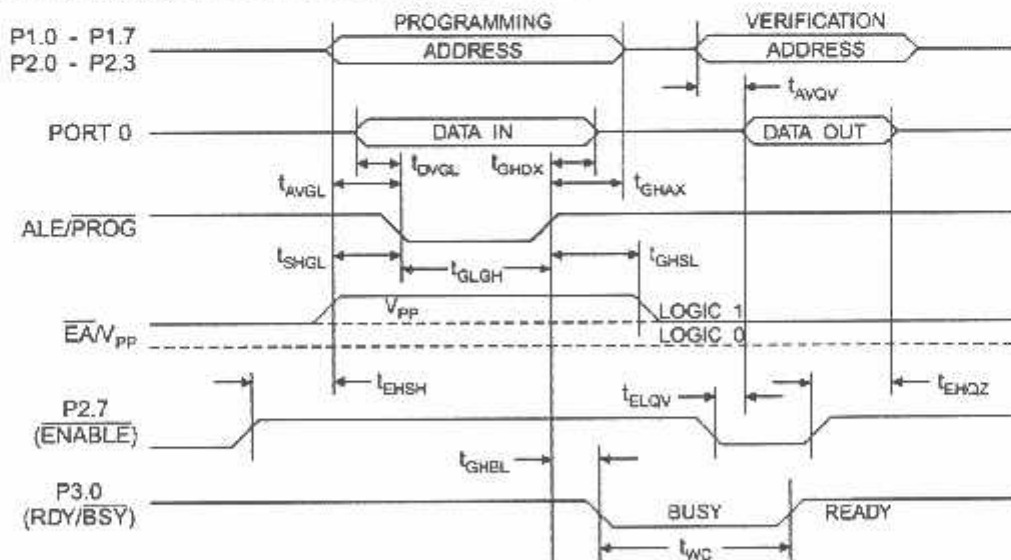
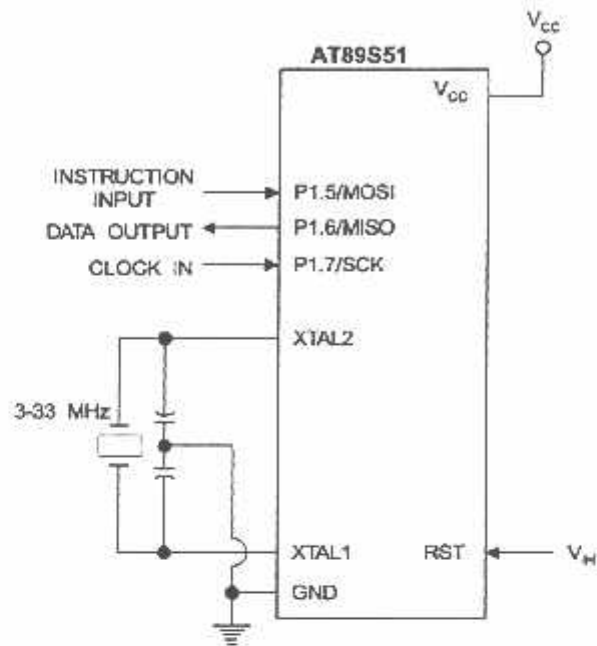


Figure 7. Flash Memory Serial Downloading



Flash Programming and Verification Waveforms – Serial Mode

Figure 8. Serial Programming Waveforms

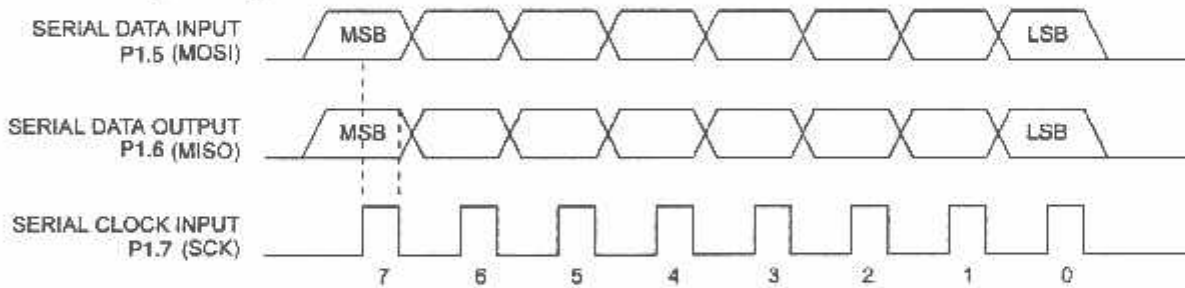


Table 8. Serial Programming Instruction Set

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

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

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

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

After Reset signal is high, SCK should be low for at least 64 system clocks before it goes high to clock in the enable data bytes. No pulsing of Reset signal is necessary. SCK should be no faster than 1/16 of the system clock at XTAL1.

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

Serial Programming Characteristics

Figure 9. Serial Programming Timing

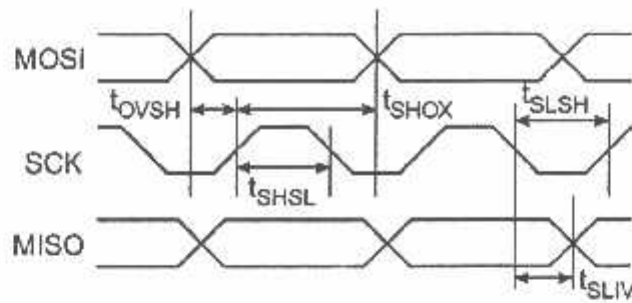


Table 9. Serial Programming Characteristics, $T_A = -40^\circ\text{C}$ to 85°C , $V_{CC} = 4.0 - 5.5\text{V}$ (Unless Otherwise Noted)

Symbol	Parameter	Min	Typ	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0		33	MHz
t_{CLCL}	Oscillator Period	30			ns
t_{SHSL}	SCK Pulse Width High	$8 t_{CLCL}$			ns
t_{SLSH}	SCK Pulse Width Low	$8 t_{CLCL}$			ns
t_{OVSH}	MOSI Setup to SCK High	t_{CLCL}			ns
t_{SHOX}	MOSI Hold after SCK High	$2 t_{CLCL}$			ns
t_{SLIV}	SCK Low to MISO Valid	10	16	32	ns
t_{ERASE}	Chip Erase Instruction Cycle Time			500	ms
t_{SWC}	Serial Byte Write Cycle Time			$64 t_{CLCL} + 400$	μs



Absolute Maximum Ratings*

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

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

DC Characteristics

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

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low Voltage	(Except \overline{EA})	-0.5	$0.2 V_{CC} - 0.1$	V
V_{IL1}	Input Low Voltage (\overline{EA})		-0.5	$0.2 V_{CC} - 0.3$	V
V_{IH}	Input High Voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
V_{IH1}	Input High Voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V
V_{OL}	Output Low Voltage ⁽¹⁾ (Ports 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.45	V
V_{OL1}	Output Low Voltage ⁽¹⁾ (Port 0, ALE, PSEN)	$I_{OL} = 3.2 \text{ mA}$		0.45	V
I_{OH}	Output High Voltage (Ports 1,2,3, ALE, 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
I_L	Logical 0 Input Current (Ports 1,2,3)	$V_{IN} = 0.45\text{V}$		-50	μA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	$V_{IN} = 2\text{V}$, $V_{CC} = 5\text{V} \pm 10\%$		-650	μA
I_U	Input Leakage Current (Port 0, \overline{EA})	$0.45 < V_{IN} < V_{CC}$		± 10	μA
R_{RST}	Reset Pulldown Resistor		50	300	$\text{K}\Omega$
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
		Power-down Mode ⁽²⁾	$V_{CC} = 5.5\text{V}$		50

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

- Maximum I_{OL} per port pin: 10 mA
- Maximum I_{OL} per 8-bit port:
- Port 0: 26 mA Ports 1, 2, 3: 15 mA
- Maximum total I_{OL} for all output pins: 71 mA

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

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

AT89S51

AC Characteristics

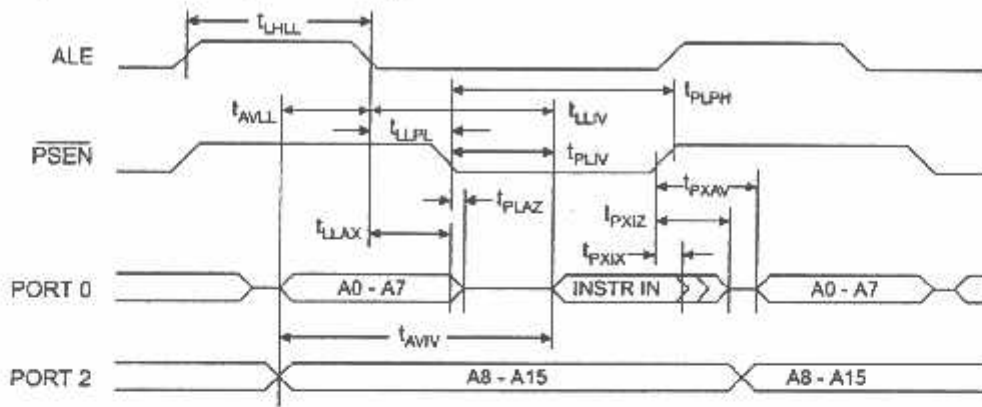
Under operating conditions, load capacitance for Port 0, ALE/ $\overline{\text{PROG}}$, and $\overline{\text{PSEN}}$ = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

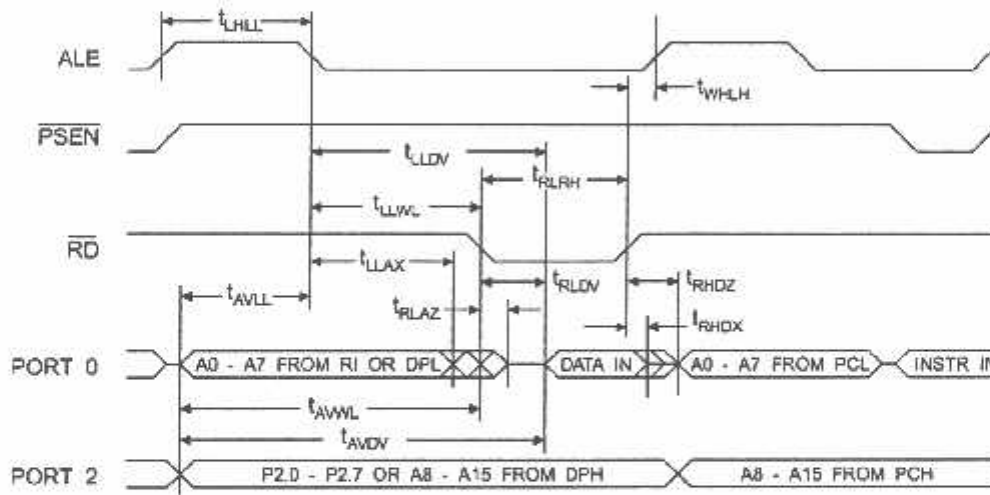
Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{\text{CLCL}}$	Oscillator Frequency			0	33	MHz
t_{HLL}	ALE Pulse Width	127		$2t_{\text{CLCL}}-40$		ns
t_{AVLL}	Address Valid to ALE Low	43		$t_{\text{CLCL}}-25$		ns
t_{LLAX}	Address Hold After ALE Low	48		$t_{\text{CLCL}}-25$		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{\text{CLCL}}-65$	ns
t_{LLPL}	ALE Low to $\overline{\text{PSEN}}$ Low	43		$t_{\text{CLCL}}-25$		ns
t_{PLPH}	$\overline{\text{PSEN}}$ Pulse Width	205		$3t_{\text{CLCL}}-45$		ns
t_{PLIV}	$\overline{\text{PSEN}}$ Low to Valid Instruction In		145		$3t_{\text{CLCL}}-60$	ns
t_{PKIK}	Input Instruction Hold After $\overline{\text{PSEN}}$	0		0		ns
t_{PKIF}	Input Instruction Float After $\overline{\text{PSEN}}$		59		$t_{\text{CLCL}}-25$	ns
t_{PXAV}	$\overline{\text{PSEN}}$ to Address Valid	75		$t_{\text{CLCL}}-8$		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{\text{CLCL}}-80$	ns
t_{PLAZ}	$\overline{\text{PSEN}}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{\text{RD}}$ Pulse Width	400		$6t_{\text{CLCL}}-100$		ns
t_{WLWH}	$\overline{\text{WR}}$ Pulse Width	400		$6t_{\text{CLCL}}-100$		ns
t_{RDV}	$\overline{\text{RD}}$ Low to Valid Data In		252		$5t_{\text{CLCL}}-90$	ns
t_{RDIX}	Data Hold After $\overline{\text{RD}}$	0		0		ns
t_{RDIZ}	Data Float After $\overline{\text{RD}}$		97		$2t_{\text{CLCL}}-28$	ns
t_{LDV}	ALE Low to Valid Data In		517		$8t_{\text{CLCL}}-150$	ns
t_{WDV}	Address to Valid Data In		585		$9t_{\text{CLCL}}-165$	ns
t_{LWL}	ALE Low to $\overline{\text{RD}}$ or $\overline{\text{WR}}$ Low	200	300	$3t_{\text{CLCL}}-50$	$3t_{\text{CLCL}}+50$	ns
t_{WWL}	Address to $\overline{\text{RD}}$ or $\overline{\text{WR}}$ Low	203		$4t_{\text{CLCL}}-75$		ns
t_{VWVX}	Data Valid to $\overline{\text{WR}}$ Transition	23		$t_{\text{CLCL}}-30$		ns
t_{VWVH}	Data Valid to $\overline{\text{WR}}$ High	433		$7t_{\text{CLCL}}-130$		ns
t_{VHDX}	Data Hold After $\overline{\text{WR}}$	33		$t_{\text{CLCL}}-25$		ns
t_{RLAZ}	$\overline{\text{RD}}$ Low to Address Float		0		0	ns
t_{VHLH}	$\overline{\text{RD}}$ or $\overline{\text{WR}}$ High to ALE High	43	123	$t_{\text{CLCL}}-25$	$t_{\text{CLCL}}+25$	ns



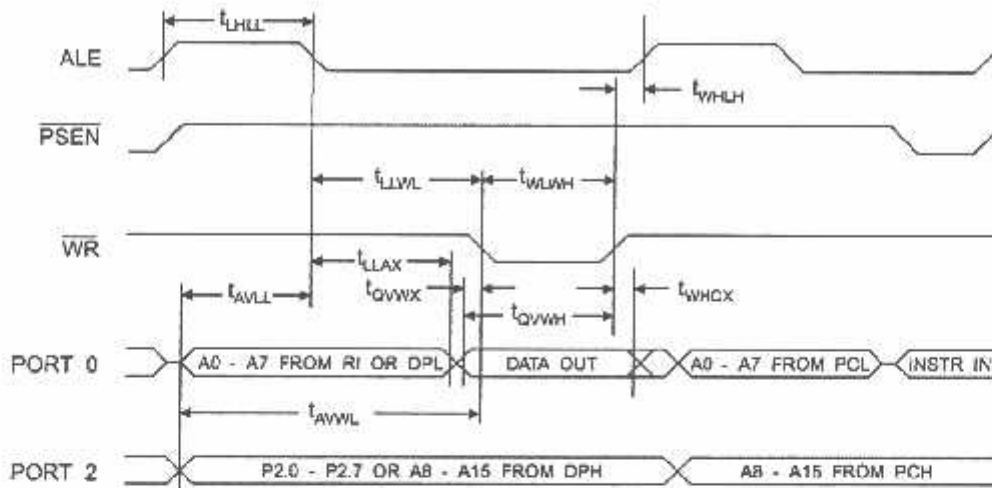
External Program Memory Read Cycle



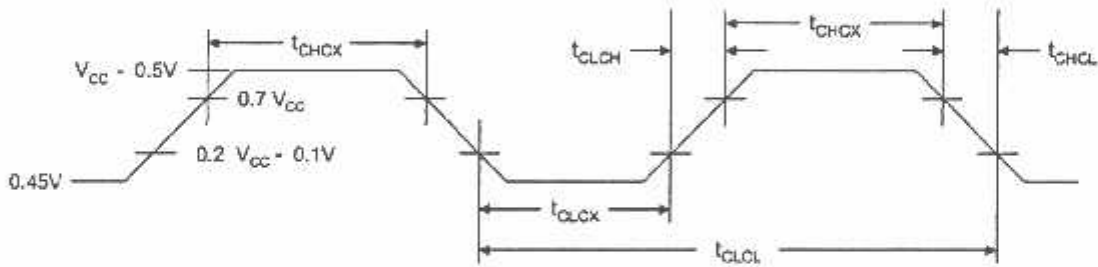
External Data Memory Read Cycle



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

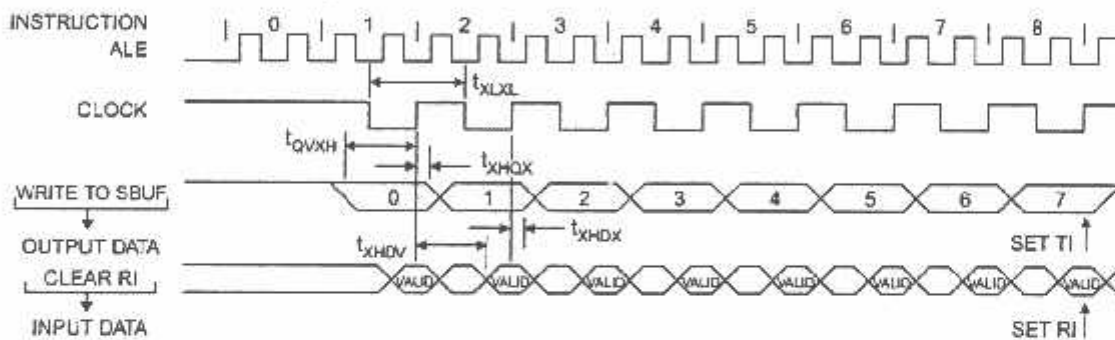
Symbol	Parameter	Min	Max	Units
f_{CLK}	Oscillator Frequency	0	33	MHz
t_{CLCL}	Clock Period	30		ns
t_{CHCX}	High Time	12		ns
t_{CLCX}	Low Time	12		ns
t_{CLCH}	Rise Time		5	ns
t_{CHCL}	Fall Time		5	ns

Serial Port Timing: Shift Register Mode Test Conditions

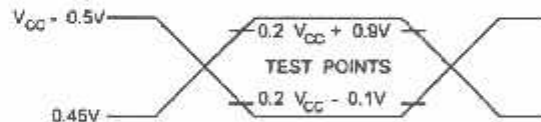
The values in this table are valid for $V_{CC} = 4.0V$ to $5.5V$ and Load Capacitance = 80 pF.

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		μs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}-133$		ns
t_{XHGX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}-80$		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHGV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}-133$	ns

Shift Register Mode Timing Waveforms



C Testing Input/Output Waveforms⁽¹⁾



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

Output Waveforms⁽¹⁾



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

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 5.5V	AT89S51-24AC	44A	Commercial (0°C to 70°C)
		AT89S51-24JC	44J	
		AT89S51-24PC	40P6	
		AT89S51-24AI	44A	Industrial (-40°C to 85°C)
		AT89S51-24JI	44J	
		AT89S51-24PI	40P6	
33	4.5V to 5.5V	AT89S51-33AC	44A	Commercial (0°C to 70°C)
		AT89S51-33JC	44J	
		AT89S51-33PC	40P6	

 = Preliminary Availability

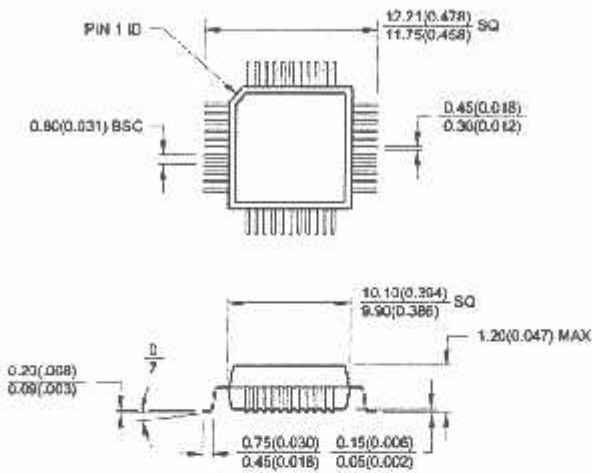
Package Type

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



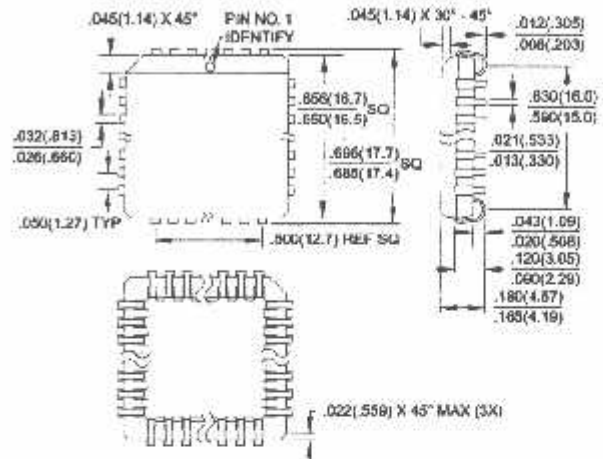
Packaging Information

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

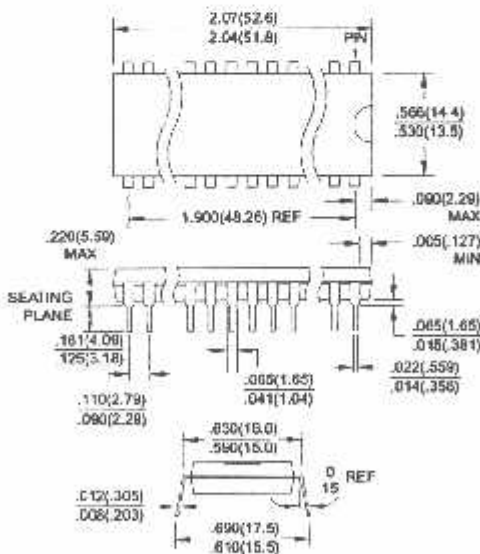


*Controlling dimension: millimeters

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



40P6, 40-pin, 0.600" Wide, Plastic Dual Inline Package (PDIP)
 Dimensions in Inches and (Millimeters)
 JEDEC STANDARD MS-011 AC





Atmel Headquarters

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

Europe
Atmel SarL
Route des Arsenaux 41
Casa Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

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 Product Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80908
TEL (719) 576-3300
FAX (719) 540-1759

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

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

Atmel Nantes
La Chantrerie
BP 70602
44308 Nantes Cedex 3, France
TEL (33) 0 2 40 18 18 18
FAX (33) 0 2 40 18 19 60

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

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

e-mail
literature@atmel.com

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

Atmel Corporation 2001.

Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any 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.

MEL[®] is the registered trademark of Atmel.

DS-51[®] is the registered trademark of Intel Corporation. Terms and product names in this document may be trademarks of others.



Printed on recycled paper.

2487A-10/01xM

ADC0801/ADC0802/ADC0803/ADC0804/ADC0805 8-Bit μ P Compatible A/D Converters

General Description

The ADC0801, ADC0802, ADC0803, ADC0804 and ADC0805 are CMOS 8-bit successive approximation A/D converters that use a differential potentiometric ladder — similar to the 256R products. These converters are designed to allow operation with the NSC800 and INS8080A derivative control bus with TRI-STATE® output latches directly driving the data bus. These A/Ds appear like memory locations or I/O ports to the microprocessor and no interfacing logic is needed.

Differential analog voltage inputs allow increasing the common-mode rejection and offsetting the analog zero input voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

Features

- Compatible with 8080 μ P derivatives — no interfacing logic needed - access time - 135 ns
- Easy Interface to all microprocessors, or operates "stand alone"

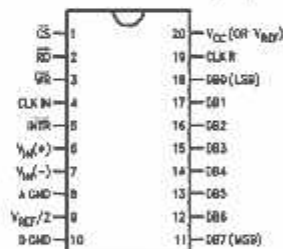
- Differential analog voltage inputs
- Logic inputs and outputs meet both MOS and TTL voltage level specifications
- Works with 2.5V (LM336) voltage reference
- On-chip clock generator
- 0V to 5V analog input voltage range with single 5V supply
- No zero adjust required
- 0.3" standard width 20-pin DIP package
- 20-pin molded chip carrier or small outline package
- Operates ratiometrically or with 5 V_{DD} , 2.5 V_{DD} , or analog span adjusted voltage reference

Key Specifications

- Resolution 8 bits
- Total error $\pm 1/4$ LSB, $\pm 1/2$ LSB and ± 1 LSB
- Conversion time 100 μ s

Connection Diagram

ADC080X
Dual-In-Line and Small Outline (SO) Packages



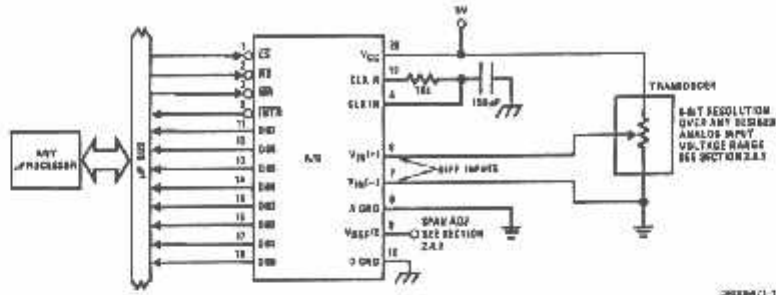
See Ordering Information

Ordering Information

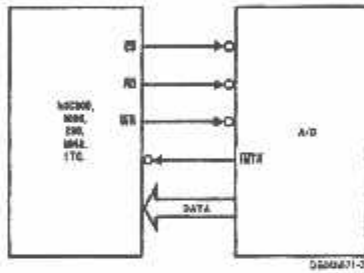
TEMP RANGE		0°C TO 70°C	0°C TO 70°C	-40°C TO +85°C
ERROR	$\pm 1/4$ Bit Adjusted			ADC0801LCN
	$\pm 1/2$ Bit Unadjusted	ADC0802LCWM		ADC0802LCN
	$\pm 1/4$ Bit Adjusted		ADC0804LCN	ADC0803LCN
	± 1 Bit Unadjusted	ADC0804LCWM		ADC0805LCN/ADC0804LCJ
PACKAGE OUTLINE		M20B — Small Outline	N20A — Molded DIP	

TRI-STATE® is a registered trademark of National Semiconductor Corp.
2.5V is a registered trademark of Texas Corp.

Typical Applications



8080 interface



Error Specification (Includes Full-Scale, Zero Error, and Non-Linearity)

Part Number	Full-Scale Adjusted	$V_{REF}/2=2.500 V_{DC}$ (No Adjustments)	$V_{REF}/2=$ No Connection (No Adjustments)
ADC0801	$\pm 1/4$ LSB		
ADC0802		$\pm 1/2$ LSB	
ADC0803	$\pm 1/2$ LSB		
ADC0804		± 1 LSB	
ADC0805			± 1 LSB

Absolute Maximum Ratings (Notes 1, 2)

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	
Logic Control Inputs	-0.3V to +18V
At Other Input and Outputs	-0.3V to ($V_{CC}+0.3V$)
Lead Temp. (Soldering, 10 seconds)	
Dual-In-Line Package (plastic)	260°C
Dual-In-Line Package (ceramic)	300°C
Surface Mount Package	
Vapor Phase (80 seconds)	215°C

Infrared (15 seconds)	220°C
Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A=25^\circ\text{C}$	875 mW
ESD Susceptibility (Note 10)	800V

Operating Ratings (Notes 1, 2)

Temperature Range		$T_{MIN} \leq T_A \leq T_{MAX}$
ADC0804LCJ		-40°C $\leq T_A \leq$ +85°C
ADC0801/02/03/05LCN		-40°C $\leq T_A \leq$ +85°C
ADC0804LCN		0°C $\leq T_A \leq$ +70°C
ADC0802/04LCWM		0°C $\leq T_A \leq$ +70°C
Range of V_{CC}		4.5 V_{DC} to 6.3 V_{DC}

Electrical Characteristics

The following specifications apply for $V_{CC}=5 V_{DC}$, $T_{MIN} \leq T_A \leq T_{MAX}$ and $f_{CLK}=640$ kHz unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
ADC0801: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/4$	LSB
ADC0802: Total Unadjusted Error (Note 8)	$V_{REF}/2=2.500 V_{DC}$			$\pm 1/2$	LSB
ADC0803: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/2$	LSB
ADC0804: Total Unadjusted Error (Note 8)	$V_{REF}/2=2.500 V_{DC}$			± 1	LSB
ADC0805: Total Unadjusted Error (Note 8)	$V_{REF}/2$ -No Connection			± 1	LSB
$V_{REF}/2$ Input Resistance (Pin 9)	ADC0801/02/03/05	2.5	8.0		k Ω
	ADC0804 (Note 9)	0.75	1.1		k Ω
Analog Input Voltage Range	(Note 4) $V(+)$ or $V(-)$	Ground-0.05		$V_{CC}+0.05$	V_{DC}
DC Common-Mode Error	Over Analog Input Voltage Range		$\pm 1/16$	$\pm 1/4$	LSB
Power Supply Sensitivity	$V_{CC}=5 V_{DC} \pm 10\%$ Over Allowed $V_{IN}(+)$ and $V_{IN}(-)$ Voltage Range (Note 4)		$\pm 1/16$	$\pm 1/4$	LSB

AC Electrical Characteristics

The following specifications apply for $V_{CC}=5 V_{DC}$ and $T_{MIN} \leq T_A \leq T_{MAX}$ unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
T_n	Conversion Time	$f_{CLK}=640$ kHz (Note 8)	103		114	μs
T_c	Conversion Time	(Notes 5, 6)	66		73	$1/f_{CLK}$
f_{CLK}	Clock Frequency	$V_{CC}=5V$, (Note 5)	100	640	1460	kHz
	Clock Duty Cycle		40		60	%
CR	Conversion Rate in Free-Running Mode	INTR tied to WR with $\overline{CS}=0 V_{DC}$, $f_{CLK}=640$ kHz	8770		9708	conv/s
t_{WIDRIL}	Width of WR Input (Start Pulse Width)	$\overline{CS}=0 V_{DC}$ (Note 7)	100			ns
t_{ACC}	Access Time (Delay from Falling Edge of RD to Output Data Valid)	$C_L=100$ pF		135	200	ns
$t_{HI, LOH}$	TRI-STATE Control (Delay from Rising Edge of RD to Hi-Z State)	$C_L=10$ pF, $R_L=10k$ (See TRI-STATE Test Circuits)		125	200	ns
$t_{WR, t_{RD}}$	Delay from Falling Edge of WR or RD to Reset of INTR			300	450	ns
C_{IN}	Input Capacitance of Logic Control Inputs			5	7.5	pF

AC Electrical Characteristics (Continued)

The following specifications apply for $V_{CC}=5 V_{DC}$ and $T_{MIN} \leq T_A \leq T_{MAX}$ unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
C_{OUT}	TRI-STATE Output Capacitance (Data Buffers)			5	7.5	pF
CONTROL INPUTS [Note: CLK IN (Pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately]						
$V_{IN}(1)$	Logical "1" Input Voltage (Except Pin 4 CLK IN)	$V_{CC}=5.25 V_{DC}$	2.0		15	V_{DC}
$V_{IN}(0)$	Logical "0" Input Voltage (Except Pin 4 CLK IN)	$V_{CC}=4.75 V_{DC}$			0.8	V_{DC}
$I_{IN}(1)$	Logical "1" Input Current (All Inputs)	$V_{IN}=5 V_{DC}$		0.005	1	μA_{DC}
$I_{IN}(0)$	Logical "0" Input Current (All Inputs)	$V_{IN}=0 V_{DC}$	-1	-0.005		μA_{DC}
CLOCK IN AND CLOCK R						
V_{T+}	CLK IN (Pin 4) Positive Going Threshold Voltage		2.7	3.1	3.5	V_{DC}
V_{T-}	CLK IN (Pin 4) Negative Going Threshold Voltage		1.5	1.8	2.1	V_{DC}
V_H	CLK IN (Pin 4) Hysteresis (V_{T+}) - (V_{T-})		0.6	1.3	2.0	V_{DC}
$V_{OUT}(0)$	Logical "0" CLK R Output Voltage	$I_O=360 \mu A$ $V_{CC}=4.75 V_{DC}$			0.4	V_{DC}
$V_{OUT}(1)$	Logical "1" CLK R Output Voltage	$I_O=-360 \mu A$ $V_{CC}=4.75 V_{DC}$	2.4			V_{DC}
DATA OUTPUTS AND INTR						
$V_{OUT}(0)$	Logical "0" Output Voltage Data Outputs INTR Output	$I_{OUT}=1.6 mA, V_{CC}=4.75 V_{DC}$ $I_{OUT}=1.0 mA, V_{CC}=4.75 V_{DC}$			0.4 0.4	V_{DC} V_{DC}
$V_{OUT}(1)$	Logical "1" Output Voltage	$I_O=-360 \mu A, V_{CC}=4.75 V_{DC}$	2.4			V_{DC}
$V_{OUT}(1)$	Logical "1" Output Voltage	$I_O=-10 \mu A, V_{CC}=4.75 V_{DC}$	4.5			V_{DC}
I_{OUT}	TRI-STATE Disabled Output Leakage (All Data Buffers)	$V_{OUT}=0 V_{DC}$ $V_{OUT}=5 V_{DC}$	-3		3	μA_{DC} μA_{DC}
I_{SOURCE}		V_{OUT} Short to Gnd, $T_A=25^\circ C$	4.5	6		mA_{DC}
I_{SINK}		V_{OUT} Short to V_{CC} , $T_A=25^\circ C$	9.0	16		mA_{DC}
POWER SUPPLY						
I_{CC}	Supply Current (Includes Ladder Current) ADC0801/02/03/04/LCJ/05 ADC0804/LCN/LCWM	$f_{CLK}=640 kHz$, $V_{REF/2}=NC$, $T_A=25^\circ C$ and $\overline{CS}=5V$				
				1.1	1.8	mA
				1.9	2.5	mA

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. The separate A Gnd point should always be wired to the D Gnd.

Note 3: A zener diode exists internally from V_{CC} to Gnd and has a typical breakdown voltage of 7 V_{DC} .

Note 4: For $V_{IN}(+) \leq V_{IN}(+)$ the digital output code will be 0000 0000. Two on-chip diodes are tied to each analog input (see block diagram) which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the V_{CC} supply. Be careful, during testing at low V_{CC} levels (4.5V), as high level analog inputs (5V) can cause this input diode to conduct—especially at elevated temperatures, and cause errors for analog inputs near full-scale. The spec allows 50 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 50 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 4.950 V_{DC} over temperature variations, initial tolerance and loading.

Note 5: Accuracy is guaranteed at $f_{CLK} = 640 kHz$. At higher clock frequencies accuracy can degrade. For lower clock frequencies, the duty cycle limits can be extended so long as the minimum clock high time interval or minimum clock low time interval is no less than 275 ns.

Note 6: With an asynchronous start pulse, up to 8 clock periods may be required before the internal clock phases are proper to start the conversion process. The start request is internally latched; see Figure 4 and section 2.0.

AC Electrical Characteristics (Continued)

Note 7: The \overline{CS} input is assumed to bracket the \overline{WR} strobe input and therefore timing is dependent on the \overline{WR} pulse width. An arbitrarily wide pulse width will hold the converter in a reset mode and the start of conversion is initiated by the low to high transition of the \overline{WR} pulse (see timing diagrams).

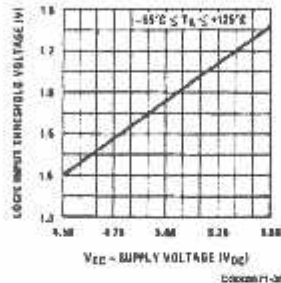
Note 8: None of these ADs requires a zero adjust (see section 2.5.1). To obtain zero code at other analog input voltages see section 2.5 and Figure 7.

Note 9: The $V_{REF/2}$ pin is the center point of a two-resistor divider connected from V_{CC} to ground. In all versions of the ADC0801, ADC0802, ADC0803, and ADC0805, and in the ADC0804LCJ, each resistor is typically 16 k Ω . In all versions of the ADC0804 except the ADC0804LCJ, each resistor is typically 2.2 k Ω .

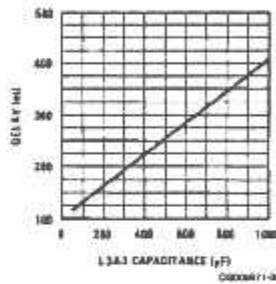
Note 10: Human body model, 100 pF discharged through a 1.5 k Ω resistor.

Typical Performance Characteristics

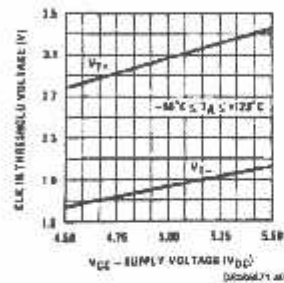
Logic Input Threshold Voltage vs. Supply Voltage



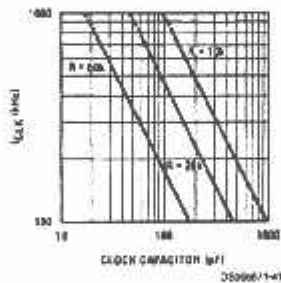
Delay From Falling Edge of RD to Output Data Valid vs. Load Capacitance



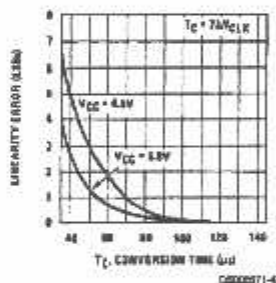
CLK IN Schmitt Trip Levels vs. Supply Voltage



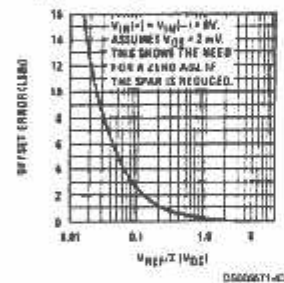
f_{CLK} vs. Clock Capacitor



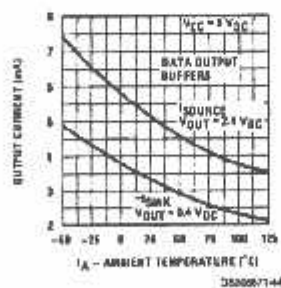
Full-Scale Error vs Conversion Time



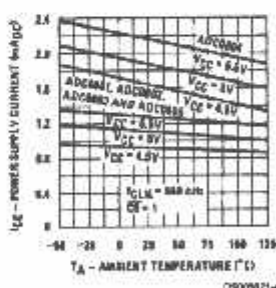
Effect of Unadjusted Offset Error vs. V_{REF/2} Voltage



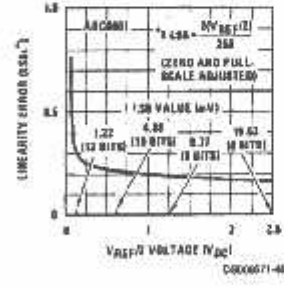
Output Current vs Temperature



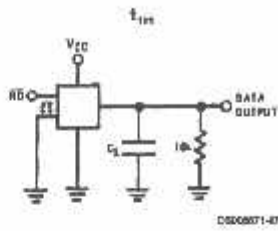
Power Supply Current vs Temperature (Note 9)



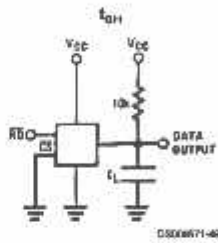
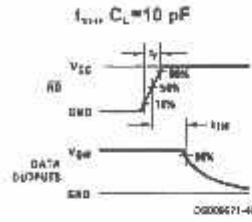
Linearity Error at Low V_{REF/2} Voltages



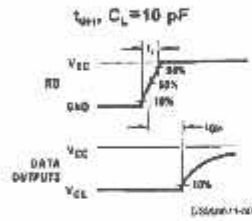
TRI-STATE Test Circuits and Waveforms



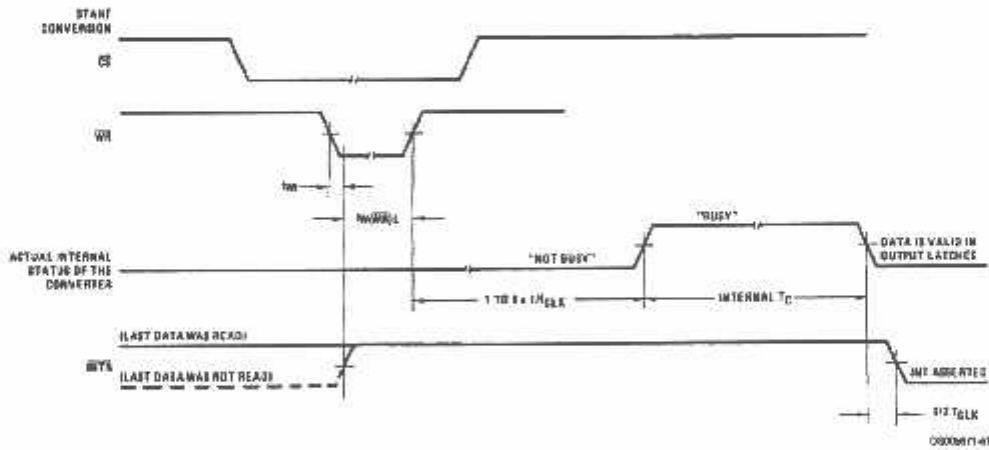
$t_c = 20 \text{ ns}$



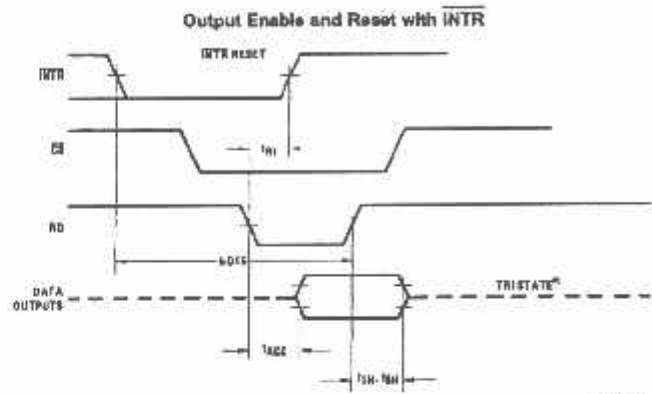
$t_c = 20 \text{ ns}$



Timing Diagrams (All timing is measured from the 50% voltage points)



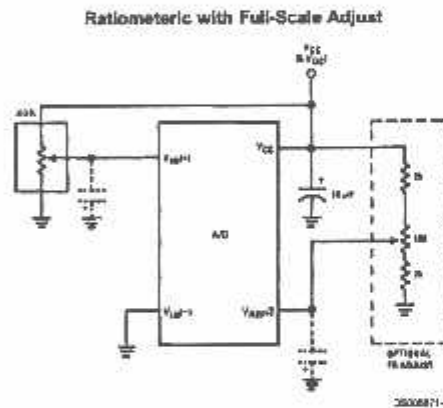
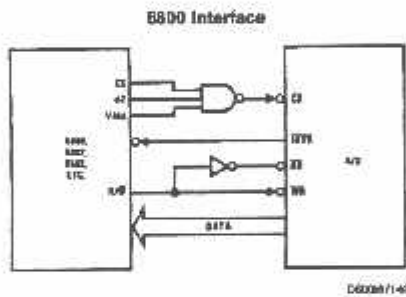
Timing Diagrams (All timing is measured from the 50% voltage points) (Continued)



D62068/140

Note: Read strobe must occur 8 clock periods ($8t_{CLK}$) after assertion of interrupt to guarantee reset of INTR.

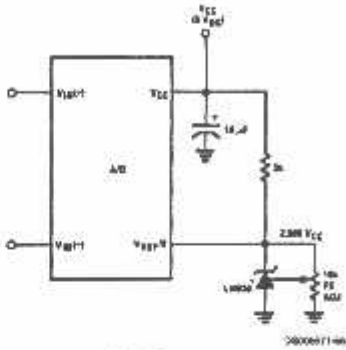
Typical Applications



Note: before using caps at V_{IN} or $V_{REF}(2)$, see section 2.3.2 Input Bypass Capacitors.

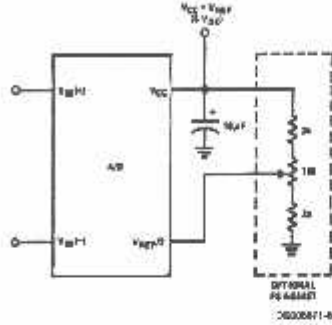
Typical Applications (Continued)

Absolute with a 2.500V Reference



06000671-06

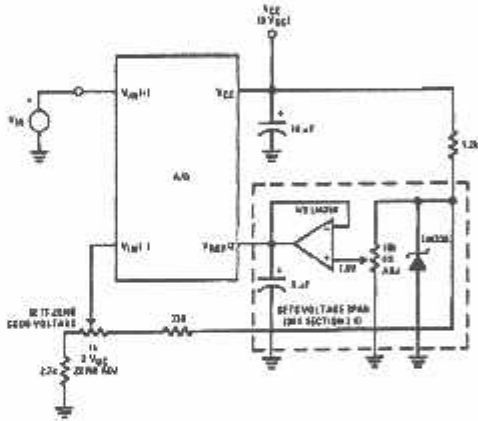
Absolute with a 5V Reference



06000671-08

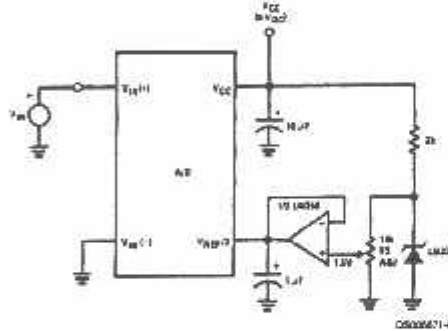
*For low power, see also LM085-2.5

Zero-Shift and Span Adjust: $2V \leq V_{IN} \leq 5V$



06000671-07

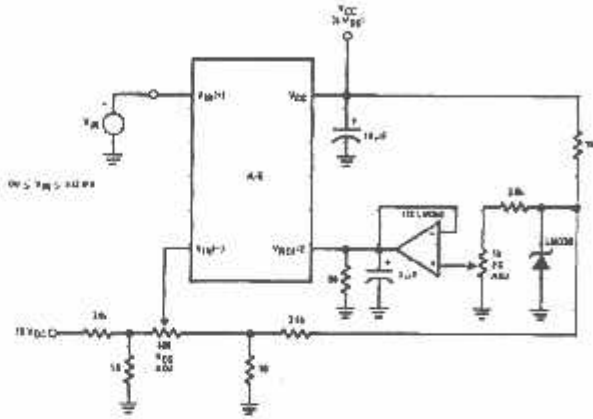
Span Adjust: $0V \leq V_{IN} \leq 3V$



06000671-08

Typical Applications (Continued)

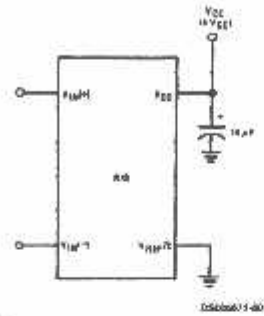
Directly Converting a Low-Level Signal



$V_{REF} = 256 \text{ mV}$

046066/1-00

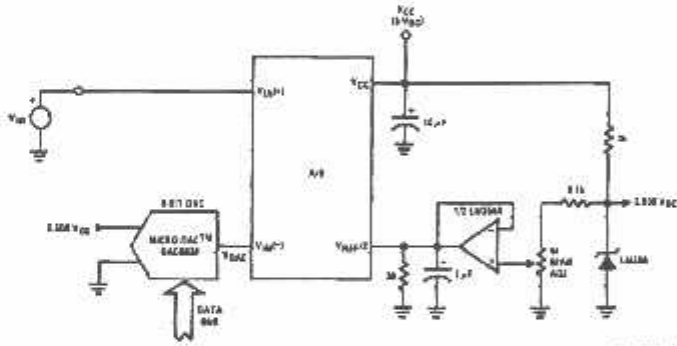
A μP Interfaced Comparator



For:
 $V_{IN(+)} > V_{IN(-)}$
 Output = FF_{HEX}
 For:
 $V_{IN(+)} < V_{IN(-)}$
 Output = 00_{HEX}

046066/1-00

1 mV Resolution with μP Controlled Range

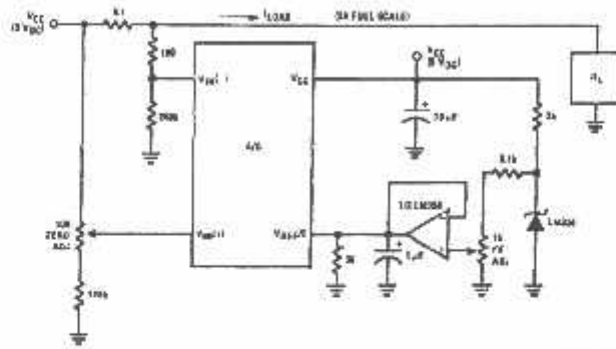


$V_{REF} = 128 \text{ mV}$
 $1 \text{ LSB} = 1 \text{ mV}$
 $V_{DAC} = V_{IN} + V_{DAC} < 256 \text{ mV}$
 $0 < V_{DAC} < 2.5 \text{ V}$

046066/1-00

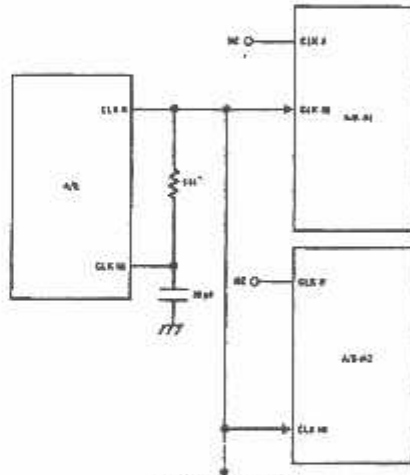
Typical Applications (Continued)

Digitizing a Current Flow



DS00071-02

Self-Clocking Multiple A/Ds

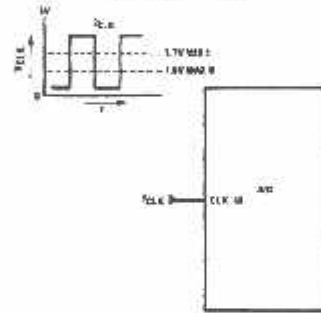


* IF MORE THAN 2 ADDITIONAL A/Ds ARE A DRIVE BUFFER IS REQUIRED

DS00071-03

* Use a large R value to reduce loading at CLK R output.

External Clocking

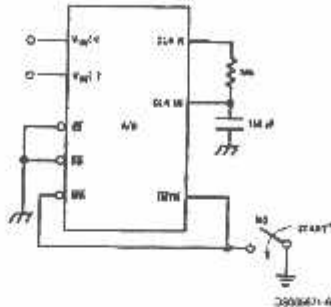


100 kHz f_{CLK} \leq 1460 kHz

DS00071-04

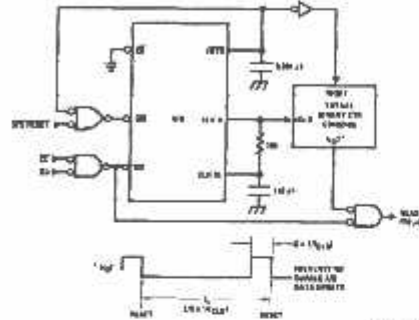
Typical Applications (Continued)

Self-Clocking in Free-Running Mode



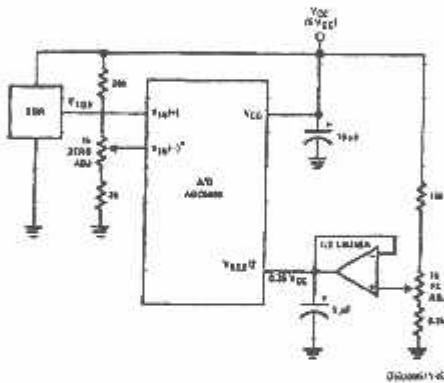
*After power-up a momentary grounding of the WE input is needed to guarantee operation.

µP Interface for Free-Running A/D



D0000671-06

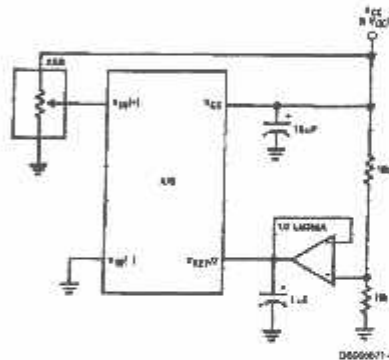
Operating with "Automotive" Ratio-metric Transducers



D0000671-07

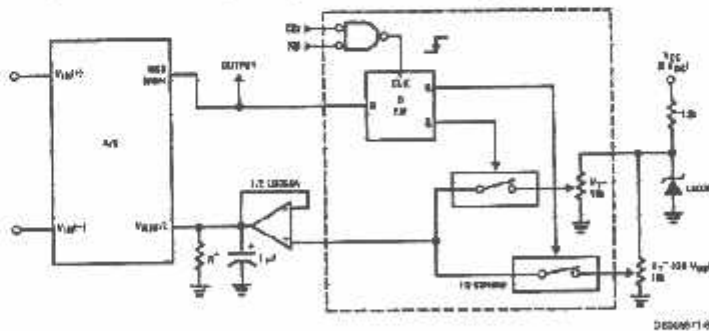
* $V_{ref-} = 0.15 V_{CC}$
 15% of V_{CC} / $V_{ref+} = 98.5\%$ of V_{CC}

Ratio-metric with $V_{REFP}/2$ Forced



D0000671-08

µP Compatible Differential-Input Comparator with Pre-Set V_{OS} (with or without Hysteresis)

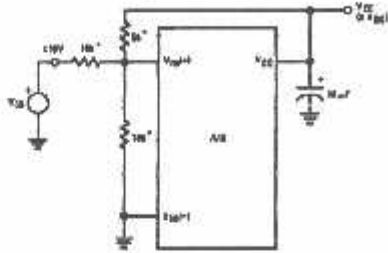


D0000671-09

*See Figure 5 to select R value
 $DB7 = 1$ for $V_{IN(+)} > V_{IN(-)} + V_{REFP}/2$
 Omit circuitry within the dotted area if hysteresis is not needed

Typical Applications (Continued)

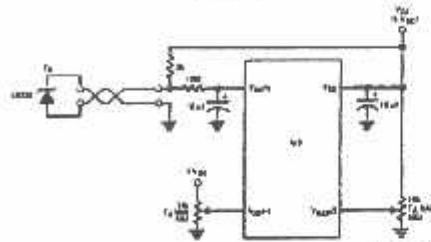
Handling $\pm 10V$ Analog Inputs



28006871-30

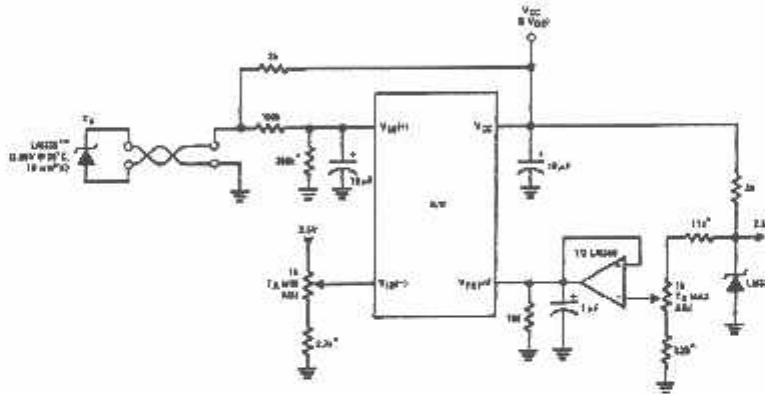
*Beckman Instruments A694-3-R10K resistor array

Low-Cost, μP Interfaced, Temperature-to-Digital Converter



72000601-21

μP Interfaced Temperature-to-Digital Converter



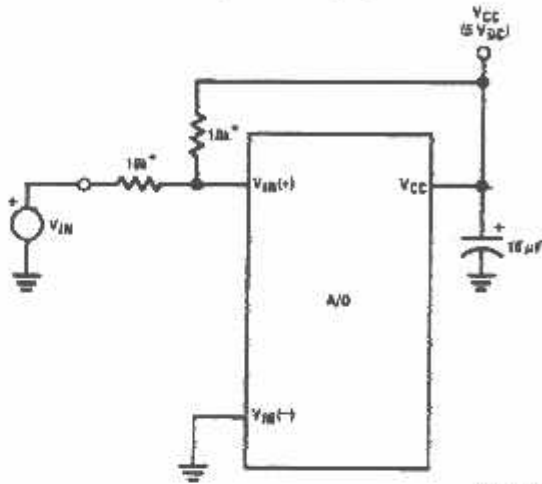
12000611-22

*Circuit values shown are for $0^{\circ}C \leq T_A \leq 125^{\circ}C$

**Can calibrate each sensor to allow easy replacement, then A/D can be calibrated with a pre-set input voltage.

Typical Applications (Continued)

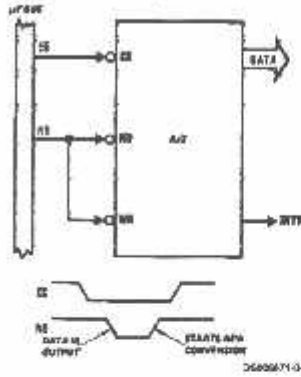
Handling $\pm 5V$ Analog Inputs



*Sackman Instruments #694-3-R150K resistor array

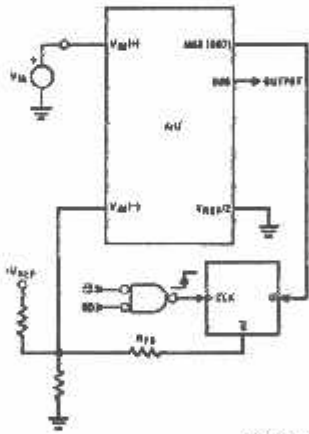
DS000671-22

Read-Only Interface



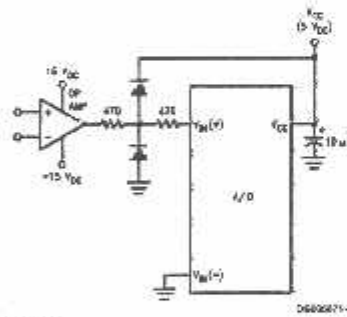
DS000671-24

μP Interfaced Comparator with Hysteresis



DS000671-26

Protecting the Input

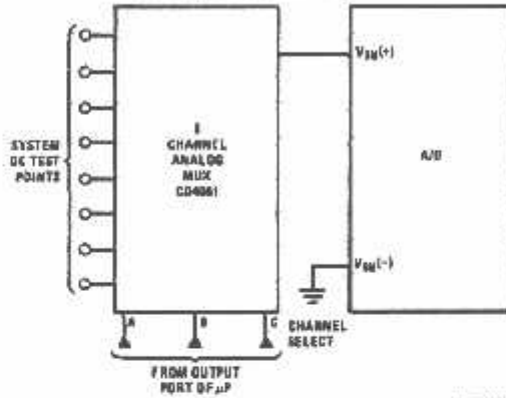


Codes are 1N914

DS000671-0

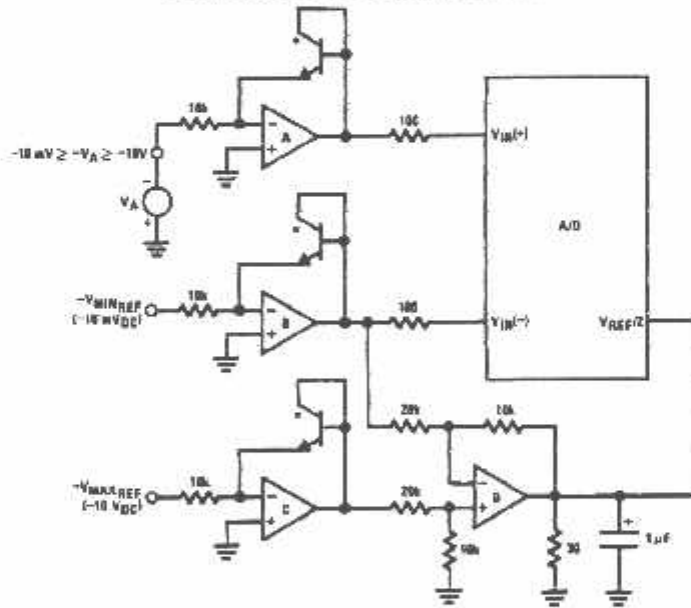
Typical Applications (Continued)

Analog Self-Test for a System



000061-01

A Low-Cost, 3-Decade Logarithmic Converter

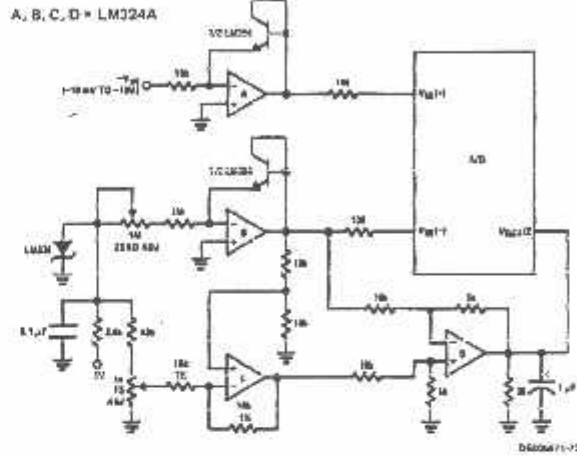


000061-02

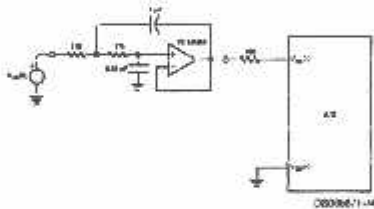
LM309 transistors
A, B, C, D = LM324A quad op amp

Typical Applications (Continued)

3-Decade Logarithmic A/D Converter

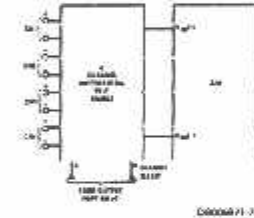


Noise Filtering the Analog Input

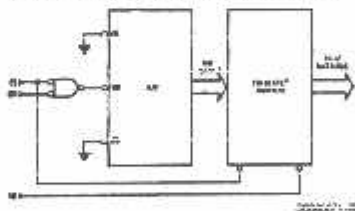


$f_c = 20 \text{ Hz}$
 Uses Chebyshev implementation for steeper roll-off unity gain, 2nd order, low-pass filter.
 Adding a separate filter for each channel increases system response time if an analog multiplexer is used.

Multiplexing Differential Inputs

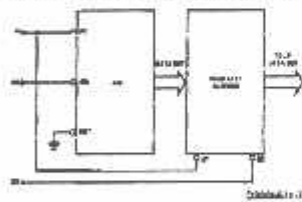


Output Buffers with A/D Data Enabled



*A/D output data is updated 1 CLK period prior to assertion of $\overline{\text{INTR}}$

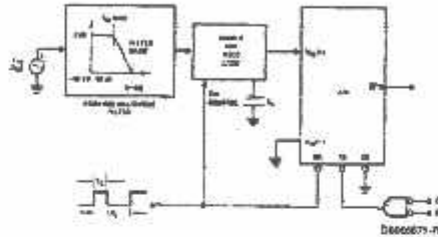
Increasing Bus Drive and/or Reducing Time on Bus



*Allows output data to set-up at falling edge of CS

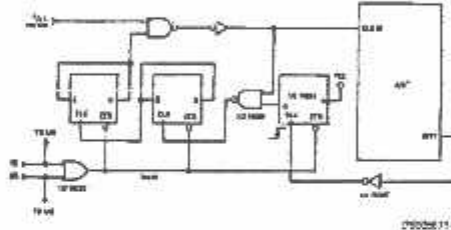
Typical Applications (Continued)

Sampling an AC Input Signal



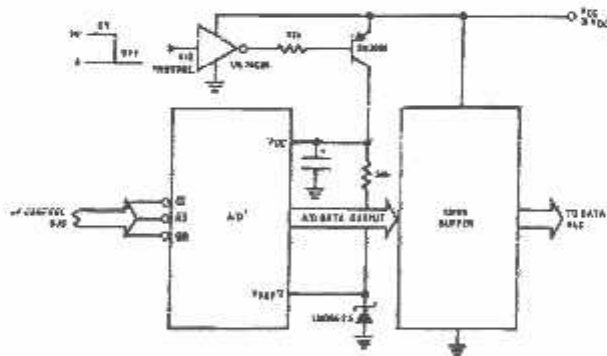
Note 11: Oversample whenever possible (keep $f_s \gg 20f_{in}$) to minimize input frequency loading (aliasing) and to allow for the slow response of the filter.
 Note 12: Consider the amplitude errors which are introduced within the passband of the filter.

70% Power Savings by Clock Gating



(Complete shutdown takes = 30 seconds.)

Power Savings by A/D and V_{REF} Shutdown



*Use ADC0801, 02, 03 or 05 for lowest power consumption.
 Note: Logic inputs can be driven to V_{CC} with A/D supply at zero volts.
 Buffer prevents data bus from overdriving output of A/D when in shutdown mode.

Functional Description

1.5 UNDERSTANDING A/D ERROR SPECS

A perfect A/D transfer characteristic (staircase waveform) is shown in Figure 1. The horizontal scale is analog input voltage and the particular points labeled are in steps of 1 LSB (19.53 mV with 2.5V tied to the $V_{REF}/2$ pin). The digital output codes that correspond to these inputs are shown as D=1, D, and D+1. For the perfect A/D, not only will center-value

(A-1, A, A+1,) analog inputs produce the correct output digital codes, but also each riser (the transitions between adjacent output codes) will be located $\pm 1/2$ LSB away from each center-value. As shown, the risers are ideal and have no width. Correct digital output codes will be provided for a range of analog input voltages that extend $\pm 1/2$ LSB from the ideal center-values. Each tread (the range of analog input voltage that provides the same digital output code) is therefore 1 LSB wide.

Functional Description (Continued)

Figure 2 shows a worst case error plot for the ADC0801. All center-valued inputs are guaranteed to produce the correct output codes and the adjacent rises are guaranteed to be no closer to the center-value points than $\pm 1/4$ LSB. In other words, if we apply an analog input equal to the center-value $\pm 1/4$ LSB, we guarantee that the A/D will produce the correct digital code. The maximum range of the position of the code transition is indicated by the horizontal arrow and it is guaranteed to be no more than $1/4$ LSB.

The error curve of Figure 3 shows a worst case error plot for the ADC0802. Here we guarantee that if we apply an analog input equal to the LSB analog voltage center-value the A/D will produce the correct digital code.

Next to each transfer function is shown the corresponding error plot. Many people may be more familiar with error plots than transfer functions. The analog input voltage to the A/D is provided by either a linear ramp or by the discrete output steps of a high resolution DAC. Notice that the error is continuously displayed and includes the quantization uncertainty of the A/D. For example the error at point 1 of Figure 1 is $+1/2$ LSB because the digital code appeared $1/2$ LSB in advance of the center-value of the trend. The error plots always have a constant negative slope and the abrupt upside steps are always 1 LSB in magnitude.

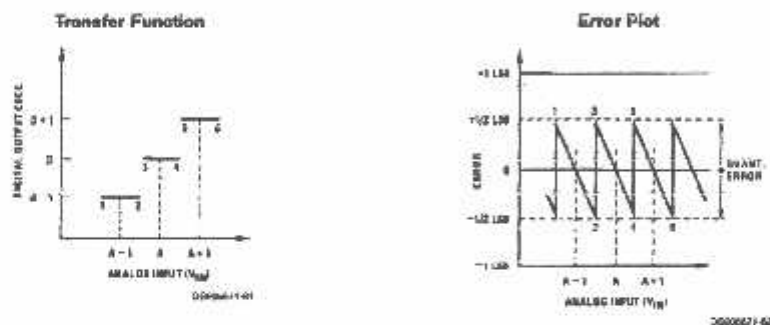


FIGURE 1. Clarifying the Error Specs of an A/D Converter
Accuracy = ± 0 LSB: A Perfect A/D

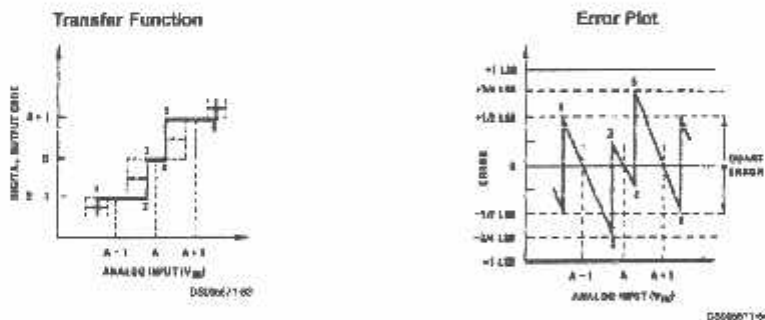


FIGURE 2. Clarifying the Error Specs of an A/D Converter
Accuracy = $\pm 1/4$ LSB

Functional Description (Continued)

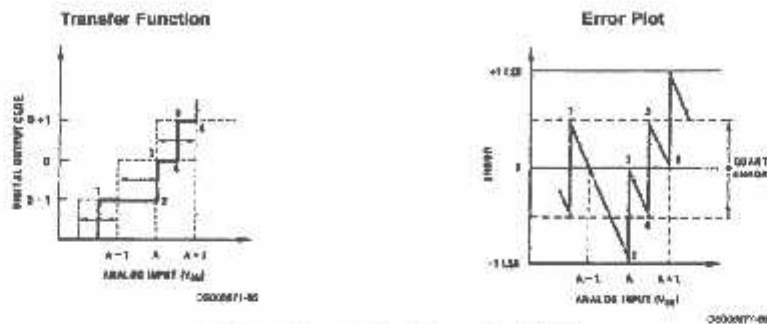


FIGURE 3. Clarifying the Error Specs of an A/D Converter
Accuracy = $\pm 1/2$ LSB

2.0 FUNCTIONAL DESCRIPTION

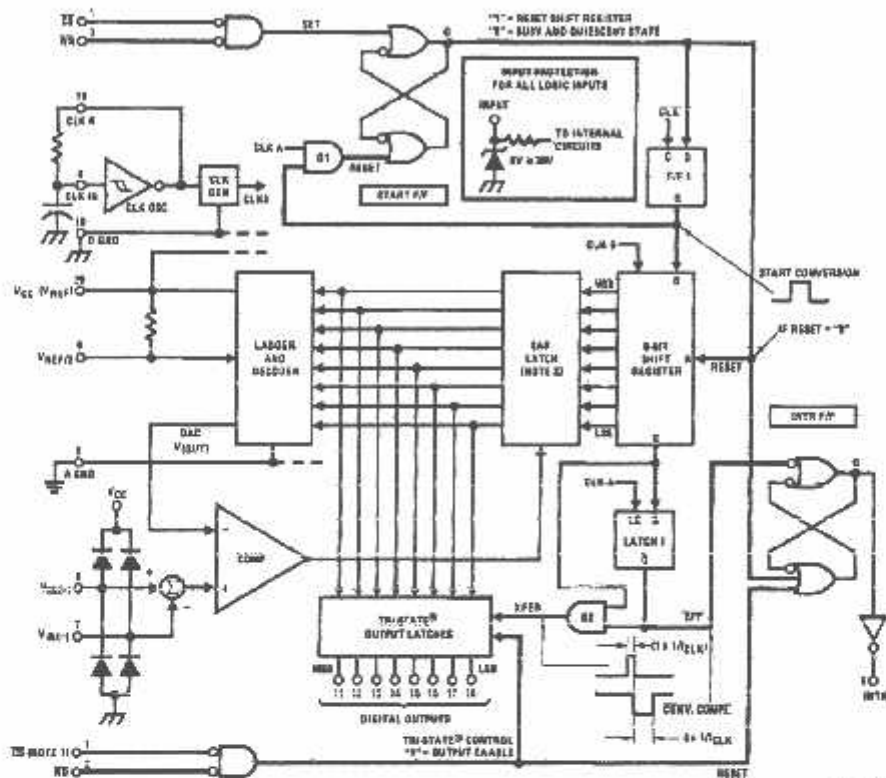
The ADC0801 series contains a circuit equivalent of the 256R network. Analog switches are sequenced by successive approximation logic to match the analog difference input voltage [$V_{IN}(+) - V_{IN}(-)$] to a corresponding tap on the R network. The most significant bit is tested first and after 8 comparisons (64 clock cycles) a digital 8-bit binary code (1111 1111 = full-scale) is transferred to an output latch and then an interrupt is asserted (INTR makes a high-to-low transition). A conversion in process can be interrupted by issuing a second start command. The device may be operated in the free-running mode by connecting INTR to the WR input with $\overline{CS} = 0$. To ensure start-up under all possible conditions, an external WR pulse is required during the first power-up cycle.

On the high-to-low transition of the \overline{WR} input the internal SAR latches and the shift register stages are reset. As long as the \overline{CS} input and \overline{WR} input remain low, the A/D will remain in a reset state. Conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low-to-high transition.

A functional diagram of the A/D converter is shown in Figure 4. All of the package pinouts are shown and the major logic control paths are drawn in heavier weight lines.

The converter is started by having \overline{CS} and \overline{WR} simultaneously low. This sets the start flip-flop (F/F) and the resulting "1" level resets the 8-bit shift register, reloads the interrupt (INTR) F/F and inputs a "1" to the D flop, F/F1, which is at the input end of the 8-bit shift register. Internal clock signals then transfer this "1" to the Q output of F/F1. The AND gate, G1, combines this "1" output with a clock signal to provide a reset signal to the start F/F. If the set signal is no longer present (either \overline{WR} or \overline{CS} is a "1") the start F/F is reset and the 8-bit shift register then can have the "1" clocked in, which starts the conversion process. If the set signal were to still be present, this reset pulse would have no effect (both outputs of the start F/F would momentarily be at a "1" level) and the 8-bit shift register would continue to be held in the reset mode. This logic therefore allows for wide \overline{CS} and \overline{WR} signals and the converter will start after at least one of these signals returns high and the internal clocks again provide a reset signal for the start F/F.

Functional Description (Continued)



Note 13: CS shown twice for clarity.

Note 14: SAR = Successive Approximation Register.

FIGURE 4. Block Diagram

After the "1" is docked through the 8-bit shift register (which completes the SAR search) it appears as the input to the D-type latch, LATCH 1. As soon as this "1" is output from the shift register, the AND gate, G2, causes the new digital word to transfer to the TRI-STATE output latches. When LATCH 1 is subsequently enabled, the Q output makes a high-to-low transition which causes the INTR F/F to set. An inverting buffer then supplies the INTR input signal.

Note that this SET control of the INTR F/F remains low for 8 of the external clock periods (as the internal clocks run at 1/4 of the frequency of the external clock). If the data output is continuously enabled (CS and RD both held low), the INTR output will still signal the end of conversion (by a high-to-low transition), because the SET input can control the Q output of the INTR F/F even though the RESET input is constantly at a "1" level in this operating mode. This INTR output will therefore stay low for the duration of the SET signal, which is 8 periods of the external clock frequency (assuming the A/D is not started during this interval).

When operating in the free-running or continuous conversion mode (INTR pin tied to WR and CS wired low — see also section 2.8), the START F/F is SET by the high-to-low transition of the INTR signal. This resets the SHIFT REGISTER

which causes the input to the D-type latch, LATCH 1, to go low. As the latch enable input is still present, the Q output will go high, which then allows the INTR F/F to be RESET. This reduces the width of the resulting INTR output pulse to only a few propagation delays (approximately 300 ns).

When data is to be read, the combination of both CS and RD being low will cause the INTR F/F to be RESET and the TRI-STATE output latches will be enabled to provide the 8-bit digital outputs.

2.1 Digital Control Inputs

The digital control inputs (CS, RD, and WR) meet standard TTL logic voltage levels. These signals have been renamed when compared to the standard A/D Start and Output Enable labels. In addition, these inputs are active low to allow an easy interface to microprocessor control buses. For non-microprocessor based applications, the CS input (pin 1) can be grounded and the standard A/D Start function is obtained by an active low pulse applied at the WR input (pin 3) and the Output Enable function is caused by an active low pulse at the RD input (pin 2).

Functional Description (Continued)

2.2 Analog Differential Voltage Inputs and Common-Mode Rejection

This A/D has additional applications flexibility due to the analog differential voltage input. The $V_{IN(-)}$ input (pin 7) can be used to automatically subtract a fixed voltage value from the input reading (tare correction). This is also useful in 4 mA, 20 mA current loop conversion. In addition, common-mode noise can be reduced by use of the differential input.

The time interval between sampling $V_{IN(+)}$ and $V_{IN(-)}$ is 4-1/2 clock periods. The maximum error voltage due to this slight time difference between the input voltage samples is given by:

$$\Delta V_{e}(\text{MAX}) = (V_P) (2\pi f_{cm}) \left(\frac{4.5}{f_{\text{CLK}}} \right)$$

where:

ΔV_e is the error voltage due to sampling delay

V_P is the peak value of the common-mode voltage

f_{cm} is the common-mode frequency

As an example, to keep this error to 1/4 LSB (-5 mV) when operating with a 80 Hz common-mode frequency, f_{cm} , and using a 640 kHz A/D clock, f_{CLK} , would allow a peak value of the common-mode voltage, V_P , which is given by:

$$V_P = \frac{(\Delta V_e(\text{MAX})) (f_{\text{CLK}})}{(2\pi f_{cm}) (4.5)}$$

or

$$V_P = \frac{(5 \times 10^{-3}) (640 \times 10^3)}{(6.28) (80) (4.5)}$$

which gives

$$V_P \approx 1.9V.$$

The allowed range of analog input voltages usually places more severe restrictions on input common-mode noise levels.

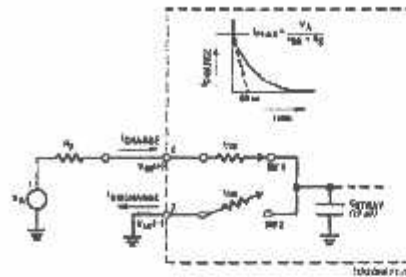
An analog input voltage with a reduced span and a relatively large zero offset can be handled easily by making use of the differential input (see section 2.4 Reference Voltage).

2.3 Analog Inputs

2.3.1 Input Current

Normal Mode

Due to the internal switching action, displacement currents will flow at the analog inputs. This is due to on-chip stray capacitance to ground as shown in Figure 5.



$$r_{\text{ON}} \text{ of SW 1 and SW 2} = 5 \text{ k}\Omega$$

$$C_{\text{STRAY}} = 5 \text{ nF} \times 12 \text{ pF} = 60 \text{ pF}$$

FIGURE 5. Analog Input Impedance

The voltage on this capacitance is switched and will result in currents entering the $V_{IN(+)}$ input pin and leaving the $V_{IN(-)}$ input which will depend on the analog differential input voltage levels. These current transients occur at the leading edge of the internal clocks. They rapidly decay and do not cause errors as the on-chip comparator is strobed at the end of the clock period.

Fault Mode

If the voltage source applied to the $V_{IN(+)}$ or $V_{IN(-)}$ pin exceeds the allowed operating range of $V_{CC} \pm 50$ mV, large input currents can flow through a parasitic diode to the V_{CC} pin. If these currents can exceed the 1 mA max allowed spec, an external diode (1N914) should be added to bypass this current to the V_{CC} pin (with the current bypassed with this diode, the voltage at the $V_{IN(+)}$ pin can exceed the V_{CC} voltage by the forward voltage of this diode).

2.3.2 Input Bypass Capacitors

Bypass capacitors at the inputs will average these charges and cause a DC current to flow through the output resistances of the analog signal sources. This charge pumping action is worse for continuous conversions with the $V_{IN(+)}$ input voltage at full-scale. For continuous conversions with a 640 kHz clock frequency with the $V_{IN(+)}$ input at 5V, this DC current is at a maximum of approximately 5 μ A. Therefore, bypass capacitors should not be used at the analog inputs or the $V_{REF/2}$ pin for high resistance sources ($> 1 \text{ k}\Omega$). If input bypass capacitors are necessary for noise filtering and high source resistance is desirable to minimize capacitor size, the detrimental effects of the voltage drop across this input resistance, which is due to the average value of the input current, can be eliminated with a full-scale adjustment while the given source resistor and input bypass capacitor are both in place. This is possible because the average value of the input current is a precise linear function of the differential input voltage.

2.3.3 Input Source Resistance

Large values of source resistance where an input bypass capacitor is not used, will not cause errors as the input currents settle out prior to the comparison time. If a low pass filter is required in the system, use a low valued series resistor ($\leq 1 \text{ k}\Omega$) for a passive RC section or add an op amp RC active low pass filter. For low source resistance applications, ($\leq 1 \text{ k}\Omega$), a 0.1 μ F bypass capacitor at the inputs will prevent noise pickup due to series lead inductance of a long wire. A

Functional Description (Continued)

100 Ω series resistor can be used to isolate this capacitor—both the R and C are placed outside the feedback loop—from the output of an op amp, if used.

2.3.4 Noise

The loads to the analog inputs (pins 6 and 7) should be kept as short as possible to minimize input noise coupling. Both noise and undesired digital clock coupling to these inputs can cause system errors. The source resistance for these inputs should, in general, be kept below 5 k Ω . Larger values of source resistance can cause undesired system noise pickup. Input bypass capacitors, placed from the analog inputs to ground, will eliminate system noise pickup but can create analog scale errors as these capacitors will average the transient input switching currents of the A/D (see section 2.3.1.). This scale error depends on both a large source resistance and the use of an input bypass capacitor. This error can be eliminated by doing a full-scale adjustment of the A/D (adjust $V_{REF}/2$ for a proper full-scale reading—see section 2.5.2 on Full-Scale Adjustment) with the source resistance and input bypass capacitor in place.

2.4 Reference Voltage

2.4.1 Span Adjust

For maximum applications flexibility, these A/Ds have been designed to accommodate a 5 V_{DC} , 2.5 V_{DC} or an adjusted voltage reference. This has been achieved in the design of the IC as shown in Figure 6.

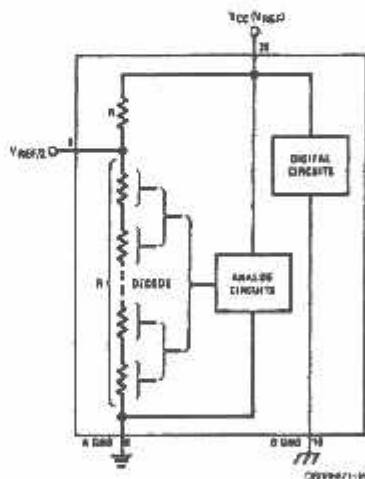


FIGURE 6. The $V_{REFERENCE}$ Design on the IC

Notice that the reference voltage for the IC is either $1/2$ of the voltage applied to the V_{CC} supply pin, or is equal to the voltage that is externally forced at the $V_{REF}/2$ pin. This allows for a ratio-metric voltage reference using the V_{CC} supply, a 5 V_{DC} reference voltage can be used for the V_{CC} supply or a voltage less than 2.5 V_{DC} can be applied to the $V_{REF}/2$ input for increased application flexibility. The internal gain to the $V_{REF}/2$ input is 2, making the full-scale differential input voltage twice the voltage at pin 9.

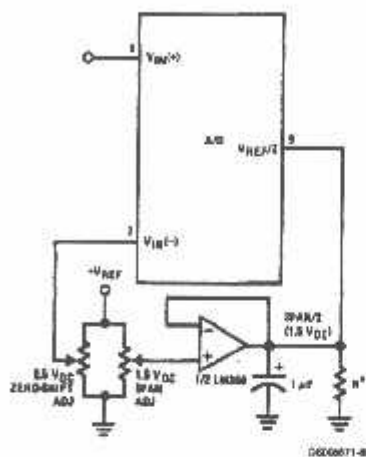
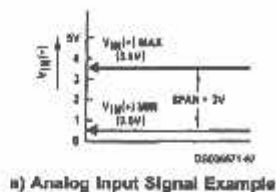
An example of the use of an adjusted reference voltage is to accommodate a reduced span—or dynamic voltage range—of the analog input voltage. If the analog input voltage were to range from 0.5 V_{DC} to 3.5 V_{DC} , instead of 0V to 5 V_{DC} , the span would be 3V as shown in Figure 7. With 3.5 V_{DC} applied to the $V_{IN}(-)$ pin to absorb the offset, the reference voltage can be made equal to $1/2$ of the 3V span or 1.5 V_{DC} . The A/D now will encode the $V_{IN}(+)$ signal from 0.5V to 3.5V with the 0.5V input corresponding to zero and the 3.5 V_{DC} input corresponding to full-scale. The full 8 bits of resolution are therefore applied over this reduced analog input voltage range.

2.4.2 Reference Accuracy Requirements

The converter can be operated in a ratio-metric mode or an absolute mode. In ratio-metric converter applications, the magnitude of the reference voltage is a factor in both the output of the source transducer and the output of the A/D converter and therefore cancels out in the final digital output code. The ADC0805 is specified particularly for use in ratio-metric applications with no adjustments required. In absolute conversion applications, both the initial value and the temperature stability of the reference voltage are important factors in the accuracy of the A/D converter. For $V_{REF}/2$ voltages of 2.4 V_{DC} nominal value, initial errors of ± 10 mV $_{DC}$ will cause conversion errors of ± 1 LSB due to the gain of 2 of the $V_{REF}/2$ input. In reduced span applications, the initial value and the stability of the $V_{REF}/2$ input voltage become even more important. For example, if the span is reduced to 2.5V, the analog input LSB voltage value is correspondingly reduced from 20 mV (5V span) to 10 mV and 1 LSB at the $V_{REF}/2$ input becomes 5 mV. As can be seen, this reduces the allowed initial tolerance of the reference voltage and requires correspondingly less absolute change with temperature variations. Note that spans smaller than 2.5V place even tighter requirements on the initial accuracy and stability of the reference source.

In general, the magnitude of the reference voltage will require an initial adjustment. Errors due to an improper value of reference voltage appear as full-scale errors in the A/D transfer function. IC voltage regulators may be used for references if the ambient temperature changes are not excessive. The LM336B 2.5V IC reference diode (from National Semiconductor) has a temperature stability of 1.8 mV typ (8 mV max) over 0°C to $T_{AS}+70^{\circ}\text{C}$. Other temperature range parts are also available.

Functional Description (Continued)



*Add 1 $V_{REF}/2 < 1 V_{CC}$ with LM358 to draw 3 mA to ground.

FIGURE 7. Adapting the A/D Analog Input Voltages to Match an Arbitrary Input Signal Range

2.5 Errors and Reference Voltage Adjustments

2.5.1 Zero Error

The zero of the A/D does not require adjustment. If the minimum analog input voltage value, $V_{IN(MIN)}$, is not ground, a zero offset can be done. The converter can be made to output 0000 0000 digital code for this minimum input voltage by biasing the A/D $V_{IN}(-)$ input at this $V_{IN(MIN)}$ value (see Applications section). This utilizes the differential mode operation of the A/D.

The zero error of the A/D converter relates to the location of the first riser of the transfer function and can be measured by grounding the $V_{IN}(-)$ input and applying a small magnitude positive voltage to the $V_{IN}(+)$ input. Zero error is the difference between the actual DC input voltage that is necessary to just cause an output digital code transition from 0000 0000 to 0000 0001 and the ideal $1/2$ LSB value ($1/2$ LSB = 9.8 mV for $V_{REF}/2=2.500 V_{DC}$).

2.5.2 Full-Scale

The full-scale adjustment can be made by applying a differential input voltage that is $1 1/2$ LSB less than the desired analog full-scale voltage range and then adjusting the magnitude of the $V_{REF}/2$ input (pin 9 or the V_{CC} supply if pin 9 is not used) for a digital output code that is just changing from 1111 1110 to 1111 1111.

2.5.3 Adjusting for an Arbitrary Analog Input Voltage Range

If the analog zero voltage of the A/D is shifted away from ground (for example, to accommodate an analog input signal that does not go to ground) this new zero reference should be properly adjusted first. A $V_{IN}(+)$ voltage that equals this desired zero reference plus $1/2$ LSB (where the LSB is calculated for the desired analog span, 1 LSB = analog span/256)

is applied to pin 6 and the zero reference voltage at pin 7 should then be adjusted to just obtain the 00_{HEX} to 01_{HEX} code transition.

The full-scale adjustment should then be made (with the proper $V_{IN}(-)$ voltage applied) by forcing a voltage to the $V_{IN}(+)$ input which is given by:

$$V_{IN}(+) \text{ is adj} = V_{MAX} - 1.5 \left[\frac{(V_{MAX} - V_{MIN})}{256} \right]$$

where:

V_{MAX} = The high end of the analog input range

and

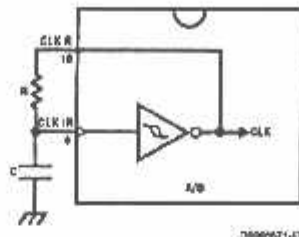
V_{MIN} = the low end (the offset zero) of the analog range. (Both are ground referenced.)

The $V_{REF}/2$ (or V_{CC}) voltage is then adjusted to provide a code change from FE_{HEX} to FF_{HEX}. This completes the adjustment procedure.

2.6 Clocking Option

The clock for the A/D can be derived from the CPU clock or an external RC can be added to provide self-clocking. The CLK IN (pin 4) makes use of a Schmitt trigger as shown in Figure 8.

Functional Description (Continued)



$$f_{CLK} = \frac{1}{1.1 RC}$$
$$R = 10 \text{ k}\Omega$$

FIGURE 8. Self-Clocking the A/D

Heavy capacitive or DC loading of the clock R pin should be avoided as this will disturb normal converter operation. Loads less than 50 pF, such as driving up to 7 A/D converter clock inputs from a single clock R pin of 1 converter, are allowed. For larger clock line loading, a CMOS or low power TTL buffer or PNP input logic should be used to minimize the loading on the clock R pin (do not use a standard TTL buffer).

2.7 Restart During a Conversion

If the A/D is restarted (\overline{CS} and \overline{WR} go low and return high) during a conversion, the converter is reset and a new conversion is started. The output data latch is not updated if the conversion in process is not allowed to be completed, therefore the data of the previous conversion remains in this latch. The \overline{INTR} output simply remains at the "1" level.

2.8 Continuous Conversions

For operation in the free-running mode an initializing pulse should be used, following power-up, to ensure circuit operation. In this application, the \overline{CS} input is grounded and the \overline{WR} input is tied to the \overline{INTR} output. This \overline{WR} and \overline{INTR} node should be momentarily forced to logic low following a power-up cycle to guarantee operation.

2.9 Driving the Data Bus

This MOS A/D, like MOS microprocessors and memories, will require a bus driver when the total capacitance of the data bus gets large. Other circuitry, which is tied to the data bus, will add to the total capacitive loading, even in TRI-STATE (high impedance mode). Backplane bussing also greatly adds to the stray capacitance of the data bus.

There are some alternatives available to the designer to handle this problem. Basically, the capacitive loading of the data bus slows down the response time, even though DC specifications are still met. For systems operating with a relatively slow CPU clock frequency, more time is available in which to establish proper logic levels on the bus and therefore higher capacitive loads can be driven (see typical characteristics curves).

At higher CPU clock frequencies time can be extended for I/O reads (and/or writes) by inserting wait states (8080) or using clock extending circuits (6800).

Finally, if time is short and capacitive loading is high, external bus drivers must be used. These can be TRI-STATE buffers

(low power Schottky such as the DM74LS240 series is recommended) or special higher drive current products which are designed as bus drivers. High current bipolar bus drivers with PNP inputs are recommended.

2.10 Power Supplies

Noise spikes on the V_{CC} supply line can cause conversion errors as the comparator will respond to this noise. A low inductance tantalum filter capacitor should be used close to the converter V_{CC} pin and values of 1 μ F or greater are recommended. If an unregulated voltage is available in the system, a separate LM340LAZ-5.0, TO-92, 5V voltage regulator for the converter (and other analog circuitry) will greatly reduce digital noise on the V_{CC} supply.

2.11 Wiring and Hook-Up Precautions

Standard digital wire wrap sockets are not satisfactory for breadboarding this A/D converter. Sockets on PC boards can be used and all logic signal wires and leads should be grouped and kept as far away as possible from the analog signal leads. Exposed leads to the analog inputs can cause undesired digital noise and hum pickup, therefore shielded leads may be necessary in many applications.

A single point analog ground that is separate from the logic ground points should be used. The power supply bypass capacitor and the self-clocking capacitor (if used) should both be returned to digital ground. Any $V_{REF}/2$ bypass capacitors, analog input filter capacitors, or input signal shielding should be returned to the analog ground point. A test for proper grounding is to measure the zero error of the A/D converter. Zero errors in excess of $1/4$ LSB can usually be traced to improper board layout and wiring (see section 2.5.1 for measuring the zero error).

3.0 TESTING THE A/D CONVERTER

There are many degrees of complexity associated with testing an A/D converter. One of the simplest tests is to apply a known analog input voltage to the converter and use LEDs to display the resulting digital output code as shown in Figure 9.

For ease of testing, the $V_{REF}/2$ (pin 9) should be supplied with $2.560 V_{DC}$ and a V_{CC} supply voltage of $5.12 V_{DC}$ should be used. This provides an LSB value of 20 mV.

If a full-scale adjustment is to be made, an analog input voltage of $5.090 V_{DC}$ ($5.120 - 1/2$ LSB) should be applied to the $V_{IN}(+)$ pin with the $V_{IN}(-)$ pin grounded. The value of the $V_{REF}/2$ input voltage should then be adjusted until the digital output code is just changing from 1111 1110 to 1111 1111. This value of $V_{REF}/2$ should then be used for all the tests.

The digital output LED display can be decoded by dividing the 8 bits into 2 hex characters, the 4 most significant (MS) and the 4 least significant (LS). Table 1 shows the fractional binary equivalent of these two 4-bit groups. By adding the voltages obtained from the "VMS" and "VLS" columns in Table 1, the nominal value of the digital display (when $V_{REF}/2 = 2.560V$) can be determined. For example, for an output LED display of 1011 0110 or B6 (in hex), the voltage values from the table are $3.520 + 0.120$ or $3.640 V_{DC}$. These voltage values represent the center-values of a perfect A/D converter. The effects of quantization error have to be accounted for in the interpretation of the test results.

Functional Description (Continued)

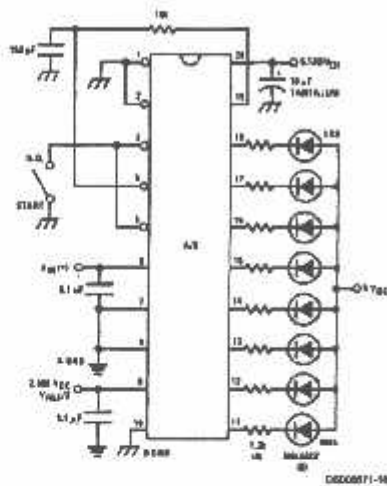


FIGURE 9. Basic A/D Tester

For a higher speed test system, or to obtain plotted data, a digital-to-analog converter is needed for the test set-up. An accurate 10-bit DAC can serve as the precision voltage source for the A/D. Errors of the A/D under test can be expressed as either analog voltages or differences in 2 digital words.

A basic A/D tester that uses a DAC and provides the error as an analog output voltage is shown in Figure 8. The 20 μ mps can be eliminated if a lab DVM with a numerical subtraction feature is available to read the difference voltage, "A-C", directly. The analog input voltage can be supplied by a low frequency ramp generator and an X-Y plotter can be used to provide analog error (Y axis) versus analog input (X axis).

For operation with a microprocessor or a computer-based test system, it is more convenient to present the errors digitally. This can be done with the circuit of Figure 11, where the output code transitions can be detected as the 10-bit DAC is incremented. This provides $\frac{1}{4}$ LSB steps for the 8-bit A/D under test. If the results of this test are automatically plotted with the analog input on the X axis and the error (in LSB's) as the Y axis, a useful transfer function of the A/D under test results. For acceptance testing, the plot is not necessary and the testing speed can be increased by establishing internal limits on the allowed error for each code.

4.0 MICROPROCESSOR INTERFACING

To discuss the interface with 8080A and 6800 microprocessors, a common sample subroutine structure is used. The microprocessor starts the A/D, reads and stores the results of 16 successive conversions, then returns to the user's program. The 16 data bytes are stored in 16 successive memory locations. All Data and Addresses will be given in hexadecimal form. Software and hardware details are provided separately for each type of microprocessor.

4.1 Interfacing 8080 Microprocessor Derivatives (8048, 8085)

This converter has been designed to directly interface with derivatives of the 8080 microprocessor. The A/D can be mapped into memory space (using standard memory address decoding for \overline{CS} and the \overline{MEMR} and \overline{MEMW} strobes) or it can be controlled as an I/O device by using the $\overline{IO R}$ and $\overline{IO W}$ strobes and decoding the address bits $A0 \rightarrow A7$ (or address bits $A8 \rightarrow A15$ as they will contain the same 8-bit address information) to obtain the \overline{CS} input. Using the I/O space provides 256 additional addresses and may allow a simpler 8-bit address decoder but the data can only be input to the accumulator. To make use of the additional memory reference instructions, the A/D should be mapped into memory space. An example of an A/D in I/O space is shown in Figure 12.

Functional Description (Continued)

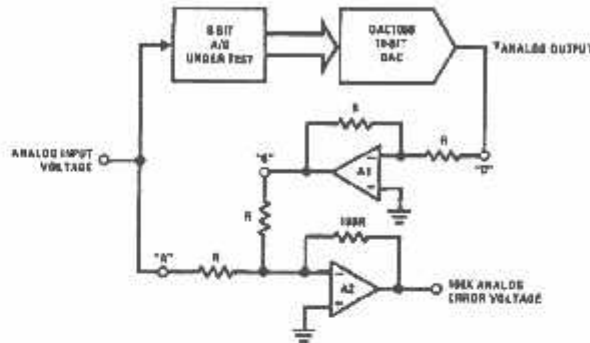


FIGURE 10. A/D Tester with Analog Error Output



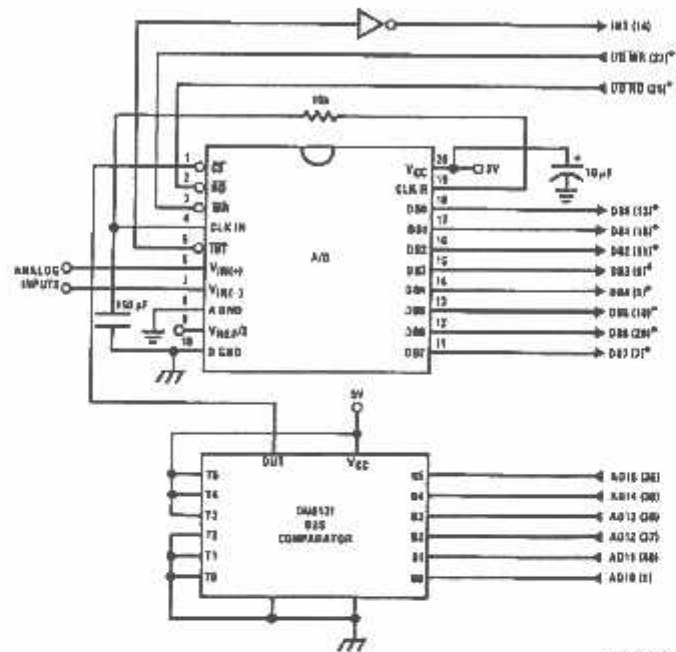
FIGURE 11. Basic "Digital" A/D Tester

TABLE 1. DECODING THE DIGITAL OUTPUT LEDs

HEX	BINARY	FRACTIONAL BINARY VALUE FOR		OUTPUT VOLTAGE CENTER VALUES WITH $V_{REF}/2 = 2.560 V_{DC}$	
		MS GROUP	LS GROUP	VMS GROUP (Note 15)	VLS GROUP (Note 15)
F	1 1 1 1	15/16	15/256	4.800	0.300
E	1 1 1 0	7/8	7/128	4.480	0.280
D	1 1 0 1	13/16	13/256	4.160	0.260
C	1 1 0 0	3/4	3/64	3.840	0.240
B	1 0 1 1	11/16	11/256	3.520	0.220
A	1 0 1 0	5/8	5/128	3.200	0.200
9	1 0 0 1	9/16	9/256	2.880	0.180
8	1 0 0 0	1/2	1/32	2.560	0.160
7	0 1 1 1	7/16	7/256	2.240	0.140
6	0 1 1 0	3/8	3/128	1.920	0.120
5	0 1 0 1	5/16	2/256	1.600	0.100
4	0 1 0 0	1/4	1/64	1.280	0.080
3	0 0 1 1	3/16	3/256	0.960	0.060
2	0 0 1 0	1/8	1/128	0.640	0.040
1	0 0 0 1	1/16	1/256	0.320	0.020
0	0 0 0 0			0	0

Note 15: Display Output = VMS Group + VLS Group

Functional Description (Continued)



Note 16: *Pin numbers for the DP6228 system controller, others are IN58C00A.
 Note 17: Pin 23 of the IN588228 must be tied to -12V through a 1 kΩ resistor to generate the R6T 7 instruction when an interrupt is acknowledged as required by the accompanying sample program.

FIGURE 12. ADC0801_INS8080A CPU Interface

0000071-00

Functional Description (Continued)

SAMPLE PROGRAM FOR Figure 12 ADC0801-INS8080A CPU INTERFACE

```

0038  C5 00 03  RST 7:      JMP    LD DATA
      *      *
      *      *
0100  21 00 02  START:      LXI H 0200H      ; HL pair will point to
                                ; data storage locations
0103  31 00 04  RETURN:      LXI SP 0400H      ; Initialize stack pointer (Note 1)
0106  7D                    MOV A, L        ; Test # of bytes entered
0107  FE 0F                    CPI 0FH        ; If # = 16, JMP to
0109  CA 13 01                JZ CONT      ; user program
010C  D3 E0                    OUT 80H        ; Start A/D
010E  FB                    EI            ; Enable interrupt
010F  00                    NOP          ; Loop until end of
0110  C3 DF 01                JMP LOOP     ; conversion
0113  *                      *
      *                      *
      *                      *
      *                      *
      *                      *
      *                      *
      *                      *
      *                      *
0300  DB E0  LD DATA:      IN 80H        ; Load data into accumulator
0302  77                    MOV M, A      ; Store data
0303  23                    INX H          ; Increment storage pointer
0304  C5 03 01                JMP RETURN

```

000001-06

Note 1B: The stack pointer must be dimensioned because a RST 7 instruction pushes the PC onto the stack.

Note 1C: All address used were arbitrarily chosen.

The standard control bus signals of the 8080 (\overline{CS} , \overline{RD} and \overline{WR}) can be directly wired to the digital control inputs of the A/D and the bus timing requirements are met to allow both starting the converter and outputting the data onto the data bus. A bus driver should be used for larger microprocessor systems where the data bus leaves the PC board and/or must drive capacitive loads larger than 100 pF.

4.1.1 Sample 8080A CPU Interfacing Circuitry and Program

The following sample program and associated hardware shown in Figure 12 may be used to input data from the converter to the INS8080A CPU chip set (comprised of the INS8080A microprocessor, the INS8228 system controller and the INS8224 clock generator). For simplicity, the A/D is controlled as an I/O device, specifically an 8-bit bi-directional port located at an arbitrarily chosen port address, E0. The TRI-STATE output capability of the A/D eliminates the need for a peripheral interface device, however address decoding is still required to generate the appropriate \overline{CS} for the converter.

It is important to note that in systems where the A/D converter is 1-of-8 or less I/O mapped devices, no address decoding circuitry is necessary. Each of the 8 address bits (A0 to A7) can be directly used as \overline{CS} inputs—one for each I/O device.

4.1.2 INS8048 interface

The INS8048 interface technique with the ADC0801 series (see Figure 13) is simpler than the 8080A CPU interface. There are 24 I/O lines and three test input lines in the 8048. With these extra I/O lines available, one of the I/O lines (bit 0 of port 1) is used as the chip select signal to the A/D, thus eliminating the use of an external address decoder. Bus control signals \overline{RD} , \overline{WR} and INT of the 8048 are tied directly to the A/D. The 16 converted data words are stored at on-chip RAM locations from 20 to 2F (Hex). The \overline{RD} and \overline{WR} signals are generated by reading from and writing into a dummy address, respectively. A sample interface program is shown below.

Functional Description (Continued)

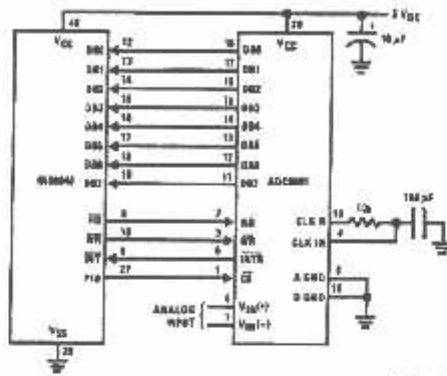


FIGURE 13. INS8048 Interface

SAMPLE PROGRAM FOR Figure 13 INS8048 INTERFACE

```

04 10          JMP      10H          ; Program starts at addr 10
          ORG      10H
04 50          JMP      60H          ; Interrupt jump vector
          ORG      60H
99 FE          ANL      P1, #0FEH   ; Main program
          ANL      P1, #0FEH   ; Chip select
81             MOVX     A, @R1      ; Read in the 1st data
          ; to reset the intr
89 01          START:  ORL      P1, #1    ; Set port pin high
88 20          MOV      RD, #20H    ; Data address
89 FF          MOV      RL, #0FFH   ; Dummy address
8A 10          MOV      R2, #10H    ; Counter for 16 bytes
83 FF          AGAIN:  MOV      A, #0FFH ; Set ACC for intr loop
99 FE          ANL      P1, #0FEH   ; Send CS (bit 0 of P1)
91            MOVX     @R1, A      ; Send WR out
05            EM       I           ; Enable interrupt
98 21          LOOP:   JNZ      LOOP  ; Wait for interrupt
8A 1B          DJNZ    R2, AGAIN    ; If 16 bytes are read
00            NOP
00            NOP
          ORG      90H
81            INDATA: MOVX     A, @R1 ; Input data, CS still low
A0            MOV      @R0, A      ; Store in memory
18            INC      R0          ; Increment storage counter
89 01          ORL      P1, #1      ; Reset CS signal
27            CLR      A           ; Clear ACC to get out of
93            RETR                ; the interrupt loop
    
```

02000471-03

4.2 Interfacing the Z-80

The Z-80 control bus is slightly different from that of the 8080. General RD and WR strobes are provided and separate memory request, MREQ, and I/O request, IORQ, signals are used which have to be combined with the generalized strobes to provide the equivalent 8080 signals. An advantage of operating the A/D in I/O space with the Z-80 is that the CPU will automatically insert one wait state (the RD and WR strobes are extended one clock period) to allow more time for the I/O devices to respond. Logic to map the A/D in I/O space is shown in Figure 14.

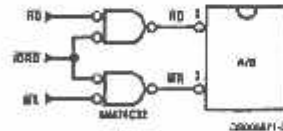


FIGURE 14. Mapping the A/D as an I/O Device for Use with the Z-80 CPU

Additional I/O advantages exist as software DMA routines are available and use can be made of the output data transfer which exists on the upper 8 address lines (A8 to A15) dur-

Functional Description (Continued)

SAMPLE PROGRAM FOR Figure 15 ADC0801-MC6800 CPU INTERFACE

```

0010 DF 36      DATAIN   STX      TEMP2      ; Save contents of X
0012 CE 00 2C   LDX      #$002C      ; Upon IRQ low CPU
0015 FF FF F8   STX      $7FF8      ; jumps to 002C
0018 B7 50 00   STAA     $5000      ; Start ADC0801
001B 0E                CLI                ;
001C 3E                CONVRT   WAI                ; Wait for interrupt
001D DE 34       LDX      TEMP1
001F 8C 02 0F   CPI      #$020F      ; Is final data stored?
0022 27 14       BEQ      ENDP
0024 B7 50 00   STAA     $5000      ; Restarts ADC0801
0027 08                INX
0028 CF 34       STX      TEMP1
002A 20 F0       BRA      CONVRT
002C DE 34       INTRPT   LDX      TEMP1
002E B8 80 00   LDAA     $8000      ; Read data
0031 A7 00       STAA     X                ; Store it at X
0033 3B                RTI
0034 02 00       TEMP1   FDB      $0200      ; Starting address for
; data storage
003B 00 00       TEMP2   FDB      $0000
003B CE 02 00   ENDP    LDX      #$0200      ; Reinitialize TEMP1
003B DF 34       STX      TEMP1
003D DE 36       LDX      TEMP2
003F 39                RTS                ; Return from subroutine
; To user's program
    
```

DS000171A1

Note 22: In order for the microprocessor to service subroutines and interrupts, the stack pointer must be dimensioned in the user's program.

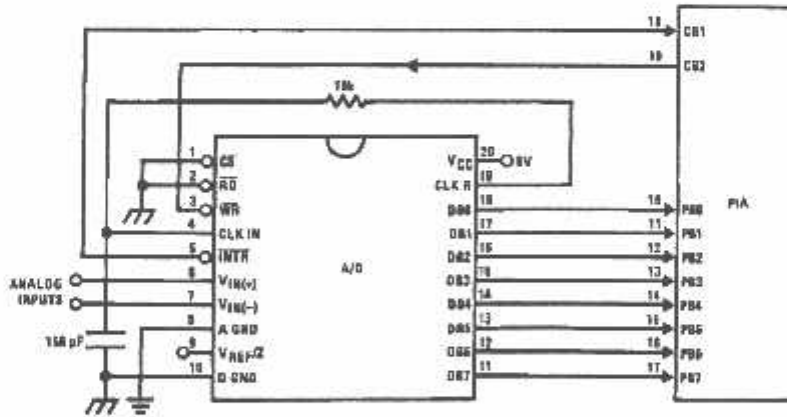


FIGURE 16. ADC0801-MC6820 PIA interface

Functional Description (Continued)

SAMPLE PROGRAM FOR Figure 16 ADC0801—MC6820 PIA INTERFACE

```

0010    CE 00 38    DATAIN    LDX    #0038    ; Upon  $\overline{IRQ}$  low CPU
0013    FF FF 78    STX    #7778    ; jumps to 0038
0016    86 80 06    LDAA   PIAORB   ; Clear possible  $\overline{IRQ}$  flags
0019    4F          CLR    CLRA
001A    B7 80 07    STAA   PIACRB
001D    B7 80 06    STAA   PIAORB   ; Set Port B as input
0020    0E          CLI
0021    C6 34    LDAB   #434
0023    86 3D    LDAA   #3D
0025    F7 80 07    CONVRT   STAB   PIACRB   ; Starts ADC0801
0028    B7 80 07    STAA   PIACRB
002E    3E          WAI          ; Wait for interrupt
002C    DE 40    LDX    TEMP1
002E    8C 02 0F    CPX    #020F   ; Is final data stored?
0031    27 0F    BEQ    ENDP
0033    08          INX
0034    D7 40    STX    TEMP1
0036    20 ED    BRA    CONVRT
0038    DE 40    INTRPT   LDX    TEMP1
005A    B6 80 08    LDAA   PIAORB   ; Read data in
003D    A7 00    STAA   X        ; Store it at X
0037    3B          RTI
0040    02 00    TEMP1   FDB   #0200   ; Starting address for
                                ; data storage
0042    CE 02 00    ENDP    LDX    #0200   ; Reinitialize TEMP1
0045    D7 40    STX    TEMP1
0047    39          RTS          ; Return from subroutine
                                ; to user's program
                                PIAORB   EQU    #8006
                                PIACRB   EQU    #8007

```

2000007-02

The following schematic and sample subroutine (DATA IN) may be used to interface (up to) 8 ADC0801's directly to the MC6800 CPU. This scheme can easily be extended to allow the interface of more converters. In this configuration the converters are (arbitrarily) located at HEX address 5000 in the MC6800 memory space. To save components, the clock signal is derived from just one RC pair on the first converter. This output drives the other A/Ds.

All the converters are started simultaneously with a STORE instruction at HEX address 5000. Note that any other HEX address of the form 5XXX will be decoded by the circuit, pulling all the \overline{CS} inputs low. This can easily be avoided by using a more definitive address decoding scheme. All the interrupts are ORed together to insure that all A/Ds have completed their conversion before the microprocessor is interrupted.

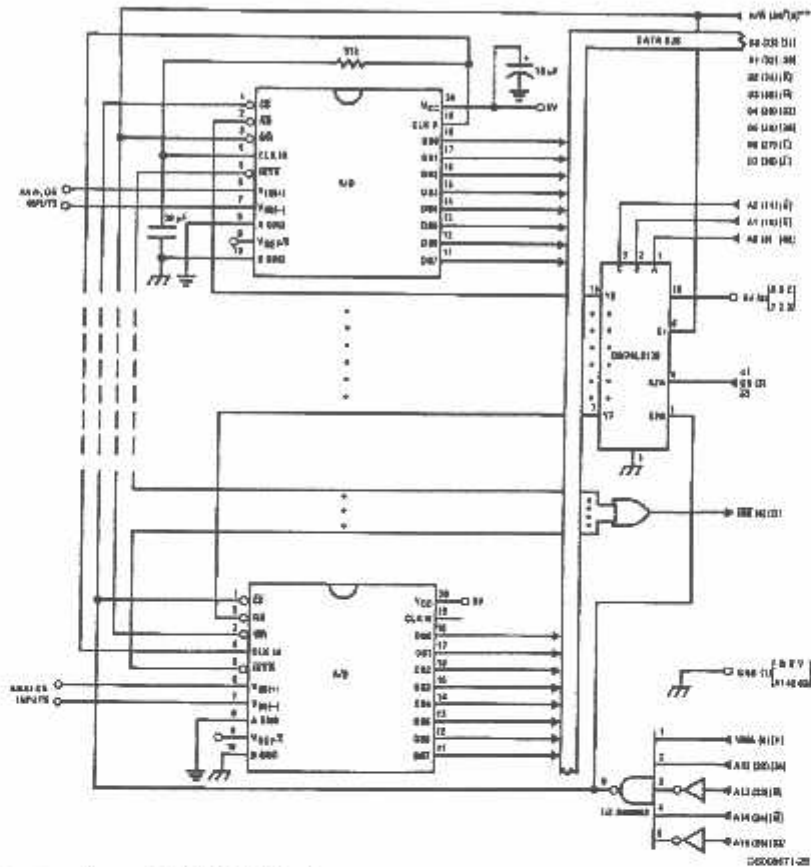
The subroutine, DATA IN, may be called from anywhere in the user's program. Once called, this routine initializes the

CPU, starts all the converters simultaneously and waits for the interrupt signal. Upon receiving the interrupt, it reads the converters (from HEX addresses 5000 through 5007) and stores the data successively at (arbitrarily chosen) HEX addresses 0200 to 0207, before returning to the user's program. All CPU registers then recover the original data they had before servicing DATA IN.

5.2 Auto-Zeroed Differential Transducer Amplifier and A/D Converter

The differential inputs of the ADC0801 series eliminate the need to perform a differential to single ended conversion for a differential transducer. Thus, one op amp can be eliminated since the differential to single ended conversion is provided by the differential input of the ADC0801 series. In general, a transducer preamp is required to take advantage of the full A/D converter input dynamic range.

Functional Description (Continued)



Note 23: Numbers in parentheses refer to MC6800 CPU pin out.

Note 24: Numbers of letters in brackets refer to standard MC6800 system common bus code.

FIGURE 17. Interfacing Multiple A/Ds in an MC6800 System

Functional Description (Continued)

SAMPLE PROGRAM FOR Figure 17 INTERFACING MULTIPLE A/D's IN AN MC6800 SYSTEM

ADDRESS	HEX CODE		MNEMONICS		COMMENTS
0010	DF 44	DATAIN	STX	TEMP	; Save Contents of X
0012	CE 00 2A		LDX	#\$002A	; Open IRQ LOW CPU
0013	7F FF F8		STX	\$\$\$F8	; Jumps to 002A
0018	B7 50 00		STAA	\$\$\$000	; Starts all A/D's
001B	0E		CLI		
001C	3E		WAI		; Wait for interrupt
001D	CE 50 00		LDX	\$\$\$000	
0020	DF 40		STX	INDEX1	; Reset both INDEX
0022	CE 02 00		LDX	\$\$\$0200	; 1 and 2 to starting
0025	DF 42		STX	INDEX2	; addresses
0027	DE 44		LDX	TEMP	
0029	39		RTS		; Return from subroutine
002A	DE 40	INTRPT	LDX	INDEX1	; INDEX1 → X
002C	A6 00		LDAA	X	; Read data in from A/D at X
002E	08		INX		; Increment X by one
002F	DF 40		STX	INDEX1	; X → INDEX1
0031	DE 42		LDX	INDEX2	; INDEX2 → X

000001-43

SAMPLE PROGRAM FOR Figure 17 INTERFACING MULTIPLE A/D's IN AN MC6800 SYSTEM

ADDRESS	HEX CODE		MNEMONICS		COMMENTS
0033	A7 00		STAA	X	; Store data at X
0035	8C 02 07		CFX	\$\$\$0207	; Have all A/D's been read?
0038	27 05		BEQ	RETURN	; Yes: branch to RETURN
003A	08		INX		; No: increment X by one
003B	DF 42		STX	INDEX2	; X → INDEX2
003D	20 8B		BRA	INTRPT	; Branch to 002A
003F	3B	RETURN	RTI		
0040	50 00	INDEX1	FDB	\$\$\$000	; Starting address for A/D
0042	02 00	INDEX2	FDB	\$\$\$0200	; Starting address for data storage
0044	00 00	TEMP	FDB	\$\$\$0000	

000001-44

Note 25: In order for the microprocessor to service subroutines and interrupts, the stack pointer must be dimensioned in the user's program.

For amplification of DC input signals, a major system error is the input offset voltage of the amplifiers used for the preamp. Figure 18 is a gain of 100 differential preamp whose offset voltage errors will be cancelled by a zeroing subroutine which is performed by the INSB080A microprocessor system. The total allowable input offset voltage error for this preamp is only 50 μ V for 1/4 LSB error. This would obviously require very precise amplifiers. The expression for the differential output voltage of the preamp is:

$$V_O = \underbrace{[V_{IN(+)} - V_{IN(-)}]}_{\text{SIGNAL}} \left[1 + \frac{2R_2}{R_1} \right] + \underbrace{[V_{OS2} - V_{OS1} - V_{OS3} \pm I_X R_X]}_{\text{DC ERROR TERM}} \left(1 + \frac{2R_2}{R_1} \right)$$

where I_X is the current through resistor R_X . All of the offset error terms can be cancelled by making $\pm I_X R_X = V_{OS1} + V_{OS3} - V_{OS2}$. This is the principle of this auto-zeroing scheme.

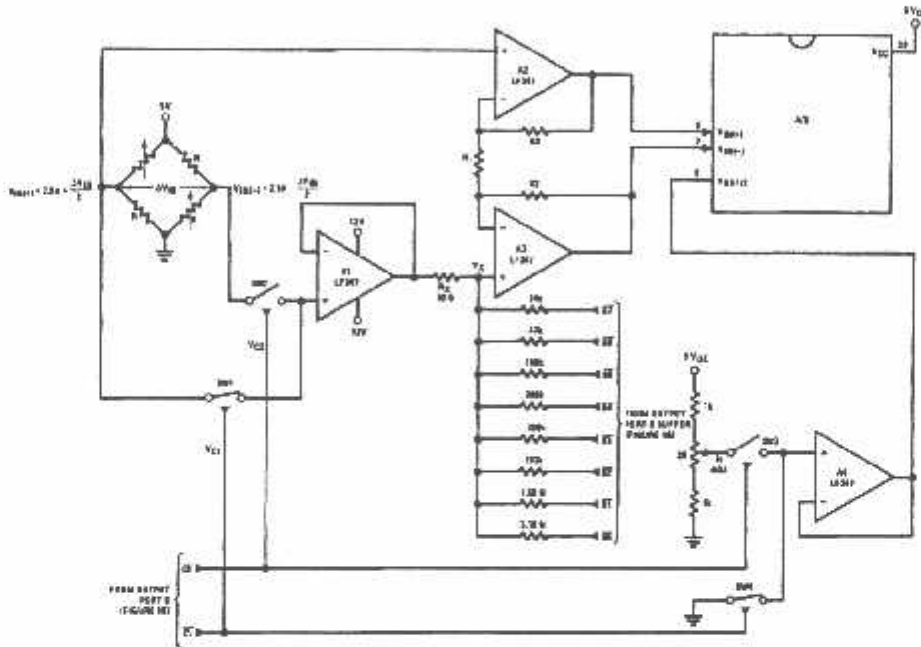
The INSB080A uses the 3 I/O ports of an INSB255 Programmable Peripheral Interface (PPI) to control the auto zeroing and input data from the ADC0801 as shown in Figure 19. The PPI is programmed for basic I/O operation (mode 0) with Port A being an input port and Ports B and C being output ports. Two bits of Port C are used to alternately open or close the 2 switches at the input of the preamp. Switch SW1 is closed to force the preamp's differential input to be zero during the zeroing subroutine and then opened and SW2 is then closed for conversion of the actual differential input signal. Using 2 switches in this manner eliminates concern for the ON resistance of the switches as they must conduct only the input bias current of the input amplifiers.

Output Port B is used as a successive approximation register by the 8080 and the binary scaled resistors in series with each output bit create a D/A converter. During the zeroing subroutine, the voltage at V_x increases or decreases as required to make the differential output voltage equal to zero. This is accomplished by ensuring that the voltage at the output of A1 is approximately 2.5V so that a logic "1" (5V) on

Functional Description (Continued)

any output of Port B will source current into node V_x , thus raising the voltage at V_x and making the output differential more negative. Conversely, a logic "0" (0V) will pull current out of node V_x and decrease the voltage, causing the differential output to become more positive. For the resistor values shown, V_x can move ± 12 mV with a resolution of 50 μ V, which will null the offset error term to $1/4$ LSB of full-scale for

the ADC0801. It is important that the voltage levels that drive the auto-zero resistors be constant. Also, for symmetry, a logic swing of 0V to 5V is convenient. To achieve this, a CMOS buffer is used for the logic output signals of Port B and this CMOS package is powered with a stable 5V source. Buffer amplifier A1 is necessary so that it can source or sink the D/A output current.



- Note 26: $R_2 = 49.5 R_1$
- Note 27: Switches are LMC1334 CMOS analog switches.
- Note 28: The 9 resistors used in the auto-zero section can be $\pm 5\%$ tolerance.

FIGURE 18. Gain of 100 Differential Transducer Preamp

Functional Description (Continued)

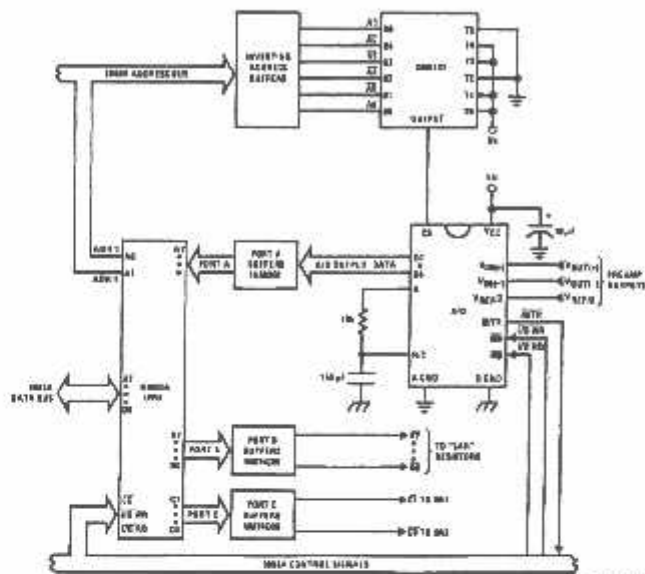


FIGURE 19. Microprocessor Interface Circuitry for Differential Preamp

A flow chart for the zeroing subroutine is shown in Figure 20. It must be noted that the ADC0801 series will output an all zero code when it converts a negative input [$V_{IN(-)} \geq V_{IN(+)}$]. Also, a logic inversion exists as all of the I/O ports are buffered with inverting gates.

Basically, if the data read is zero, the differential output voltage is negative, so a bit in Port B is cleared to pull V_X more negative which will make the output more positive for the next conversion. If the data read is not zero, the output voltage is positive so a bit in Port B is set to make V_X more positive and the output more negative. This continues for 8 approximations and the differential output eventually converges to within 5 mV of zero.

The actual program is given in Figure 21. All addresses used are compatible with the BLC 80/10 microcomputer system. In particular:

- Port A and the ADC0801 are at port address E4
- Port B is at port address E5
- Port C is at port address E6
- PP1 control word port is at port address E7
- Program Counter automatically goes to ADDR:3C3D upon acknowledgement of an interrupt from the ADC0801

5.3 Multiple A/D Converters in a Z-80 Interrupt Driven Mode

In data acquisition systems where more than one A/D converter (or other peripheral device) will be interrupting program execution of a microprocessor, there is obviously a

need for the CPU to determine which device requires servicing. Figure 22 and the accompanying software is a method of determining which of 7 ADC0801 converters has completed a conversion (INTR asserted) and is requesting an interrupt. This circuit allows starting the A/D converters in any sequence, but will input and store valid data from the converters with a priority sequence of A/D 1 being read first, A/D 2 second, etc., through A/D 7 which would have the lowest priority for data being read. Only the converters whose INT is asserted will be read.

The key to decoding circuitry is the DM74LS373, 8-bit D type flip-flop. When the Z-80 acknowledges the interrupt, the program is vectored to a data input Z-80 subroutine. This subroutine will read a peripheral status word from the DM74LS373 which contains the logic state of the INTR outputs of all the converters. Each converter which initiates an interrupt will place a logic "0" in a unique bit position in the status word and the subroutine will determine the identity of the converter and execute a data read. An identifier word (which indicates which A/D the data came from) is stored in the next sequential memory location above the location of the data so the program can keep track of the identity of the data entered.

Functional Description (Continued)

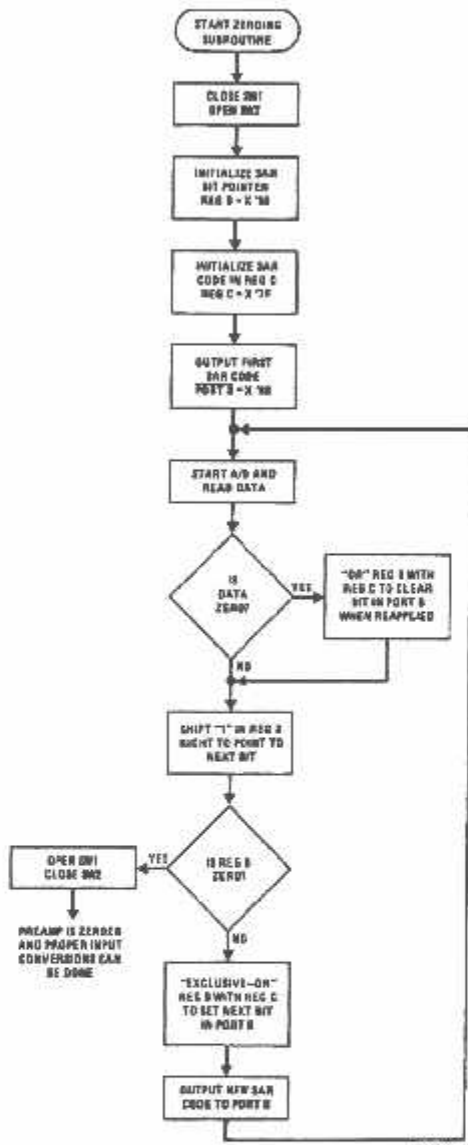


FIGURE 20. Flow Chart for Auto-Zero Routine

Functional Description (Continued)

```

3D00 3E90 MVI 90
3D02 D5E7 Out Control Port ; Program PPI
3D04 2601 MVI H 01 Auto-Zero Subroutine
3D06 7C MOV A, H
3D07 D5E6 OUT C ; Close SW1 open SW2
3D09 0680 MVI B 80 ; Initialize SAR bit pointer
3D0B 3E7F MVI A 7F ; Initialize SAR code
3D0D 4F MOV C, A Return
3D0E D5E5 OUT B ; Port B = SAR code
3D10 31AA3D LXI SP 3DAA Start ; Dimension stack pointer
3D13 D5E4 OUT A ; Start A/D
3D15 FB IE Loop ; Loop until INT asserted
3D16 00 NOP
3D17 C3163D JMP Loop
3D1A 7A MOV A, D Auto-Zero
3D1B C600 ADI 00
3D1D CA2D3D JZ Set C ; Test A/D output data for zero
3D20 7B MOV A, B Shift B
3D21 F600 ORI 00 ; Clear carry
3D23 1F RAR ; Shift "1" in B right one place
3D24 FE00 CPI 00 ; Is B zero? If yes last
3D26 CA373D JZ Done ; approximation has been made
3D29 47 MOV B, A
3D2A C3333D JMP New C
3D2D 79 MOV A, C Set C
3D2E B0 ORA B ; Set bit in C that is in same
3D2F 4F MOV C, A ; position as "1" in B
3D30 C3203D JMP Shift B
3D33 A9 XRA C New C ; Clear bit in C that is in
3D34 C5D33D JMP Return ; same position as "1" in B
3D37 47 MOV B, A Done ; then output new SAR code.
3D38 7C MOV A, H ; Open SW1, close SW2 then
3D39 EE03 XRI 03 ; proceed with program. Preamp
3D3B D5E6 OUT C ; is now zeroed.
3D3D * Normal
*
*
Program for processing
proper data values
3C3D DBE4 IN A Read A/D Subroutine ; Read A/D data
3C3F EEFF XRI FF ; Invert data
3C41 57 MOV D, A
3C42 7B MOV A, B ; Is B Reg = 0? If not stay
3C43 EEFF XRI FF ; in auto zero subroutine
3C45 C21A3D JNZ Auto-Zero
3C48 C35D3D JMP Normal

```

DS02671-48

Note 28: All numerical values are hexadecimal representations.

FIGURE 21. Software for Auto-Zeroed Differential A/D

5.3 Multiple A/D Converters in a Z-80 Interrupt Driven Mode (Continued)

The following notes apply:

- It is assumed that the CPU automatically performs a RST 7 instruction when a valid interrupt is acknowledged (CPU is in interrupt mode 1). Hence, the subroutine starting address of X0038.
- The address bus from the Z-80 and the data bus to the Z-80 are assumed to be inverted by bus drivers.
- A/D data and identifying words will be stored in sequential memory locations starting at the arbitrarily chosen address X 3E00.
- The stack pointer must be dimensioned in the main program as the RST 7 instruction automatically pushes the PC onto the stack and the subroutine uses an additional 6 stack addresses.
- The peripherals of concern are mapped into I/O space with the following port assignments:

Functional Description (Continued)

HEX PORT ADDRESS	PERIPHERAL
00	MM74C374 8-bit BIP-flop
01	A/D 1
02	A/D 2
03	A/D 3

HEX PORT ADDRESS

PERIPHERAL

04	A/D 4
05	A/D 5
06	A/D 6
07	A/D 7

This port address also serves as the A/D identifying word in the program.

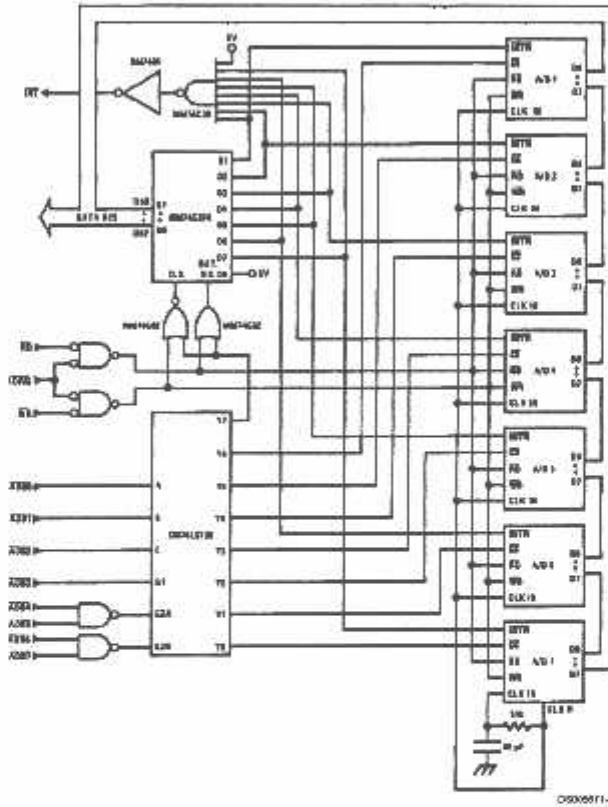


FIGURE 22. Multiple A/Ds with Z-80 Type Microprocessor

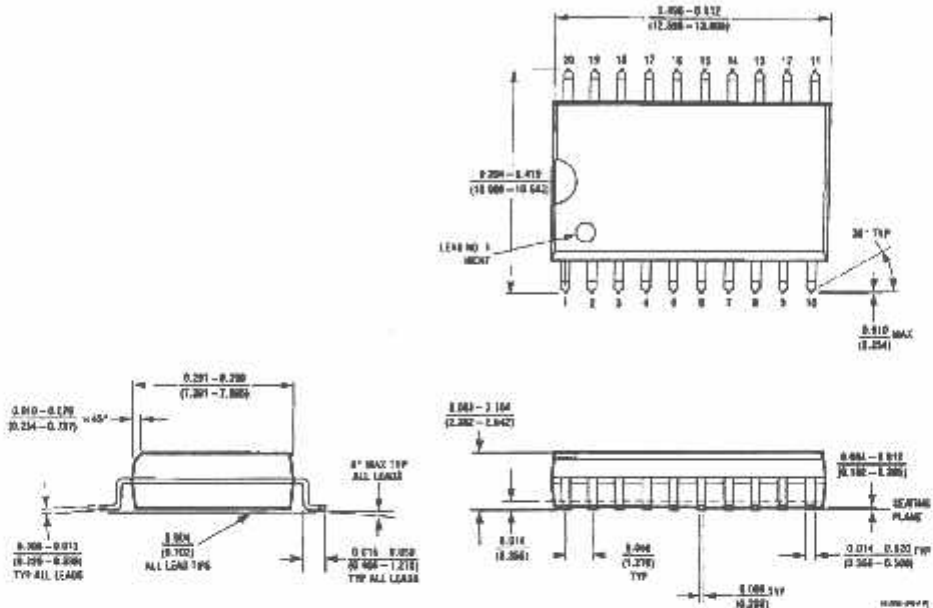
Functional Description (Continued)

LOC	OBJ CODE	SOURCE	STATEMENT	COMMENT
0038	E5		PUSH HL	; Save contents of all registers affected by
0039	C5		PUSH BC	; this subroutine.
003A	F5		PUSH AF	; Assumed INT mode 1 earlier set.
003B	21 00 3E		LD (HL), X3E00	; Initialize memory pointer where data will be stored.
003E	0E 01		LD C, X01	; C register will be port ADDR of A/D converters.
0040	D300		OUT X00, A	; Load peripheral status word into 8-bit latch.
0042	0B00		IN A, X00	; Load status word into accumulator.
0044	47		LD B, A	; Save the status word.
0045	79	TEST	LD A, C	; Test to see if the status of all A/D's have
0046	F8 08		CF, X08	; been checked. If so, exit subroutine
0048	CA 60 00		JFZ, DONE	
004B	78		LD A, B	; Test a single bit in status word by looking for
004C	1F		RRA	; a "1" to be rotated into the CARRY (an INT
004D	47		LD B, A	; is loaded as a "1"). If CARRY is set then load
004E	DA 5500		JPC, LOAD	; contents of A/D at port ADDR in C register.
0051	0C	NEXT	INC C	; If CARRY is not set, increment C register to point
0052	E3 4500		JP, TEST	; to next A/D, then test next bit in status word.
0055	ED 78	LOAD	IN A, (C)	; Read data from interrupting A/D and invert
0057	EE 7F		XOR FF	; the data.
0059	77		LD (HL), A	; Store the data
005A	2C		INCL	
005B	71		LD (HL), C	; Store A/D identifier (A/D port ADDR).
005C	2C		INCL	
005D	03 51 00		JP, NEXT	; Test next bit in status word.
0060	F1	DONE	POP AF	; Re-establish all registers as they were
0061	C1		POP BC	; before the interrupt.
0062	B1		POP HL	
0063	09		RET	; Return to original program

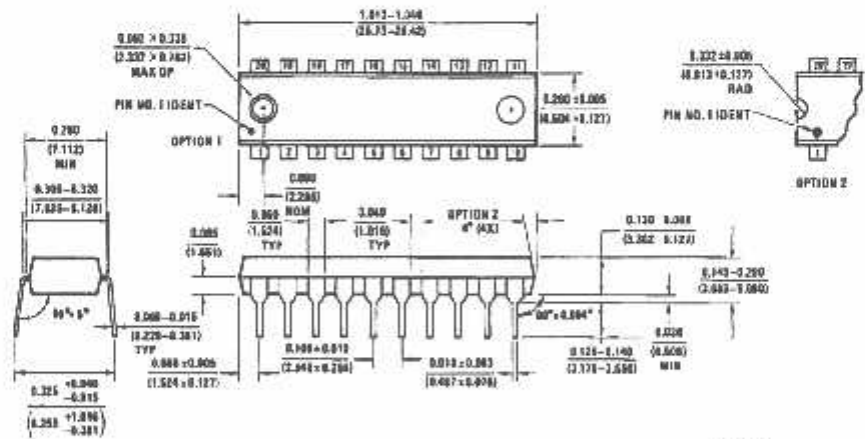
000071-06

ADC0801/ADC0802/ADC0803/ADC0804/ADC0805

Physical Dimensions inches (millimeters) unless otherwise noted



SO Package (M)
 Order Number ADC0802LCWM or ADC0804LCWM
 NS Package Number M20B



Molded Dual-in-Line Package (N)
 Order Number ADC0801LCN, ADC0802LCN,
 ADC0803LCN, ADC0804LCN or ADC0805LCN
 NS Package Number N20A

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: 1-800-272-6958
Fax: 1-800-737-7019
Email: support@nsc.com

www.national.com

National Semiconductor
Europe

Fax: +49 (0) 1 80-530 85 85
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 1 80-530 85 85
English Tel: +49 (0) 1 80-532 76 32
Français Tel: +49 (0) 1 80-532 63 58
Italiano Tel: +49 (0) 1 80-534 16 90

National Semiconductor
Asia Pacific Customer

Response Group
Tel: 65-2544488
Fax: 65-2504466
Email: asen.support@nsc.com

National Semiconductor
Japan Ltd.

Tel: 81-3-5639-7500
Fax: 81-3-5639-7507

LIQUID CRYSTAL DISPLAY MODULE

M 1 6 3 2

USER MANUAL

Seiko Instruments Inc.

1. GENERAL

1.1 General

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

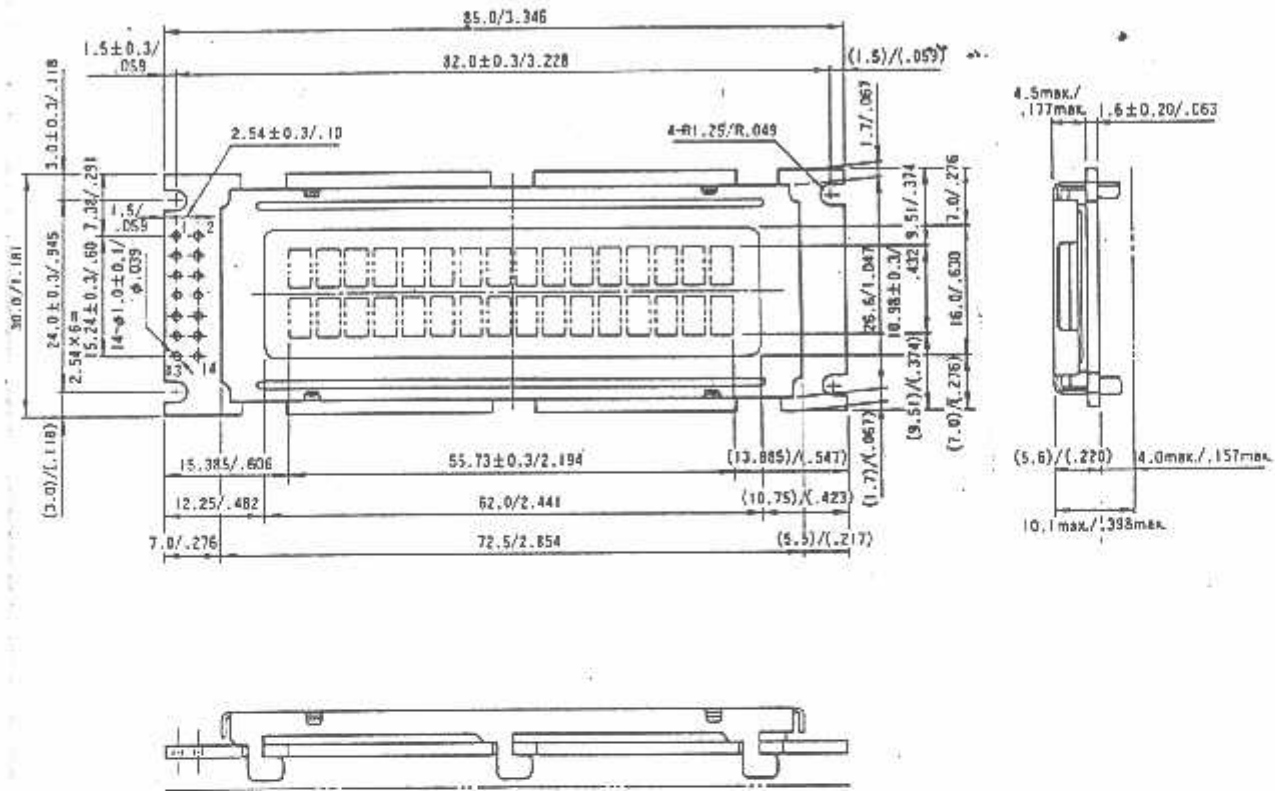
1.2 Features

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

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

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

1.3 Dimensions Diagram



Unit : mm/inch
 General tolerance : ± 0.5 mm

I/O terminal symbol

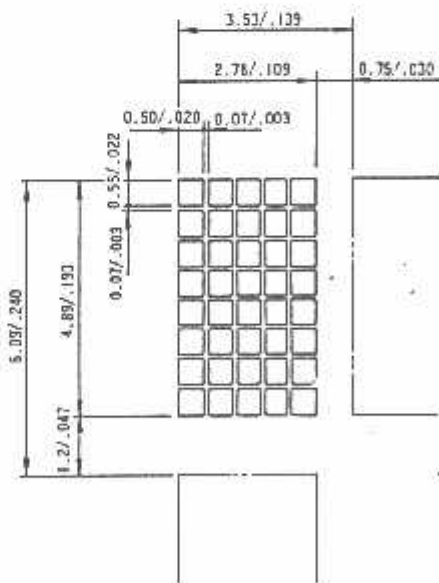
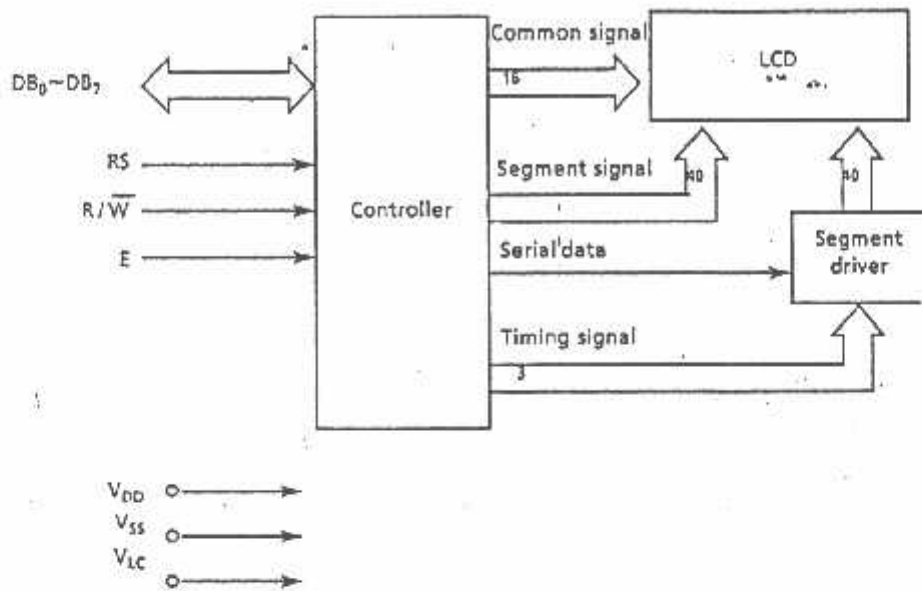


Figure 1 Dimensions diagram

No.	Symbol	Level	Function	
1	Vss	-	Power Supply	0V (GND)
2	Vcc	-		5V ±10%
3	Vee	-		for LCD Drive
4	RS	H/L	H: Data Input L: Instruction Input	
5	R/W	H/L	H:READ L:WRITE	
6	E	H,	Enable Signal	
7	DB0	H/L	Data Bus	
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L		
12	DB5	H/L		
13	DB6	H/L		
14	DB7	H/L		
15	V+ BL	-	Back Light Supply	4 - 4.2V 50-200mA
16	V- BL	-		0V (GND)

1.4 Block Diagram



1.5 Absolute Maximum Ratings

 $V_{SS} = 0V$

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

1.6 Electrical Characteristics

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

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

1.7 Optical Characteristics

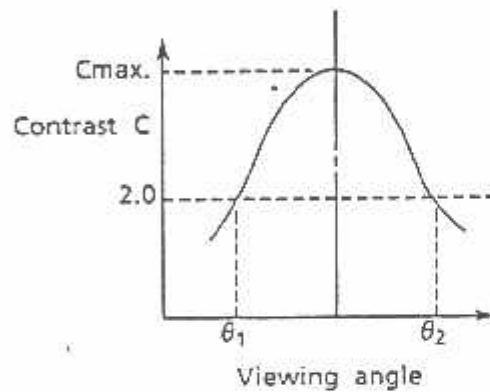
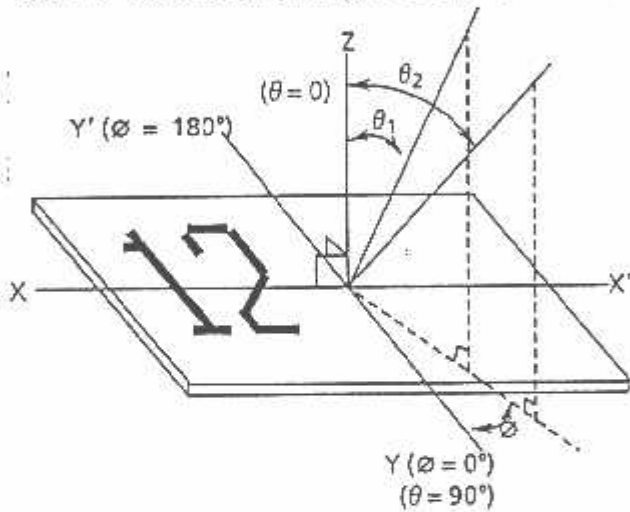
1.7.1 Optical characteristics

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

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

Note 1: Definition of angles \varnothing and θ

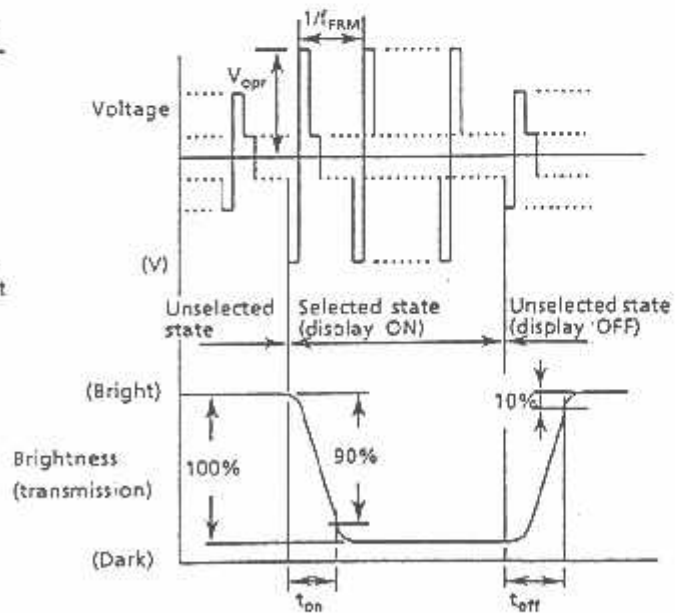
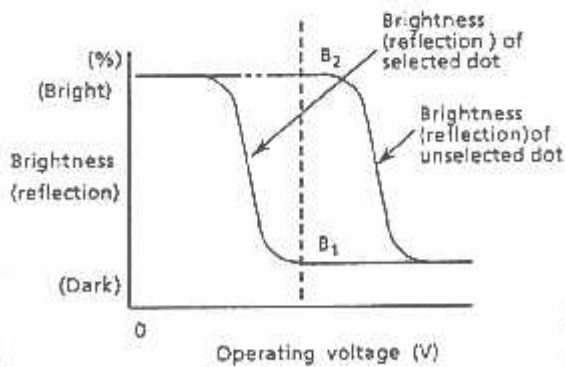
Note 2: Definition of viewing angles θ_1 and θ_2



Note 3: Definition of contrast C

Note 4: Definition of response time

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



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

1.7.2 Recommended operating voltage

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

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

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

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

2. OPERATING INSTRUCTIONS

2.1 Terminal Functions

Table 1 Terminal functions

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

2.2 Basic Operations

2.2.1 Registers

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

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

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

Table 2 Register selection

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

2.2.2 Busy flag (BF)

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

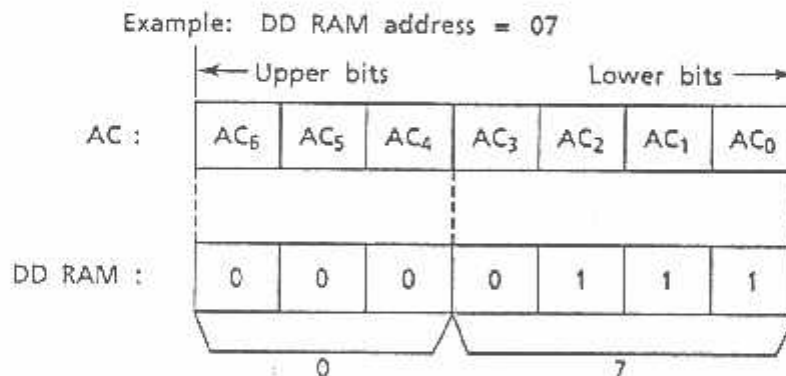
2.2.3 Address counter (AC)

The counter specifies an address when data is written into DD RAM or CG RAM and the data stored in DD RAM or CG RAM is read out. If an Address Set instruction (for DD RAM or CG RAM) is written in the IR, the address information is transferred from the IR to the AC. When display data is written into or read from DD RAM or CG RAM, the AC is automatically incremented or decremented by one according to the Entry Mode Set. The contents of the AC are output to DB₀ to DB₆ as shown in Table 2 if RS = 0 and $R/\overline{W} = 1$.

2.2.4 Display data RAM (DD RAM)

DD RAM has a capacity of up to 80×8 bits and stores display data of 80 eight-bit character codes. Some storage areas of DD RAM which are not used for display can be used as general data RAM.

A DD RAM address to be set in the AC is expressed in hexadecimal form as follows.



00H to 0FH of the DD RAM address is set in the line 1, and 40H to 4FH in the line 2.

Note : The addresses in the digit 16 of line 1 and the digit 1 of line 2 are not consecutive.

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

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

*Left shift

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

*Right shift

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

2.2.5 Character generator ROM (CG ROM)

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

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

2.2.6 Character generator RAM (CG RAM)

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

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

Table 3 Correspondence between character codes and character patterns

Upper bit 4 bit Lower bit 4 bit	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
x x x x 0000	CG RAM (1)		0	a	P	\	P		レ	9	3	0	D
	(2)	!	1	A	Q	a	9	u	7	7	4	3	Q
x x x x 0010	(3)	"	2	B	R	b	r	r	4	7	x	8	0
	(4)	#	3	C	S	c	s	u	7	7	8	8	0
x x x x 0100	(5)	*	4	D	T	d	t	\	7	7	7	u	0
	(6)	%	5	E	U	e	u	.	7	7	1	0	0
x x x x 0110	(7)	&	6	F	V	f	v	9	カ	二	9	0	Σ
	(8)	'	7	G	W	g	w	7	7	7	7	0	π
x x x x 1000	(1)	(8	H	X	h	x	4	0	*	u	7	Σ
	(2))	9	I	Y	i	y	9	7	7	u	'	u
x x x x 1010	(3)	*	#	J	Z	j	z	7	コ	△	△	i	7
	(4)	+	:	K	E	k	e	7	7	7	0	*	7
x x x x 1100	(5)	:	<	L	*	l	l	7	3	7	7	0	7
	(6)	---	=	M	I	m	i	u	7	7	7	7	+
x x x x 1110	(7)	..	>	N	^	n	→	3	7	7	*	7	
	(8)	/	?	O	_	o	+	u	7	7	7	0	7
x x x x 1111													7

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

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

Notes: • In CG RAM data, 1 corresponds to Selection and 0 to Non-selection on the display.

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

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

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

2.3 Timing Characteristics

2.3.1 Write timing characteristics

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

Item	Symbol	Standard		Unit
		Min.	Max.	
Enable cycle time	t_{cycE}	1000	—	ns
Enable pulse width	High level	PW_{EH}	—	ns
Enable rise and fall time	t_{er}, t_{ef}	—	25	ns
Setup time	$RS, R\bar{W} \rightarrow E$	t_{AS}	—	ns
Address hold time	t_{AH}	10	—	ns
Data setup time	t_{DSW}	195	—	ns
Data hold time	t_H	10	—	ns

Write operation

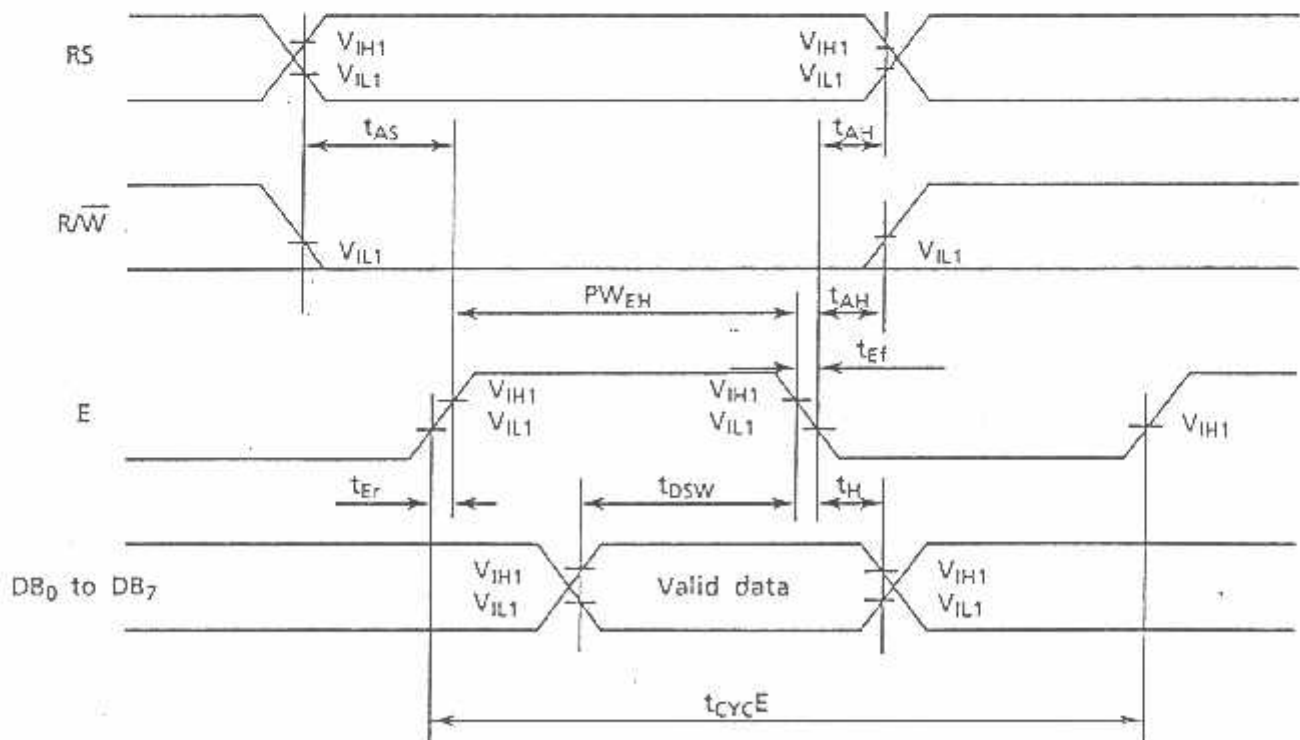


Figure 3 Data write from MPU to module

2.3.2 Read timing characteristics

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

Item	Symbol	Standard		Unit
		Min.	Max.	
Enable cycle time	t_{CYCE}	1000	-	ns
Enable pulse width	High level	PW_{EH}	-	ns
Enable rise and fall time	t_{Er}, t_{Ef}	-	25	ns
Setup time	RS, $\overline{R/W} \rightarrow E$	t_{AS}	-	ns
Address hold time	t_{AH}	10	-	ns
Data delay time	t_{DDR}	-	320	ns
Data hold time	t_H	20	-	ns

Read operation

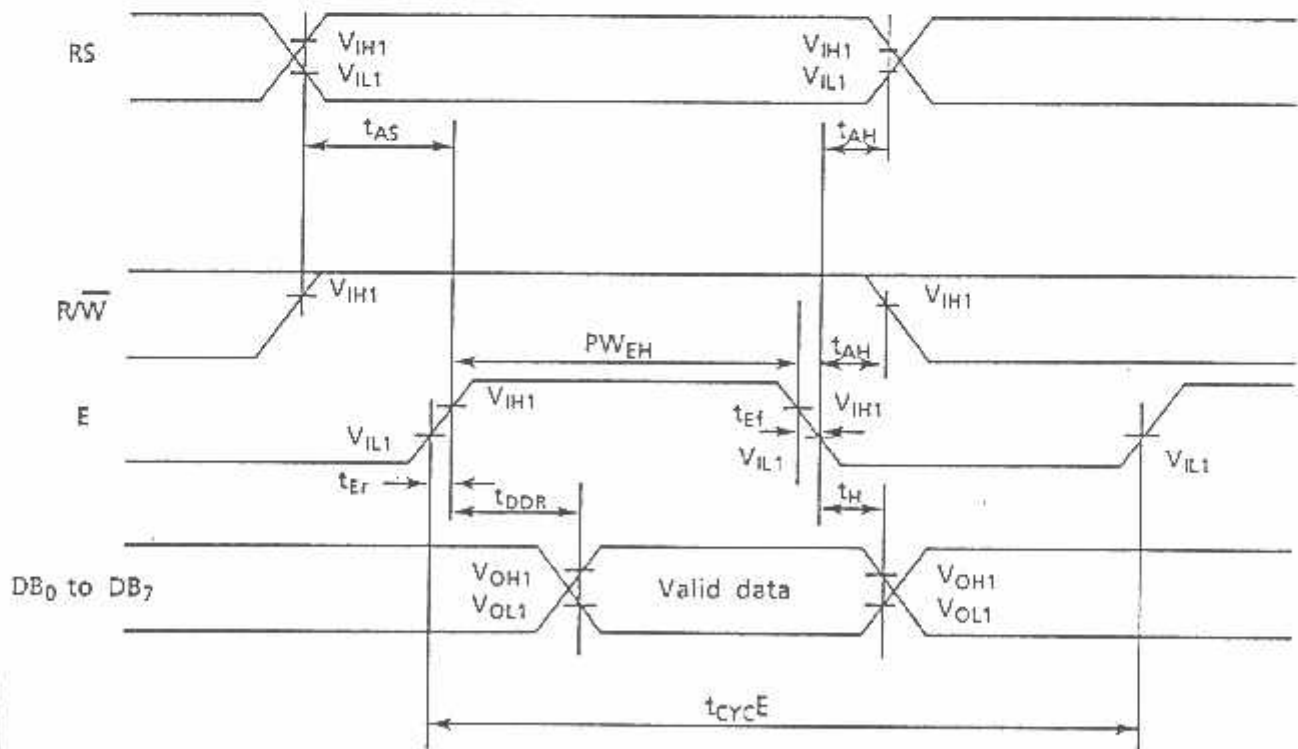


Figure 4 Data read from module to MPU

2.4 Instruction Outline

Table 5 List of instructions

Instruction	Code										Function	Execution time
	RS	R/W	DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀		
(1) Display clear	0	0	0	0	0	0	0	0	0	1	Clears all display and returns cursor to home position (address 0)	1.64 ms
(2) Cursor Home	0	0	0	0	0	0	0	0	1	*	Returns cursor to home position. Shifted display returns to home position and DD RAM contents do not change.	1.64 ms
(3) Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets direction of cursor movement and whether display will be shifted when data is written or read	40 μs
(4) Display ON / OFF control	0	0	0	0	0	0	1	D	C	B	Turns ON/OFF total display (D) and cursor (C), and makes cursor position column start blinking (B)	40 μs
(5) Cursor/Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves cursor and shifts display without changing DD RAM contents	40 μs
(6) Function Set	0	0	0	0	1	DL	1	*	*	*	Sets interface data length (DL)	40 μs
(7) CG RAM Address Set	0	0	0	1	A _{CG}					Sets CG RAM address to start transmitting or receiving CG RAM data	40 μs	
(8) DD RAM Address Set	0	0	1	A _{DD}					Sets DD RAM address to start transmitting or receiving DD RAM data	40 μs		
(9) BF/Address Read	0	1	BF	AC					Reads BF indicating module in Internal operation and AC contents (used for both CG RAM and DD RAM)	0 μs		
(10) Data Write to CG RAM or DD RAM	1	0	Write Data					Writes data into DD RAM or CG RAM	40 μs			
(11) Data Read from CG RAM or DD RAM	1	1	Read Data					Reads data from DD RAM or CG RAM	40 μs			

* : Invalid bit

A_{CG} : CG RAM addressA_{DD} : DD RAM address

I/D = 1 : Increment

I/D = 0 : Decrement

C = 1 : Cursor ON

C = 0 : Cursor OFF

R/L = 1 : Right shift

R/L = 0 : Left shift

S = 1 : Display shift

S = 0 : No display shift

B = 1 : Blink ON

B = 0 : Blink OFF

DL = 1 : 8 bits

DL = 0 : 4 bits

D = 1 : Display ON

D = 0 : Display OFF

S/C = 1 : Display shift

S/C = 0 : Cursor movement

BF = 1 : Internal operation in progress

BF = 0 : Instruction can be accepted

2.5 Instruction Details

(1) Display Clear

	RS	R/W	DB ₇							DB ₀
Code	0	0	0	0	0	0	0	0	0	1

Display Clear clears all display and returns cursor to home position (address 0).

Space code 20 (hexadecimal) is written into all the addresses of DD RAM, and DD RAM address 0 is set to the AC. If shifted, the display returns to the original position. After execution of the Display Clear instruction, the entry mode is incremented.

Note: When executing the Display Clear instruction, follow the restrictions listed in Table 6.

(2) Cursor Home

	RS	R/W	DB ₇							DB ₀
Code	0	0	0	0	0	0	0	0	1	*

* : Invalid bit

Cursor Home returns cursor to home position (address 0).

DD RAM address 0 is set to the AC. The cursor returns to the home position. If shifted, the display returns to the original position. The DD RAM contents do not change. If the cursor or blinking is ON, it returns to the left side.

Note: When executing the Cursor Home instruction, follow the restrictions listed in Table 6.

Table 6 Restrictions on execution of Display Clear and Cursor Home instructions

Conditions of use	Restrictions
When executing the Display Clear or Cursor Home instruction when the display is shifted (after execution of Display Shift instruction)	The Cursor Home instruction should be executed again immediately after the Display Clear or Cursor Home instruction is executed. Do not leave an interval of a multiple of $400/f_{osc}$ * second after the first execution. Example: 1.5 ms, 3 ms, 4.5 ms for $f_{osc} = 270$ kHz * f_{osc} : Oscillation frequency
When 23 _H , 27 _H , 63 _H , or 67 _H is used as a DD RAM address to execute Cursor Home instruction	Before executing the Cursor Home instruction, the data of the four DD RAM addresses given at the left should be read and saved. After execution, write the data again in DD RAM. (This restriction is necessary to prevent the contents of the DD RAM addresses from being destroyed after the Cursor Home instruction has been executed.)

(3) Entry Mode Set

	RS	R \overline{W}	DB ₇					DB ₀		
Code	0	0	0	0	0	0	0	1	I/D	S

Entry Mode Set sets the direction of cursor movement and whether display will be shifted.

I/D : The DD RAM address is incremented or decremented by one when a character code is written into or read from DD RAM. This is also true for writing into or reading from CG RAM.

When I/D = 1, the address is incremented by one and the cursor or blink moves to the right.

When I/D = 0, the address is decremented by one and the cursor or blink moves to the left.

S : If S = 1, the entire display is shifted either to the right or left for writing into DD RAM. The cursor position does not change, only the display moves. There is no display shift for reading from DD RAM.

When S = 1 and I/D = 1, the display shifts to the left.

When S = 1 and I/D = 0, the display shifts to the right.

If S = 0, the display does not shift.

(4) Display ON/OFF Control

	RS	R \overline{W}	DB ₇					DB ₀		
Code	0	0	0	0	0	0	1	D	C	B

Display ON/OFF Control turns the total display and the cursor ON and OFF, and makes the cursor position start blinking. Cursor ON/OFF and blinking is done at the column indicated by the specified DD RAM address by the AC.

D : When D = 1, the display is turned ON.

When D = 0, the display is turned OFF.

If D = 0 is used, display data remains in DD RAM. Change 0 to 1 to display data.

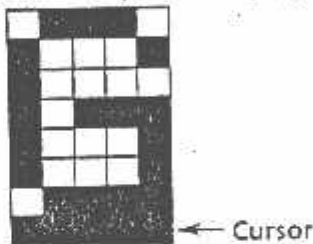
C : When $C = 1$, the cursor is displayed.
When $C = 0$, the cursor is not displayed.

The cursor is displayed in the dot line below the 5 x 7 dot-matrix character fonts. If the cursor is OFF, display data is written into DD RAM in the order specified by I/D.

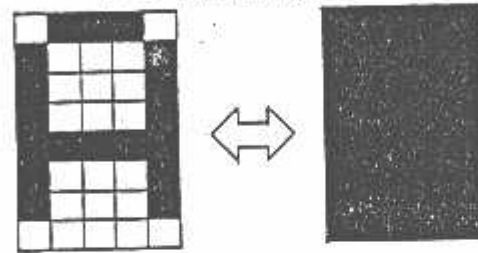
B : When $B = 1$, the character at the cursor position starts blinking.
When $B = 0$, it does not blink.

For blinking, all-black dots and the character are switched about every 0.4 seconds. The cursor and blinking can be set at the same time.

Example: $C = 1$ (cursor display)



$B = 1$ (blinking)



(5) Cursor/Display Shift

Code	RS	R/W	DB ₇					DB ₀			
	0	0	0	0	0	1	S/C	R/L	*	*	* : Invalid bit

Cursor/Display Shift moves the cursor and shifts the display without changing the DD RAM contents.

The cursor position and the AC contents match. This instruction is available for display correction and retrieval because the cursor position or display can be shifted without writing or reading display data. Since the DD RAM capacity is 40-character and two lines, the cursor is shifted from digit 40 of line 1 to digit 1 of line 2. Displays of lines 1 and 2 are shifted at the same time. Therefore, the display pattern of line 2 is not shifted to line 1.

S/C	R/L	Operation
0	0	The cursor position is shifted to the left (the AC decrements one).
0	1	The cursor position is shifted to the right (the AC increments one).
1	0	The entire display is shifted to the left with the cursor.
1	1	The entire display is shifted to the right with the cursor.

Note: If only display shift is done, the AC contents do not change.

(6) Function Set

	RS	R/W	DB ₇								DB ₀	
Code	0	0	0	0	1	DL	1	*	*	*	* : Invalid bit	

Function Set sets the interface data length.

DL : Interface data length

When DL = 1, the data length is set at eight bits (DB₇ to DB₀).

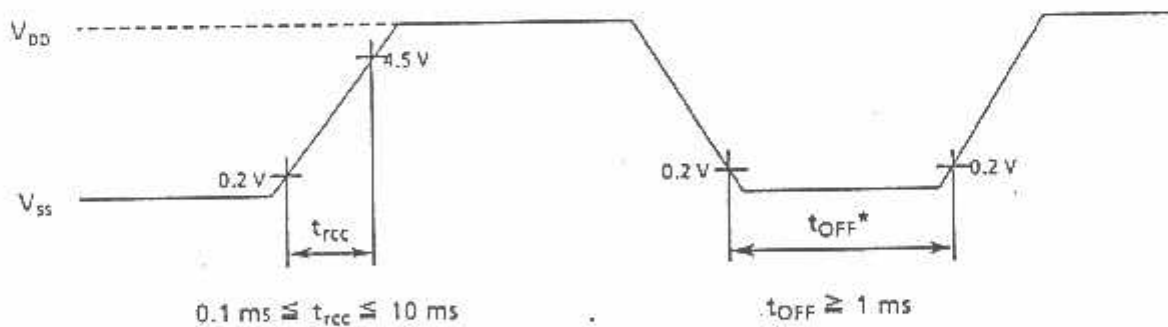
When DL = 0, the data length is set at four bits (DB₇ to DB₄).

The upper four bits are transferred first, then the lower four bits follow.

The Function Set instruction must be executed prior to all other instructions except for Busy Flag/Address Read. If another instruction is executed first, no function instruction except changing the interface data length can be executed.

Remarks: Initialization

The system is automatically initialized at power-on if the following power supply conditions are satisfied.



* t_{OFF} : Time when power supply is OFF if cut instantaneously or turned ON and OFF repeatedly

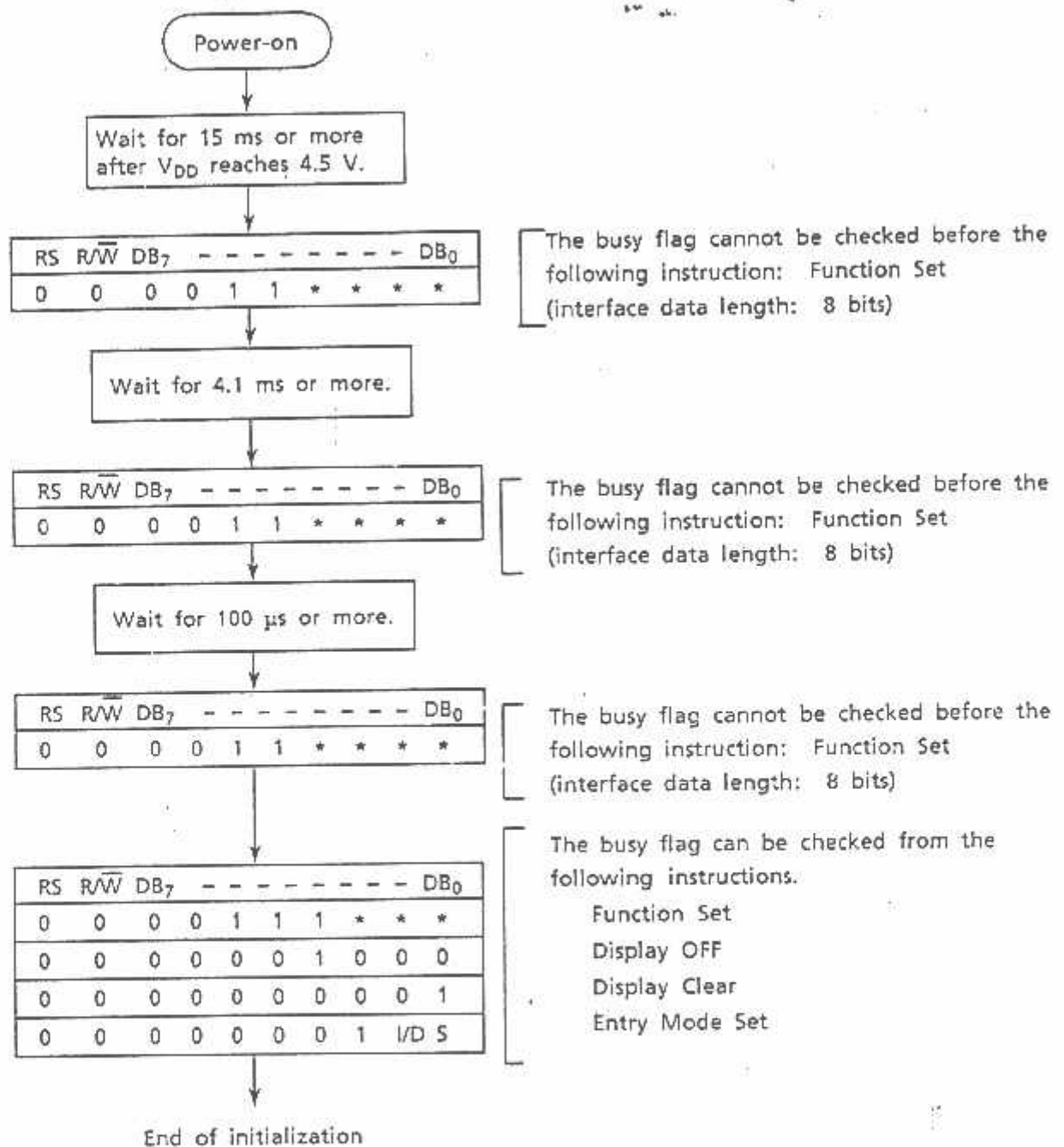
The following instructions are executed for initialization.

- 5 x 7 dot-matrix character font: 1/8 duty
- Display clear
- Function Set DL = 1: Interface data length: 8 bits
- Display ON/OFF Control D = 0: Display OFF
 C = 0: Cursor OFF
 B = 0: Blink OFF
- Entry mode I/O = 1: Increment
 S = 0: No display shift

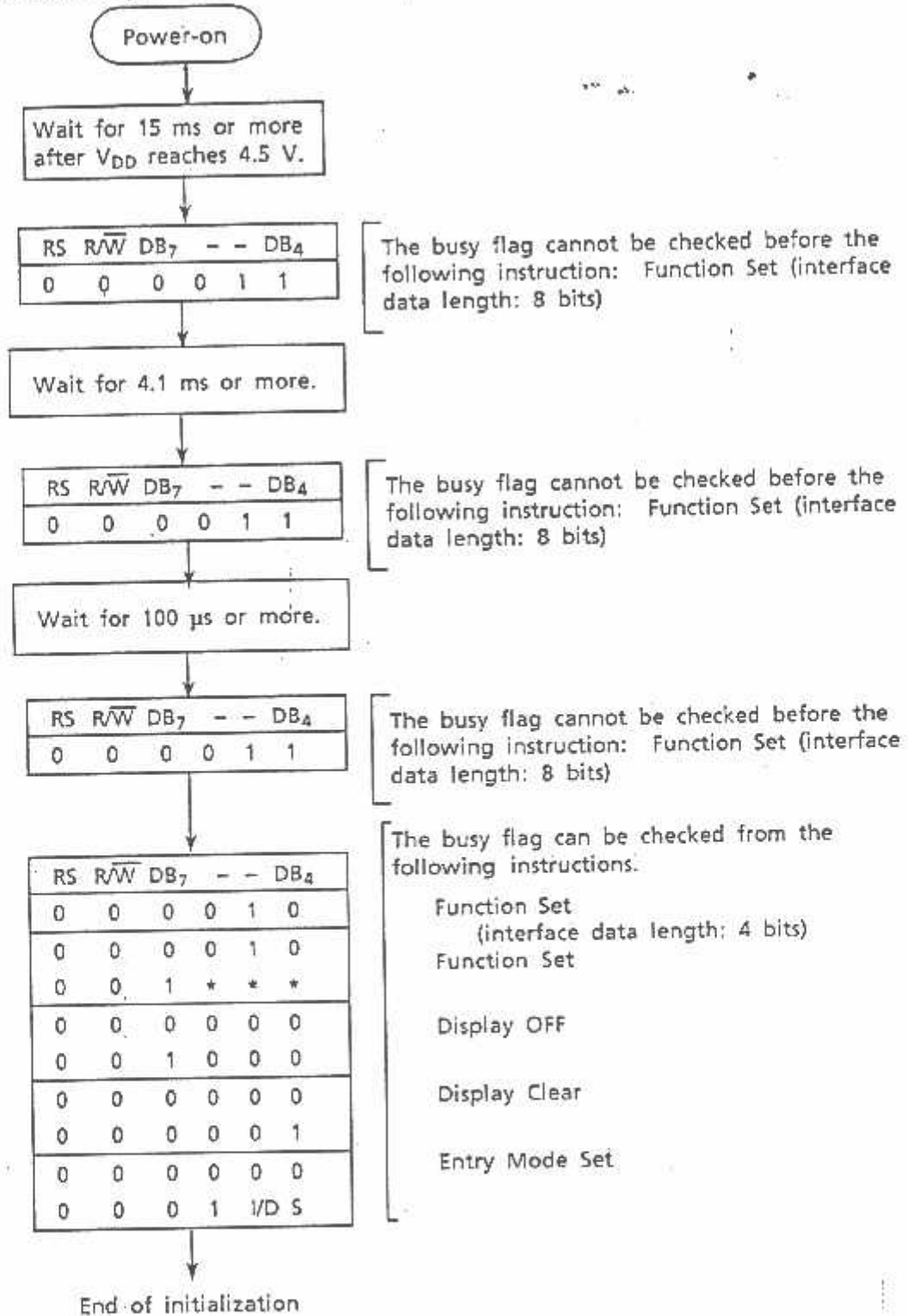
Since the condition is not suitable for the M1632, further function setting is necessary.

If automatic initialization is not executed because the above power supply conditions are not satisfied, use the instruction from next page on.

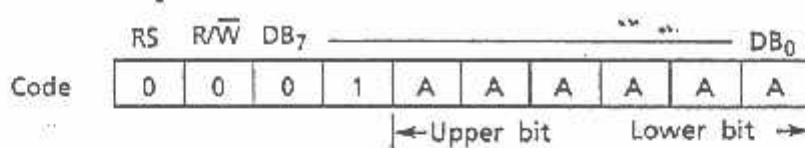
(a) Interface data length : Eight bits



(b) Interface data length: Four bits

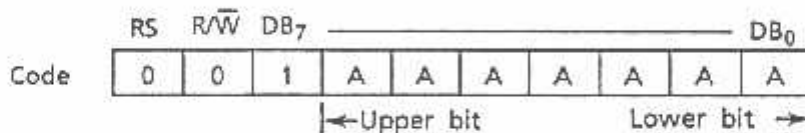


(7) CG RAM Address Set



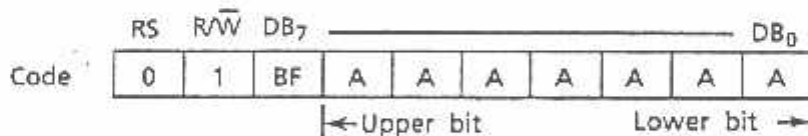
CG RAM addresses expressed as binary AAAAAA are set to the AC. Then data in CG RAM is written from or read to the MPU.

(8) DD RAM Address Set



DD RAM addresses expressed as binary AAAAAAA are set to the AC. Then data in DD RAM is written from or read to the MPU. The addresses used for display in line 1 (AAAAAAA) are 00_H to 27_H and those for line 2 (AAAAAAA) are 40_H to 67_H.

(9) Busy Flag/Address Read



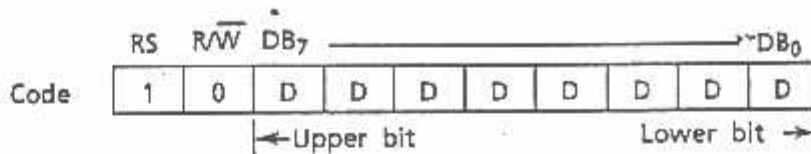
The BF signal is read out, indicating that the module is working internally because of the previous instruction.

When BF = 1, the module is working internally and the next instruction cannot be accepted until the BF value becomes 0.

When BF = 0, the next instruction can be accepted.

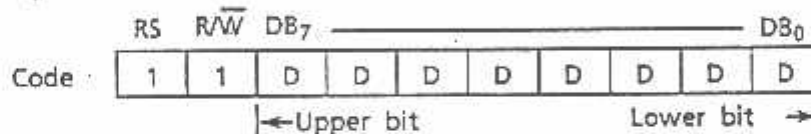
Therefore, make sure that BF = 0 before writing the next instruction. The AC values of binary AAAAAAA are read out at the same time as reading the busy flag. The AC addresses are used for both CG RAM and DD RAM but the address set before execution of the instruction determines which address is to be used.

(10) Data Write to CG RAM or DD RAM



Binary eight-bit data DDDDDDDD is written into CG RAM or DD RAM. The CG RAM Address Set instruction of (7) or the DD RAM Address Set instruction of (8) before this instruction selects either RAM. After the write operation, the address and display shift are determined by the entry mode setting.

(11) Data Read from CG RAM or DD RAM



Binary eight-bit data DDDDDDDD is read from CG RAM or DD RAM. The CG RAM Address Set instruction of (7) or the DD RAM Address Set instruction of (8) before this instruction selects either RAM. In addition, either instruction (7) or (8) must be executed immediately before this instruction. If no address set instruction is executed before a read instruction, the first data read becomes invalid. If read instructions are executed consecutively, data is normally read from the second time. However, if the cursor is shifted by the Cursor Shift instruction when reading DD RAM, there is no need to execute an address set instruction because the Cursor Shift instruction does this.

After the read operation, the address is automatically incremented or decremented by one according to the entry mode, but the display is not shifted.

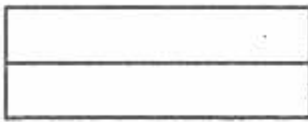
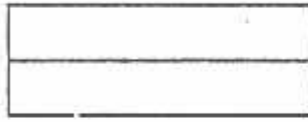


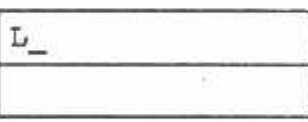
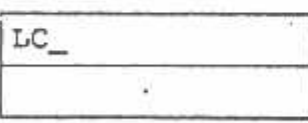
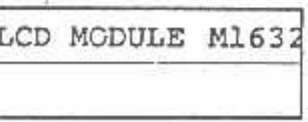
Note : The AC is automatically incremented or decremented by one according to the entry mode after a write instruction is executed to write data in CG RAM or DD RAM. However, the data of the RAM selected by the AC are not read out even if a read instruction is executed immediately afterwards.

Correct data is read out under the following conditions.

- An address set instruction is executed immediately before readout.
- For DD RAM, the Cursor Shift instruction is executed immediately before readout.
- The second, or later, instruction is executed in consecutive execution of read instructions.

2.6 Examples of Instruction Use

(1) Interface data length: Eight bits

No.	Instruction	Display	Operation										
1	Power-on <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>/</td> <td>/</td> <td colspan="3">/</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	/	/	/				The built-in reset circuit initializes the module.
RS	R/W	DB ₇	—	DB ₀									
/	/	/											
2	Function Set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 1 1 1</td> <td>*</td> <td>*</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	0	0	0 0 1 1 1	*	*		The interface data length is set to 8 bits. The character format becomes 5 x 7 dot-matrix at 1/16 duty cycle.
RS	R/W	DB ₇	—	DB ₀									
0	0	0 0 1 1 1	*	*									
3	Display ON/OFF Control <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 0 0 0</td> <td>1 1 1 0</td> <td></td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	0	0	0 0 0 0 0	1 1 1 0			The display and cursor are turned ON, but nothing is displayed.
RS	R/W	DB ₇	—	DB ₀									
0	0	0 0 0 0 0	1 1 1 0										
4	Entry Mode Set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 0 0 0</td> <td>0 1 1 0</td> <td></td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	0	0	0 0 0 0 0	0 1 1 0			The address is incremented by one and the cursor shifts to the right in a write operation to internal RAM. The display is not shifted.
RS	R/W	DB ₇	—	DB ₀									
0	0	0 0 0 0 0	0 1 1 0										
5	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 1 0 0 1</td> <td>1 1 0 0</td> <td></td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	1	0	0 1 0 0 1	1 1 0 0			L is written. The AC is incremented by one and the cursor shifts to the right.
RS	R/W	DB ₇	—	DB ₀									
1	0	0 1 0 0 1	1 1 0 0										
6	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 1 0 0 0</td> <td>0 0 1 1</td> <td></td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	1	0	0 1 0 0 0	0 0 1 1			C is written.
RS	R/W	DB ₇	—	DB ₀									
1	0	0 1 0 0 0	0 0 1 1										
7													
8	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 0 1 1 0</td> <td>0 0 1 0</td> <td></td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	1	0	0 0 1 1 0	0 0 1 0			2 is written in digit 16. Cursor disappears.
RS	R/W	DB ₇	—	DB ₀									
1	0	0 0 1 1 0	0 0 1 0										

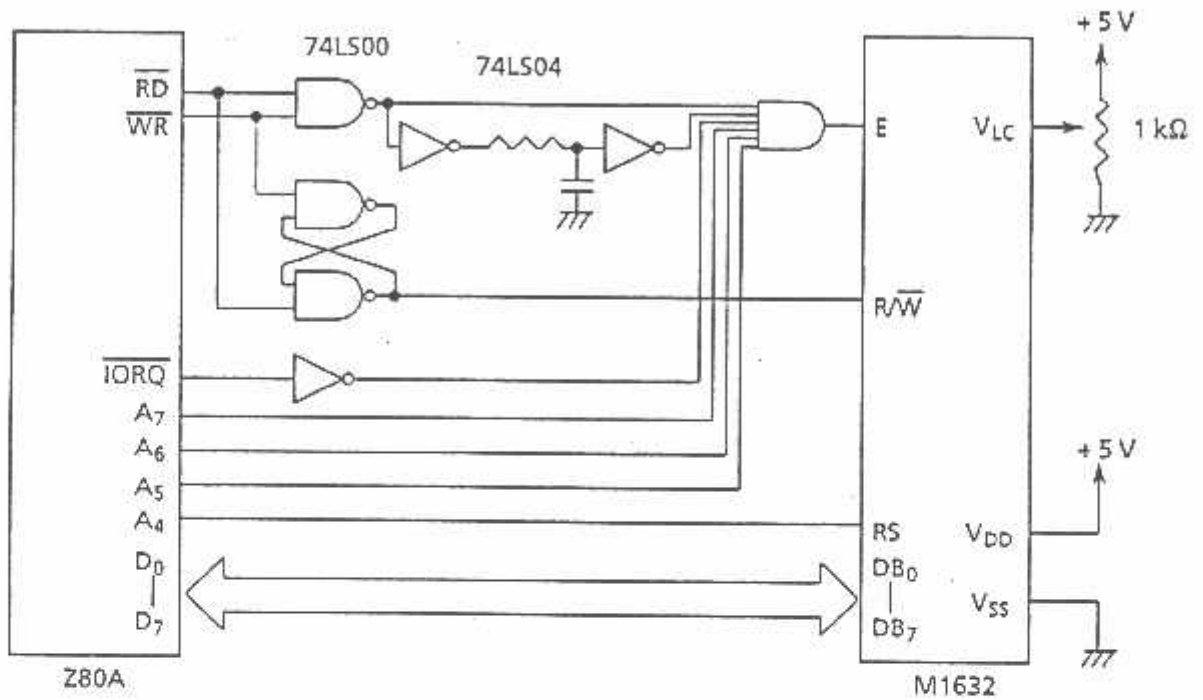
No.	Instruction	Display	Operation										
9	DD RAM address set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>0 0 0 0 0 0 0</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	0	0	1	1	0 0 0 0 0 0 0	LCD MODULE M1632 —	The DD RAM address is set so that the cursor appears at digit 1 of line 2.
RS	R/W	DB ₇	—	DB ₀									
0	0	1	1	0 0 0 0 0 0 0									
10	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 0 1 1</td> <td>0 0 0 1</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	1	0	0 0 1 1	0 0 0 1	LCD MODULE M1632 1_	1 is written.	
RS	R/W	DB ₇	—	DB ₀									
1	0	0 0 1 1	0 0 0 1										
11	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 0 1 1</td> <td>0 1 1 0</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	1	0	0 0 1 1	0 1 1 0	LCD MODULE M1632 16_	6 is written.	
RS	R/W	DB ₇	—	DB ₀									
1	0	0 0 1 1	0 1 1 0										
12													
13	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 1 0 1</td> <td>0 0 1 1</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	1	0	0 1 0 1	0 0 1 1	LCD MODULE M1632 16DIGITS, 2LINES	5 is written.	
RS	R/W	DB ₇	—	DB ₀									
1	0	0 1 0 1	0 0 1 1										
14	DD RAM address set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>0</td> <td>0</td> <td>1 0 0 0</td> <td>0 0 0 0 0</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	0	0	1 0 0 0	0 0 0 0 0	LCD MODULE M1632 16DIGITS, 2LINES	The cursor returns to the home position.	
RS	R/W	DB ₇	—	DB ₀									
0	0	1 0 0 0	0 0 0 0 0										
15	Display clear <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇</td> <td>—</td> <td>DB₀</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 0 0</td> <td>0 0 0 0 0 1</td> </tr> </table>	RS	R/W	DB ₇	—	DB ₀	0	0	0 0 0 0	0 0 0 0 0 1	—	All the display disappears and the cursor remains at the home position.	
RS	R/W	DB ₇	—	DB ₀									
0	0	0 0 0 0	0 0 0 0 0 1										
16													

(2) Interface data length: Four bits

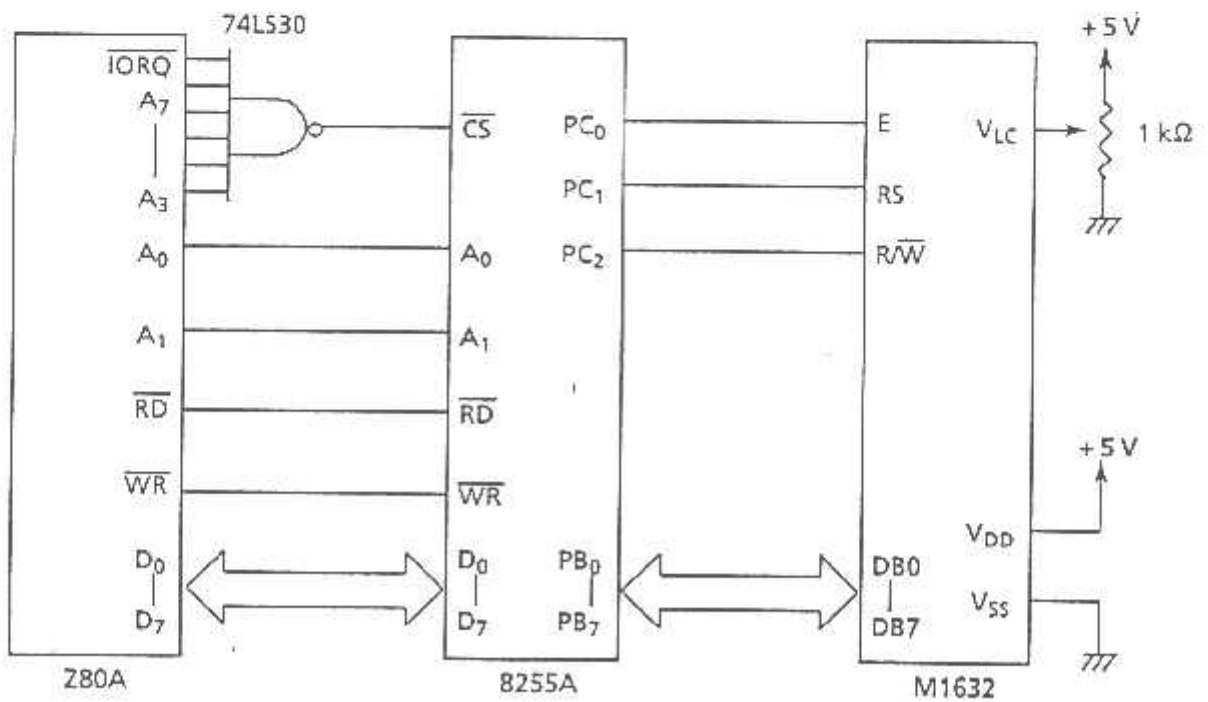
No.	Instruction	Display	Operation									
1	Power-on <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇ — DB₄</td> </tr> <tr> <td>/</td> <td>/</td> <td>/</td> </tr> </table>	RS	R/W	DB ₇ — DB ₄	/	/	/		The built-in reset circuit initializes the module.			
RS	R/W	DB ₇ — DB ₄										
/	/	/										
2	Function Set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇ — DB₄</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 1 0</td> </tr> <tr> <td>/</td> <td>/</td> <td>/</td> </tr> </table>	RS	R/W	DB ₇ — DB ₄	0	0	0 0 1 0	/	/	/		Four-bit operation mode is set. *Eight-bit operation mode is set by initialization, and the instruction is executed only once.
RS	R/W	DB ₇ — DB ₄										
0	0	0 0 1 0										
/	/	/										
3	Function Set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇ — DB₄</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 1 0</td> </tr> <tr> <td>0</td> <td>0</td> <td>1 * * *</td> </tr> </table>	RS	R/W	DB ₇ — DB ₄	0	0	0 0 1 0	0	0	1 * * *		The 4-bit operation mode, 1/16 duty cycle, and 5 x 7 dot-matrix character format are selected. Then 4-bit operation mode starts.
RS	R/W	DB ₇ — DB ₄										
0	0	0 0 1 0										
0	0	1 * * *										
4	Display ON/OFF Control <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇ — DB₄</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 0 0</td> </tr> <tr> <td>0</td> <td>0</td> <td>1 1 1 0</td> </tr> </table>	RS	R/W	DB ₇ — DB ₄	0	0	0 0 0 0	0	0	1 1 1 0		The display and cursor are turned ON, but nothing is displayed.
RS	R/W	DB ₇ — DB ₄										
0	0	0 0 0 0										
0	0	1 1 1 0										
5	Entry Mode Set <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇ — DB₄</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 0 0 0</td> </tr> <tr> <td>0</td> <td>0</td> <td>0 1 1 0</td> </tr> </table>	RS	R/W	DB ₇ — DB ₄	0	0	0 0 0 0	0	0	0 1 1 0		The address is incremented by one and the cursor shifts to the right in a write operation to internal RAM. The display is not shifted.
RS	R/W	DB ₇ — DB ₄										
0	0	0 0 0 0										
0	0	0 1 1 0										
6	Write to CG RAM or DD RAM <table border="1"> <tr> <td>RS</td> <td>R/W</td> <td>DB₇ — DB₄</td> </tr> <tr> <td>1</td> <td>0</td> <td>0 1 0 0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1 1 0 0</td> </tr> </table>	RS	R/W	DB ₇ — DB ₄	1	0	0 1 0 0	1	0	1 1 0 0		L is written. the AC is incremented by one and the cursor shifts to the right.
RS	R/W	DB ₇ — DB ₄										
1	0	0 1 0 0										
1	0	1 1 0 0										

2.7 MPU Connection Diagrams

2.7.1 Z80A



2.7.2 Z80A and 8255A



BAB II

PENGETAHUAN TENTANG PESAWAT OV-10 BRONCO

Karakteristik pesawat OV-10 Bronco

1. Data umum

- a. Type : OV-10
- b. Penggunaan : Tempur taktis
- c. Pabrik : The North American Rockwell
- d. Motor : Dua buah motor jet jenis turbo propeller T-76 G buatan Garret Air Search Amerika Serikat.
- e. Propeller : Menggunakan propeller jenis Hamilton standart dengan three blade
- f. Oil : MIL-L-3699, MIL-L-7808, ASTO 500, ESCO. Kapasitas oil system adalah 2,25 galon 1.5 US galon di oil tank
- g. Oxygen : MIL-C-2720 dengan tekanan 1800 psi. isi botol 514 inch³.
- h. Nitrogen : MIL-N-G Oil untuk pengisian press pada landing gear 75 psi dan wheel tyre 65 psi

2. Kemampuan pesawat

- a. Awak pesawat : Dua orang
- b. Penumpang : Lima orang pasukan/500 kg barang
- c. Daya angkut : 3200 lbs
- d. Kecepatan maximum : 380 knot
- e. Kecepatan jelajah : 180-200 knot
- f. Tinggi terbang : 17600 feet
- g. Senjata : YEA 7,62 mm dimodifikasi menjadi 13 mm yang terletak pada dua sponson

- h. Roket : LAU 10 4 buah dengan diameter 50" LAU 3 9 buah dengan diameter 2,75"
 - i. Bomb : Fire bomb BLU-I CLB
3. Avionik
- a. Komunikasi : VHF/FM : VHF 20 B
VHF/FM : 622 A
 - b. Navigasi : Compass : AN/ASN 75
TACAN : ARN/118
4. Listrik
- a. 28 volt DC
 - b. 115 Volt AC / 400 Hz
5. Environmental system. Suatu alat / sistem agar awak pesawat dalam kondisi baik. Terdiri dari :
- a. Head and vent system (system AC di cockpit)
 - b. Rain removal system (system penghapus di kaca depan)
 - c. Anti Gsuit system (system untuk mengatasi pengaruh gravitasi bagi penerbang).

Airframe pesawat OV-10 Bronco

6. Sebenarnya bagian-bagian pesawat itu sama, tetapi karena pesawat banyak macam dan jenisnya maka bagian-bagian pesawat itu berbeda. Untuk pesawat OV-10 bagian-bagiannya meliputi :
- a. Fuselage
 - b. Wing section
 - c. Lateral control system
 - d. Longitudinal control system
 - e. Directional control system
 - f. Empenage
 - g. Sponson
 - h. Engine nacelle/Boom assy
 - i. Flight control surface
 - j. Wing flap control surface
 - k. Selector landing system

- l. Landing gear selector handle
- m. Selector handle down
- n. Selector handle up

1) Fuselage(Body). Konstruksi wing pesawat ini adalah semi monocoque yang ditempelkan pada wing dan diperkuat dengan empat buah bolt dan fitting dan dirivet.

Fuselage dibagi menjadi dua bagian yaitu :

- a) Nose section, meliputi :
 - (1) Cockpit pilot
 - (2) Cockpit observer
 - (3) Penyangga nose landing gear (nose wheel cover) yang terbuat dari fiber glass. Dan masuk kecockpit dilengkapi dengan tangga yaitu untuk mempermudah membuka dan menutup canopy glass dari luar. Dan dilengkapi juga dengan 4 buah pintu cockpit depan dan belakang masing-masing dua buah pintu.

- b) Tail section, meliputi :
 - (1) Sambungan wing ke badan
 - (2) Sambungan body ke sponson dan cargobay. Dan di bawah pilot observer dipasang plat dari alumunium alloy dengan tebal 0,5" untuk penahan peluru jenis senjata ringan.

2) Wing section. Pada pesawat OV-10 type wing adalah "High Wing Metal Construction" yang terdiri dari 5 bagian yaitu :

- a) Satu buah center wing (wing yang terletak di tengah). Pada center wing terdapat self sealing fuel tank yang berguna apabila terkena tembakan peluru, bagian tersebut akan merapat sendiri sehingga bahan bakar tidak akan mengalami kebocoran. Center wing juga merupakan

tempat melekatkan 2 buah inboard wing flap selain tempat melekatkan fuselage bagian belakang yaitu boom assy. Control surface dan engine control juga terletak disini tepatnya di belakang leading edge spar.

b) Dua buah intermediate wing (kanan dan kiri). Pada intermediate wing panel terdapat dua buah self sealing fuel cell dan juga melekat satu self sealing pada fuel tank. Disini juga diletakkan out board wing flap serta dua bagian spoiler (kanan dan kiri) yang berguna untuk membantu pesawat saat melakukan gerakan rolling. Gerakan spoiler ini mengikuti gerakan aileron.. Jika pada aileron dalam posisi up maka spoiler juga up, jika aileron dalam posisi down maka spoiler akan tetap pada posisinya. Pada aileron terdapat pula trim tab yang berguna untuk membantu pilot mengembalikan kestabilan pesawat jika terjadi unbalance baik miring ke kanan ataupun miring ke kiri. Di intermediate wing ini juga diletakkan pylon senjata tepatnya di bawah jack point.

c) Dua buah outer wing (kanan dan kiri) yang merupakan tempat melekatnya aileron dan wing tip yang terbuat dari bahan fiber glass. Juga diletakkan lampu yang berguna untuk keperluan saat terbang.

3) Lateral control system. Yang termasuk dalam System ini diantaranya adalah sebagai berikut :

a) Control ini menjadi satu gerakannya dengan aileron dan spoiler sebelah kanan.

b) Gerakan antara aileron dan spoiler sebelah kiri berlawanan dengan gerakan aileron dan spoiler sebelah kanan

c) Gerakan aileron up 35° dan down 35° , gerakan aileron spring down 20° .

d) Spoiler up 82° dari permukaan wing.

e) Gear aileron tab up 25° dan down 25°

f) Aileron dan spoiler bisa digerakkan dari control stick pilot, juga bisa digerakkan dari control stick observer melalui bellcrank, push pull rod cable, termasuk spring tab dan gear tab.

g) Spring tab bekerja sebagai pendorong permukaan secara aerodinamic menggunakan udara yang ada di aileron.

h) Apabila spring membelok maximum, maka tab akan menggerakkan drive linkage menyentuh top, dan kedua aileron bergerak berlawanan.

i) Antara kedua aileron dipasang balance cable untuk membantu gerakan sinkron dengan tegangan cable 50 lbs.

j) Keempat plat spoiler berhubungan langsung dengan aileron.

k) Apabila aileron bergerak ke atas maka plat spoiler akan ikut naik sehingga dapat mengurangi gaya angkat dan hal ini menyebabkan wing menjadi turun.

4) Longitudinal control system. bagian-bagian Yang termasuk longitudinal control system ini adalah:

a) Control ini dipergunakan melalui stick dari cockpit observer melalui drive linkage, bell crank, push pull rod cable yang dihubungkan dengan spring tab yang ada di trailing edge elevator.

elevator.

b) Tegangan cable yang terletak di cargo bay ke depan (5 pounds)

c) Tegangan cable yang ada pada permukaan atas wing baik kanan maupun kiri ± 6 pounds.

5) Directional control system.

a) Control ini dilengkapi dengan alat yang disebut jaw damper yang berguna untuk menjaga kelurusan arah terbang pesawat pada saat take off dan pada saat di udara.

b) Rudder deflection left 35° , right 35° dan digerakkan secara bersamaan.

c) Tegangan cable rudder adalah 80 ± 5 pounds dan di boom 100 ± 5 pounds.

d) Diameter cable $1/5$ sampai $1/8$ inch.

e) Control ini digerakkan dari cockpit observer dengan menggunakan rudder pedals yang dihubungkan dengan push pull rod dan cable control ke rudder.

f) penyesuaian panjang pendeknya kaki kita bisa disesuaikan yaitu dengan cara mengatur dengan menggunakan screw adjustment.

g) Gerakan rudder dengan electric mengatur push button switch yang ada di control stick, secara normal dan selector yang ada di control detection sebelah kiri pilot, secara alternatif. Kedua alat itu menggerakkan actuator yang ada di dinding sebelah kanan dari cargo bay dengan bantuan alternate.

h) Dengan trim actuator dioperasikan ke kiri ataupun ke kanan, atau digerakkan oleh rudder pedals dengan gerakan kebalikan.

i) Tegangan cable dilaksanakan pada

- 6) Empenage. Empenage atau bidang ekor terdiri dari :
- a) Dua buah vertical stabilizer/fin yang melekat pada ujung belakang boom kanan dan kiri, dan pada vertical stabilizer dipasang dua buah rudder.
 - b) Horizontal stabilizer dimana kedua ujungnya melekat pada ujung atas kedua vertical stabilizer tempat melekatnya rudder.
- 7) Sponson. Konstruksi sponson tidak termasuk dalam susunan fuselage pesawat OV-10 Bronco. Dan susunan sponson terdiri dari spr, tib dan shin. Tiap sponson dilekatkan ke badan pesawat dengan 3 buah bolt pada station 106,3. Dan 3 buah bolt lagi pada station 204. Dan pada sponson sebelah kanan dan kiri masing-masing dilengkapi juga dengan tiga buah senjata kaliber 12,7 mm. Spar depan dan belakang bisa membawa amunisi sekitar 600 pounds. Didalam suatu sponson ditempatkan senjata seberat 600 pounds. Dan dua pucuk senjata dengan kaliber 7,62 mm dengan 500 butir peluru. Sebuah pintu besar dari sponson digunakan untuk memasang dan digunakan untuk perawatan senjata. Tiap sponson diletakkan ke fuselage dengan tiga buah bolt pada station 186,3 dan tiga lagi pada station 204.
- 8) Engine nacelle/ Boom assy. Pada pesawat OV-10 ini mempunyai dua buah engine nacelle dan boom assy dimana yang satu dengan yang lain bisa ditukar serta masing - masing nacelle diletakkan satu buah engine. Engine nacelle terletak pada station 131,60 sampai dengan 256,00. Dan boom assy terletak pada station 204-25. Boom yang merupakan kontriksi tempat melekatnya horizontal stabilizer yaitu bagian belakang. Dan kedua ujungnya untuk menyangga jalanya cable control surface dan

electrical wiring.

9) Flight control system. Pada pesawat OV-10 Bronco ini bidang kemudi utama adalah aileron, rudder, elevator dan semua bidang kemudi tersebut dapat dicontrol dari cockpit yaitu dengan system bellcrank, cable, pulley, push pull rod dan bungee. Untuk trim digerakkan dengan system electric mekanik. Flight control ini mengontrol pesawat pada tiga sumbu sedangkan wing flap adalah suatu system yang digunakan untuk mendapatkan tambahan gaya angkat yang besar pada waktu take off dan pada waktu landing serta diperlulan sebagai airbrake. Pergerakan flap sebagai penghasil gaya angkat dan sebagai penghasil drag tergantung dari sudut buka bidang flap. Yang dimaksud tiga sumbu pesawat adalah :

- a) Lateral control system ini untuk mengontrol pesawat pada sumbu longitudinal. System ini digerakkan dengan menggunakan control stick yang dicontrol oleh spoiler plate, spring tab dan gear tab.
- b) Longitudinal control system untuk mengontrol pesawat pada sumbu lateral. Control ini digerakkan dengan menggunakan control stick yang dicontrol oleh elevator spring dan elevator gear tab.
- c) Directional control system untuk mengontrol posisi pesawat pada vertical axis. Control ini digerakkan dengan menggunakan rudder pedals dan selanjutnya akan menggerakkan rudder.

10) Wing flap control system. Wing flap terdiri dari empat buah yaitu :

- a) 2 buah inner flap
- b) 2 buah outer flap.

Keempat wing flap tersebut bergerak bersama-sama dengan perantaraan drive shaft. Drive shaft ini digerakkan

secara hydraulic pressure serta menggunakan electric motor yang ada di gear box. Keuntungan pemakaian wing flap yaitu :

- a) Lift yang besar adalah dari wing flap double slotted.
- b) Slot door dan wing flap dihubungkan dengan linkage untuk membuka wing atau down position dan menutup wing flap atau up position.
- c) Lift yang paling besar didapat pada waktu wing flap down 40° dan pesawat dalam keadaan berkecepatan penuh.
- d) Flap handle terdapat di console sebelah kiri.
- e) Flap handle terdapat tanda up, hold, T.D dan down.
- f) Flap deflection up 20° dan down 40°
- g) Penempatan handle pada up atau down berarti menggerakkan hydraulic system.

11) Selector landing system.

- a) Operator wing flap cargo door harus dalam posisi tertutup untuk menghindari cargo door itu sendiri.
- b) Pada waktu handle flap ditempatkan pada posisi down maka wing flap juga harus down dan apabila flap handle ditempatkan pada up maka wing flap juga harus up.
- c) Posisi handle pada waktu take off harus 20°
- d) Flap handle harus ada pada posisi hold apabila kita menggerakkan flap dengan menggunakan alternate flap switch dan dalam hal ini gerakan wing flap akan lebih lambat.

12) Landing gear selector handle. Landing gear

selector handle terletak di sebelah kiri control panel pilot dan observer. Menggerakkan landing gear handle berarti menggerakkan selector valve yang berada di cargo bay.

13) Selector handle up. Tekan release lock dan letakkan pada posisi up lock sehingga selector pada posisi up juga yang akan menyebabkan menutupnya pil cooler door.

14) Selector handle down. Tekan release lock yang kemudian letakkan pada aileron dan gerakan ini diperingan oleh balance yang terletak di bawah selector handle. Hubunganya selain membuka landing gear up lock dan membuka hydraulic selector down, juga membuka oil cooler door.

Engine System OV-10 Bronco

7. Pesawat OV-10 Bronco menggunakan engine buatan Garret Air Search Amerika Serikat. Yaitu buatan pabrik North American Rockwell dan propeller jenis Hamilton Standart. Jenis engine ini adalah turboprop artinya engine dari jenis turbo lalu penghasil thrust dihasilkan dari putaran propeller. Dengan kata lain putaran dari poros yang dihasilkan oleh turbine digunakan untuk memutar propeller. Arah putaran turbine sama dengan putaran jarum jam (CW) jika dilihat dari belakang. Sedangkan arah putaran propeller satu dengan yang lain saling berlawanan. Karena pada pesawat OV-10 Bronco ini menggunakan dua buah engine. Hal ini dimaksudkan untuk menghilangkan twist. Propeller kiri berputar searah jarum jam dan propeller kanan berputar berlawanan dengan jarum jam (CCW). Putaran kedua engine sama, yaitu searah dengan jarum jam sedangkan putaran poros propeller berlawanan. Engine kiri bernomor genap yaitu T-76 G-416 dan mempunyai berat 310 lbs. Engine kanan bernomor ganjil yaitu T-76 G-417 dan mempunyai berat 310 lbs. Engine ini berkekuatan 1715 shp. Bahan bakar yang digunakan yaitu JP-4 atau JP-5 (avtur) dan juga dipakai bahan bakar

BAB III

PENGETAHUAN TENTANG PESAWAT CASA 212

Karakteristik pesawat CASA 212

1. Data umum
 - a. Engine type : TPE 331-10-510 C
 - 1) Buatan : Garret Air Search USA
 - 2) Berat engine : 380 lbs
 - 3) Berat pesawat : 7640 kg
 - b. Propeller type : Hartzell propeller model
 - 1) Berat : 75 kg
 - 2) Buatan : Hartzell
 - c. Bahan bakar : JP 4 (avtur 50)
 - 1) Isi total : 2000 lt
 - 2) Terdiri dari : Inboard tank : 1400 lt
outboard tank : 600 lt
 - d. Panjang sayap : 19 m
 - e. Panjang pesawat : 15,2 m
 - f. Tinggi pesawat : 6,30 m
 - g. Daya engine : 900 HP
 - h. Kecepatan pesawat : 145 - 150 knot
 - i. mission :
 - 1) Untuk angkut ringan
 - 2) Untuk SAR
 - 3) Untuk pembuatan hujan
 - 4) Untuk dropping
 - 5) Untuk terjun
 - 6) Untuk pemotretan

Pesawat ini adalah buatan IPTN Bandung.

Airframe CASA 212

2. Airframe pesawat CASA ini dibagi menjadi 4 bagian, yaitu:
 - a. Fuselage. Fuselage pada pesawat CASA ini termasuk jenis semi monocoque yang terdiri dari yaitu :
 - 1) Bulkhead
 - 2) Frame

- 3) Longeron
- 4) Skin
- 5) Stringer
- 6) Former

Pada konstruksi semi monocoque ini beban atau stress diterima oleh seluruh konstruksi. Pada fuselage pesawat ini dibagi menjadi tiga bagian, yaitu :

- 1) Nose section/forward section. Yang terdiri dari:
 - a) Nose landing gear
 - b) Cockpit
 - c) Lavatory
 - d) Electrical cabinet

Pada bagian depan nose landing gear terpasang landing light dan taxi head lamp. Juga bagian ini bisa dibuka untuk memudahkan pengecekan pada hydraulic system dan nose steering system. Pada bagian cockpit terdapat wind shield yang terdiri dari :

- a) 3 di depan
- b) 2 di samping kanan & kiri

Sebuah pada bagian kiri dan kanan bisa digeser sebagai jendela.

- 2) Center section (frame 1-17). Bagian dalam center fuselage ini sebagai main cabin atau ruang cargo. Bagian atas dari center fuselage ini terpasang wing pada frame nomor 6 & 8. Sedangkan bagian luar dari center fuselage terpasang main landing gear pada frame no 8 & 9.

- 3) Tail section/rear section (frame 17-33). Pada rear fuselage bagian atas adalah tail cone dan bagian bawah terdapat cargo & ramp door. Door dan window pesawat ini terdiri atas 11 window dan 4 door.

4 door antara lain :

- a) Main cabin door
- b) Crew door
- c) Emergency door/pintu darurat
- d) Cargo and ramp door

11 window antara lain :

- a) 5 buah di kiri
- b) 6 buah di kanan

Ukuran door :

- a) Main cabin door dengan ukuran 158 x 70 cm atau 62 x 27,6 inch.
- b) Crew door dengan ukuran 57 x 110 cm atau 22,5 x 43,3 inch.
- c) Emergency door dengan ukuran 91 x 50 cm atau 38,5 x 19,7.

Emergency door ada dua, yaitu :

- a) Upper emergency exit dengan ukuran 74 x 60 cm atau 29 x 23,6 inch.
- b) Lower emergency exit dengan ukuran 76 x 70 cm atau 29,9 x 23,6 inch.

f. Wing. Pada pesawat CASA ini konstruksi wing adalah termasuk High Wing Metal Construction Cantilever Wing. Yaitu jenis sayap yang terletak di atas fuselage. Struktur sayap pesawat CASA ini adalah Multi Spar Trapezoidal planform. Sayapnya dibagi menjadi tiga bagian, yaitu :

- 1) 1 buah center wing
- 2) 2 buah outer wing

Wing ini mempunyai tiga bagian spar, yaitu :

- 1) 1 buah di bagian depan tepatnya di depan chord line rib q 25%.
- 2) 1 buah di bagian tengah yaitu pada chord line rib q 52,5%.
- 3) 1 buah di bagian belakang yaitu pada chord line rib q 68,5%.

Wing ini selain sebagai penghasil gaya angkat juga antara spar bagian tengah dan depan diletakkan fuel tank

(integral tank) sampai ujung sayap 46 cm (18 inch) dari wing tip.

1) Center wing. mempunyai ukuran panjang 7m (23 ft) dengan 4 buah bolt pada frame no 6 dan 11 dari center fuselage. Pada leading spar station 2380 dan 2900 terletak engine mounting. Kemudian pada station 1676 dan 2300 terpasang bracket untuk inner flap (80,1%) chord line. Di bagian atas center wing terdapat dua buah hoisting point untuk mengangkat sayap. Sambungan antara center wing dan outer wing adalah pada station 3500 dan dihubungkan dengan sejumlah baut pada bagian atas dan bawah sayap.

2) Outer wing. Bagian ini menempel pada center wing sebelah kanan dan kiri.

Station numbering adalah suatu garis khayal yang ditentukan oleh pencipta pesawat tersebut. Station numbering ini terdapat pada fuselage, wing dan stabilizer. Dan semua ukuran diukur dengan "inch" yang dimulai "0" (zero point). Gunanya station numbering adalah untuk memudahkan peletakan komponen-komponen pada pesawat dan juga untuk memudahkan penentuan letak center of gravity pada pesawat tersebut.

c. Flight control. Flight control pada pesawat ini dibagi menjadi dua, yaitu : primary control dan secondary control. Semua flight control dapat dicontrol dari cockpit pesawat. Dan untuk primary control dibuat dengan double system. Gerakan-gerakan flight control ini melalui komponen : control column-bell crank-cables-pulley-gears-control rod. Kecuali wing flap digerakkan dengan hydraulic system. Pesawat ini dilengkapi juga dengan gust lock lever yang berguna untuk mengelock primary control pada kedudukan netral saat berada di darat. Wing flap ini juga digerakkan dengan cara mekanik dengan menggunakan hand pump.

1) Aileron control system. Aileron pada pesawat

terbang Casa ini hanya aileron sebelah kiri saja yang dilengkapi dengan trim tab. Pergerakan aileron ini pada posisi up 20° dan pada posisi down 25° . Tegangan cablenya 176 lbs atau 80 kg pada temperature 15° C dengan diameter cable $5/32$ inch. Komponen-komponen aileron control system adalah control column, connecting rod, aileron quadrant, pulley, pulley wheel, control bar dan aileron control wheel.

2) Elevator control system. System kemudi ini dilengkapi dengan trim tab dan servo tab, baik pada elevator sebelah kanan maupun sebelah kiri. System ini gerakanya melalui : control rod, control column, quadrant arm, quadrant angle, elevator surface. Gerakan ini pada posisi up adalah 30° dan pada posisi down 20° . Sedangkkn tegangan cablenya adalah 176 lbs pada temperature 15° dengan diameter $5/32$ inch.

3) Rudder control system. Rudder terpasang pada fin dan dicontrol dari flight deck oleh dua pasang pedals yang dapat disetel. System kemudi rudder ini dilengkapi dengan trim tab dan servo tab (sebelah atas). Pergerakan rudder ini pada posisi sebelah kanan 35° dan pada posisi sebelah kiri 35° sedangkan tegangan cablenya 198 lbs pada temperature 15° C dengan diameterr cablenya $5/32$ inch. Komponen-komponen rudder control ini adalah pedals, control rod, cable sigment, pulley dan quadrant shaft.

4) Tabs control system. Semua primary control dilengkapi dengan tab control system unuk membantu gerakan-gerakan dari control tersebut. Control tab umumnya ada tiga yaitu spring tab, balance tab, dan trib tab dan balance tab. Untuk pesawat Casa memakai rudder trim dan balance tab, Aileron trim dan balance tab, dan elevator trim dan balance tab. Semua control tab dicontrol dari cockpit melalui

hand wheel yang terletak pada pasestall bagian atas dan samping kiri serta kanan. Gerakan dari hand wheel diteruskan oleh cable ke pulley kemudian ke jack screw dan ke tab surface. Diameter untuk tab control adalah 2,4 mm. Gerakan tab untuk masing-masing control adalah :

- a) Untuk aileron
up 7° ; netral 0° ; down 21°
- b) Untuk elevator
up 17° ; netral 0° ; down 29°
- c) Untuk rudder
 3° 15' left ; 35° 21' right

5) Flap control system. Flap adalah suatu bidang yang digunakan untuk menghasilkan gaya angkat tambahan dimana diperlukan pada saat akan take off dan pada saat pesawat akan landing. Flap control pada pesawat Casa ini digerakkan dengan menggunakan dua system, yaitu :

- a) Memakai hydraulic pressure dari hydraulic pump.
- b) Memakai mechanic force system.

Jika memakai hydraulic pressure, maka hydraulic pressure ini hanya akan sampai di actuator. Untuk selanjutnya gerakan diteruskan melalui bell crank, intermediate bell crank, control rod, dan operating rod. Wing flap pesawat Casa ini adalah type double slotted wing flap. Wing flap ini masing-masing mempunyai dua buah center bell crank. Gerakan-gerakan wing flap yaitu :

- a) Untuk take off 20° .
- b) Untuk approach 40° .
- c) Untuk landing 80° .

d. Landing gear. Pesawat Casa mempunyai fixed three cycle landing gear. Landing gear pesawat Casa terdiri dari :

1) Dua main landing gear yang terpasang di center fuselage. Tiap main landing gear dilengkapi dengan telescopic OLEO pneumatic shock absorber. Type wheel memakai disk brake.

a) Shock absorber strut. Typenya telescopic OLEO pneumatic strut yang terdiri baja yang diametrnya tidak sama antara bagian luar dan dalamnya. Di dalam cylinder bagian luar ada dinding. Dinding pemisah tersebut terdapat restrictor dan piston drill. Di bagian bawah pada kedua cylinder dihubungkan dengan torque link yang berguna untuk menjaga agar jangan sampai cylinder bagian luar berputar. Di atas cylinder bagian luar terdapat oil filling valve dan bleed valve.

b) Wheel assembly. Wheel terdiri dari dua buah alloy light half wheel. Salah satu diantaranya sebagai housing dalam disk plate dengan dipasang menjadi satu dengan sejumlah baut sampai batas tanda merah.

c) Brake assy. Terdidri dari cylinder block dengan thrust blade terpasang menjadi satu. Pada cylinder block dipasang moving friction parts. Dan pada thrust blade dipasang static friction parts. Kemudian disk plate terlatak antara moving parts dan static parts. Moving parts dapat bergerak dengan perantaraan hydraulic pressure. Torque wheel 1035 kg/cm dan 345 kg/cm. Bearing 1035 kg/cm dikendorkan. Dan untuk main wheel ditorque 345 kg/cm. untuk nose wheel dengan torque 200 kg/cm. Pemasangan brake assembly dengan torque 1280 kg/cm dan axelnya bolt 6 ea.

2) Nose landing gear. Nose landing gear ini dilengkapi dengan nose steering hydraulic system.

Fuel System Casa 212

3. Fuel tank pesawat Casa terdiri dari :
- a. 2 main fuel tank
 - b. 2 outer tank
 - c. 1 collector tank
 - d. 1 pilot tank
 - e. 1 vent tank

Fuel yang digunakan adalah MIL-T-5624, JP-4 atau JP-5 dan JP-1. Maksud dan tujuannya adalah untuk menjamin positif flow pada fuel agar engine beroperasi dengan baik. Komponen dari fuel system adalah :

- a. Float switch. Kalau di collector volumenya kurang dari 70 ltr maka float switch akan bekerja memberikan sensing ke cockpit sehingga warning light menyala.
- b. Drain valve. Gerakan dari drain valve ini adalah push dan push turn. Fuel drain valve ini terletak pada fuel flow divider. Gunanya adalah untuk membuang sedikit fuel pada waktu start dimana fuel yang merupakan sisa yang tidak terbakar dan selesai mematikan engine.
- c. Micro filter. Berfungsi membersihkan bahan bakar setelah melalui booster pump untuk dialirkan ke engine fuel system. Letak dari micro filter di bagian kanan pada station code 12D dan di bagian kiri pada station code 11 I.
- d. Free check shut off valve. Fungsinya untuk mempercepat pengisian pilot tank dan untuk mengetahui pilot tank sudah penuh atau belum.
- e. Refueling valve. Berguna pada waktu pengisian agar pengisian tersebut tidak merupakan tekanan dan untuk menyetop pengisian bila sudah penuh.
- f. Booster pump. Yaitu untuk mempositifkan fuel yang masuk ke system compressor. Letak booster pump adalah 2

di sebelah kanan dengan station code 54 D dan 2 di sebelah kiri dengan station code 53 I.

Jika booster pump rusak dan pressure-nya juga rendah maka warning light panel akan menyala.

g. Fuel solenoid valve. Untuk usable operating refueling. Total fuel capacity internal tank 1400 ltr. Penggunaan fuel yang mempunyai grade yang berlainan dengan standart diperbolehkan pada emergency dan batasan tertentu. Yaitu 80-87 octan, 1000 US galon dalam 100 jam. Pressure refueling 45-50 psi. Dengan pressure refueling maka hanya inner tank kanan dan kiri yang terisi oleh fuel. Untuk fuel line pada fuel system digunakan pipa aluminium yang dipasang pada fuel tank dan fuel shut off valve.

h. Safety valve. Untuk membocorkan kelebihan pressure yang dihasilkan oleh booster pump dan mengalirkannya menuju internal tank.

i. Fuel shut off valve. Untuk menutup aliran fuel dari tank ke engine. FSO akan menutup aliran dengan jalan fuel feeding lever pada shut off valve. Fuel metering valve mengatur kebutuhan bahan bakar yang kerjanya diatur oleh below assy.

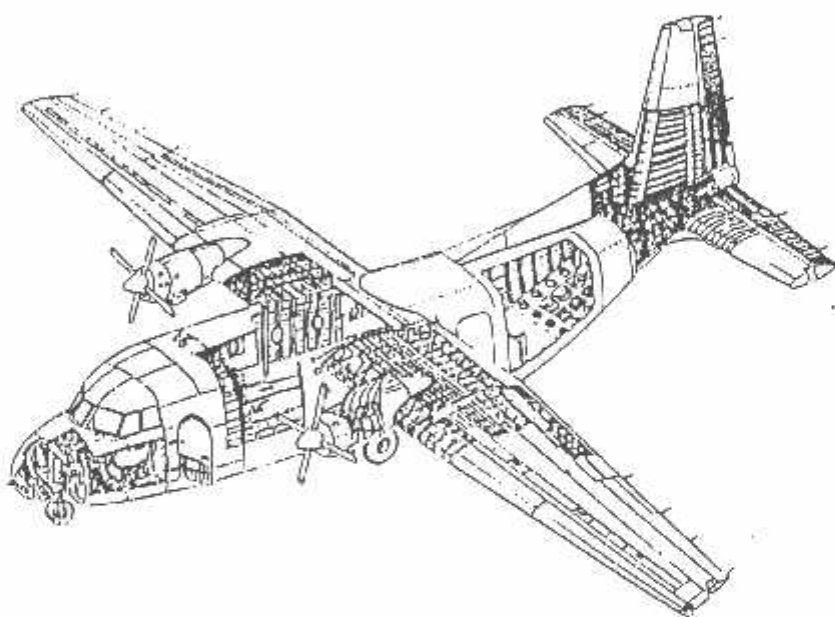
j. Fuel control differential pressure regulator. Fungsinya adalah untuk membuka start pressure regulator untuk penambahan fuel pada waktu start.



OPERATIONS MANUAL
C.212 Avlocar Model C.212-CC



AIRFRAME



FUSELAGE

The fuselage is a stressed skin semi-monocoque structure of aluminum alloy. It consists

of three sections, forward center and aft. Electrical bonding and anti-corrosion treatment have been provided as necessary.



OPERATIONS MANUAL

C.212 Aviocar

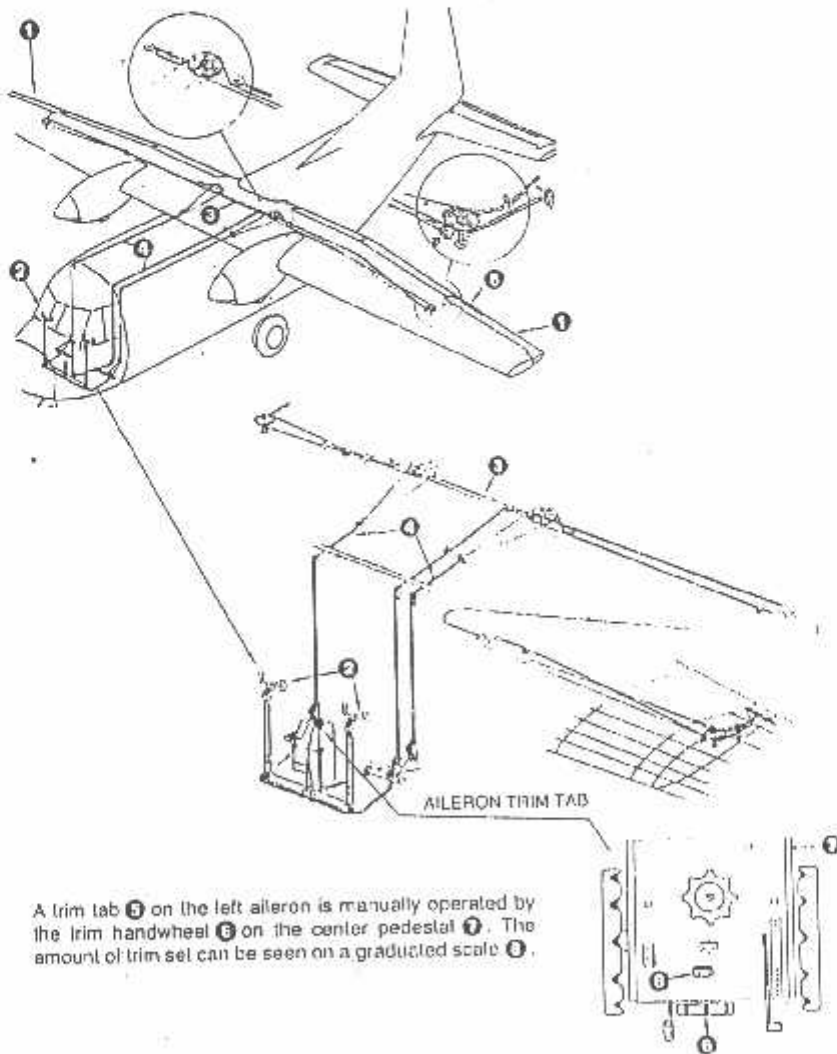
Model C.212-CC



AILERONS

Ailerons **1** provide roll control effective into the stall. The aileron control wheels **2** are in-

terconnected by rods and a synchronization bar **3**. This, together with a duplicate cable system **4**, permits either pilot to have control if the opposite set of cables fail.



A trim tab **5** on the left aileron is manually operated by the trim handwheel **6** on the center pedestal **7**. The amount of trim set can be seen on a graduated scale **8**.

SECTION 8 SYSTEMS DESCRIPTION

Part 2 Airframe & Flight Controls

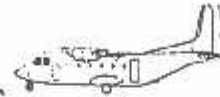
Page 0-2-0
Date: 1 JAN 83

Printed by Jeppesen Sandersen



OPERATIONS MANUAL

C.212 Avlocar Model C.212-CC

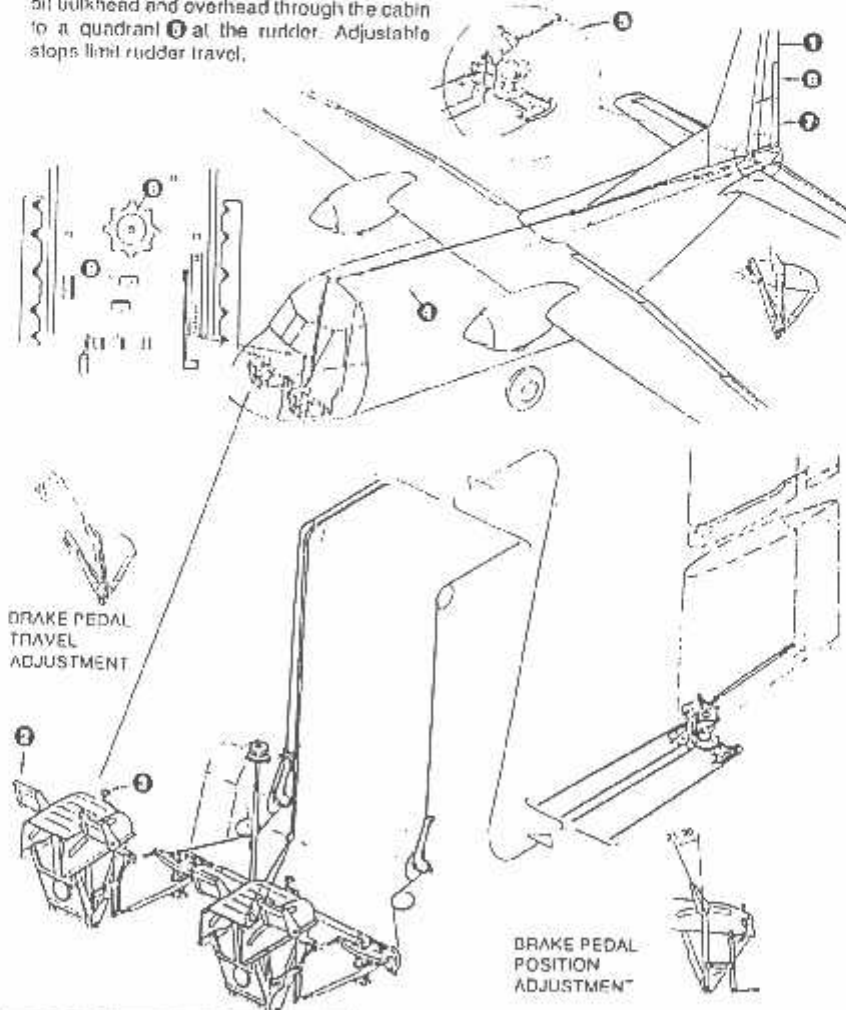


RUDDER

Rudder ① provides yaw control through a pair of pedals ② which are interconnected. This allows either pilot control through synchronized pedals. The pedals are adjustable for rake by a crank handle ③ on the pedal assembly.

A single cable system ④ passes up the cockpit bulkhead and overhead through the cabin to a quadrant ⑤ at the rudder. Adjustable stops limit rudder travel.

The rudder has a servo ⑥ and trim tab ⑦. The trim tab is the inner tab operated from the cockpit by the rudder trim handwheel ⑧. A graduated scale ⑨ indicates the amount of rudder trim set.





OPERATIONS MANUAL

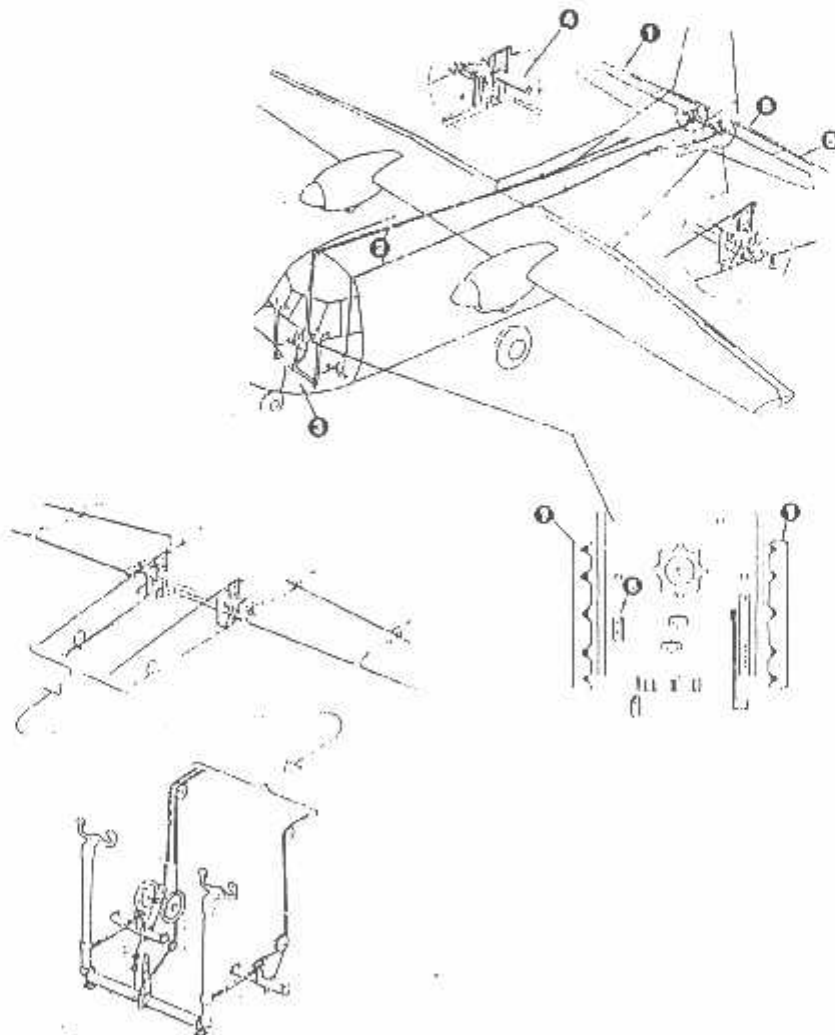
C.212 Aviocar Model C.212-CC



ELEVATORS

Elevators ① provide pitch control. A duplicate cable system ② and an interconnection at the yoke ③ and the elevators ④ allow either pilot control if one set of cables fail.

Each elevator has a servo ⑤ and trim tab ⑥. The trim tab is the inner tab operated from the cockpit by the pitch trim handwheel ⑦. A graduated scale ⑧ indicates the amount of nose up or nose down trim set.





OPERATIONS MANUAL

C.212 Aviocar Model C.212-CC



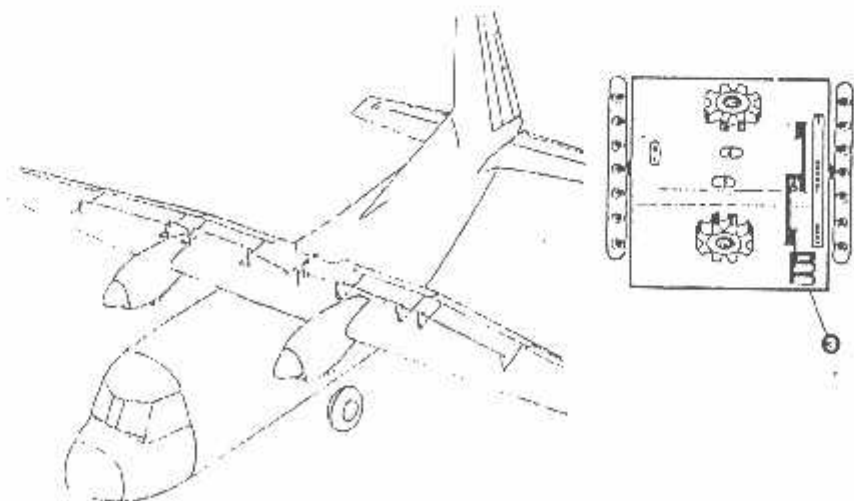
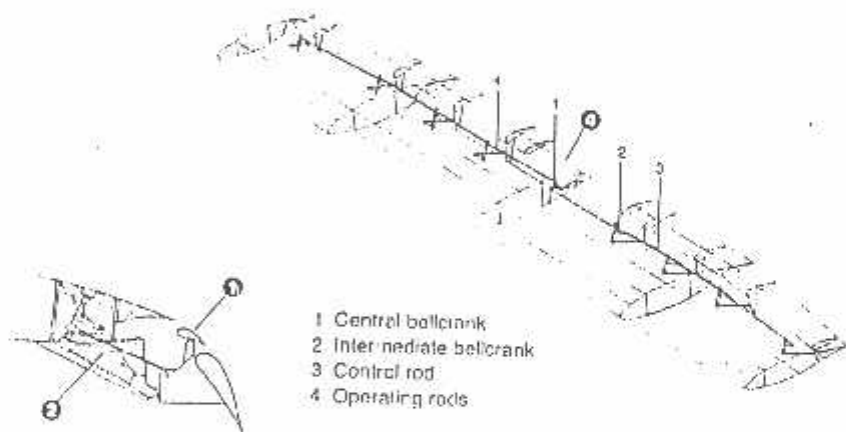
FLAPS

Flaps are double slotted **1** with maximum extension of 40° and are adjustable at the extension rods **4**.

They are hydraulically activated by a lever **6** on the center pedestal which controls movement of a cylinder in the overhead cabin.

A mechanical linkage of rods and bellcranks **2** extends from the cylinder to actuate the flaps. This linkage incorporates protection against an asymmetric flap condition.

Flaps should fully extend or retract within 15 seconds.





OPERATIONS MANUAL

C.212 Aviocar

Model C.212-CC

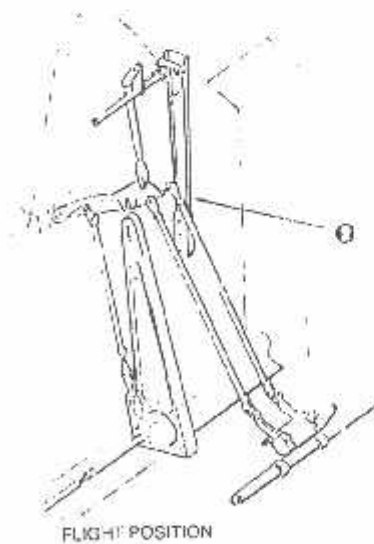
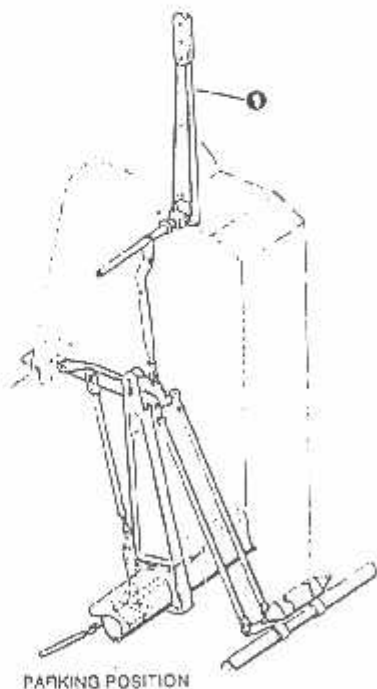


GUST LOCK

A gust lock lever which engages the primary flight controls is mounted on the co-pilot side of the center pedestal.

To engage the lever ❶ center the rudder pedals, elevators and ailerons, pull up the lever and engage in the lock position. Any resistance to the lever indicates the controls are not correctly centered.

To release the lever ❷, pull out of the holding socket and drop anti-clockwise towards the floor.



SECTION 8 SYSTEMS DESCRIPTION

Part 2 Airframe & Flight Controls

Page 8-2-12

Date 1 JAN 83

Printed by Jayproan Sinteron

LM78XX Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expended to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the out-

put, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

Features

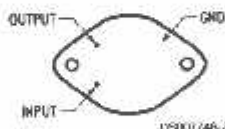
- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

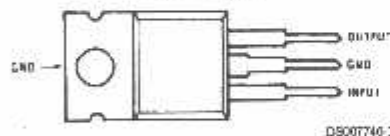
Connection Diagrams

**Metal Can Package
TO-3 (K)
Aluminum**



**Bottom View
Order Number LM7805CK,
LM7812CK or LM7815CK
See NS Package Number KC02A**

**Plastic Package
TO-220 (T)**



**Top View
Order Number LM7805CT,
LM7812CT or LM7815CT
See NS Package Number T03B**

LM78XX

Rangkaian Regulator Voltage

Gambaran Umum

Seri LM78XX dari tiga regulator terminal tersedia dengan beberapa pengatur voltage output yang menjadikan mereka bermanfaat dalam bermacam-macam aplikasi. Salah satunya adalah regulasi lokal yang mungkin, mengeliminir masalah-masalah distribusi yang diasosiasikan dengan regulasi bertitik tunggal. Voltage-voltage yang ada menyediakan regulator-regulator yang bisa digunakan di dalam sistem logika, instrumentasi, HiFi, dan keadaan peralatan elektronik yang solid. Walaupun didesain sebagai regulator voltase yang telah ditentukan, maka bagian-bagian ini bisa digunakan dengan komponen-komponen eksternal untuk mendapatkan voltase dan arus yang bisa disesuaikan.

Seri LM78XX tersedia dalam sebuah paket aluminium TO-3s yang akan membiarkan muatan arus di atas 1.0A jika tersedia panas yang cukup. Pembatasan arus dimaksudkan untuk membatasi arus output puncak sampai pada nilai yang aman. Proteksi daerah aman untuk transistor output

bertujuan untuk membatasi pemborosan power internal. Jika pemborosan power internal terlalu tinggi untuk panas yang tenggelam, maka stop kontak untuk menghentikan panas mencegah IC mengalami panas yang berlebihan.

Usaha-usaha yang mungkin telah dikembangkan untuk membuat seri regulator LM78XX mudah digunakan dan mengurangi jumlah komponen luar. Tidak perlu menghindari output, walaupun hal ini akan meningkatkan respon yang bersifat sementara. Input bypass dibutuhkan hanya jika regulator tersebut diletakkan jauh dari filter kapasitor power supply.

Selama output voltase berbeda dari 5V, 12V dan 15V maka seri LM117 menyediakan sebuah voltase output yang berjarak antara 1.2V sampai 57V.

Ciri-ciri

- Arus output lebih dari 1A
- Panas internal melebihi beban proteksi.
- Tidak ada komponen-komponen eksternal yang dibutuhkan
- Proteksi daerah aman transistor output
- Batas arus internal sirkuit pendek
- Terdapat pada paket aluminium TO-3

Jarak voltase

LM7805C	5V
LM7812C	12V
LM7815C	15V

Keterangan gambar

1. Resistansi panas paket T)-3 (K, KC) merupakan persimpangan ke kotak dan kotak 35° C sampai ambient. Resistansi panas paket T)-220 (T) merupakan persimpangan 4° C ke kotak dan kotak 50° C sampai pada ambient.
2. semua karakter diukur dengan kapasitor melintasi input 0.22 μ F, dan kapasitor output 0.1 μ . semua karakteristik kecuali voltase yang memiliki bunyi dan perbandingan rejeksi reaksi suara diukur dengan menggunakan teknik-teknik pulse ($t_w \leq 10$ ms, lingkaran tugas $\leq 5\%$). Perubahan voltase output disebabkan perubahan-perubahan di dalam temperatur internal harus di masukkan ke dalam catatan secara terpisah.
3. Rating maksimum yang mutlak menunjukkan batas-batas yang melebihi kerusakan yang terjadi pada bagian