SKRIPSI

PERANCANGAN DAN PEMBUATAN ALAT PENGONTROL DIMMER LAMPU DALAM RUANGAN BERBASIS MIKROKONTROLLER AT89S52

Disusun Oleh :

SIMON RICHARD

02 17 088

KONSENTRASI TEKNIK ELEKTRONIKA

JURUSAN TEKNIK ELEKTRO S-1

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL

MALANG

2009

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN ALAT PENGONTROL DIMMER LAMPU DALAM RUANGAN BERBASIS MIKROKONTROLLER AT89S52

SKRIPSI

Diajukan Untuk Memenuhi salah satu Syarat Memperoleh Gelar Sarjana Teknik Pada Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika

Oleh :

Simon Richard

02 17 088

Malang, April 2009

Diperiksa dan Disetujui,	Mengetahui,
Dosen Pembimbing	**Ketua Jurusan Teknik Elektro S-1**
I Komang Somawirata ST, MT	**Ir. F. Yudi Limpraptono, MT**
NIP.P.1030100361	**NIP.Y.1039500274**

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

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa	: SIMON RICHARD
NIM	: 02.17.088
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Judul Skripsi	:

" PERANCANGAN DAN PEMBUATAN ALAT PENGONTROL DIMMER LAMPU DALAM RUANGAN BERBASIS MIKROKONTROLLER AT89S52 "

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

Hari	: Senin
Tanggal	: 23 Maret 2009
Dengan Nilai	: B+ (72,6)

Panitia Ujian Skripsi

Ketua Majelis Penguji	Sekretaris Majelis Penguji
(Ir. Sidik. Noertjahjono, MT)	**(Ir. F. Yudi Limpraptono, MT)**
NIP.Y.1028700163	**NIP.Y. 1039500274**

Anggota Penguji :

Penguji I	Penguji II
(Ir. F.Yudi Limpraptono, MT)	**(Irmalia Suryani Faradisa ST, MT)**
NIP.Y.1039500274	**NIP.P.1030100365**

PERANCANGAN DAN PEMBUATAN ALAT PENGONTROL DIMMER LAMPU DALAM RUANGAN BERBASIS MIKROKONTROLLER AT89S52

SIMON RICHARD
Jurusan Teknik Elektronika S-1. Institut Teknologi Nasional, Malang
Email: richard_rumaropen@yahoo.com

Abstrak

Pada makalah ini telah direalisasikan sebuah Alat Pengontrol Dimmer Lampu Dalam Ruangan Berbasis Mikrokontroler AT89S52. dengan metode memudahkan dalam penggunaanya. Untuk merealisasikan pengontrolan lampu dalam ruangan maka diperlukan beberapa alat dan rangkaian penunjang, untuk menghidupkan dan mematikan lampu diperlukan dua sensor, yaitu sensor infrared masuk dan sensor infrared keluar serta microphones condenser sebagai pengontrol redup terangnya lampu. Pada pengontrolan dimmer lampu, nantinya akan lebih memanjakan dan memudahkan pengguna, guna menghidupkan, mematikan dan mengontrol redup terangnya lampu didalam suatu ruangan, serta lebih meminimalisir penggantian atau penggunaan lampu lampu dengan watt yang lebih kecil, juga dilengkapi dengan LCD yang dapat memonitor apakah lampu tersebut On, menyala redup, menyala terang. Kita juga dapat mengetahui jumlah orang didalam ruangan tersebut serta penggunaan mode apakah yang sedang dipakai, apakah mode penerangan manual atau mode penerangan dengan sensor suara (tepuk) melalui LCD tersebut

Diharapkan perancangan dan pembuatan Alat Pengontrol Dimmer Lampu Dalam Ruangan Berbasis Mikrokontroler AT89S52 dapat meminimalisir pergantian lampu dengan watt yang lebih rendah, serta lebih memudahkan pengguna dalam pengoperasiannya

Kata Kunci : Pengontrol Dimmer Lampu Dalam Ruangan Berbasis Mikrokontroler AT89S52

Abstract

At this handing have been reslized a Comptroller of Dimmer Lamp In Room Base on Mikrokontroler AT89S52. with method facilitate for using. To realize a controler of lamp in room hence needed some supporter network and appliances, to start and put-off the light to be needed two censor, censor of infrared enter and censor of infrared out and also condenser microphones as gloomy lamp controler brightness. for supervison of lamp dimmer, later will be more pamper and facilitate consumer to animate, turn-off and controlling gloomy of lamp brightness in a room, and also more to minimized replacement or usage of lamps with smaller watt, is also provided with LCD able to monitor if lamp is On, gloomy, blazing bold. We also can know the amount of people in the room and also usage of mode what is wearing, is mode manual or with voice censor (clap) pass the LCD display.

Expected by scheme and making of Comptroller of Dimmer Lamp In Room Base on Mikrokontroler AT89S52 have been minimized commutation of lamp with lower watt, and also more facilitate consumer in the operation.

Keyword : Controling Dimmer Lamp In Room Base on Mikrocontroler AT89S52.

KATA PENGANTAR

Terima kasih!, serta puji dan syukur kepada Tuhan Yang Maha Esa atas semua yang telah diberikan dalam hidupku selama ini, karena apapun yang kulakukan takkan bisa seperti dan sejauh ini tanpa ada restuNYA, tak lupa juga saya ingin berterima kasih kepada semua pihak yang sudah membantu dan mendorong saya untuk menyusun skripsi ini dengan judul "PERANCANGAN DAN PEMBUATAN ALAT PENGONTROL DIMMER LAMPU DALAM RUANG BERBASIS MIKROKONTROLER AT89S52", dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika ITN Malang dan untuk mencapai gelar Sarjana Teknik. untuk itu saya ingin berterima kasih kepada:

1. Prof DR Eng. Ir. Abraham lomi , MSEE, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektronika S1.
3. Bapak I Komang Somawirata, ST, MT Selaku Dosen Pembimbing
4. Kedua Orang Tuaku, Bpk Alfons Rumaropen dan Ibu Nurhasanah atas semua yang diberikan, Doa, perhatian, kasih sayang, dukungan, kepercayaan dan motivasi. serta segala kebutuhanku yang sudah dipenuhi selama ini.

5. Adik adikku yang tersayang, Christien, Fernando, Melinda, tak lupa "my little Angie" (*thanks for singing 'kasih ibu, pelangi pelangi" everytime I call"*).
6. Diah Ayu Fitria (Thanks For *Motivation, Support, and Pray*).
7. Simon Sofian Lakamola beserta keluarga besar disumba.
8. Semua Jajaran, Dosen, Laboratorium, Himpunan, rekan mahasiswa serta pegawai yang berada dalam lingkup Institut Teknologi Nasional Malang.
9. Serta Teman teman se-perjuangan yang sudah membantu secara langsung maupun tidak langsung dalam penyelesaian Skripsi ini

Apa yang telah penyusun buat semaksimal mungkin, tak luput dari kekurangan dan keterbatasan untuk menyelesaikan laporan ini. Untuk itu penyusun mengharapkan kritik, saran dari pembaca yang bersifat membangun, guna kesempurnaan laporan ini.

Dan berharap semoga apa yang sudah penyusun buat dapat memberikan manfaat bagi perkembangan IPTEK. dan pembaca pada umumnya.

Malang, April 2009

penyusun

Terima kasih Wahai Penguasa Bumi dan Jagad Alam ini Atas Berkah, Rahmat dan HidayahMU, bahkan nafas ini adalah kemurahanMU..

Tak lepas dari semua itu saya ingin mengucapkan terima kasih kepada,

***Kedua Orang tua tercinta* ♥ Bpk Alfons Rumaropen & Ibu Nurhasanah**

♥ **(Christien (katin), Fernando (Katet) & wife, Melinda (Gendut), little kariting 'angie', Simon Cilik)** Dihatiku hanya ada namamu, Arti cintamu terukir dihati, Falling in love I'm with you.. *Doaku Semoga Kalian Selalu dilindungi Oleh* Tuhan Yang Maha Esa.. *Amin!!!*

(TAK PERLU TAKUT DENGAN PERUBAHAN DAN TAK PERLU KUATIR DENGAN YANG TELAH PERGI KARENA SATU-SATUNYA ALASAN UNTUK HIDUP ADALAH MENIKMATINYA)

Big Family

Pakde Soekardi sekeluarga <sedayu>, Pakde Solikhin sekeluarga <solo>, Pakde Sutris sekeluarga <krebet>, Bude Endang sekeluarga <singosari>,Bulek Suyati, Paklek Gianto sekeluarga <solo>, Bulek Solikah sekelurga <solo>, Paklek Untung sekeluarga <solo>, serta Keluarga besar Di BIAK (PAPUA)

♀ Caroline (jayapura) <trima kasih kawan atas semuanya dan persahabatan selama ini>, Aniek (solo) <gak kerasa ya sudah hampir 12th kita berteman>, mbak tutik (solo) <jadilah seorang ibu yang bijak>

- ▶ Dhodon & Wife (Bali)
- ▶ Teman-teman di Tembagapura (PAPUA)

(Elka 3 2002)

♂ Jaka (Jack), Dona (Anod), Arip (Remek) <thanks for sharing, bantuan, ngobrol ngalor-ngidul,ngrasani,solusi, "etc" Hehe>, Rio (bro) <bilyardnya muanstap>, dwi vendi (talk less do more), hendraM, BambangH, Adin, Wahid, dan yang lain nya yang tak dapat kusebut satu persatu, Trima Kasih Buat Kebersamaan kita selama ini dan semoga menjadi seorang "ST" yang berguna.(We Are The Best!!!)

(Other Elka)

♂ Yunanri, ghafur, iwanS, Sinchan, Roni, Danang (blontang), Ovan (01), Helmy, Nugroho <akhirnya terkabulkan juga jadi seorang SARJANA TEKNIK>

(Campus)

♂ P. Sugeng, P.diglam, M.pur, kantin on campus, penjaga parkir,etc..

(Sedayu)

Ψ Bu'e <matur suwun semangat'e, maem'e lan liya-liya'ne>, P.waris+keluarga ,retno, serta orang-orang yang sudah baik kepadaku selama ini (GOD BLESS U'R)

♂ M.Teguh.Y *CONDROMOWO* (kostrad jember) <k-70 nang stadion kanjuruhan pancen oye, sing endi sing durung mbok COR? Hehehe>, EDS&Dwi (polres sorong), CAgungB, Bhakti, ROy, Arip (menot), chandra (condro) <ayo makaryo!>, Hendra (tombro), jefri (jupri) Ludy, Arie (c_p), Iir (kotak), denis (plolong) m.miril (bakso), Fery (p.jen) and more...

(Malang)

De'kus, mbak seh, mbak har, mbak anik, emak & husband, Maz Is (Ngawi), toko sahabat, warung california, mas mul, abah, erin, angel, nira,valin, rahma,dwi,wiwid,mas muji&wife, Dan semua nya yang tak dapat kusebut satu persatu.

...yang terpenting adalah apa yang kamu *kerjakan* dengan apa *yang kamu miliki*,... dan tak seorang manusiapun yang sempurna, jadi jangan pernah merendahkan *orang dan jangan* takut mengakui jika kamu mempunyai masalah..

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dengan perkembangan teknologi yang pesat pada jaman ini telah dibuktikan dengan banyaknya muncul teknologi dan penemuan alat baru dibidang elektronika. ada juga teknologi lama, namun diperbaharui serta di lengkapi dengan berbagai macam fiture dan aplikasi sehingga lebih mempermudah pengguna dalam menggunakan dan mengoperasikan nya, lampu merupakan salah satu dari sekian banyak kebutuhan yang sangat kita perlukan, sehingga dalam pengoperasian dan penggunaan nya dituntut bisa lebih mudah, praktis dan tentunya aman buat si pengguna

Dalam kehidupan sehari-hari kita tak pernah lepas dari lampu penerangan yang banyak macam dan kegunaan nya. Ada pun penggunaannya cukup sederhana, dengan menekan tombol *On* untuk menghidupkan dan *Off* untuk mematikan. Salah satu yang tak bisa di hindari ketika mata kita jenuh dengan terang dan menginginkan penerangan yang redup, akankah menganti lampu dengan Watt yang lebih rendah adalah cara yang efesien? Hal ini kurang efektif di era moderen saat ini. Oleh karena itu pada tugas akhir ini dikembangkan alat untuk pengontrol dimmer lampu dalam ruangan, adapun komponen inti dalam perancangan dan pembuatan alat ini adalah; 2 komponen photodiode infrared (*sensor infrared in dan sensor infrared out*), dan 1 *mic condenser* (sensor suara) sebagai pengontrol redup dan terang nya lampu, dalam perancangan alat ini *sensor infrared in* sebagai tombol *on* untuk menyalakan lampu, sedangkan *sensor*

infra out sebagai tombol *off* pada lampu, adapun penggunaan alat ini dapat di aplikasikan pada kamar hotel atau ruangan yang tidak begitu membutuhkan banyak pencahayaan, sebagai cara yang efisien dalam penggunaan nya. Dengan memberi tiga kondisi yaitu *on, redup dan off.*

1.2 Tujuan

Adapun tujuan dari tugas akhir ini adalah untuk merancang dan membuat alat pengontrol dimmer lampu dalam ruang dengan kondisi menyala, redup dan mati.

1.3 Rumusan Masalah

1. Bagaimana cara membuat suatu alat yang dapat mengontrol lampu pada suatu ruangan yang berbasis mikrokontroler
2. Bagaimana membuat software untuk mengolah data yang nantinya dapat digunakan sebagaimana fungsinya yaitu Pembuatan Alat Pengontrol Dimmer Lampu Dalam Ruang Berbasis Mikrokontroller AT89S52

1.4 Batasan Masalah

1. Lampu yang digunakan adalah lampu pijar
2. Penggunaannya diprioritaskan untuk kamar atau ruangan yang berkelas (VIP, VVIP)
3. Tidak membahas catu daya

1.5 Metodologi

- ***Study literature***

  Yaitu dengan melakukan studi kepustakaan dari berbagai sumber untuk memperoleh berbagai teori serta gambaran tentang masalah yang akan dibahas.

- **Perencanaan dan pembuatan alat**

  Perencanaan dan pembuatan alat dsini meliputi :

  1. Perencanaan sistem secara keseluruhan (pembuatan blok diagram sistem)
  2. Mendeskripsikan fungsi dari masing-masing blok diagram
  3. Membuat perangkat keras *(hardware)* dan perangkat lunaknya *(software)*
  4. Implementasi *software* yang telah dirancang ke dalam Mikrokontroler

1.6 Sistematika Penulisan

Adapun sistematika dari penyusunan laporan tugas akhir ini adalah sebagai berikut :

BAB I Berisi latar belakang, tujuan, rumusan masalah, batasan masalah, metodologi dan sistematika penyusunan laporan tugas akhir.

BAB II Membahas tentang dasar teori sistem Membahas teori – teori dasar yang menunjang perancangan tugas akhir ini.

Sistem Mikrokontroler, serta pengaksesannya, Dan Teori – teori lain yang menunjang perancangan tugas akhir ini.

BAB III Membahas tentang perencanaan *Hardware* dan *software* dari sistem yang akan dibuat.

BAB IV Membahas tentang pengujian *hardware* dan *software* dari sistem yang telah dibuat.

BAB V Berisi kesimpulan dan saran

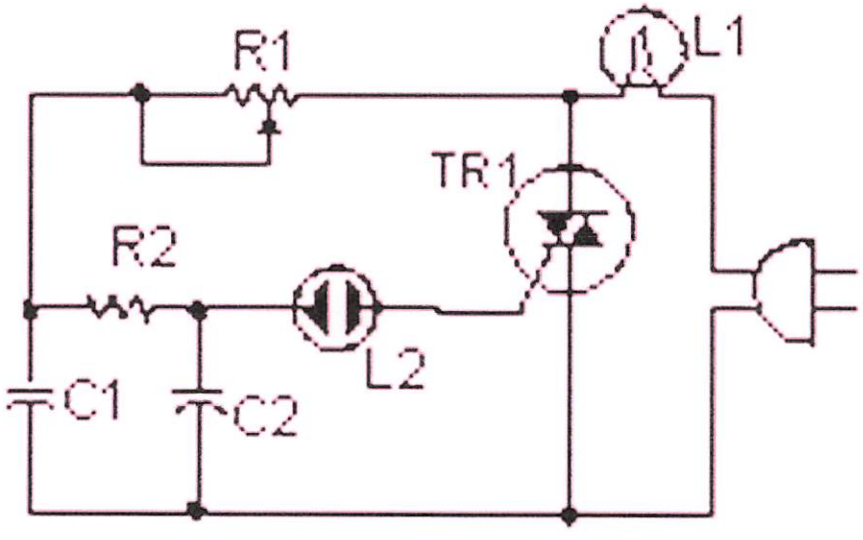
BAB II

LANDASAN TEORI

2.1. Sistem Dimmer

2.1.1. Umum

Dimmer adalah perangkat yang digunakan untuk mengontrol terang redup cahaya yang di sebabkan oleh peningkatan atau penurunan tegangan. Meskipun variabel tegangan perangkat tersebut digunakan untuk berbagai keperluan, istilah lampu *dim* biasanya untuk mengatur pencahayaan, rangkaian dimmer merupakan rangkaian yang sudah umum digunakan, antara lain untuk mengatur terang-redup lampu bohlam.



Gambar 2.1 *Rangkaian Dimmer*

2.2. Sensor

Sensor adalah perangkat yang mengukur besaran fisik dan merubah nilai besaran listrik tersebut dalam bentuk sinyal yang dapat dibaca oleh pengamat atau

tertentu. Hampir seluruh peralatan elektronik yang ada mempunyai sensor didalamnya. Ada juga yang menyebutkan piranti yang mentransform (mengubah) suatu nilai (isyarat/energi) fisik ke nilai fisik yang lain, Menghubungkan antara fisik nyata dan industri electric dan piranti elektronika Di dunia industri berguna untuk monitoring, controlling, dan proteksi, Sering disebut juga dengan Transducer. Pada saat ini, sensor tersebut telah dibuat dengan ukuran sangat kecil dengan orde nanometer. Ukuran yang sangat kecil ini sangat memudahkan pemakaian dan menghemat energi. Sensor yang digunakan dalam sehari-hari objek seperti sensitif sentuhan tombol lift dan lampu yang redup. banyak sekali aplikasi untuk sensor yang kebanyakan orang tidak pernah sadar. Aplikasi termasuk mobil, mesin, aerospace, obat-obatan, manufaktur dan robotics.

2.2.1. Jenis sensor

Ada 2 macam jenis sensor yaitu sensor fisika dan sensor kimia

- **Sensor fisika**

  Sensor fisika mendeteksi besaran suatu besaran berdasarkan hukum-hukum fisika. Contoh sensor fisika adalah sensor cahaya, sensor suara, sensor kimia, , sensor gaya, sensor kecepatan, dan sensor percepatan, dan sensor suhu.

- **Sensor kimia**

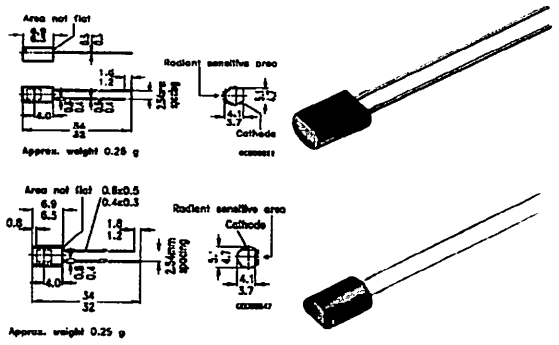
  Sensor kimia mendeteksi jumlah suatu zat kimia dengan cara mengubah besaran kimia menjadi besaran listrik. Biasanya melibatkan beberapa reaksi kimia. Contoh sensor kimia adalah Sensor pH, sensor Oksigen, sensor ledakan, dan sensor gas

Namun sensor yang digunakan pada perencanaan alat ini adalah sensor fisika, yang terdiri dari 2 komponen yaitu:

1. Photodiode infrared
2. Kondensor Microphones

2.2.1.1. Photodiode infrared

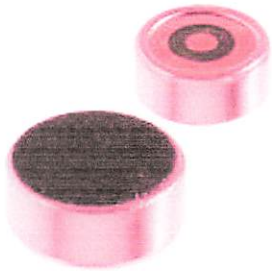
Photodiode pertama kali ditemukan sekitar tahun 1873, Sinar infra merah yang dipancarkan oleh pemancar infra merah tentunya mempunyai aturan tertentu agar data yang dipancarkan dapat diterima dengan baik di receiver. Oleh karena itu baik di transmitter infra merah maupun receiver infra merah harus mempunyai aturan yang sama dalam mentransmisikan (bagian transmitter) dan menerima sinyal tersebut kemudian mendekodekannya kembali menjadi data biner (bagian receiver). Untuk memperoleh jarak yang cukup jauh, dioda infra red memerlukan sinyal dengan frekwensi 30 hingga 50 khz. Berbeda dengan dioda led yang hanya memerlukan tegangan dc saja untuk mengaktifkan led, dioda infra red memerlukan sinyal ac dengan frekwensi 30 hingga 50 khz untuk mengaktifkan nya. Cahaya infra red tersebut tidak dapat di tangkap oleh mata manusia, sehingga diperlukan phototransistor untuk mendeteksinya.



Gambar 2.2 *Photodiode infrared*

2.2.1.2. Condenser Microphones

Perangkat yang mengkonversi gelombang suara menjadi gelombang listrik, Kondensor microphones membutuhkan daya dari baterai atau sumber eksternal. Sinyal suara yang dihasilkan lebih kuat dibandingkan dengan sinyal dinamis. Condensers juga cenderung lebih sensitif dan responsif, sehingga cocok untuk menangkap nuansa halus dalam suara. Kondensor tidak ideal untuk suara yang tinggi, bagaimana *Kondensor Microphones* bekerja? Sebuah kapasitor mempunyai dua piring dengan tegangan diantaranya. Kondensator mic dalam satu piring dibuat dari bahan sangat ringan sebagai diafragma. Diafragma yang bergetar bila terkena oleh suara mengubah jarak antara dua piring dan karenanya mengubah capacitance. Khususnya, bila piring yang dekat bersama, sehingga capacitance meningkat.



Gambar 2.3 *Condenser microphones*

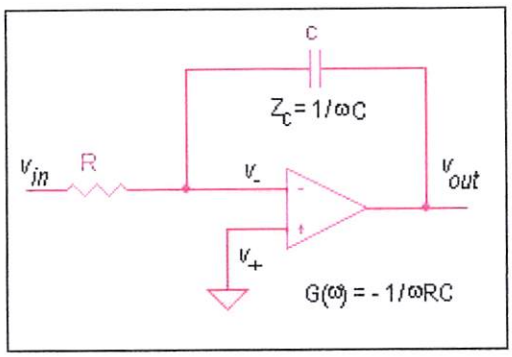
2.3. Penguat Instrumentasi

Operational Amplifier atau di singkat op-amp merupakan salah satu komponen analog yang popular digunakan dalam berbagai aplikasi rangkaian elektronika. Aplikasi op-amp popular yang paling sering dibuat antara lain adalah rangkaian inverter, non-inverter, integrator dan differensiator. Pada pokok bahasan kali ini akan dipaparkan beberapa aplikasi op-amp yang paling dasar, dimana rangkaian feedback (umpan balik) negatif memegang peranan penting. Secara umum, umpanbalik positif akan menghasilkan osilasi sedangkan umpan balik negatif menghasilkan penguatan yang dapat terukur.

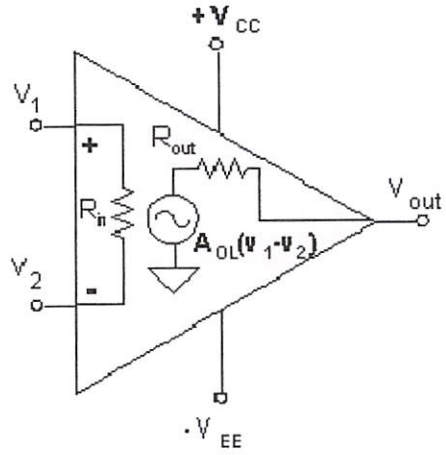
2.3.1. Op-amp Integrator

Op-amp bisa juga digunakan untuk membuat rangkaian-rangkaian dengan respons frekuensi, misalnya rangkaian penapis (filter). Salah satu contohnya adalah rangkaian integrator seperti yang ditunjukkan pada gambar 2.4. Rangkaian dasar sebuah integrator adalah rangkaian op-amp inverting, hanya saja rangkaian umpanbaliknya (*feedback*) bukan resistor melainkan menggunakan capasitor C.

Pada prakteknya, rangkaian *feedback* integrator mesti diparalel dengan sebuah resistor dengan nilai misalnya 10 kali nilai R atau satu besaran tertentu yang diinginkan. Ketika inputnya berupa sinyal dc (frekuensi = 0), kapasitor akan berupa saklar terbuka. Jika tanpa resistor feedback seketika itu juga outputnya akan saturasi sebab rangkaian umpanbalik op-amp menjadi open loop (penguatan open loop opamp ideal tidak berhingga atau sangat besar). Nilai resistor feedback sebesar 10R akan selalu menjamin *output offset voltage* (offset tegangan keluaran) sebesar 10x sampai pada suatu frekuensi *cutoff* tertentu.



Gambar 2.4 : Integrator



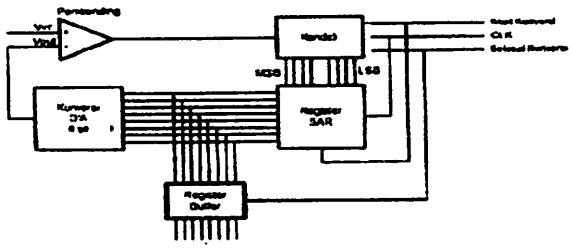
Gambar 2.5 : Diagram schematic simbol Op-Amp

1.4 ADC (Analog-to-Digital Converter) ADC0804

Analog to Digital Converter (ADC) adalah sebuah piranti yang dirancang untuk mengubah sinyal-sinyal analog menjadi bentuk sinyal digital. IC ADC 0804 dianggap dapat memenuhi kebutuhan dari rangkaian yang akan dibuat. IC jenis ini bekerja secara cermat dengan menambahkan sedikit komponen sesuai dengan spesifikasi yang harus diberikan dan dapat mengkonversikan secara cepat suatu masukan tegangan. Hal-hal yang juga perlu diperhatikan dalam penggunaan ADC ini adalah tegangan maksimum yang dapat dikonversikan oleh ADC dari rangkaian pengkondisi sinyal, resolusi, pewaktu eksternal ADC, tipe keluaran, ketepatan dan waktu konversinya. Ada banyak cara yang dapat digunakan untuk mengubah sinyal analog menjadi sinyal digital yang nilainya proposional. Jenis ADC yang biasa digunakan dalam perancangan adalah jenis Successive Approximation Convertion (SAR) atau pendekatan bertingkat yang memiliki waktu konversi jauh lebih singkat dan tidak tergantung pada nilai masukan analognya atau sinyal yang akan diubah. Gambar 2.6 memperlihatkan diagram blok ADC tersebut

Secara singkat prinsip kerja dari konverter A/D adalah semua bit-bit diset kemudian diuji, dan bilamana perlu sesuai dengan kondisi yang telah ditentukan. Dengan rangkaian yang paling cepat, konversi akan diselesaikan sesudah 8 clock, dan keluaran D/A merupakan nilai analog yang ekivalen dengan nilai register SAR. Apabila konversi telah dilaksanakan, rangkaian kembali mengirim sinyal selesai konversi yang berlogika rendah. Sisi turun sinyal ini akan menghasilkan

Secara singkat prinsip kerja dari konverter A/D adalah semua bit-bit diset kemudian diuji, dan bilamana perlu sesuai dengan kondisi yang telah ditentukan. Dengan rangkaian yang paling cepat, konversi akan diselesaikan sesudah 8 clock, dan keluaran D/A merupakan nilai analog yang ekivalen dengan nilai register SAR. Apabila konversi telah dilaksanakan, rangkaian kembali mengirim sinyal selesai konversi yang berlogika rendah. Sisi turun sinyal ini akan menghasilkan data digital yang ekivalen ke dalam register buffer. Dengan demikian, output digital akan tetap tersimpan sekalipun akan dimulai siklus konversi yang baru.



Gambar 2.6 Diagram blok ADC

2.5 Mikrokontroler AT89S52

Dengan berkembangnya teknologi mikroprosesor 8 bit dan 16 bit, seiring dengan itu muncul pula kebutuhan agar perangkat elektronika dapat dikemas sekecil mungkin. Seperti Atari, Nintendo, Sega, dan peralatan hiburan serta peralatan rumah tangga seperti AC dan Audio/Video.

Untuk mendukung hal tersebut, tidak dapat dilakukan oleh mikroprosesor standar. Hal ini dikarenakan mikroprosesor membutuhkan komponen eksternal tambahan seperti Memori, pengolah analog ke digital dan perangkat komunikasi

serial misalnya. Oleh karena itu dikembangkanlah chip yang di dalam kemasan tersebut sudah terdapat mikroprosesor, I/O Pendukung, Memori, bahkan ADC yang dikenal dengan istilah mikrokontroler.

Mikrokontroler dapat disebut sebagai "one chip solution" karena terdiri dari;

- **CPU (central processing unit)**

  CPU ialah bagian yang paling penting dari suatu mikroprosesor, ia melakukan pemrosesan data.

- **RAM (Random Access Memory)**

  RAM digunakan Untuk menimpan data sementara.

- **EPROM/PROM/ROM (Erasable Programmable Read Only Memory)**

  ROM digunakan untuk menyimpan program yang bersifat permanent.

- **I/O (input/output) - serial and parallel**

  Unit ini berfungsi agar mikrokontroler dapat berkomunikasi dalam format serial atau paralel, sehingga dapat berkomunikasi dengan mudah dengan PC dan devais standar digital lainnya.

- **Timers**

  Timer berguna untuk mengatur pwaktuan pada system berbasis mikrokontroler, misal untuk delay atau pencacah.

- **Interrupt Controller**

Berfungsi menangani suatu request pada saat mikrokontroler sedang running.

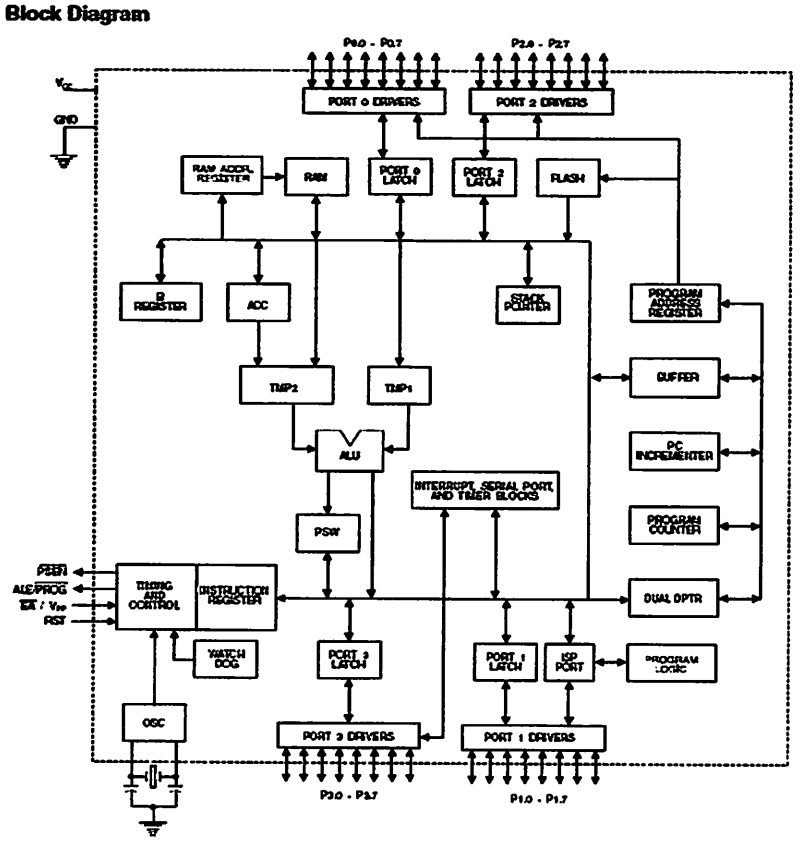
Mikrokontroler AT89S52 merupakan versi terbaru dibandingkan mikrokontroler AT89C51 yang telah banyak digunakan saat ini, Mikrokontroler AT89S52 ialah mikrokomputer CMOS 8 bit dengan 8KB *Flash Programmable dan Erasable Read Only Memory* (PEROM). Mikrokontroler berteknologi memori *non volatile* kerapatan tingi dari Atmel ini kompatibel dengan mikrokontroler standar industri MCS-51 baik pin kaki IC maupun set instruksinya, Spesifikasi penting AT89S52 :

Mikrokontroler AT89S52 memiliki :

- Sebuah CPU (*Central Processing Unit*) 8 Bit.
- 256 byte RAM (*Random Acces Memory*) internal.
- Empat buah port I/O, yang masing masing terdiri dari 8 bit
- Osilator internal dan rangkaian pewaktu.
- Dua buah *timer/counter* 16 bit
- Lima buah jalur interupsi (2 buah interupsi eksternal dan 3 interupsi internal).
- Sebuah port serial dengan *full duplex* UART (*Universal Asynchronous Receiver Transmitter*).
- Mampu melaksanakan proses perkalian, pembagian, dan Boolean.
- EPROM yang besarnya 8 KByte untuk memori program.

- Kecepatan maksimum pelaksanaan instruksi per siklus adalah 0,5 µs pada frekuensi *clock* 24 MHz. Apabila frekuensi *clock* mikrokontroler yang digunakan adalah 12 MHz, maka kecepatan pelaksanaan instruksi adalah 1 µs

Mikrokontroler AT89S52 menggunakan 256 bytes RAM dimana 128 bytes bagian atas menempati alamat parallel ke *special function register* (SFR). Artinya 128 bagian atas mempunyai alamat yang sama dengan SFR namun secara fisik terpisah dari SFR. Ketika instruksi mengakses lokasi internal diatas 7FH, mode alamat yang digunakan pada instruksi menentukan apakah CPU mengakses 128 btyes atas atau SFR. Instruksi yang menggunakan pengalamatan langsung akan mengakses ruang SFR, dijelaskan sesuai pada Gambar 2.7 di bawah ini :



Gambar 2.7 Blok diagram AT89S52
Sumber: Datasheet Atmel AT89S52

2.5.1. Organisasi Memori MC AT89S52

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

2.5.1.1 Memori Program

Pada AT89S52, jika EA dihubungkan pada Vcc, pengambilan program untuk alamat 0000H menuju 1FFFH langsung kepada memori internal dan pengambilan untuk alamat 2000H menuju FFFFH langsung ke eksternal memori.

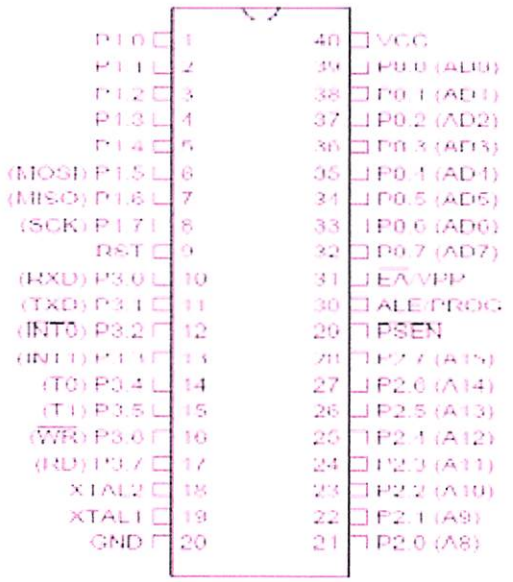
2.5.1.2 Memori Data

AT89S52 terdiri 256 bytes dari RAM internal. Ruang alamat pararel diatas 128 bytes menduduki Fungsi register special (SFR). Ini berarti bahwa yang berada diatas 128 bytes mempunyai alamat yang sama seperti ruang SFR tetapi secara fisik terpisah dari ruang SFR.

Ketika sebuah instruksi mengakses lokasi internal diatas alamat 7FH, mode pengalamatan menggunakan spesifikasi instruksi apakah CPU mengakses RAM yang berada diatas 128 bytes atau ruang SFR. Instruksi yang menggunakan pengalamatan langsung mengakses ruang SFR.

2.5.2. Konfigurasi kaki-kaki Mikrokontroler AT89S52

Berikut ini adalah fisik dan konfigurasi dari kaki – kaki pada MC AT89S52. diperlihatkan pada Gambar 2.8



Gambar 2.8. Gambar fisik MC AT89S52(3)

Fungsi dari masing-masing pin AT89S52 adalah :

1. Pin 1 sampai 8 (Port 1) merupakan port pararel 8 bit dua arah (*bidirectional*) yang dapat digunakan untuk berbagai keperluan (*general purpose*).
2. Pin 9 merupakan pin reset, reset aktif jika mendapat catuan tinggi.
3. Pin 10 sampai 17 (Port 3) adalah port pararel 8 bit dua arah yang memiliki fungsi pengganti sebagai berikut :
    - P3.0 (10) : RXD (port serial penerima data)
    - P3.1 (11) : TXD (port serial pengirim data)
    - P3.2 (12) : INT0 (input interupsi eksternal 0, aktif *low*)

- P3.3 (13) : INT1 (input interupsi ekstrernal 1, aktif *low*)
- P3.4 (14) : T0 (eksternal input *timer / counter* 0)
- P3.5 (15) : T1 (eksternal input *timer / counter* 1)
- P3.6 (16) : WR (*Write*, aktif *low*) Sinyal kontrol penulisan data dari port 0 ke memori data dan input-output eksternal.
- P3.7 (17) : RD (*Read*, aktif *low*) Sinyal kontrol pembacaan memori data input-output eksternal ke port 0.

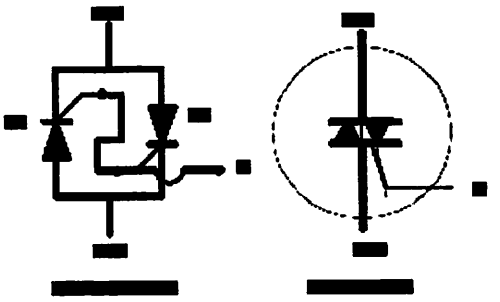
4. Pin 18 sebagai XTAL 2, keluaran osilator yang terhubung pada kristal.
5. Pin 19 sebagai XTAL 1, masukan ke osilator berpenguatan tinggi, terhubung pada kristal.
6. Pin 20 sebagai Vss, terhubung ke 0 atau ground pada rangkaian.
7. Pin 21 sampai 28 (Port 2) adalah port pararel 8 bit dua arah. Port ini mengirim byte alamat bila pengaksesan dilakukan pada memori eksternal.
8. Pin 29 sebagai PSEN (*Program Store Enable*) adalah sinyal yang digunakan untuk membaca, memindahkan program memori eksternal (ROM / EPROM) ke mikrokontroler (aktif *low*).
9. Pin 30 sebagai ALE (*Address Latch Enable*) untuk menahan alamat bawah selama mengakses memori eksternal. Pin ini juga

berfungsi sebagai PROG (aktif *low*) yang diaktifkan saat memprogram internal *flash* memori pada mikrokontroler (*on chip*).

10. Pin 31 sebagai EA (*External Accesss*) untuk memilih memori yang akan digunakan, memori program internal (EA = Vcc) atau memori program eksternal (EA = Vss), juga berfungsi sebagai Vpp (*programming supply voltage*) pada saat memprogram internal *flash* memori pada mikrokontroler.
11. Pin 32 sampai 39 (Port 0) merupakan port pararel 8 bit dua arah. Berfungsi sebagai alamat bawah yang dimultipleks dengan data untuk mengakses program dan data memori eksternal.
12. Pin 40 sebagai Vcc, terhubung ke +5 V sebagai catuan untuk mikrokontroler.

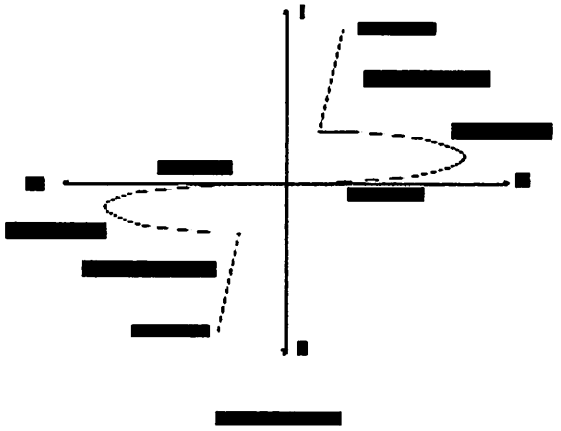
2.6. Triac

Triac dapat bersifat konduktif dalam dua arah dan biasa digunakan untuk pengendali fasa ac. Hal ini dapat dianggap sebagai dua buah SCR yang tersambung antipararel dengan koneksi gerbang seperti gambar 2.9

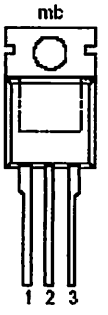


Gambar 2.9 simbol TRIAC dan ekivalensi

Sedangkan karekteristik TRIAC diperlihatkan pada gambar 2.10



Gambar 2.10 Karakteristik TRIAC



Gambar 2.11 TRIAC

Kerena TRIAC merupakan komponen bidirectional, terminalnya tidak dapat ditentukan sebagai anoda/katode. Jika terminal MT2 positif terhadap MT1. TRIAC dapat dimatikan dengan memberikan sinyal gerbang positif antara gerbang G dan MT1. Jika terminal MT2 negatif terhadap MT1, maka TRIAC dapat dihidupkan dengan memberikan sinyal pulsa negatif antara gerbang dan terminal MT1. tidak perlu untuk memiliki kedua sinyal gerbang positif dan negatif dan TRIAC dapat dihidupkan baik oleh sinyal gerbang positif maupun negatif.

Dalam prakteknya sensitivitas berfariasi antara satu kuadran dengan kuadran lain, dan TRIAC biasanya beroperasi dikuadran I atau kuadran III.

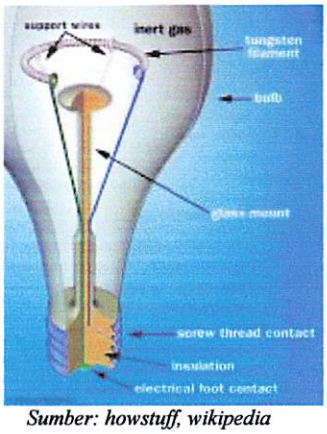
2.7 Lampu Pijar

Bola lampu, bohlam atau lampu pijar adalah sumber cahaya buatan yang dihasilkan melalui penyaluran arus listrik melalui filamen yang kemudian memanas dan menghasilkan foton. Kaca yang menyelubungi filament panas tersebut menghalangi oksigen di udara dari berhubungan dengannya sehingga filament tidak akan langsung rusak akibat teroksidasi. Salah satu kelebihan bola lampu adalah dapat dihasilkannya bola lampu dalam berbagai besar voltase, dari puluhan hingga ratusan volt, namun karena jumlah listrik yang diperlukan bola lampu untuk menghasilkan cahaya yang terang lebih besar dibandingkan dengan sumber cahaya buatan lainnya, maka secara bertahap bola lampu mulai digantikan lampu neon, LED, dan lain-lain.

Gambar 2.12 Bola lampu(4)

Bola lampu diperkenalkan pertama kalinya kepada umum oleh Thomas Alva Edison pada 31 Desember 1879.

2.7.1 Cara Kerja Lampu Pijar



Sumber: howstuff, wikipedia

Saat bola lampu pijar di hidupkan, arus listrik akan mengalir dari Electrical contact menuju filamen dengan melewati kawat penghubung. Akibatnya akan terjadi pergerakan elektron bebas dari kutub negatif ke kutup positif Elektron di sepanjang filamen ini secara konstan akan menabrak atom pada filamen. Energinya akan mengetarkan atom atau arus listrik memanaskan atom. Ikatan elektron dalam atom-atom yang bergetar ini akan mendorong atom pada tingkatan tertinggi secara berkala. Saat energinya kembali ketingkat normal, elektron akan melepaskan energi ekstra dalam bentuk poton. Atom-atom yang dilepaskan ini dalam bentuk poton-poton sinar infrared yang tidak mungkin dilihat oleh mata manusia. Tetapi bila dipanaskan sampai temperatur 2.200 derajat Celcius, cahaya yang dipancarkan dapat kita lihat seperti halnya bola lampu pijar yang sering kita pakai sehari-hari.

2.8 LCD *(Liquid Crystal Display)*

Bila diartikan dalam bahasa Indonesia, *liquid crystal* berarti kristal cair, padat dan cair merupakan dua sifat benda yang berbeda. Molekul-molekul benda padat tersebar secara teratur dan posisinya tidak berubah-ubah, sedangkan molekul-molekul zat cair letak dan posisinya tidak teratur karena dapat bergerak acak ke segala arah. Pada tahun 1888 seorang ahli botani, *Friedrich Reinitzer*, menemukan fase yang berada di tengah-tengah antara fase padat dan cair. Fase ini memiliki sifat-sifat padat dan cair secara bersama-sama. Molekul- molekulnya memiliki arah yang sama seperti sifat padat, tetapi molekul-molekul itu dapat bergerak bebas seperti pada cairan. Fase kristal cair ini berada lebih dekat dengan fase cair karena dengan sedikit penambahan temperatur (pemanasan), fasenya langsung berubah menjadi cair. Sifat ini menunjukkan sensitivitas yang tinggi terhadap temperatur. Sifat inilah yang menjadi dasar utama pemanfaatan kristal cair dalam teknologi.

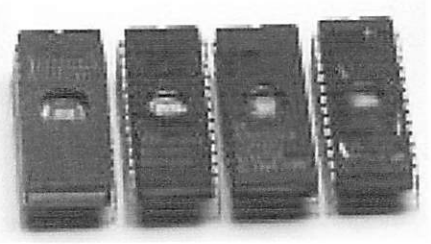
Gambar 2.13 LCD *(Liquid Crystal Display)(5)*

Selain temperatur, kristal cair juga sangat sensitif terhadap arus listrik (beda potensial). Prinsip semacam inilah yang digunakan dalam teknologi LCD. Ini sebabnya layar laptop terkadang terlihat berbeda di musim dingin atau saat digunakan di cuaca sangat panas, Jenis kristal cair yang digunakan dalam pengembangan teknologi LCD adalah tipe nematic (molekulnya memiliki pola tertentu dengan arah tertentu). Tipe yang paling sederhana adalah *twisted nematic* (TN) yang memiliki struktur molekul yang terpilin secara alamiah (dikembangkan

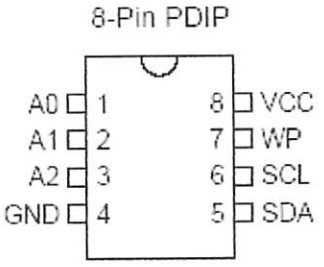
pada tahun 1967). Struktur TN terpilin secara alamiah 90. Struktur TN ini dapat dilepas pilinannya (untwist) dengan menggunakan arus listrik.

2.9 Eeprom

EEPROM yang digunakan adalah IC AT24C04. IC ini memiliki kapasitas 8K*Byte*. Susunan kaki keluarga AT24Cxx adalah sama, perbedaanya hanya pada kapasitasnya.



Gambar 2.14 Fisik Eeprom



Gambar 2.15 Arsitektur AT24C04

Deskripsi Pin :

- *Serial Clock* (SCL) :

Input SCL digunakan sebagai ujung *clock* data positif ke dalam masing-masing EEPROM dan untuk ujung *clock* data *negative* keluar EEPROM.

- *Serial Data* (SDA) :

  Pin SDA digunakan untuk transfer data 2 arah.

- *Device/Page Adresses* (A2, A1, A0) :

  Pin A2, A1, A0 adalah pin sebagai alamat input yang dikabelkan atau bisa juga tidak dihubungkan dengan perangkat yang kompatibel dengan AT24C64.

- *Write Protect* (WP) :

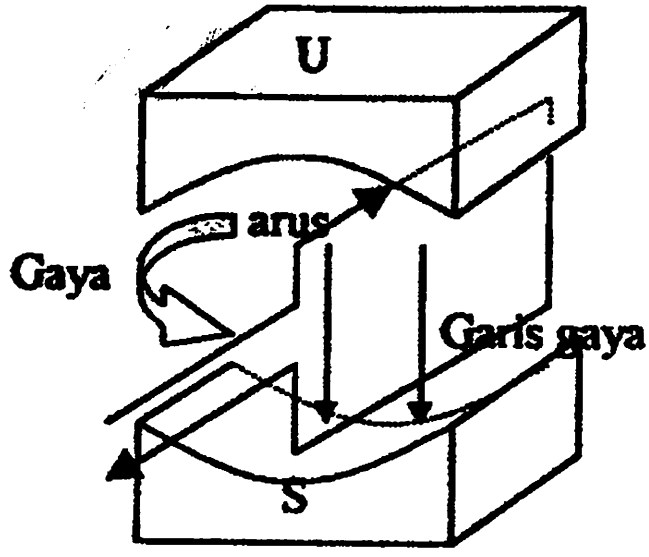
  Pin WP merupakan pin yang dipakai untuk melindungi isi yang disimpan di dalam IC *serial* EEPROM. Ketika input WP dihubungkan ke GND, maka akan menjalankan operasi normal. Ketika WP dihubungkan dengan Vcc, semua operasi penulisan ke kuadran yang lebih atas dari memori tidak dapat dikerjakan (terprotek). Jika pin ini tidak dihubungkan maka secara internal WP dihubungkan ke GND.

EEPROM ini membutuhkan 8 bit alamat data mengikuti kondisi *start* untuk mengaktifkan chip untuk operasi pembacaan atau penulisan.

2.10 Motor DC

Motor arus searah suatu mesin listrik yang berfungsi merubah tenaga listrik arus searah menjadi tenaga mekanik, perubahan tenaga listrik menjadi tenaga mekanik berdasarkan pada suatu kaidah yang menerangkan bila suatu

penghantar yang membawa arus listrik berada pada sebuah medan magnet, maka penghantar tersebut akan mengalami suatu gaya seperti terlihat pada gambar 2.16



Gambar 2.16 Gaya pada Motor DC

Gaya tersebut menimbulkan torsi yang akan menghasilkan gerak pada motor, gerak tersebut dirubah menjadi suatu gerak putaran.

BAB III

PERENCANAAN DAN PEMBUATAN ALAT

3.1. Umum

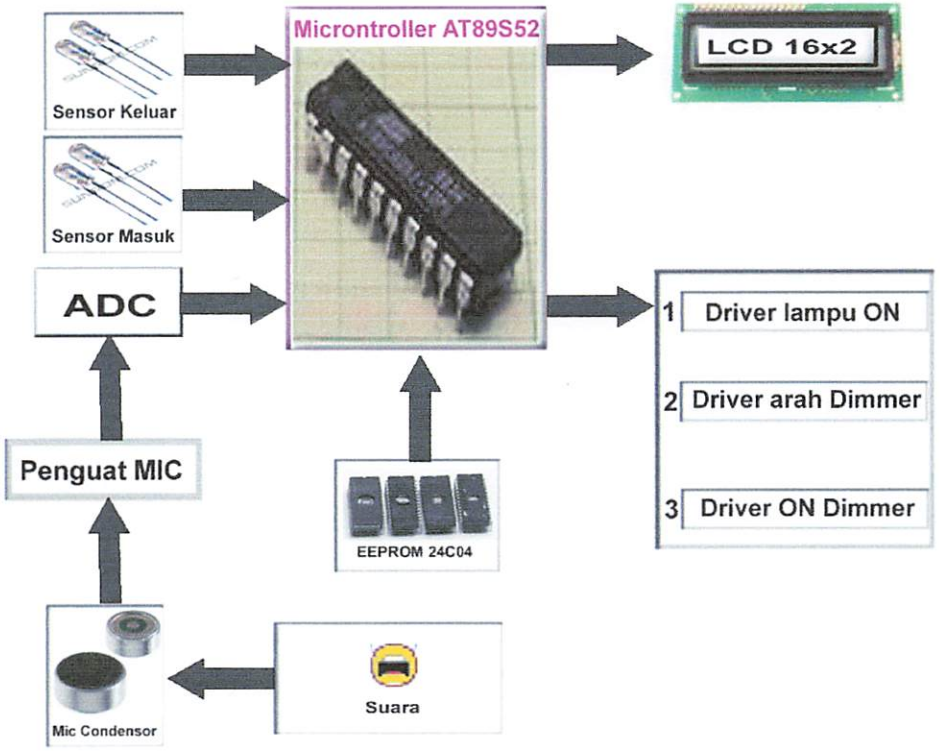
Pada bab ini akan membahas tentang Perancangan Dan Pembuatan Alat Pengontrol Dimmer Lampu Dalam Ruang Berbasis Mikrokontroller AT89S52 Perencanaan tersebut meliputi

- Perancangan blok rangkaian
- Perancangan perangkat keras (Hardware)
- Perancangan perangkat lunak (Software)

Pada perancangan perangkat keras akan meliputi periphera-peripheral yang digunakan pada sistem ini. Sedangkan pada perancangan perangkat lunak akan meliputi diagram alir dan software secara umum. Akan tetapi kedua perangkat ini dalam kerjanya saling menunjang satu sama lain. dan menggunakan mikrokontroller AT89S52 sebagai pengendali.

3.2. Blok Diagram Sistem

Blok diagram rangkaian terintegrasi dari rancangan alat ini dapat dilihat pada Gambar 3.1 di bawah ini.



Gambar 3.1. *Diagram Blok rangkaian terintegrasi*

3.2.1. Prinsip kerja dari diagram blok:

- **Sensor**

Sensor yang digunakan menggunakan 2 komponen

1. **Infrared Photodiode**

   Berfungsi mendekteksi sekaligus menghidupkan, dan mematikan lampu bila sensor tersebut dilewati

2. **Condenser Microphones**

   Berfungsi untuk mendeteksi suara (*tepuk*) yang mana fungsi dari mic condenser ini untuk memberi sinyal pada motor untuk meredupkan dan menerangkan lampu sesuai kemauan pengguna

- **Penguat MIC**

  Rangkaian ini berfungsi untuk menguatkan inputan awal (sensor suara), pada komponen IC ini memiliki 2 input tegangan dan 1 output tegangan, dimana tegangan output-nya adalah proporsional terhadap perbedaan tegangan antara kedua inputnya itu.

- **ADC (analog-to digital converter)**

  Merubah sinyal analog yang dihasilkan penguat instrument menjadi sinyal digital.

- **Mikrokontroler AT89S52**

  Berfungsi mengolah data yang dihasilkan, dan dirancang berdiri sendiri karena sudah terdapat EPROM, RAM, serta port I/O. Dalam rangkaian ini, mikrokontroller AT89S52 berfungsi mengontrol seluruh proses, mulai menjalankan perintah dari sensor, penguat instrumentasi dan ADC, kemudian menjalankan program tersebut melalui driver dan out put *(lampu)*

- **LCD (display)**

  LCD adalah suatu peralatan yang digunakan untuk menampilkan (display) data yang dihasilkan mikrokontroler.

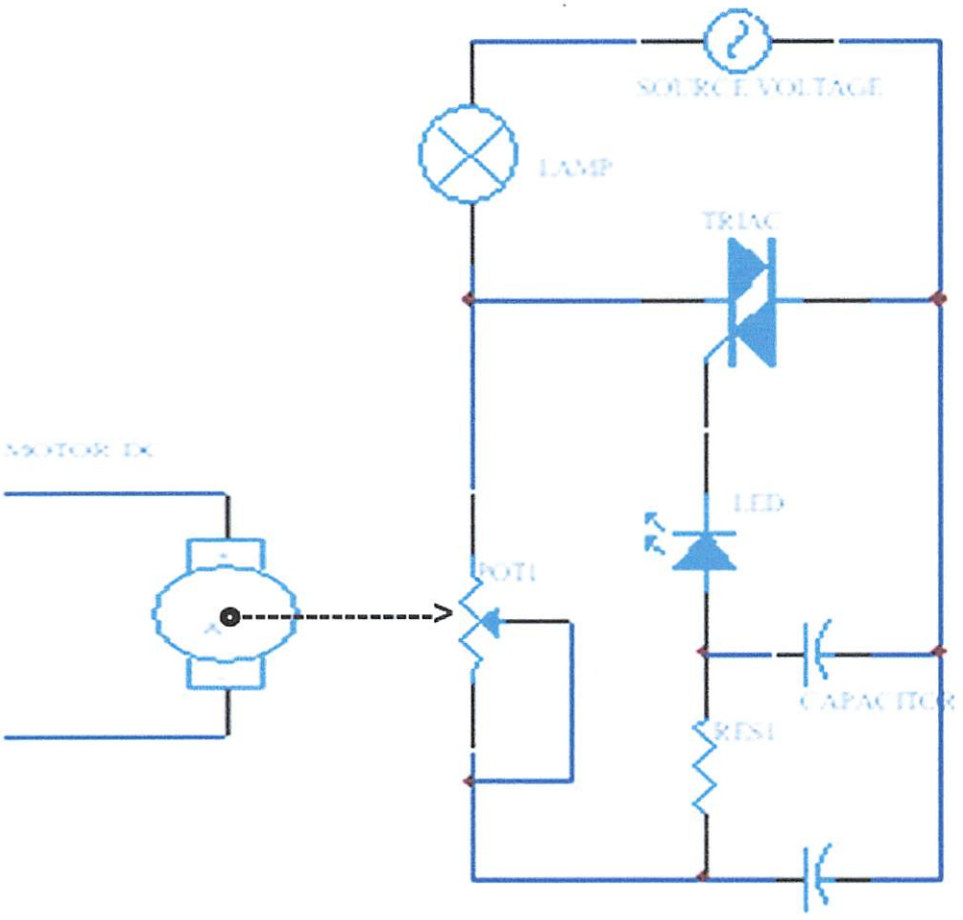
- **EEPROM 24C04**

  EEPROM disini merupakan memori *eksternal* yang akan digunakan untuk menyimpan dan meng-counter berapa banyak orang yang masuk ataupun keluar dari ruangan

3.3.Perencanaan Hardware

3.3.1.Rangkaian *Dimmer*

Potensio yang terdapat pada rangkaian dimmer dicontrol oleh motor DC seperti terlihat pada gambar 3.2



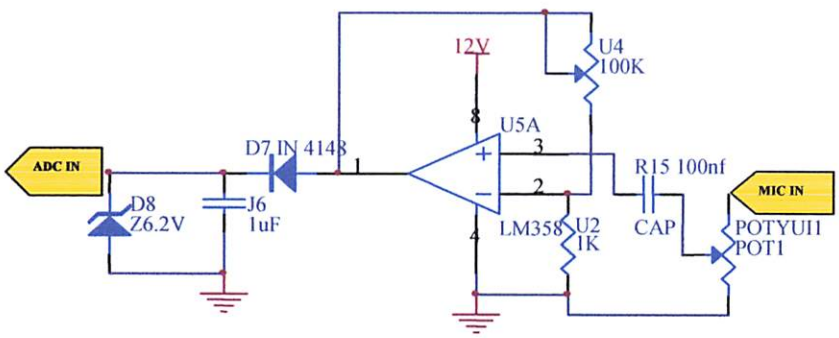
Gambar 3.2 Rangkaian Dimmer yang dikontrol Motor DC

Dimmer digunakan untuk pencahayaan umum hingga ke unit daya tinggi, contoh digunakan dalam theater besar atau arsitektur pencahayaan instalasi dalam ruangan. dimmers berukuran kecil biasanya dapat langsung dikontrol oleh pengguna, Dimmers modern biasanya dikontrol oleh sistem kontrol digital seperti

DMX atau ethernet. di industri, perubahan cahaya dalam intensitas disebut "fades" (memudar) Dimmers dengan kontrol manual memiliki batas pada kecepatan, program dimmer dibuat dari silicon-dikontrol rectifiers (SCR) sebagai pengganti variabel resistors potentiometers karena memiliki efisiensi tinggi.

3.3.2. Rangkaian Penguat Microphones Condenser

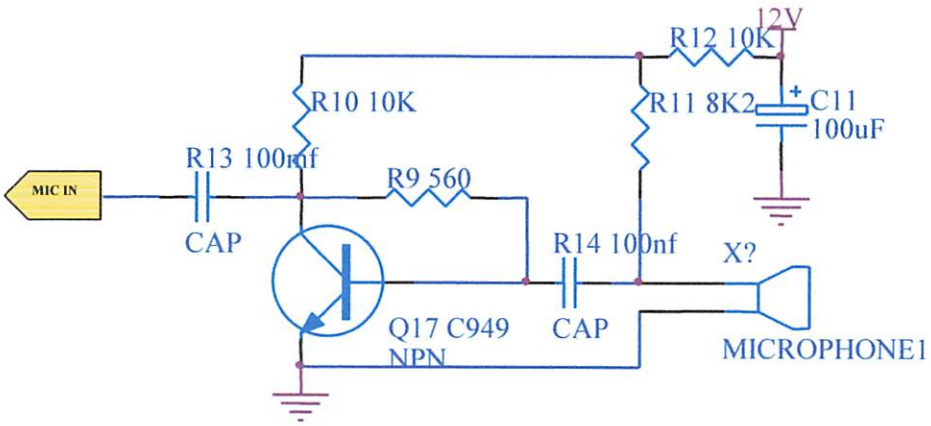
Prinsip kerja dari rangkaian ini adalah menguatkan suara (tepuk) yang dihasil kan, *Operational Amplifier* atau di singkat op-amp merupakan salah satu komponen analog yang popular digunakan dalam berbagai aplikasi rangkaian elektronika. Aplikasi op-amp yang digunakan adalah op-amp integrator Pada pokok bahasan kali ini akan dipaparkan gambar rangkaian dibawah ini



Gambar 3.3 *Rangkaian penguat microphones condenser*

3.3.3. Rangkaian Sensor Suara

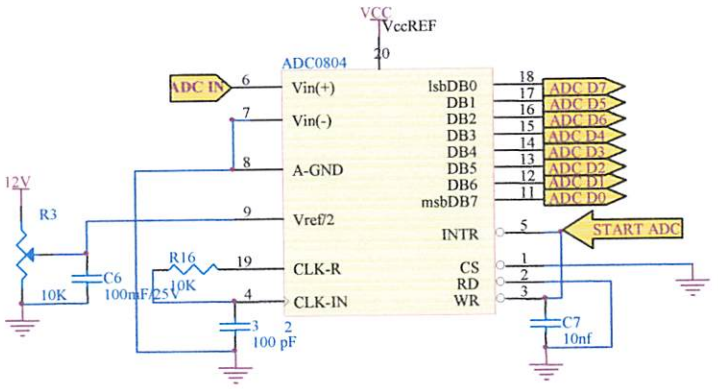
Sensor suara yang dihasilkan pengguna akan ditangkap oleh *mic condenser*, dari *mic condenser* gelombang suara dirubah menjadi gelombang *AC*, yang kemudian dikuatkan dengan *transistor* yang kemudian dikuatkan dan disearahkan lagi dengan *op-am* baru kemudian akan dirubah menjadi sinyal digital oleh *ADC*



Gambar 3.4 Rangkaian Sensor Suara

3.3.4. Rangkaian ADC (analog-to digital converter)

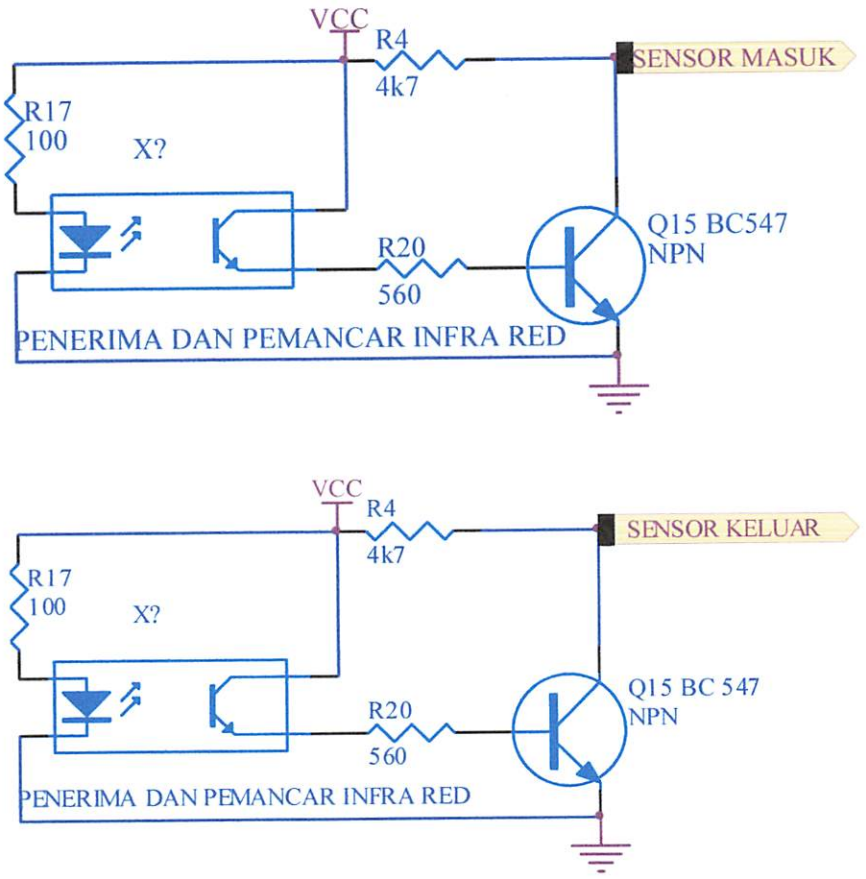
Prinsip kerja dari rangkaian ini adalah meng-convert sinyal analog yang di hasilkan oleh *penguat mic* menjadi sinyal digital, Comparator membandingkan antara tegangan masukkan analog dengan tegangan D/A Converter, apabila tegangan masukkan yang akan dikonversi belum sama dengan tegangan keluaran dari D/A Converter maka keluaran comparator = "1" (A>B) sehingga clock dapat memberikan masukkan counter dan hitungan counter naik. Bila diperoleh masukan A = B maka output comparator 0, dan clock berhenti, dan inilah nilai digit-digit yang dibaca oleh ADC.



Gambar 3.5 Rangkaian ADC (Analog-to Digital Converter)

3.3.5. Rangkaian *sensor masuk dan sensor keluar (infrared photodiode)*

Fungsi dari rangkaian ini dirancang untuk mendeteksi orang yang masuk atau yang keluar dari ruangan

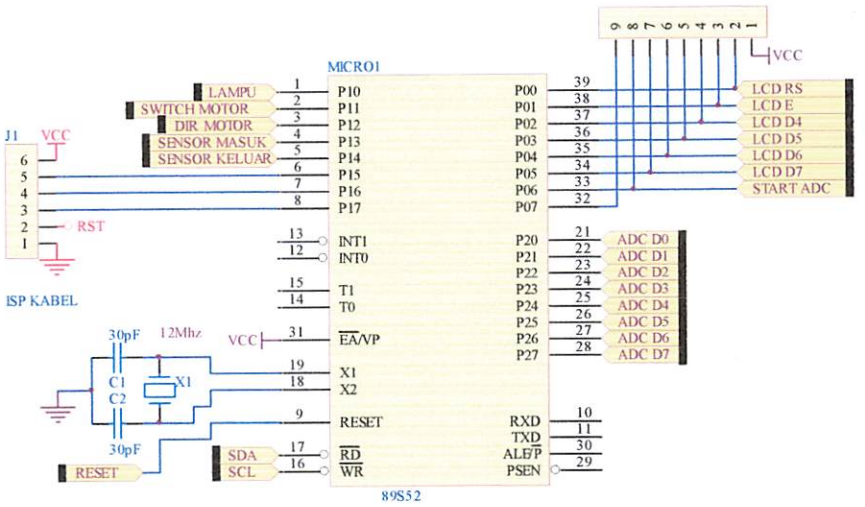


Gambar 3.6 *Rangkaian infrared photodiode*

Dari gambar rangkaian diatas maka dapat diketahui jumlah orang yang masuk atau orang yang keluar, sehingga sinyal yang ditangkap oleh sensor dikirimkan ke *microkontroler AT89S52* yang kemudian data tersebut disimpan pada *eeprom*, yang nantinya data tersebut akan digunakan untuk memerintahkan kapan lampu dapat menyala atau kapan lampu dapat mati

3.3.6. Rangkaian *Mikrocontroller AT89S52*

Mikrokontroler AT89S52 harus didukung oleh beberapa rangkaian lain agar dapat melakukan prosesnya, yaitu berupa rangkaian *clock* dan *reset*. Selain itu juga harus ditentukan penggunaan port-portnya dan sinyal sinyal yang digunakan untuk mendukung proses yang akan dilakukan. Mikrokontroler AT89S52 harus didukung oleh beberapa rangkaian lain agar dapat melakukan prosesnya, yaitu berupa rangkaian *clock* dan *reset*. Selain itu juga harus ditentukan penggunaan port-portnya dan sinyal sinyal yang digunakan untuk mendukung proses yang akan dilakukan.

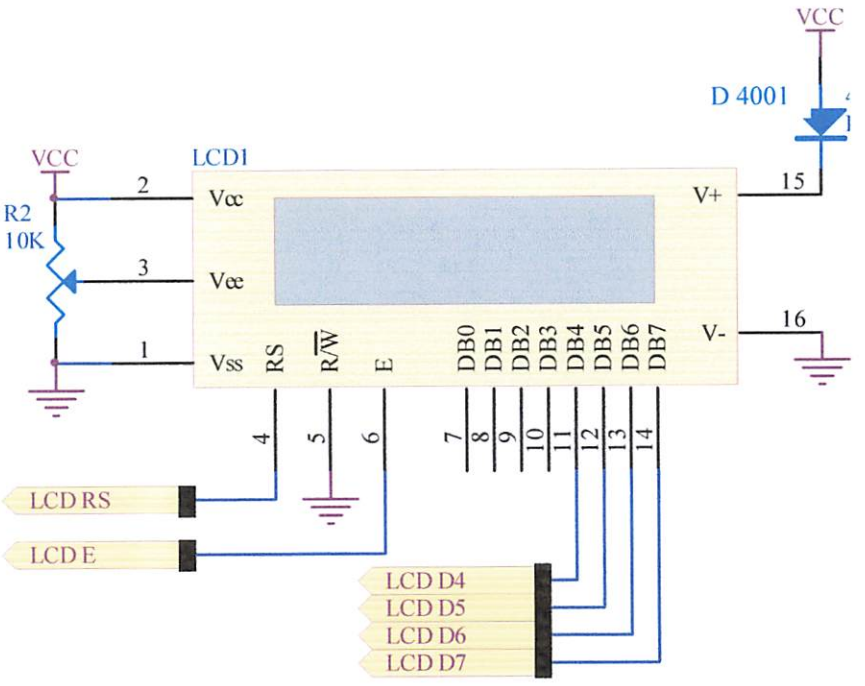


Gambar 3.7 *Rangkaian Microkontroller AT89S52*

Mikrokontroller tipe Atmel AT89S52 termasuk kedalam keluarga MCS51 merupakan suatu mikrokomputer CMOS 8-bit dengan daya rendah, kemampuan tinggi, memiliki 8K byte Flash Programable and Erasable Read Only Memory (PEROM). Perangkat ini dibuat menggunakan teknologi memori nonvolatile (tidak kehilangan data bila kehilangan daya listrik). Set instruksi dan kaki keluaran AT89S52 sesuai dengan standar industri 80C51 dan 80C52. Atmel AT89S52 adalah mikrokomputer yang sangat bagus dan fleksibel dengan harga yang rendah untuk banyak aplikasi sistem kendali.

3.3.7. Rangkaian *liqud crystal display* (LCD)

Pada sistem yang direncanakan akan digunakan LCD (*Liquid Crystal Display*) sebagai tampilan. LCD yang digunakan adalah jenis TM162ABC yang merupakan LCD dua baris dengan tiap barisnya terdiri dari 16 karakter.LCD ini membutuhkan 3 sinyal kontrol, R/W (*read/write*) untuk menentukan apakah data akan dibaca atau ditulis, E (*Enable*) yang merupakan sinyal untuk meng-enable-kan dan RS (*Register Select*) untuk memilih register yang diakses. LCD TM162ABC memiliki 2 register yaitu register data dan register instruksi.



Gambar 3.8 *rangkaian LCD*

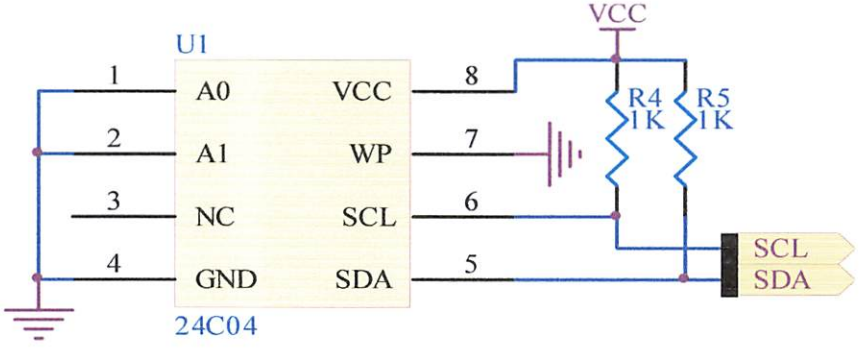
LCD atau *Liquid Crystal Display* digunakan sebagai alat komunikasi antara alat dengan pemakai. Dengan adanya LCD semua aktifitas dan respon dari peralatan dapat dilihat dan kemudian pemakai dapat menentukan langkah selanjutnya terhadap peralatan. Perintah dasar yang diberikan pada modul LCD antara lain adalah membersihkan layar, mengendalikan kursor,dan pengendalian posisi karakter

3.3.8. Rangkaian *EEPROM*

Memori yang digunakan EEPROM AT 24C04. Dengan menggunakan memori ini data dapat disimpan tanpa adanya baterai *back up* sebab memori ini merupakan jenis *nonvolatile* atau datanya tidak akan hilang walaupun catu dimatikan. Digunakannya komponen ini karena frekuensi penulisannya yang tidak terlalu sering sehingga tidak perlu menggunakan memori jenis *volatile*.

Perancangan kedua adalah alokasi penyimpanan data dalam memori. Data yang dimaksud adalah data jumlah orang yang masuk dan data jumlah orang keluar. Dan untuk menyimpan dalam memori perlu diberikan batasan – batasan agar dalam operasional perangkat lunak dapat dikontrol. Adapun batasan tersebut adalah :

1. Panjang maksimal data yang diperbolehkan adalah 15 angka.
2. Panjang maksimal nama yang diperbolehkan adalah 15 karakter.
3. *Bit* pertama digunakan untuk mengenali lokasi tersebut dipakai atau tidak.

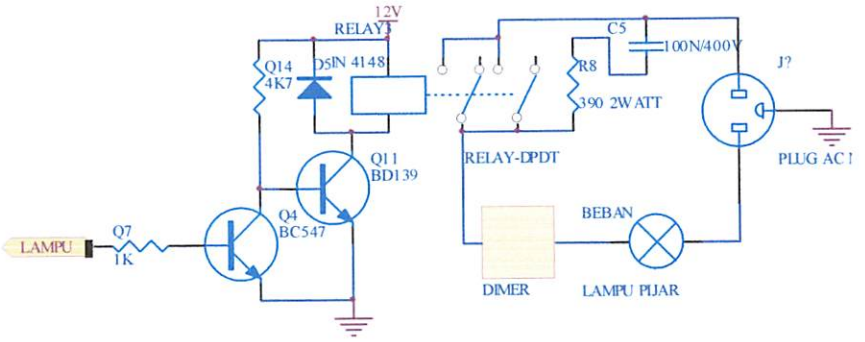


Gambar 3.9 *rangkaian EEPROM 24C04*

Dari batasan tersebut diperoleh panjang memori yang dibutuhkan dalam satu *record* adalah 25 *byte*, sehingga untuk memenuhi penyimpanan 80 record dibutuhkan panjang memori sebesar 25 x 80 *byte* atau sebesar 2000 *byte*. Dari hasil perhitungan tersebut maka memori masih tersisa 48 *byte*.

3.3.9. Rangkaian *driver lampu on*

Pada rangkaian ini lampu dikendalikan oleh *sensor masuk* dan *sensor keluar* dan *rangkaian mic*,



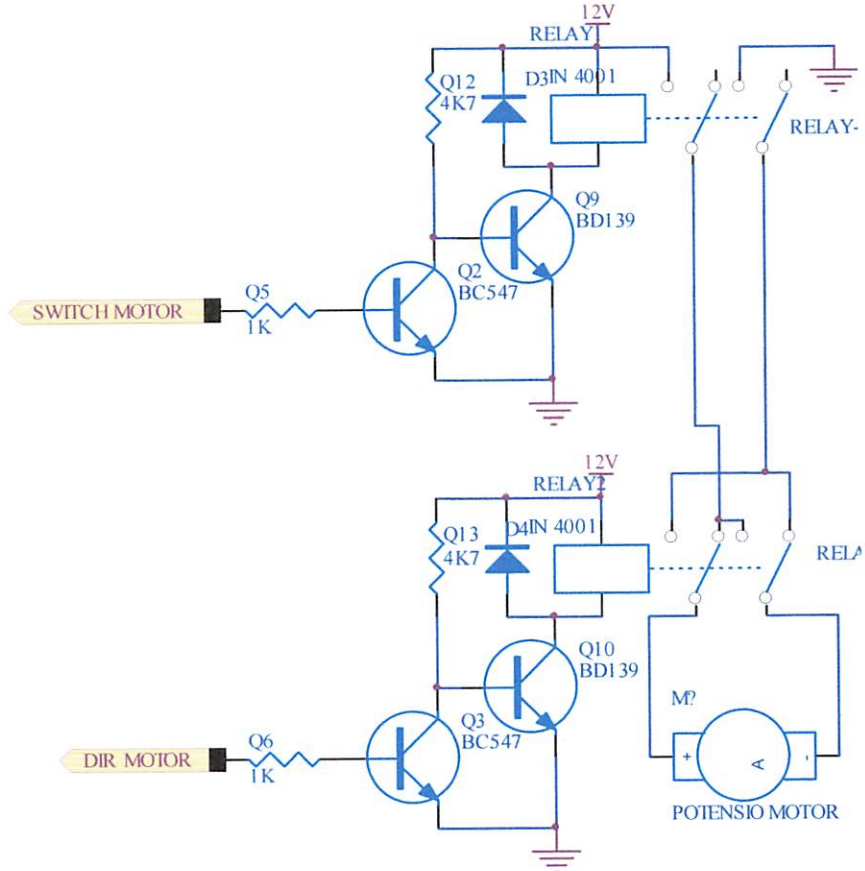
Gambar 3.10 *rangkaian lampu on*

Untuk menyalakan lampu, rangkaian yang digunakan adalah *rangkaian lampu on*

3.3.10. Rangkaian *Driver Arah Dimmer dan driver on dimmer*

Pada rangkaian ini berfungsi untuk dimmer on dan arah dimmer

Seperti tertera pada gambar 3.10



Gambar 3.11 *Rangkaian Driver Arah Dimmer dan driver on dimmer*

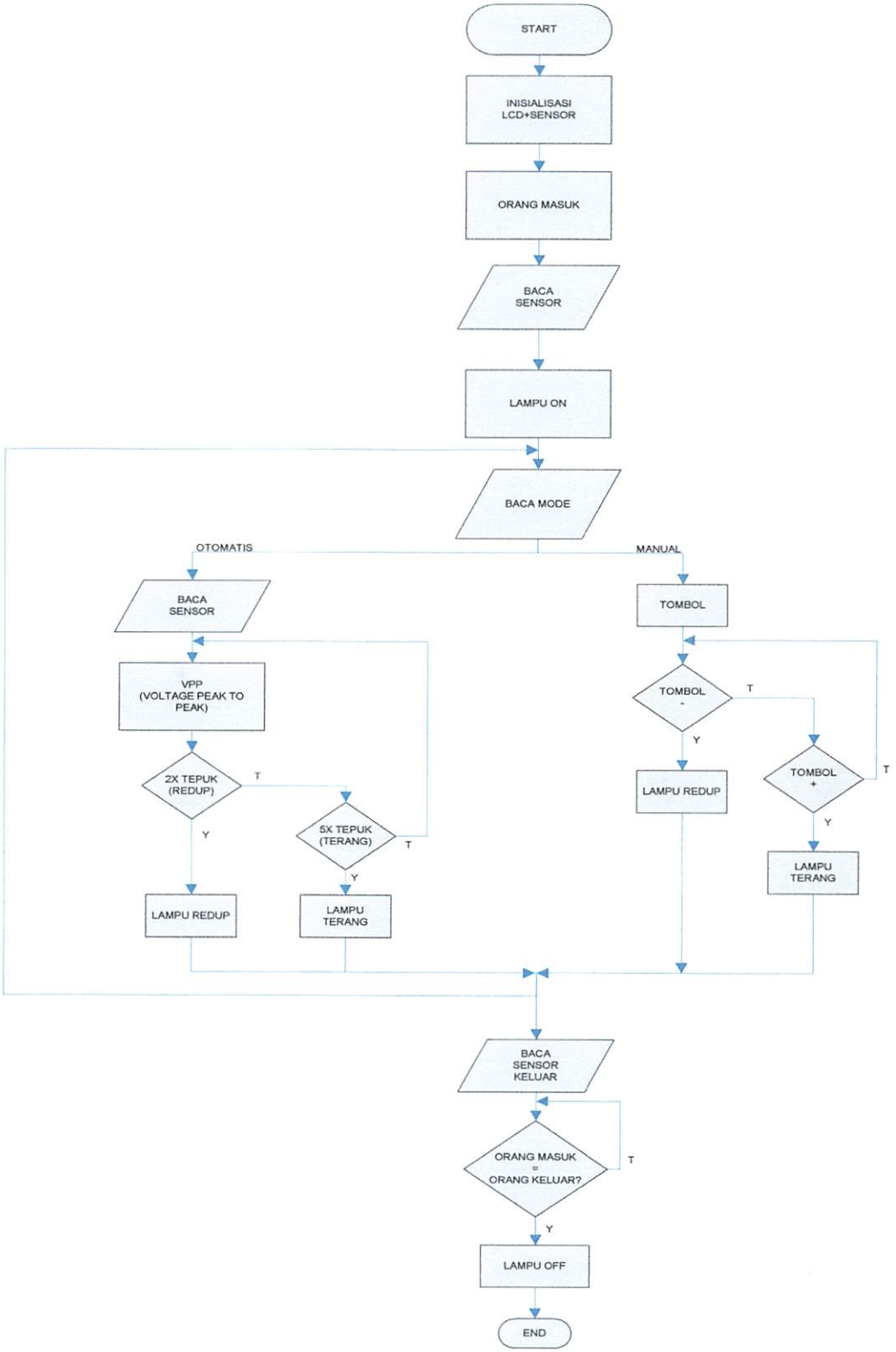
Agar lampu bekerja redup dan terang pada rangkaian diatas menggunakan switch dan dir putar potensio

3.4 Perancangan Perangkat Lunak

Perancangan perangkat lunak (*software*) didasarkan pada perancangan perangkat keras yang telah dibuat sebelumnya, guna mendapatkan system kerja yang diharapkan. Berikut diagram alur dari perencangan perangkat lunak, dalam perancangan alat ini diagram alur terbagi menjadi 3 bagian ;

- Sensor Deteksi Orang Masuk *(infrared photodiode)*
- Sensor Deteksi Suara (tepuk) *(mic condenser)*
- Sensor Deteksi Orang Keluar *(infrared photodiode)*

3.4.1 Diagram Alir Rangkaian Keseluruhan



BAB IV

ANALISA DAN PENGUJIAN ALAT

4.1 Umum

Untuk memastikan apa alat ini berjalan sesuai dengan prosedur, oleh sebab itu di perlukan ada nya pengujian

Pada bab ini membahas cara pengujian dan analisa dari alat yang dirancang, sehingga dapat diketahui apakah alat tersebut dapat bekerja sesuai dengan yang telah direncanakan. Dalam rangka pengujian alat tersebut diuraikan percobaan yang dilakukan untuk mengetahui respon dari keseluruhan alat yang telah dirancang.

Untuk mengetahui kemampuan alat dan sistem kerja sesuai dengan program yang telah dibuat maka dilakukan pengujian pada alat dan sistem kerja alat .

4.2. Pengujian Rangkaian *sensor masuk dan sensor keluar (infrared photodiode)*

4.2.1. Tujuan

Tujuan pengujian rangkaian *sensor masuk dan sensor keluar (infrared photodiode)* adalah untuk mengetahui apakah Photodiode infrared memiliki keadaan yang baik untuk melakukan *switching* dengan kondisi logika 1 dan 0 dalam melakukan pendekteksian apakah ada orang masuk atau orang keluar.

4.2.2. Peralatan yang Digunakan

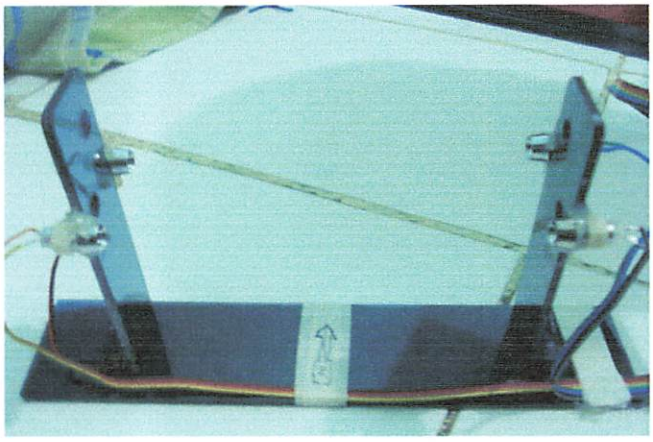
1. 1 Buah multi meter winner M-890C
2. Rangkaian sensor masuk infrared

3. Led infra red Photodiode
4. Lampu Bohlam

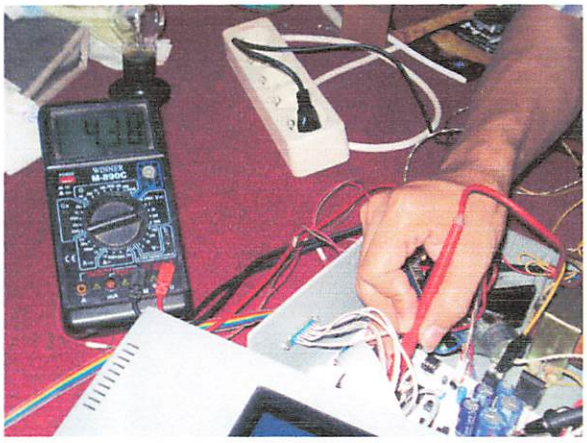
4.2.3. Langkah-Langkah Pengujian

1. Hubungkan kabel multimeter warna hitam pada ground dan kabel merah pada port tombol pada kaki mikrocontroler
2. Halangi sinar infrared (masuk), Halangi sinar infrared (keluar)
3. Mengukur tegangan pada saat sensor dihalangi dan tidak dihalangi

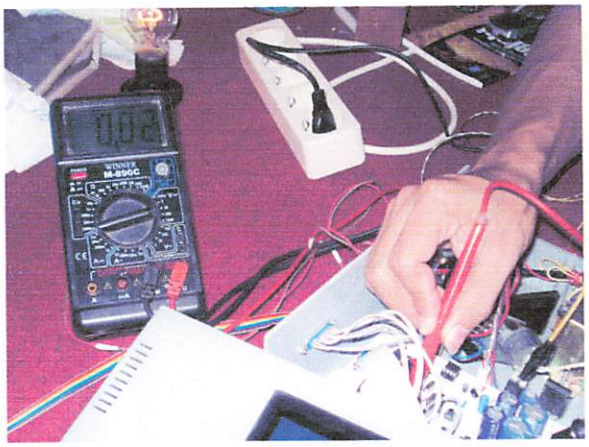
4.2.4. Gambar pengujian alat



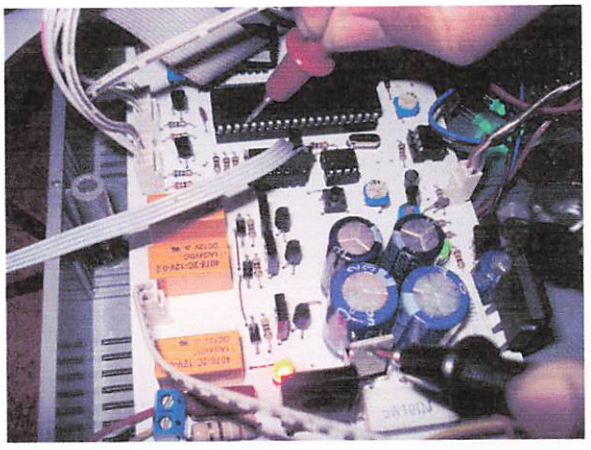
Gambar 4.1 Infra red



Gambar 4.2 Pengujian rangkaian *sensor yang dihalangi*



Gambar 4.3 pengujian rangkaian sensor yang tidak dihalangi



Gambar 4.4 cara mengukur tegangan pada rangkaian sensor

Tegangan yang didapat pada hasil pengukuran sensor yang di halangi sebesar 4,38 volt, dan nilai tegangan pada sensor yang tidak dihalangi sebesar 0,02 volt

Kesimpulan

- Pada saat sensor tidak dihalangi maka port mikrokontroler terhubung ground sehingga sensor berlogika "low" sehingga mendapat tegangan 0,02 volt

- Pada saat sensor dihalangi, sensor mendapat logika "high" dari pull-up, sehingga mendapat tegangan 4,38 volt

4.3. Pengujian Rangkaian ADC (analog-to digital converter)

4.3.1. Tujuan

Tujuan Pengukuran ini adalah untuk mengetahui kondisi tanpa suara dan kondisi pada saat ada suara

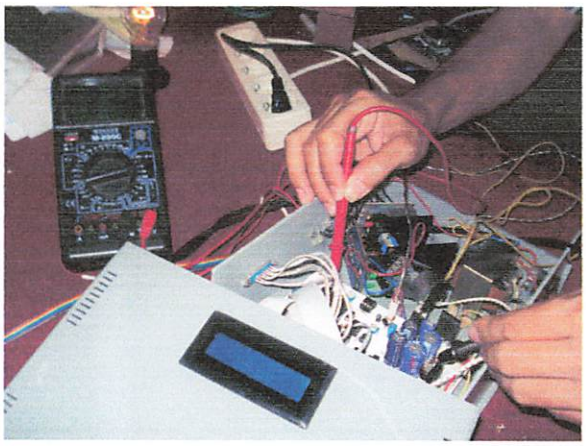
4.3.2 Peralatan yang digunakan

1. 1 buah multimeter winner M-890C
2. Rangkaian ADC
3. Lampu Bohlam

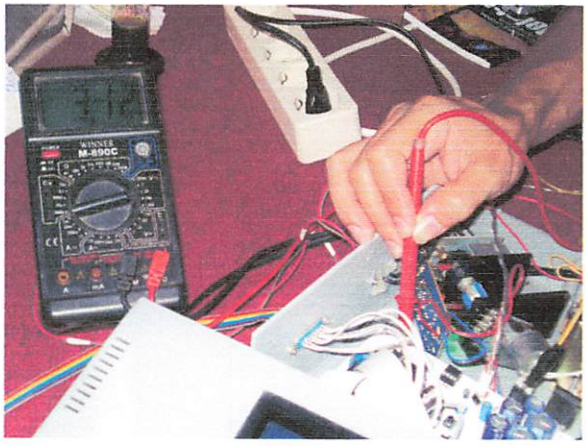
4.3.3 Langkah-Langkah Pengujian

1. Hubungkan kabel multimeter warna hitam pada ground dan kabel warna merah pada rangkaian output penguat (output penguat = input ADC)
2. Beri inputan suara (tepuk 2X redup, 5X untuk terang) pada mic condenser
3. Mengukur tegangan pada input ADC
4. Tidak beri inputan suara pada mic condenser
5. Mengukur tegangan pada Input ADC

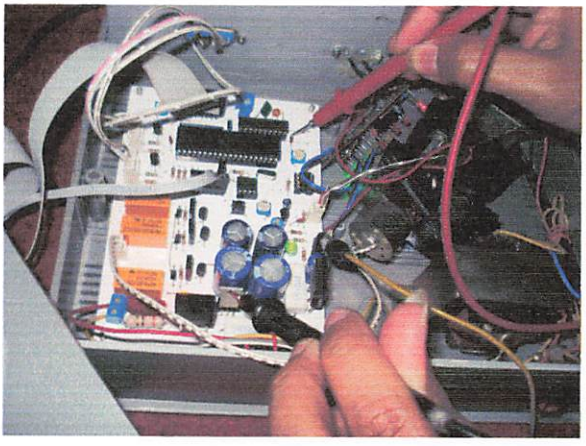
4.3.4. Gambar pengujian alat



Gambar 4.5 ADC pada kondisi tanpa suara



Gambar 4.6 ADC pada kondisi ada suara



Gambar 4.7 cara mengukur tegangan pada ADC

Tegangan yang dihasilkan pada pengukuran ini adalah sebagai berikut:

1. Untuk tegangan pada saat ADC dalam kondisi tanpa suara sebesar 2,32 volt
2. Sedangkan tegangan pada saat ADC dalam kondisi diberi input suara (tepuk) sebesar 3,12 volt

Kesimpulan

1. Pada ADC *tanpa suara*, inputan mic nilainya kecil, sehingga tegangan yang didapatkan rendah (2,32 volt)
2. Pada ADC *dengan inputan suara* (tepuk) inputan yang didapatkan tinggi sehingga penguatannya tinggi sehingga tegangan yang didapatkan besar (3,12 volt)

4.4. Pengujian pada Motor DC

4.4.1.Tujuan

Tujuan dari pengujian alat ini adalah untuk mengetahui berapa besar tegangan yang dikeluarkan motor pada saat motor bekerja putar ke kanan dan putar ke kiri.

4.4.2.Peralatan yang digunakan

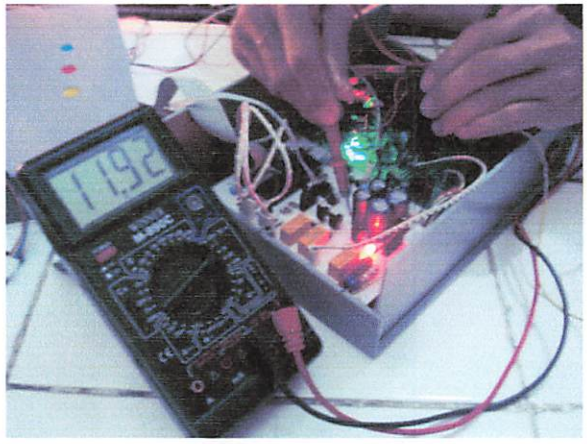
1. 1 buah multimeter winner M-890C
2. Rangkaian Motor DC
3. 1 buah Lampu Bohlam

4.4.3. Langkah pengujian

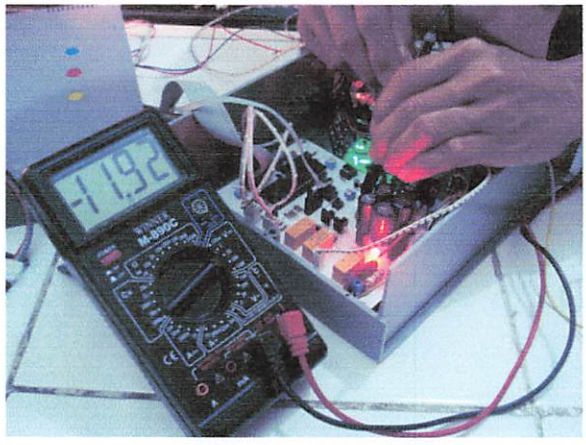
1. Hubungkan kabel multimeter pada output rangkaian motor DC
2. Motor digerakkan putar ke kanan

3. Ukur tegangan pada output rangkaian motor DC
4. Motor digerakkan putar ke kiri
5. Ukur tegangan pada output rangkaian motor DC

4.4.4. Gambar pengujian alat



Gambar 4.8 pengukuran saat motor DC putar ke kanan



Gambar 4.9 pengukuran motor DC putar ke kiri

Tegangan yang dihasilkan pengujian pada motor putar ke kiri sebesar 11,92 volt, sedangkan pada saat motor putar ke kanan sebesar – 11,92 volt

Kesimpulan

1. Voltase yang dihasilkan pada putaran menunjukkan bahwa untuk nilai tegangan pada saat motor putar ke kanan mendapat nilai +(plus), sedangkan untuk putaran ke kiri nilainya - (min)

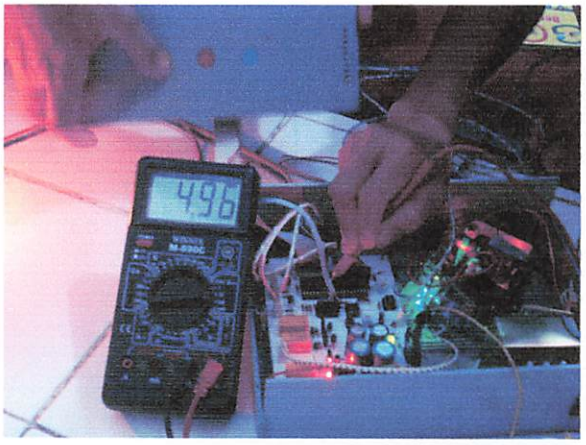
4.5. Pengujian tegangan AC saat lampu hidup level 1 hingga level 4

4.5.1. Tujuan

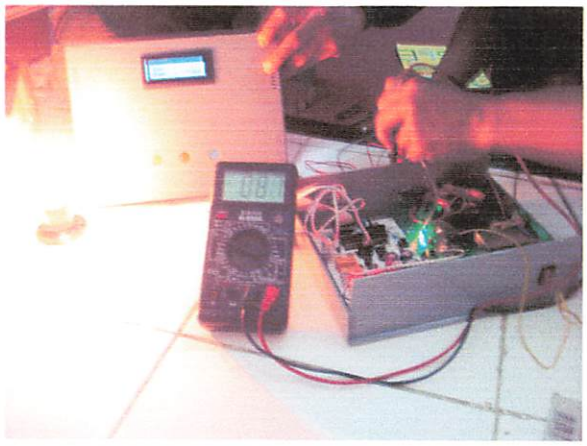
Pada pengujian ini bertujuan untuk mengetahui tegangan output yang keluar pada saat lampu menyala level 1 hingga level 4

4.5.2. Peralatan yang digunakan

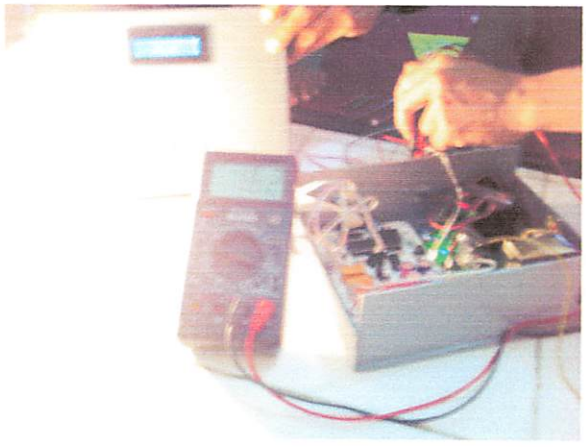
1. 1 buah multimeter winner M-890C
2. Rangkaian lampu ON
3. 1 buah lampu bohlam



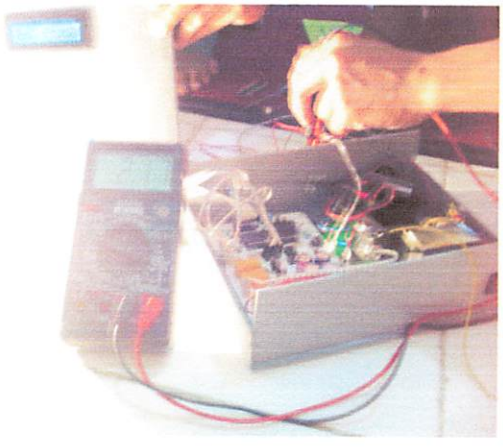
Gambar 4.10 Tegangan lampu pada saat level 1 (minimum)



Gambar 4.11 Tegangan lampu pada saat level 2



Gambar 4.12 Tegangan lampu pada saat level 3



Gambar 4.13 Tegangan lampu pada saat level 4 (maksimum)

Hasil pengukuran tegangan nya adalah sebagai berikut;

1. Untuk lampu menyala pada level 1 (minimum) didapatkan tegangan pada output lampu sebesar 4,96 volt
2. Untuk lampu menyala pada level 2, didapatkan tegangan pada output lampu sebesar 8,7 volt
3. Untuk lampu menyala pada level 3, didapatkan tegangan pada output lampu sebesar 16,2 volt
4. Dan untuk lampu menyala pada level 4, didapatkan tegangan pada output lampu sebesar 21,7 volt

Posisi Lampu	Tegangan Sumber (Volt)	Pengujian dengan Mulitimeter		Perhitungan		Error
		R (ohm)	I (Arus)	R(ohm)	I (Arus)	
Level 1 (Minimum)	220	1692.4	0.13	1692.4	0.129993	5,38%
Level 2	220	1378	0.16	1375	0.15	6,66%
Level 3	220	956.6	0.23	956.6	0.229981	8,26%
Level 4 (Maksimum)	220	666.67	0.33	666.67	0.329998	6,06%

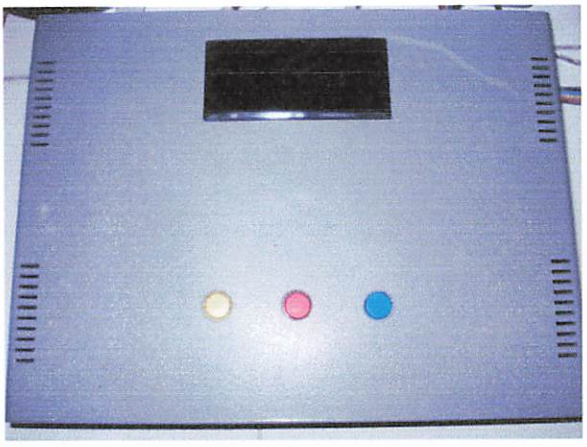
Tabel 4.1 Pengujian dan Perhitungan

$$\text{Error} = \frac{\text{Pengujian - Perhitungan}}{\text{Perhitungan}} \times 100\%$$

4.6. Pengujian dengan tombol manual

4.6.1 Tujuan

Pada pengujian tombol manual tegangan yang dihasilkan sama dengan Prosedur pada rangkaian ADC, beda nya dengan ADC adalah; pada ADC prosedur pengontrolan lampu menggunakan *suara* (tepuk) sebagai tombol up (terang) dan down (redup), sedangkan pada penggunaan manual digunakan 3 buah tombol *NO* (normaly Open) untuk mengontrolnya, adapun gambar, warna dan kegunaan tombol sebagai berikut:



Gambar 4.14 mode tombol

4.6.2. fungsi masing masing tombol

1. Tombol warna merah, agar lampu menyala terang
2. Tombol warna kuning untuk meredupkan lampu
3. Sedangkan tombol warna biru untuk tombol mode, tombol mode disini digunakan untuk memberi 2 pilihan untuk mengoperasikan lampu dengan cara manual atau dengan menggunakan sensor suara (tepukan)

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan pada teori-teori yang telah dibahas Maka dapat ditarik kesimpulan sebagai berikut

1. Saat sensor tidak dihalangi maka port mikrokontroler terhubung ground sehingga sensor berlogika "low" sehingga mendapat tegangan 0,02 volt
2. Dan pada saat sensor dihalangi, sensor mendapat logika "high" dari pull-up, sehingga mendapat tegangan 4,38 volt
3. Pada ADC *tanpa suara*, inputan mic nilainya kecil, sehingga tegangan yang didapatkan rendah (2,32 volt)
4. Pada ADC *dengan inputan suara* (tepuk) inputan yang didapatkan tinggi sehingga penguatannya tinggi dan tegangan yang didapatkan juga besar (3,12 volt)
5. Pada percobaan motor dc arah putaran motor ditentukan hardware dan software pada mikrokontroler AT89S52
6. Pada pengujian redup terang lampu menggunakan potensio (hal 50) dimmer berfungsi sebagai pengontrolnya, yang jika nilai R besar maka arus kecil sehingga lampu menjadi redup, dan jika nilai R kecil maka arus yang dikeluarkan besar sehingga lampu menyala terang
7. Dalam pengujian berulang ulang alat yang saya buat ini dapat bekerja dengan baik.

5.2. Saran

Beberapa tambahan yang diperlukan dalam mengembangkan kemampuan alat ini adalah:

1. Menambah kemampuan alat, agar dapat diaplikasikan untuk pengontrolan alat elektronik yang lain seperti AC,TV,Heater dll
2. Menambah tenaga cadangan pada rangkaian, dalam hal ini baterai atau UPS, yang bilamana terjadi mati listrik sensor dan rangkaian nya akan tetap bekerja sehingga data orang yang masuk, orang yang keluar, atau kondisi nyala lampu dapat dibaca dan disimpan, sehingga bila listrik sudah menyala data orang yang masuk, orang keluar dan juga kondisi lampu dapat menyala sesuai dengan kerja nya rangkaian pada saat menggunakan tenaga cadangan tadi

DAFTAR PUSTAKA

[1]. www.atmel.com

[2].lampu http://id.wikipedia.org

[3].http//www.duniaelektronika.com

[4]. http://id.wikipedia.org

[5]. fisik@net -http://www.fisika.net

[6].Yulikuspartono, S.kom, *pengantar logika dan algoritma*

[7]. http://www.semiconductors.philips.com

[8]. www.national.com

LAMPIRAN

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

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

NAMA : SIMON RICHARD
NIM : 0217088
Perbaikan meliputi :

1. Tata tulis laporan diperbaiki diliat di buku laporannya.
2. Pengujian bab IV. ditambahkan bab IV
3. Gambar 3 rangkaian diperbaiki

Malang, 23 - 03 2009

(_____)

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

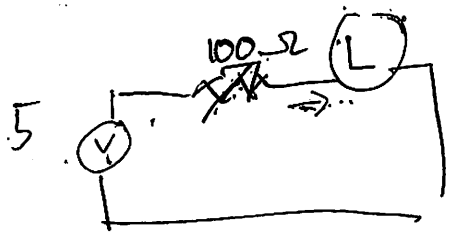
Formulir Perbaikan Ujian Skripsi

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

NAMA : SIMON RICHARD
NIM : 0217088
Perbaikan meliputi :

Malang, 200

( )



$$I = \frac{V}{R}$$

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

FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian komprehensip jenjang strata Satu (S-1) Jurusan Teknik Elektro, Konsentrasi Elektronika yang diselenggarakan pada:

Hari : Senin
Tanggal : 23 Maret 2009

Telah dilakukan perbaikan skripsi oleh :

Nama : SIMON RICHARD
N.I.M : 02.17.088
Masa Bimbingan : 05 Januari 2009 – 05 Juni 2009
Judul : "PERANCANGAN DAN PEMBUATAN ALAT PENGONTROL DIMMER LAMPU DALAM RUANGAN BERBASIS MIKROKONTROLLER AT89S52"

Perbaikan Meliputi :

No	Tanggal	Materi Perbaikan	Paraf Dosen Penguji
1.	23 – 03 - 2009	Tata tulis laporan diperbaiki. Dilihat dibuku laporan nya	
2.	23 – 03 – 2009	Pengujian Bab IV, Ditambahkan ke Bab IV	
3.	23 – 03 – 2009	Gambar gambar Rangkaian diperbaiki	

Disetujui Oleh:

Penguji Pertama

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

Penguji Kedua

(Irmalia Suryani Faradisa ST, MT)
NIP.P.1030100365

Mengetahui,
Dosen Pembimbing

(I Komang Somawirata, ST, MT)
NIP : P.1030100361

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

FORMULIR BIMBINGAN SKRIPSI

Nama : SIMON RICHARD
NIM : 0217088
Masa Bimbingan : 05 januari 2009 – 05 juni 2009
Judul Skripsi : Perancangan Dan Pembuatan Alat Pengontrol Dimmer Lampu Dalam Ruang Berbasis Mikrokontroller AT89S52

No.	Tanggal	Uraian	Paraf Pembimbing
1.	02-02-2009	Revisi Bab 1, 2, 3	
2.	16-02-2009	Acc bab 1, 2, Revisi Bab 3, maju Bab 4	
3.	24-02-2009	Acc Bab 3 Revisi Bab 4 dan foto alat	
4.	25-02-2009	Acc Bab 4, maju Bab 5	
5.	27-02-2009	Acc Bab 5	
6.	28-02-2009	Acc Makalah Seminar Hasil	
7.	21-03-2009	Acc Maju Ujian Skripsi	
8.			
9.			
10.			

Malang, 2009
Dosen Pembimbing

I Komang Somawirata ST, MT
NIP.P. 1030100361

Listing Program

```
/**************** Subrutin Serial EEPROM ******************/

#define FADDR 0xa0;
#define PADDR 0x00;

sbit sda = P3^7; // pin 6
sbit scl = P3^6; // pin 5

unsigned char b,a,zdata,addr_lo,x;
bit c;

nop(){

}

delay_eeprom(unsigned char max) {
  unsigned char i;
  for(i=0;i != max;i++);
}

NAK() {
      sda = 1;
      nop();nop();nop();nop();
      scl = 1;
      nop();nop();nop();nop();nop();nop();
      scl = 0;
}



stop() {
      sda = 0;
      nop();nop();nop();nop();
      scl = 1;
      nop();nop();nop();nop();nop();nop();
      sda = 1;
}

start(){

      //Send START, defined as high-to-low SDA with SCL high.
      //Return with SCL, SDA low.
      //Returns CY set if bus is not available.
```

```
    sda = 1;
    scl = 1;

    if( (sda) && (scl) ) {
        nop();nop();
        sda = 0;
        nop();nop();nop();nop();nop();nop();nop();
        scl = 0;
        c  = 0;
    } else {
        c  = 1;
    }
}


shout(){
unsigned char i;
    for(i=0;i<=7;i++){
        if(a & 0x80) sda =1; else sda = 0;
        a = a << 1;
        nop();nop();
        scl = 1;
        nop();nop();nop();nop();nop();nop();
        scl = 0;
    }
    sda = 1;
    nop();nop();nop();nop();
    scl = 1;
    nop();nop();nop();nop();nop();nop();
    if(sda) c=1;else c = 0;
    scl = 0;
}


shin() {
unsigned char i;
    c = 0;
    sda = 1;
    for (i=0;i<=7;i++){
        nop();nop();nop();nop();nop();nop();
        scl = 1;
        nop();nop();nop();nop();
        a = a << 1;
        if(sda) a = a | 0x01; else a = a & 0xfe;
        scl  = 0;
```

```
    }
}


read_current(){
    //AT24Cxx Current Address Read function.
    //Called with programmable address in A. Returns data in A.
    //Returns CY set to indicate that the bus is not available
    //or that the addressed device failed to acknowledge.
    delay_eeprom(100);
    start();
    if(c==0) {
        a = a << 1;
        a = a | FADDR
        a = a | 0x01;
        shout();
        if(c==1) {
            stop();
        } else {
            shin();
            NAK();
            c = 0;
            stop();
        }
    }

}

read_random(){

    //AT24Cxx Random Read function.
    //Called with programmable address in A, byte address in
    //register pair ADDR_HI:ADDR_LO. Returns data in A.
    //Returns CY set to indicate that the bus is not available
    //or that the addressed device failed to acknowledge.

    b = a;

    //Send dummy write command to set internal address.
    start();
    if(c==0) {
        a = a << 1;
        a = a | FADDR
        a = a & 0xfe;
        shout();
```

```
        if(c==1) {
            stop();
        } else {
            a = addr_lo;
            shout();
            if(c==1) { stop();
            } else {
                //Call Current Address Read function.
                a = b;
                read_current();
            }
        }
    }
}


write_byte(){

    //AT24Cxx Byte Write function.
    //Called with programmable address in A, byte address in
    //register pair ADDR_HI:ADDR_LO, data in register XDATA.
    //Does not wait for write cycle to complete.
    //Returns CY set to indicate that the bus is not available
    //or that the addressed device failed to acknowledge.
    //Destroys A.

    start();
    if(c==0) {
        a = a << 1;
        a = a | FADDR
        a = a & 0xfe;
        shout();
        if(c==1) { stop();
        } else {
            a = addr_lo;
            shout();
            if(c==1) { stop();
            } else {
                a = zdata;
                shout();
                if(c==1) { stop();
                } else {
                    c = 0;
                    stop();
                }
```

```
                }
            }
        }
}

unsigned char read_eeprom(unsigned char alamat) {
unsigned char i;
    c = 0;
    for(i=0;i<=8;i++){
        a = PADDR;
        addr_lo = alamat;
        read_random();
        if(c==0)break;
    }
    return(a);
}


bit  write_eeprom(unsigned char alamat,dataa)  {
unsigned char i;
    c = 0;
    for(i=0;i<=8;i++){
        a = PADDR;
        addr_lo = alamat;
        zdata = dataa;
        write_byte();
        if(c==0)break;
    }
    return(c);
}

/************* Akhir Subrutin Serial EEPROM *****************/
```

```
delayMSEC(unsigned int max) {
  unsigned char j;
  unsigned int i;
  for(i=0;i !=max;i++)
      for(j=0;j !=131;j++);
}
/*
delaySEC(unsigned int max) {
  unsigned char j;
  unsigned int i,k;
  for(i=0;i !=max;i++)
      for(j=0;j !=131;j++)
         for(k=0;k !=600;k++);
}

 */
```

```
#include <c:\Gaza04X\FileH\reg52.h>
#include <c:\Gaza04X\FileH\DelayW.h>
#include <c:\Gaza04X\FileH\LcdiW4b.h>
#include <c:\Gaza04X\FileH\24c01aW.h>

sbit switch_ac   = P1^0;  // int1
sbit switch_motor = P1^1;
sbit arah_motor  = P1^2;

sbit sensor1     = P1^3;
sbit sensor2     = P1^4;

sbit start_adc   = P0^6;
sfr  data_adc    = 0xA0;

sbit  dimm_up    = P1^5;
sbit  dimm_dwn   = P1^6;

sbit  pb_mode    = P1^7;

unsigned char count,baca_adc,data_old,data_new,pengunjung;
unsigned char index,z,potensio;
unsigned int counter_voice,delay_scan = 0;//countH,countL;

bit mode,tampil,dec,stchar;
bit sen1_old,sen1_new,sen2_old,sen2_new;
bit sen3_old,sen3_new,sen4_old,sen4_new;
bit sen5_old,sen5_new,sen6_old,sen6_new;
bit fmasuk,fkeluar,lampL,lampH,lock = 0;


clear(){
  hapus_layar();
  delayMSEC(10);
}

external_nol() interrupt 0 using 2 {
     counter_voice++;
}

void redupkan(){
   clear();
   tulis(1,1,"  L a m p u   ");
   tulis(1,2,"  R e d u p !  ");
   arah_motor  = 0; delayMSEC(50);
```

```
    switch_motor = 0; delayMSEC(400);
    switch_motor = 1;
    potensio--;
    //tulis_huruf(16,1,potensio);
    write_eeprom(5,potensio);
    tampil = 1;
}

void terangkan(){
    clear();
    tulis(1,1,"  L a m p u   ");
    tulis(1,2," T e r a n g ! ");
    arah_motor  = 1; delayMSEC(50);
    switch_motor = 0; delayMSEC(400);
    switch_motor = 1;
    potensio++;
    write_eeprom(5,potensio);
    tampil = 1;
}


void cek_pintu(){

    sen5_old = sen5_new;
    sen5_new = sensor1;

    if((!sen5_old)&&(sen5_new)){
      if((fkeluar==0)&&(!sensor2)) fmasuk=1;
      else
      if((fkeluar==1)&&(pengunjung>0)) {   // &&(sensor2==1)
        pengunjung--;
        if(pengunjung>0) {tulis_byte(8,1,pengunjung);} else {tulis(8,1,"---");}
        write_eeprom(10,pengunjung);
        if(pengunjung==0) {switch_ac = 1; write_eeprom(15,0);
tulis_huruf(14,1,'0');}
        fmasuk = fkeluar = 0;
        sen6_old = sen6_new = 0;
        sen5_old = sen5_new = 0;
      }
      delayMSEC(10);
    }

    sen6_old = sen6_new;
    sen6_new = sensor2;
```

```
    if((!sen6_old)&&(sen6_new)){
      if((fmasuk==0)&&(!sensor1)) fkeluar=1;
      else
      if((fmasuk==1)&&(pengunjung<200)) {  // &&(sensor1==1)
        pengunjung++;
        tulis_byte(8,1,pengunjung);
        write_eeprom(10,pengunjung);
        switch_ac = 0;
        write_eeprom(15,1);
        tulis_huruf(14,1,'1');
        fmasuk = fkeluar = 0;
        sen6_old = sen6_new = 0;
        sen5_old = sen5_new = 0;
      }
      delayMSEC(10);
    }
}


void cek_button(){

    sen1_old = sen1_new;
    sen1_new = dimm_up;

    if((sen1_old)&&(!sen1_new)&&(potensio>1)){
      clear();
      tulis(1,1,"  L a m p u   ");
      tulis(1,2,"  R e d u p !  ");
      arah_motor  = 0; delayMSEC(50);
      switch_motor = 0; delayMSEC(400);
      switch_motor = 1;
      potensio--;
      write_eeprom(5,potensio);
      tampil = 1;
    }

    sen2_old = sen2_new;
    sen2_new = dimm_dwn;

    if((sen2_old)&&(!sen2_new)&&(potensio<=3)){
      clear();
      tulis(1,1,"  L a m p u   ");
      tulis(1,2,"  T e r a n g !  ");
      arah_motor  = 1; delayMSEC(50);
      switch_motor = 0; delayMSEC(400);
```

```
        switch_motor = 1;
        potensio++;
        write_eeprom(5,potensio);
        tampil = 1;
    }

}

void main(){

    inisialisasi_lcd();
    clear();

    EA  = 1;
    EX0 = 0;
    IT0 = 1;

    sensor1 = 1;
    sensor2 = 1;
    switch_ac = 1;
    switch_motor = 1;
    arah_motor  = 1;

    dimm_up  = 1;
    dimm_dwn = 1;

    index = 0;
    tulis(1,1,"              ");
    tulis(1,2,"              ");
    delayMSEC(100);
    clear();
    delayMSEC(100);
    tulis(1,1,"Alat Pengontrol ");
    tulis(1,2," D i m m e r  ");
    delayMSEC(1000);
    tulis(1,1,"   Oleh :     ");
    tulis(1,2,"              ");
    delayMSEC(800);
    tulis(1,1," Richard Simon  ");
    tulis(1,2," NIM 0217088  ");
    delayMSEC(1500);
    start_adc = 0;

    baca_adc = 0;
    tampil  = 0;
```

```
    fmasuk  = 0;
    fkeluar  = 0;
    sen5_new = 0;
    sen5_old = 0;
    sen6_new = 0;
    sen6_old = 0;

    start_adc = 1;

    if(read_eeprom(9)!='$'){ write_eeprom(10,0);  write_eeprom(9,'$');}    // reset
pengunjung

    pengunjung = read_eeprom(10);

    potensio  = read_eeprom(5);
    pb_mode   = 1;
    z         = read_eeprom(7);                    // address 7 = mode
    if(z==1) mode = 1; else {write_eeprom(7,0); mode = 0;}

    clear();

    tulis(1,1,"Guest: --- L:1/1");
    tulis(1,2,"Mode :    VCn: ");

    if(pengunjung>0) tulis_byte(8,1,pengunjung);
    tulis_huruf(8,2,read_eeprom(7)|0x30);
    if(read_eeprom(15)==0) tulis_huruf(14,1,'0'); else tulis_huruf(14,1,'1');
    tulis_huruf(16,1,potensio|0x30);
    if(counter_voice<10) tulis_huruf(16,2,counter_voice|0x30); else
    if(counter_voice>=10) tulis_huruf(16,2,'H');
    delay_scan = 0;

    while(1){


        tulis_huruf(8,2,mode|0x30);

        data_old = data_new;
        data_new = data_adc;
        baca_adc = data_new;


        tulis_byte(13,2,baca_adc);

        cek_pintu();
```

```
        if(mode==0) {
          if(pengunjung>0) switch_ac = 0; else if(pengunjung==0) switch_ac = 1;
          if(counter_voice<10) tulis_huruf(16,2,counter_voice|0x30); else
          if(counter_voice>=10) tulis_huruf(16,2,'H');
          if(delay_scan<60) {EX0 = 1; delay_scan++;}
          else if(delay_scan>=60) {
            EX0 = 0;
            delay_scan = 0;
            if((counter_voice==5)&&(potensio<=3)) {terangkan();}
            else if((counter_voice==2)&&(potensio>1)) {redupkan();}
            counter_voice = 0;
          }
        } else
        if((read_eeprom(15)>0)&&(mode==1)) {EX0 = 0; counter_voice = 0;
cek_button();}

        sen4_old = sen4_new;
        sen4_new = pb_mode;

        if((sen4_old)&&(!sen4_new)){
          if(mode==0) { mode = 1; write_eeprom(7,1); }
          else { mode = 0; write_eeprom(7,0); }
          delayMSEC(100);
        }

        if (tampil){
          clear();
          tampil = 0;
          tulis(1,1,"Guest: --- L:1/1");
          tulis(1,2,"Mode :    VCn: ");

          if(pengunjung>0) tulis_byte(8,1,pengunjung);
          tulis_huruf(8,2,read_eeprom(7)|0x30);
          if(read_eeprom(15)==0) tulis_huruf(14,1,'0'); else tulis_huruf(14,1,'1');
          tulis_huruf(16,1,potensio|0x30);
          if(counter_voice<10) tulis_huruf(16,2,counter_voice|0x30); else
          if(counter_voice>=10) tulis_huruf(16,2,'H');
          /*tulis(1,1,"Guest: --- Light");
          tulis(1,2,"Mode :");
          tulis_byte(8,1,pengunjung);
          tulis_huruf(8,2,mode|0x30);
          if(read_eeprom(15)>0) tulis(13,2,"On "); else tulis(13,2,"Off");      */
          if(mode==0) EX0 = 1;
        }
```

```
            delayMSEC(50);
        }
    }
```

```
/*
  LCD.H
  Berisi rutin-rutin display LCD 4 Bit
*/


sbit rs      = P0^0;
sbit en      = P0^1;
sbit D0      = P0^2;
sbit D1      = P0^3;
sbit D2      = P0^4;
sbit D3      = P0^5;


unsigned char __P0=0x00;


void lcddelay()
{
  char i;
  for (i=1;i<=50;i++);
}

update_low(){
      if (__P0 & 0x01) D0 =1; else D0 = 0;
      if (__P0 & 0x02) D1 =1; else D1 = 0;
      if (__P0 & 0x04) D2 =1; else D2 = 0;
      if (__P0 & 0x08) D3 =1; else D3 = 0;
}

update_high(){
      if (__P0 & 0x10) D0 =1; else D0 = 0;
      if (__P0 & 0x20) D1 =1; else D1 = 0;
      if (__P0 & 0x40) D2 =1; else D2 = 0;
      if (__P0 & 0x80) D3 =1; else D3 = 0;
}


writetolcd8(unsigned char registr, unsigned char dataa) {
 if (registr) rs = 1; else rs = 0;
 __P0 = dataa;
 update_low();

// PORTDATA = dataa | 0xf0;
 lcddelay();
```

```
  en = 0;
  lcddelay();
  en = 1;
  lcddelay();
}

writetolcd(unsigned char registr, unsigned char dataa) {
  if (registr) rs = 1; else rs = 0;
  __P0 =  dataa;
  update_high();

  en = 0;
  lcddelay();
  en = 1;
  lcddelay();

  __P0 =  dataa;
  update_low();

  en = 0;
  lcddelay();
  en = 1;
  lcddelay();
}


inisialisasi_lcd() {
   writetolcd8(0,0x03);
   lcddelay(); lcddelay(); lcddelay();
   lcddelay(); lcddelay(); lcddelay();
   writetolcd8(0,0x03);
   lcddelay(); lcddelay(); lcddelay();
   lcddelay(); lcddelay(); lcddelay();
   writetolcd8(0,0x03);
   writetolcd8(0,0x02);
   writetolcd8(0,0x02);
   writetolcd8(0,0x08);
   writetolcd8(0,0x00);
   writetolcd8(0,0x01);
   writetolcd8(0,0x00);
   writetolcd8(0,0x06);
   writetolcd8(0,0x06);
}
```

```
void blink(char i)
{
  if (i==0)
    writetolcd(0, 0xC);
  else
    writetolcd(0, 0xD);
  lcddelay();
}

void gotoxy(char x, char y)
{
  if (y==1)
    writetolcd(0, 0x80 + x - 1);
  else if (y==2)
    writetolcd(0, 0xC0 + x - 1);
}

void tulis_huruf(char x, char y, char ch)
{
  gotoxy(x, y);
  lcddelay();
  writetolcd(1, ch);
  lcddelay();
}

void tulis(char x, char y, char* s)
{
  char i = 0;

  while(s[i] != 0)
  {
    tulis_huruf(x+i, y, s[i]);
    i++;
  }
}

void tulis_byte(char x, char y, unsigned char s){
   gotoxy(x,y);
   writetolcd(1, (s / 100) | 0x30);
   writetolcd(1, ((s % 100)/10) | 0x30);
   writetolcd(1, (s % 10) | 0x30);
}
/*
void tulis_hex(char x, char y, unsigned char s){
```

```
unsigned char d;
   gotoxy(x,y);
   d = s >> 4;

   if (d >= 10) {
      d = d - 10 + 'A';
      writetolcd(1, d);
   } else {
      writetolcd(1, d | 0x30);
   }

   d = s & 0x0f;
   if(d>=10) {
      d=d-10 + 'A';
      writetolcd(1, d);
   } else  {
      writetolcd(1, d | 0x30);
   }
}

void tulis_waktu(char x, char y, unsigned char s,unsigned char t,unsigned char u, char
v){
   gotoxy(x,y);
   writetolcd(1, (s >> 4   ) | 0x30);
   writetolcd(1, (s  & 0x0f) | 0x30);

   writetolcd(1, v);

   writetolcd(1, (t >> 4   ) | 0x30);
   writetolcd(1, (t  & 0x0f) | 0x30);

   writetolcd(1, v);

   writetolcd(1, (u >> 4   ) | 0x30);
   writetolcd(1, (u  & 0x0f) | 0x30);
}
*/

/*
void tulis_int(char x, char y, unsigned int s){
unsigned int t;
unsigned char l;
   gotoxy(x,y);
   t = (s / 10000);  l = t; writetolcd(1, l | 0x30);
   t = (s % 10000) / 1000; l = t; writetolcd(1, l | 0x30);
```

```
   t = (s % 1000)  /  100; l = t; writetolcd(1, l | 0x30);
   t = (s % 100)   /   10; l = t; writetolcd(1, l | 0x30);
   t = (s % 10)    /    1; l = t; writetolcd(1, l | 0x30);
}
/*
void tulis_int51(char x, char y, unsigned long int s){
unsigned long int t;
unsigned char l;
   gotoxy(x,y);

   t = (s / 10000);  l = t; if (l >0) writetolcd(1, l | 0x30); else writetolcd(1,' ');
   t = (s % 10000) / 1000; l = t; writetolcd(1, l | 0x30);
   writetolcd(1,'.');
   t = (s % 1000)  /  100; l = t; writetolcd(1, l | 0x30);
   t = (s % 100)   /   10; l = t; writetolcd(1, l | 0x30);
   t = (s % 10)    /    1; l = t; writetolcd(1, l | 0x30);
   writetolcd(1,'V');
}

*/
void tulis_int30(char x, char y, unsigned int st){
unsigned int t,s;
unsigned char l;
   s = st;
   gotoxy(x,y);
   if (s > 999) s = 999;
   t = (s / 100);       l = t; writetolcd(1, l | 0x30);
   t = (s % 100) / 10; l = t; writetolcd(1, l | 0x30);
   //writetolcd(1,'.');
   t = (s % 10)  /  1; l = t; writetolcd(1, l | 0x30);
}

/*
void tulis_int31(char x, char y, unsigned int st){
unsigned int t,s;
unsigned char l;
   s = st;
   gotoxy(x,y);
   if (s > 9999) s = 9999;
   t = (s / 1000);       l = t; writetolcd(1, l | 0x30);
   t = (s % 1000) / 100; l = t; writetolcd(1, l | 0x30);
   t = (s % 100) / 10; l = t; writetolcd(1, l | 0x30);
   writetolcd(1,'.');
   t = (s % 10)  /  1; l = t; writetolcd(1, l | 0x30);
}
```

```
/*
void tulis_int32(char x, char y, unsigned long int st){
unsigned int t,s;
unsigned char l;
   s = st;
   gotoxy(x,y);
   if (s > 99999) s = 99999;
   t = (s / 10000);       l = t; writetolcd(1, l | 0x30);
   t = (s % 10000) / 1000; l = t; writetolcd(1, l | 0x30);
   t = (s % 1000) / 100;  l = t; writetolcd(1, l | 0x30);
   t = (s % 100) / 10;    l = t; writetolcd(1, l | 0x30);
   writetolcd(1,'.');
   t = (s % 10) / 1;      l = t; writetolcd(1, l | 0x30);
}



void tulis_int322(char x, char y, unsigned long int st){
unsigned int t,s;
unsigned char l;
   s = st;
   gotoxy(x,y);
   //if (s > 99999) s = 99999;
   t = (s / 10000);       l = t; writetolcd(1, l | 0x30);
   t = (s % 10000) / 1000; l = t; writetolcd(1, l | 0x30);
   t = (s % 1000) / 100;  l = t; writetolcd(1, l | 0x30);
   t = (s % 100) / 10;    l = t; writetolcd(1, l | 0x30);
   t = (s % 10) / 1;      l = t; writetolcd(1, l | 0x30);
}



void tulis_int32T(char x, char y, unsigned long int st){
unsigned int t,s;
unsigned char l;
   s = st;
   gotoxy(x,y);
   if (s > 99999) s = 99999;
   t = (s / 10000);       l = t; writetolcd(1, 0x31);
   t = (s % 10000) / 1000; l = t; writetolcd(1, l | 0x30);
   t = (s % 1000) / 100;  l = t; writetolcd(1, l | 0x30);
   t = (s % 100) / 10;    l = t; writetolcd(1, l | 0x30);
   writetolcd(1,'.');
   t = (s % 10) / 1;      l = t; writetolcd(1, l | 0x30);
}
```

```
void tulis_int33(char x, char y, unsigned int st){
unsigned int t,s;
unsigned char l;
   s = st;
   gotoxy(x,y);
   if (s > 2359) s = 0000;
   t = (s / 1000);       l = t; writetolcd(1, l | 0x30);
   t = (s % 1000) / 100; l = t; writetolcd(1, l | 0x30);
   writetolcd(1,':');
   t = (s % 100) / 10; l = t; writetolcd(1, l | 0x30);
   t = (s % 10)  /  1; l = t; writetolcd(1, l | 0x30);
}
*/
/*
void tulis_bin(char x, char y, unsigned char s){
   gotoxy(x,y);
   if (s & 0x80) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x40) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x20) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x10) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x08) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x04) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x02) writetolcd(1,'1'); else writetolcd(1,'0');
   if (s & 0x01) writetolcd(1,'1'); else writetolcd(1,'0');
}
 */

void hapus_layar()
{
  writetolcd(0, 0xC);
  lcddelay();
  writetolcd(0, 6);
  lcddelay();
  writetolcd(0, 1);
  lcddelay();
}

/* End Of File */
```

```
/*
 *-------------------------------------------------------------------------
 * Copyright (c) KEIL ELEKTRONIK GmbH and Franklin Software, Inc., 1987-
1992
 *-------------------------------------------------------------------------
 */
/* 8052 Processor Declarations */

/* BYTE Registers */
sfr P0   = 0x80;
sfr P1   = 0x90;
sfr P2   = 0xA0;
sfr P3   = 0xB0;
sfr PSW  = 0xD0;
sfr ACC  = 0xE0;
sfr B    = 0xF0;
sfr SP   = 0x81;
sfr DPL  = 0x82;
sfr DPH  = 0x83;
sfr PCON = 0x87;
sfr TCON = 0x88;
sfr TMOD = 0x89;
sfr TL0  = 0x8A;
sfr TL1  = 0x8B;
sfr TH0  = 0x8C;
sfr TH1  = 0x8D;
sfr IE   = 0xA8;
sfr IP   = 0xB8;
sfr SCON = 0x98;
sfr SBUF = 0x99;

/* 8052 Extensions */
sfr T2CON = 0xC8;
sfr RCAP2L = 0xCA;
sfr RCAP2H = 0xCB;
sfr TL2  = 0xCC;
sfr TH2  = 0xCD;


/* BIT Registers */
/* PSW */
sbit CY   = 0xD7;
sbit AC   = 0xD6;
sbit F0   = 0xD5;
sbit RS1  = 0xD4;
```

```
sbit RS0  = 0xD3;
sbit OV   = 0xD2;
sbit P    = 0xD0;

/* TCON */
sbit TF1  = 0x8F;
sbit TR1  = 0x8E;
sbit TF0  = 0x8D;
sbit TR0  = 0x8C;
sbit IE1  = 0x8B;
sbit IT1  = 0x8A;
sbit IE0  = 0x89;
sbit IT0  = 0x88;

/* IE */
sbit EA   = 0xAF;
sbit ES   = 0xAC;
sbit ET1  = 0xAB;
sbit EX1  = 0xAA;
sbit ET0  = 0xA9;
sbit EX0  = 0xA8;

/* IP */
sbit PS   = 0xBC;
sbit PT1  = 0xBB;
sbit PX1  = 0xBA;
sbit PT0  = 0xB9;
sbit PX0  = 0xB8;

/* P3 */
sbit RD   = 0xB7;
sbit WR   = 0xB6;
sbit T1   = 0xB5;
sbit T0   = 0xB4;
sbit INT1 = 0xB3;
sbit INT0 = 0xB2;
sbit TXD  = 0xB1;
sbit RXD  = 0xB0;

/* SCON */
sbit SM0  = 0x9F;
sbit SM1  = 0x9E;
sbit SM2  = 0x9D;
sbit REN  = 0x9C;
sbit TB8  = 0x9B;
```

```
sbit RB8  = 0x9A;
sbit TI   = 0x99;
sbit RI   = 0x98;

/* 8052 Extensions */
/* IE */
sbit ET2  = 0xAD;

/* IP */
sbit PT2  = 0xBD;

/* P1 */
sbit T2EX = 0x91;
sbit T2   = 0x90;

/* T2CON */
sbit TF2  = 0xCF;
sbit T2IP = 0xCE;
sbit T2IE = 0xCD;
sbit T2RSE = 0xCC;
sbit BGEN = 0xCB;
sbit TR2  = 0xCA;
sbit C_T2 = 0xC9;
sbit CP_RL2= 0xC8;
```

DATA SHEET

Microcontroller
(AT89S52)

atures

- ɔmpatible with MCS®-51 Products
- ‹ Bytes of In-System Programmable (ISP) Flash Memory
  - – Endurance: 1000 Write/Erase Cycles
- 0V to 5.5V Operating Range
- ılly Static Operation: 0 Hz to 33 MHz
- ıree-level Program Memory Lock
- ;6 x 8-bit Internal RAM
- ? Programmable I/O Lines
- ıree 16-bit Timer/Counters
- ight Interrupt Sources
- ıll Duplex UART Serial Channel
- ɔw-power Idle and Power-down Modes
- terrupt Recovery from Power-down Mode
- 'atchdog Timer
- ıal Data Pointer
- ɔwer-off Flag
- ıst Programming Time
- exible ISP Programming (Byte and Page Mode)
- reen (Pb/Halide-free) Packaging Option

Description

AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K ɜs of in-system programmable Flash memory. The device is manufactured using el's high-density nonvolatile memory technology and is compatible with the indus- ıtandard 80C51 instruction set and pinout. The on-chip Flash allows the program nory to be reprogrammed in-system or by a conventional nonvolatile memory pro- nmer. By combining a versatile 8-bit CPU with in-system programmable Flash on onolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a ıly-flexible and cost-effective solution to many embedded control applications.

AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes AM, 32 I/O lines, Watchdog timer, two data pointers, three 16-bit timer/counters, a vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, clock circuitry. In addition, the AT89S52 is designed with static logic for operation n to zero frequency and supports two software selectable power saving modes. Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and rupt system to continue functioning. The Power-down mode saves the RAM con- s but freezes the oscillator, disabling all other chip functions until the next interrupt ardware reset.

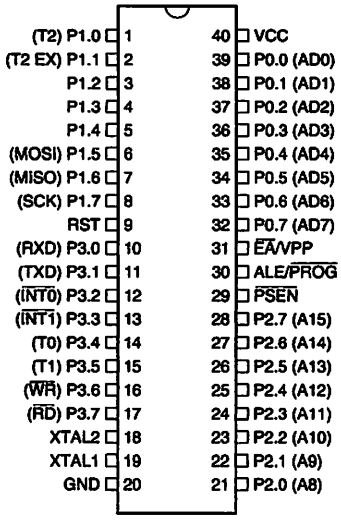
8-bit Microcontroller with 8K Bytes In-System Programmable Flash

AT89S52

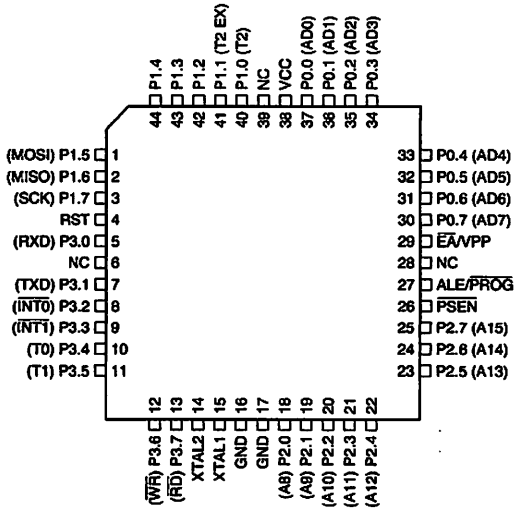
1919C–MICRO–3/05

Pin Configurations

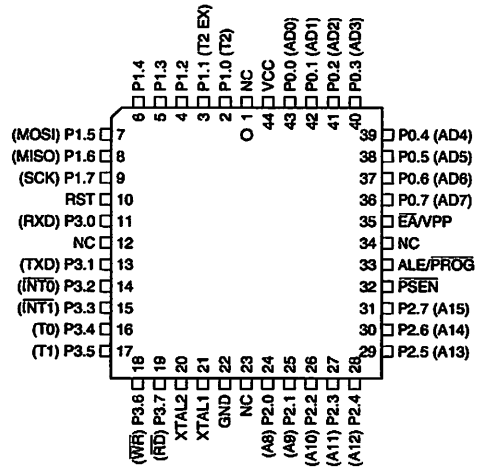
40-lead PDIP



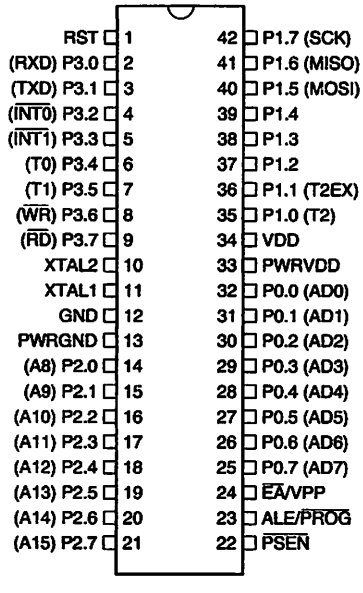
44-lead TQFP



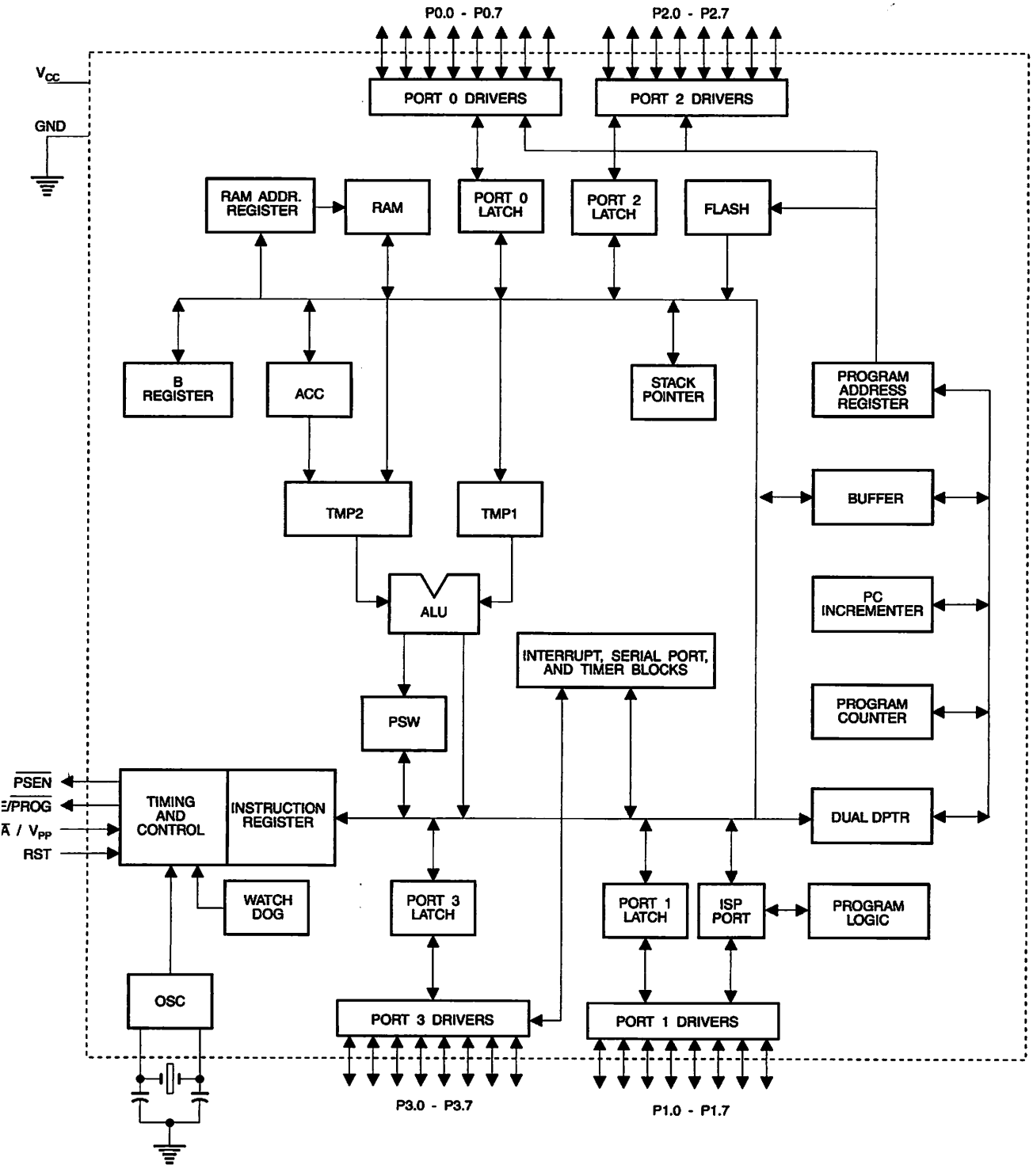
2.3 44-lead PLCC



2.4 42-lead PDIP



Block Diagram



Pin Description

VCC

Supply voltage.

GND

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.

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

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

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)
P1.5	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 AT89S52, as shown in the following table.

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

RST

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

ALE/$\overline{PROG}$

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 ($\overline{PROG}$) 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 AT89S52 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

0 $\overline{\text{EA}}$/VPP

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

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

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

1 XTAL1

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

2 XTAL2

Output from the inverting oscillator amplifier.

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 5-1.

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

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

Timer 2 Registers: Control and status bits are contained in registers T2CON (shown in Table 5-2) and T2MOD (shown in Table 10-2) for Timer 2. The register pair (RCAP2H, RCAP2L) are the Capture/Reload registers for Timer 2 in 16-bit capture mode or 16-bit auto-reload mode.

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

e 5-1. AT89S52 SFR Map and Reset Values

F8H								0FFH
F0H	B 00000000							0F7H
E8H								0EFH
E0H	ACC 00000000							0E7H
D8H								0DFH
D0H	PSW 00000000							0D7H
C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000		0CFH
C0H								0C7H
B8H	IP XX000000							0BFH
B0H	P3 11111111							0B7H
A8H	IE 0X000000							0AFH
A0H	P2 11111111		AUXR1 XXXXXXX0				WDTRST XXXXXXXX		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

le 5-2. T2CON – Timer/Counter 2 Control Register

T2CON Address = 0C8H — Reset Value = 0000 0000B

Bit Addressable

Bit	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
	7	6	5	4	3	2	1	0

mbol	Function
2	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
F2	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).
LK	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
LK	Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EN2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
2	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
$\overline{T2}$	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
/$\overline{RL2}$	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

le 5-3. AUXR: Auxiliary Register

XR — Address = 8EH — Reset Value = XXX00XX0B

Not Bit Addressable

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

Reserved for future expansion

SALE — 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

SRTO — Disable/Enable Reset out

DISRTO	
0	Reset pin is driven High after WDT times out
1	Reset pin is input only

DIDLE — Disable/Enable WDT in IDLE mode

WDIDLE	
0	WDT continues to count in IDLE mode
1	WDT halts counting in IDLE mode

l Data Pointer Registers: To facilitate accessing both internal and external data memory, two banks of 16-bit Data ıter Registers are provided: DP0 at SFR address locations 82H-83H and DP1 at 84H-85H. Bit DPS = 0 in SFR AUXR1 cts DP0 and DPS = 1 selects DP1. The user should **ALWAYS** initialize the DPS bit to the appropriate value before ɔssing the respective Data Pointer Register.

ver 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 lt can be set and rest under software control and is not affected by reset.

le 5-4. AUXR1: Auxiliary Register 1

XR1 — Address = A2H — Reset Value = XXXXXXX0B

Not Bit Addressable

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

Reserved for future expansion

S — 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 AT89S52, if $\overline{EA}$ is connected to V_{CC}, program fetches to addresses 0000H through 1FFFH are directed to internal memory and fetches to addresses 2000H through FFFFH are to external memory.

Data Memory

The AT89S52 implements 256 bytes of on-chip RAM. The upper 128 bytes occupy a parallel address space to the Special Function Registers. This means that the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space.

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

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

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

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

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

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

Watchdog Timer (One-time Enabled with Reset-out)

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upsets. The WDT consists of a 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 AT89S52 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 AT89S52 while in IDLE mode, the user should always set up a timer that will periodically exit IDLE, service the WDT, and reenter IDLE mode.

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

UART

The UART in the AT89S52 operates the same way as the UART in the AT89C51 and AT89C52. For further information on the UART operation, please click on the document link below:

http://www.atmel.com/dyn/resources/prod_documents/DOC4316.PDF

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S52 operate the same way as Timer 0 and Timer 1 in the AT89C51 and AT89C52. For further information on the timers' operation, please click on the document link below:

http://www.atmel.com/dyn/resources/prod_documents/DOC4316.PDF

Timer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 5-2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 10-1. Timer 2 consists of two 8-bit registers, TH2 and TL2. In the Timer function, the TL2 register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

Table 10-1. Timer 2 Operating Modes

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

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure that a given level is sampled at least once before it changes, the level should be held for at least one full machine cycle.

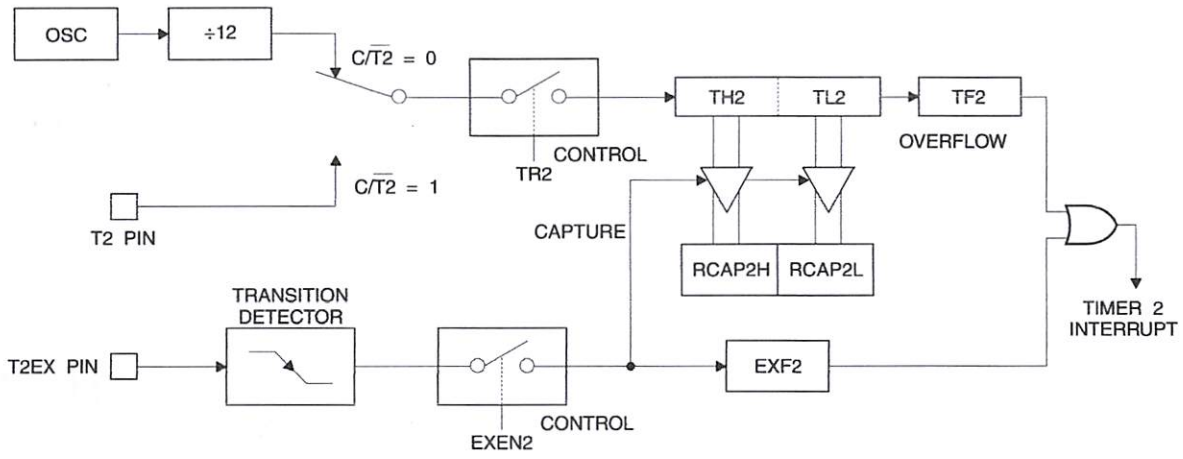
Capture Mode

In the capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 performs the same operation, but a 1-to-0 transition at external input T2EX also causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt. The capture mode is illustrated in Figure 10-1.

Auto-reload (Up or Down Counter)

Timer 2 can be programmed to count up or down when configured in its 16-bit auto-reload mode. This feature is invoked by the DCEN (Down Counter Enable) bit located in the SFR T2MOD (see Table 10-2). Upon reset, the DCEN bit is set to 0 so that timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down, depending on the value of the T2EX pin.

re 10-1. Timer in Capture Mode



e 10-2. T2MOD – Timer 2 Mode Control Register

T2MOD Address = 0C9H — Reset Value = XXXX XX00B

Not Bit Addressable

	–	–	–	–	–	–	T2OE	DCEN
Bit	7	6	5	4	3	2	1	0

mbol	Function
	Not implemented, reserved for future
DE	Timer 2 Output Enable bit
EN	When set, this bit allows Timer 2 to be configured as an up/down counter

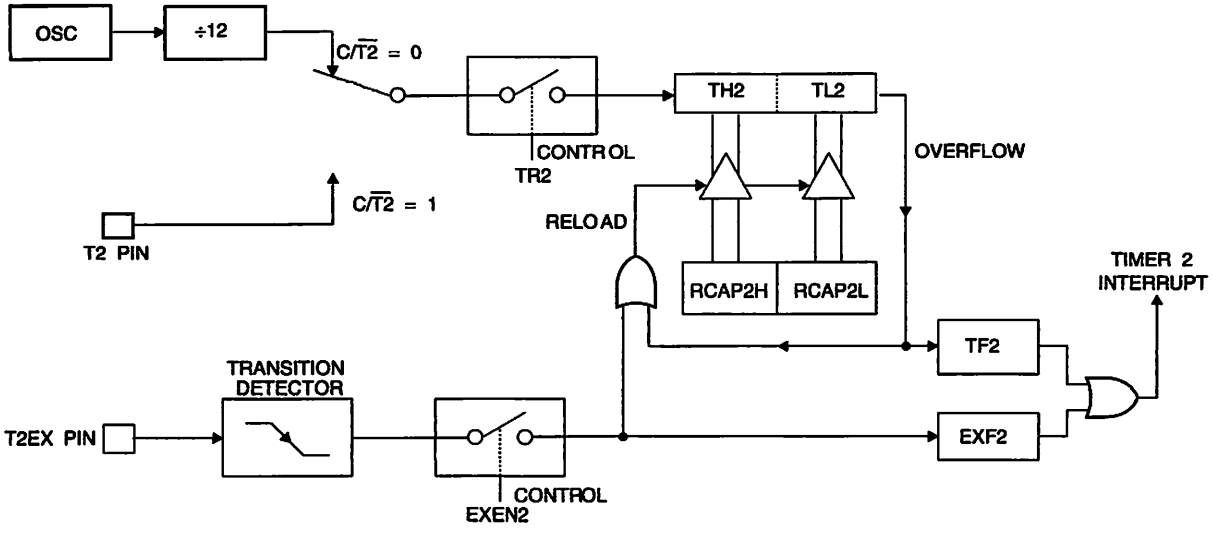
Figure 10-2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in Timer in Capture ModeRCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 10-2. In this mode, the T2EX pin controls the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16-bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

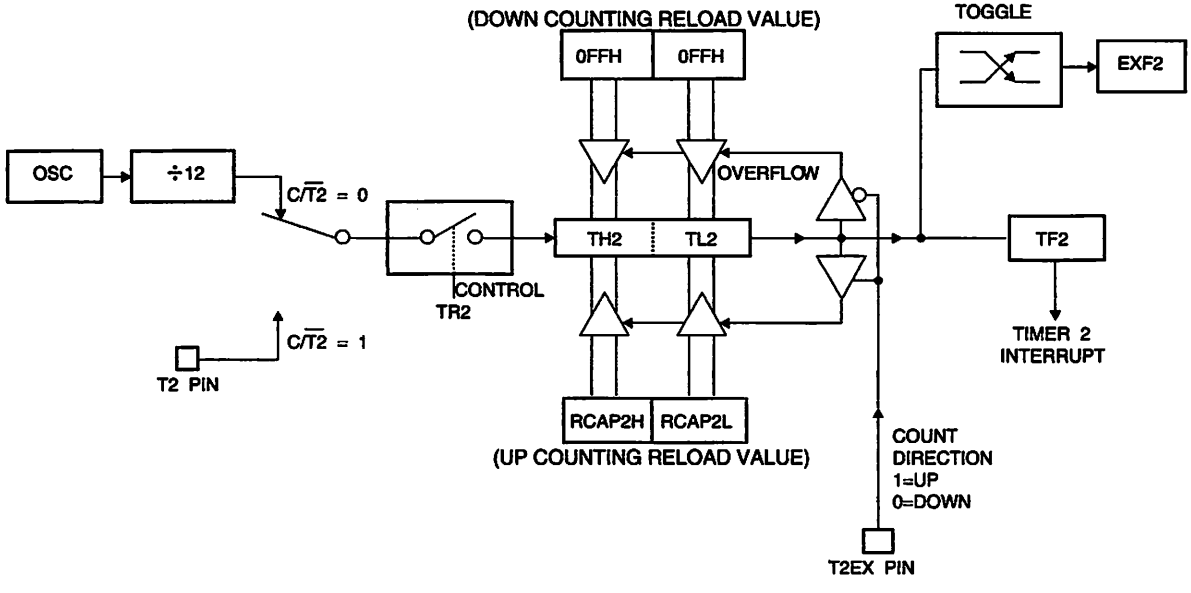
A logic 0 at T2EX makes Timer 2 count down. The timer underflows when TH2 and TL2 equal the values stored in RCAP2H and RCAP2L. The underflow sets the TF2 bit and causes 0FFFFH to be reloaded into the timer registers.

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

re 10-2. Timer 2 Auto Reload Mode (DCEN = 0)



re 10-3. Timer 2 Auto Reload Mode (DCEN = 1)



Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (Table 5-2). Note that the baud rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for the other function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 11-1.

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

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

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

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

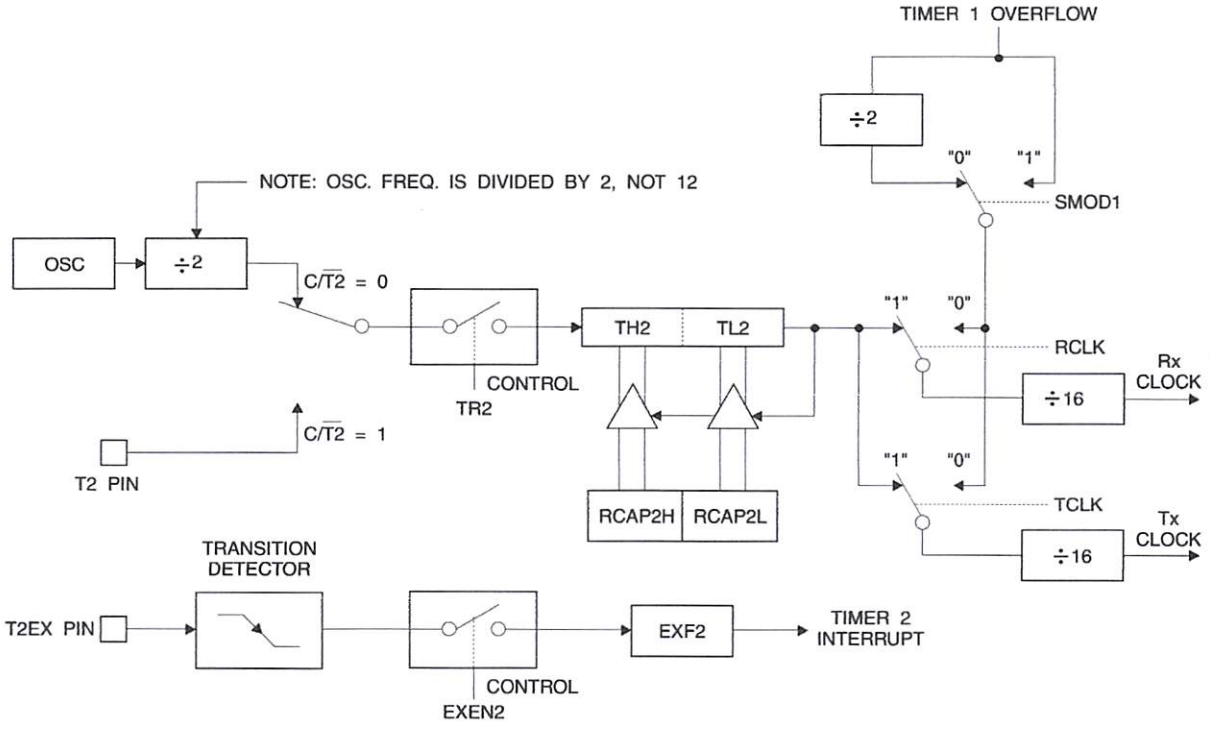
$$\frac{\text{Modes 1 and 3}}{\text{Baud Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536\text{-RCAP2H,RCAP2L})]}$$

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

Timer 2 as a baud rate generator is shown in Figure 11-1. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus, when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

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

ıre 11-1. Timer 2 in Baud Rate Generator Mode



Programmable Clock Out

A 50% duty cycle clock can be programmed to come out on P1.0, as shown in Figure 12-1. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed to input the external clock for Timer/Counter 2 or to output a 50% duty cycle clock ranging from 61 Hz to 4 MHz (for a 16-MHz operating frequency).

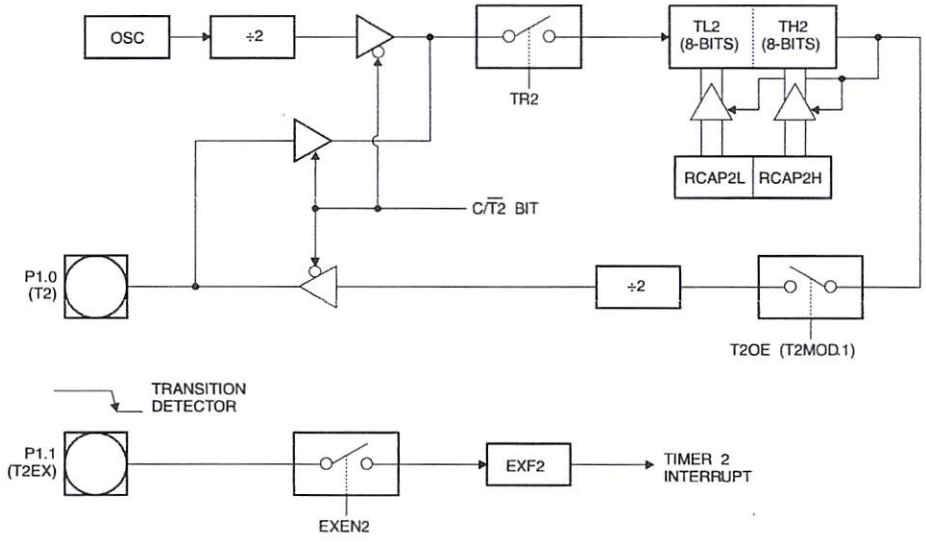
To configure the Timer/Counter 2 as a clock generator, bit C/$\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

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

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

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

ıre 12-1. Timer 2 in Clock-Out Mode



Interrupts

The AT89S52 has a total of six interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (Timers 0, 1, and 2), and the serial port interrupt. These interrupts are all shown in Figure 13-1.

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 13-1 shows that bit position IE.6 is unimplemented. User software should not write a 1 to this bit position, since it may be used in future AT89 products.

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

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

e 13-1. Interrupt Enable (IE) Register

(MSB)							(LSB)
EA	–	ET2	ES	ET1	EX1	ET0	EX0

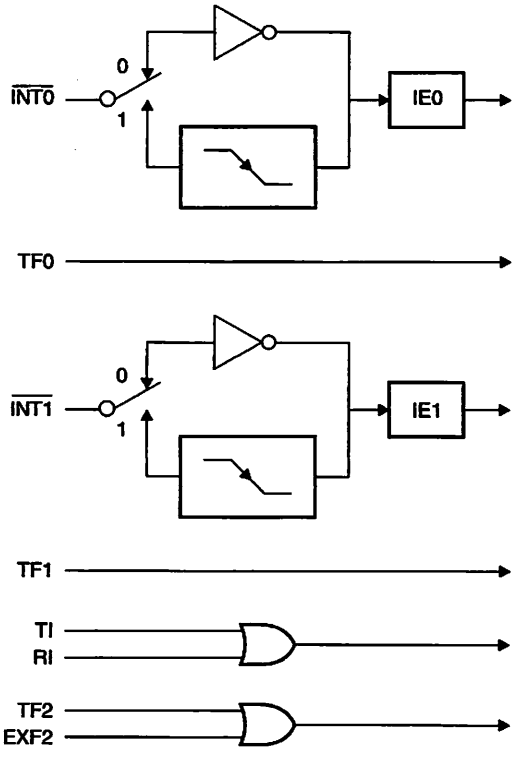
Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

nbol	Position	Function
	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.
2	IE.5	Timer 2 interrupt enable bit.
	IE.4	Serial Port interrupt enable bit.
1	IE.3	Timer 1 interrupt enable bit.
1	IE.2	External interrupt 1 enable bit.
0	IE.1	Timer 0 interrupt enable bit.
0	IE.0	External interrupt 0 enable bit.

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

re 13-1. 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 16-1. 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 16-2. 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.

Idle Mode

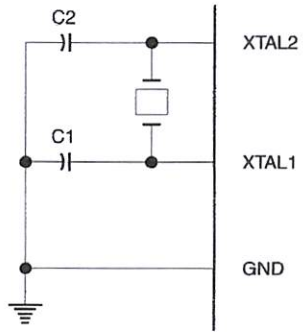
In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware 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 an enabled external interrupt. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Figure 16-1. Oscillator Connections



Note: 1. C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Figure 16-2. External Clock Drive Configuration

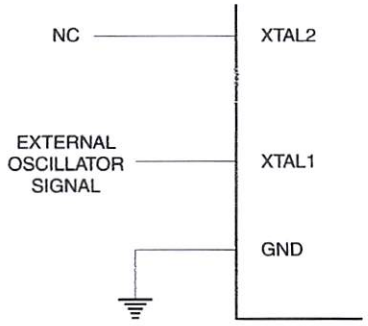


Table 16-1. Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	$\overline{PSEN}$	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 AT89S52 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in Table 17-1.

Table 17-1. Lock Bit Protection Modes

Program Lock Bits				
	LB1	LB2	LB3	Protection Type
1	U	U	U	No program lock features
2	P	U	U	MOVC 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 AT89S52 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 AT89S52 code memory array is programmed byte-by-byte.

Programming Algorithm: Before programming the AT89S52, the address, data, and control signals should be set up according to the "Flash Programming Modes" (Table 22-1) and Figure 22-1 and Figure 22-2. To program the AT89S52, 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 µs. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

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

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

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

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

(000H) = 1EH indicates manufactured by Atmel
(100H) = 52H indicates AT89S52
(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/$\overline{PROG}$ low for a duration of 200 ns - 500 ns.

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

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

Programming the Flash – Serial Mode

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

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

Either an external system clock can be supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK) 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 AT89S52 in the serial programming mode, the following sequence is recommended:

1. Power-up sequence:
   a. Apply power between VCC and GND pins.
   b. 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 which returns the content at the selected address at serial output MISO/P1.6.
5. At the end of a programming session, RST can be set low to commence normal device operation.

Power-off sequence (if needed):

1. Set XTAL1 to "L" (if a crystal is not used).
2. Set RST to "L".
3. Turn V_{CC} power off.

Data Polling: The $\overline{\text{Data}}$ 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 24-1.

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.

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

le 22-1. Flash Programming Modes

de	V_{CC}	RST	$\overline{PSEN}$	ALE/ $\overline{PROG}$	$\overline{EA}$/ V_{PP}	P2.6	P2.7	P3.3	P3.6	P3.7	P0.7-0 Data	P2.4-0 Address	P1.7-0 Address
te Code Data	5V	H	L	(2)	12V	L	H	H	H	H	D_{IN}	A12-8	A7-0
ad Code Data	5V	H	L	H	H	L	L	L	H	H	D_{OUT}	A12-8	A7-0
te Lock Bit 1	5V	H	L	(3)	12V	H	H	H	H	H	X	X	X
te Lock Bit 2	5V	H	L	(3)	12V	H	H	H	L	L	X	X	X
te Lock Bit 3	5V	H	L	(3)	12V	H	L	H	H	L	X	X	X
ad Lock Bits , 3	5V	H	L	H	H	H	H	L	H	L	P0.2, P0.3, P0.4	X	X
p Erase	5V	H	L	(1)	12V	H	L	H	L	L	X	X	X
ad Atmel ID	5V	H	L	H	H	L	L	L	L	L	1EH	X 0000	00H
ad Device ID	5V	H	L	H	H	L	L	L	L	L	52H	X 0001	00H
ad Device ID	5V	H	L	H	H	L	L	L	L	L	06H	X 0010	00H

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

Figure 22-1. Programming the Flash Memory (Parallel Mode)

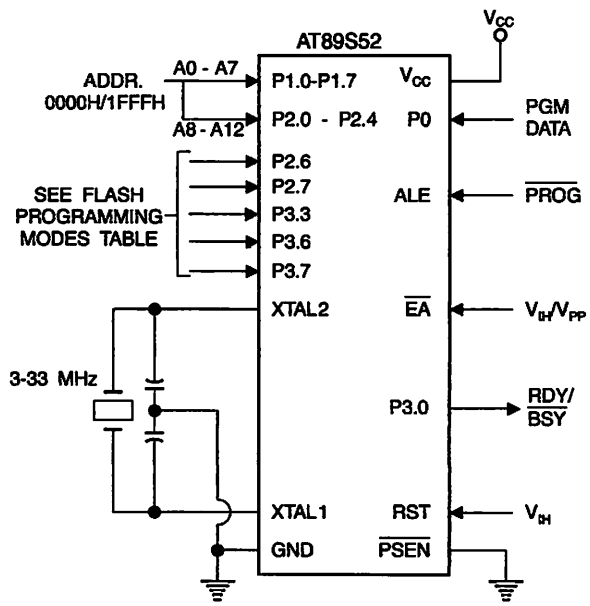
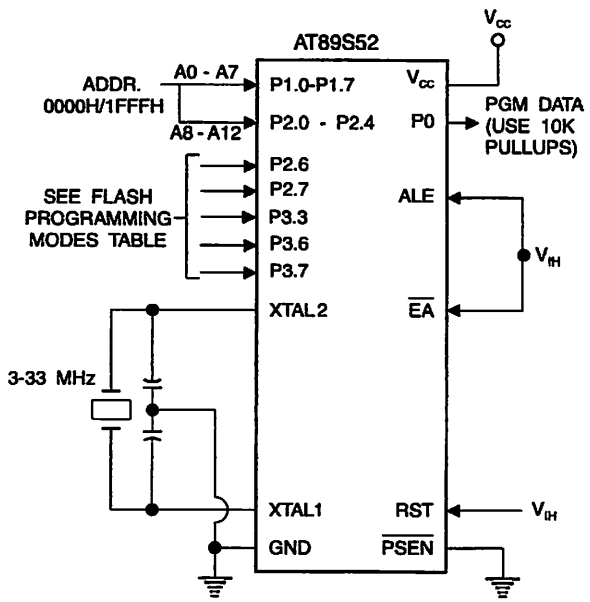


Figure 22-2. Verifying the Flash Memory (Parallel Mode)



Flash Programming and Verification Characteristics (Parallel Mode)

20°C to 30°C, V_{CC} = 4.5 to 5.5V

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

re 23-1. Flash Programming and Verification Waveforms – Parallel Mode

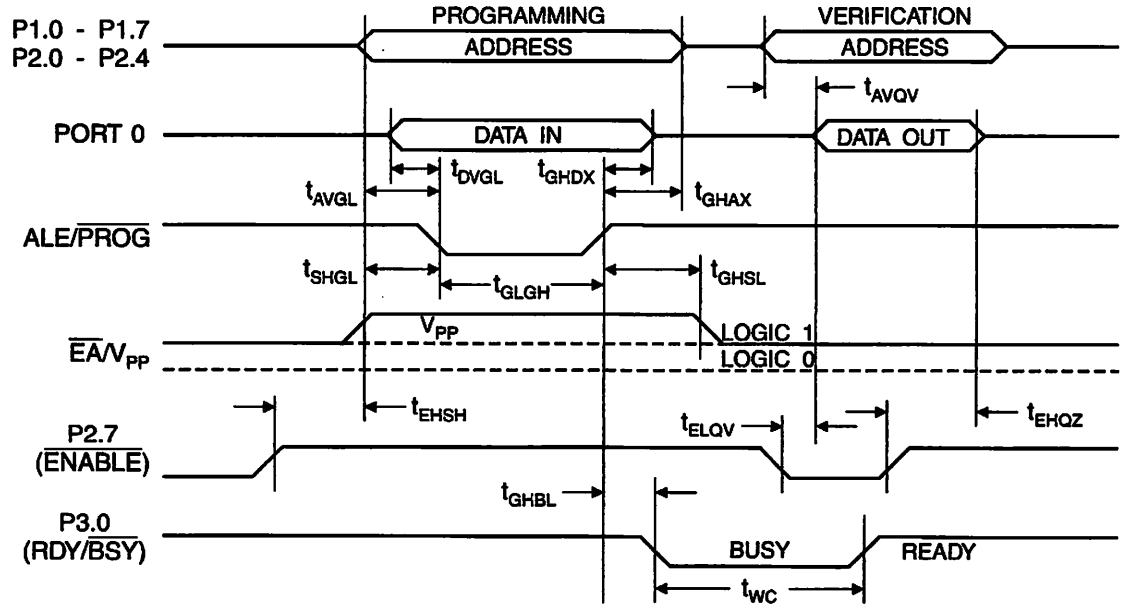
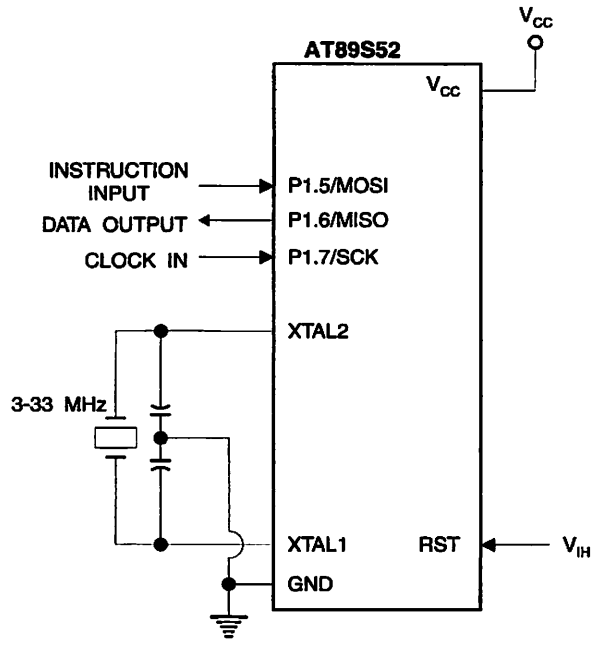
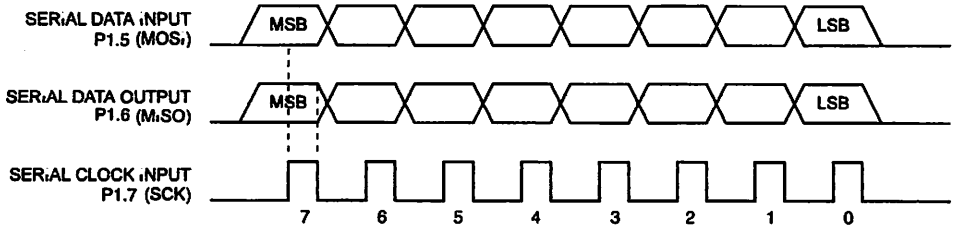


Figure 23-2. Flash Memory Serial Downloading



Flash Programming and Verification Waveforms – Serial Mode

Figure 24-1. Serial Programming Waveforms



e 24-1. Serial Programming Instruction Set

truction	Instruction Format Byte 1	Byte 2	Byte 3	Byte 4	Operation
gramming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx 0110 1001 (Output on MISO)	Enable Serial Programming while RST is high
p Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash memory array
ıd Program Memory te Mode)	0010 0000	xxx A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Read data from Program memory in the byte mode
te Program Memory te Mode)	0100 0000	xxx A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Write data to Program memory in the byte mode
te Lock Bits[1]	1010 1100	1110 00 B1 B2	xxxx xxxx	xxxx xxxx	Write Lock bits. See Note (1).
ıd Lock Bits	0010 0100	xxxx xxxx	xxxx xxxx	xxx LB3 LB2 LB1 xx	Read back current status of the lock bits (a programmed lock bit reads back as a "1")
ıd Signature Bytes	0010 1000	xxx A12 A11 A10 A9 A8	A7 xxx xxx0	Signature Byte	Read Signature Byte
ıd Program Memory ge Mode)	0011 0000	xxx A12 A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Read data from Program memory in the Page Mode (256 bytes)
te Program Memory ge Mode)	0101 0000	xxx A12 A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Write data to Program memory in the Page Mode (256 bytes)

1. 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

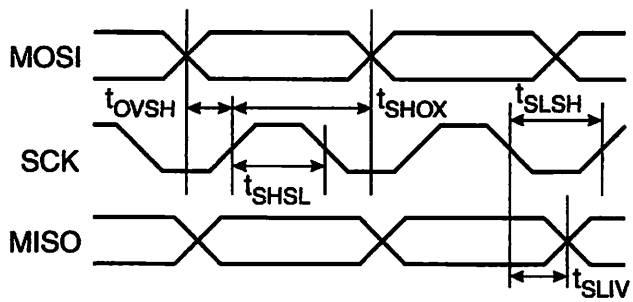
   } Each of the lock bit modes needs to be activated sequentially before Mode 4 can be executed.

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

ıre 25-1. Serial Programming Timing



e 25-1. Serial Programming Characteristics, T_A = -40°C to 85°C, V_{CC} = 4.0 - 5.5V (Unless Otherwise Noted)

nbol	Parameter	Min	Typ	Max	Units
LCL	Oscillator Frequency	3		33	MHz
L	Oscillator Period	30			ns
L	SCK Pulse Width High	8 t_{CLCL}			ns
H	SCK Pulse Width Low	8 t_{CLCL}			ns
H	MOSI Setup to SCK High	t_{CLCL}			ns
X	MOSI Hold after SCK High	2 t_{CLCL}			ns
	SCK Low to MISO Valid	10	16	32	ns
SE	Chip Erase Instruction Cycle Time			500	ms
C	Serial Byte Write Cycle Time			64 t_{CLCL} + 400	µs

Absolute Maximum Ratings*

erating Temperature	-55°C to +125°C
rage Temperature	-65°C to +150°C
tage on Any Pin n Respect to Ground	-1.0V to +7.0V
ximum Operating Voltage	6.6V
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

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

nbol	Parameter	Condition	Min	Max	Units
	Input Low Voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
	Input Low Voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
	Input High Voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
	Input High Voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
	Output Low Voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
	Output Low Voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
	Output High Voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
	Output High Voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-300	µA
	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
ST	Reset Pulldown Resistor		50	300	KΩ
	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode[1]	V_{CC} = 5.5V		50	µA

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

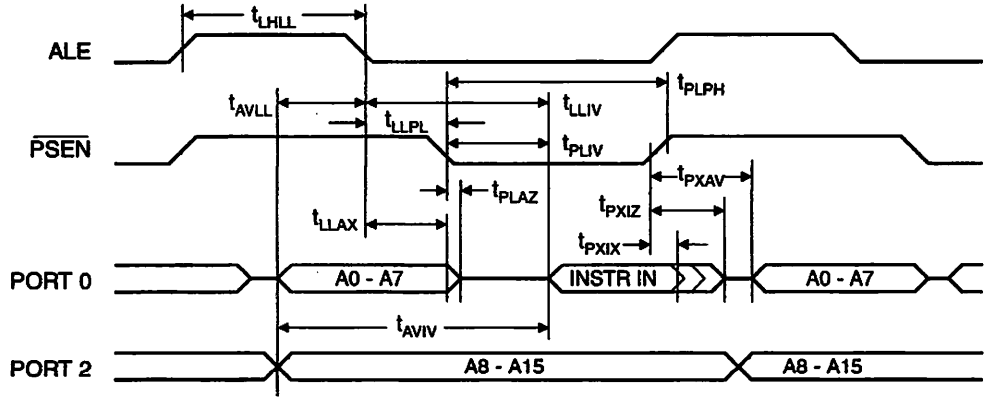
AC Characteristics

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

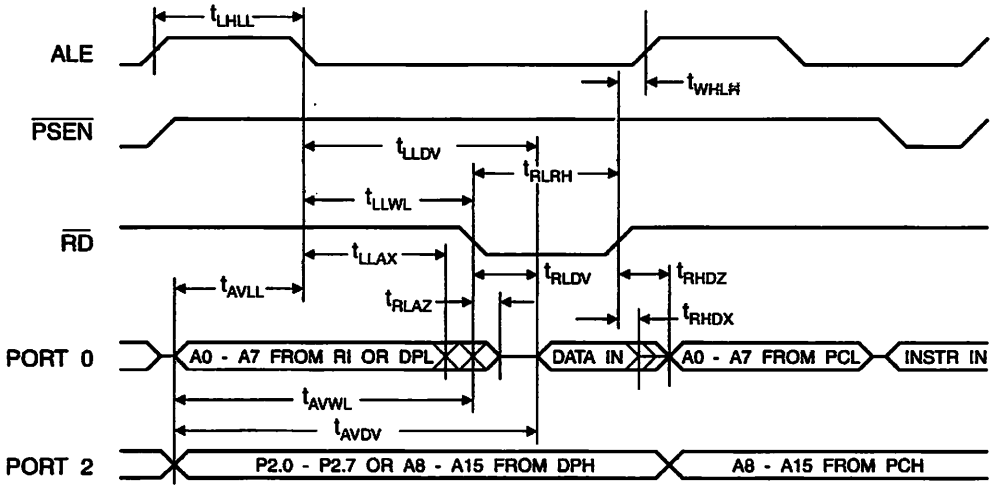
External Program and Data Memory Characteristics

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

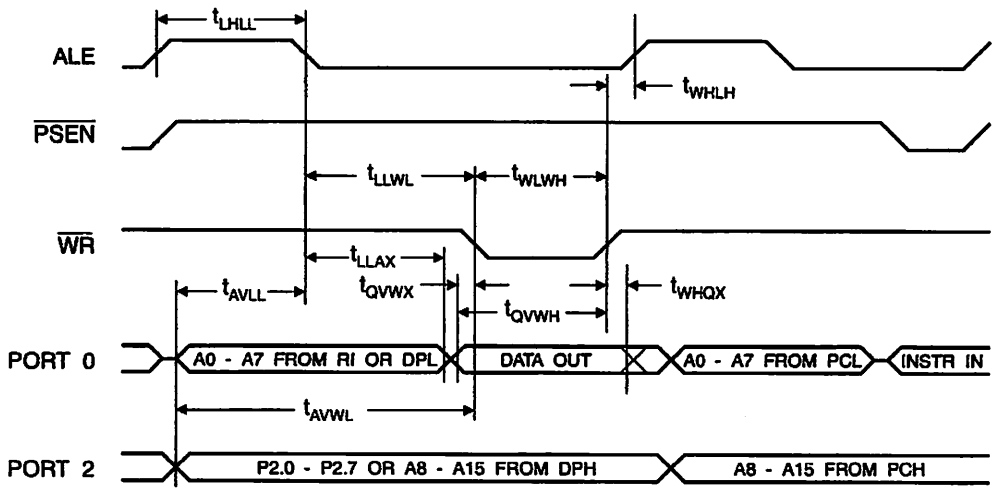
External Program Memory Read Cycle



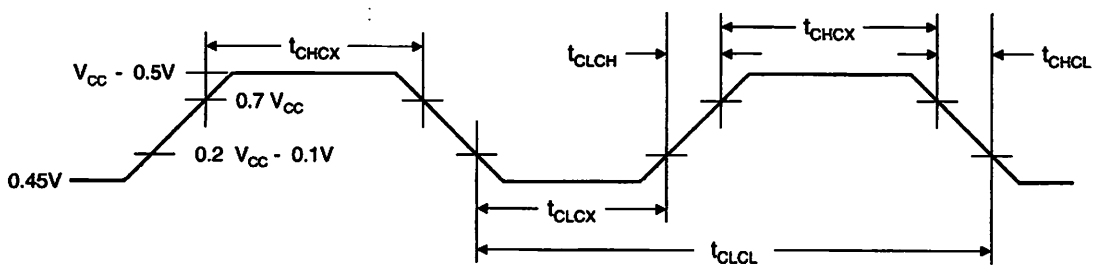
External Data Memory Read Cycle



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

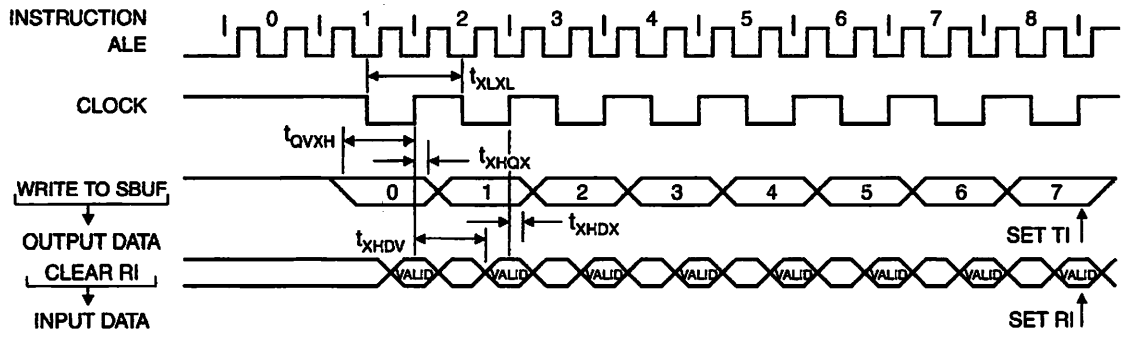
ıbol	Parameter	Min	Max	Units
LCL	Oscillator Frequency	0	33	MHz
L	Clock Period	30		ns
X	High Time	12		ns
X	Low Time	12		ns
H	Rise Time		5	ns
L	Fall Time		5	ns

Serial Port Timing: Shift Register Mode Test Conditions

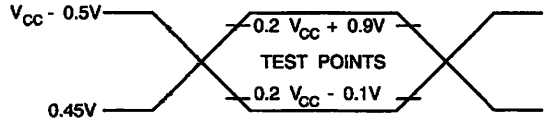
values in this table are valid for V_{CC} = 4.0V to 5.5V and Load Capacitance = 80 pF.

nbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
L	Serial Port Clock Cycle Time	1.0		12 t_{CLCL}		µs
QV	Output Data Setup to Clock Rising Edge	700		10 t_{CLCL}-133		ns
QX	Output Data Hold After Clock Rising Edge	50		2 t_{CLCL}-80		ns
DX	Input Data Hold After Clock Rising Edge	0		0		ns
DV	Clock Rising Edge to Input Data Valid		700		10 t_{CLCL}-133	ns

Shift Register Mode Timing Waveforms

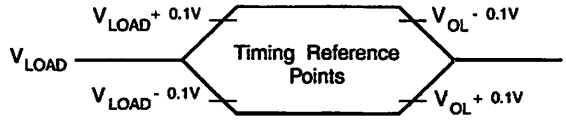


AC Testing Input/Output Waveforms(1)



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

Float Waveforms(1)



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

Standard Package

peed MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 5.5V	AT89S52-24AC	44A	Commercial (0°C to 70°C)
		AT89S52-24JC	44J	
		AT89S52-24PC	40P6	
		AT89S52-24SC	42PS6	
		AT89S52-24AI	44A	Industrial (-40°C to 85°C)
		AT89S52-24JI	44J	
		AT89S52-24PI	40P6	
		AT89S52-24SI	42PS6	
33	4.5V to 5.5V	AT89S52-33AC	44A	Commercial (0°C to 70°C)
		AT89S52-33JC	44J	
		AT89S52-33PC	40P6	
		AT89S52-33SC	42PS6	

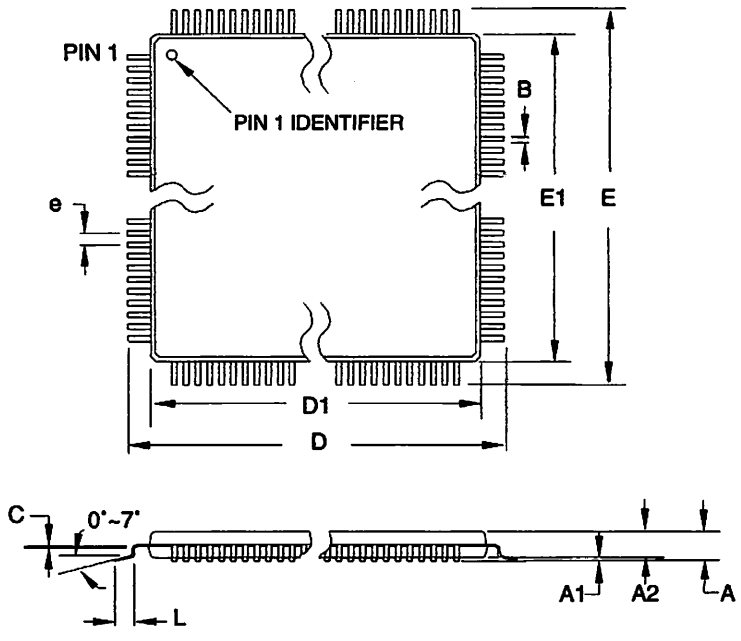
Green Package Option (Pb/Halide-free)

peed MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 5.5V	AT89S52-24AU	44A	Industrial (-40°C to 85°C)
		AT89S52-24JU	44J	
		AT89S52-24PU	40P6	

	Package Type
	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
	44-lead, Plastic J-leaded Chip Carrier (PLCC)
6	40-pin, 0.600" Wide, Plastic Dual Inline Package (PDIP)
S6	42-pin, 0.600" Wide, Plastic Dual Inline Package (PDIP)

Packaging Information

44A – TQFP



COMMON DIMENSIONS
(Unit of Measure = mm)

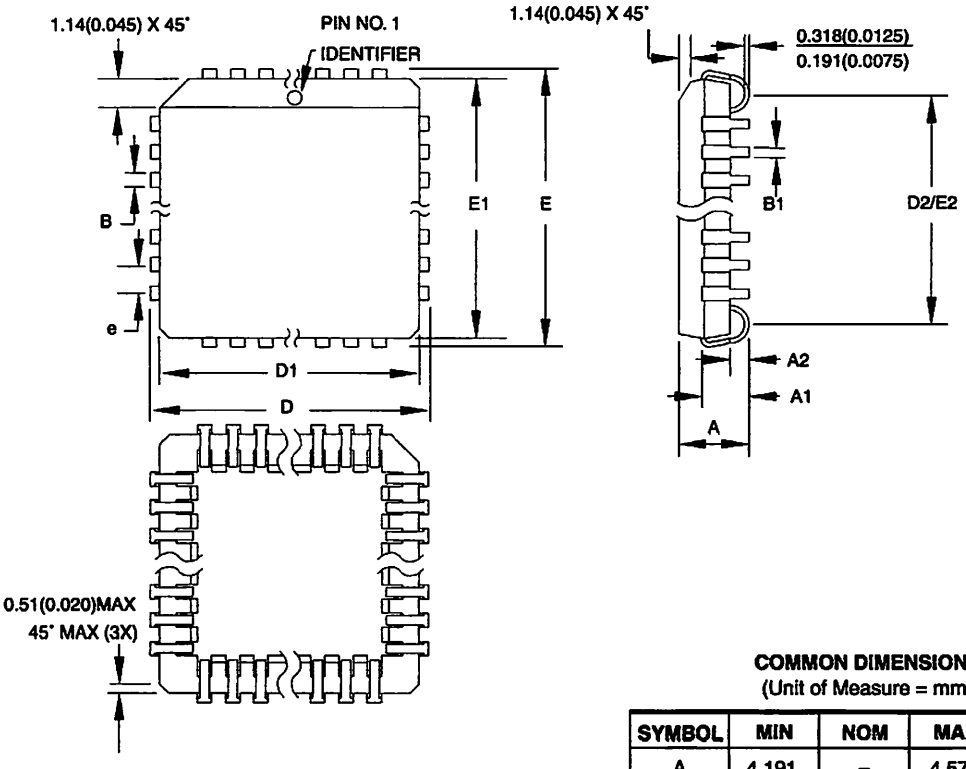
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

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

10/5/2001

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**44A**, 44-lead, 10 x 10 mm Body Size, 1.0 mm Body Thickness, 0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)	44A	B

44J – PLCC



COMMON DIMENSIONS
(Unit of Measure = mm)

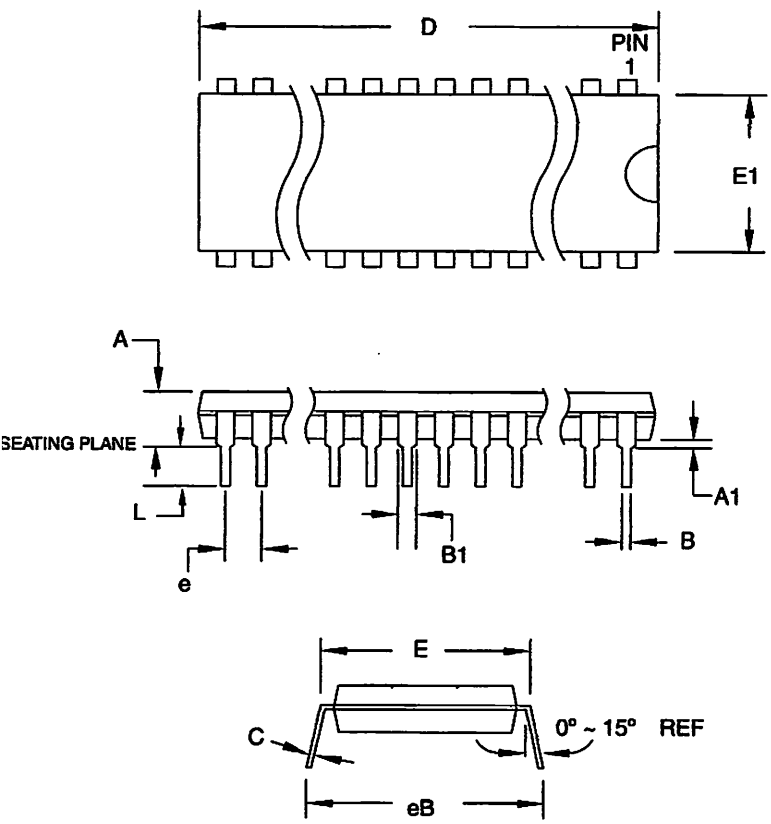
SYMBOL	MIN	NOM	MAX	NOTE
A	4.191	–	4.572	
A1	2.286	–	3.048	
A2	0.508	–	–	
D	17.399	–	17.653	
D1	16.510	–	16.662	Note 2
E	17.399	–	17.653	
E1	16.510	–	16.662	Note 2
D2/E2	14.986	–	16.002	
B	0.660	–	0.813	
B1	0.330	–	0.533	
e	1.270 TYP			

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

10/04/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)	44J	B

3 40P6 – PDIP



COMMON DIMENSIONS
(Unit of Measure = mm)

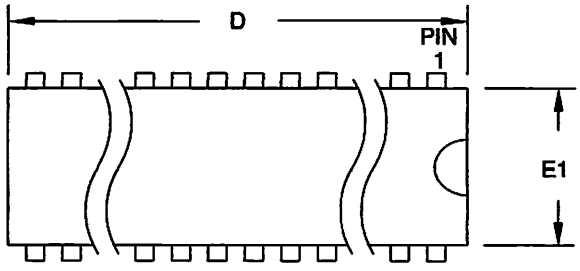
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	4.826	
A1	0.381	–	–	
D	52.070	–	52.578	Note 2
E	15.240	–	15.875	
E1	13.462	–	13.970	Note 2
B	0.356	–	0.559	
B1	1.041	–	1.651	
L	3.048	–	3.556	
C	0.203	–	0.381	
eB	15.494	–	17.526	
e	2.540 TYP			

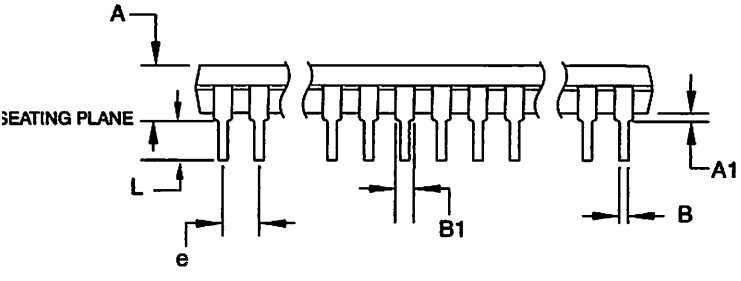
Notes:
1. This package conforms to JEDEC reference MS-011, Variation AC.
2. Dimensions D and E1 do not include mold Flash or Protrusion. Mold Flash or Protrusion shall not exceed 0.25 mm (0.010").

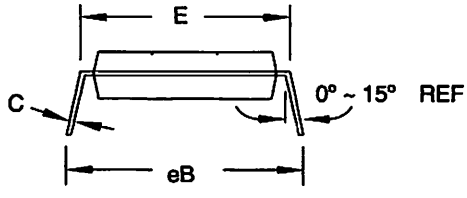
09/28/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**40P6**, 40-lead (0.600"/15.24 mm Wide) Plastic Dual Inline Package (PDIP)	40P6	B

42PS6 – PDIP







COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	4.83	
A1	0.51	–	–	
D	36.70	–	36.96	Note 2
E	15.24	–	15.88	
E1	13.46	–	13.97	Note 2
B	0.38	–	0.56	
B1	0.76	–	1.27	
L	3.05	–	3.43	
C	0.20	–	0.30	
eB	–	–	18.55	
e	1.78 TYP			

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

11/6/03

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**42PS6**, 42-lead (0.600"/15.24 mm Wide) Plastic Dual Inline Package (PDIP)	42PS6	A

nel Corporation

325 Orchard Parkway
an Jose, CA 95131, USA
el: 1(408) 441-0311
ax: 1(408) 487-2600

gional Headquarters

ope
tmel Sarl
oute des Arsenaux 41
ase Postale 80
H-1705 Fribourg
witzerland
el: (41) 26-426-5555
ax: (41) 26-426-5500

a
oom 1219
hinachem Golden Plaza
7 Mody Road Tsimshatsui
ast Kowloon
ong Kong
el: (852) 2721-9778
ax: (852) 2722-1369

an
F, Tonetsu Shinkawa Bldg.
-24-8 Shinkawa
huo-ku, Tokyo 104-0033
apan
el: (81) 3-3523-3551
ax: (81) 3-3523-7581

Atmel Operations

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

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

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

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

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

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

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

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

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

Printed on recycled paper.

1919C–MICRO–3/05 xM

DATA SHEET

Infrared Photodiode

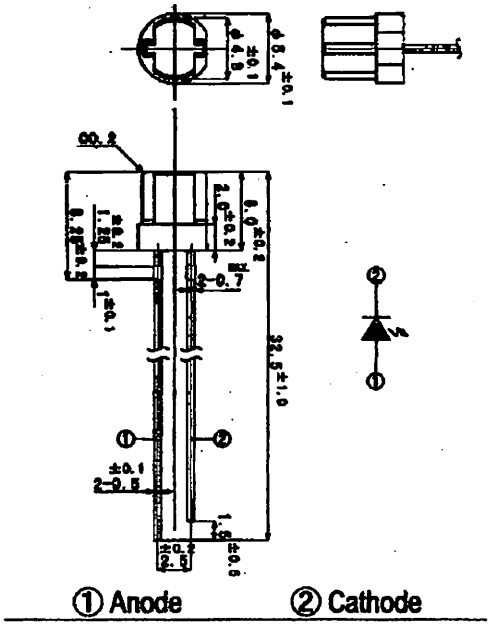
Infrared Photodiode EPD-1300-5-1

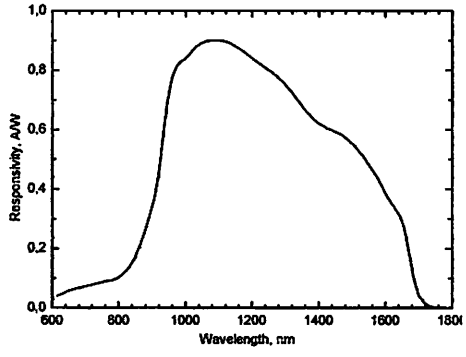
Preliminary data

Spectral range	Type	Technology	Case
Infrared	EPD-1300-5-1	InGaAs/InP	5 mm with flat top

Description
Photodiodes with enhanced sensitivity in near infrared range

Applications
Optical communications, remote control, safety equipment





Maximum Ratings

Parameter	Value	Unit
Storage Temperature	-20 to +80	°C
Operating Temperature	-30 to +100	°C
Soldering Temperature	260 for 3 sec.	°C

Optical and Electrical Characteristics
T_{amb} = 25°C, unless otherwise specified

Parameter	Test conditions	Symbol	Min	Typ	Max	Unit
Spectral range at 0.1 S_λ		λ_{Smin}-λ_{Smax}		800...1700		nm

ROITHNER LASERTECHNIK
Schoenbrunner Str. 7, A-1040 Vienna, Austria, Tel: +43-1-586 52 43-0, Fax: +43-1-586 52 43 44, office@roithner-laser.com

Spectral range at 0.5 S_λ		λ_{Smin}- λ_{Smax}		920...1600		nm
Acceptance angle at 0.5 S_λ				110		deg.
Responsivity	λ = 1300 nm	S_λ		0.75		A/W
Dark current	V_R = 5 V, E_e=0	I_D		30	120	nA
Reverse voltage	I_R = 10 µA	V_R		35		V
Junction capacitance	V_R = 5 V, E_e=0	C		40		pF

ROITHNER LASERTECHNIK
Schoenbrunner Str. 7, A-1040 Vienna, Austria, Tel: +43-1-586 52 43-0, Fax: +43-1-586 52 43 44, office@roithner-laser.com

INTERFACING PC STANDARD PARALLEL PORT TO DST

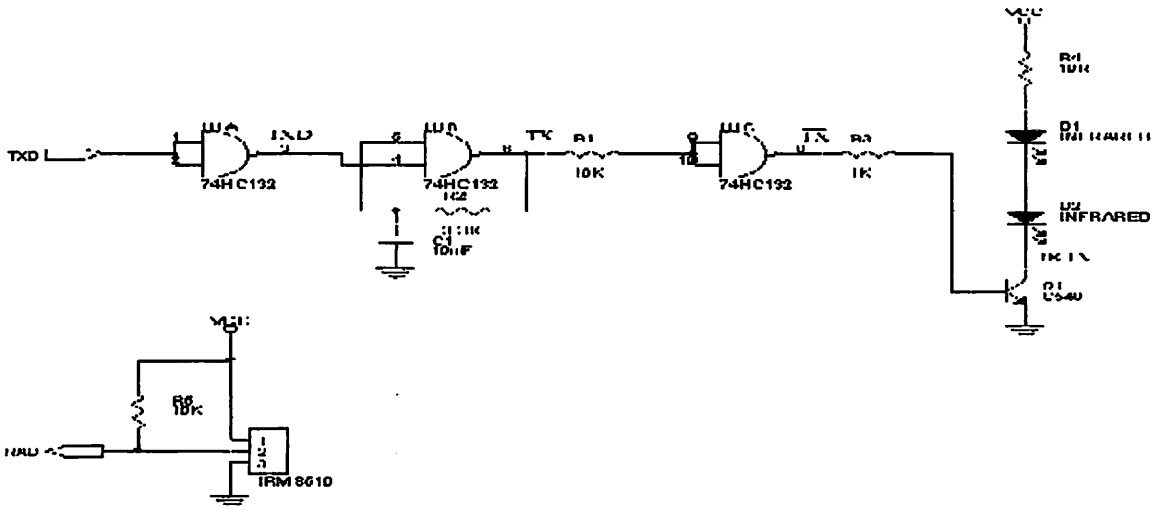
AN-008IR-8510 INFRARED TRANSCEIVER

Infrared Transceiveradalah sebuah sistem yang terdiri dari Infrared Transmitterdan Infrared Receiver di mana sistem ini berfungsi untuk proses komunikasi data

Aplikasi

- Wireless Data Communication
- Alarm System
- Universal Remote Control

Deskripsi



Skema Infrared Transceiver

Untuk memperoleh jarak yang cukup jauh, Diode Infrared memerlukan sinyal dengan frekwensi 30 hingga 50 KHz. Berbeda dengan Diode LED yang hanya memerlukan level tegangan **DC** saja untuk mengaktifkan LED, Diode Infraredmemerlukan sinyal **AC** dengan frekwensi **30** hingga **50 KHz** untuk mengaktifkannya. Cahaya infrared tersebut tidak dapat ditangkap oleh mata manusia, sehingga diperlukan **phototransistor**untuk mendeteksinya.

Phototransistoradalah merupakan sebuah transistor yang akan **saturasi** pada saat menerima sinar infrared dan **cut off** pada saat tidak ada sinar infrared. **IR Module** adalah sebuah rangkaian yang terdiri dari sebuah phototransistor dan filter yang terbentuk dalam satu modul di mana collector dari phototransistoradalah merupakan output dari modul ini. Pada saat phototransistor cut offmaka tidak terjadi aliran arus dari **collector** menuju ke **emitter** sehingga collector yang merupakan output dari IR Module akan berkondisi high. Apabila phototransistor saturasi maka arus mengalir dari collector ke emitter dan output dari IR Module akan berkondisi low.

Transmisi data dilakukan dengan menggunakan prinsip aktif dan non aktifnya LED Infraredsebagai kondisi logic 0 dan logic 1. Seperti telah dijelaskan sebelumnya bahwa untuk mengaktifkan LED Infrared diperlukan frekwensi sebesar 30 hingga 40 KHz, maka dalam hal ini logic 0 berarti sinyal berfrekwensi 30 KHz mengalir ke LED Infrareddan logic 1 berarti tidak ada sinyal yang mengalir ke LED Infrared,hal ini seperti yang tampak pada hubungan antara TXD dan TX pada Timing Diagram berikut.

Untuk menghasilkan sinyal seperti yang tampak pada TX dibutuhkan sebuah rangkaian **modulator** yang terdiri dari sebuah gerbang dan rangkaian **R-C** sebagai **oscillator**. Gerbang tersebut menggunakan IC 74HC132 di mana pada saat pin TXD berkondisi high dan TXD berkondisi low maka output dari IC ini sesuai dengan tabel kebenaran yang ada pada data sheet adalah high. Namun bila sebaliknya TXD berkondisi high maka sesaat output dari IC ini berubah ke low sehingga capacitor C1 akan membuang muatannya melalui R1. Bila tegangan C1 terbuang hingga di bawah **tegangan ambang** 74HC132 maka input pin nomor 4 dari IC ini akan dianggap berkondisi low sehingga outputnya berubah menjadi high.

C1 kembali terisi melalui R1 hingga tegangan pada capacitor ini melebihi tegangan ambang dan input pin nomor 4 dianggap berkondisi high. Bila pada saat itu TXD masih berkondisi high maka output dari gerbang ini yaitu pin nomor 6 akan berkondisi low dan C1 kembali membuang, demikian seterusnya C1 akan terisi hingga di atas tegangan ambang 74HC132 (2,5 V) dan terbuang hingga di bawah tegangan ambang 74HC132 pula. Pengisian dan pembuangan pada C1 yang terjadi berkali-kali ini menyebabkan terjadinya osilasi dengan frekwensi yang dapat dihitung dengan menggunakan rumus berikut:

T = Waktu Pengisian C1 + Waktu Pembuangan C1

Waktu Pengisian C1 = Waktu Pembuangan C1 maka

T = 2 * Waktu Pengisian C1

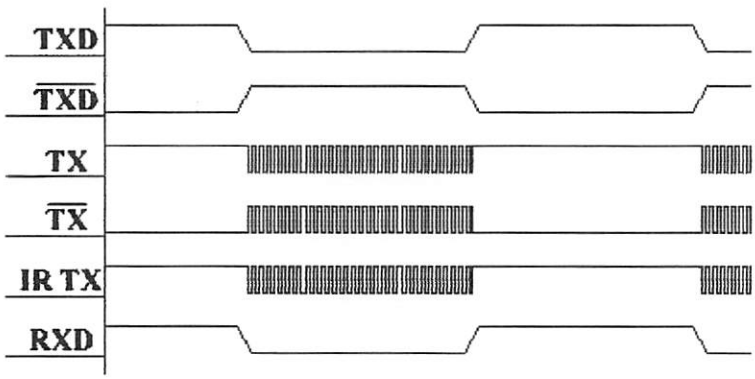
$$T = 2RC\left[\ln\left[\frac{Vs - VT-}{Vs}\right] - \ln\left[\frac{Vs - VT+}{Vs}\right]\right]$$

di mana VT- adalah batas bawah tegangan ambang 74HC132 yaitu sekitar 2 Volt dan VT+ adalah batas atas dari tegangan ambang 74HC132 yaitu sekitar 3 Volt. Dengan R sebesar 3,9K, C10nF dan Vsupply = 5Volt maka akan diperoleh harga T = 31,63 uS

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

$$F = 31{,}616\ KHz$$

Jadi pada intinya apabila input TXD berkondisi high maka frekwensi oscillator sebesar 31,616 KHz yang terjadi pada pin nomor 4 akan dilewatkan ke outputnya dengan frekwensi yang sama persis, namun bila TXD berkondisi low maka osilasi pada pin nomor 4 akan berhenti dan output dari gerbang adalah high.



Timing Diagram

Ayunan sinyal berfrekwensi 31,6 KHz ini diperkuat lagi oleh gerbang lain dari 74HC132 yang dibentuk menjadi inverter dan diteruskan ke transistor BD400 yangmengalirkan sinyal-sinyal frekwensi hasil dari modulator tersebut ke Diode Infrared.

Pancaran Diode Infrared diterima oleh IR Module dan membuat output modul ini menjadi low hingga pancaran Diode Infared berhenti dan output dari modul menjadi high. Hasil output dari modul ini yaitu RXD seperti yang tampak pada timing diagrammempunyai bentuk gelombang yang sama persis dengan TXD.

DATA SHEET

Analog to- Digital Converter
(ADC 0804)

National Semiconductor

November 1999

ADC0801/ADC0802/ADC0803/ADC0804/ADC0805 8-Bit µP 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

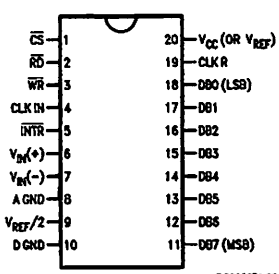
- Compatible with 8080 µP derivatives—no interfacing logic needed - access time - 135 ns
- Easy interface to all microprocessors, or operates "stand alone"
- Differential analog voltage inputs
- Logic inputs and outputs meet both MOS and TTL voltage level specifications
- Works with 2.5V (LM336) voltage 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_{DC}, 2.5 V_{DC}, or analog span adjusted voltage reference

Key Specifications

- Resolution 8 bits
- Total error ±¼ LSB, ±½ LSB and ±1 LSB
- Conversion time 100 µs

Connection Diagram

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



DS005671-30

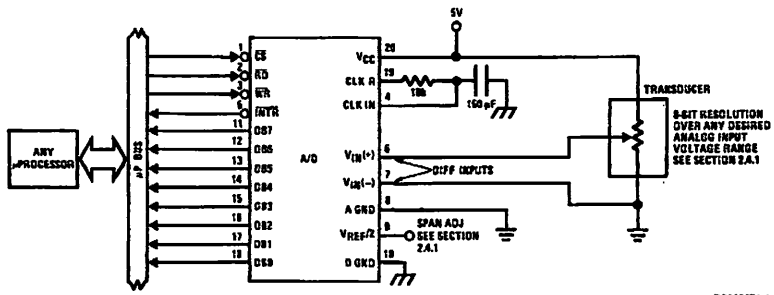
See Ordering Information

Ordering Information

TEMP RANGE		0°C TO 70°C	0°C TO 70°C	−40°C TO +85°C
ERROR	±¼ Bit Adjusted			ADC0801LCN
	±½ Bit Unadjusted	ADC0802LCWM		ADC0802LCN
	±½ Bit Adjusted			ADC0803LCN
	±1Bit Unadjusted	ADC0804LCWM	ADC0804LCN	ADC0805LCN/ADC0804LCJ
PACKAGE OUTLINE		M20B—Small Outline	N20A—Molded DIP	

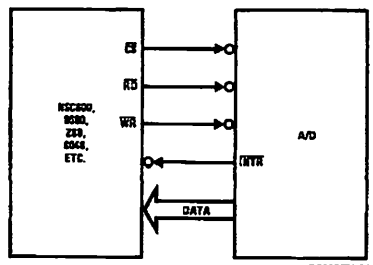
TRI-STATE® is a registered trademark of National Semiconductor Corp.
Z-80® is a registered trademark of Zilog Corp.

Typical Applications



DS005671-1

8080 Interface



DS005671-31

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	±¼ LSB		
ADC0802		±½ LSB	
ADC0803	±½ 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 (60 seconds)	215°C
Infrared (15 seconds)	220°C
Storage Temperature Range	−65°C to +150°C
Package Dissipation at T_A=25°C	875 mW
ESD Susceptibility (Note 10)	800V

Operating Ratings (Notes 1, 2)

Temperature Range	$T_{MIN} \le T_A \le T_{MAX}$
ADC0804LCJ	−40°C≤T_A≤+85°C
ADC0801/02/03/05LCN	−40°C≤T_A≤+85°C
ADC0804LCN	0°C≤T_A≤+70°C
ADC0802/04LCWM	0°C≤T_A≤+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}, $T_{MIN} \le T_A \le T_{MAX}$ 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)			±¼	LSB
ADC0802: Total Unadjusted Error (Note 8)	$V_{REF}/2$=2.500 V_{DC}			±½	LSB
ADC0803: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			±½	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		kΩ
	ADC0804 (Note 9)	0.75	1.1		kΩ
Analog Input Voltage Range	(Note 4) V(+) or V(−)	Gnd−0.05		V_{CC}+0.05	V_{DC}
DC Common-Mode Error	Over Analog Input Voltage Range		±1/16	±⅛	LSB
Power Supply Sensitivity	V_{CC}=5 V_{DC} ±10% Over Allowed V_{IN}(+) and V_{IN}(−) Voltage Range (Note 4)		±1/16	±⅛	LSB

AC Electrical Characteristics

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

Symbol	Parameter	Conditions	Min	Typ	Max	Units
T_C	Conversion Time	f_{CLK}=640 kHz (Note 6)	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	$\overline{INTR}$ tied to $\overline{WR}$ with $\overline{CS}$ =0 V_{DC}, f_{CLK}=640 kHz	8770		9708	conv/s
$t_{W(\overline{WR})L}$	Width of $\overline{WR}$ Input (Start Pulse Width)	$\overline{CS}$ =0 V_{DC} (Note 7)	100			ns
t_{ACC}	Access Time (Delay from Falling Edge of $\overline{RD}$ to Output Data Valid)	C_L=100 pF		135	200	ns
t_{1H}, t_{0H}	TRI-STATE Control (Delay from Rising Edge of $\overline{RD}$ to Hi-Z State)	C_L=10 pF, R_L=10k (See TRI-STATE Test Circuits)		125	200	ns
t_{WI}, t_{RI}	Delay from Falling Edge of $\overline{WR}$ or $\overline{RD}$ to Reset of $\overline{INTR}$			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 µA V_{CC}=4.75 V_{DC}			0.4	V_{DC}
V_{OUT} (1)	Logical "1" CLK R Output Voltage	I_O=−360 µA V_{CC}=4.75 V_{DC}	2.4			V_{DC}
DATA OUTPUTS AND $\overline{INTR}$						
V_{OUT} (0)	Logical "0" Output Voltage Data Outputs $\overline{INTR}$ 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 µA, V_{CC}=4.75 V_{DC}	2.4			V_{DC}
V_{OUT} (1)	Logical "1" Output Voltage	I_O=−10 µA, 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°C	4.5	6		mA_{DC}
I_{SINK}		V_{OUT} Short to V_{CC}, T_A=25°C	9.0	16		mA_{DC}
POWER SUPPLY						
I_{CC}	Supply Current (Includes Ladder Current) ADC0801/02/03/04LCJ/05 ADC0804LCN/LCWM	f_{CLK}=640 kHz, $V_{REF}/2$=NC, T_A=25°C and $\overline{CS}$ =5V		 1.1 1.9	 1.8 2.5	 mA 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 $V_{IN}(-) \geq V_{IN}(+)$ 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).

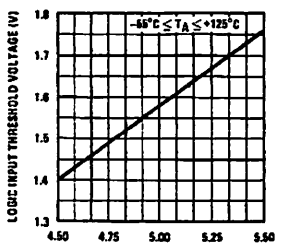
Note 8: None of these A/Ds requires a zero adjust (see section 2.5.1). To obtain zero code at other analog input voltages see section 2.5 and *Figure 7*.

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 kΩ. In all versions of the ADC0804 except the ADC0804LCJ, each resistor is typically 2.2 kΩ.

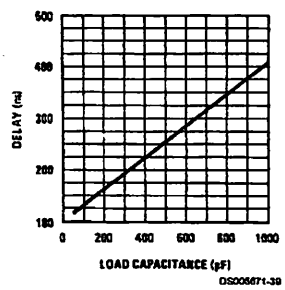
Note 10: Human body model, 100 pF discharged through a 1.5 kΩ resistor.

Typical Performance Characteristics

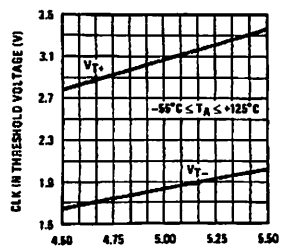
Logic Input Threshold Voltage vs. Supply Voltage



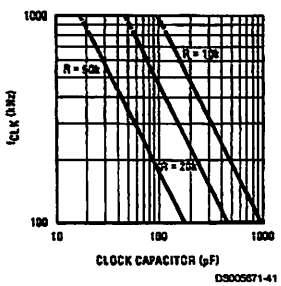
Delay From Falling Edge of RD to Output Data Valid vs. Load Capacitance



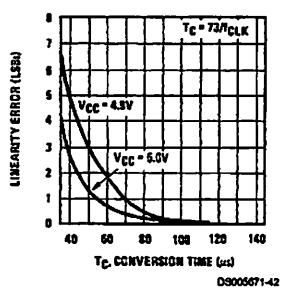
CLK IN Schmitt Trip Levels vs. Supply Voltage



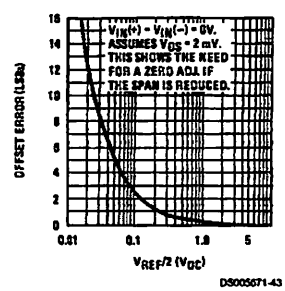
f_{CLK} vs. Clock Capacitor



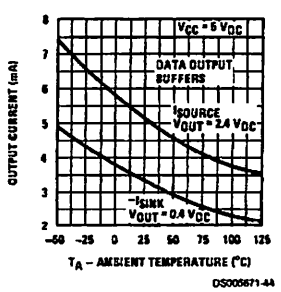
Full-Scale Error vs Conversion Time



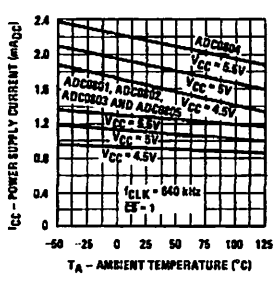
Effect of Unadjusted Offset Error vs. $V_{REF}/2$ Voltage



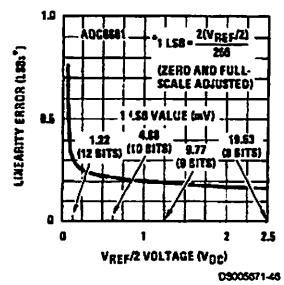
Output Current vs Temperature



Power Supply Current vs Temperature (Note 9)

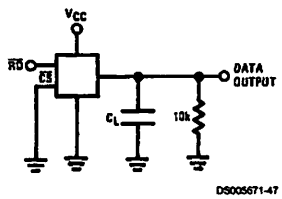


Linearity Error at Low $V_{REF}/2$ Voltages

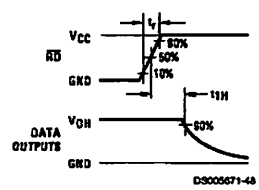


TRI-STATE Test Circuits and Waveforms

t_{1H}

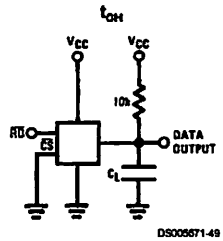


t_{1H}, C_L=10 pF

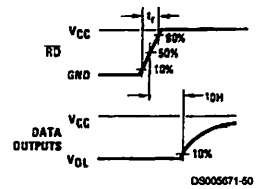


t_r=20 ns

t_{0H}

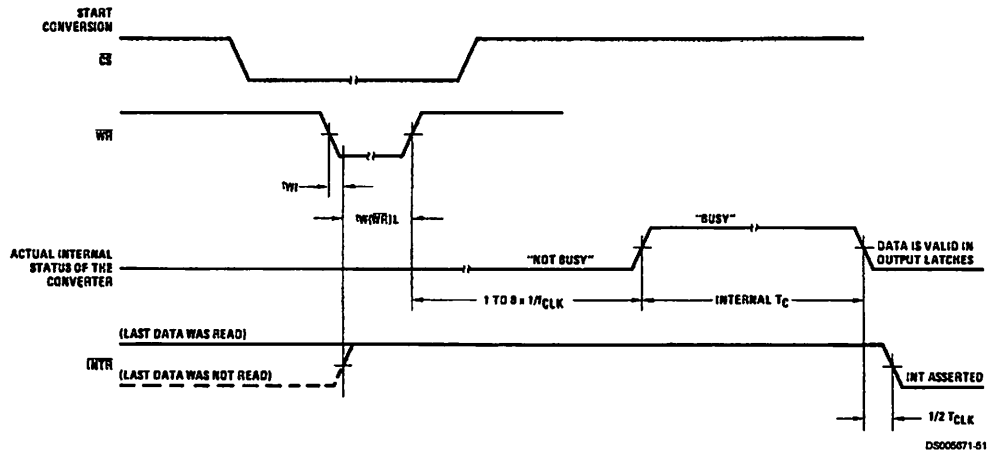


t_{0H}, C_L=10 pF



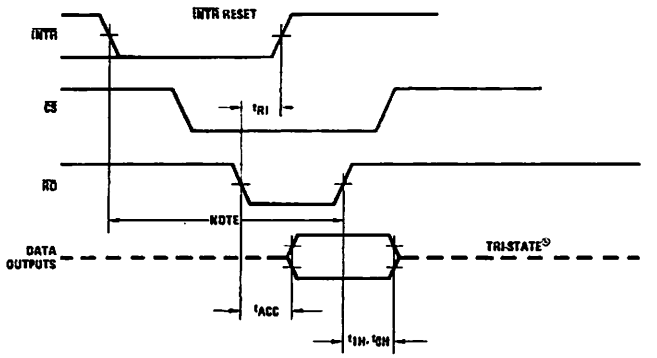
t_r=20 ns

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



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

Output Enable and Reset with $\overline{INTR}$

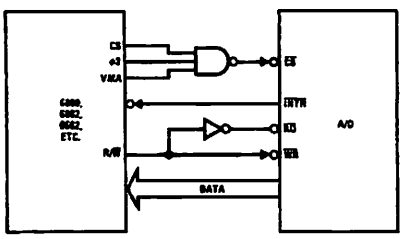


DS005671-52

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

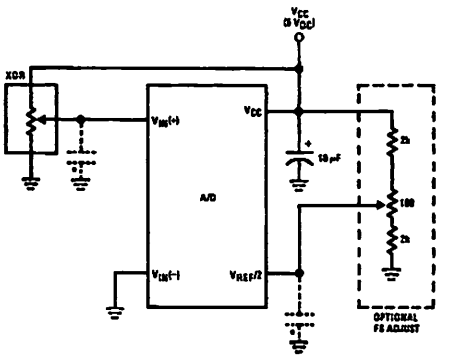
Typical Applications

6800 Interface



DS005671-53

Ratiometeric with Full-Scale Adjust

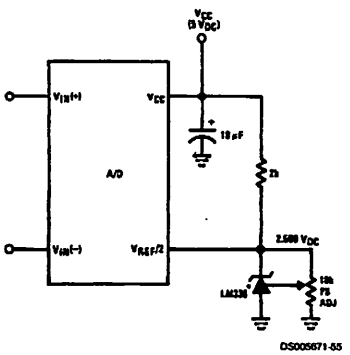


DS005671-54

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

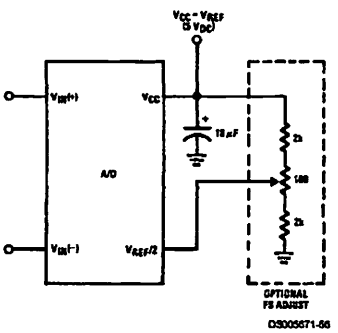
Typical Applications (Continued)

Absolute with a 2.500V Reference

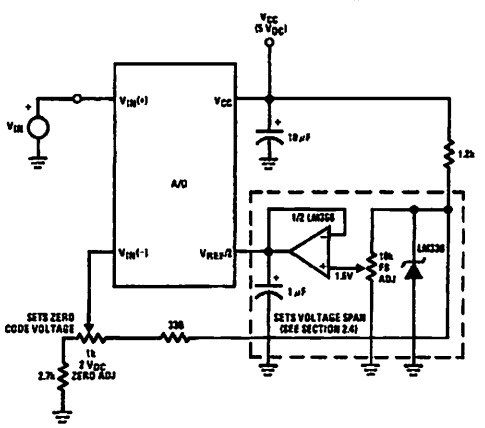


*For low power, see also LM385–2.5

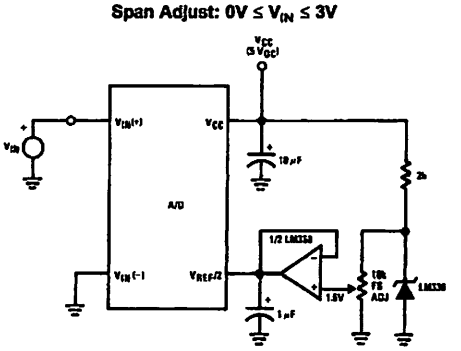
Absolute with a 5V Reference



Zero-Shift and Span Adjust: 2V ≤ V_{IN} ≤ 5V

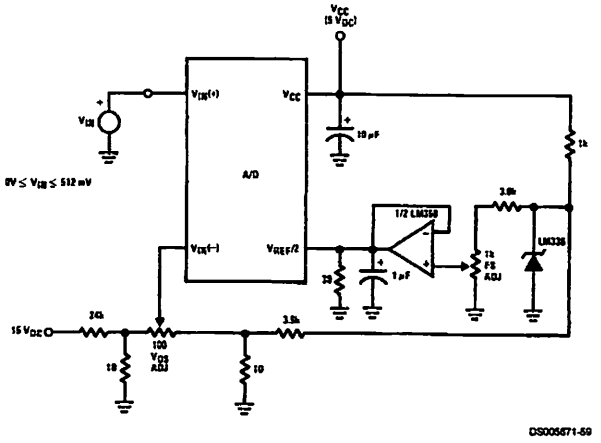


Span Adjust: 0V ≤ V_{IN} ≤ 3V



Typical Applications (Continued)

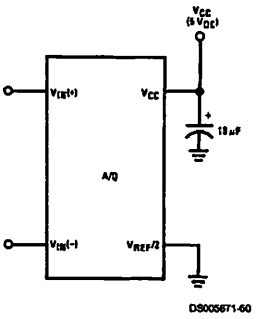
Directly Converting a Low-Level Signal



DS005671-59

$V_{REF}/2$=256 mV

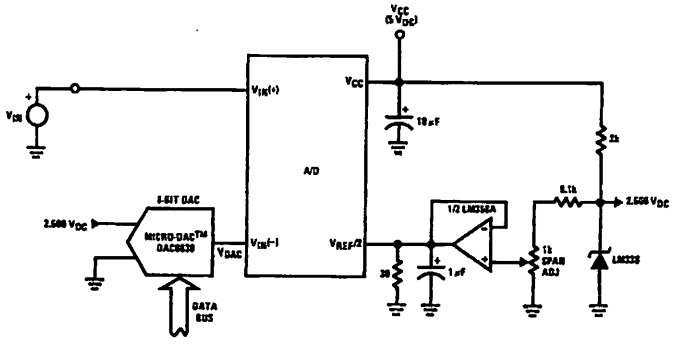
A µP Interfaced Comparator



DS005671-60

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

1 mV Resolution with µP Controlled Range

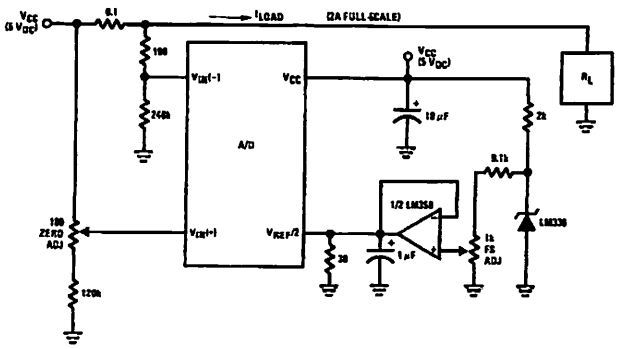


DS005671-61

$V_{REF}/2$=128 mV
1 LSB=1 mV
$V_{DAC} \leq V_{IN} \leq (V_{DAC}+256$ mV)
$0 \leq V_{DAC} < 2.5V$

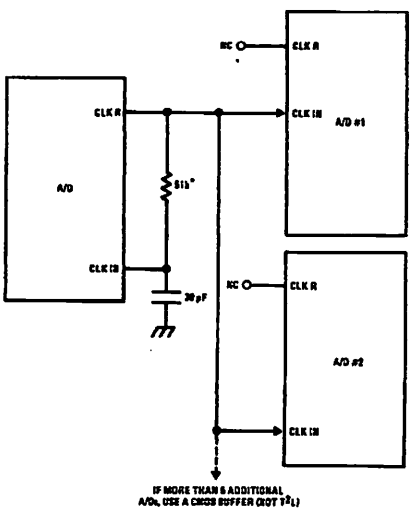
Typical Applications (Continued)

Digitizing a Current Flow



DS005671-62

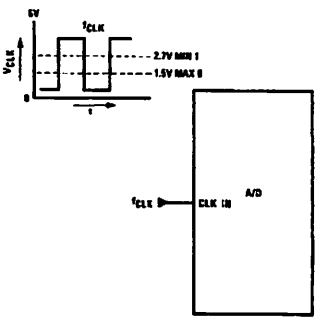
Self-Clocking Multiple A/Ds



DS005671-63

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

External Clocking

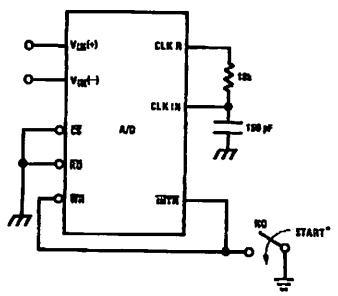


DS005671-64

100 kHz≤f_{CLK}≤1460 kHz

Typical Applications (Continued)

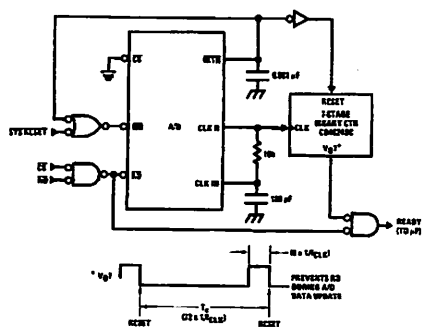
Self-Clocking in Free-Running Mode



DS005671-65

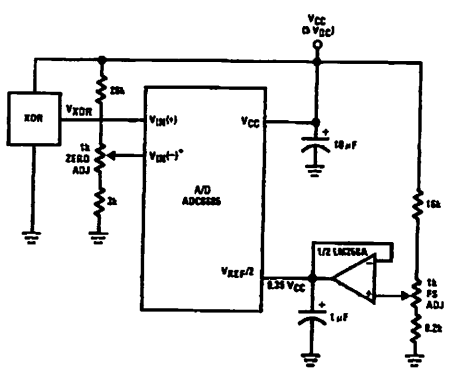
*After power-up, a momentary grounding of the $\overline{WR}$ input is needed to guarantee operation.

µP Interface for Free-Running A/D



DS005671-66

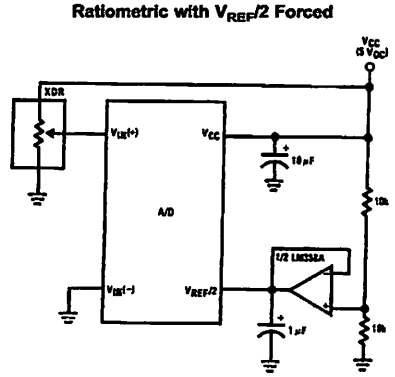
Operating with "Automotive" Ratiometric Transducers



DS005671-67

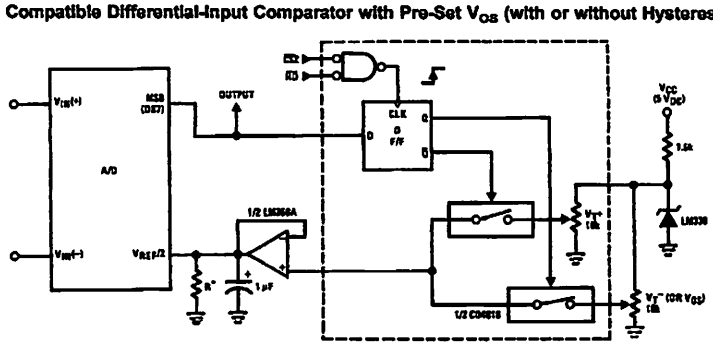
*$V_{IN}(-)$=0.15 V_{CC}
15% of $V_{CC} \leq V_{XDR} \leq 85\%$ of V_{CC}

Ratiometric with $V_{REF}/2$ Forced



DS005671-68

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

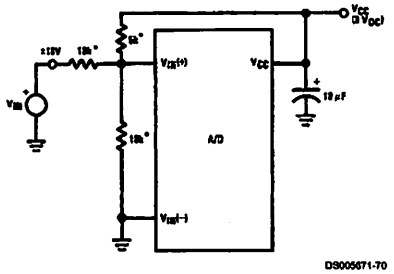


DS005671-69

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

Typical Applications (Continued)

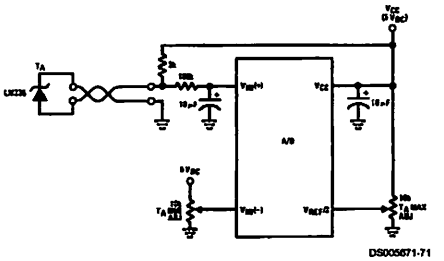
Handling ±10V Analog Inputs



DS005671-70

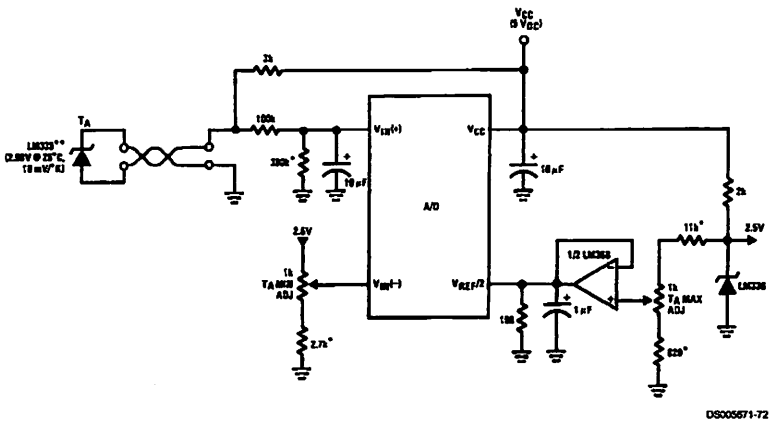
*Beckman Instruments #694-3-R10K resistor array

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



DS005671-71

µP Interfaced Temperature-to-Digital Converter



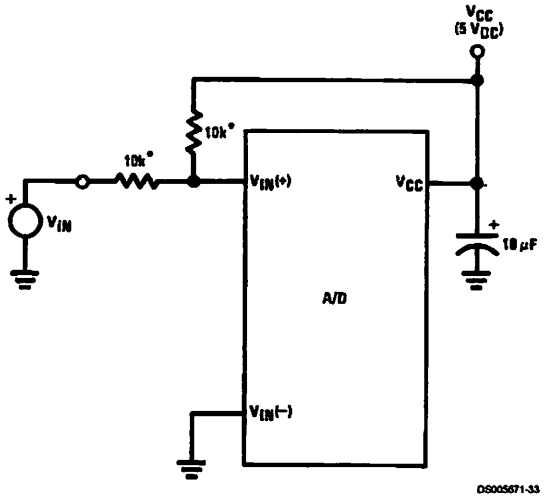
DS005671-72

*Circuit values shown are for $0°C \leq T_A \leq +128°C$

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

Typical Applications (Continued)

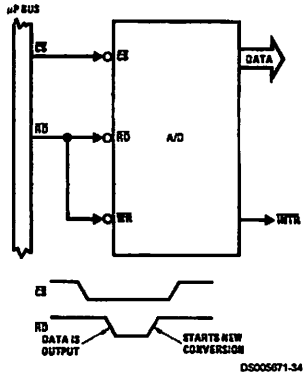
Handling ±5V Analog Inputs



DS005671-33

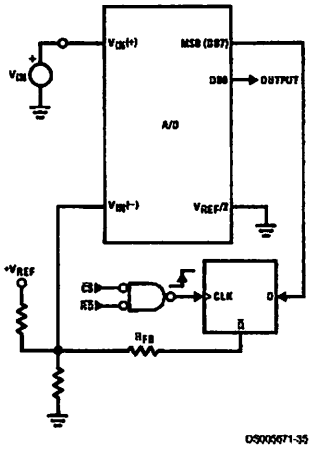
*Beckman Instruments #694-3-R10K resistor array

Read-Only Interface



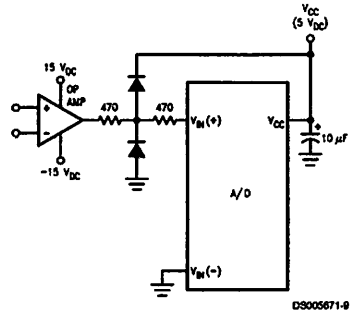
DS005671-34

µP Interfaced Comparator with Hysteresis



DS005671-35

Protecting the Input

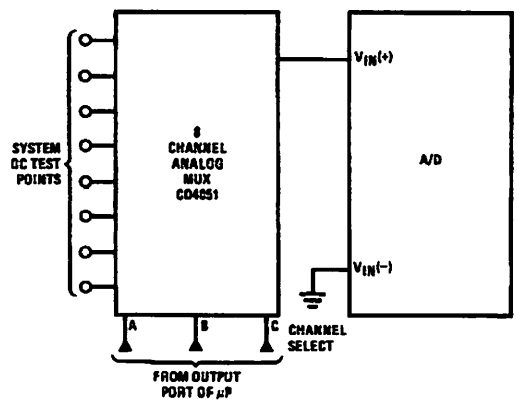


DS005671-9

Diodes are 1N914

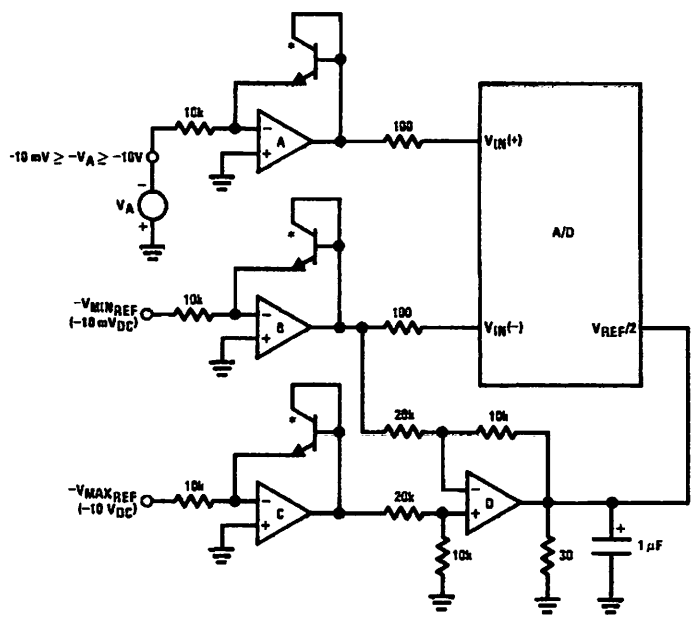
Typical Applications (Continued)

Analog Self-Test for a System



DS005671-36

A Low-Cost, 3-Decade Logarithmic Converter

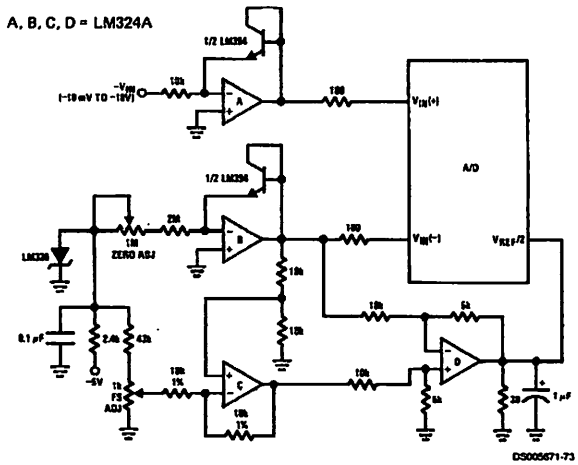


DS005671-37

*LM389 transistors
A, B, C, D = LM324A quad op amp

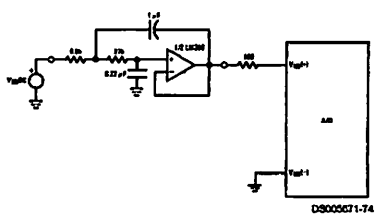
Typical Applications (Continued)

3-Decade Logarithmic A/D Converter



DS005671-73

Noise Filtering the Analog Input



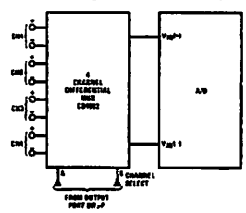
DS005671-74

f_C=20 Hz

Uses Chebyshev implementation for steeper roll-off unity-gain, 2nd order, low-pass filter

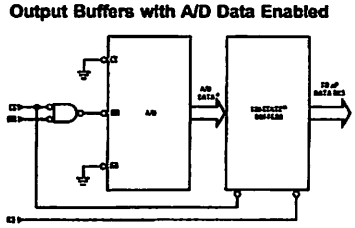
Adding a separate filter for each channel increases system response time if an analog multiplexer is used

Multiplexing Differential Inputs



DS005671-75

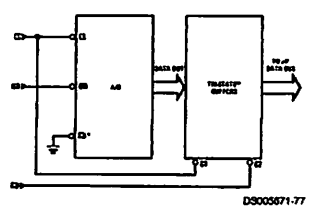
Output Buffers with A/D Data Enabled



DS005671-76

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

Increasing Bus Drive and/or Reducing Time on Bus

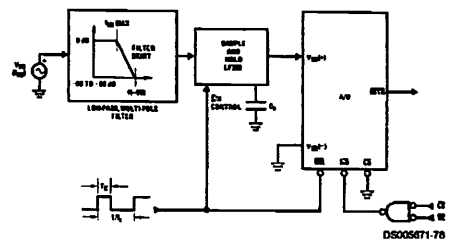


DS005671-77

*Allows output data to set-up at falling edge of $\overline{CS}$

Typical Applications (Continued)

Sampling an AC Input Signal

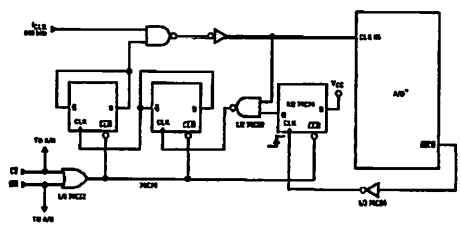


DS005671-78

Note 11: Oversample whenever possible [keep fs > 2f(−60)] to eliminate input frequency folding (aliasing) and to allow for the skirt response of the filter.

Note 12: Consider the amplitude errors which are introduced within the passband of the filter.

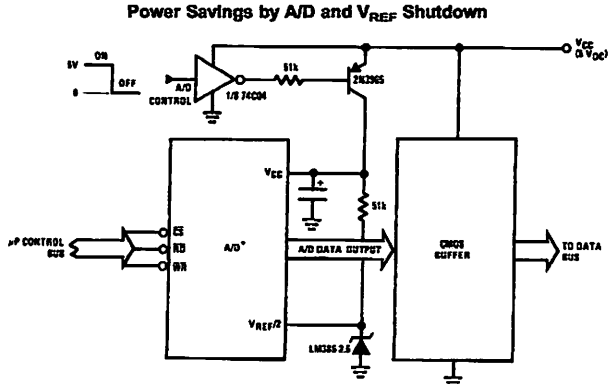
70% Power Savings by Clock Gating



DS005671-79

(Complete shutdown takes ≈ 30 seconds.)

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



DS005671-80

*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

1.0 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 ±½ 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 ±½ 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 risers are guaranteed to be no closer to the center-value points than ±¼ LSB. In other words, if we apply an analog input equal to the center-value ±¼ 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 ½ 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 +½ LSB because the digital code appeared ½ LSB in advance of the center-value of the tread. The error plots always have a constant negative slope and the abrupt upside steps are always 1 LSB in magnitude.

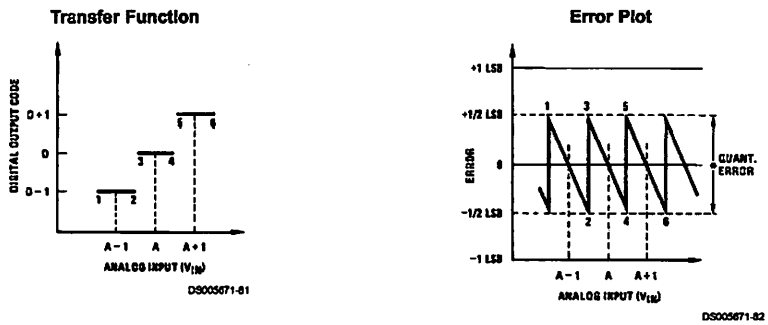


FIGURE 1. Clarifying the Error Specs of an A/D Converter Accuracy=±0 LSB: A Perfect A/D

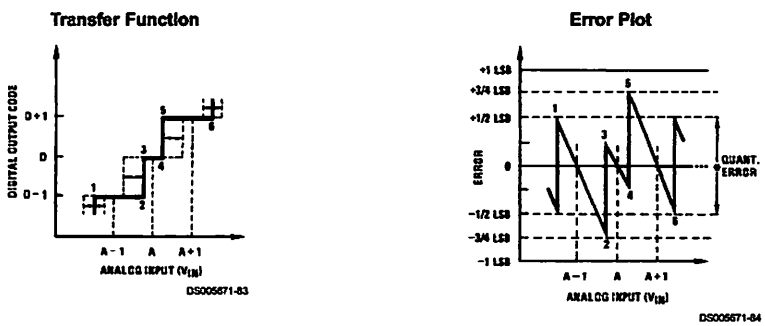


FIGURE 2. Clarifying the Error Specs of an A/D Converter Accuracy=±¼ LSB

Functional Description (Continued)

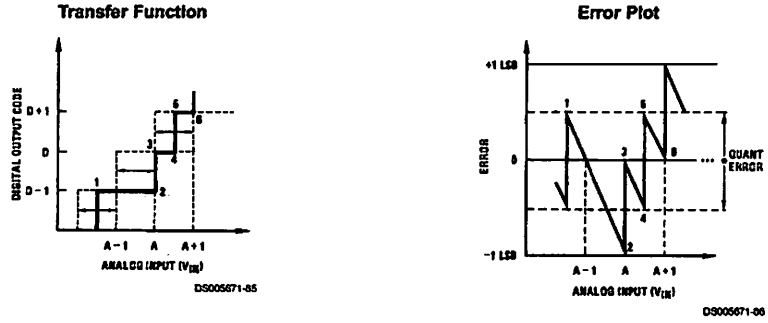


FIGURE 3. Clarifying the Error Specs of an A/D Converter
Accuracy=±½ LSB

2.0 FUNCTIONAL DESCRIPTION

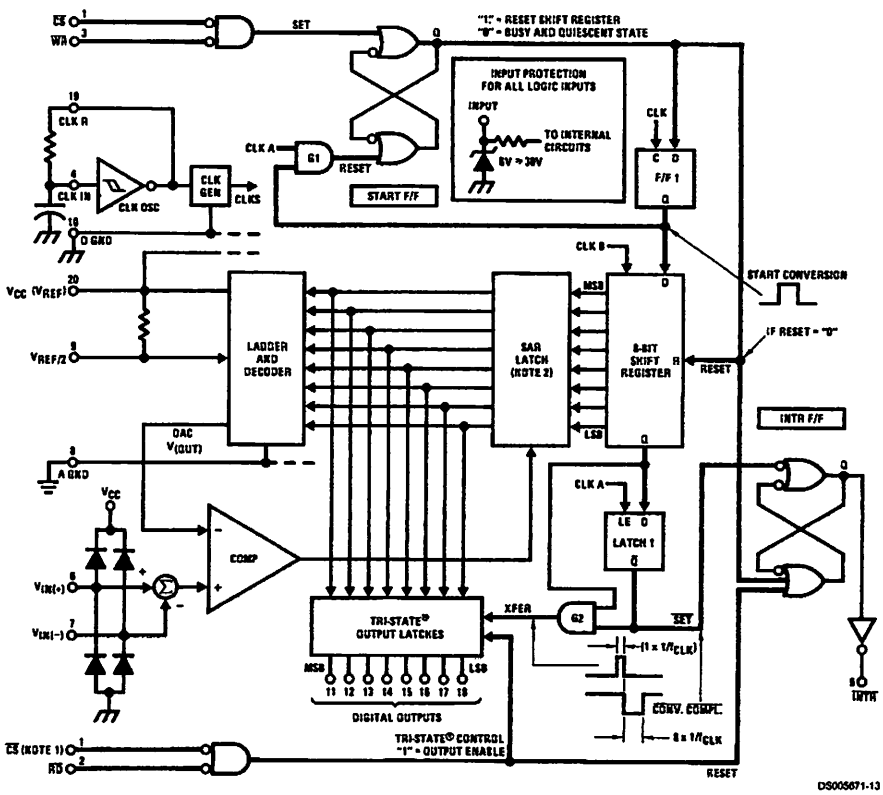
The ADC0801 series contains a circuit equivalent of the 256R network. Analog switches are sequenced by successive approximation logic to match the analog difference input voltage [$V_{IN}(+) - V_{IN}(-)$] to a corresponding tap on the R network. The most significant bit is tested first and after 8 comparisons (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 ($\overline{INTR}$ 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 $\overline{INTR}$ to the $\overline{WR}$ input with $\overline{CS}$ =0. To ensure start-up under all possible conditions, an external $\overline{WR}$ 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, resets 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: $\overline{CS}$ shown twice for clarity.

Note 14: SAR = Successive Approximation Register.

FIGURE 4. Block Diagram

After the "1" is clocked 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 $\overline{INTR}$ input signal.

Note that this $\overline{SET}$ control of the INTR F/F remains low for 8 of the external clock periods (as the internal clocks run at 1/8 of the frequency of the external clock). If the data output is continuously enabled ($\overline{CS}$ and $\overline{RD}$ both held low), the $\overline{INTR}$ output will still signal the end of conversion (by a high-to-low transition), because the $\overline{SET}$ 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 $\overline{INTR}$ output will therefore stay low for the duration of the $\overline{SET}$ 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 ($\overline{INTR}$ pin tied to $\overline{WR}$ and $\overline{CS}$ wired low—see also section 2.8), the START F/F is SET by the high-to-low transition of the $\overline{INTR}$ 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 $\overline{Q}$ output will go high, which then allows the INTR F/F to be RESET. This reduces the width of the resulting $\overline{INTR}$ output pulse to only a few propagation delays (approximately 300 ns).

When data is to be read, the combination of both $\overline{CS}$ and $\overline{RD}$ 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 ($\overline{CS}$, $\overline{RD}$, and $\overline{WR}$) meet standard T^2L 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 busses. For non-microprocessor based applications, the $\overline{CS}$ input (pin 1) can be grounded and the standard A/D Start function is obtained by an active low pulse applied at the $\overline{WR}$ input (pin 3) and the Output Enable function is caused by an active low pulse at the $\overline{RD}$ 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 4-½ clock periods. The maximum error voltage due to this slight time difference between the input voltage samples is given by:

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

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 ¼ LSB (~5 mV) when operating with a 60 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:

$$V_P = \frac{[\Delta V_{e(MAX)}(f_{CLK})]}{(2\pi f_{cm})(4.5)}$$

or

$$V_P = \frac{(5 \times 10^{-3})(640 \times 10^3)}{(6.28)(60)(4.5)}$$

which gives

$V_P \cong 1.9V$.

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

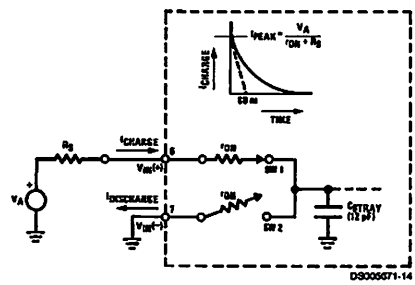
An analog input voltage with a reduced span and a relatively large zero offset can be handled easily by making use of the differential input (see section 2.4 Reference Voltage).

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_{ON} of SW 1 and SW 2 ≅ 5 kΩ

$\tau = r_{ON} C_{STRAY} \cong 5\ k\Omega \times 12\ pF = 60\ ns$

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}+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 µA. Therefore, *bypass capacitors should not be used at the analog inputs or the $V_{REF}/2$ pin* for high resistance sources (> 1 kΩ). 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 (≤ 1 kΩ) for a passive RC section or add an op amp RC active low pass filter. For low source resistance applications, (≤ 1 kΩ), a 0.1 µF bypass capacitor at the inputs will prevent noise pickup due to series lead inductance of a long wire. A

Functional Description (Continued)

100Ω 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 leads 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 kΩ. 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.*

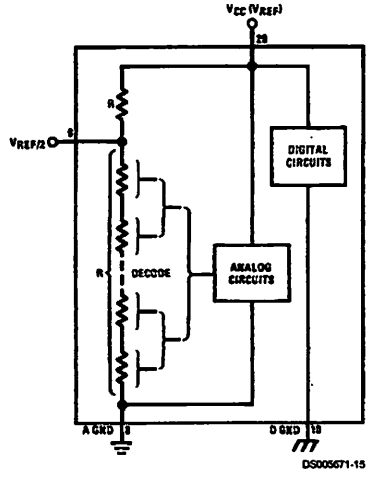


FIGURE 6. The $V_{REFERENCE}$ Design on the IC

Notice that the reference voltage for the IC is either ½ 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 ratiometric 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.

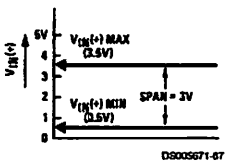
An example of the use of an adjusted reference voltage is to accommodate a reduced span—or dynamic voltage range of the analog input voltage. If the analog input voltage were to range from 0.5 V_{DC} to 3.5 V_{DC}, instead of 0V to 5 V_{DC}, the span would be 3V as shown in *Figure 7.* With 0.5 V_{DC} applied to the $V_{IN}(-)$ pin to absorb the offset, the reference voltage can be made equal to ½ of the 3V span or 1.5 V_{DC}. The A/D now will encode the $V_{IN}(+)$ signal from 0.5V to 3.5 V 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

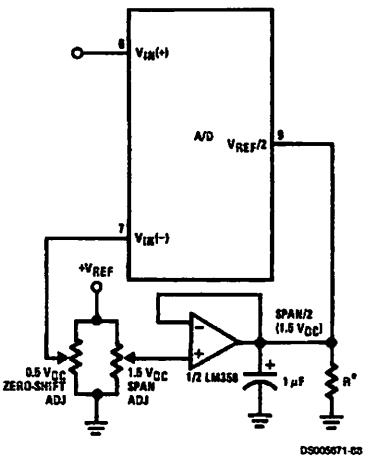
The converter can be operated in a ratiometric mode or an absolute mode. In ratiometric converter applications, the magnitude of the reference voltage is a factor in both the output of the source transducer and the output of the A/D converter and therefore cancels out in the final digital output code. The ADC0805 is specified particularly for use in ratiometric 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 (6 mV max) over $0°C \leq T_A \leq +70°C$. Other temperature range parts are also available.

Functional Description (Continued)



a) Analog Input Signal Example



*Add if $V_{REF}/2 \leq 1\ V_{DC}$ with LM358 to draw 3 mA to ground.

b) Accommodating an Analog Input from 0.5V (Digital Out = 00_{HEX}) to 3.5V (Digital Out=FF_{HEX})

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 ½ LSB value (½ 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 1½ 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 ½ 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:

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

where:

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

and

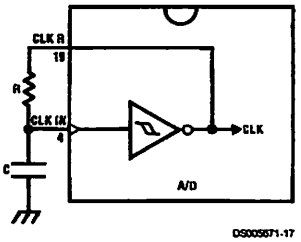
V_{MIN}=the low end (the offset zero) of the analog range. (Both are ground referenced.)

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)



DS005671-17

$$f_{CLK} \cong \frac{1}{1.1\,RC}$$

$R \cong 10\ k\Omega$

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 µF 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 ¼ 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} (5.120–1½ 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)

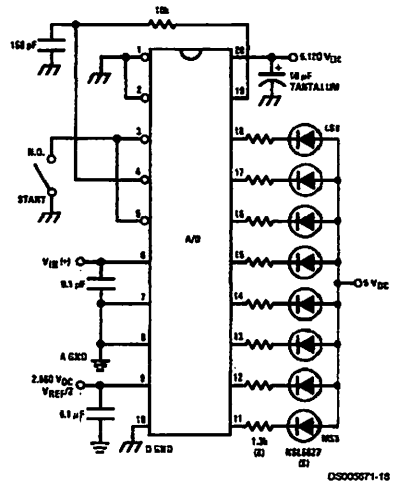


FIGURE 9. Basic A/D Tester

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 2 op amps 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 ¼ 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 dicuss 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 $\overline{I/O\ R}$ and $\overline{I/O\ W}$ strobes and decoding the address bits A0 → A7 (or address bits A8 → 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)

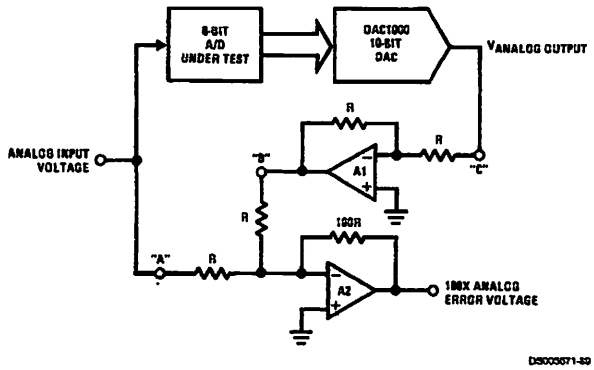


FIGURE 10. A/D Tester with Analog Error Output

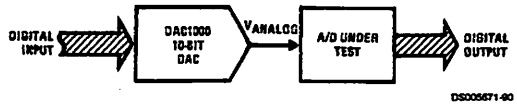


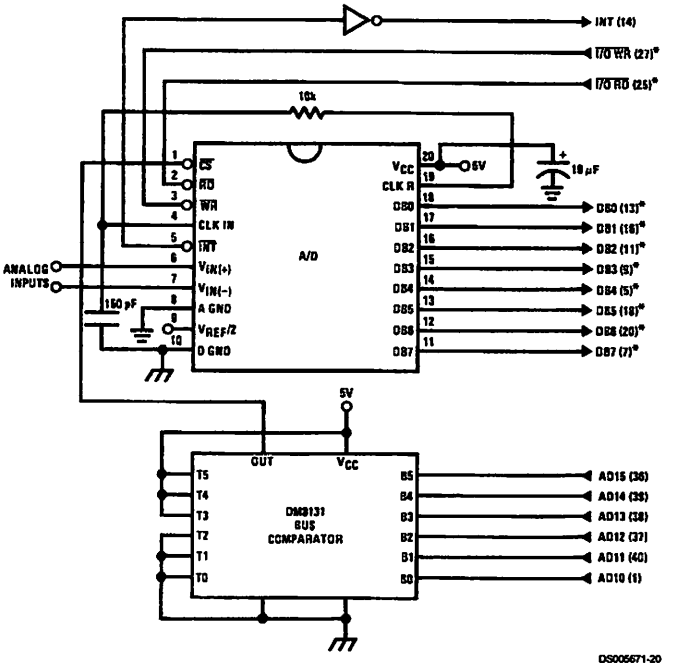
FIGURE 11. Basic "Digital" A/D Tester

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 DP8228 system controller, others are INS8080A.

Note 17: Pin 23 of the INS8228 must be tied to +12V through a 1 kΩ resistor to generate the RST 7 instruction when an interrupt is acknowledged as required by the accompanying sample program.

FIGURE 12. ADC0801_INS8080A CPU Interface

Functional Description (Continued)

SAMPLE PROGRAM FOR *Figure 12* ADC0801–INS8080A CPU INTERFACE

```
0038    C3 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 0F H            ;If # = 16. JMP to
0109    CA 13 01                    JZ CONT             ;user program
010C    D3 E0                       OUT E0 H            ;Start A/D
010E    FB                          EI                  ;Enable interrupt
010F    00          LOOP:           NOP                 ;Loop until end of
0110    C3 0F 01                    JMP LOOP            ;conversion
0113      •         CONT:             •
  •       •           •               •
  •       •         (User program to  •
  •       •         process data)     •
  •       •           •               •
  •       •           •               •
0300    DB E0       LD DATA:        IN E0 H             ;Load data into accumulator
0302    77                          MOV M, A            ;Store data
0303    23                          INX H               ;Increment storage pointer
0304    C3 03 01                    JMP RETURN
```

DS005671-99

Note 18: The stack pointer must be dimensioned because a RST 7 instruction pushes the PC onto the stack.

Note 19: 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

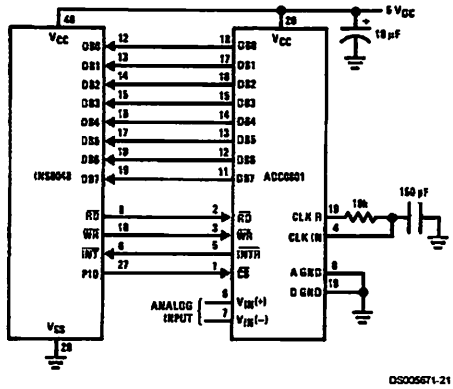
The following sample program and associated hardware shown in *Figure 12* may be used to input data from the converter to the INS8080A CPU chip set (comprised of the INS8080A microprocessor, the INS8228 system controller and the INS8224 clock generator). For simplicity, the A/D is controlled as an I/O 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 $\overline{INT}$ 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)



DS005671-21

FIGURE 13. INS8048 Interface

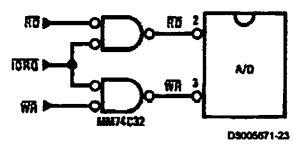
SAMPLE PROGRAM FOR *Figure 13* INS8048 INTERFACE

```
04 10                   JMP     10H             ;Program starts at addr 10
                        ORG     3H
04 50                   JMP     50H             ;Interrupt jump vector
                        ORG     10H             ;Main program
99 FE                   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
B8 20                   MOV     R0, #20H        ;Data address
B9 FF                   MOV     R1, #0FFH       ;Dummy address
BA 10                   MOV     R2, #10H        ;Counter for 16 bytes
23 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                      EN      I               ;Enable interrupt
96 21           LOOP:   JNZ     LOOP            ;Wait for interrupt
EA 1B                   DJNZ    R2, AGAIN       ;If 16 bytes are read
00                      NOP                     ;go to user's program
00                      NOP
                        ORG     50H
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
```

DS005671-A0

4.2 Interfacing the Z-80

The Z-80 control bus is slightly different from that of the 8080. General $\overline{RD}$ and $\overline{WR}$ strobes are provided and separate memory request, $\overline{MREQ}$, and I/O request, $\overline{IORQ}$, 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 $\overline{RD}$ and $\overline{WR}$ 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*.



DS005671-23

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)

ing I/O input instructions. For example, MUX channel selection for the A/D can be accomplished with this operating mode.

4.3 Interfacing 6800 Microprocessor Derivatives (6502, etc.)

The control bus for the 6800 microprocessor derivatives does not use the $\overline{RD}$ and $\overline{WR}$ strobe signals. Instead it employs a single R/$\overline{W}$ line and additional timing, if needed, can be derived fom the $\phi 2$ clock. All I/O devices are memory mapped in the 6800 system, and a special signal, VMA, indicates that the current address is valid. *Figure 15* shows an interface schematic where the A/D is memory mapped in the 6800 system. For simplicity, the $\overline{CS}$ decoding is shown using ½ DM8092. Note that in many 6800 systems, an already decoded $\overline{4/5}$ line is brought out to the common bus at pin 21. This can be tied directly to the $\overline{CS}$ pin of the A/D, provided that no other devices are addressed at HX ADDR: 4XXX or 5XXX.

The following subroutine performs essentially the same function as in the case of the 8080A interface and it can be called from anywhere in the user's program.

In *Figure 16* the ADC0801 series is interfaced to the M6800 microprocessor through (the arbitrarily chosen) Port B of the MC6820 or MC6821 Peripheral Interface Adapter, (PIA). Here the $\overline{CS}$ pin of the A/D is grounded since the PIA is already memory mapped in the M6800 system and no $\overline{CS}$ decoding is necessary. Also notice that the A/D output data lines are connected to the microprocessor bus under program control through the PIA and therefore the A/D $\overline{RD}$ pin can be grounded.

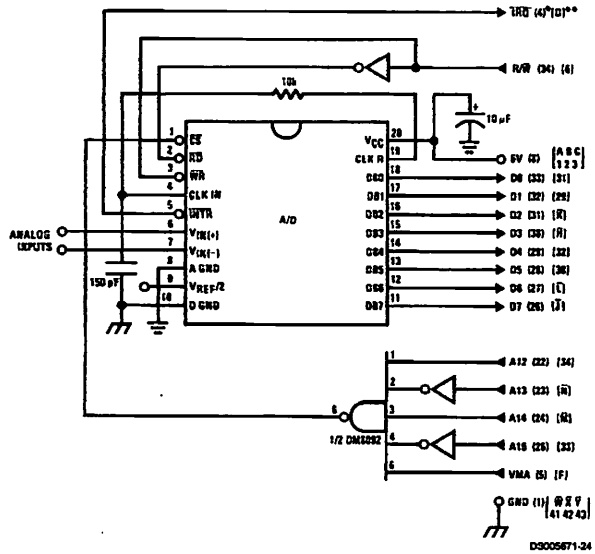
A sample interface program equivalent to the previous one is shown below *Figure 16*. The PIA Data and Control Registers of Port B are located at HEX addresses 8006 and 8007, respectively.

5.0 GENERAL APPLICATIONS

The following applications show some interesting uses for the A/D. The fact that one particular microprocessor is used is not meant to be restrictive. Each of these application circuits would have its counterpart using any microprocessor that is desired.

5.1 Multiple ADC0801 Series to MC6800 CPU Interface

To transfer analog data from several channels to a single microprocessor system, a multiple converter scheme presents several advantages over the conventional multiplexer single-converter approach. With the ADC0801 series, the differential inputs allow individual span adjustment for each channel. Furthermore, all analog input channels are sensed simultaneously, which essentially divides the microprocessor's total system servicing time by the number of channels, since all conversions occur simultaneously. This scheme is shown in *Figure 17*.



Note 20: Numbers in parentheses refer to MC6800 CPU pin out.

Note 21: Number or letters in brackets refer to standard M6800 system common bus code.

FIGURE 15. ADC0801-MC6800 CPU Interface

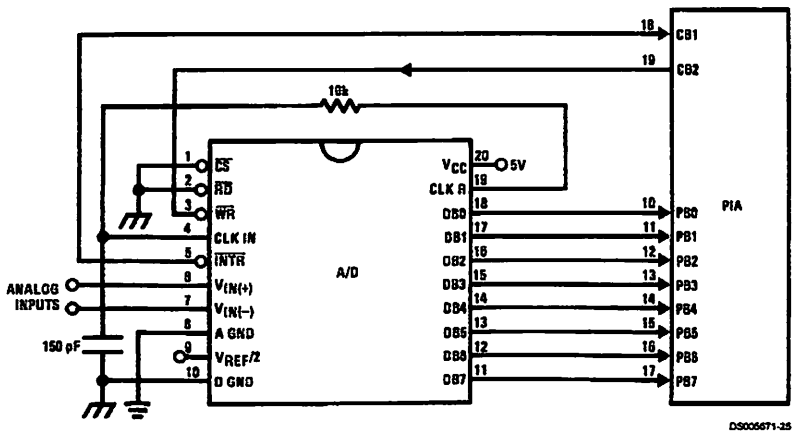
Functional Description (Continued)

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    $FFF8    ; 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            CPX    #$020F   ; Is final data stored?
0022   27 14               BEQ    ENDP
0024   B7 50 00            STAA   $5000    ; Restarts ADC0801
0027   08                  INX
0028   DF 34               STX    TEMP1
002A   20 F0               BRA    CONVRT
002C   DE 34      INTRPT   LDX    TEMP1
002E   B6 50 00            LDAA   $5000    ; 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
0036   00 00      TEMP2    FDB    $0000
0038   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
```

DS005671-A1

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



DS005671-25

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 IRQ low CPU
0013  FF FF F8            STX   $FFF8   ; jumps to 0038
0016  B6 80 06            LDAA  PIAORB  ; Clear possible IRQ flags
0019  4F                  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  #$34
0023  86 3D               LDAA  #$3D
0025  F7 80 07   CONVRT   STAB  PIACRB  ; Starts ADC0801
0028  B7 80 07            STAA  PIACRB
002B  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  DF 40               STX   TEMP1
0036  20 ED               BRA   CONVRT
0038  DE 40      INTRPT   LDX   TEMP1
003A  B6 80 06            LDAA  PIAORB  ; Read data in
003D  A7 00               STAA  X       ; Store it at X
003F  3B                  RTI
0040  02 00       TEMP1   FDB   $0200   ; Starting address for
                                        ; data storage
0042  CE 02 00    ENDP    LDX   #$0200  ; Reinitialize TEMP1
0045  DF 40               STX   TEMP1
0047  39                  RTS           ; Return from subroutine
                  PIAORB  EQU   $8006   ; To user's program
                  PIACRB  EQU   $8007
```

DS005671-A2

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.

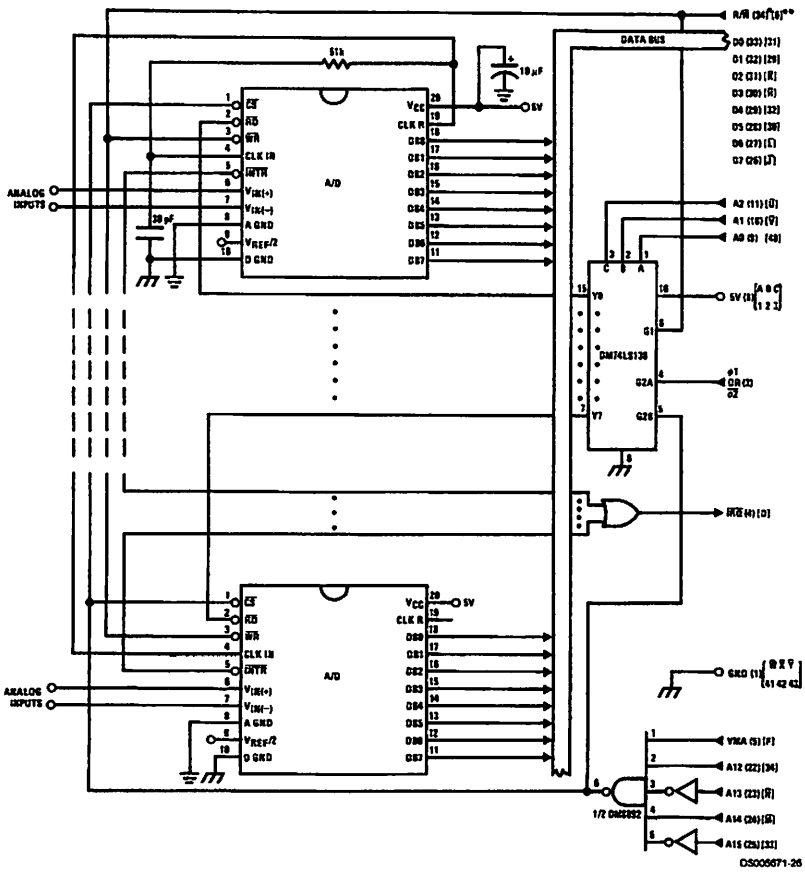
All the converters are started simultaneously with a STORE instruction at HEX address 5000. Note that any other HEX address of the form 5XXX will be decoded by the circuit, pulling all the $\overline{CS}$ inputs low. This can easily be avoided by using a more definitive address decoding scheme. All the 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 M6800 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	; Upon $\overline{IRQ}$ LOW CPU
0015	FF FF F8		STX	$FFF8	; Jumps to 002A
0018	B7 50 00		STAA	$5000	; Starts all A/D's
001B	0E		CLI		
001C	3E		WAI		; Wait for interrupt
001D	CE 50 00		LDX	#$5000	
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

DS005671-A3

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		CPX	#$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 EB		BRA	INTRPT	; Branch to 002A
003F	3B	RETURN	RTI		
0040	50 00	INDEX1	FDB	$5000	; Starting address for A/D
0042	02 00	INDEX2	FDB	$0200	; Starting address for data storage
0044	00 00	TEMP	FDB	$0000	

DS005671-A4

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 INS8080A microprocessor system. The total allowable input offset voltage error for this preamp is only 50 μV for ¼ LSB error. This would obviously require very precise amplifiers. The expression for the differential output voltage of the preamp is:

$$V_O = \underbrace{[V_{IN}(+) - V_{IN}(-)]}_{\text{SIGNAL}} \underbrace{\left[1 + \frac{2R2}{R1}\right]}_{\text{GAIN}} +$$

$$\underbrace{(V_{OS_2} - V_{OS_1} - V_{OS_3} \pm I_X R_X)}_{\text{DC ERROR TERM}} \underbrace{\left(1 + \frac{2R2}{R1}\right)}_{\text{GAIN}}$$

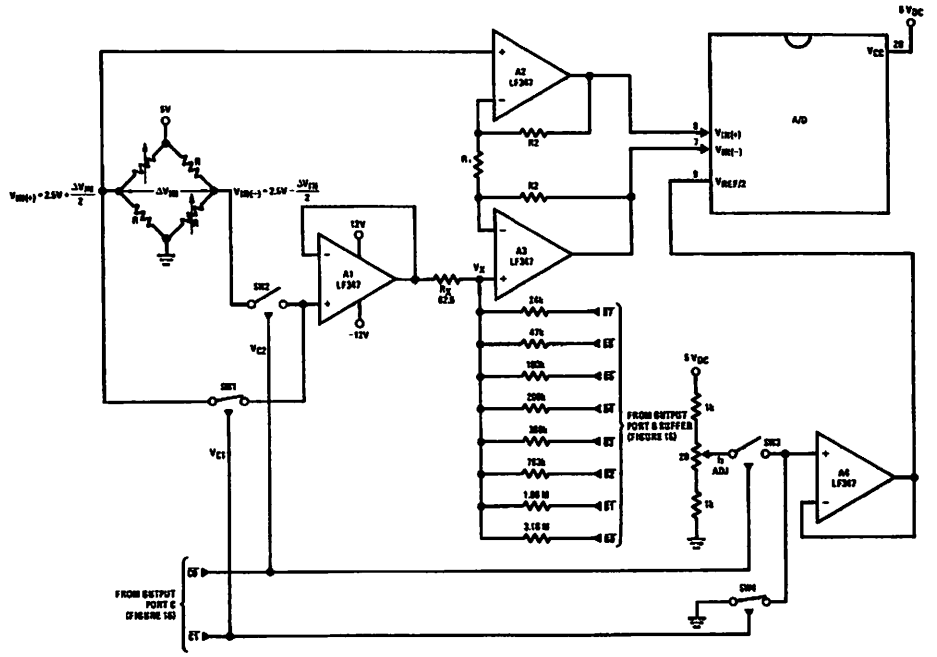
where I_X is the current through resistor R_X. All of the offset error terms can be cancelled by making $\pm I_X R_X = V_{OS1} + V_{OS3} - V_{OS2}$. This is the principle of this auto-zeroing scheme.

The INS8080A uses the 3 I/O ports of an INS8255 Programable 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 µV, which will null the offset error term to ¼ 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.



DS005671-61

Note 26: R2 = 49.5 R1

Note 27: Switches are LMC13334 CMOS analog switches.

Note 28: The 9 resistors used in the auto-zero section can be ±5% tolerance.

FIGURE 18. Gain of 100 Differential Transducer Preamp

Functional Description (Continued)

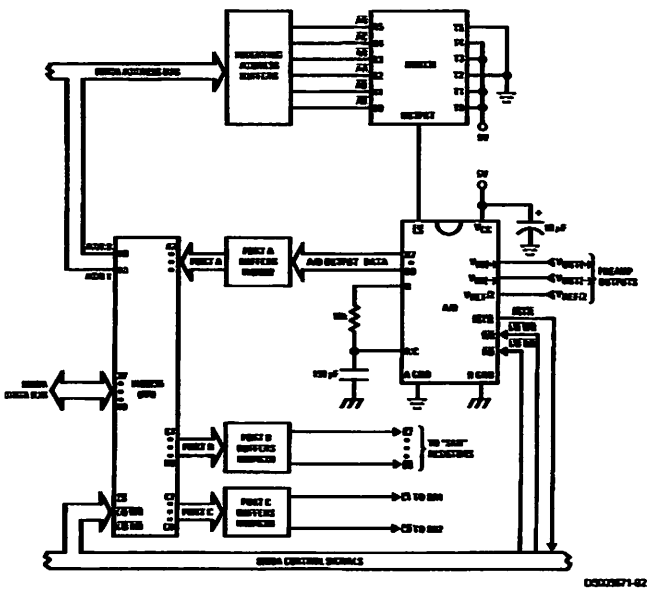


FIGURE 19. Microprocessor Interface Circuitry for Differential Preamp

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 SLC 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

PPI 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 ($\overline{INTR}$ 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 $\overline{INTR}$ 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)

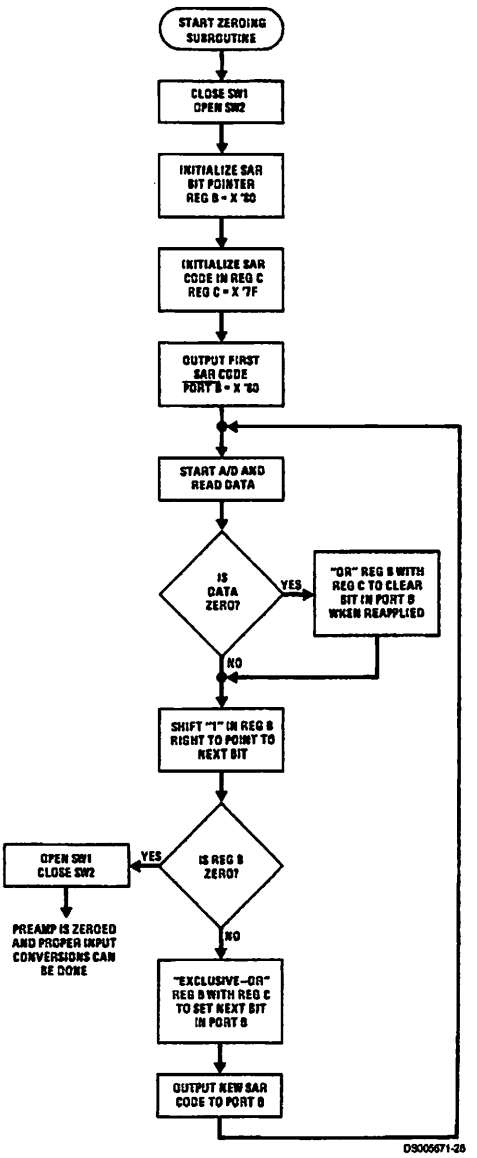


FIGURE 20. Flow Chart for Auto-Zero Routine

Functional Description (Continued)

```
3D00   3E90     MVI 90
3D02   D3E7     Out Control Port                    ;Program PPI
3D04   2601     MVI H 01          Auto-Zero Subroutine
3D06   7C       MOV A,H
3D07   D3E6     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   D3E5     OUT B                              ;Port B = SAR code
3D10   31AA3D   LXI SP 3DAA       Start            ;Dimension stack pointer
3D13   D3E4     OUT A                              ;Start A/D
3D15   FB       IE
3D16   00       NOP               Loop             ;Loop until INT asserted
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   78       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   C30D3D   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   D3E6     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   78       MOV A,B                            ;Is B Reg = 0? If not stay
3C43   E6FF     ANI FF                             ;in auto zero subroutine
3C45   C21A3D   JNZ Auto-Zero
3C48   C33D3D   JMP Normal
```

DS005671-A5

Note 29: 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 flip-flop
01	A/D 1
02	A/D 2
03	A/D 3
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.

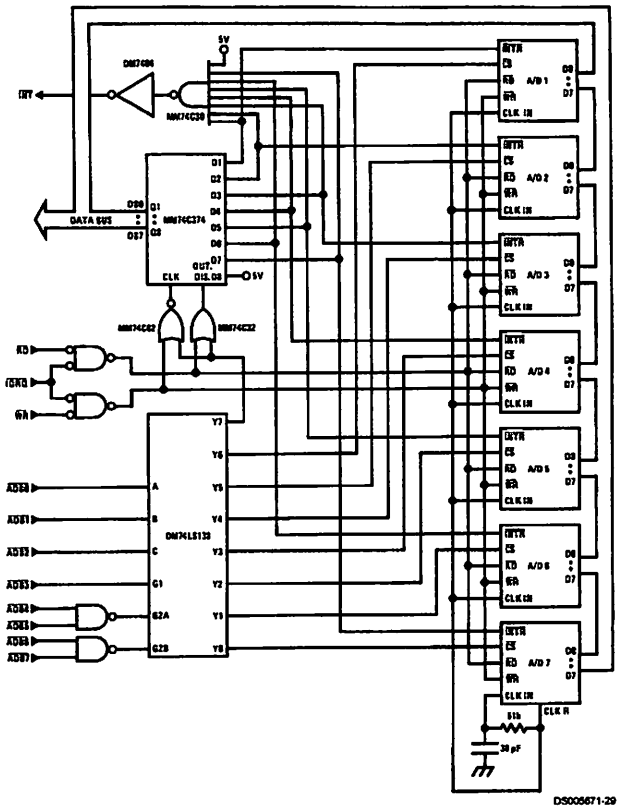


FIGURE 22. Multiple A/Ds with Z-80 Type Microprocessor

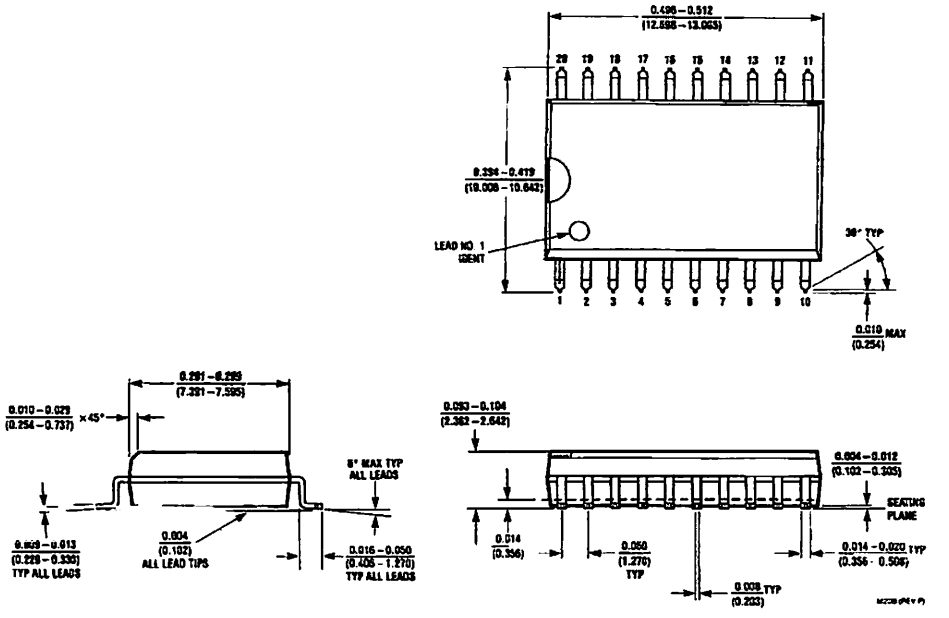
Functional Description (Continued)

INTERRUPT SERVICING SUBROUTINE

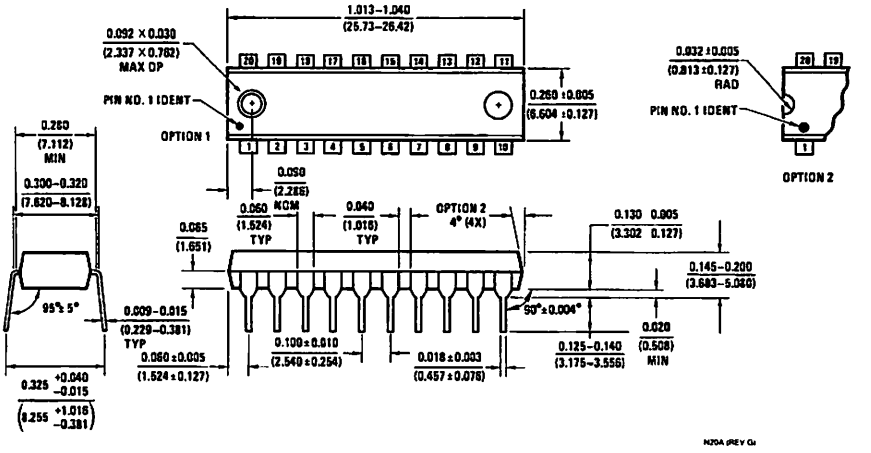
LOC	OBJ CODE		SOURCE STATEMENT	COMMENT
0038	E5		PUSH HL	; Save contents of all registers affected by
0039	C5		PUSH BC	; this subroutine.
003A	F5		PUSH AF	; Assumed INT mode 1 earlier set.
003B	21 00 3E		LD (HL),X3E00	; Initialize memory pointer where data will be stored.
003E	0E 01		LD C, X01	; C register will be port ADDR of A/D converters.
0040	D300		OUT X00, A	; Load peripheral status word into 8-bit latch.
0042	DB00		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	FE 08		CP, X08	; been checked. If so, exit subroutine
0048	CA 60 00		JPZ, 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 $\overline{INT}$
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	C3 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 FF		XOR FF	; the data.
0059	77		LD (HL),A	; Store the data
005A	2C		INC L	
005B	71		LD (HL),C	; Store A/D identifier (A/D port ADDR).
005C	2C		INC L	
005D	C3 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	E1		POP HL	
0063	C9		RET	; Return to original program

DS005671-A6

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: 1-800-272-9959
Fax: 1-800-737-7018
Email: support@nsc.com

www.national.com

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

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

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

DATA SHEET

Eeprom
(AT24C04)

tures

w-voltage and Standard-voltage Operation
- **2.7 (V_{CC} = 2.7V to 5.5V)**
- **1.8 (V_{CC} = 1.8V to 5.5V)**

rnally Organized 128 x 8 (1K), 256 x 8 (2K), 512 x 8 (4K), 4 x 8 (8K) or 2048 x 8 (16K)
o-wire Serial Interface
mitt Trigger, Filtered Inputs for Noise Suppression
irectional Data Transfer Protocol
kHz (1.8V) and 400 kHz (2.7V, 5V) Compatibility
te Protect Pin for Hardware Data Protection
yte Page (1K, 2K), 16-byte Page (4K, 8K, 16K) Write Modes
tial Page Writes Allowed
f-timed Write Cycle (5 ms max)
h-reliability
- **Endurance: 1 Million Write Cycles**
- **Data Retention: 100 Years**

tomotive Devices Available
ad JEDEC PDIP, 8-lead JEDEC SOIC, 8-lead Ultra Thin Mini-MAP (MLP 2x3), 5-lead T23, 8-lead TSSOP and 8-ball dBGA2 Packages
- **Sales: Wafer Form, Waffle Pack and Bumped Wafers**

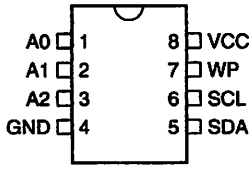
scription

AT24C01A/02/04/08A/16A provides 1024/2048/4096/8192/16384 bits of serial rically erasable and programmable read-only memory (EEPROM) organized as 256/512/1024/2048 words of 8 bits each. The device is optimized for use in many strial and commercial applications where low-power and low-voltage operation ssential. The AT24C01A/02/04/08A/16A is available in space-saving 8-lead , 8-lead JEDEC SOIC, 8-lead Ultra Thin Mini-MAP (MLP 2x3), 5-lead SOT23 4C01A/AT24C02/AT24C04), 8-lead TSSOP, and 8-ball dBGA2 packages and is ssed via a Two-wire serial interface. In addition, the entire family is available in (2.7V to 5.5V) and 1.8V (1.8V to 5.5V) versions.

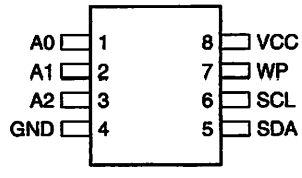
e 1. Pin Configuration

Name	Function
A2	Address Inputs
	Serial Data
	Serial Clock Input
	Write Protect
	No Connect
	Ground
	Power Supply

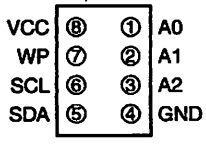
8-lead TSSOP



8-lead SOIC

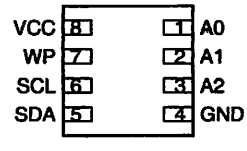


8-ball dBGA2

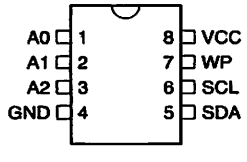


Bottom View

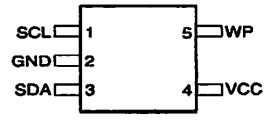
8-lead Ultra Thin Mini-MAP (MLP 2x3)



8-lead PDIP



Bottom View
5-lead SOT23



Two-wire Serial EEPROM

1K (128 x 8)

2K (256 x 8)

4K (512 x 8)

8K (1024 x 8)

16K (2048 x 8)

AT24C01A[1]
AT24C02[2]
AT24C04
AT24C08A
AT24C16A[3]

Notes:
1. Not Recommended for new design; Please refer to AT24C01B datasheet.
2. Not Recommended for new design; Please refer to AT24C02B datasheet.
3. Not Recommended for new design; Please refer to AT24C16B datasheet

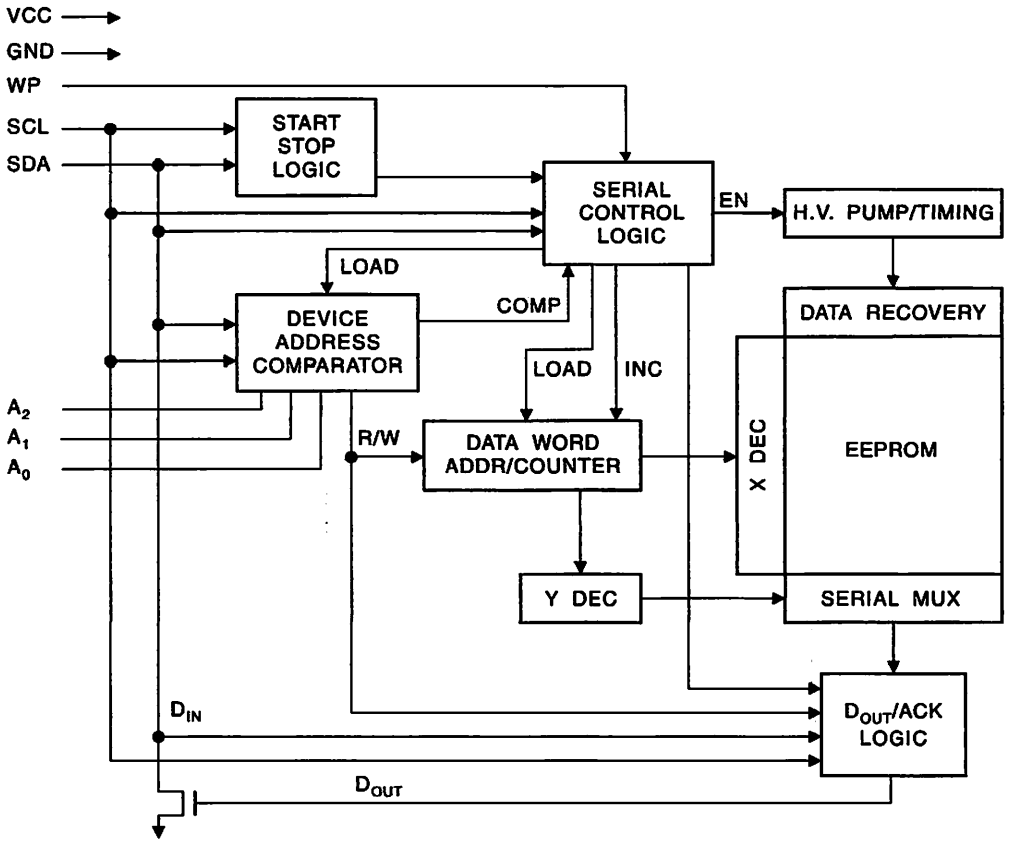
0180Z1–SEEPR–5/07

solute Maximum Ratings

rating Temperature	–55°C to +125°C
age Temperature	–65°C to +150°C
age on Any Pin Respect to Ground	–1.0V to +7.0V
imum Operating Voltage	6.25V
Output Current	5.0 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.

re 1. Block Diagram



Description

SERIAL CLOCK (SCL): The SCL input is used to positive edge clock data into each EEPROM device and negative edge clock data out of each device.

SERIAL DATA (SDA): The SDA pin is bidirectional for serial data transfer. This pin is open-drain driven and may be wire-ORed with any number of other open-drain or open-collector devices.

DEVICE/PAGE ADDRESSES (A2, A1, A0): The A2, A1 and A0 pins are device address inputs that are hard wired for the AT24C01A and the AT24C02. As many as eight 1K/2K devices may be addressed on a single bus system (device addressing is discussed in detail under the Device Addressing section).

The AT24C04 uses the A2 and A1 inputs for hard wire addressing and a total of four 4K devices may be addressed on a single bus system. The A0 pin is a no connect and can be connected to ground.

The AT24C08A only uses the A2 input for hardwire addressing and a total of two 8K devices may be addressed on a single bus system. The A0 and A1 pins are no connects and can be connected to ground.

The AT24C16A does not use the device address pins, which limits the number of devices on a single bus to one. The A0, A1 and A2 pins are no connects and can be connected to ground.

WRITE PROTECT (WP): The AT24C01A/02/04/08A/16A has a Write Protect pin that provides hardware data protection. The Write Protect pin allows normal Read/Write operations when connected to ground (GND). When the Write Protect pin is connected to V_{CC}, the write protection feature is enabled and operates as shown in Table 2.

Table 2. Write Protect

WP Pin Status	Part of the Array Protected				
	24C01A	24C02	24C04	24C08A	24C16A
At V_{CC}	Full (1K) Array	Full (2K) Array	Full (4K) Array	Full (8K) Array	Full (16K) Array
At GND	Normal Read/Write Operations				

nory Organization

AT24C01A, 1K SERIAL EEPROM: Internally organized with 16 pages of 8 bytes each, the 1K requires a 7-bit data word address for random word addressing.

AT24C02, 2K SERIAL EEPROM: Internally organized with 32 pages of 8 bytes each, the 2K requires an 8-bit data word address for random word addressing.

AT24C04, 4K SERIAL EEPROM: Internally organized with 32 pages of 16 bytes each, the 4K requires a 9-bit data word address for random word addressing.

AT24C08A, 8K SERIAL EEPROM: Internally organized with 64 pages of 16 bytes each, the 8K requires a 10-bit data word address for random word addressing.

AT24C16A, 16K SERIAL EEPROM: Internally organized with 128 pages of 16 bytes each, the 16K requires an 11-bit data word address for random word addressing.

3. Pin Capacitance[1]

cable over recommended operating range from T_A = 25°C, f = 1.0 MHz, V_{CC} = +1.8V

bol	Test Condition	Max	Units	Conditions
	Input/Output Capacitance (SDA)	8	pF	$V_{I/O}$ = 0V
	Input Capacitance (A_0, A_1, A_2, SCL)	6	pF	V_{IN} = 0V

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

4. DC Characteristics

cable over recommended operating range from: T_{AI} = –40°C to +85°C, V_{CC} = +1.8V to +5.5V, V_{CC} = +1.8V to +5.5V ss otherwise noted)

bol	Parameter	Test Condition	Min	Typ	Max	Units
	Supply Voltage		1.8		5.5	V
	Supply Voltage		2.7		5.5	V
	Supply Voltage		4.5		5.5	V
	Supply Current V_{CC} = 5.0V	READ at 100 kHz		0.4	1.0	mA
	Supply Current V_{CC} = 5.0V	WRITE at 100 kHz		2.0	3.0	mA
	Standby Current V_{CC} = 1.8V	V_{IN} = V_{CC} or V_{SS}		0.6	3.0	µA
	Standby Current V_{CC} = 2.5V	V_{IN} = V_{CC} or V_{SS}		1.4	4.0	µA
	Standby Current V_{CC} = 2.7V	V_{IN} = V_{CC} or V_{SS}		1.6	4.0	µA
	Standby Current V_{CC} = 5.0V	V_{IN} = V_{CC} or V_{SS}		8.0	18.0	µA
	Input Leakage Current	V_{IN} = V_{CC} or V_{SS}		0.10	3.0	µA
	Output Leakage Current	V_{OUT} = V_{CC} or V_{SS}		0.05	3.0	µA
	Input Low Level[1]		–0.6		V_{CC} x 0.3	V
	Input High Level[1]		V_{CC} x 0.7		V_{CC} + 0.5	V
	Output Low Level V_{CC} = 3.0V	I_{OL} = 2.1 mA			0.4	V
	Output Low Level V_{CC} = 1.8V	I_{OL} = 0.15 mA			0.2	V

1. V_{IL} min and V_{IH} max are reference only and are not tested.

5. AC Characteristics

cable over recommended operating range from T_{AI} = –40°C to +85°C, V_{CC} = +1.8V to +5.5V, V_{CC} = +2.7V to +5.5V, 1 TTL Gate and 100 pF (unless otherwise noted)

ıbol	Parameter	1.8-volt		2.7, 5.0-volt		Units
		Min	Max	Min	Max	
	Clock Frequency, SCL		100		400	kHz
	Clock Pulse Width Low	4.7		1.2		µs
	Clock Pulse Width High	4.0		0.6		µs
	Noise Suppression Time[1]		100		50	ns
	Clock Low to Data Out Valid	0.1	4.5	0.1	0.9	µs
	Time the bus must be free before a new transmission can start[1]	4.7		1.2		µs
TA	Start Hold Time	4.0		0.6		µs
TA	Start Setup Time	4.7		0.6		µs
AT	Data In Hold Time	0		0		µs
AT	Data In Setup Time	200		100		ns
	Inputs Rise Time[1]		1.0		0.3	µs
	Inputs Fall Time[1]		300		300	ns
TO	Stop Setup Time	4.7		0.6		µs
	Data Out Hold Time	100		50		ns
	Write Cycle Time		5		5	ms
urance[1]	5.0V, 25°C, Byte Mode	1M		1M		Write Cycles

1. This parameter is characterized.

rice Operation

CLOCK and DATA TRANSITIONS: The SDA pin is normally pulled high with an external device. Data on the SDA pin may change only during SCL low time periods (see Figure 4 on page 7). Data changes during SCL high periods will indicate a start or stop condition as defined below.

START CONDITION: A high-to-low transition of SDA with SCL high is a start condition which must precede any other command (see Figure 5 on page 8).

STOP CONDITION: A low-to-high transition of SDA with SCL high is a stop condition. After a read sequence, the stop command will place the EEPROM in a standby power mode (see Figure 5 on page 8).

ACKNOWLEDGE: All addresses and data words are serially transmitted to and from the EEPROM in 8-bit words. The EEPROM sends a zero to acknowledge that it has received each word. This happens during the ninth clock cycle.

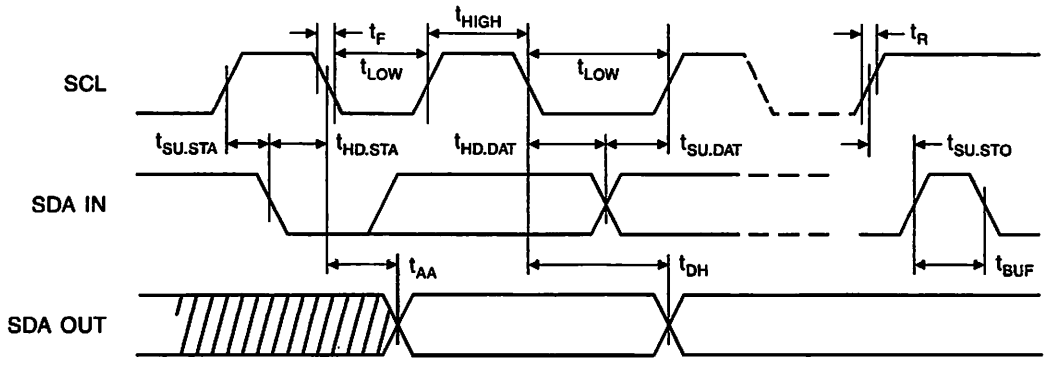
STANDBY MODE: The AT24C01A/02/04/08A/16A features a low-power standby mode which is enabled: (a) upon power-up and (b) after the receipt of the STOP bit and the completion of any internal operations.

MEMORY RESET: After an interruption in protocol, power loss or system reset, any 2-wire part can be reset by following these steps:

1. Clock up to 9 cycles.
2. Look for SDA high in each cycle while SCL is high.
3. Create a start condition.

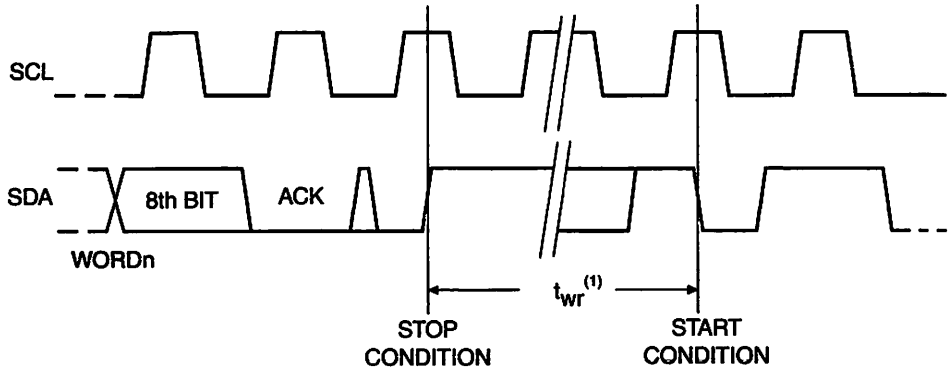
s Timing

re 2. SCL: Serial Clock, SDA: Serial Data I/O®



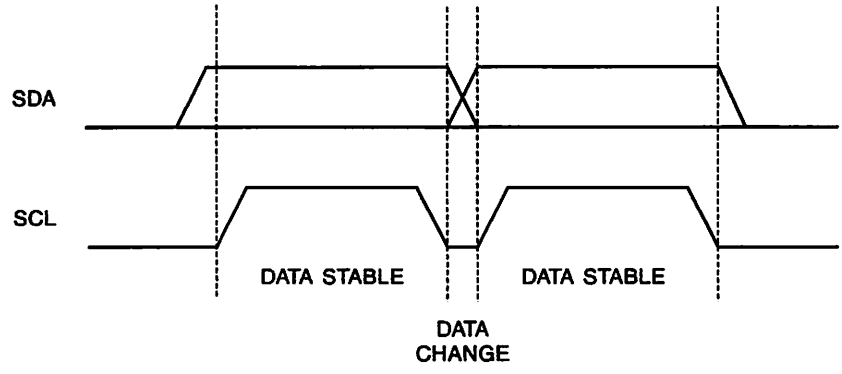
te Cycle Timing

re 3. SCL: Serial Clock, SDA: Serial Data I/O

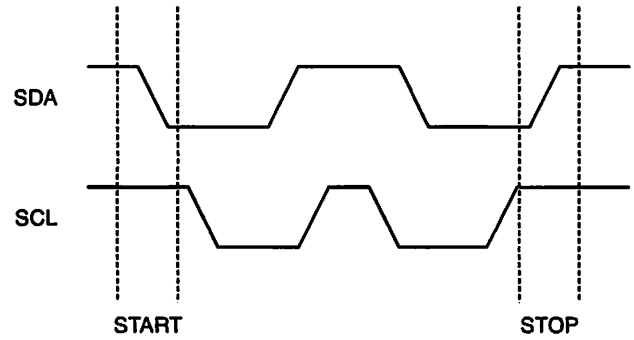


1. The write cycle time t_{WR} is the time from a valid stop condition of a write sequence to the end of the internal clear/write cycle.

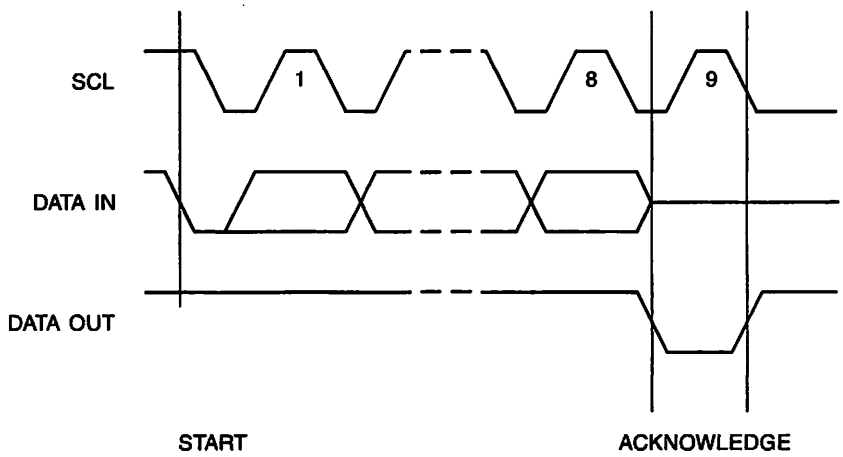
re 4. Data Validity



re 5. Start and Stop Definition



re 6. Output Acknowledge



/ice Addressing

The 1K, 2K, 4K, 8K and 16K EEPROM devices all require an 8-bit device address word following a start condition to enable the chip for a read or write operation (refer to Figure 7).

The device address word consists of a mandatory one, zero sequence for the first four most significant bits as shown. This is common to all the EEPROM devices.

The next 3 bits are the A2, A1 and A0 device address bits for the 1K/2K EEPROM. These 3 bits must compare to their corresponding hard-wired input pins.

The 4K EEPROM only uses the A2 and A1 device address bits with the third bit being a memory page address bit. The two device address bits must compare to their corresponding hard-wired input pins. The A0 pin is no connect.

The 8K EEPROM only uses the A2 device address bit with the next 2 bits being for memory page addressing. The A2 bit must compare to its corresponding hard-wired input pin. The A1 and A0 pins are no connect.

The 16K does not use any device address bits but instead the 3 bits are used for memory page addressing. These page addressing bits on the 4K, 8K and 16K devices should be considered the most significant bits of the data word address which follows. The A0, A1 and A2 pins are no connect.

The eighth bit of the device address is the read/write operation select bit. A read operation is initiated if this bit is high and a write operation is initiated if this bit is low.

Upon a compare of the device address, the EEPROM will output a zero. If a compare is not made, the chip will return to a standby state.

te Operations

BYTE WRITE: A write operation requires an 8-bit data word address following the device address word and acknowledgment. Upon receipt of this address, the EEPROM will again respond with a zero and then clock in the first 8-bit data word. Following receipt of the 8-bit data word, the EEPROM will output a zero and the addressing device, such as a microcontroller, must terminate the write sequence with a stop condition. At this time the EEPROM enters an internally timed write cycle, t_{WR}, to the nonvolatile memory. All inputs are disabled during this write cycle and the EEPROM will not respond until the write is complete (see Figure 8 on page 11).

PAGE WRITE: The 1K/2K EEPROM is capable of an 8-byte page write, and the 4K, 8K and 16K devices are capable of 16-byte page writes.

A page write is initiated the same as a byte write, but the microcontroller does not send a stop condition after the first data word is clocked in. Instead, after the EEPROM acknowledges receipt of the first data word, the microcontroller can transmit up to seven (1K/2K) or fifteen (4K, 8K, 16K) more data words. The EEPROM will respond with a zero after each data word received. The microcontroller must terminate the page write sequence with a stop condition (see Figure 9 on page 11).

The data word address lower three (1K/2K) or four (4K, 8K, 16K) bits are internally incremented following the receipt of each data word. The higher data word address bits are not incremented, retaining the memory page row location. When the word address, internally generated, reaches the page boundary, the following byte is placed at the beginning of the same page. If more than eight (1K/2K) or sixteen (4K, 8K, 16K) data words are transmitted to the EEPROM, the data word address will "roll over" and previous data will be overwritten.

ACKNOWLEDGE POLLING: Once the internally timed write cycle has started and the EEPROM inputs are disabled, acknowledge polling can be initiated. This involves sending a start condition followed by the device address word. The read/write bit is representative of the operation desired. Only if the internal write cycle has completed will the EEPROM respond with a zero allowing the read or write sequence to continue.

ıd ərations

Read operations are initiated the same way as write operations with the exception that the read/write select bit in the device address word is set to one. There are three read operations: current address read, random address read and sequential read.

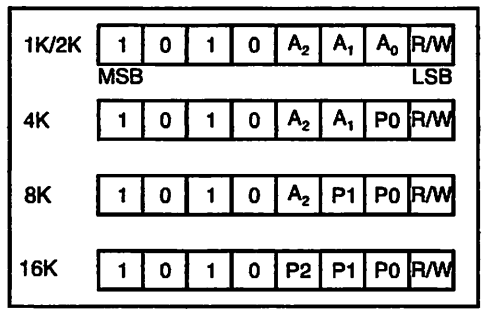
CURRENT ADDRESS READ: The internal data word address counter maintains the last address accessed during the last read or write operation, incremented by one. This address stays valid between operations as long as the chip power is maintained. The address "roll over" during read is from the last byte of the last memory page to the first byte of the first page. The address "roll over" during write is from the last byte of the current page to the first byte of the same page.

Once the device address with the read/write select bit set to one is clocked in and acknowledged by the EEPROM, the current address data word is serially clocked out. The microcontroller does not respond with an input zero but does generate a following stop condition (see Figure 10 on page 12).

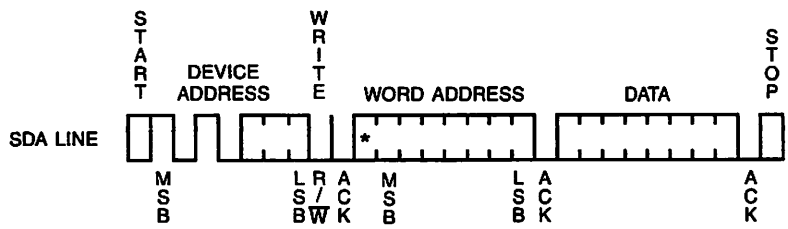
RANDOM READ: A random read requires a "dummy" byte write sequence to load in the data word address. Once the device address word and data word address are clocked in and acknowledged by the EEPROM, the microcontroller must generate another start condition. The microcontroller now initiates a current address read by sending a device address with the read/write select bit high. The EEPROM acknowledges the device address and serially clocks out the data word. The microcontroller does not respond with a zero but does generate a following stop condition (see Figure 11 on page 12).

SEQUENTIAL READ: Sequential reads are initiated by either a current address read or a random address read. After the microcontroller receives a data word, it responds with an acknowledge. As long as the EEPROM receives an acknowledge, it will continue to increment the data word address and serially clock out sequential data words. When the memory address limit is reached, the data word address will "roll over" and the sequential read will continue. The sequential read operation is terminated when the microcontroller does not respond with a zero but does generate a following stop condition (see Figure 12 on page 12).

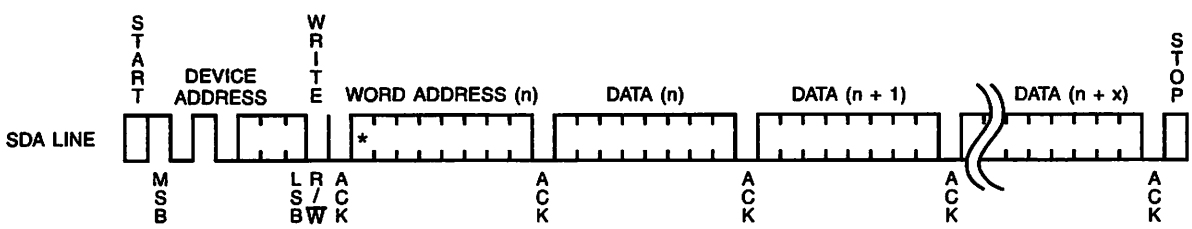
re 7. Device Address



re 8. Byte Write

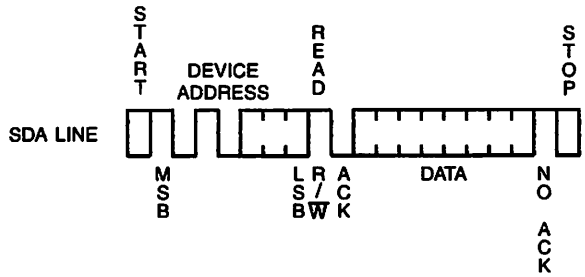


re 9. Page Write

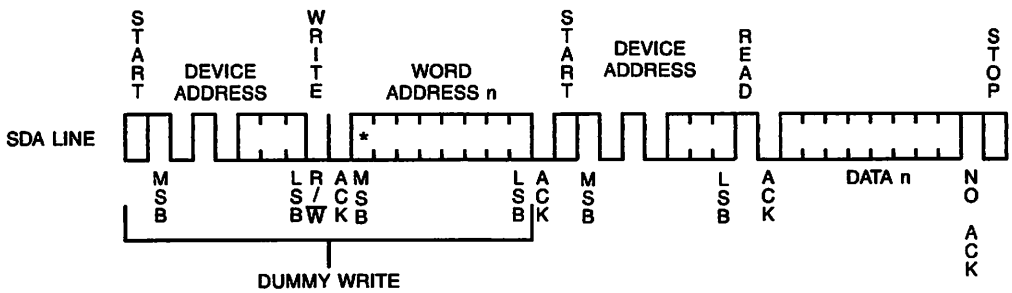


(* = DON'T CARE bit for 1K)

re 10. Current Address Read

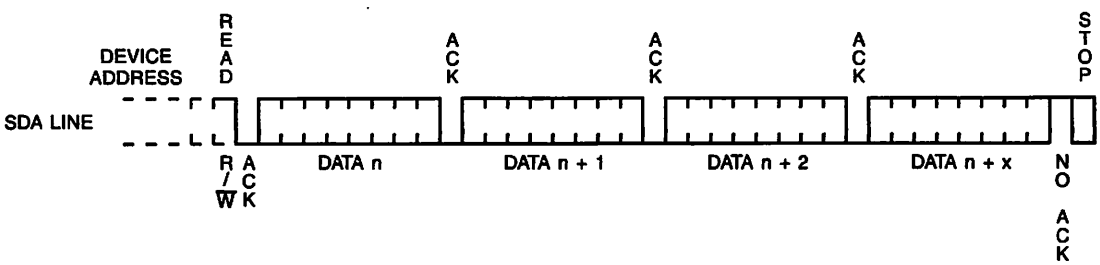


re 11. Random Read



(* = DON'T CARE bit for 1K)

re 12. Sequential Read



24C01A Ordering Information[1]

Ordering Code	Package	Operation Range
AT24C01A-10PU-2.7[2]	8P3	Lead-free/Halogen-free/ Industrial Temperature (–40°C to 85°C)
AT24C01A-10PU-1.8[2]	8P3	
AT24C01A-10SU-2.7[2]	8S1	
AT24C01A-10SU-1.8[2]	8S1	
AT24C01A-10TU-2.7[2]	8A2	
AT24C01A-10TU-1.8[2]	8A2	
AT24C01A-10TSU-1.8[2]	5TS1	
AT24C01AU3-10UU-1.8[2]	8U31	
AT24C01AY1-10YU-1.8[2] (Not recommended for new design)	8Y1	
AT24C01AY6-10YH-1.8[3]	8Y6	
AT24C01A-W1.8-11[4]	Die Sale	Industrial Temperature (–40°C to 85°C)

s: 1. This device is not recommended for new design. Please refer to AT24C01B datasheet. For 2.7V devices used in the 4.5V to 5.5V range, please refer to performance values in the AC and DC characteristics table.
2. "U" designates Green Package + RoHS compliant.
3. "H" designates Green Package + RoHS compliant, with NiPdAu Lead Finish.
4. Available in waffle pack and wafer form; order as SL788 for inkless wafer form. Bumped die available upon request. Please contact Serial EEPROM Marketing.

	Package Type
	8-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
	8-lead, 0.150" Wide, Plastic Gull Wing Small Outline (JEDEC SOIC)
	8-lead, 4.4 mm Body, Plastic Thin Shrink Small Outline Package (TSSOP)
	8-lead, 4.90 mm x 3.00 mm Body, Dual Footprint, Non-leaded, Miniature Array Package (MAP)
	8-lead, 2.00 x 3.00 mm Body, 0.50 mm Pitch, Ultra Thin Mini-MAP, Dual No Lead Package (DFN), (MLP 2x3 mm)
1	5-lead, 2.90 mm x 1.60 mm Body, Plastic Thin Shrink Small Outline Package (SOT23)
-1	8-ball, die Ball Grid Away Package (dBGA2)
	Options
	Low-voltage (2.7V to 5.5V)
	Low-voltage (1.8V to 5.5V)

24C02 Ordering Information[1]

Ordering Code	Package	Operation Range
AT24C02-10PU-2.7[2]	8P3	Lead-free/Halogen-free/ Industrial Temperature (–40°C to 85°C)
AT24C02-10PU-1.8[2]	8P3	
AT24C02N-10SU-2.7[2]	8S1	
AT24C02N-10SU-1.8[2]	8S1	
AT24C02-10TU-2.7[2]	8A2	
AT24C02-10TU-1.8[2]	8A2	
AT24C02Y1-10YU-1.8[2]	8Y1	
AT24C02-10TSU-1.8[2]	5TS1	
AT24C02U3-10UU-1.8[2]	8U3-1	
AT24C02-W2.7-11[3]	Die Sale	Industrial Temperature (–40°C to 85°C)

1. This device is not recommended for new design. Please refer to AT24C02B datasheet. For 2.7V devices used in the 4.5V to 5.5V range, please refer to performance values in the AC and DC characteristics table.
2. "U" designates Green Package + RoHS compliant.
3. Available in waffle pack and wafer form; order as SL719 for wafer form. Bumped die available upon request. Please contact Serial EEPROM Marketing.

Package Type	
	8-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
	8-lead, 0.150" Wide, Plastic Gull Wing Small Outline (JEDEC SOIC)
	8-lead, 4.4 mm Body, Plastic Thin Shrink Small Outline Package (TSSOP)
	8-lead, 4.90 mm x 3.00 mm Body, Dual Footprint, Non-leaded, Miniature Array Package (MAP)
1	5-lead, 2.90 mm x 1.60 mm Body, Plastic Thin Shrink Small Outline Package (SOT23)
-1	8-ball, die Ball Grid Away Package (dBGA2)
Options	
	Low-voltage (2.7V to 5.5V)
	Low-voltage (1.8V to 5.5V)

?4C04 Ordering Information[1]

Ordering Code	Package	Operation Range
AT24C04-10PU-2.7[2]	8P3	Lead-free/Halogen-free/ Industrial Temperature (–40°C to 85°C)
AT24C04-10PU-1.8[2]	8P3	
AT24C04N-10SU-2.7[2]	8S1	
AT24C04N-10SU-1.8[2]	8S1	
AT24C04-10TU-2.7[2]	8A2	
AT24C04-10TU-1.8[2]	8A2	
AT24C04Y1-10YU-1.8[2] (Not recommended for new design)	8Y1	
AT24C04Y6-10YH-1.8[3]	8Y6	
AT24C04-10TSU-1.8[2]	5TS1	
AT24C04U3-10UU-1.8[2]	8U3-1	
AT24C04-W1.8-11[4]	Die Sale	Industrial Temperature (–40°C to 85°C)

1. For 2.7V devices used in the 4.5V to 5.5V range, please refer to performance values in the AC and DC characteristics table.
2. "U" designates Green Package + RoHS compliant.
3. "H" designates Green Package + RoHS compliant, with NiPdAu Lead Finish.
4. Available in waffle pack and wafer form; order as SL788 for inkless wafer form. Bumped die available upon request. Please contact Serial EEPROM Marketing.

	Package Type
	8-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
	8-lead, 0.150" Wide, Plastic Gull Wing Small Outline (JEDEC SOIC)
	8-lead, 4.4 mm Body, Plastic Thin Shrink Small Outline Package (TSSOP)
	8-lead, 4.90 mm x 3.00 mm Body, Dual Footprint, Non-leaded, Miniature Array Package (MAP)
	8-lead, 2.00 x 3.00 mm Body, 0.50 mm Pitch, Ultra Thin Mini-MAP, Dual No Lead Package (DFN), (MLP 2x3 mm)
1	5-lead, 2.90 mm x 1.60 mm Body, Plastic Thin Shrink Small Outline Package (SOT23)
-1	8-ball, die Ball Grid Away Package (dBGA2)
	Options
	Low-voltage (2.7V to 5.5V)
	Low-voltage (1.8V to 5.5V)

?4C08A Ordering Information[1]

Ordering Code	Package	Operation Range
AT24C08A-10PU-2.7[2]	8P3	Lead-free/Halogen-free/ Industrial Temperature (–40°C to 85°C)
AT24C08A-10PU-1.8[2]	8P3	
AT24C08AN-10SU-2.7[2]	8S1	
AT24C08AN-10SU-1.8[2]	8S1	
AT24C08A-10TU-2.7[2]	8A2	
AT24C08A-10TU-1.8[2]	8A2	
AT24C08AY1-10YU-1.8[2] (Not recommended for new design)	8Y1	
AT24C08AY6-10YH-1.8[3]	8Y6	
AT24C08AU2-10UU-1.8[2	8U2-1	
AT24C08A-W1.8-11[4]	Die Sale	Industrial Temperature (–40°C to 85°C)

1. For 2.7V devices used in the 4.5V to 5.5V range, please refer to performance values in the AC and DC characteristics table.
2. "U" designates Green Package + RoHS compliant.
3. "H" designates Green Package + RoHS compliant, with NiPdAu Lead Finish.
4. Available in waffle pack and wafer form; order as SL788 for inkless wafer form. Bumped die available upon request. Please contact Serial EEPROM Marketing.

	Package Type
	8-pin, 0.300" Wide, Plastic Dual Inline Package (PDIP)
	8-lead, 0.150" Wide, Plastic Gull Wing Small Outline (JEDEC SOIC)
	8-lead, 4.4 mm Body, Plastic Thin Shrink Small Outline Package (TSSOP)
	8-lead, 4.90 mm x 3.00 mm Body, Dual Footprint, Non-leaded, Miniature Array Package (MAP)
	8-lead, 2.00 x 3.00 mm Body, 0.50 mm Pitch, Ultra Thin Mini-MAP, Dual No Lead Package (DFN), (MLP 2x3 mm)
-1	8-ball, die Ball Grid Array Package (dBGA2)
	Options
	Low Voltage (2.7V to 5.5V)
	Low Voltage (1.8V to 5.5V)

?4C16A Ordering Information[1]

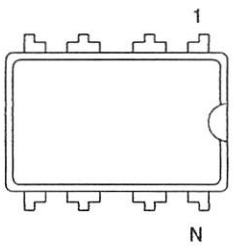
Ordering Code	Package	Operation Range
AT24C16A-10PU-2.7[2]	8P3	Lead-free/Halogen-free/ Industrial Temperature (–40°C to 85°C)
AT24C16A-10PU-1.8[2]	8P3	
AT24C16AN-10SU-2.7[2]	8S1	
AT24C16AN-10SU-1.8[2]	8S1	
AT24C16A-10TU-2.7[2]	8A2	
AT24C16A-10TU-1.8[2]	8A2	
AT24C16AY1-10YU-1.8[2] (Not recommended for new design)	8Y1	
AT24C16AY6-10YH-1.8[3]	8Y6	
AT24C16AU2-10UU-1.8[2]	8U2-1	
AT24C16A-W1.8-11[3]	Die Sale	Industrial Temperature (–40°C to 85°C)

1. This device is not recommended for new design. Please refer to AT24C16B datasheet. For 2.7V devices used in the 4.5V to 5.5V range, please refer to performance values in the AC and DC characteristics table.
2. "U" designates Green Package + RoHS compliant.
3. "H" designates Green Package + RoHS compliant, with NiPdAu Lead Finish.
4. Available in waffle pack and wafer form; order as SL788 for inkless wafer form. Bumped die available upon request. Please contact Serial EEPROM Marketing.

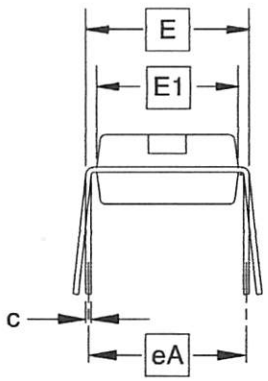
	Package Type
	8-pin, 0.300" Wide, Plastic Dual Inline Package (PDIP)
	8-lead, 0.150" Wide, Plastic Gull Wing Small Outline (JEDEC SOIC)
	8-lead, 0.170" Wide, Thin Shrink Small Outline Package (TSSOP)
	8-lead, 4.90 mm x 3.00 mm Body, Dual Footprint, Non-leaded, Miniature Array Package (MAP)
	8-lead, 2.00 x 3.00 mm Body, 0.50 mm Pitch, Ultra Thin Mini-MAP, Dual No Lead Package (DFN), (MLP 2x3 mm)
-1	8-ball, die Ball Grid Array Package (dBGA2)
	Options
	Low Voltage (2.7V to 5.5V)
	Low Voltage (1.8V to 5.5V)

ckaging Information

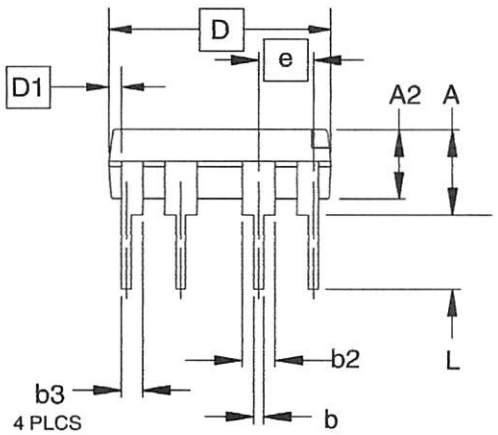
– PDIP



Top View



End View



Side View

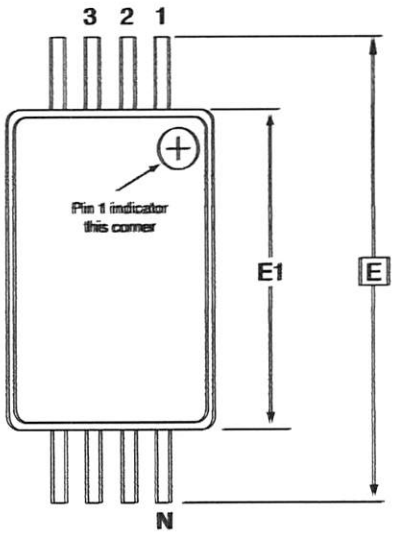
COMMON DIMENSIONS
(Unit of Measure = inches)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	0.210	2
A2	0.115	0.130	0.195	
b	0.014	0.018	0.022	5
b2	0.045	0.060	0.070	6
b3	0.030	0.039	0.045	6
c	0.008	0.010	0.014	
D	0.355	0.365	0.400	3
D1	0.005	–	–	3
E	0.300	0.310	0.325	4
E1	0.240	0.250	0.280	3
e	0.100 BSC			
eA	0.300 BSC			4
L	0.115	0.130	0.150	2

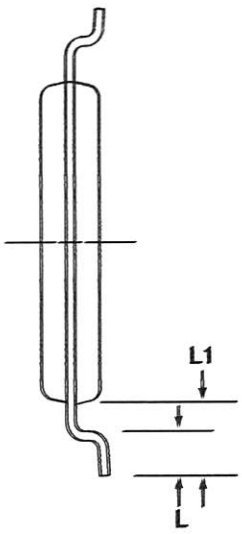
Notes:
1. This drawing is for general information only; refer to JEDEC Drawing MS-001, Variation BA, for additional information.
2. Dimensions A and L are measured with the package seated in JEDEC seating plane Gauge GS-3.
3. D, D1 and E1 dimensions do not include mold Flash or protrusions. Mold Flash or protrusions shall not exceed 0.010 inch.
4. E and eA measured with the leads constrained to be perpendicular to datum.
5. Pointed or rounded lead tips are preferred to ease insertion.
6. b2 and b3 maximum dimensions do not include Dambar protrusions. Dambar protrusions shall not exceed 0.010 (0.25 mm).

01/09/02

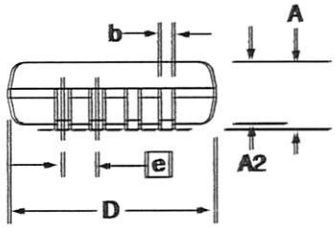
	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**8P3**, 8-lead, 0.300" Wide Body, Plastic Dual In-line Package (PDIP)	8P3	B



Top View



End View



Side View

COMMON DIMENSIONS
(Unit of Measure = mm)

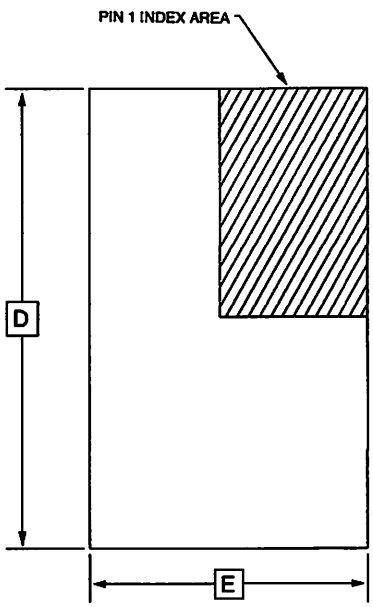
SYMBOL	MIN	NOM	MAX	NOTE
D	2.90	3.00	3.10	2, 5
E		6.40 BSC		
E1	4.30	4.40	4.50	3, 5
A	–	–	1.20	
A2	0.80	1.00	1.05	
b	0.19	–	0.30	4
e		0.65 BSC		
L	0.45	0.60	0.75	
L1		1.00 REF		

otes:
1. This drawing is for general information only. Refer to JEDEC Drawing MO-153, Variation AA, for proper dimensions, tolerances, datums, etc.
2. Dimension D does not include mold Flash, protrusions or gate burrs. Mold Flash, protrusions and gate burrs shall not exceed 0.15 mm ((0.006 in) per side.
3. Dimension E1 does not include inter-lead Flash or protrusions. Inter-lead Flash and protrusions shall not exceed 0.25 mm ((0.010 in) per side.
4. Dimension b does not include Dambar protrusion. Allowable Dambar protrusion shall be 0.08 mm total in excess of the b dimension at maximum material condition. Dambar cannot be located on the lower radius of the foot. Minimum space between protrusion and adjacent lead is 0.07 mm.
5. Dimension D and E1 to be determined at Datum Plane H.

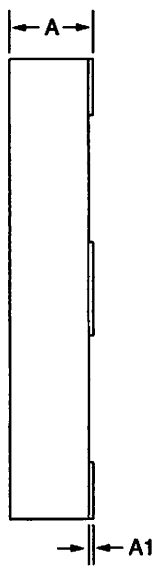
5/30/02

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**8A2**, 8-lead, 4.4 mm Body, Plastic Thin Shrink Small Outline Package (TSSOP)	8A2	B

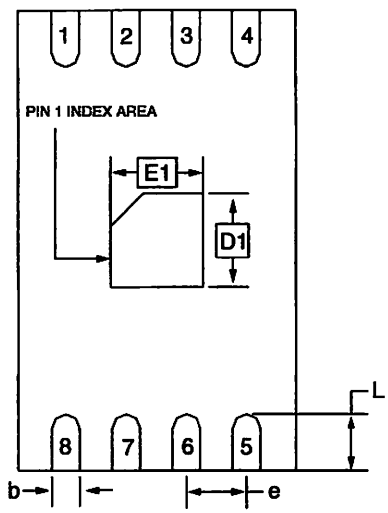
– MAP



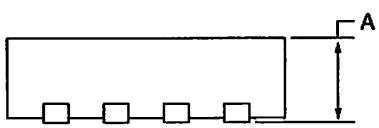
Top View



End View



Bottom View



Side View

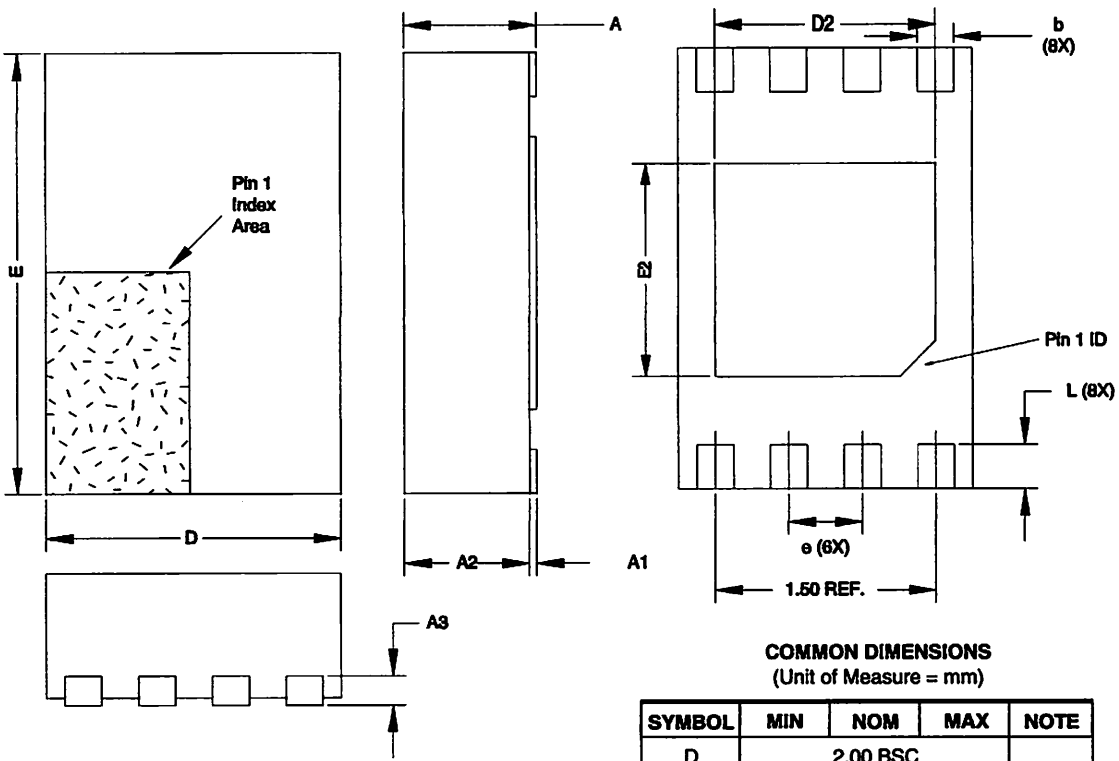
COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	0.90	
A1	0.00	–	0.05	
D	4.70	4.90	5.10	
E	2.80	3.00	3.20	
D1	0.85	1.00	1.15	
E1	0.85	1.00	1.15	
b	0.25	0.30	0.35	
e	0.65 TYP			
L	0.50	0.60	0.70	

2/28/03

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**8Y1,** 8-lead (4.90 x 3.00 mm Body) MSOP Array Package (MAP) Y1	8Y1	C

– Mini-MAP (MLP 2x3 mm)



COMMON DIMENSIONS
(Unit of Measure = mm)

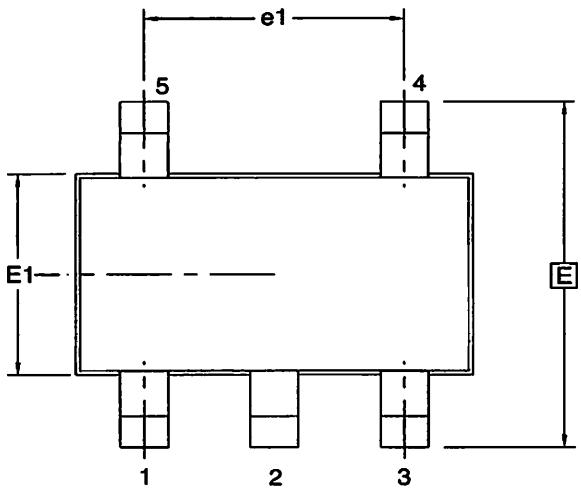
SYMBOL	MIN	NOM	MAX	NOTE
D		2.00 BSC		
E		3.00 BSC		
D2	1.40	1.50	1.60	
E2	-	-	1.40	
A	-	-	0.60	
A1	0.0	0.02	0.05	
A2	-	-	0.55	
A3		0.20 REF		
L	0.20	0.30	0.40	
e		0.50 BSC		
b	0.20	0.25	0.30	2

lotes:
1. This drawing is for general information only. Refer to JEDEC Drawing MO-229, for proper dimensions, tolerances, datums, etc.
2. Dimension b applies to metallized terminal and is measured between 0.15 mm and 0.30 mm from the terminal tip. If the terminal has the optional radius on the other end of the terminal, the dimension should not be measured in that radius area.

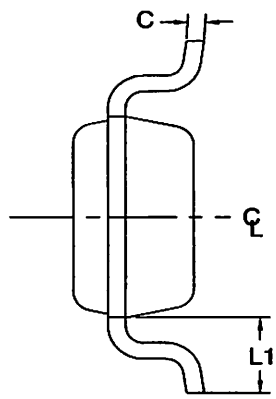
8/26/05

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**8Y6**, 8-lead 2.0 x 3.0 mm Body, 0.50 mm Pitch, Ultra Thin Mini-Map, Dual No Lead Package (DFN) ,(MLP 2x3)	8Y6	C

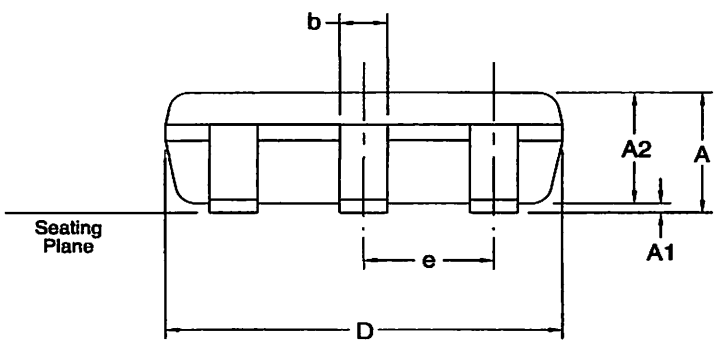
51 – SOT23



Top View



End View



Side View

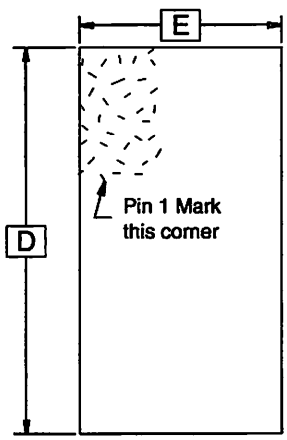
COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.10	
A1	0.00	–	0.10	
A2	0.70	0.90	1.00	
c	0.08	–	0.20	4
D		2.90 BSC		2, 3
E		2.80 BSC		2, 3
E1		1.60 BSC		2, 3
L1		0.60 REF		
e		0.95 BSC		
e1		1.90 BSC		
b	0.30	–	0.50	4, 5

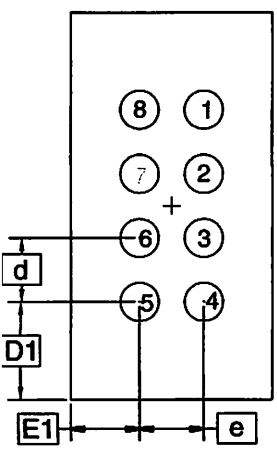
NOTES:
1. This drawing is for general information only. Refer to JEDEC Drawing MO-193, Variation AB, for additional information.
2. Dimension D does not include mold flash, protrusions, or gate burrs. Mold flash, protrusions, or gate burrs shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion. Interlead flash or protrusion shall not exceed 0.15 mm per side.
3. The package top may be smaller than the package bottom. Dimensions D and E1 are determined at the outermost extremes of the plastic body exclusive of mold flash, tie bar burrs, gate burrs, and interlead flash, but including any mismatch between the top and bottom of the plastic body.
4. These dimensions apply to the flat section of the lead between 0.08 mm and 0.15 mm from the lead tip.
5. Dimension "b" does not include Dambar protrusion. Allowable Dambar protrusion shall be 0.08 mm total in excess of the "b" dimension at maximum material condition. The Dambar cannot be located on the lower radius of the foot. Minimum space between protrusion and an adjacent lead shall not be less than 0.07 mm.

6/25/03

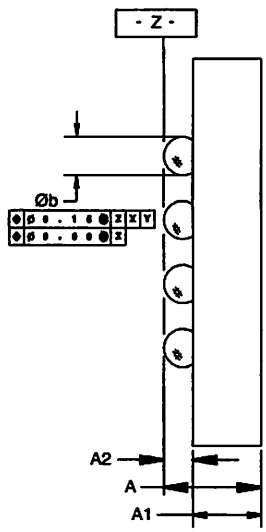
	TITLE	DRAWING NO.	REV.
1150 E. Cheyenne Mtn. Blvd. Colorado Springs, CO 80906	**5TS1**, 5-lead, 1.60 mm Body, Plastic Thin Shrink Small Outline Package (SHRINK SOT)	PO5TS1	A



Top View



Bottom View



Side View

COMMON DIMENSIONS
(Unit of Measure = mm)

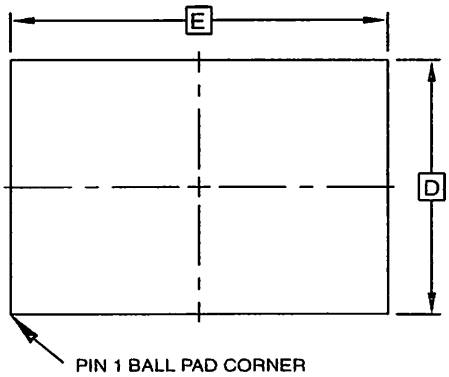
SYMBOL	MIN	NOM	MAX	NOTE
D			5.10	
D1		1.43 TYP		
E			3.25	
E1		1.25 TYP		
e		0.75 TYP		
d		0.75 TYP		
A		0.90 REF		
A1	0.49	0.52	0.55	
A2	0.35	0.38	0.41	
øb	0.47	0.50	0.53	

Notes: 1. These drawings are for general information only. No JEDEC Drawing to refer to for additional information.
2. Dimension is measured at the maximum solder ball diameter, parallel to primary datum Z.

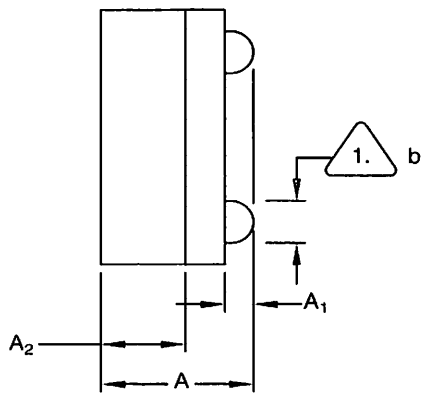
02/04/02

	TITLE	DRAWING NO.	REV.
1150 E. Cheyenne Mtn. Blvd. Colorado Springs, CO 80906	**8U2**, 8-ball 0.75 pitch, Die Ball Grid Array Package (dBGA) AT24C512 (AT19870)	8U2	A

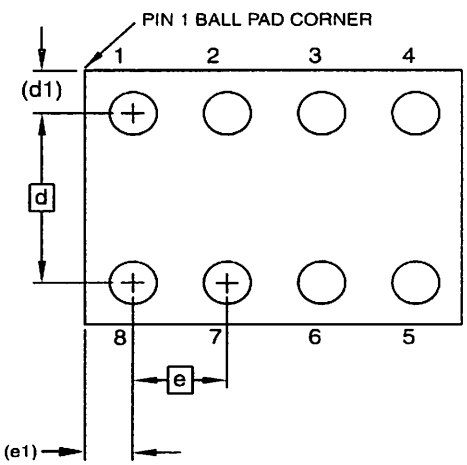
-1 – dBGA2



Top View



Side View



Bottom View
8 SOLDER BALLS

COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	0.71	0.81	0.91	
A1	0.10	0.15	0.20	
A2	0.40	0.45	0.50	
b	0.20	0.25	0.30	
D	1.50 BSC			
E	2.00 BSC			
e	0.50 BSC			
e1	0.25 REF			
d	1.00 BSC			
d1	0.25 REF			

1. Dimension "b" is measured at the maximum solder ball diameter.

This drawing is for general information only.

6/24/03

	TITLE	DRAWING NO.	REV.
1150 E. Cheyenne Mtn. Blvd. Colorado Springs, CO 80906	**8U3-1**, 8-ball, 1.50 x 2.00 mm Body, 0.50 mm pitch, Small Die Ball Grid Array Package (dBGA2)	PO8U3-1	A

ision History

Doc. No.	Date	Comments
0180Z1	5/2007	Implemented revision history. Changed formatting on page 16

nel Corporation

325 Orchard Parkway
an Jose, CA 95131, USA
el: 1(408) 441-0311
ax: 1(408) 487-2600

gional Headquarters

ope
tmel Sarl
oute des Arsenaux 41
ase Postale 80
H-1705 Fribourg
witzerland
el: (41) 26-426-5555
ax: (41) 26-426-5500

oom 1219
hinachem Golden Plaza
7 Mody Road Tsimshatsui
ast Kowloon
ong Kong
el: (852) 2721-9778
ax: (852) 2722-1369

an
F, Tonetsu Shinkawa Bldg.
24-8 Shinkawa
huo-ku, Tokyo 104-0033
apan
el: (81) 3-3523-3551
ax: (81) 3-3523-7581

Atmel Operations

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

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

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

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

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

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

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

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

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

0180Z1–SEEPR–5/07