SKRIPSI

PERANCANGAN DAN PEMBUATAN DATA LOGGER SUHU DAN KELEMBABAN MENGGUNAKAN MC AT89S8252 BERBASIS PC

Disusun oleh :

Romeu Rivelino Da Silva Belo

00.17.194

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK ELEKTRONIKA

2008

PREFACE

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN DATA LOGGER SUHU DAN KELEMBABAN MENGGUNAKAN MC AT89S8252 BERBASIS PC

SKRIPSI

Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat-Syarat
Guna Mencapai Gelar Sarjana Teknik

Disusun oleh :

Romeu Rivelino Da Silva Belo

00.17.194

Diperiksa dan disetujui,
Dosen Pembimbing 1

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

Diperiksa dan disetujui,
Dosen Pembimbing 2

Ir. Yusuf Ismail Nakhoda, MP
NIP.Y.1018800189

Mengetahui,
Ketua Jurusan Teknik Elektro S-1

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO
STRATA - 1

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

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
2008

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa	:	Romeu Rivelino Da Silva Belo
NIM	:	00.17.194
Jurusan	:	Teknik Elektro S-1
Konsentrasi	:	Teknik Elektronika
Judul Skripsi	:	**PERANCANGAN DAN PEMBUATAN DATA LOGGER SUHU DAN KELEMBABAN MENGGUNAKAN MC AT89S8252 BERBASIS PC**

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

Hari	:	Senin
Tanggal	:	17 Maret 2008
Dengan Nilai	:	79,1 (B+)

Ketua Majelis Penguji

Ir. Mochtar Asroni, MSME
NIP.Y.1018100036

Sekretaris Majelis Penguji

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

Penguji I

Ir. Yoseph Dedy Irawan, ST., MT
NIP.132315178

Penguji II

Dr. Cahyo Chrysdian, MSc
NIP.103040412

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

FORMULIR BIMBINGAN SKRIPSI

Mahasiswa : Romeu Rivelino D. S. Belo

: 00.17.194

imbingan : 23 Juli 2007 s/d 23 januari 2008

kripsi : PERANCANGAN DAN PEMBUATAN DATA LOGGER SUHU DAN KELEMBABAN MENGGUNAKAN MC AT89S8252 BERBASIS PC

Tanggal	Uraian	Paraf Pembimbing
	Bab I	
	Bab II	
	Bab III	
	Bab IV	
	Bab V	
	Demo	

Malang,
Dosen Pembimbing 1

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

Form S-4a

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

FORMULIR BIMBINGAN SKRIPSI

a Mahasiswa : Romeu Rivelino D. S. Belo

: 00.17.194

a Bimbingan : 23 Juli 2007 s/d 23 Januari 2008

l Skripsi : PERANCANGAN DAN PEMBUATAN DATA LOGGER SUHU DAN KELEMBABAN MENGGUNAKAN MC AT89S8252 BERBASIS PC

	Tanggal	Uraian	Paraf Pembimbing
	2-3-2008	Presentasi Alat, revisi miniatur ruangan.	
	6-3-2008	Revisi BAB III	
	6-3-2008	Revisi BAB IV	
	6-3-2008	Revisi BAB V	
	11-3-2008	Makalah seminar Hasil	
	14-32008	ACC BAB I, BAB II, BAB III, BAB IV dan BAB V.	
	14-3-2008	Lembar Persetujuan beserta keseluruhan laporan.	

Malang, Maret 2008

Dosen Pembimbing 2

Ir. Yusuf Ismail Nakhoda, MT

NIP.Y.1018800189

Form S-4a

ABSTRAKSI

Romeu Rivelino D. S. Belo, 00.17.194, 2008, **Perancanagan dan Pembuatan Data Logger Suhu dan Kelembaban Menggunakan MC AT89S8252 Berbasis PC**, Skripsi, Teknik Elektro S-1 / Teknik Elektronika, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang, Dosen Pembimbing ke-1 Ir. F. Yudi Limpraptono, MT dan dosen pembimbing ke-2 Ir. Yusuf Ismail Nakhoda, MT.

Aplikasi ilmu elektro telah merambah hampir semua ruang lingkup kehidupan manusia yang terindikasi dengan maraknya barang-barang dengan sistim otomatisasi modern. Salah satunya adalah dalam hal pengaturan suhu dan tingkat kelembaban suatu ruang menggunakan sensor suhu dan sensor kelembaban. Sering kita jumpai banyak bangunan yang kondisinya menurun atau bahkan rusak karena pengaruh cuaca dan iklim (siklus cuaca dalam setahun). Selain itu, masih banyak bangunan yang belum mengadaptasi unsur kesehatan sebagai bagian yang penting (ruangannya lembab ataupun kurangnya sirkulasi udara yang menyebabkan suhu panas). Berangkat dari masalah inilah penulis ingin merancang dan membuat suatu alat yang dapat mendeteksi suhu dan kelembaban di dalam serta di luar ruangan, dimana alat tersebut akan memberikan referensi data bagi seorang arsitek untuk tujuan perancangan bangunan yang sehat dan berkualitas.

Alat ini menggunakan 2 buah sensor suhu dan kelembaban (SHT11), masing-masing untuk di dalam dan di luar ruangan. Kedua SHT11 tersebut setiap saat akan mendeteksi suhu dan kelembaban. Setiap satu jam sekali MC AT89S51 akan mengambil data hasil pendeteksian tersebut dan mengirimkannya ke MC AT89S8252 untuk disimpan selama maksimal 7 hari. Sewaktu-waktu user dapat melakukan pengambilan data menggunakan PC untuk disimpan sebagai data base sehingga data-data tersebut dapat digunakan oleh si arsitek.

Untuk mempermudah komunikasi pada alat maka dalam tugas akhir ini menggunakan tiga microcontroller, yaitu dua buah MC AT89S51 sebagai pengambil data dari sensor SHT11 dan satu MC AT89S8252 sebagai master untuk mengirimkan data ke PC. User hanya dapat mengambil data selama maksimal 7 hari karena keterbatasan memory pada MC AT89S8252 yang hanya sebesar 2KB. Dari hasil komparasi didapatkan hasil; pada hari ke-1 untuk sensor di dalam ruangan terdapat *error* sebesar 2,04% untuk suhu dan 3,62% untuk kelembaban; pada hari ke-1 untuk sensor di luar ruangan terdapat *error* sebesar 2,04% untuk suhu dan 5,19% untuk kelembaban; pada hari ke-2 untuk sensor di dalam ruangan terdapat *error* sebesar 1,64% untuk suhu dan 2,70% untuk kelembaban; pada hari ke-2 untuk sensor di luar ruangan terdapat *error* sebesar 1,64% untuk suhu dan 2,70% untuk kelembaban.

Kata Kunci : MC AT89S51, MC AT89S8252, SHT11 dan PC.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa yang telah melimpahkan rahmat dan petunjuk-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul Perancangan dan Pembuatan *Data Logger* Suhu dan Kelembaban Menggunakan MC AT 89S8252 Berbasis PC. Penyusunan skripsi ini merupakan salah satu syarat dalam rangka menyelesaikan studi di Fakultas Teknologi Industri Institut Teknologi Nasional Malang.

Dalam kesempatan ini, penulis ingin menyampaikan banyak terima kasih kepada :

- Kedua orang tua dan ketiga saudariku untuk doa dan dukungannya.
- Bapak Dr. Ir. Abraham Lomi, MSEE selaku rektor Institut Teknologi Nasional Malang.
- Bapak Ir. F. Yudi Limpraptono, MT selaku ketua jurusan teknik elektro S-1 dan dosen pembimbing pertama Institut Teknologi Nasional Malang.
- Bapak Ir. Yusuf Ismail Nakhoda, MT selaku dosen pembimbing kedua.
- Segenap dosen teknik elektro S-1 Institut Teknologi Nasional Malang.
- Teman-teman dan semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini.

Penulis menyadari bahwa skripsi ini belum dikategorikan sebagai pencapaian yang sempurna, namun penulis mempunyai harapan bahwa skripsi ini dapat menjadi bahan bagi pengembangan untuk masa mendatang sehingga dapat memberikan manfaat bagi masyarakat. Akhir kata, saran dan kritik yang membangun dari pembaca akan tetap penulis nantikan.

Malang, Maret 2008

Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Gambar **Halaman**

DAFTAR TABEL

Tabel **Halaman**

BAB I

PENDAHULUAN

1.1. Latar Belakang

Ilmu pengetahuan dan teknologi pada masa sekarang telah berkembang sangat pesat, dimana ilmu elektro (teknik elektronika, teknik energi listrik dan teknik informatika) memegang peranan yang sangat besar dalam perkembangan tersebut. Pengaplikasian ilmu elektro telah merambah hampir semua ruang lingkup kehidupan manusia. Hal tersebut salah satunya ditandai dengan banyak terciptanya alat-alat yang telah bekerja secara otomatis. Dengan sistem otomatisasi tersebut, segala aktifitas manusia akan lebih mudah terlaksana sehingga faktor efisiensi dan efektifitas akan didapatkan. Salah satu sistem otomatisasi yang diterapkan dalam kehidupan manusia adalah dalam hal pengaturan suhu dan tingkat kelembaban suatu ruang menggunakan sensor suhu dan sensor kelembaban.

Sering kita jumpai banyak bangunan yang kondisinya menurun atau bahkan rusak karena pengaruh cuaca dan iklim (siklus cuaca dalam setahun). Selain itu, masih banyak bangunan yang belum mengadaptasi unsur kesehatan sebagai bagian yang penting. Misalnya, ruangannya lembab atau kurangnya sirkulasi udara (panas) yang membuat penghuninya sering menggunakan pemanas ruangan ataupun *air conditioner* untuk mendapatkan faktor kenyamanan dan kesehatan, padahal itu bukan merupakan solusi terbaik melainkan pemborosan energi dan biaya. Cara terbaik yang mungkin dapat diadaptasi untuk mengurangi ataupun mengatasi dampak buruk dari iklim adalah dengan merancang dan

membuat suatu bangunan yang sesuai dengan iklim di tempat akan dibangun bangunan tersebut.

Berdasarkan kenyataan inilah penulis ingin membuat suatu alat yang dapat mengambil data suhu dan data kelembaban dari sensor suhu dan kelembaban, kemudian mengirimkannya kepada komputer sebagai *data base* sehingga sewaktu-waktu dapat diolah oleh para arsitek. Keberadaan alat ini diharapkan akan dapat membantu kerja para arsitek dalam merancang suatu bangunan berdasarkan referensi data suhu dan kelembaban tersebut, sehingga para arsitek dapat menentukan bahan material dan rancangan yang cocok dengan kondisi iklim di sekitar lokasi pembuatan bangunan. Pada akhirnya, akan tercipta bangunan yang sehat dan tahan lama. Untuk keperluan tersebut, penulis menyusun skripsi dengan judul :

Perancangan dan Pembuatan Data Logger Suhu Dan Kelembaban Menggunakan MC AT89S8252 Berbasis PC

1.2. Rumusan Masalah

1. Bagaimana merancang dan membuat suatu alat yang dapat mengambil data suhu dan data kelembaban dari sensor suhu dan kelembaban serta menyimpannya di PC sebagai *database*.
2. Bagaimana membuat program untuk *microcontroller* dan PC.

1.3. Batasan Masalah

Dalam menyusun skripsi ini diperlukan suatu batasan masalah agar tidak menyimpang dari ruang lingkup yang akan dibahas. Batasan-batasan masalahnya antara lain sebagai berikut.

1. Pengambilan data oleh MC AT89S51 adalah setiap satu jam dan data tersebut akan disimpan di MC AT89S8252 selama maksimal 7 hari. Jadi, data tersebut akan terhapus setelah hari ke-7.
2. Pengguna bisa mengambil data hari ke-1 sampai data hari ke-7, baik menggunakan PC ataupun *laptop*.
3. LCD hanya menampilkan data yang sifatnya *real time*.
4. Tidak membahas mengenai catu daya.

1.4. Tujuan

Tujuan dari penyusunan skripsi ini adalah membuat suatu alat yang dapat mengambil data suhu dan data kelembaban dari sensor suhu dan kelembaban (SHT11) kemudian mengirimkannya kepada PC sebagai *database* sehingga dapat diolah oleh para arsitek untuk tujuan perancangan bangunan.

1.5. Metodologi

Untuk menguraikan dan menjelaskan tentang penulisan laporan skripsi ini, maka dilakukan langkah-langkah ataupun metode-metode sebagai berikut.

1. Mencari, membaca dan mengambil data dari berbagai referensi yang berhubungan dengan penyusunan skripsi ini.
2. Melakukan diskusi dengan dosen pembimbing mengenai teori-teori dan komponen-komponen apa saja yang digunakan dalam pembuatan alat.

3. Merancang dan membuat *hardware* beserta *software*-nya.
4. Mengadakan pengujian *software* dan *hardware* secara keseluruhan kemudian menarik kesimpulan.

1.6. Sistematika Penulisan

Dalam pembuatan laporan skripsi ini penulis menggunakan sistematika sebagai berikut.

BAB I. Pendahuluan

Membahas tentang latar belakang permasalahan, rumusan masalah, tujuan, batasan masalah, metodologi penulisan serta sistematika susunan penulisan dari laporan skripsi ini.

BAB II. Teori Penunjang

Bab ini berisikan tentang teori-teori penunjang yang digunakan dalam perencanaan dan pembuatan skripsi.

BAB III. Perencanaan Sistem

Pada bab ini akan dibahas perencanaan dan pembuatan perangkat keras dan perangkat lunak yang digunakan.

BAB IV. Pengujian Alat

Berisi tentang data hasil pegujian alat yang telah dibuat secara keseluruhan

BAB V. Kesimpulan

Berisi kesimpulan yang diperoleh dari perencanaan, realisasi dan Pengujian alat.

BAB II

LANDASAN TEORI

2.1. Pendahuluan

Bab ini akan membahas teori yang menunjang perencanaan dan pembuatan alat. Diawali dengan membahas tentang *microcontroller* yang diterapkan untuk unit kontrol utama selain komputer. Pada bagian lain juga dibahas tentang IBM PC, m*icrocontroller* AT89S8252, komunikasi data serial, RS-232, sensor suhu dan kelembaban (SHT11), LCD tipe M1632, Keypad 3x4, dan bahasa pemrograman *Borland Delphi* untuk unit kontrol pada PC.

2.2. IBM PC

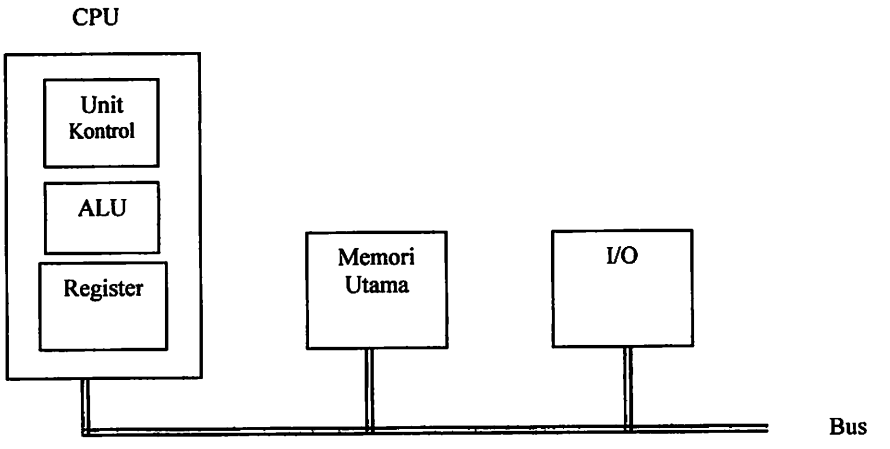
IBM PC mempunyai organisasi dan struktur yang sama dengan PC yang lainnya, diantaranya yaitu prosesor, memori utama, memori sekunder dan *input / output.*

2.2.1. Prosesor

Organisasi sebuah komputer sederhana yang berorientasi pada *bus* ditampilkan pada gambar 2.1. *Central Processing Unit* (CPU) adalah "otak" dari sebuah komputer. Fungsi CPU adalah menjalankan program-program yang disimpan dalam memori utama dengan cara mengambil instruksi-instruksi, menguji instruksi tersebut dan menjalankannya satu demi satu. Komponen-komponen itu dihubungkan oleh sebuah *bus*, yaitu sekumpulan kabel-kabel paralel untuk mentransmisikan alamat (*address*), data dan sinyal-sinyal kontrol. *Bus* dapat berada di luar CPU, yang menghubungkan CPU dengan memori dan

peralatan I/O (*input* / *output*). CPU terdiri dari dari beberapa bagian berbeda. Unit kontrol bertanggung jawab mengambil instruksi-instruksi dari memori utama dan menentukan jenis instruksi tersebut. Unit logika aritmatik (ALU) menjalankan operasi-operasi seperti penjumlahan dan *Boolean AND*.

CPU juga berisi sebuah memori kecil berkecapatan tinggi yang digunakan untuk menyimpan hasil-hasil sementara dan informasi kontrol tertentu. Memori ini terdiri dari sejumlah *register*, yang masing-masing memiliki ukuran dan fungsi tersendiri. Biasanya, seluruh *register* itu memiliki ukuran sama. Setiap *register* dapat menyimpan satu bilangan, hingga mencapai jumlah maksimum tertentu tergantung pada ukuran register tersebut. *Register-register* dapat dibaca dan ditulis dengan kecepatan tinggi karena berada dalam CPU. *Register* yang paling penting adalah *Program Counter* (PC), yang menunjuk ke instruksi berikutnya yang harus diambil untuk dijalankan. Fungsi penting lainnya adalah *Instruction Register* (IR), yang menyimpan instruksi yang sedang dijalankan.



Gambar 2-1. Organisasi komputer sederhana dengan CPU dan peralatan I/O[3]

2.2.2. Memori Utama

Memori adalah bagian dari komputer tempat program-program dan data-data disimpan. Tanpa sebuah memori sebagai tempat untuk mendapatkan informasi guna dibaca dan ditulis oleh mikroprosesor, maka tidak akan ada komputer-komputer *digital* dengan sistem penyimpanan program.

2.2.3. Memori Sekunder

Memori sekunder terdiri dari disk magnetik, *floppy disc*, disk-disk IDE, disk-disk SCSI, RAID, CD-ROM, CD-RW, DVD dan banyak lagi yang lainnya.

2.2.4. Bus

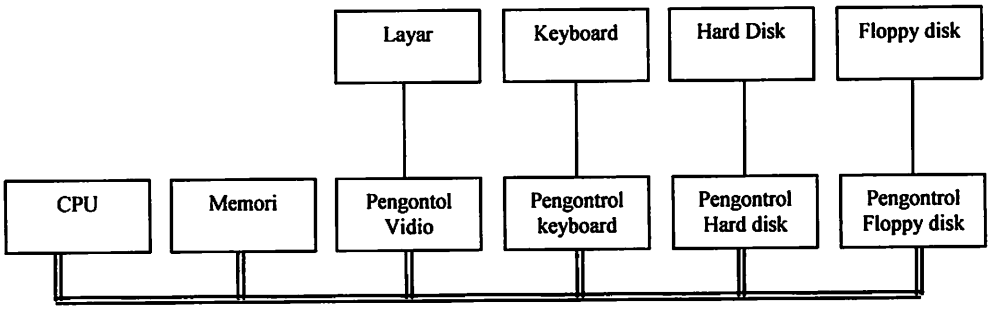
Struktur logika sebuah komputer pribadi *low-end* sederhana diperlihatkan pada gambar 2.2. Struktur komputer memiliki sebuah *bus* tunggal yang digunakan untuk menghubungkan CPU, memori dan piranti-piranti I/O; sebagian besar sistem memilki dua *bus* atau lebih. Masing-masing piranti I/O terdiri dari dua bagian; bagian pertama memuat sebagian besar elektronik yang disebut pengontrol, dan bagian satunya lagi memuat piranti I/O itu sendiri, seperti misalnya *disc drive*. Pengontrol biasanya terpasang pada sebuah papan yang disisipkan ke dalam slot bebas, kecuali untuk pengontrol yang bukan merupakan pilihan (misalnya *keyboard*), yang kadang-kadang terletak pada *motherboard*. Meskipun layar bukan merupakan satu opsi, namun pengontrol video kadang-kadang terletak pada papan *plug-in* guna memungkinkan pemakai untuk memilih di antara papan-papan yang dilengkapi dan yang tidak dilengkapi dengan *acsellerator* grafis, memori tambahan dan sebagainya. Pengontrol dihubungkan

dengan pirantinya melalui sebuah kabel yang dipasang pada penghubung di bagian kotak (*chassing*).

Tugas pengontrol adalah untuk mengontrol piranti I/O dan menangani akses *bus* untuk piranti I/O tersebut. Sebagai contoh, ketika sebuah program menghendaki data dari *disk*, program memberikan perintah kepada pengontrol *disk* yang selanjutnya menerbitkan pencarian dan perintah-perintah lainnya kepada *drive*. Ketika *track* dan *sector* yang tepat ditemukan, *drive* memulai memilih data sebagai suatu aliran *bit* serial kepada pengontrol. Tugas pengontrol adalah untuk memilah-milahkan aliran *bit* menjadi satuan-satuan, dan menuliskan setiap satuan ke dalam memori. Sebuah satuan adalah *word* atau lebih. Sebuah pengontrol yang membaca atau menuliskan data pada atau dari memori tanpa intervensi CPU akan dikatakan sebagai melakukan akses memori langsung, yang lebih dikenal melalui singkatan DMA. Ketika transfer selesai, pengontrol biasanya menyebabkan suatu *interrupt*, yang memaksa CPU untuk menunda menjalankan programnya saat ini dan mulai menjalankan suatu prosedur khusus, yang disebut *interrupt handler*, untuk meriksa kesalahan-kesalahan, mengambil tindakan khusus tertentu yang diperlukan dan memberitahu sistem pengoperasian bahwa I/O saat ini telah selesai. Ketika *interrupt handler* diselesaikan, CPU melanjutkan dengan program yang telah ditunda ketika *interrupt* terjadi.

Bukan hanya *bus* yang digunakan oleh pengontrol I/O saja, tetapi juga digunakan oleh CPU untuk mencocokkan instruksi dan data. Jika CPU dan pengontrol I/O menggunakan *bus* pada saat bersamaan, maka *chip* yang disebut *bus arbiter* memutuskan mana yang mendapat giliran pertama. Pada umumnya piranti-piranti I/O diberi prioritas di atas CPU, sebab disk dan piranti-piranti

bergerak lainnya tidak dapat dihentikan. Jika memaksa piranti-piranti tersebut untuk menunggu, maka akan kehilangan data. Ketika tidak ada I/O yang tidak bekerja, CPU dapat memiliki semua siklus *bus*. Tetapi ketika piranti I/O tetentu juga sedang bekerja, maka piranti itu akan meminta siklus tersebut. Proses ini disebut siklus pengambilalihan dan akan memperlambat komputer.



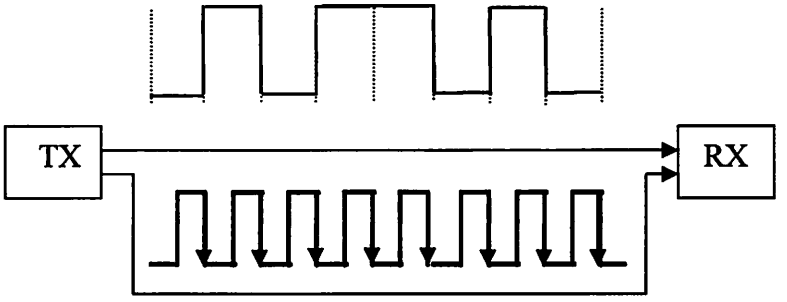
Gambar 2-2. Stuktur logika untuk PC sederhana[3]

2.3. Komunikasi Serial

Komunikasi serial ialah pengiriman data secara serial (data dikirim satu per satu secara berurutan) sehingga komunikasi serial jauh lebih lambat daripada komunikasi paralel. Serial *port* lebih sulit ditangani karena peralatan yang dihubungkan ke serial *port* harus berkomunikasi mengunakan transmisi serial sedangkan data di komputer diolah secara paralel. Oleh karena itu data dari / ke serial *port* harus dikonversikan ke / dari bentuk paralel untuk bisa digunakan. Kelebihan komunikasi serial ialah jangkauan panjang kabel yang lebih jauh dibandingkan paralel karena serial *port* mengirimkan logika 1 (*high*) dengan kisaran tegangan -3 hingga -25 *volt* dan logika 0 (*low*) sebagai +3 hingga +25 *volt* sehingga kehilangan daya karena panjangnya kabel bukan masalah utama.

Komunikasi serial ada dua macam, *asynchronous* serial dan *synchronous* serial. *Synchronous* serial adalah komunikasi dimana hanya ada satu pihak (pengirim atau penerima) yang menghasilkan *clock* dan mengirimkan *clock* tersebut bersama-sama dengan data. Contoh pengunaan *synchronous* serial terdapat pada transmisi data *keyboard. Asynchronous* serial adalah komunikasi dimana kedua pihak (pengirim dan penerima) masing-masing menghasilkan *clock* namun hanya data yang ditransmisikan (tanpa *clock*). Agar data yang dikirim sama dengan data yang diterima, maka kedua frekuensi *clock* harus sama dan harus terdapat sinkronisasi. Setelah adanya sinkronisasi, pengirim akan mengirimkan datanya sesuai dengan frekuensi *clock* pengirim dan penerima akan membaca data sesuai dengan frekuensi *clock* penerima.

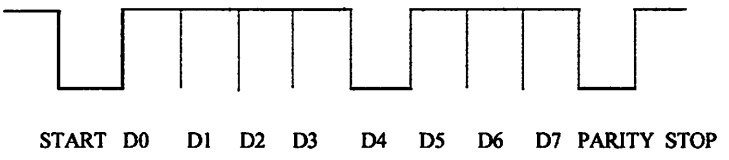
2.3.1. Komunikasi *Sinkron*



Gambar 2-3. Komunikasi serial dengan sinyal sinkronisasi[8]

Setiap TX mengirimkan Dx atau *bit* ke x dari satu *byte* data akan diikuti dengan sinyal sinkronisasi yang berupa sinyal transisi dari rendah ke tinggi atau tinggi ke rendah. RX akan mengetahui bahwa dijalur data ada data milik Dx sesuai dengan banyaknya sinyal sinkronisasi yang diterima, saat sinyal sinkronisasi pertama, berarti data milik D0, kedua milik D1dan seterusnya.

2.3.2. Komunikasi *Asinkron*



Gambar 2-4. Format sinyal serial *asinkron*[8]

Cara kedua dengan komunikasi *asinkron*, yaitu dengan menetapkan kecepatan *bit* dan menyisipkan beberapa *bit* protokol, yaitu bit *START*, *PARITY bit* dan *STOP* seperti diperlihatkan pada gambar 2.4. diatas.

- Kecepatan *bit* disebut sebagai *baud rate* atau pesat bit dan disingkat *bps (bit per second)*; pada standar komunikasi diantaranya adalah 1200, 4800, 9600 bps.
- Makin besar pesat *bit*, makin cepat data ditransmisikan, tetapi memerlukan *bandwidth* (jalur data) yang semakin lebar. Pada penggunaan kabel biasa atau kabel telepon, kecepatan transmisi data dibatasi oleh *bandwidth* kabel tersebut.
- Saluran tanpa data bertegangan '1' , *start bit* selebar 1 pulsa, selalu '0'.
- Setelah *bit start*, diikuti serial data, jumlah data dapat 7 atau 8.
- Setelah data-data *bit* bisa diikuti (jika diperlukan) oleh *parity bit*; dimana jika dipilih *parity even*, maka *bit parity* akan menggenapkan jumlah *bit* '1'-nya; sedangkan jika dipilih *parity odd*, maka *bit parity* akan mengganjilkan jumlah *bit* '1'.
- Akhir data adalah stop *bit* yang selalu '1'

2.3.3. Arah Pengiriman Data

Dikenal tiga macam arah pengiriman data, yaitu *Simplex, Half Duplex,* dan *Full Duplex.*

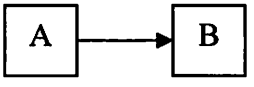
Simplex adalah sistem pemindahan data yg hanya satu arah saja, misalnya dari A ke B, dimana A sebagai pengirim dan B sebagai penerima, dan tidak dapat mengirimkan data dari B ke A.

Half Duplex adalah sistem pemindahan data dua arah, tetapi tidak dapat dilakukan secara bersamaan, harus bergantian.

Full Duplex adalah sistem pemindahan data dua arah dan dapat berlangsung secara bersamaan dalam satu waktu.

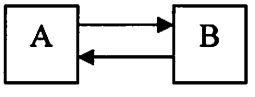
Untuk lebih jelasnya, dapat dilihat pada gambar berikut.

a. Komunikasi data *Simplex*



b. Komunkasi data *Half Duplex*

c. Komunikasi data *Full Duplex*



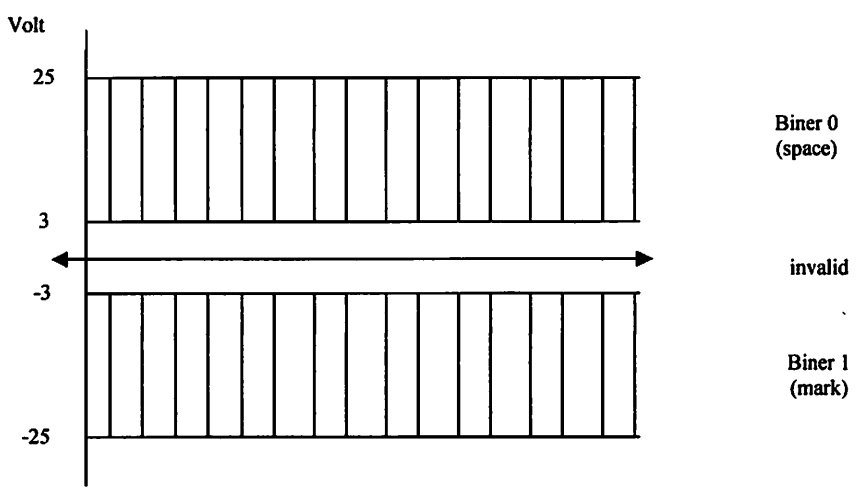
2.4. RS-232

RS-232 (*Recommended Standart number 232*) adalah suatu standar antar muka yang dipublikasikan oleh EISA (*Electronic Industries Association*). Antarmuka RS-232 digunakan dalam banyak aplikasi komunikasi data. Hal-hal yang perlu diketahui dalam standar antarmuka RS232 antara lain: tegangan

standar bagi data *biner* 0 dan 1, sinyal-sinyal yang dipergunakan, dan cara interkoneksi antar RS-232.

2.4.1. Tegangan RS-232

Untuk menampilkan data biner dibutuhkan dua besaran tegangan. *Biner* 1 atau disebut *mark* dinyatakan dengan tegangan antara –3 V sampai dengan –25 V, sedang *biner* 0 atau yang disebut *space* dinyatakan tegangan antara +3 V sampai +25 V. tegangan antara –3 V sampai dengan +3 adalah tegangan *invalid.* Tegangan yang mewakili *biner* nol dan satu ditunjukkan dalam gambar 2-5 berikut ini.



Gambar 2-5. Tegangan yang mewakili *biner* 0 dan 1[8]

2.4.2. Sinyal-sinyal RS-232

Standar antarmuka RS-232 mendefinisikan sinyal-sinyal yang dipakai baik untuk mengirim/menerima data maupun untuk proses jabat tangan (*handshaking*). Dalam aplikasinya, sinyal-sinyal RS-232 ini dihubungkan dengan *connector* 25

pin (DB-25) atau 9 *pin* (DB-9). Pada tabel 2-1 berikut ini diberikan nama sinyal penting beserta hubungannya dengan *connector* DB-9 maupun DB-25.

Tabel 2-1. Sinyal-Sinyal untuk *Connector* DB-9 dan DB-25[3]

Nomor *Pin*		Nama Sinyal
DB 9	DB 25	
1	8	DCD (*Data Carrier Detect*)
2	3	RD (*Receive Data*)
3	2	TD (*Transmit Data*)
4	20	DTR (*Data Teminal Ready*)
5	7	SG (*Signal Ground*)
6	6	DSR (*Data Set Ready*)
7	4	RTS (*Request To Send*)
8	5	CTS (*Clear To Send*)

2.4.3. Antarmuka serial RS-232 pada IBM PC

Pada IBM PC terdapat antarmuka serial yang mengikuti standar antarmuka RS-232. antarmuka ini menggunakan rangkaian terintegrasi UART (*Universal Asyncronous Reciever/ Transmitter*). IBM PC dapat mempunyai beberapa antarmuka serial, dua diantaranya yang paling penting adalah *primary asynchronous communication adaptor* yang disebut COM 1 dan *secondary asynchronous adaptor* yang disebut COM 2.

Tabel 2-2. *Address Register* RS-232[8]

Nama *Register*	COM1	COM2
TX Buffer (Transmit Buffer)	03F8H	02F8H
RX Buffer (Receive Buffer)	03F8H	02F8H
Baud rate devisor latch LSB	03F8H	02F8H
Baud rate devisor latch MSB	03F9H	02F9H
Interrupt Enable Register	03F9H	02F9H
Interrupt identification Register	03FAH	02FAH
Line Control Register	03FBH	02FBH
Modem Control Register	03FCH	02FCH
Line Status Rgister	03FDH	02FDH
Modem Status Register	03FEH	02FEH

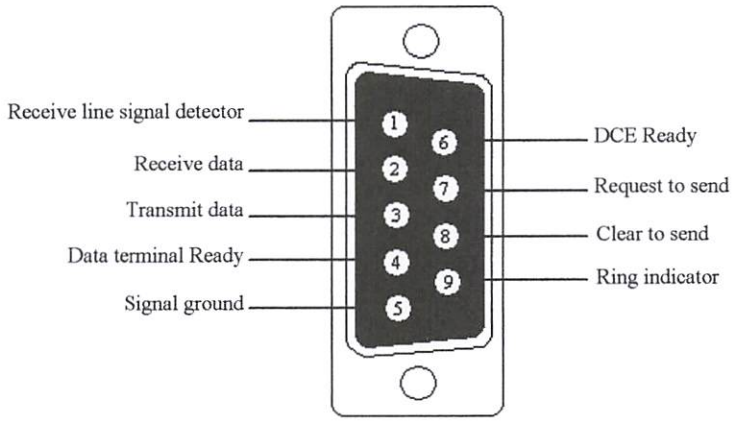
Fungsi dari beberapa *register* diantaranya adalah sebagai berikut:

a) TX *buffer*: menampung dan menyimpan data yang akan dikirim keluar. Data ini dikirim oleh CPU ke TX *buffer* setelah mengecek kepastian tentang diperbolehkannya melakukan pengiriman.

b) RX *buffer* : menampung dan menyimpan data yang diterima dari luar. Data itu harus dibaca oleh CPU setelah mengecek kepastian tentang masukannya data.

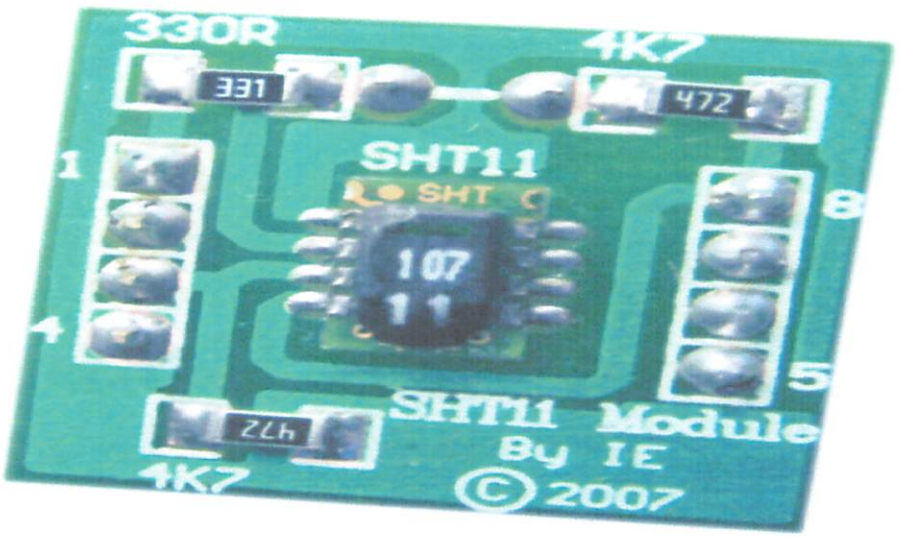
c) *Baud rate devisor last significant bit*: menampung angka byte bobot rendah untuk pembagi *clock* yang akan dimasukkan agar didapat *baud rate* yang diplih. Angka pembagi dapat dipilih antara 01H hingga FFH.

d) *Baud rate devisor most significant bit*: menampung angka *byte* bobot tinggi untuk pembagi *clock* yang akan dimasukkan agar didapat *baud rate* yang dipilih. Angka pembagi dapat dipilih antara 00H hingga FFH.



Gambar 2-6. Skematik DB 9[3]

2.5. Sensor Suhu dan Kelembaban (SHT11)



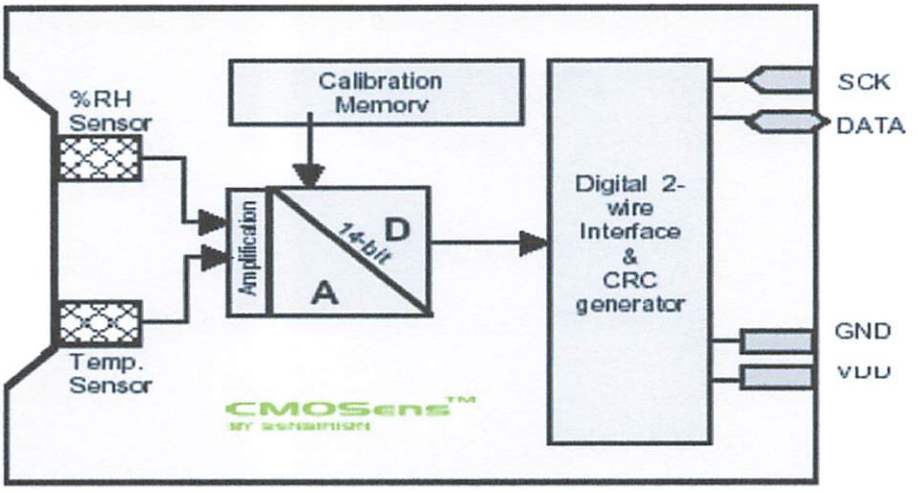
Gambar 2-7. SHT11[1]

2.5.1. Umum

SHT11 adalah sebuah *chip* tunggal multi sensor suhu dan kelembaban dalam sebuah modul yang telah dikalibrasi oleh pabriknya dan mempunyai keluaran *digital* sehingga tidak memerlukan eksternal ADC lagi (untuk keperluan pengubahan sinyal *analog* dari sensor tersebut menjadi sinyal *digital*). Modul ini menjamin adanya ketahanan dan tingkat kestabilan yang sangat bagus dalam waktu yang lama.

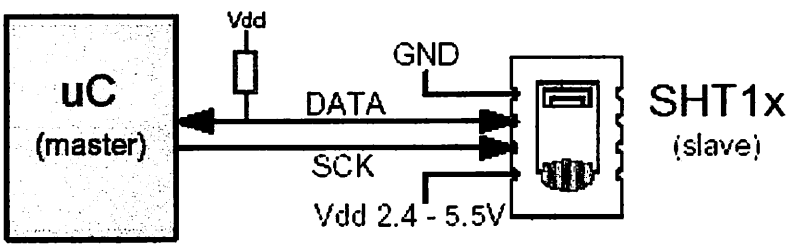
Modul ini memasukkkan sebuah elemen pengindera *capasitive polymer* untuk kelembaban dan sebuah *bandgap* (plester celah) untuk sensor suhu. Keduanya disatukan pada sebuah ADC dan sebuah sirkuit *serial inteface* dalam *chip* yang sama. Modul ini menghasilkan kualitas sinyal yang sangat bagus, respon waktu yang cepat, tidak terlalu peka terhadap gangguan dari luar dan mempunyai harga yang terjangkau.

Sensor ini bisa diaplikasikan untuk keperluan HVAC, otomotif, stasiun cuaca, tes dan pengukuran, *data loging* (pengambilan data), otomatisasi dan kesehatan. Berikut ini adalah blok diagram SHT11.



Gambar 2-8. Blok diagram SHT11[1]

2.5.2. Spesifikasi *Interface*



Gambar 2-9. Sirkuit untuk Aplikasi SHT11[1]

a) *Power Pins*

SHT11 membutuhkan suplai tegangan diantara 2,4 *volt* dan 5,5 *volt*. Setelah diberi tegangan, modul ini membutuhkan 11ms untuk mencapai keadaan tidurnya, dimana tidak boleh ada perintah apapun yang dikirim sebelum waktu tersebut tercapai. *Pin-pin power supply* (VDD, GND) dihubung dengan sebuah kapasitor sebesar 100nF.

b) *Serial Interface* (*Bidirectional* 2-*wire*)

Serial interface dari SHT11 dioptimasi untuk pembacaan sensor dan konsumsi daya serta tidak cocok (*compatible*) dengan *interface-interface* I2C.

- *Serial Clock Input* (SCK)

Digunakan untuk mensinkronisasikan komunikasi antara sebuah mikrokontrol dengan SHT11 tersebut.

- Data Serial (DATA)

Pin DATA digunakan untuk mengirim data *in* dan *out* dari sensor tersebut. DATA berubah setelah kurva menurun dan *valid* pada kurva meninggi dari *serial clock* SCK. Selama pengiriman, garis DATA harus tetap stabil ketika SCK tinggi. Untuk menghindari adanya tabrakan sinyal, mikrokontrol hanya boleh menjalankan DATA rendah. Sebuah resistor *pull-up* eksternal dibutuhkan

untuk menarik sinyal agar tinggi. Resistor-resistor *pull-up* biasanya sudah termasuk di dalam sirkuit-sirkuit I/O dari mikrokontrol-mikrokontrol.

2.6. *Microcontroller* AT89S8252

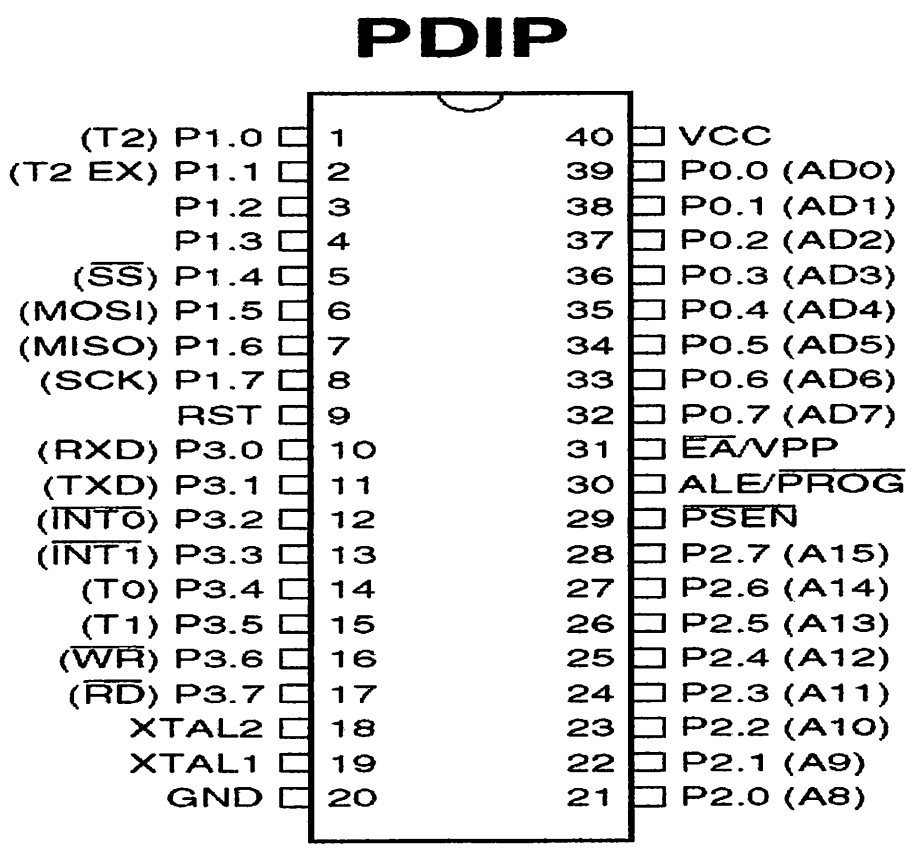
2.6.1. Umum

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

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

2.6.2. Konfigurasi Kaki-Kaki *Microcontroller* AT89S8252

Konfigurasi dari kaki-kaki (*pin*) mikrokontrol AT89S8252 diperlihatkan pada gambar 2-10 berikut.



Gambar 2-10. Konfigurasi *pin Microcontroller* AT89S8252[7]

Fungsi tiap-tiap *pin* adalah sebagai berikut.

1. GND : dihubungkan dengan *ground* rangkaian.
2. VCC : dihubungkan dengan sumber tegangan.
3. *Port* 0 (P0.0 - P0.7) : merupakan *port* I/0 8 *bit* dua arah. *Port* ini digunakan sebagai multipleks *bus* alamat rendah (A0-A7) dan data selama pengaksesan program *memory* dan data memori eksternal.
4. *Port* 1 (P1.0 - P1.7) : dapat berfungsi sebagai *input* atau *output* dan bekerja baik untuk operasi *bit* maupun *byte*, tergantung dari pengaturan *software*.
5. *Port* 2 (P2.0 - P2.7) : merupakan *port input output* dengan *internal pull up*. *Port* 2 mengeluarkan *high order address byte* selama pengambilan program memori eksternal dan selama mengakses data memori eksternal.

Port 2 juga menerima *high order address bit* dan beberapa sinyal kontrol selama pemrograman dan verifikasi.

6. *Port* 3 (P3.0-P3.7) : selain berfungsi sebagai I/O juga mempunyai fungsi khusus sebagai berikut.
   - $\overline{RD}$ (P3.7) : Sinyal pembacaan memori dari luar
   - $\overline{WR}$ (P3.6) : Sinyal penulisan memori luar data
   - T1 (P3.5) : Masukan dari pewaktu/pencacah 1
   - T0 (P3.4) : Masukan dari pewaktu/pencacah 0
   - $\overline{INT1}$ (P3.3) : *Interupt* 1 eksternal
   - $\overline{INT0}$ (P3.2) : *Interupt* 0 eksternal
   - TXD (P3.1) : Keluaran pengiriman data untuk serial *port* (asinkron) atau sebagai keluaran *clock* (sinkron).
   - RXD (P3.0) : Masukan penerimaan data serial (asinkron) atau sebagai masukan / keluaran data (sinkron).
7. RST : merupakan *pin* yang berfungsi untuk mereset sistem mikrokontrol AT89S8252. Perubahan taraf tegangan dari rendah ke tinggi akan mereset mikrokontrol.
8. ALE (*Address Latch Enable*)/PROG : *pin* ALE aktif tinggi mengeluarkan pulsa *output* untuk me-*latch* satu *byte* alamat rendah selama mengakses ke memori luar. ALE dapat mengendalikan 8 beban TTL. *Pin* ini juga merupakan *input* pulsa program yang aktif rendah selama pemrograman EPROM.

Pada operasi normal, ALE dikeluarkan pada suatu kecepatan yang konstan yaitu 1/6 dari frekuensi osilator dan dapat digunakan untuk *timing* eksternal atau untuk tujuan peng-*clock*-an.

9. PSEN (*Program Strobe Enable*) : *pin* ini aktif rendah yang merupakan *strobe* pembacaan ke program memori eksternal.
10. XTAL 1 dan XTAL 2

    Pin XTAL 1 merupakan *pin input* ke penguat osilator pembalik dan *pin* XTAL 2 merupakan *pin output* dari penguat osilator pembalik.
11. EA / VPP (*Eksternal Acces/Programming Supply Voltage*)

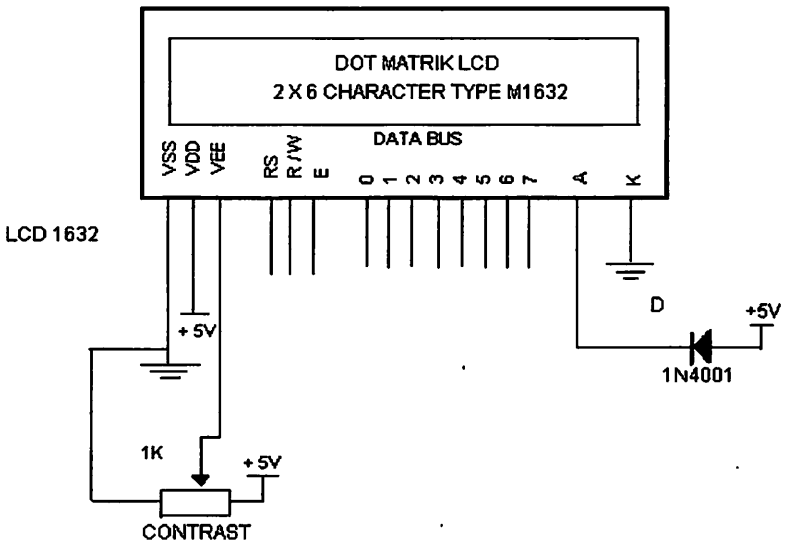
    Pin EA harus dihubungkan ke VCC agar mikrokontrol AT89S8252 dapat mengakses kode mesin dari program memori *internal* dengan lokasi 0000H sampai 0FFFH.

2.7. LCD (*Liquid Cristal Display*)

Modul peragaan yang digunakan dalam aplikasi ini adalah LCD modul M1632. modul ini membutuhkan daya yang kecil dan dilengkapi dengan panel LCD dengan tingkat kontras yang cukup tinggi serta pengendali CMOS yang terpasang pada modul tersebut. Pengendali mempunyai pembangkit karakter ROM / RAM dan *display* data RAM. Semua fungsi *display* diatur oleh instruksi-instruksi sehingga modul LCD ini dapat dengan mudah dihubungkan dengan unit mikroprosesor. LCD tipe ini tersusun sebanyak 2 baris dengan 16 karakter. Masukan yang diperlukan untuk mengendalikan modul berupa *bus* data yang telah ter-*multiplekster* dengan *bus* alamat serta 3 *bit* sinyal kontrol. Sementara

pengendalian LCD dilakukan secara *internal* oleh *controller* yang sudah terpasang dalam modul LCD.

LCD ini terdiri atas sel dari 2 lembar kaca yang di sampingnya tertutup rapat. Antara 2 lembar kaca tersebut diberi bahan kristal cair (*liquid crystal*) yang tembus cahaya. Permukaan luar dari masing-masing keping kaca mempunyai lapisan penghantar tembus cahaya seperti *oxida* timah (*tinoxide*) atau *oxida indium* (*indium oxide*). Sel mempunyai ketebalan sekitar $1X10^{-5}$ meter dan diisi dengan kristal cair.



Gambar 2-11. Rangkaian LCD[6]

Keterangan :

1. A= *common signal*
2. B= *segmen signal*
3. C= serial data
4. D= *timing digital*

LCD modul M1632 mempunyai spesifikasi perangkat keras sebagai berikut.

1. 16 karakter dan 2 baris tampilan yang terdiri dari 5X7 dot matrik ditambah *kursor*.
2. Pembangkit karakter ROM untuk 192 jenis karakter.
3. pembangkit karakter RAM untuk 8 jenis karakter.
4. 80X8 *display* data RAM (maksimal 8 karakter).
5. Osilator *internal*.
6. Catu daya ± 5 Volt.
7. Secara otomatis akan reset saat catu daya dinyalakan.

LCD M1632 mempunyai 16 *pin* atau penyemat yang mempunyai fungsi-fungsi seperti yang ditunjukkan dalam tabel 2-3 di bawah ini.

Tabel 2-3. Fungsi-Fungsi *Pin* pada LCD M1632[6]

No	Nama Penyemat	Fungsi
1	Vss	Terminal *ground*
2	Vcc	Tegangan catu ± 5 Volt
3	Vee	Drive LCD
4	Rs	Sinyal pemilih *register* 0 = instruksi *register* 1 = data *register* (tulis dan baca)
5	R / W	Sinyal seleksi tulis / baca 0 = tulis 1 = baca
6	E	Sinyal operasi awal, sinyal ini mengaktifkan data tulis dan baca
7-14	DB0 - DB7	Merupakan saluran data, berisi perintah dan data yang akan ditampilkan

15	V+BL	Pengendali kecerahan latar belakang LCD 4-4,42V dan 50-500mA
16	V-BL	Pengendali kecerahan latar belakang LCD 0V

2.8. Bahasa Pemrograman *Borland Delphi*

Delphi adalah perangkat lunak untuk menyusun program aplikasi yang berdasarkan pada bahasa pemrograman bahasa *pascal* dan bekerja dalam lingkungan sistem operasi *Windows.* Dengan *Delphi* diperoleh kemudahan dalm menyusun program aplikasi, karena *Delphi* menggunakan komponen-komponen yang akan menghemat penulisan program dengan fasilitas VCL *(Visual Component Library).*

Dalam pembuatan sebuah program, *Delphi* menggunakan sistem yang disebut RAD *(Rapid Aplication Development).* Sistem ini memanfaatkan bahasa pemrograman visual yang membuat seorang *programmer* lebih mudah mendesain tampilan program (*User Interface*). Cara ini bermanfaat untuk membuat program yang bekerja pada sistem *Windows* yang memang tampilan layarnya lebih rumit (sekaligus dapat dilihat dengan indah) dibanding dengan sistem DOS.

Aplikasi dalam tatanan GUI *(Graphical User Interface)* yaitu karakter program aplikasi yang menggunakan sarana perantara grafis dapat dibentuk dengan *Delphi.* Seperti kotak dialog (*dialog box*), tombol (*button, menu*) dan lain sebagainya. Contoh program GUI adalah program-program *Windows.* Dengan *Delphi* sebuah *Windows* yang mengandung tombol-tombol, kotak cek, tombol pilihan panel dan komponen lainnya dapat dengan mudah diciptakan.

2.8.1. IDE (*Integrated Development Environment*) *Delphi*

IDE adalah suatu lingkungan dimana mempunyai sebuah *tools* yang diperlukan untuk desain, menjalankan, mengetes sebuah aplikasi, dan disajikan serta terhubung dengan baik sehingga memudahkan pengembangan program. Pada *Delphi* terdiri dari *main Windows, Component Palette, Toolbar, Form designer, Code Editor, Code Explorer.* Intergrasi ini memberikan kemudahan dalam mengembangkan aplikasi yang kompleks.

- ***Main Windows*** **(jendela utama)**

Main Widows adalah bagian utama dari IDE. *Main Windows* mempunyai semua fungsi utama dari program-program *Windows* lainnya. *Main Windows* dibagi tiga yaitu menu utama, *toolbar*, dan *component palette*.

Menu utama. Seperti program *Windows* lainnya, menu utama dipakai untuk membuat dan menyimpan *file* memanggil *wziard*, menampilkan jendela lain, mengubah *option* dan lain sebagainya setiap pilihan pada menu juga dapat dipanggil dengan sebuah tombol pada *toolbar*.

Toolbar. Beberapa operasi pada menu utama dapat dilakukan melalui *toolbar.* Setiap tombol pada *toolbar* mempunyai sebuah *tooltip* yang berisi informasi mengenai fungsi dari tombol tersebut. Selain *component palette,* ada 5 *toolbar* terpisah yaitu *debug, desktop, standart, view, dan custom.*

Component Palette adalah toolbar dengan ketinggian ganda, yaitu berisi page kontrol dengan semua komponennnya. Urutan dan tampilan dari *page* dan

komponen pada komponen *palette* dapat diatur dengan klik kanan atau dengan memilih menu *component configure component* dari menu utama.

- ***Form designer***

Diawali dengan jendela kosong yang memungkinkan untuk merancang aplikasi *Windows*. Dari sini dapat ditentukan tampilan aplikasi sesuai dengan yang diingnkan. Berinteraksi dengan *form designer* dengan cara memilih komponen *palette* dan meletakkannya ke dalam *form*, posisi dan ukuran dapat diubah-ubah dengan menggunakan *mouse*. Untuk mengubah tampilan dan perilaku komponen maka digunakan *object inspector* dan *code editor.*

- ***Object Editor***

Object Editor terdiri dari dua tab yaitu *tab properties* dan *tab events. Tab properties* memberi fasilitas untuk melihat dan mengubah *property* dari setiap item. Klik pada sebuah form kosong, dan perhatikan atribut-atribut yang ada. Jika terdapat tanda + disamping property maka *property* tersebut berarti mempunyai *sub property.* Contohnya *property font*, jika diklik ganda pada *property font* maka akan ditampilkan *sub property*nya seperti *color, height,name* dan lain-lain. *Tab Event* berisi *event-event* yang dapat direspon oleh sebuah obyek. Klik tab *event* disebelah kanan tab *properties*. Misalnya ingin sesuatu dikerjakan pada saat *form* ditutup, maka tindakan tersebut (berupa sebuah *procedure*) pada *Onclose.*

2.8.2. Struktur Menu *Delphi*

Struktur menu Delphi memberikan *tools* untuk mengakses lingkungan *Delphi*.

1. *File*

Menu *file* adalah menu paling penting dan akan dijabarkan pada bagian berikut:

New. Digunakan untuk memulai obyek baru.

New Application. Dengan memilih menu ini, berarti akan membuat *project* baru. Jika belum membuka sebuah *project* atau *object* yang dibuka sudah disimpan ke disk. *Delphi* akan menutup *project* tersebut dan akan membuat *project* baru, termasuk membuat jendela *editor* baru dengan nama *file* UNIT.PAS, *form baru* (*form* 1) dan menampilkan *object inspector*.

New Form. Menu ini dipakai untuk membuat *form* baru.

New frame. Untuk membuat *frame* kosong dan menambahkannya ke dalam *project*.

Open. Menyatakan pada Delphi bahwa akan dibuka sebuah object dapat berupa sebuah program atau seluruh *project*.

Open Project. Untuk membuka sebuah *project*.

Reopen. Menu ini dipakai untuk membuka *object favorit* yang sudah pernah dibuka.

Save. Menu ini dipakai untuk menyimpan *module* yang sedang aktif.

Save as. Dipakai untuk menyimpan *module* dengan nama lain.

Save project as. Menu ini dipakai untuk menyimpan *project* dengan nama baru.

Save all. Menyimpan sebuah *object* yang dibuka.

Close. Untuk menutup modul program dengan *form*-nya. Jika modul tersebut belum disimpan, saat menutup, *Delphi* akan menanyakan apakah modul tersebut akan disimpan.

Close all. Menutup *project*.

Use Unit. Delphi akan menambahkan *klauda uses* pada program yang dibuat. Artinya sebuah unit akan dipakai dalam *project*.

Print. Mencetak item *Delphi* yang telah dipilih.

Exit. Keluar dari aplikasi *Delphi*.

2. Edit

Dipakai untuk menyuting program.

3. Search

Dipakai untuk mencari dan mengganti kata-kata pada saat menyuting program.

4. ***View***

Dipakai untuk menampilkan atau menyembunyikan jendela-jendela tertentu, misalnya *object inspector, code explorer, debug* dan lain-lain.

5. ***Project***

Dipakai untuk mengelola *project. Form* dapat ditambah dan dibuang dari *object*, mengkompilasi *project* dan lain-lain.

6. ***Run***

Menu ini dipakai untuk menjalankan program dan memantau jalannya program. Pada saat di-*run*, apabila terjadi salah tulis akan dapat diketahui.

7. ***Component***

Dengan menu ini komponen baru dapat ditambah atau di-*instal*.

2.9. *Microcontroller* AT89S51

2.9.1. Pendahuluan

Microcontroller bisa dipandang sebagai sebuah mini komputer yang terintegrasi dalam sebuah *chip*. Didalam satu *chip microcontroller* sudah terdapat bagian-bagian seperti dalam sebuah komputer. Bagian-bagian itu antara lain ; ALU (*Arithmetic Logic Unit*), PC (*Program Counter*), SP (*Stack Pointer*), Register, ROM (*Read Only Memory*), RAM (*Random Acces Memory*), Paralel I/O, Serial I/O, *Counter* dan sebuah rangkaian *Clock*.

Seperti sebuah *microprocessor*, *microcontroller* adalah sebuah perangkat serbaguna, yang fungsi kerjanya dapat ditentukan melalui sebuah perangkat lunak yang mendeskripsikan sebuah sistem yang diinginkan.

Pada saat ini terdapat banyak keluarga *microcontroller* salah satunya adalah keluarga MCS51. Salah satu tipe *microcontroller* yang termasuk dalam keluarga MCS51 adalah AT89S51 buatan *Atmel*.

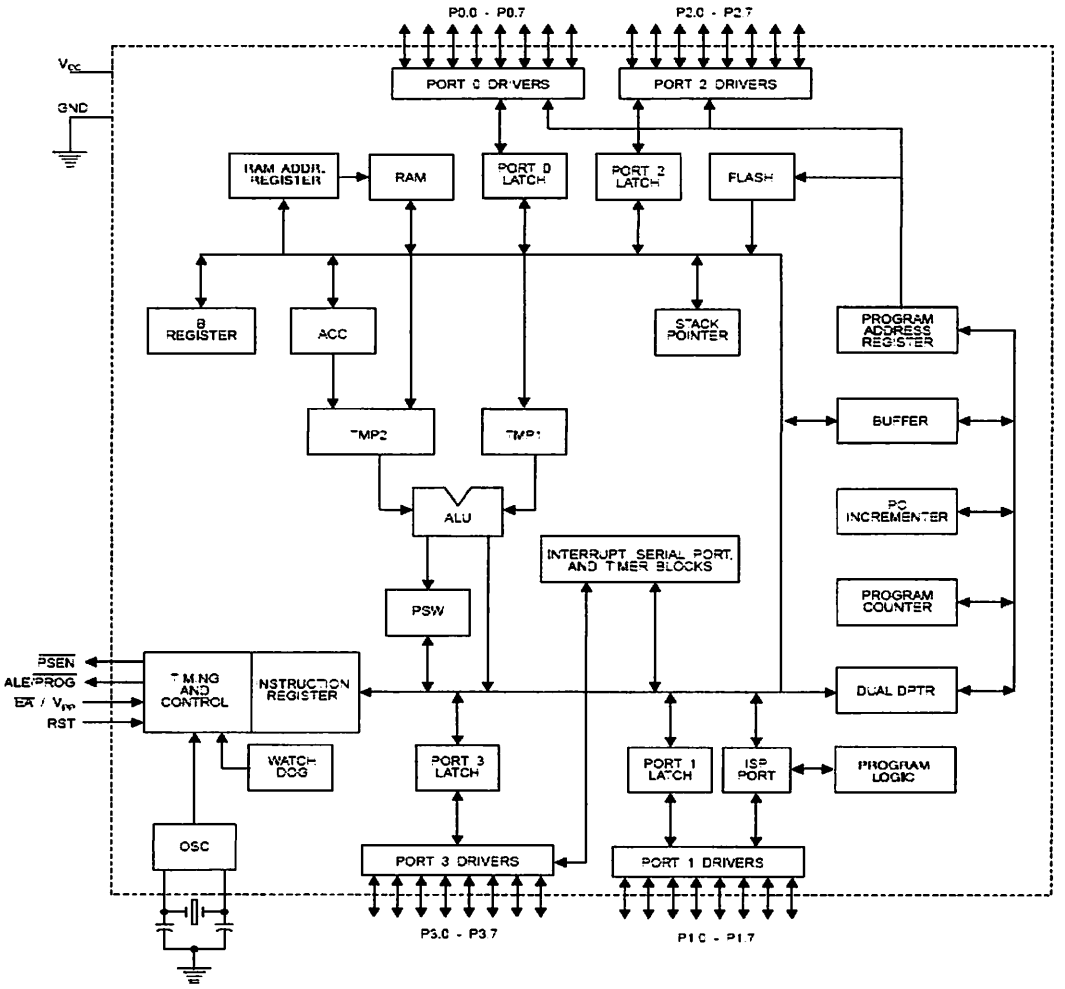
AT89S51 adalah *microcontroler* keluaran *atmel* dengan 4K *byte Flash* PEROM (*Programmable and Erasable Read Only Memory*), AT89S51 merupakan memori dengan teknologi *nonvolatile memory,* artinya isi memori tersebut dapat diisi ulang ataupun dihapus berulang kali.

Memori ini biasa digunakan untuk menyimpan instruksi (Perintah) berstandar MCS – 51 code sehingga memungkinkan mikrokontroler ini untuk bekerja dalam mode *Single Chip Operation* (Mode Operasi Keping Tunggal) yang tidak memerlukan *Eksternal Memori* (Memori luar) untuk menyimpan *source code* tersebut.

2.9.2. Arsitektur AT89S51

Microcontroller AT89S51 secara umum memiliki:

- CPU 8bit
- *Memory*
- Port I/O yang dapat diprogram
- *Timer* dan *Counter*
- Sumber *Interupt*
- Program serial yang dapat diprogram
- *Osilator* dan *clock*



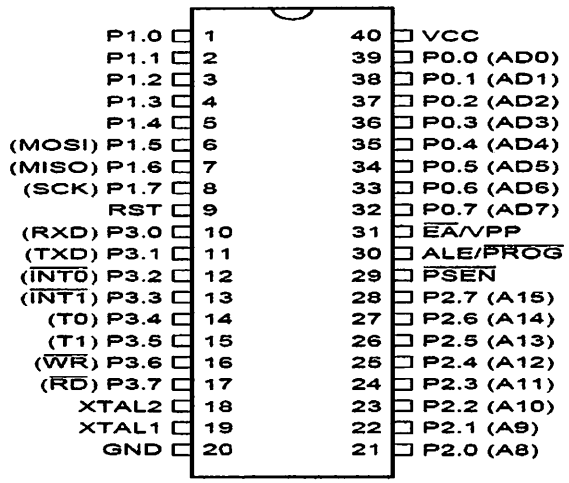
Gambar 2-12. Blok Diagram AT89S51[9]

Arsitektur *microcontroller* AT89S51 adalah sebagai berikut:

1. CPU (Central Processing Unit) 8-bit dengan register A (*Accumulator*)dan B.
2. 16-bit program *counter* (PC) dan data *pointer* (DTPR).
3. 8-bit program status *word* (PSW).
4. 4-bit *stack pointer* (SP).
5. 5 K*byte* internal EPROM.
6. 128 *Byte internal* RAM
   - 4 bank register, masing- masing berisi 8 register.
   - 16 *Byte* yang dapat dialamati pada bit *level.*
   - 80 *Byte general purpose memory* data.
7. 32 pin *input- output* tersusun atas PO-P3. masing-masing 8-bit.
8. 2 buah 16-bit *Timer/counter.*
9. *Receiver/Tansmiter* data serial *Full Duplex*: SBUF.
10. Control Register, yaitu: TCON, TMOP, SCON, PCON, PCON, IP, dan IE.
11. 5 buah sumber interupsi (2buah sumber *interrup external* dan 3 buah sumber *interup internal*).
12. *Osilator* dan *Clock Internal.*

2.9.3. Konfigurasi Pin - Pin Mikrokontroller AT89S51

Untuk konfigurasi dari pin-pin kaki IC AT89S51 dapat dilihat pada gambar 2-2 dibawah ini.



Gambar 2-13. Pin – Pin AT89S51[9]

Fungsi dari tiap- tiap *port* adalah sebagai berikut.

1. VCC (*Suplay* tegangan).
2. GND (*Ground*).
3. *PORT* 0

   Merupakan *port input- output* dua arah dan dikonfigurasikan sebagai *multipleks* dua *bus* alamat rendah (A0 – A7) dan data selama pengaksesan program memori dan data memori *external.*
4. *PORT* 1

   Merupakan port input – output dengan *pull up internal* yang dapat dipergunakan untuk berganti keperluan. Port 1 juga menerima alamat *byte* rendah selama program EPROM maupun ROM didalamnya
5. *PORT* 2

   Merupakan *port input – output* dengan *pull up internal. Port* 2 juga merupakan *bus* alamat *byte* tinggi selama adanya akses ke memori program luar dan selama mengakses memori data *eksternal* yang menggunakan pengalamatan 16 bit. MOVX @DPTR digunakan dalam pengaksesan memori

yang menggunakan pengalamatan 8 bit (MOVX @ R1), *port* 2 mengeluarkan isi dari P2 SFR (*Special Function Generator*). *Port* 2 juga dapat berfungsi mengeluarkan sebagai *bus* alamat atas selama pemrograman EPROM dan saat verifikasi ROM dan EPROM pada mikrokontroller yang dilengkapi ROM atau EPROM dikepingnya.

6. *PORT* 3

   Merupakan *port input-output* dengan *internal pull-up*.

   Port 3 juga memiliki fungsi khusus, yaitu:

   - RXD (P3.0) : *Port input* serial.
   - TXD (P3.1) : *Port output* serial.
   - INT0 (P3.2) : *Interupt* 0 *external.*
   - INT1 (P3.3) : *Interupt 1 external.*
   - T0 (P3.4) : *Input external timer* 0.
   - T1 (P3.5) : *Input external timer* 1.
   - WR (P3.6) : *Strobe* tulis data *memory external.*
   - RD (P3.7) : *Strobe* baca data *memory external.*

7. RST

   Kondisi *high* pada pin ini selama dua siklus ketika *osilator* bekerja akan mereset *microcontroller*.

8. ALE/ PROG

   Pulsa output ALE digunakan untuk proses *'latching' byte address* rendah (A0-A7) selama pengaksesan ke *external memory*. Pin ini juga digunakan untuk memasukkan pulsa program (prog) selama pemprograman.

9. PSEN

PSEN (*Program Store Enable*) merupakan sinyal baca untuk memori program *eksternal.*

10. EA/VPP

External address enable EA di*ground*kan jika mengakses *memory external.* Untuk mengakseskan *memory internal* maka dihubungkan ke VCC.

11. X-TALL1 dan X-TALL2

Kaki ini dihubungkan dengan kristal bila menggunakan *osilator internal.* X-TALL1 merupakan *input inverting osilator amplifier* sedangkan X_TALL2 merupakan *output inverting osilator amplifier.*

2.9.4. Organisasi *Memory*

Di dalam AT89S51 ruang alamat telah dibedakan untuk program *memory* dan data *memory.*

2.9.4.1. Program *Memory Internal*

AT89S51 memiliki program *memory internal* sebesar 4 *Kbyte* dengan ruangan alamat 0000H- 0FFFH. Jika alamat-alamat program lebih tinggi dari pada 0FFFH, yang melebihi kapasitas ROM *internal* menyebabkan AT89S51 secara otomatis mengambil *code byte* dari program *memory external. Code byte* juga dapat diambil hanya dari *external memory* dengan alamat 0000H-FFFFH dengan cara menghubungkan pin EA ke *ground.*

2.9.4.2. Data Memory (RAM) Internal

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

- Empat *bank* register

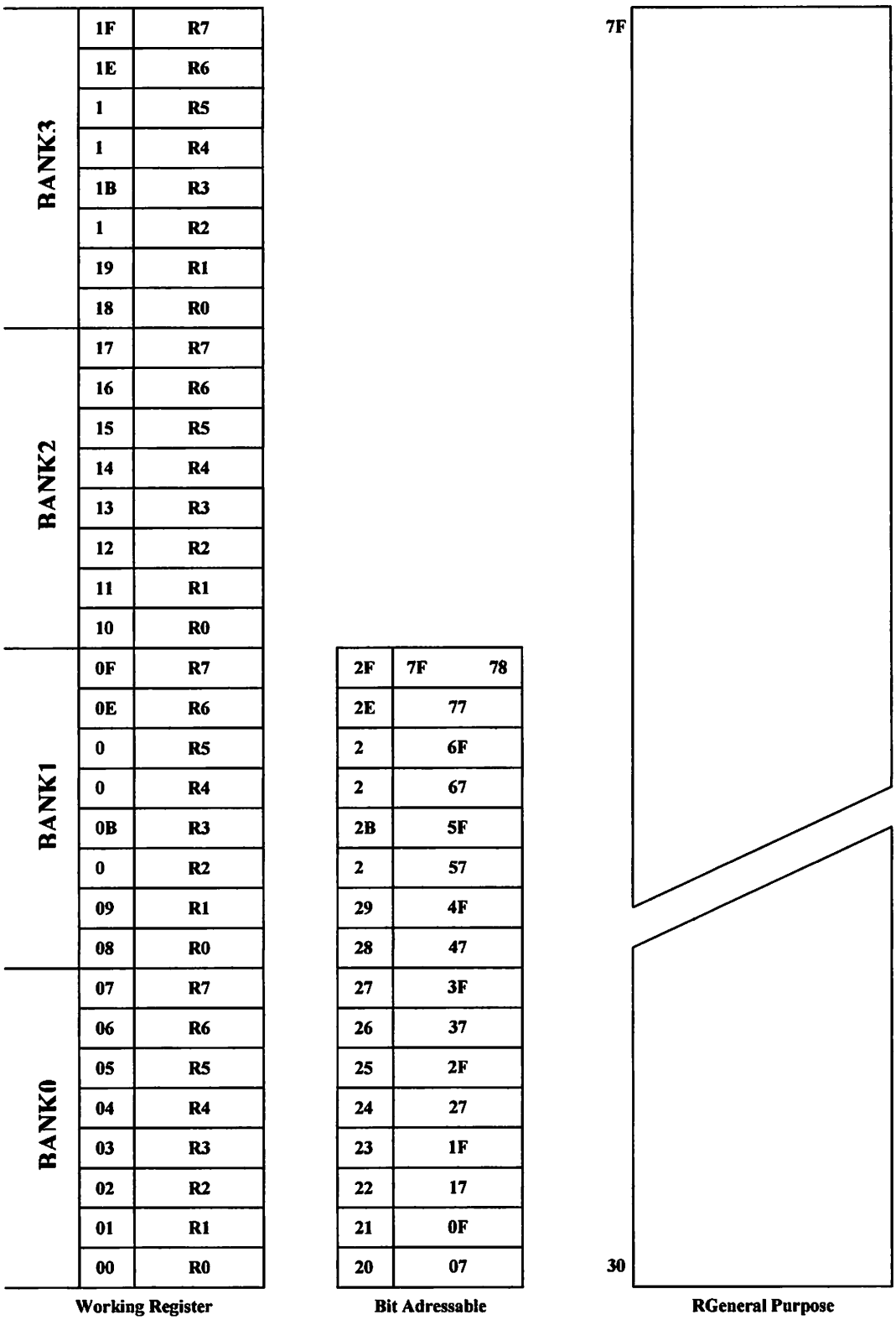
  Setiap *bank* terdiri dari 8 register (R0-R7) sehingga jumlah register untuk keempat *bank* register menjadi 32 buah register yang menempati ruang alamat 00H-1FH. Mengaktifkan salah satu *bank* register dapat dilakukan dengan mengatur RS0-RS1 pada program *status word* (PSW).

- Bit *Addressable*

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

- *Stratc Pad Area*

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



Gambar 2-14. Ilustrasi Pembagian Ruang RAM AT89S51[9]

Tabel 2 – 4. Pengaturan RS0-RS1 *Select Register Ban* [9]

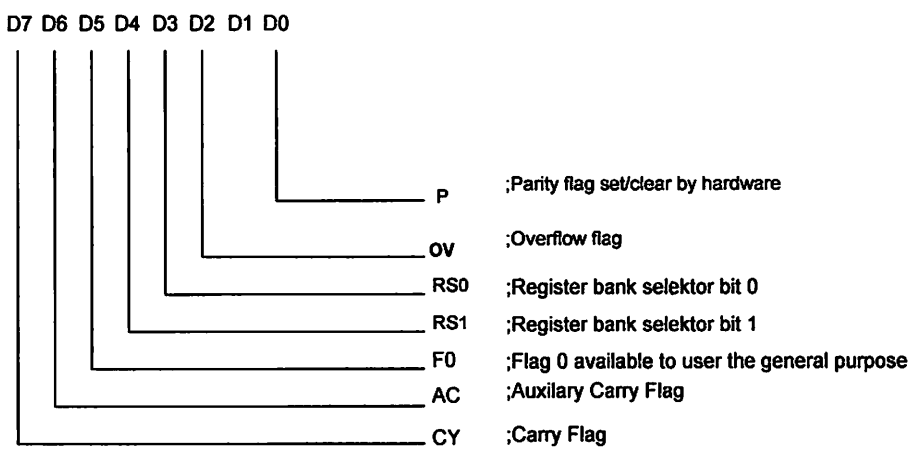
RS1	RS0	Select Register Bank
0	0	Bank 0
0	1	Bank 1
1	0	Bank 2
1	1	Bank 3

2.9.4.3. SFR (*Special Function Register*)

Untuk operasi AT89S51 yang tidak menggunakan alamat *internal* RAM (00H-7FH) dilakukan oleh SFR yang ber*address* 80H- FFH, tetapi tidak semua *address* tersebut digunakan sebagai SFR, hanya *address* tertentu saja.

2.9.4.4. Program Status *Word*

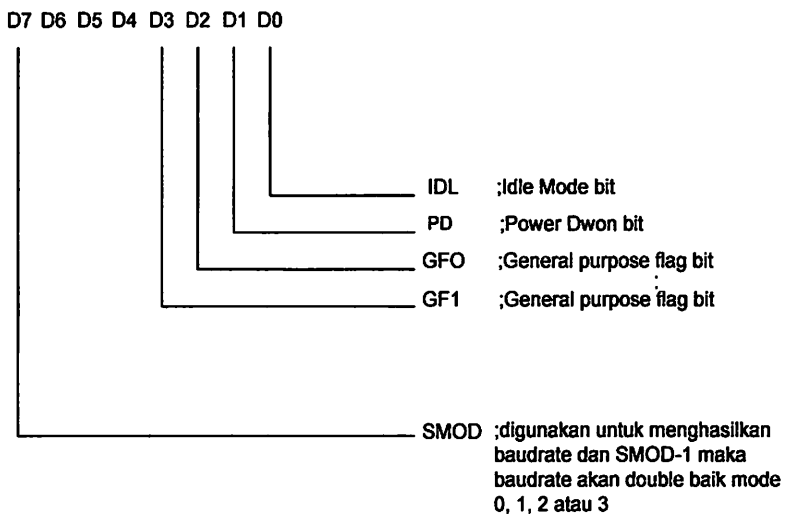
Register ini terletak di alamat D0H. Cara mendefenisikannya adalah sebagai berikut:



Gambar 2-15. Skema Medefinisikan PSW[9]

2.9.4.5. PCON (*Power Control*)

Register ini terletak pada alamat 87H. Cara mendefinisikan adalah sebagai berikut:



Gambar 2-16. Skema Mendefinisikan PCON[9]

2.9.4.6. Sistem *Interupt*

Microcontroller AT89S51 mempunyai 5 buah sumber interupt yang dapat mengakibatkan permintaan interupt, yaitu: INT0, INT1, T1, T2 dan *port* serial.

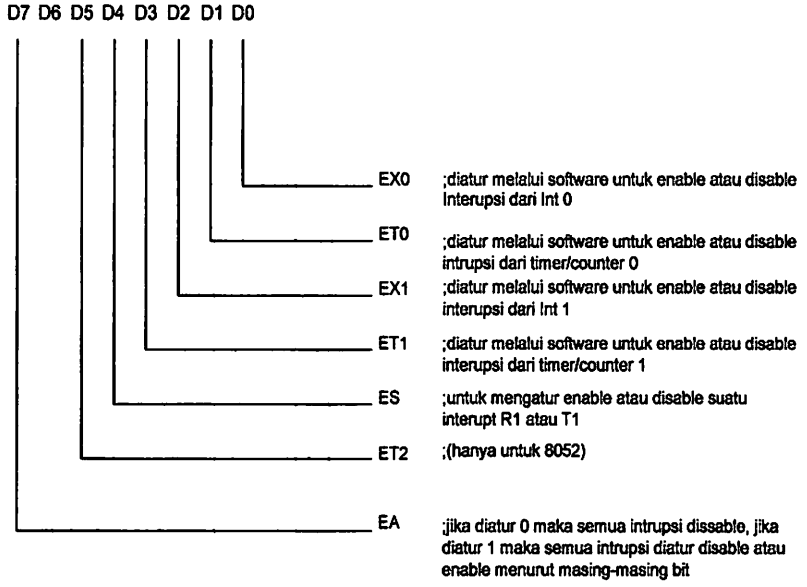
Saat terjadi interup *microcontroller* secara otomatis akan menuju ke subrutin pada alamat tersebut. Setelah interupt *service* selesai dikerjakan, mikrokontroller akan mengerjakan program semula. Dua sumber external adalah INT0, INT1, dimana kedua interupsi *eksternal* akan aktif atau aktif transisi tergantung isi IT0 dan IT1 pada register TCON. Interupsi T0 dan T1 aktif pada saat timer yang sesuai mengalami *roll over*. Interupsi serial dibangkitkan dengan melakukan operasi OR pada R1 dan T1. Tiap – tiap sumber interupsi dapat *enable* atau *disable* secara *software.*

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

Tabel 2 – 5. Alamat Sumber Interupsi[9]

Sumber Interupt	Alamat Awal
Power On Reset	0000h
Interupt Luar 0 (INT 0)	0003h
Pewaktu/ Pencacah 0 (T0)	000Bh
Interup Luar 1 (INT 1)	0013h
Pewaktu/ Pencacah 1 (T1)	001B
Port I/O Serial	0023h

Register yang berperan dalam mengatur aktif tidaknya interupsi adalah *interrupt enable register*, berikut ini adalah susunan dari bit-bit beserta kegunaanya:



Gambar 2-17. Kegunaan *Interupt Enable Register*[9]

2.9.4.7. *Timer/ Counter*

Pengendalian kerja dari *timer/counter* dilakukan dengan pengaturan register yang berhubungan dengan kerja dari *Timer/counter* yaitu melalui sebuah *timer/counter mode control.*

Untuk mengaktifkan *timer/counter* yang meliputi penentuan fungsi sebagai *timer* atau sebagai *counter* serta pemilihan *mode* operasi dapat diatur melalui TMOD yang beralamat pada 89H.

- *Gate*

  Bila *gate* = 1, *Timer/counter* x *enable* hanya pada saat pin INTx tinggi dan TRX 1,. Saat *gate* 0, *timer/counter enable* jika bit TRx 1

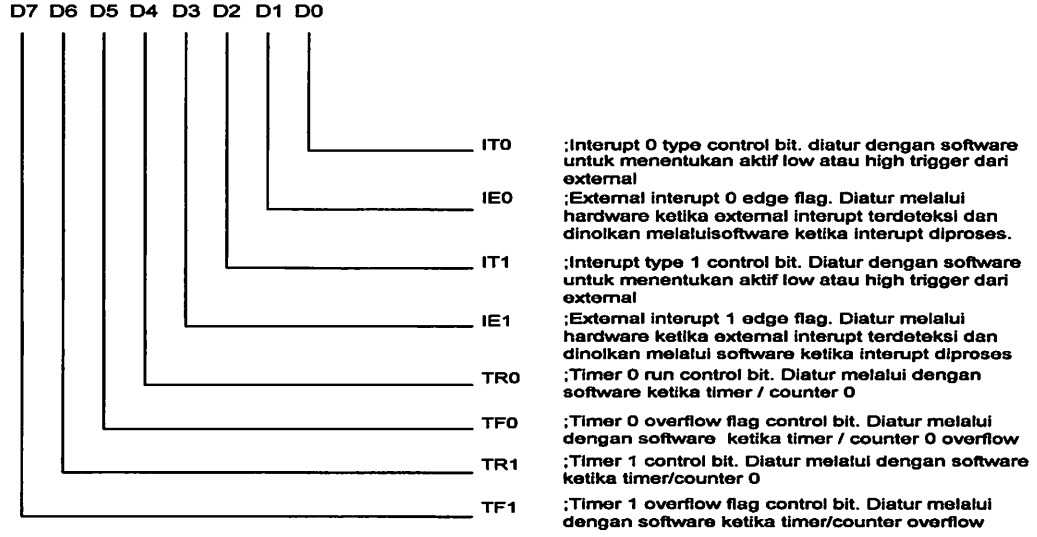
- CT

  Jika bit C/T = 0, maka *timer/counter* x akan berfungsi sebagai *timer*. Jika C/T = 1, maka *timer/counter* x akan beroperasi sebagai *counter.*

- M1 dan M2

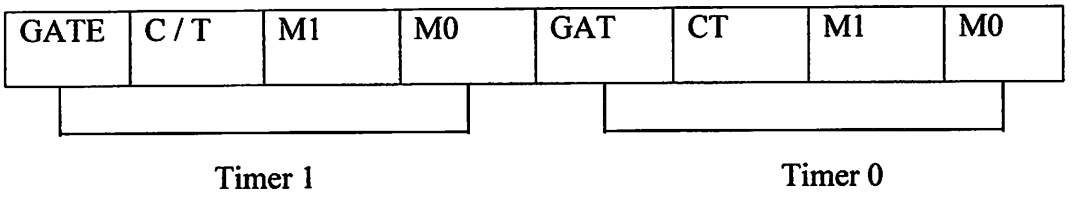
  Menentukan *mode.*

Kofigurasi yang dimaksud adalah sebagai berikut:



Gambar 2-18. Konfigurasi dan Guna TMOD[9]

TMOD : Timer/Counter Mode Control Register



Tabel 2 – 6. *Mode Operasi Timer/Counter*[9]

M1	M0	*Operating Mode*
0	0	*Timer* 13 bit
0	1	*Timer/Counter* 16 bit
1	0	8 bit *Auto reload Timer /Counter*
1	1	TL0 dari *Timer* adalah 8 Bit *Timer/Counter* dikendalikan oleh *kontrol bit timer* 0. TH0 adalah 8 bit yang dikendalikan oleh *Timer* 1 *control bit*

2.9.4.8. Metode Pengalamatan

1. Pengalamatan Bit (*Direct Bit Addressing*)

   Pengalamatan langsung tiap bit ini hanya dilakukan pada lokasi RAM *internal* yaitu 20H-2FH, d *port* 1, *port* 2 *port* 3, TCON register, SCON register, IE register, PSW register, ACC dan ACC dan B register.

2. Pengalamatan Tak Langsung

   Pada pengalamatan tak langsung, instruksi menunjukkan suatu register yang isinya adalah alamat dari *operand, eksternal* dan *internal* RAM dapat

dapat berupa R0 dan R1 yang digunakan untuk memilih angka register atau *stack pointer*. Register alamat untuk data, dengan lebar 16 bit digunakan data *pointer* DPTR.

3. Pengalamatan Ber*indeks*

   Yang dapat diakses dengan pengalamatan ber*indeks* hanya memori program. *Mode* ini dimaksudkan untuk membaca *look-up table program*.

4. *Konstanta Immediat*

   Pengalamatan langsung dilakukan dengan memberikan nilai ke suatu register secara langsung, dilakukan dengan menggunakan tanda #.

   Contoh: Mov a# 100

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Bab ini membahas mengenai perancangan alat *data logger* suhu dan kelembaban menggunakan dua buah MC AT89S51 (sebagai pengambil data dari sensor SHT11) dan sebuah MC AT89S8252 (sebagai penyimpan data selama maksimal 7 hari) berbasis PC. Secara garis besar, perancangan alat meliputi dua bagian yaitu:

1. Perancangan perangkat keras *(hardware)*
2. Perancangan perangkat lunak *(software)*

3.1. Gambaran Umum

Sistem kerja dari alat ini merupakan sebuah sistem pengambilan data (*data logging*) dari sensor suhu dan kelembaban (SHT11) menggunakan dua buah MC AT89S51 (sebagai pengambil data dari sensor) dan sebuah MC AT89S8252 (sebagai penyimpan data) berbasis PC atau *laptop*. Prinsip kerjanya secara garis besar adalah, kedua SHT11 (baik yang berada di dalam maupun di luar ruangan) setiap saat akan mendeteksi suhu dan kelembaban. Setiap satu jam, dua buah MC AT89S51 masing-masing akan mengambil data hasil pendeteksian tersebut dan mengirimnya ke MC AT89S8252 untuk disimpan selama maksimal 7 hari.

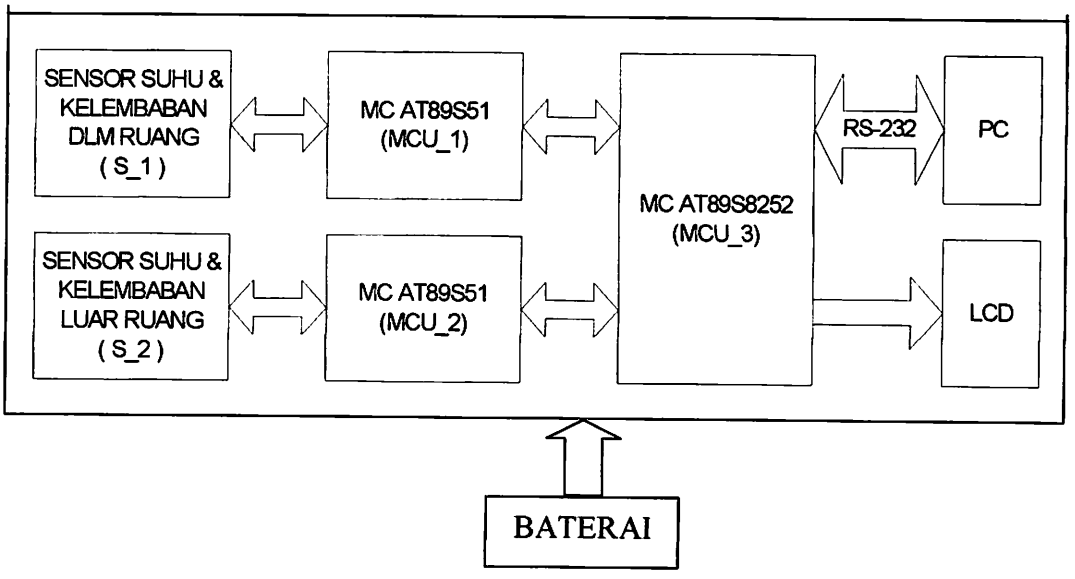
Sewaktu-waktu *user* bisa mengambil data dari MC AT89S8252 menggunakan PC atau *laptop* dan menyimpannya sebagai *database* sehingga nantinya dapat diolah oleh para arsitek sebagai bahan referensi dalam

perancangan suatu bangunan. Data yang bisa diambil oleh *laptop* adalah antara data hari ke-1 sampai dengan maksimal data hari ke-7.

3.2. Perancangan Perangkat Keras

3.2.1. Blok Diagram Perancangan Alat

Adapun blok diagram dari perancangan alat ini secara garis besar adalah :



Gambar 3-1. Diagram Blok Keseluruhan Sistem

Penjelasan Blok Diagram

1. MC AT89S8252 (MCU_3)

Berfungsi sebagai pengontrol utama alat serta penyimpan sementara data suhu dan kelembaban dari MC AT89S51. MC AT89S8252 mempunyai EEPROM sebesar 2KB.

2. MC AT89S51 (MCU_1 & MCU_2)

Digunakan dua buah MC AT89S51 yang masing-masing berfungsi sebagai pengambil data suhu dan kelembaban dari sensor suhu dan kelembaban (SHT11) untuk kemudian mengirimkannya ke MC AT89S8252 untuk disimpan selama 7 hari.

3. Sensor Suhu dan Kelembaban (SHT11)

Sensor suhu dan sensor kelembaban berada dalam 1 paket sensor bernama SHT11, dimana akan digunakan 2 buah SHT11 yaitu satu berada di dalam ruang dan satu lagi berada di luar ruang. Masing-masing berfungsi untuk mendeteksi kondisi suhu dan kelembaban di dalam serta di luar ruang. SHT11 ini selain sudah dikalibrasi oleh pabriknya, keluarannya juga sudah *digital* jadi tidak perlu menggunakan ADC lagi.

4. PC

Berfungsi untuk mengambil data suhu dan data kelembaban serta menyimpannya sehingga nantinya dapat diolah oleh pengguna, dalam hal ini oleh para arsitek.

5. LCD

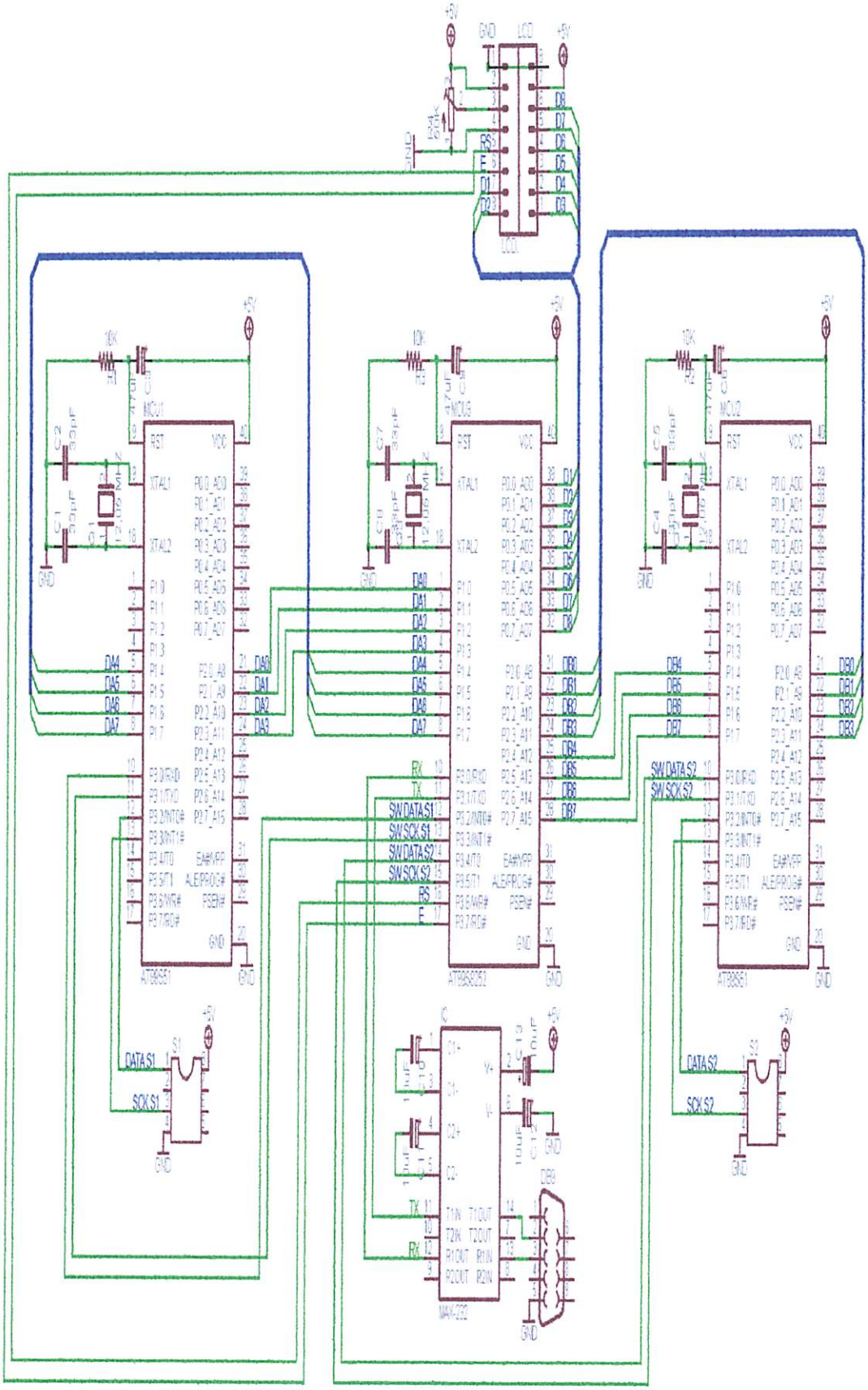
Berfungsi untuk menampilkan data suhu dan kelembaban yang sifatnya *real time*.

6. RS-232

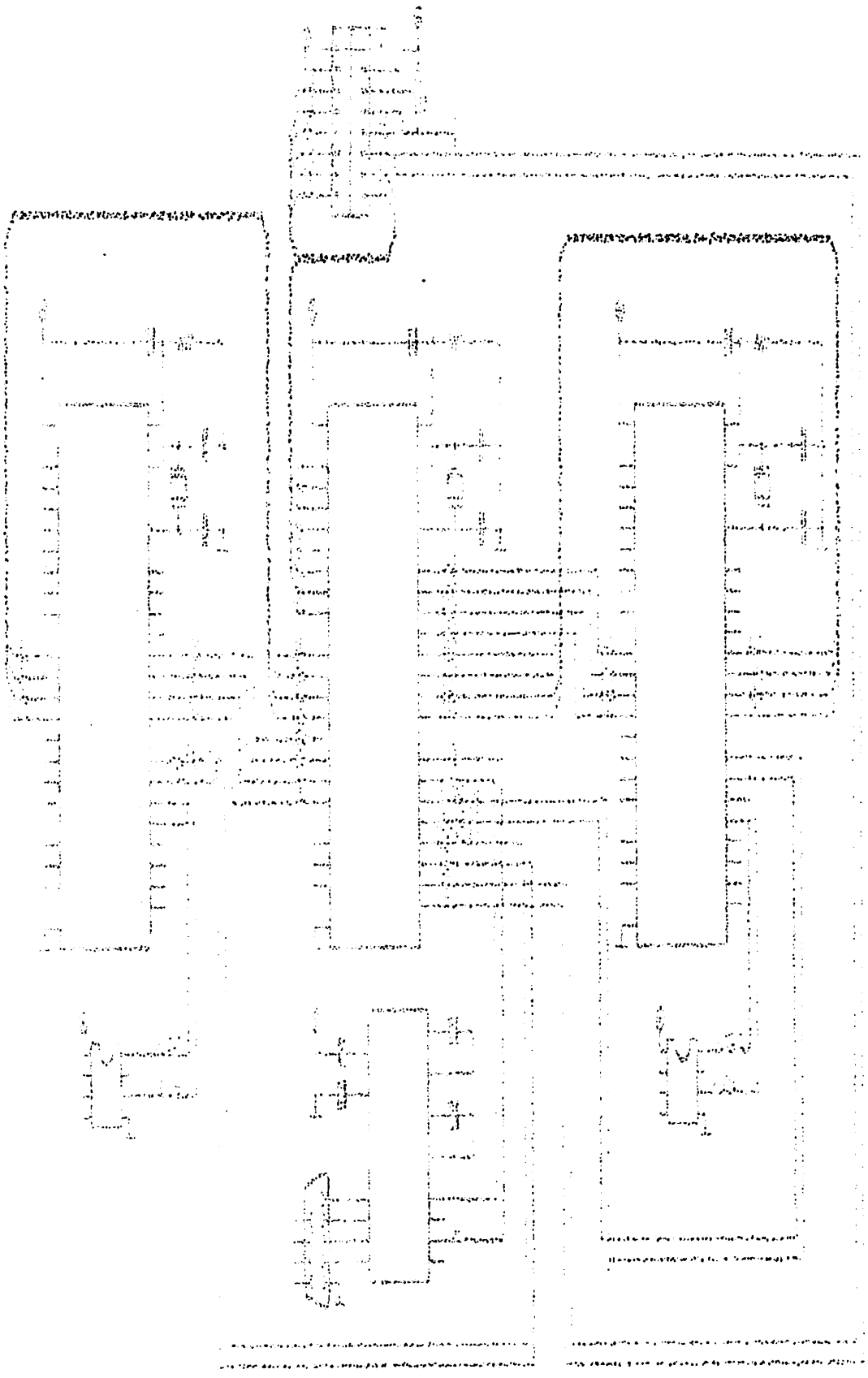
Berfungsi sebagai penghubung antara MC AT89S8252 dengan PC.

7. Baterai

Fungsinya adalah sebagai penyedia daya cadangan bagi rangkaian ketika listrik mati sehingga rangkaian tetap dapat bekerja.



Gambar 3-2. Gambar Perancangan Alat



3.2.2. Prinsip Kerja Sistem

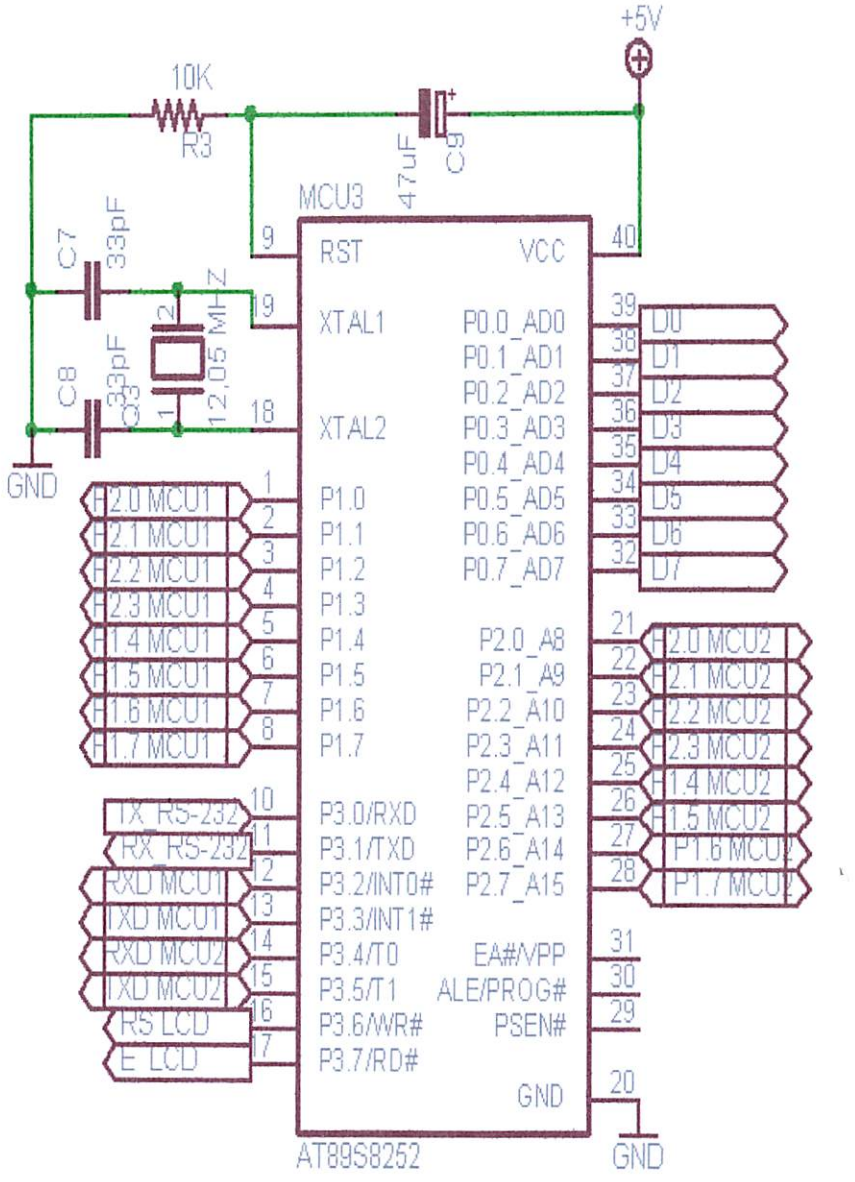
Prinsip kerjanya secara garis besar adalah, kedua SHT11 (baik yang berada di dalam maupun di luar ruangan) setiap saat akan mendeteksi suhu dan kelembaban. Setiap satu jam, dua buah MC AT89S51 masing-masing akan mengambil data hasil pendeteksian tersebut dan mengirimnya ke MC AT89S8252 untuk disimpan selama maksimal 7 hari.

Sewaktu-waktu *user* bisa mengambil data dari MC AT89S8252 menggunakan PC atau *laptop* dan menyimpannya sebagai *database* sehingga nantinya dapat diolah oleh para arsitek sebagai bahan referensi dalam perancangan suatu bangunan. Data yang bisa diambil oleh *laptop* adalah antara data hari ke-1 sampai dengan maksimal data hari ke-7.

3.2.3. Perancangan Rangkaian MC AT89S8252

Rangkaian MC AT89S8252 di sini berfungsi menerima data suhu dan kelembaban dari kedua MC AT89S51 serta menyimpannya selama aksimal 7 hari. Agar dapat melakukan prosesnya, MC AT89S8252 harus didukung oleh beberapa komponen tambahan yaitu berupa rangkaian *clock* dan *reset.*

Penentuan *port-port* dan sinyal-sinyal yang digunakan untuk mendukung proses yang akan dilakukan adalah sangat penting. MC AT89S8252 adalah suatu *chip* IC yang terdiri dari 40 *pin.* Gambar perancangan alat beserta *pin-pin* yang digunakan adalah sebagai berikut.



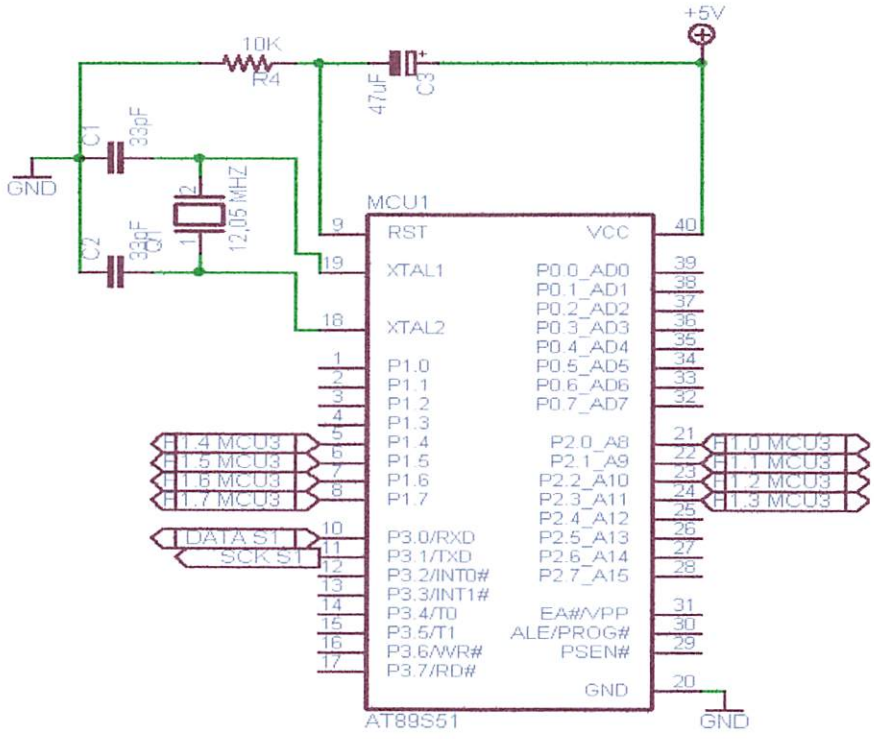
Gambar 3-3. Perancangan Rangkaian MC AT89S8252 (MCU_3)

1. *Pin* 9 (*RESET*) berfungsi sebagai *Reset.*
2. *Pin* 10 (*Port* 3.0 / RXD) sebagai masukan yang dihubungkan dengan keluaran dari RS-232.
3. *Pin* 11 (*Port* 3.1 / TXD) sebagai keluaran yang dihubungkan dengan masukan dari RS-232.

5. *Pin* 13 dihubungkan dengan *pin* 3 dari sensor 1.
6. *Pin* 14 dihubungkan dengan *pin* 2 dari sensor 2.
7. *Pin* 15 dihubungkan dengan *pin* 3 dari sensor 2.
8. *Pin* 18 (XTAL 2) sebagai pembangkit *ossilator (clock)* XTAL 2.
9. *Pin* 19 (XTAL 1) sebagai pembangkit *ossilator* (*clock*) XTAL 1.
10. *Pin* 20 (GND) sebagai *ground.*
11. *Pin* 21 dihubungkan dengan *pin* 21 pada sensor 1.
12. *Pin* 22 dihubungkan dengan *pin* 22 pada sensor 1.
13. *Pin* 23 dihubungkan dengan *pin* 23 pada sensor 1.
14. *Pin* 24 dihubungkan dengan *pin* 24 pada sensor 1.
15. *Pin* 25 dihubungkan dengan *pin* 5 pada sensor 1.
16. *Pin* 26 dihubungkan dengan *pin* 6 pada sensor 1.
17. *Pin* 27 dihubungkan dengan *pin* 7 pada sensor 1.
18. *Pin* 28 dihubungkan dengan *pin* 8 pada sensor 1.
19. *Pin* 32 – 39 dihubungkan dengan LCD.
20. *Pin* 40 untuk VCC + 5 *volt.*

3.2.4. Perancangan Rangkaian MC AT89S51 (MCU_1)

MCU_1 di sini berfungsi sebagai pengambil data dari sensor 1. Berikut ini adalah perancangan rangkaiannya.



Gambar 3-4. Gambar Perancangan Rangkaian MCU_1

1. *Pin* 5 dihubungkan dengan *pin* 5 pada MCU_3.

2. *Pin* 6 dihubungkan dengan *pin* 6 pada MCU_3.

3. *Pin* 7 dihubungkan dengan *pin* 7 pada MCU_3.

4. *Pin* 8 dihubungkan dengan *pin* 8 pada MCU_3.

5. *Pin* 10 digunakan untuk mengirim data suhu ke *pin* 12 pada MCU_3.

6. *Pin* 11 digunakan untuk mengirim data kelembaban ke *pin* 13 pada MCU_3.

7. *Pin* 12 digunakan untuk menerima data suhu dari *pin* 2 pada sensor 1.

8. *Pin* 13 digunakan untuk menerima data kelembaban dari *pin* 3 pada sensor 1.

9. *Pin* 18 (XTAL 2) sebagai pembangkit *ossilator (clock)* XTAL 2.

10. *Pin* 19 (XTAL 1) sebagai pembangkit *ossilator* (*clock*) XTAL 1.

11. *Pin* 21 dihubungkan dengan *pin* 1 pada MCU_3.

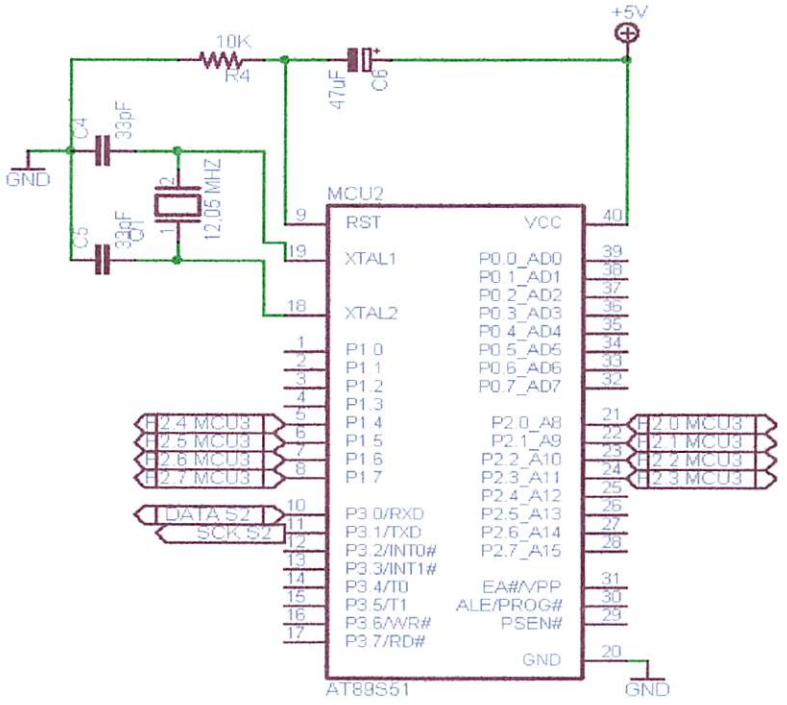
12. *Pin* 22 dihubungkan dengan *pin* 2 pada MCU_3.

13. *Pin* 23 dihubungkan dengan *pin* 3 pada MCU_3.

14. *Pin* 24 dihubungkan dengan *pin* 4 pada MCU_3.

3.2.5. Perancangan Rangkaian MC AT89S51 (MCU_2)

MCU_2 berfungsi sebagai pengambil data dari sensor 2.



Gambar 3-5. Gambar Perancangan Rangkaian MCU_2

1. *Pin* 5 dihubungkan dengan *pin* 25 pada MCU_3.

2. *Pin* 6 dihubungkan dengan *pin* 26 pada MCU_3.

3. *Pin* 7 dihubungkan dengan *pin* 27 pada MCU_3.

4. *Pin* 8 dihubungkan dengan *pin* 28 pada MCU_3.

5. *Pin* 10 digunakan untuk mengirim data suhu ke *pin* 14 pada MCU_3.

6. *Pin* 11 digunakan untuk mengirim data kelembaban ke *pin* 15 pada MCU_3.

7. *Pin* 12 digunakan untuk menerima data suhu dari *pin* 2 pada sensor 2.

8. *Pin* 13 digunakan untuk menerima data kelembaban dari *pin* 3 pada sensor 2.

9. *Pin* 18 (XTAL 2) sebagai pembangkit *ossilator (clock)* XTAL 2.

10. *Pin* 19 (XTAL 1) sebagai pembangkit *ossilator* (*clock*) XTAL 1.

11. *Pin* 21 dihubungkan dengan *pin* 21 pada MCU_3.

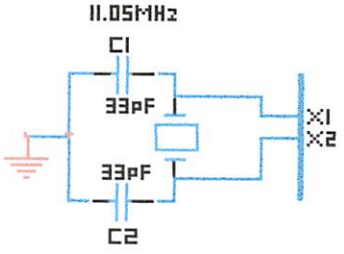
12. *Pin* 22 dihubungkan dengan *pin* 22 pada MCU_3.

13. *Pin* 23 dihubungkan dengan *pin* 23 pada MCU_3.

14. *Pin* 24 dihubungkan dengan *pin* 24 pada MCU_3.

3.2.6. Rangkaian *Clock* Minimum Sistem (X-TAL 1 dan X-TAL 2; pin 19 dan 18)

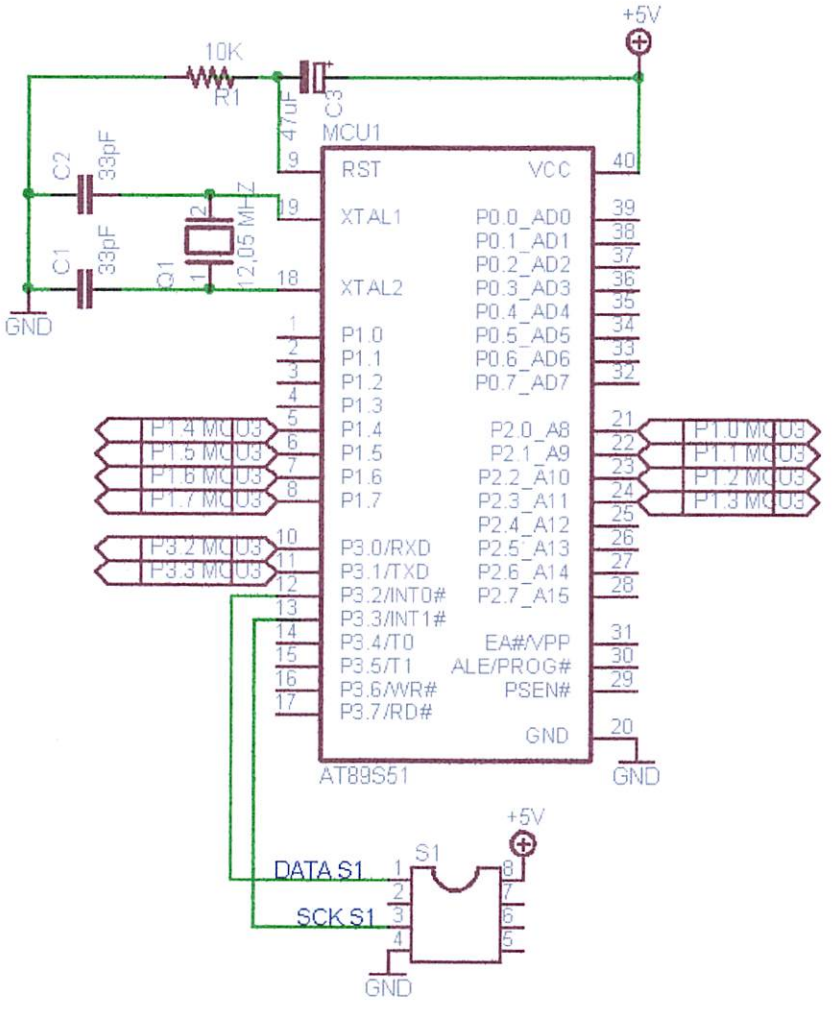
Pin ini dihubungkan dengan kristal bila menggunakan *osilator internal*. X-TAL 1 merupakan masukan ke rangkaian *osilator internal* sedangkan X-TAL 2 keluaran dari rangkaian *osilator internal*. Untuk keperluan ini diperlukan kapasitor penstabil sebesar 33pF. Dan nilai dari X-TAL tersebut antara 3–24 Mhz. Untuk lebih jelasnya dapat dilihat gambar pemasangan X-TAL serta kapasitor yang digunakannya.



Gambar 3-6. Rangkaian Pewaktuan dengan *Osilator Internal* [7]

3.2.7. Perancangan dan Analisa Rangkaian Sensor Suhu dan Kelembaban

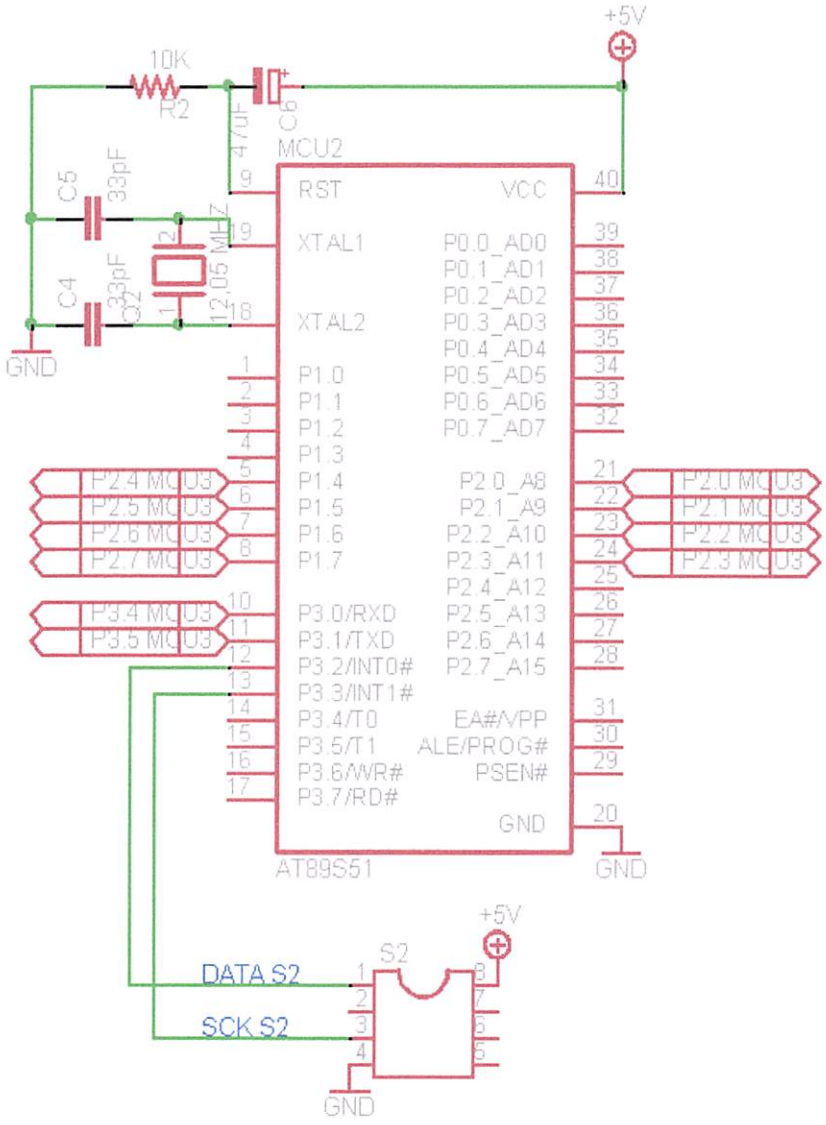
3.2.7.1. Perancangan Rangkaian Sensor_1



Gambar 3-7. Perancangan Rangkaian Sensor_1

1. *Pin* 4 (GND) sebagai *ground.*
2. *Pin* 3 (SCK) atau *clock* berfungsi mengirim data kelembaban ke *pin* 13 MCU_1.
3. *Pin* 2 (DATA) mengirim data suhu ke *pin* 12 MCU_1.
4. *Pin* 8 (VCC) adalah VCC +5 *volt.*

3.2.7.2. Perancangan Rangkaian Sensor_2



Gambar 3-8. Perancangan Rangkaian Sensor_2

1. *Pin* 4 (GND) sebagai *ground.*
2. *Pin* 3 (SCK) atau *clock* berfungsi mengirim data kelembaban ke *pin* 13 MCU_2.
3. *Pin* 2 (DATA) mengirim data suhu ke *pin* 12 MCU_2.
4. *Pin* 8 (VCC) adalah VCC +5 *volt.*

➢ **Perhitungan Kelembaban Relatif (RH) :**

RH linier = C1 + C2 x SO_{RH} + C3 x SO^2_{RH}

SO_{RH}	C_1	C_2	C_3
12 bit	-4	0.0405	$-2.8 * 10^{-6}$
8 bit	-4	0.648	$-7.2 * 10^{-4}$

Apabila suhu berbeda jauh dengan 25°C maka kita perlu menggunakan lagi koefisien kelembaban dengan formula sebagai berikut.

RH true = (T °C – 25) x (t1 + t2 x SO_{RH}) + RH linier

SO_{RH}	t_1	t_2
12 bit	0.01	0.00008
8 bit	0.01	0.00128

➢ **Perhitungan Suhu :**

Suhu = d1 + d2 x SO_T

VDD	d_1 [°C]	d_1 [°F]
5V	-40.00	-40.00
4V	-39.75	-39.55
3.5V(3)	-39.66	-39.39
3V(3)	-39.60	-39.28
2.5V(3)	-39.55	-39.19

SO_T	d_2 [°C]	d_2 [°F]
14bit	0.01	0.018
12bit	0.04	0.072

SORH , SOT merupakan data hasil pembacaan dari sensor.

Rumus-rumus di atas langsung diaplikasikan dalam program sehingga akan didapatkan hasil keluaran sensor yang sama dengan kondisi sebenarnya. Rumus-rumus di atas diformulasikan untuk kedua sensor.

3.2.8. Perancangan Komunikasi Serial antara Alat dengan PC

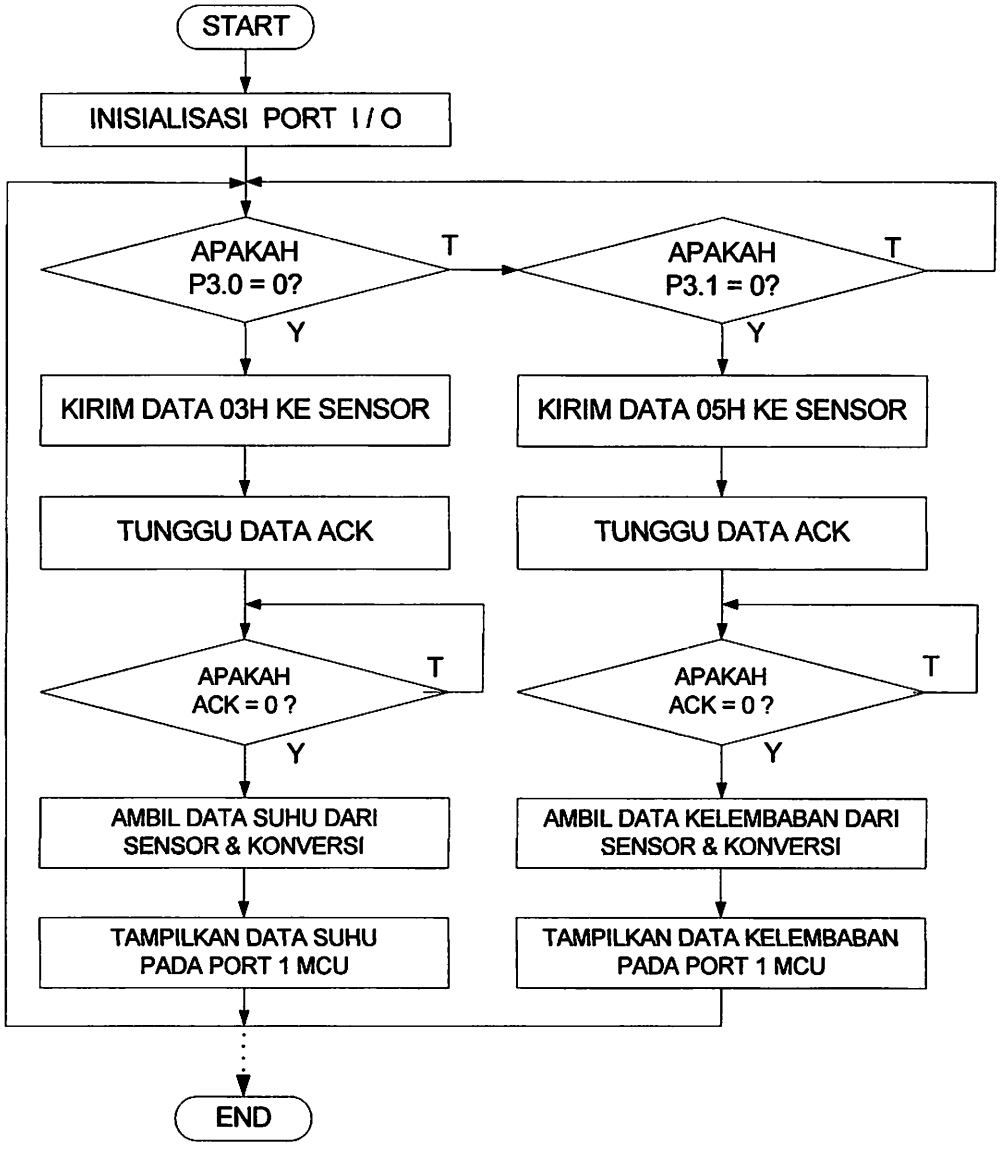
Komunikasi antara alat dengan PC akan dilakukan secara serial menggunakan *port* serial (*Com*) dari PC. Dikarenakan standar komunikasi serial dari PC menggunakan standar komunikasi RS-232 yang memiliki *range* logika tinggi (H) sebesar -15 *volt* sampai –3 *volt* dan logika rendah (L) sebesar 3 *volt* sampai 15 *volt*, sedangkan *microcontroller* menggunakan logika TTL yang memiliki logika tinggi (H) sebesar 3 *volt* sampai 5 *volt* dan logika rendah (L) 0 *volt* sampai 0,45 *volt*; maka dibutuhkan rangkaian penyesuai untuk kedua kondisi logika tersebut. Perangkat utama dari rangkaian penyesuai kondisi logika RS-232 dengan kondisi logika TTL adalah IC MAX-232.

3.3. Perancangan Perangkat Lunak (*Software*)

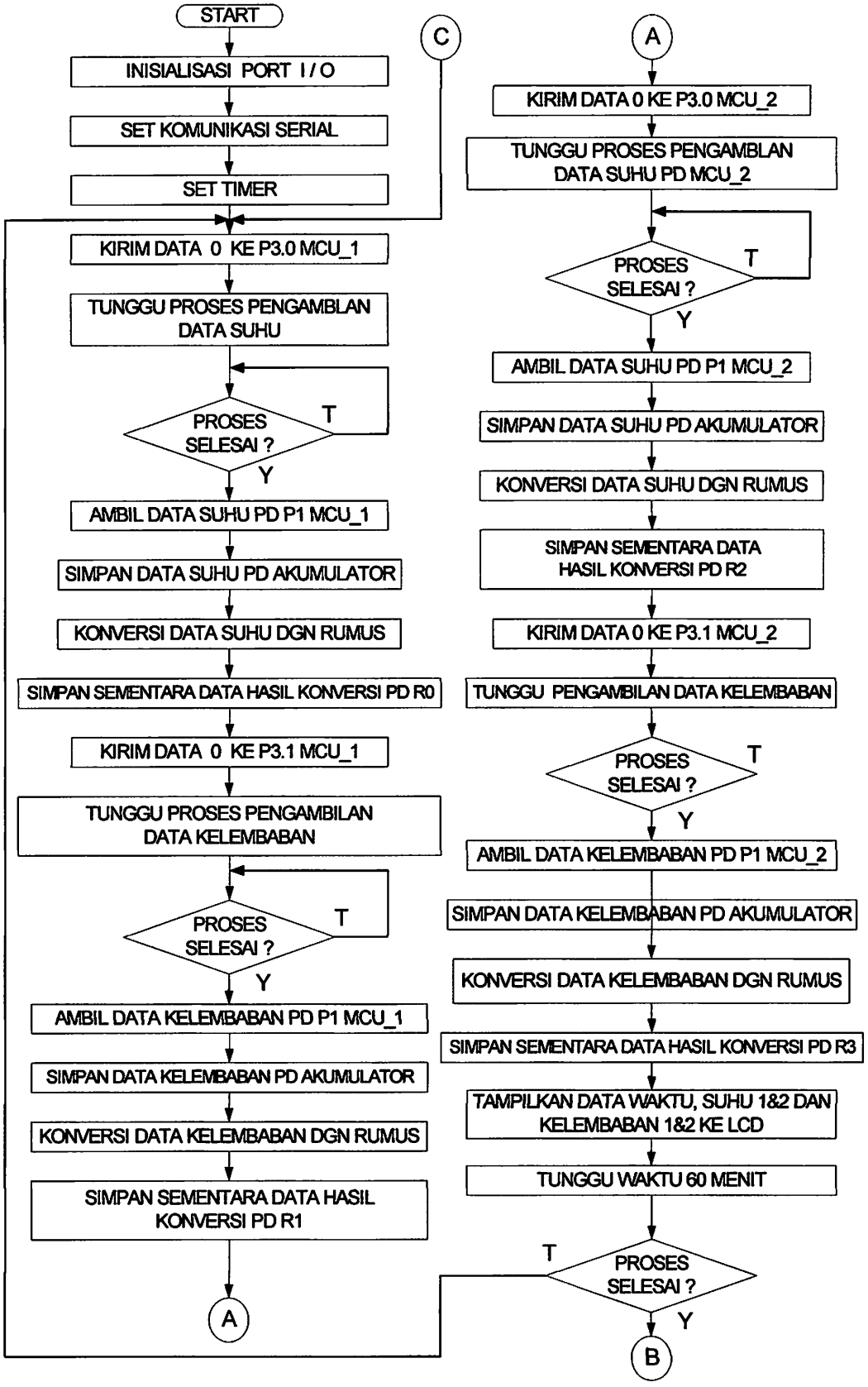
Untuk mendukung agar perangkat keras berfungsi sesuai dengan perencanaan, maka diperlukan perangkat lunak sebagai penunjangnya, yang berfungsi untuk mengatur dan mengendalikan keseluruhan sistem perangkat keras yang telah dibuat. Sebuah mikrokontrol tidak akan bekerja bila tidak diberikan program kepadanya. Program tersebut memberitahukan apa yang harus dilakukan oleh mikrokontrol. Sistem aplikasi mikrokontrol AT89S8252 ini dapat mengatur dan mengendalikan keseluruhan sistem apabila ada urutan instruksi yang mendefinisikan secara jelas urutan tugas yang harus dikerjakan. Urutan instruksi ini sangat penting untuk didefinisikan, karena mikrokontrol bekerja secara pasti berdasarkan urutan instruksi ini. Susunan logika perancangan yang salah tidak dapat diketahui oleh mikrokontrol. Selama instruksi yang diterima sesuai dengan aturannya, mikrokontrol akan tetap mengerjakan instruksi tersebut. Kesalahan

seperti ini baru diketahui ketika kerja sistem aplikasi tidak sesuai dengan spesifikasi awal. Oleh karena itu, perancangan perangkat keras sangat menentukan keberhasilan pembuatan perangkat lunak.

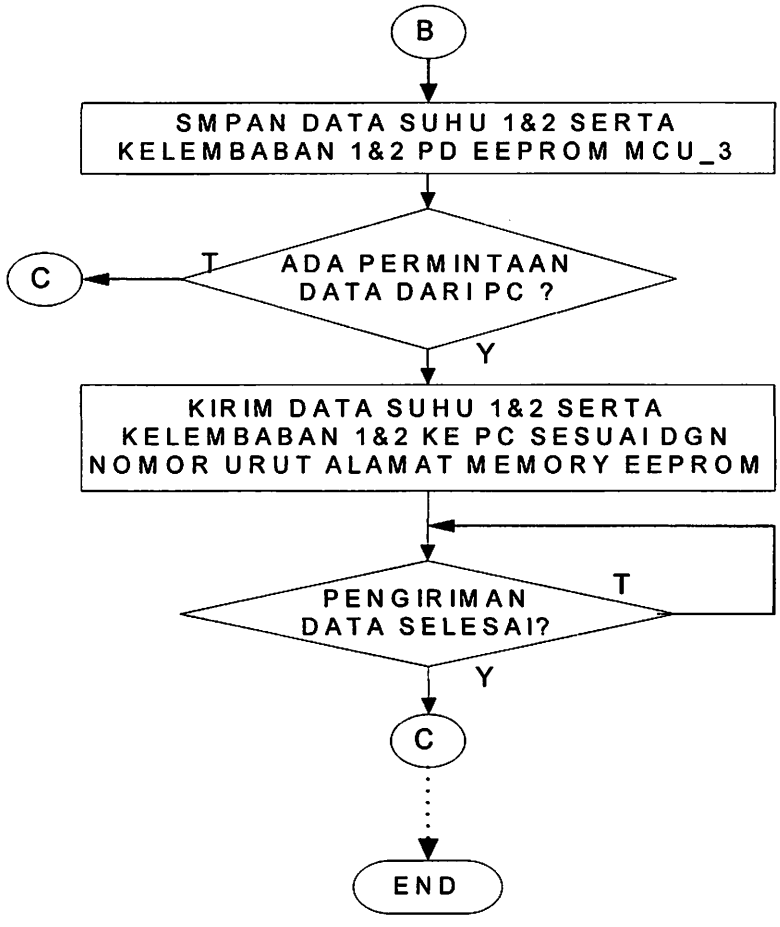
3.3.1. Perancangan *Software* pada *Microcontroller*



Gambar 3-9. *Flowchart Software* pada MCU_1 dan MCU_2

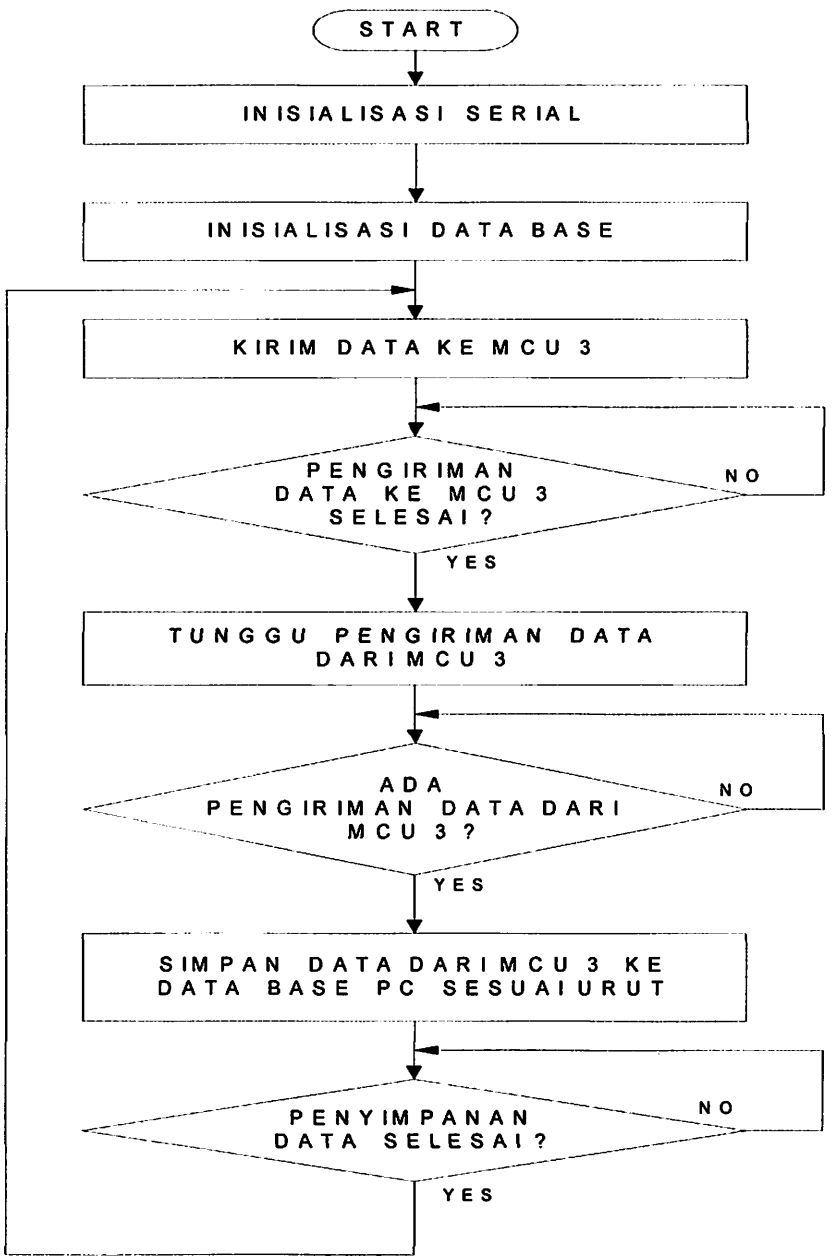


Gambar 3-10. *Flowchart Software* pada MCU_3



Gambar 3-11. Lanjutan *Flowchart Software* pada MCU_3

3.3.2. Perancangan *Flowchart Software Delphi* pada PC



Gambar 3-12. *Flowchart Software Delphi* pada PC

BAB IV

PENGUJIAN ALAT

4.1. Umum

Bab ini akan membahas pengujian alat yang telah dirancang, dirakit dan direalisasikan. Tujuan pengujian alat ini adalah mengetahui kerja dari masing-masing sistem yang dibuat secara perblok sehingga dapat diketahui kepresisian kerja dari alat yang direncanakan dan dibuat. Secara umum tujuan dari pengujian alat tersebut adalah sebagai berikut.

1. Mengetahui proses kerja dari masing-masing rangkaian.
2. Memudahkan pendataan spesifikasi alat.
3. Mengetahui hasil dari perancangan yang telah dibuat.
4. Memudahkan perawatan dan perbaikan apabila sewaktu-waktu terjadi kerusakan.

Macam-macam pengujian :

1. Pengujian *microcontroller* AT89S8252 dan AT89S51.
2. Pengujian rangkaian sensor suhu dan kelembaban (SHT11).
3. Pengujian komunikasi serial (RS-232).
4. Pengujian LCD
5. Pengujian keseluruhan.

4.2. Spesifikasi Alat

4.2.1. Spesifikasi Elektronik

1. Menggunakan *microcontroller* AT89S8252 dan AT89S51.
2. Menggunakan sensor suhu dan kelembaban (SHT11) sebanyak dua buah untuk di dalam dan di luar ruangan.
3. *Power Supply* Alat adalah 220 *volt* Ac dan 5V DC.
4. Menggunakan komputer.
5. Menggunakan IC MAX-232 sebagai sarana komunikasi serial.

4.3. Pengujian *Microcontroller*

4.3.1. Tujuan

Tujuannya adalah untuk mengetahui kondisi awal dari *microcontroller* apakah sudah sesuai dengan yang direncanakan.

4.3.2. Peralatan yang Dibutuhkan

1. Komputer (PC).
2. LED *Display.*
3. MC AT89S51 dan MC AT89S8252.

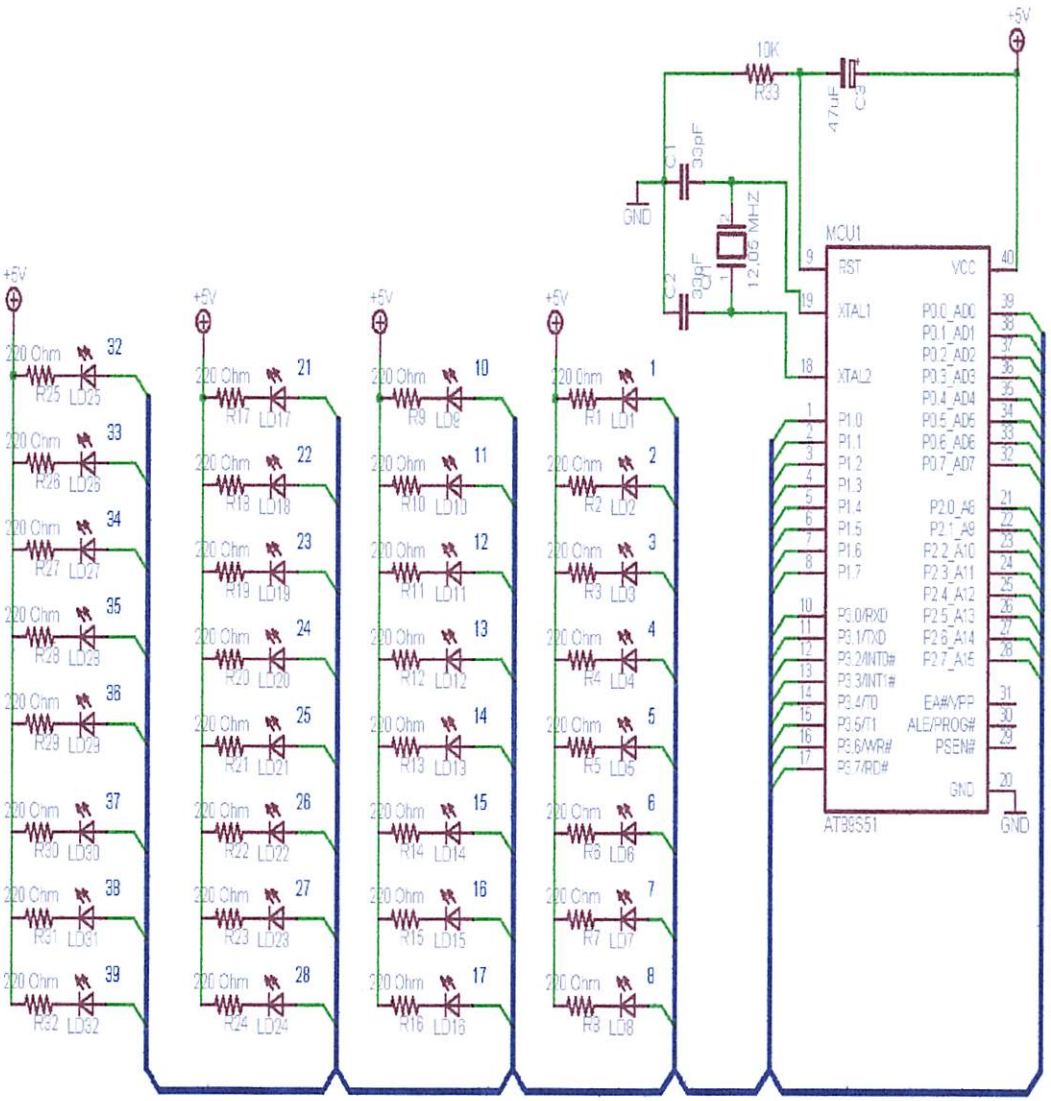
4.3.3. Langkah-langkah Pengujian MC AT89S51 dan MC AT89S8252

1. Membuat program untuk digunakan dalam pengujian *microcontroller* dimana program yang dibuat adalah sebagai berikut.

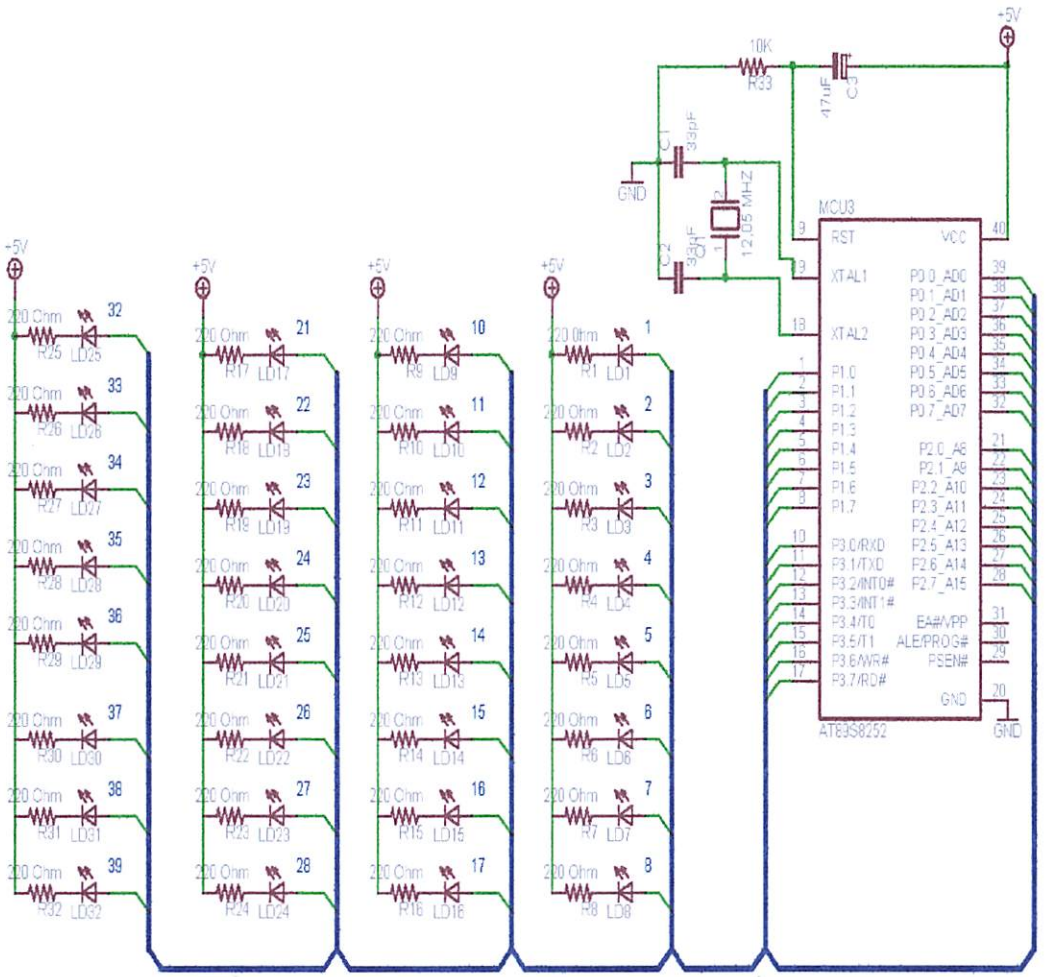
```
Org    00h
Nop
mulai: mov p0,#00h
```

```
        mov p1,#00h
        mov p2,#00h
        mov p3,#00h
jmp     mulai
ret
end
```

2. Rangkaian dibuat seperti gambar 4-1.
3. Memasang catu daya rangkaian sebesar 5 *volt*.
4. Men-*download* program di atas menggunakan komputer.
5. Mengamati dan mencatat keluaran pada LED *Display*.



Gambar 4-1. Rangkaian Pengujian *Microcontroller* AT 89S51



Gambar 4-2. Rangkaian Pengujian *Microcontroller* AT 89S8252

4.3.4. Hasil Pengujian Sistem *Microcontroller*

Hasil pengujian *microcontroller* ditunjukkan dalam tabel 4-1 berikut.

Tabel 4-1. Hasil Pengujian *Microcontroller* AT89S51 dan AT89S8252

Wkt	Logika Pada Port	L7	L6	L5	L4	L3	L2	L1	L0
1	00000000B	H	H	H	H	H	H	H	H
2	11111111B	M	M	M	M	M	M	M	M

Keterangan:

- **H** : Hidup
- **M** : Mati

Kesimpulan dari hasil pengujian rangkaian *microcontroller* di atas adalah rangkaian *microcontroller* berlogika *Low* untuk dapat menyalakan lampu led dan berlogika *High* untuk mematikan lampu led.

Cara yang sama digunakan untuk pengujian MC AT89S8252.

4.4. Pengujian Rangkaian Sensor Suhu dan Kelembaban (SHT11)

4.4.1. Tujuan

Untuk mengetahui apakah rangkaian sensor dapat mendeteksi dengan baik suhu dan kelembaban di dalam maupun di luar ruangan.

4.4.2. Langkah-langkah Percobaan

1. Men-*donwload* program yang telah dibuat ke MC AT89S8252.
2. Amati tampilannya pada LCD untuk megetahui apakah SHT11 sudah dapat mendeteksi suhu dan kelembaban atau belum.

3. Jika sudah, lakukan komparasi yaitu membandingkan hasilnya dengan alat pembaca suhu dan kelembaban *digital* (yang sudah dikalibrasi).
4. Ambil foto dan buat analisa untuk menunjukkan perbandingannya.

➢ **Perhitungan *Error* untuk Data Suhu di Dalam Ruangan**

Untuk hari ke-1 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{26°C - 7{,}4°C}{27{,}4°C} \text{ x100 \%}$$

$$= \frac{1{,}4°C}{27{,}4°C} \text{ x100 \%}$$

$$= 5{,}11 \%$$

Untuk hari ke-2 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27°C - 26{,}3°C}{26{,}3°C} \text{ x100 \%}$$

$$= \frac{0{,}7°C}{26{,}3°C} \text{ x100 \%}$$

$$= 2{,}66 \%$$

Untuk hari ke-3 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{25°C - 26{,}6°C}{26{,}6°C} \text{x100 \%}$$

$$= \frac{1{,}6°C}{26{,}6°C} \text{x100 \%}$$

$$= 6{,}01 \%$$

Untuk hari ke-4 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27°C - 27°C}{27°C} \text{x100 \%}$$

$$= \frac{0°C}{27°C} \text{x100 \%}$$

$$= 0 \%$$

Untuk hari ke-5 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{28°C - 26{,}4°C}{26{,}6°C} \text{x100 \%}$$

$$= \frac{1{,}6°C}{26{,}4°C} \text{x100 \%}$$

$$= 6{,}06 \%$$

Untuk hari ke-6 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27°C - 26,3°C}{26,3°C} \text{ x100 \%}$$

$$= \frac{0,7°C}{26,3°C} \text{ x100 \%}$$

$$= 2,66 \%$$

Untuk hari ke-7 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{29°C - 27,2°C}{27,2°C} \text{ x100 \%}$$

$$= \frac{1,8°C}{27,2°C} \text{ x100 \%}$$

$$= 6,62 \%$$

➢ **Perhitungan *Error* untuk Data Suhu di Luar Ruangan**

Untuk hari ke-1 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27°C - 27,4°C}{27,4°C} \text{ x100 \%}$$

$$= \frac{0,6°C}{27,4°C} \text{ x100 \%}$$

$$= 2,19 \%$$

Untuk hari ke-2 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27\,^{\circ}\text{C} - 26,3\,^{\circ}\text{C}}{26,3\,^{\circ}\text{C}} \text{ x100 \%}$$

$$= \frac{0,7\,^{\circ}\text{C}}{26,3\,^{\circ}\text{C}} x100\ \%$$

$$= 2,66\ \%$$

Untuk hari ke-3 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{26\,^{\circ}\text{C} - 26,6\,^{\circ}\text{C}}{26,6\,^{\circ}\text{C}} \text{ x100 \%}$$

$$= \frac{0,4\,^{\circ}\text{C}}{26,6\,^{\circ}\text{C}} \text{ x100 \%}$$

$$= 1,50\ \%$$

Untuk hari ke-4 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{26\,^{\circ}\text{C} - 27\,^{\circ}\text{C}}{27\,^{\circ}\text{C}} \text{ x100 \%}$$

$$= \frac{1\,^{\circ}\text{C}}{27\,^{\circ}\text{C}} \text{ x100 \%}$$

$$= 3,70\ \%$$

Untuk hari ke-5 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27°C - 26,4°C}{26,4°C} \text{ x100 \%}$$

$$= \frac{0,6°C}{26,4°C} \text{ x100 \%}$$

$$= 2,27 \%$$

Untuk hari ke-6 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{27°C - 26,3°C}{26,3°C} \text{ x100 \%}$$

$$= \frac{0,7°C}{26,3°C} \text{ x100 \%}$$

$$= 2,66 \%$$

Untuk hari ke-7 :

$$\text{Error} = \frac{\textit{selisih antara suhu SHT11 dengan suhu alat pembanding}}{\textit{suhu alat pembanding}} \text{x100 \%}$$

$$= \frac{28°C - 27,2°C}{27,2°C} \text{ x100 \%}$$

$$= \frac{0,8°C}{27,2°C} \text{ x100 \%}$$

$$= 2,94 \%$$

➢ **Perhitungan *Error* untuk Kelembaban di Dalam Ruangan**

Untuk hari ke-1 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{73\%RH - 2\%RH\,°\text{C}}{73\%RH} \text{x}100\,\%$$

$$= \frac{1\%RH}{73\%RH} \text{x}100\,\%$$

$$= 1{,}36\,\%$$

Untuk hari ke-2 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{71\%RH - 71\%RH\,°\text{C}}{71\%RH} \text{x}100\,\%$$

$$= \frac{0\%RH}{71\%RH} \text{x}100\,\%$$

$$= 0\,\%$$

Untuk hari ke-3 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{77\%RH - 76\%RH\,°\text{C}}{77\%RH} \text{x}100\,\%$$

$$= \frac{1\%RH}{77\%RH} \text{x}100\,\%$$

$$= 1{,}29\,\%$$

Untuk hari ke-4 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{75\%RH - 74\%RH\,°\text{C}}{75\%RH} \text{x}100\,\%$$

$$= \frac{1\%\,RH}{75\,\%\,RH} \text{x}100\,\%$$

$$= 1{,}33\,\%$$

Untuk hari ke-5 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{74\%RH - 74\%RH°\text{C}}{74\%RH} \text{x}100\,\%$$

$$= \frac{0\%\,RH}{74\,\%\,RH} \text{x}100\,\%$$

$$= 0\,\%$$

Untuk hari ke-6 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{71\%RH - 71\%RH\,°\text{C}}{71\%RH} \text{x}100\,\%$$

$$= \frac{0\%\,RH}{71\,\%\,RH} \text{x}100\,\%$$

$$= 0\,\%$$

Untuk hari ke-7 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{72\%RH - 71\%RH\,°\text{C}}{71\%RH} \text{x}100\ \%$$

$$= \frac{1\%RH}{71\%RH} \text{x}100\ \%$$

$$= 1{,}41\ \%$$

➢ **Perhitungan *Error* untuk Kelembaban di Luar Ruangan**

Untuk hari ke-1 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{73\%RH - 71\%RH\,°\text{C}}{73\%RH} \text{x}100\ \%$$

$$= \frac{2\%RH}{73\%RH} \text{x}100\ \%$$

$$= 2{,}74\ \%$$

Untuk hari ke-2 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{71\%RH - 70\%RH\,°\text{C}}{71\%RH} \text{x}100\ \%$$

$$= \frac{1\%RH}{71\%RH} \text{x}100\ \%$$

$$= 1{,}41\ \%$$

Untuk hari ke-3 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{77\%RH - 74\%RH\,°\text{C}}{77\%RH} \text{x}100\,\%$$

$$= \frac{3\%RH}{77\%RH} \text{x}100\,\%$$

$$= 3{,}89\,\%$$

Untuk hari ke-4 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{75\%RH - 75\%RH\,°\text{C}}{75\%RH} \text{x}100\,\%$$

$$= \frac{0\%RH}{75\%RH} \text{x}100\,\%$$

$$= 0\,\%$$

Untuk hari ke-5 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{74\%RH - 74\%RH\,°\text{C}}{74\%RH} \text{x}100\,\%$$

$$= \frac{0\%RH}{74\%RH} \text{x}100\,\%$$

$$= 0\,\%$$

Untuk hari ke-6 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{72\%RH - 71\%RH\,°\text{C}}{71\%RH} \text{x}100\,\%$$

$$= \frac{1\%RH}{71\%RH} \text{x}100\,\%$$

$$= 1{,}41\,\%$$

Untuk hari ke-7 :

$$\text{Error} = \frac{\textit{selisih kelembaban SHT11 dengan kelembaban alat pembanding}}{\textit{kelembaban alat pembanding}} \text{x}100\%$$

$$= \frac{72\%RH - 71\%RH°\text{C}}{71\%RH} \text{x}100\,\%$$

$$= \frac{1\%RH}{71\%RH} \text{x}100\,\%$$

$$= 1{,}41\,\%$$

4.4.3. Hasil Pengujian Sensor

Dari komparasi didapatkan hasil sebagai berikut.

- Pada hari ke-1, untuk sensor di dalam ruangan terdapat *error* sebesar 5,11 % untuk suhu dan 1,36 % untuk kelembaban sedangkan untuk sensor di luar ruangan terdapat *error* sebesar 2,19 % untuk suhu dan 2,74 % untuk kelembaban.
- Pada hari ke-2, untuk sensor di dalam ruangan terdapat *error* sebesar 2,66 % untuk suhu dan 0 % untuk kelembaban sedangkan untuk sensor

di luar ruangan terdapat *error* sebesar 2,66 % untuk suhu dan 1,41 % untuk kelembaban.

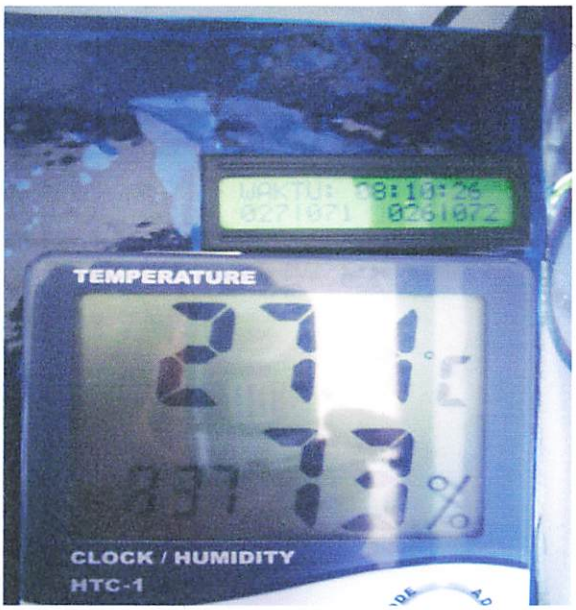
- Pada hari ke-3, untuk sensor di dalam ruangan terdapat *error* sebesar 6,01 % untuk suhu dan 1,29 % untuk kelembaban sedangkan untuk sensor di luar ruangan terdapat *error* sebesar 1,5 % untuk suhu dan 3,89 % untuk kelembaban.
- Pada hari ke-4, untuk sensor di dalam ruangan terdapat *error* sebesar 0 % untuk suhu dan 1,33 % untuk kelembaban sedangkan untuk sensor di luar ruangan terdapat *error* sebesar 3,7 % untuk suhu dan 0 % untuk kelembaban.
- Pada hari ke-5, untuk sensor di dalam ruangan terdapat *error* sebesar 6,06 % untuk suhu dan 0 % untuk kelembaban sedangkan untuk sensor di luar ruangan terdapat *error* sebesar 2,27 % untuk suhu dan 0 % untuk kelembaban.
- Pada hari ke-6, untuk sensor di dalam ruangan terdapat *error* sebesar 2,66 % untuk suhu dan 0 % untuk kelembaban sedangkan untuk sensor di luar ruangan terdapat *error* sebesar 2,66 % untuk suhu dan 1,41 % untuk kelembaban.
- Pada hari ke-7, untuk sensor di dalam ruangan terdapat *error* sebesar 6,62 % untuk suhu dan 1,41 % untuk kelembaban sedangkan untuk sensor di luar ruangan terdapat *error* sebesar 2,94 % untuk suhu dan 1,41 % untuk kelembaban.

Tabel 4-2. Tabel Komparasi (Perbandingan) antara Data pada Alat Dengan Data pada Alat Pembanding

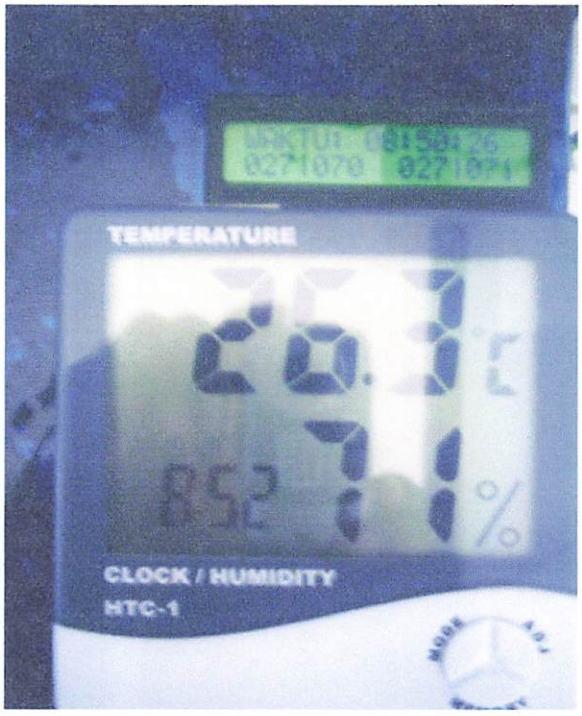
Hari Ke	Data Sensor SHT11				Data Alat Pembanding		*Error (%)*			
	Dalam Ruangan		Luar Ruangan				Dalam Ruangan		Luar Ruangan	
	SH (°C)	KLBB (%RH)	SH (°C)	KLBB (%RH)	SH (°C)	KLBB (%RH)	SH	KLBB	SH	KLBB
1	27	71	26	72	27,4	73	5,11	1,36	2,19	2,74
2	27	70	27	71	26,3	71	2,66	0	2,66	1,41
3	26	74	25	76	26,6	77	6,01	1,29	1,50	3,89
4	26	75	27	74	27	75	0	1,33	3,7	0
5	27	74	28	74	26,4	74	6,06	0	2,27	0
6	27	72	27	71	26,3	71	2,66	0	2,66	1,41
7	28	72	29	72	27,2	71	6,62	1,41	2,94	1,41

Keterangan :

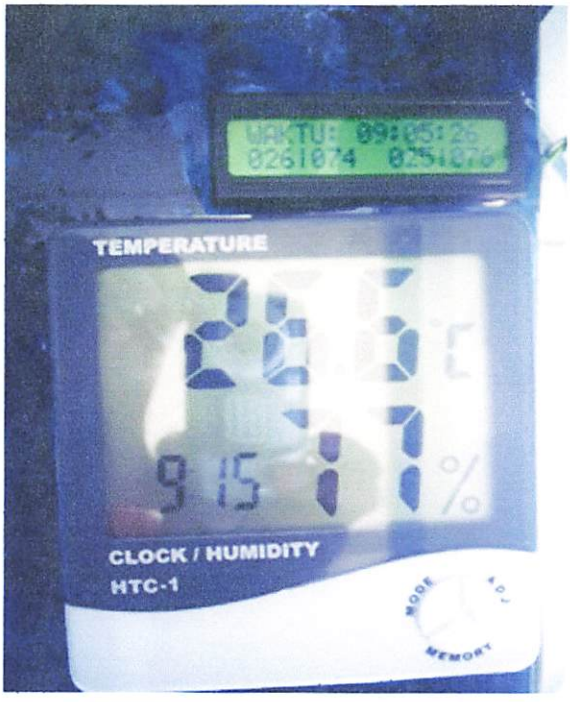
- SH adalah suhu dalam °C.
- KLB adalah kelembaban dalam %RH.



Gambar 4-3. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-1

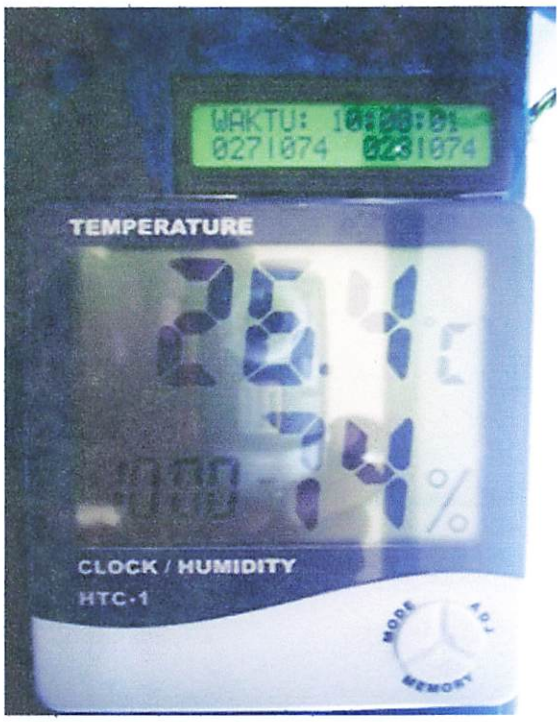


Gambar 4-4. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-2

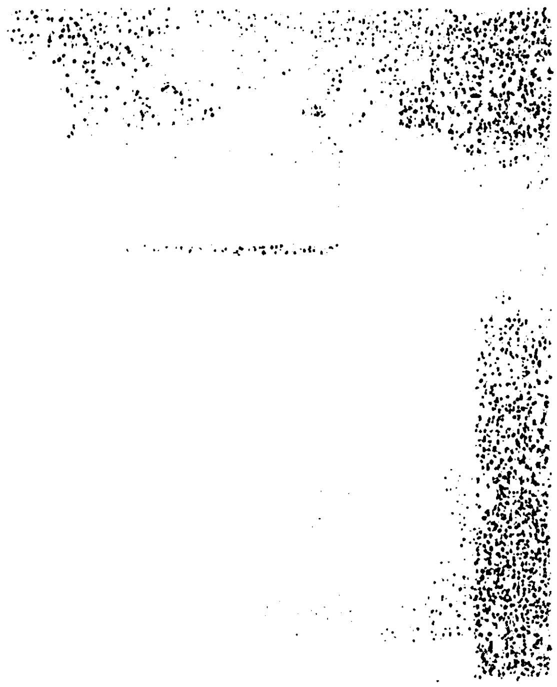


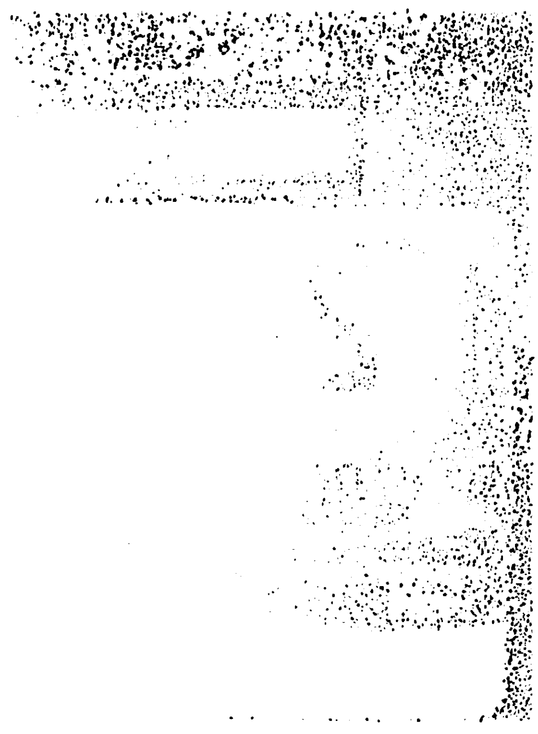
Gambar 4-5. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-3

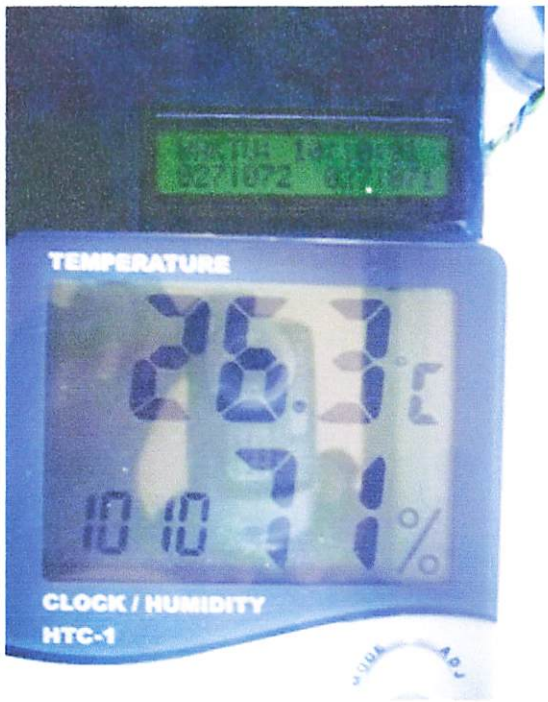
Gambar 4-6. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-4



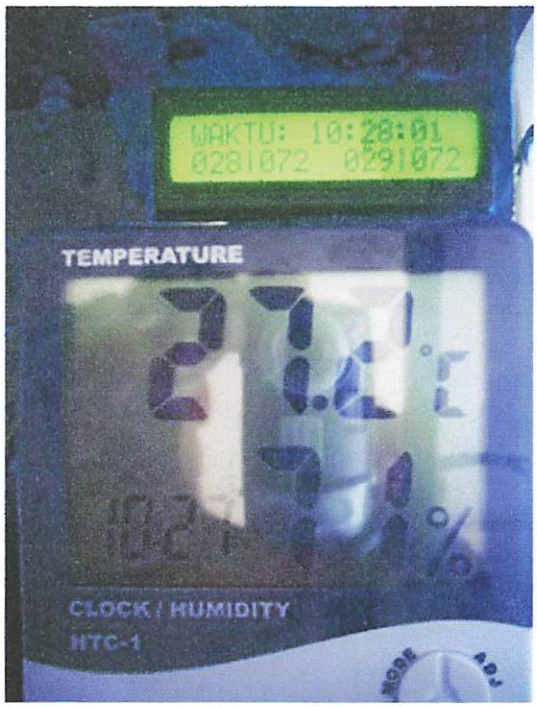
Gambar 4-7. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-5



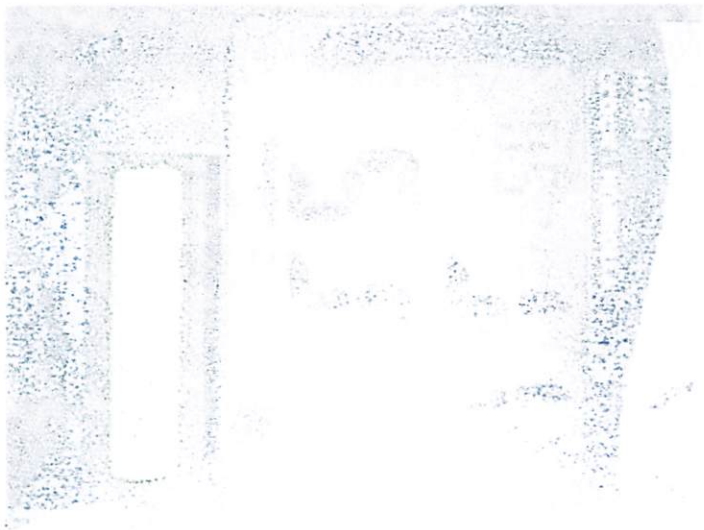




Gambar 4-8. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-6



Gambar 4-9. Gambar Komparasi antara Data pada Alat Dengan Data pada Alat Pembanding untuk Hari ke-7



4.5. Pengujian Komunikasi Serial RS-232

4.5.1. Tujuan Komunikasi Serial

Untuk mengetahui apakah data yang dikirimkan secara serial sudah sesuai dengan data yang diterima PC ataukah belum.

4.5.2. Peralatan yang digunakan

1. Komputer (PC).
2. Alat.
3. *Power supply* 5V.
4. Kabel serial RS-232.

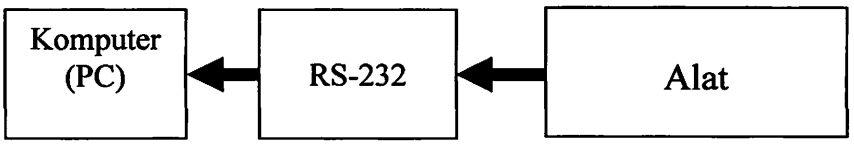
4.5.3. Prosedur Pengujian Komunikasi Serial

1. Men-*download* program seperti di bawah ini ke MC AT89S8252.

```
;=============================================
; Program komunikasi serial antar
; MCU dengan baud rate = 9600 bps
; x-tal = 11.059200 MHz dan SMOD = 0
; File name : SERIAL.H51
;=============================================
     ORG  0H
     JMP  MULAI
     ORG  23H
     CLR  ES
     JNB  RI,$
     CLR  RI
     MOV  P1,SBUF
     SETB ES
     RET
;
MULAI:
     MOV  B,#0
     MOV  SCON,#50H
     MOV  TMOD,#20H
     MOV  TL1,#0FDH
     MOV  TH1,#0FDH
     SETB TR1
     MOV  IE,#90H
```

```
      SETB EA          ; enable interupt
      SETB ES          ; enable serial interupt

ulang1:
      MOV  A,B
      INC  A
      MOV  B,A

       MOV 50H,#0FFH
TUNDAL:
      MOV  51H,#0FFH
      DJNZ 51H,$
      DJNZ 50H,TUNDAL

KIRIM:     CLR  ES    ; matikan serial interupt saat
mengirim
      MOV  A,B
      MOV  SBUF,A     ; isi serial buffer dengan data yg
dikirim
      JNB  TI,$ ; tunggu pengiriman selesai
      CLR  TI    ; clear transmit interupt flag
      MOV  50H,#0FFH
TUNDAL1:
      MOV  51H,#0FFH
      DJNZ 51H,$
      DJNZ 50H,TUNDAL1
      SETB ES    ; hidupkan kembali serial interupt
      JMP  ULANG1     ; ulangi
      END
```

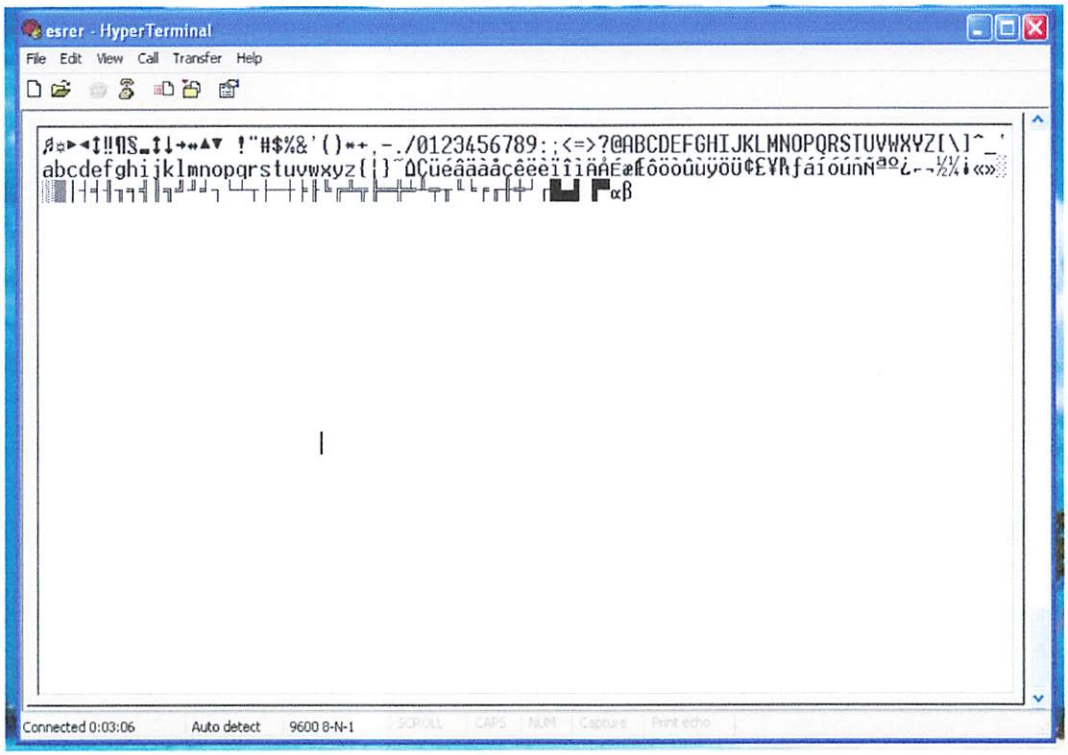
2. Menghubungkan alat dengan PC menggunakan kabel serial RS-232 berdasarkan gambar 4-4.
3. Lakukan *setting* pada PC untuk dapat menerima data dari alat.
4. Mengabadikan hasil keluarannya melalui prosedur *print screen.*



Gambar 4-10. Diagram Blok Pengujian Komunikasi Serial

4.5.4. Hasil Pengujian Komunikasi Serial

Dalam melakukan pengujian komunikasi serial pada PC, data diterima dan ditampilkan menggunakan *Hyperterminal* yang telah disediakan oleh *Windows*. Hasil penerimaan data pada *Hyperterminal* dapat dilihat sebagai berikut.



Gambar. 4-11. Foto Tampilan Pada *Hyperterminal*

Dengan hasil pengujian seperti ditunjukkan dalam gambar 4-5 terlihat bahwa PC menerima data sesuai dengan isi program pada alat.

4.6. Pengujian LCD

4.6.1. Tujuan

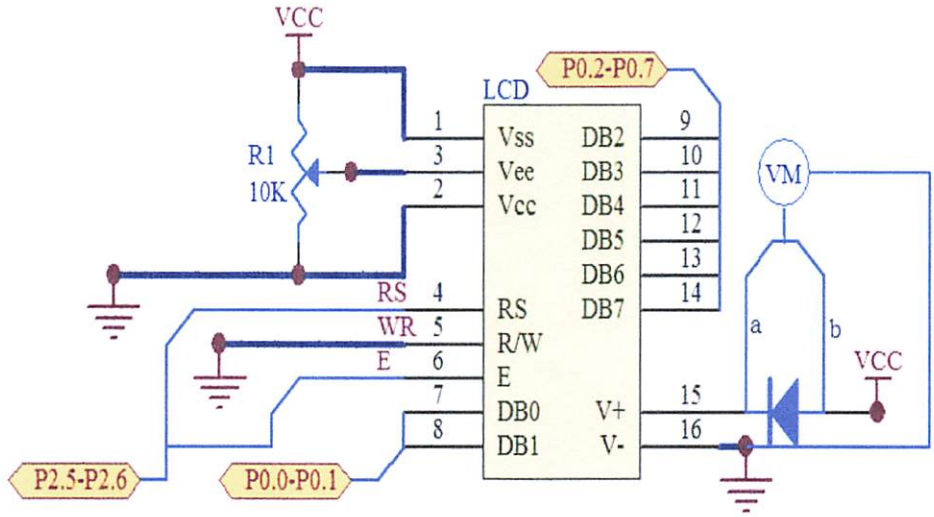
Bertujuan untuk mengetahui kondisi dan tampilan LCD serta mencatat nilai tegangan yang masuk pada LCD sebelum dan sesudah melewati dioda.

4.6.2. Alat-alat yang digunakan

- LCD.
- Rangkaian MC AT 89S51.
- Catu daya.
- *Voltmeter digital.*

4.6.3. Prosedur pengujian

1. Menyusun rangkaian pengujian seperti pada gambar 4-6.



Gambar 4-12. Rangkaian Pengujian LCD

2. Men-*download* program seperti di bawah ini ke MC AT89S8252.

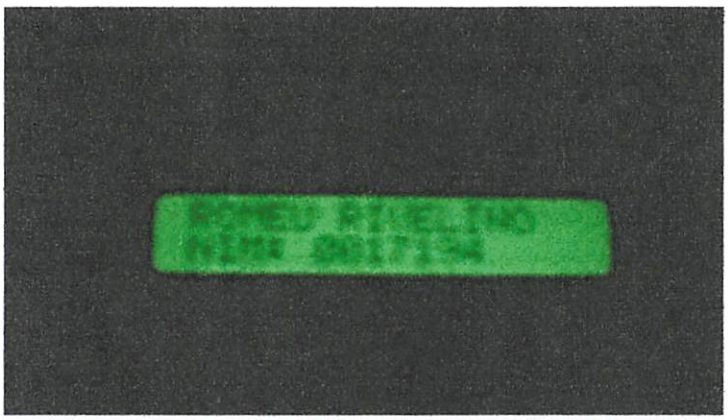
```
; LCD CONSTANTA

DISPCLR   EQU 00000001B
BLINK     EQU 00001101B
ENTRMOD   EQU 00000110B
DISPON    EQU 00001100B
CURSOR    EQU 00011100B
FUNCSET   EQU 00111000B
;
;DEVICE ADDRESS [LCD]
RS        BIT P2.6  ;LCD
E         BIT P2.7  ;LCD

          JMP   MULAI
MULAI:
          LCALL
          INIT_LCD

INISIALISASI:
;*************************
;* INISIALISASI LCD    *
;*************************
DELAY_INIT_LCD:
          MOV R6,#20H
DLY_LCD_LP:

          MOV   R7,#0
          DJNZ  R7,$
          DJNZ  R6,DLY_LCD_LP
          RET
INIT_LCD:
          SETB RS
          CLR E
          MOV A,#DISPCLR
          LCALL CONTROLOUT
          LCALL DELAY_INIT_LCD
          MOV   A,#FUNCSET
          LCALL  CONTROLOUT
          MOV   A,#DISPON
          LCALL  CONTROLOUT
          MOV   A,#ENTRMOD
          LCALL  CONTROLOUT
          MOV  DPTR,#NAMA
          LCALL
          PRINTSTRING1
          MOV  DPTR,#SEKOLAH
```

```
        LCALL
        PRINTSTRING2
        JMP $

;*********************************
;* KUMPULAN RUTIN PELAYANAN LCD *
;*********************************
POSISI2_1:
        MOV   A,#1
POSISI2:
        ADD   A,#11000000B
        SJMP  POSISI_SUB
POSISI1_1:
        MOV   A,#1
POSISI1:
        ADD   A,#10000000B
POSISI_SUB:
        DEC   A
        LCALL CONTROLOUT
        RET
PRINTSTRING2:
        LCALL POSISI2_1
        SJMP  PRINTSTRING
PRINTSTRING1:
        LCALL POSISI1_1
PRINTSTRING:
        SJMP  OUTSTRING
PRINTSTRINGLOOP:
        LCALL DATAOUT
        INC   DPTR
OUTSTRING:
        CLR   A
        MOVC  A,@A+DPTR
        JNZ   PRINTSTRINGLOOP
        RET
CONTROLOUT:
        CPL RS
        CPL E
        MOV P0,A
        CPL E
        CPL RS
        MOV P0,#0FFH
        SJMP LCD_OUT
DATAOUT:
        ;CPL RS
        CPL E
        MOV P0,A
        CPL E
```

```
        ;CPL RS
LCD_OUT:
        MOVX  @DPTR,A
DELAY_LCD:
        PUSH ACC
        MOV   A,#250
        DJNZ  ACC,$
        POP ACC
        RET

NAMA:  DB ' ROMEU RIVELINO ',0
NIM:   DB ' NIM: 0017194   ',0
        END
```

4.6.4. Hasil pengujian LCD

Dari hasil pengujian maka didapatkan tampilan seperti yang terlihat pada gambar berikut ini.



Gambar 4-13. Hasil Pengujian LCD

Tabel 4-2. Hasil Pengukuran pada Pengujian Rangkaian LCD

No	Tegangan Awal LCD (*Volt*)	Tegangan Setelah Melewati Dioda (*Volt*)
1	4,87	4,25

4.7. Pengujian Sistem Keseluruhan

4.7.1. Tujuan

Untuk mengetahui hasil pengujian akhir dari sistem yang telah dibuat, apakah sudah sesuai dengan yang perancangan atau belum.

4.7.2. Peralatan Yang digunakan

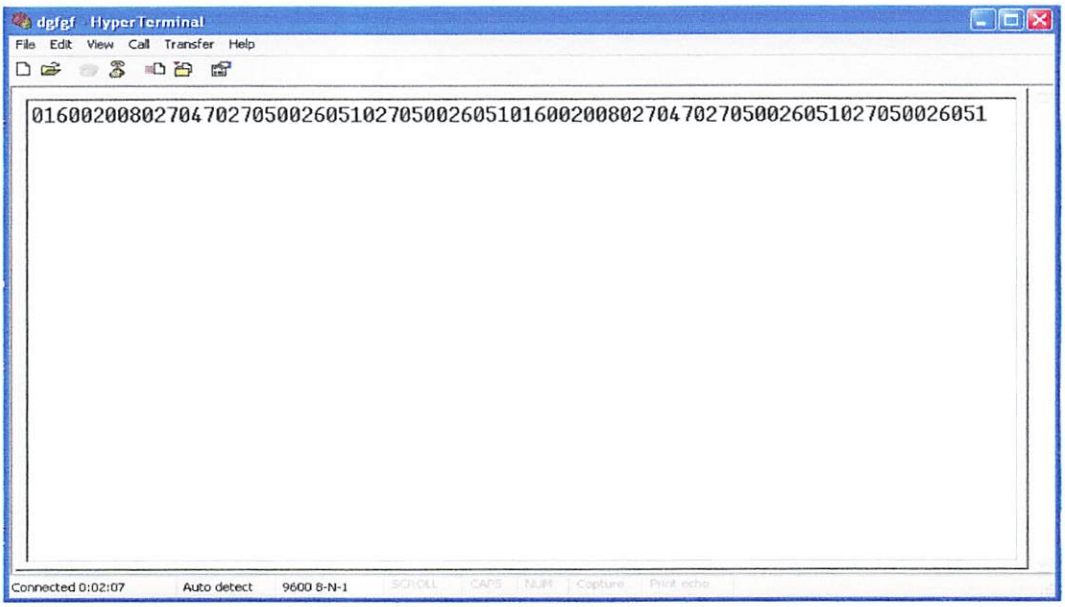
1. Komputer.
2. Rangkaian IC MAX-232, kabel serial (RS-232) .
3. Alat.

4.7.3. Prosedur Pengujian

1. Hubungkan alat ke komputer dengan cara menghubungkan kabel serial RS-232 ke db-9 pada komputer.
2. Cek pada *hyperterminal* apakah *hardware* dalam status *connected* atau *not connected.*
3. Setelah *connected* lakukan pengambilan data menggunakan program *Delphi* dan amati hasilnya pada tampilan *Delphi* di PC. Tampilannya seperti pada gambar 4-14 di bawah.

4.7.4. Hasil Pengujian Keseluruhan

Berikut adalah tampilan *hyperterminal* yang menunjukkan komunikasi antara alat dengan PC dapat berjalan dengan baik.



Gambar 4-14. Tampilan Komunikasi antara Alat dengan PC melalui *Hyperterminal*

Tampilnya *hyperterminal* di atas menunjukkan bahwa komunikasi antara alat dengan PC sudah aktif. Angka-angka pada hyperterminal di atas menunjukkan urutan tanggal, bulan, tahun, suhu luar ruang jam ke-1, kelembaban luar ruang jam ke-1, suhu dalam ruang jam ke-1, kelembaban dalam ruang jam ke-1, begitu seterusnya sampai suhu dan kelembaban jam ke-24.

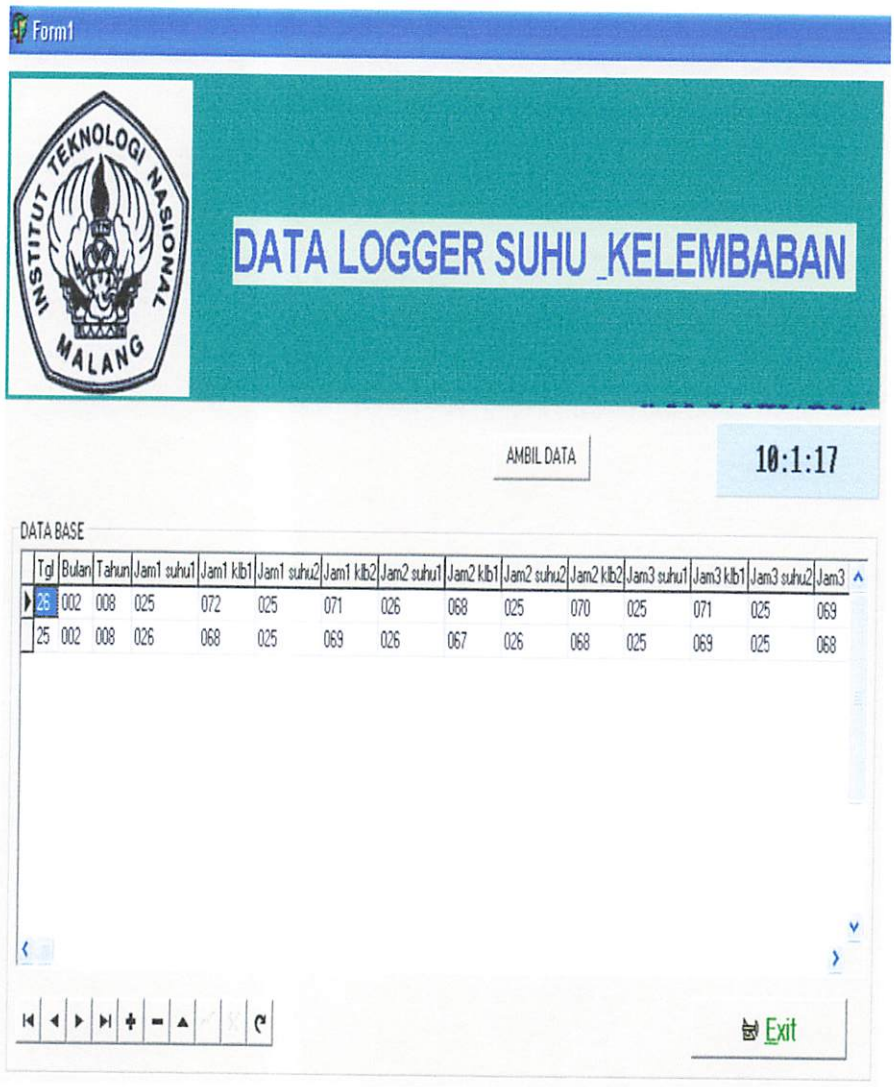
Gambar 4-15. Tampilan *Delphi* pada saat Pengambilan Data Tanggal 25 Februari 2008

Pengambilan di atas adalah pengambilan data untuk tanggal 25 Februari 2008 dengan penjelasan sebagai berikut.

1. Jam1 suhu1 menunjukkan data jam ke-1 untuk suhu luar ruang.
2. Jam1 klb1 menunjukkan data jam ke-1 untuk kelembaban luar ruang.
3. Jam1 suhu2 menunjukkan data jam ke-1 untuk suhu dalam ruang.
4. Jam1 klb2 menunjukkan data jam ke-1 untuk kelembaban dalam ruang.
5. Jam2 suhu1 menunjukkan data jam ke-2 untuk suhu luar ruang.
6. Jam2 klb1 menunjukkan data jam ke-2 untuk kelembaban luar ruang.

7. Jam2 suhu2 menunjukkan data jam ke-2 untuk suhu dalam ruang.
8. Jam2 klb2 menunjukkan data jam ke-2 untuk kelembaban dalam ruang.
9. Begitu dan seterusnya.

- Pada saat jam ke-1 suhu1 (luar ruang), suhu menunjukkan 26°c dengan kelembaban sebesar 68 % RH. Kemudian untuk jam ke-1 suhu2 (dalam ruang), terjadi penurunan suhu menjadi 25°c dengan kelembabannya meningkat menjadi 69 % RH.
- Pada saat jam ke-2 suhu1 (luar ruang) menunjukkan 26°c dengan kelembaban sebesar 67 % RH. Kemudian untuk jam ke-2 suhu2 (dalam ruang), suhu tetap 26°c dengan kelembaban yang menurun menjadi 68 % RH.
- Pada saat jam ke-3 suhu1 (luar ruang), suhu menunjukkan 25°c dengan kelembaban sebesar 69 % RH. Kemudian untuk jam ke-3 suhu2 (dalam ruang), suhunya tetap 25°c dengan kelembaban yang tetap 68 % RH.



Gambar 4-16. Tampilan *Delphi* pada saat Pengambilan Data Tanggal 26 Februari 2008

Program *Delphi* akan menyimpan semua data yang pernah diambil oleh *user*, seperti terlihat pada gambar 4-16 yang menunjukkan bahwa ketika *user* melakukan pengambilan data tanggal 26 Februari, data untuk pengambilan tanggal 25 Februari 2008 juga masih tersimpan di *Delphi*.

BAB V

PENUTUP

5.1. Kesimpulan dan Saran

5.1.1. Kesimpulan

Dari hasil perancangan dan pengujian alat dapat diambil kesimpulan sebagai berikut.

1. Untuk mempermudah komunikasi pada alat maka dalam tugas akhir ini menggunakan tiga *microcontroller*, yaitu dua buah MC AT89S51 sebagai pengambil data dari sensor SHT11 dan satu MC AT89S8252 sebagai *master* untuk menyimpan data dan mengirimkannya ke PC.
2. *User* hanya dapat mengambil data selama maksimal 7 hari karena keterbatasan *memory* pada MC AT89S8252 yang hanya sebesar 2KB.
3. Data *error* rata rata untuk suhu dalam ruang sebesar 4,16 %, untuk suhu luar ruang sebesar 2,56 %, untuk kelembaban dalam ruang sebesar 0,77 % dan untuk kelembaban luar ruang sebesar 2,08 %.

5.2. Saran-Saran

- Perancangan dan pembuatan alat ini masih jauh dari sempurna. Mungkin akan lebih sempurna jika ditambahkan *memory* eksternal sehingga dapat melakukan *logging* data yang tak terbatas pada kuantitas waktu.
- Sebaiknya memperlakukan sensor SHT11 sesuai dengan ketentuan yang ada pada *data sheet* sensor tersebut untuk mendapatkan hasil keluaran sensor yang bagus.

DAFTAR PUSTAKA

[1]. *www.sensirion.com.*

[2]. Tokhiem Roger, , *"Elektronika Digital. Edisi Kedua"*, Alih Bahasa Tjia May Ong, PhD., Erlangga, Jakarta, 1990.

[3]. Malvino and Leach, *"Prinsip-prinsip dan Penerapan Digital"*, Erlangga, Jakarta, 1992.

[4]. *"Perancangan Borland Delphi 6.0"*, PT Elex Media Komputindo, Jakarta.

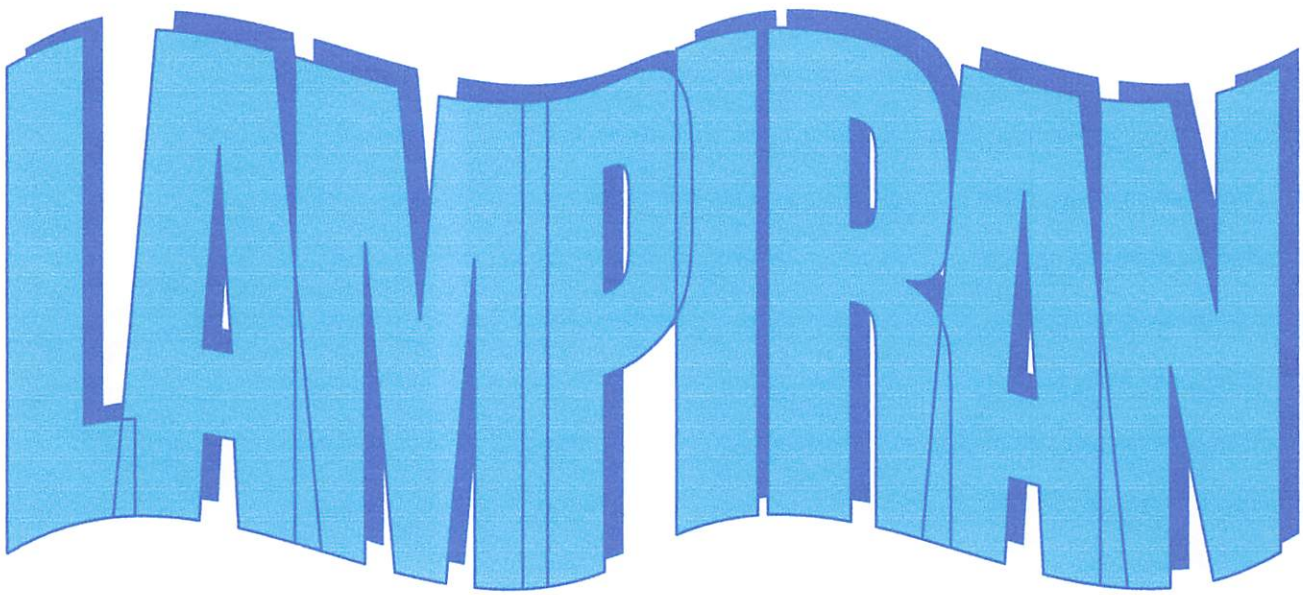
[5]. Jason Prestiliano, *"Strategi Bahasa Assembler"*, Gava Media, Yogyakarta, 2005.

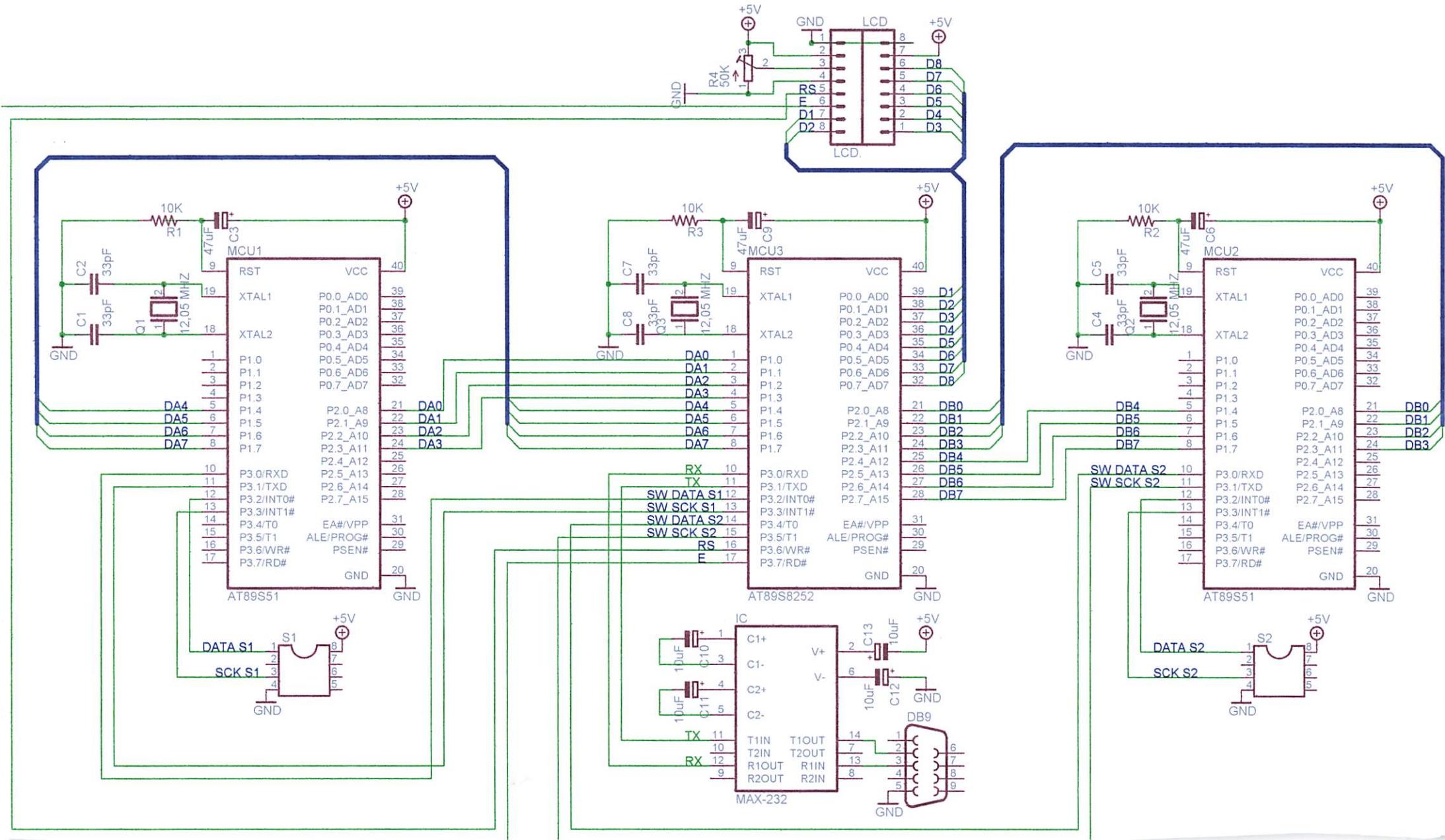
[6]. Millman and Hakias, *"Elektronika Terpadu"*, Erlangga, Jakarta, 1993.

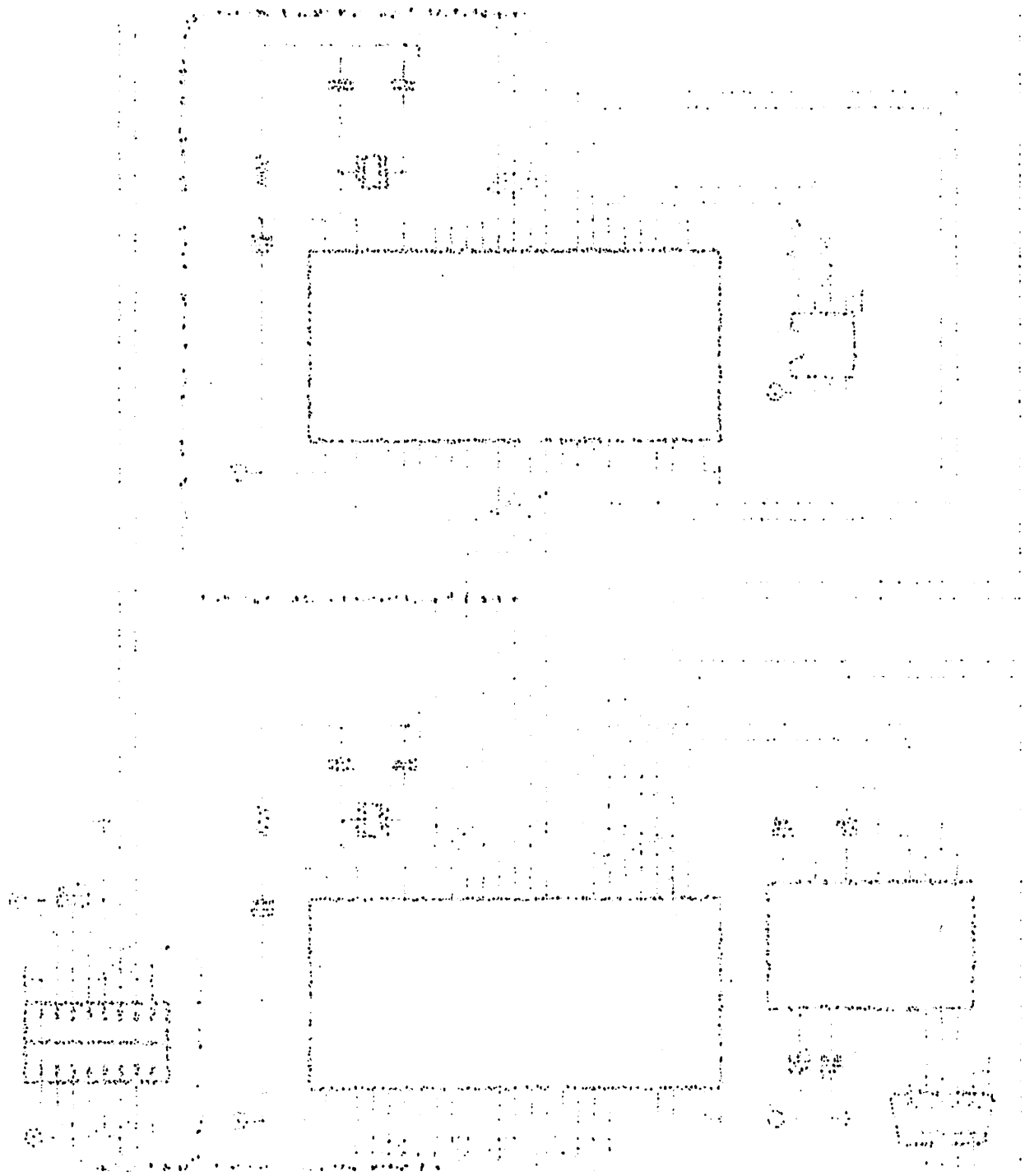
[7]. *Microcontroller* AT 89S8252 *Datasheet.*

[8]. *RS-232 Datasheet, ICS Data Com.*

[9]. *Microcontroller* AT 89S51 *Datasheet.*


LAMPIRAN


+5V
GND
LCD
R4
50K
RS
E
D1
D2
D8
D7
D6
D5
D4
D3
LCD.
MCU1
MCU3
MCU2
10K
R1
R3
R2
47uF
C3
C9
C6
C2
C1
C7
C8
C5
C4
33pF
12,05 MHZ
Q1
Q3
Q2
RST
VCC
XTAL1
XTAL2
P0.0_AD0
P0.1_AD1
P0.2_AD2
P0.3_AD3
P0.4_AD4
P0.5_AD5
P0.6_AD6
P0.7_AD7
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
P2.0_A8
P2.1_A9
P2.2_A10
P2.3_A11
P2.4_A12
P2.5_A13
P2.6_A14
P2.7_A15
P3.0/RXD
P3.1/TXD
P3.2/INT0#
P3.3/INT1#
P3.4/T0
P3.5/T1
P3.6/WR#
P3.7/RD#
EA#/VPP
ALE/PROG#
PSEN#
GND
AT89S51
AT89S8252
DA0
DA1
DA2
DA3
DA4
DA5
DA6
DA7
DB0
DB1
DB2
DB3
DB4
DB5
DB6
DB7
D1
D2
D3
D4
D5
D6
D7
D8
RX
TX
SW DATA S1
SW SCK S1
SW DATA S2
SW SCK S2
RS
E
S1
S2
DATA S1
SCK S1
DATA S2
SCK S2
IC
MAX-232
C1+
C1-
C2+
C2-
V+
V-
T1IN
T2IN
R1OUT
R2OUT
T1OUT
T2OUT
R1IN
R2IN
C10
C11
C12
C13
10uF
DB9



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

FORMULIR PERBAIKAN SKRIPSI

ama Mahasiswa : Romeu Rivelino Da Silva Belo
IM : 00.17.194
asa Bimbingan : 23 Juli 2007 s/d 23 Januari 2008
dul Skripsi : PERANCANGAN DAN PEMBUATAN *DATA LOGGER* SUHU DAN KELEMBABAN MENGGUNAKAN MC AT89S8252 BERBASIS PC

Penguji/Tanggal	Uraian	Paraf
Penguji I 28 April 2008	Pengujian (MCU, sensor, serial, LCD)	
	Rangkaian lengkap taruh di lampiran	
	Rangkain sensor diperbaiki	
	Kesimpulan (data *error*, manajemen *memory* dan penjelasan poin 1)	
Penguji II 28 April 2008	Rangkaian hubungan antara MCU dengan sensor dan penjelasannya	
	Rangkaian hubungan ketiga MCU	

Mengetahui,

Dosen Pembimbing I

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

Dosen Pembimbing II

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

Dosen Penguji I

Joseph Dedy Irawan, ST, MT
NIP. 103315178

Dosen Penguji II

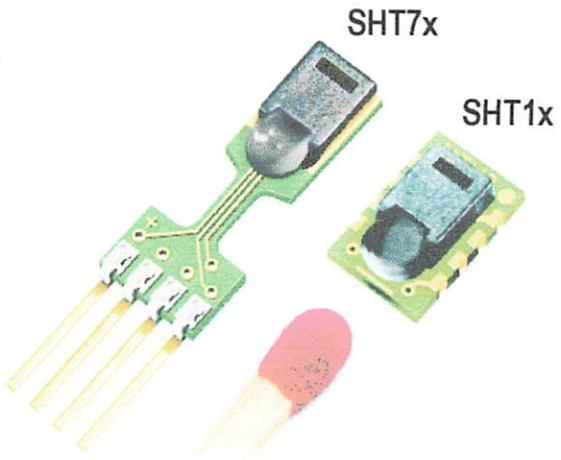
Dr. Cahyo Crysdian, MSc
NIP.103040412

SENSIRION
THE SENSOR COMPANY

T1x / SHT7x

midity & Temperature Sensor

Evaluation Kit Available



- elative humidity and temperature sensors
- ew point
- lly calibrated, digital output
- cellent long-term stability
- external components required
- tra low power consumption
- rface mountable or 4-pin fully interchangeable
- nall size
- tomatic power down

T1x / SHT7x Product Summary

SHTxx is a single chip relative humidity and erature multi sensor module comprising a calibrated l output. Application of industrial CMOS processes with nted micro-machining (CMOSens® technology) ensures est reliability and excellent long term stability. The e includes a capacitive polymer sensing element for ve humidity and a bandgap temperature sensor. Both eamlessly coupled to a 14bit analog to digital converter a serial interface circuit on the same chip. This results in rior signal quality, a fast response time and insensitivity ternal disturbances (EMC) at a very competitive price.

SHTxx is individually calibrated in a precision humidity nber with a chilled mirror hygrometer as reference. The calibration coefficients are programmed into the OTP memory. These coefficients are used internally during measurements to calibrate the signals from the sensors.

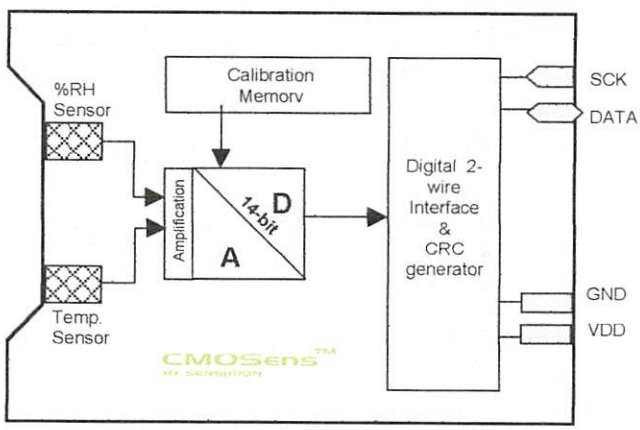
The 2-wire serial interface and internal voltage regulation allows easy and fast system integration. Its tiny size and low power consumption makes it the ultimate choice for even the most demanding applications.

The device is supplied in either a surface-mountable LCC (Leadless Chip Carrier) or as a pluggable 4-pin single-in-line type package. Customer specific packaging options may be available on request.

olications

- AC
- tomotive
- nsumer Goods
- ather Stations
- midifiers
- humidifiers
- _ Test & Measurement
- _ Data Logging
- _ Automation
- _ White Goods
- _ Medical

Block Diagram



ering Information

nber	Humidity accuracy [%RH]	Temperature accuracy [K] @ 25 °C	Package
11	±3.0	±0.4	SMD (LCC)
15	±2.0	±0.3	SMD (LCC)
71	±3.0	±0.4	4-pin single-in-line
75	±1.8	±0.3	4-pin single-in-line

Sensor Performance Specifications

meter	Conditions	Min.	Typ.	Max.	Units
idity					
olution (2)		0.5	0.03	0.03	%RH
		8	12	12	bit
eatability			±0.1		%RH
racy (1) ertainty	linearized	see figure 1			
changeability		Fully interchangeable			
inearity	raw data		±3		%RH
	linearized		<<1		%RH
ge		0		100	%RH
onse time	1/e (63%) slowly moving air		4		s
eresis			±1		%RH
term stability	typical		< 0.5		%RH/yr
perature					
olution (2)		0.04	0.01	0.01	°C
		0.07	0.02	0.02	°F
		12	14	14	bit
atability			±0.1		°C
			±0.2		°F
racy		see figure 1			
e		-40		123.8	°C
		-40		254.9	°F
onse Time	1/e (63%)	5		30	s

1 Sensor Performance Specifications

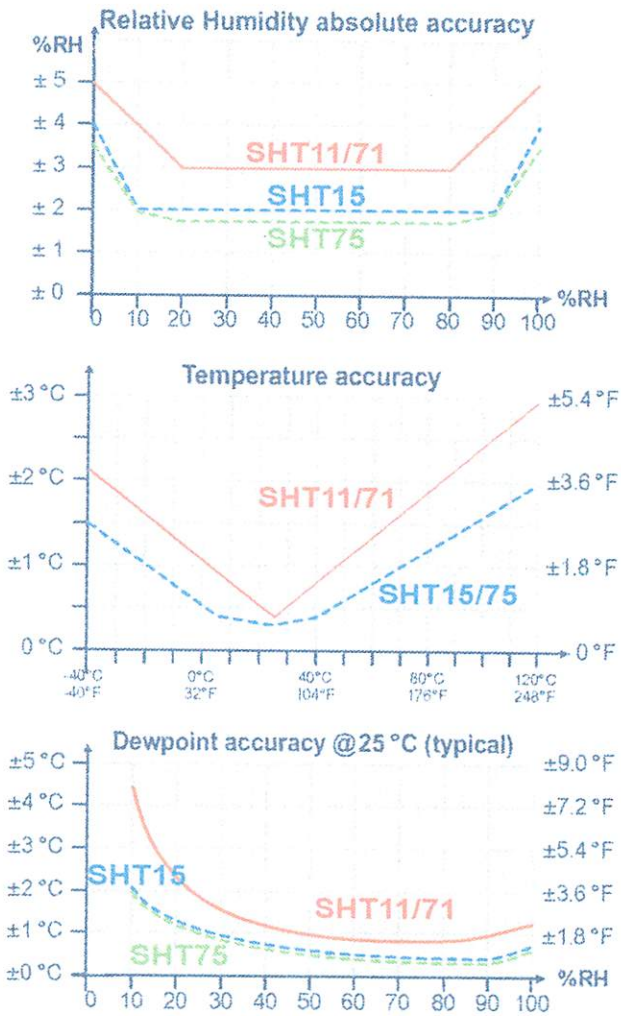
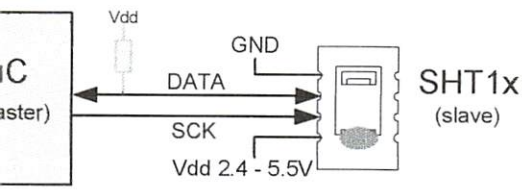


Figure 1 Rel. Humidity, Temperature and Dewpoint accuracies

Interface Specifications



2 Typical application circuit

Power Pins

HTxx requires a voltage supply between 2.4 and 5.5 V. powerup the device needs 11ms to reach its "sleep" No commands should be sent before that time.
r supply pins (VDD, GND) may be decoupled with a F capacitor.

Serial Interface (Bidirectional 2-wire)

erial interface of the SHTxx is optimized for sensor it and power consumption and is not compatible with erfaces, see FAQ for details.

2.2.1 Serial clock input (SCK)

The SCK is used to synchronize the communication between a microcontroller and the SHTxx. Since the interface consists of fully static logic there is no minimum SCK frequency.

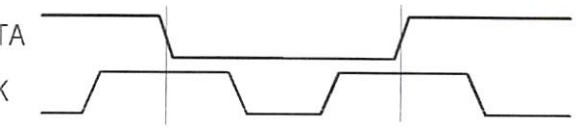
2.2.2 Serial data (DATA)

The DATA tristate pin is used to transfer data in and out of the device. DATA **changes after the falling edge** and is **valid on the rising edge** of the serial clock SCK. During transmission the DATA line must remain stable while SCK is high. To avoid signal contention the microcontroller should only drive DATA low. An external pull-up resistor (e.g. 10 kΩ) is required to pull the signal high. (See Figure 2) Pull-up resistors are often included in I/O circuits of microcontrollers. See Table 5 for detailed IO characteristics.

SHTxx is tested to be fully within RH accuracy specifications at 25 °C (77 °F) and 48 °C (118.4 °F)
efault measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit through the status register.

3 Sending a command

nitiate a transmission, a "Transmission Start" sequence to be issued. It consists of a lowering of the DATA line e SCK is high, followed by a low pulse on SCK and ng DATA again while SCK is still high.



e 3 "Transmission Start" sequence

subsequent command consists of three address bits "000" is currently supported) and five command bits. SHTxx indicates the proper reception of a command by g the DATA pin low (ACK bit) after the falling edge of 8th SCK clock. The DATA line is released (and goes after the falling edge of the 9th SCK clock.

mmand	Code
served	0000x
asure Temperature	**00011**
asure Humidity	**00101**
d Status Register	00111
e Status Register	00110
served	0101x-1110x
t reset, resets the interface, clears the us register to default values minimum 11 ms before next command	**11110**

2 SHTxx list of commands

Measurement sequence (RH and T)

issuing a measurement command ('00000101' for RH, 0011' for Temperature) the controller has to wait for the urement to complete. This takes approximately /210 ms for a 8/12/14bit measurement. The exact time s by up to ±15% with the speed of the internal oscillator. gnal the completion of a measurement, the SHTxx pulls the data line and enters idle mode. The controller **must** for this "data ready" signal before restarting SCK to ut the data. Measurement data is stored until readout, therefore the controller can continue with other tasks and readout as convenient.

Two bytes of measurement data and one byte of CRC checksum will then be transmitted. The uC must acknowledge each byte by pulling the DATA line low. All values are MSB first, right justified. (e.g. the 5th SCK is MSB for a 12bit value, for a 8bit result the first byte is not used).
Communication terminates after the acknowledge bit of the CRC data. If CRC-8 checksum is not used the controller may terminate the communication after the measurement data LSB by keeping ack high.
The device automatically returns to sleep mode after the measurement and communication have ended.

Warning: To keep self heating below 0.1 °C the SHTxx should not be active for more than 10% of the time (e.g. max. 2 measurements / second for 12bit accuracy).

2.2.5 Connection reset sequence

If communication with the device is lost the following signal sequence will reset its serial interface:
While leaving DATA high, toggle SCK 9 or more times. This must be followed by a "Transmission Start" sequence preceding the next command. This sequence resets the interface only. The status register preserves its content.

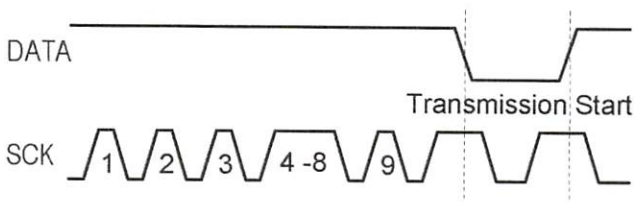
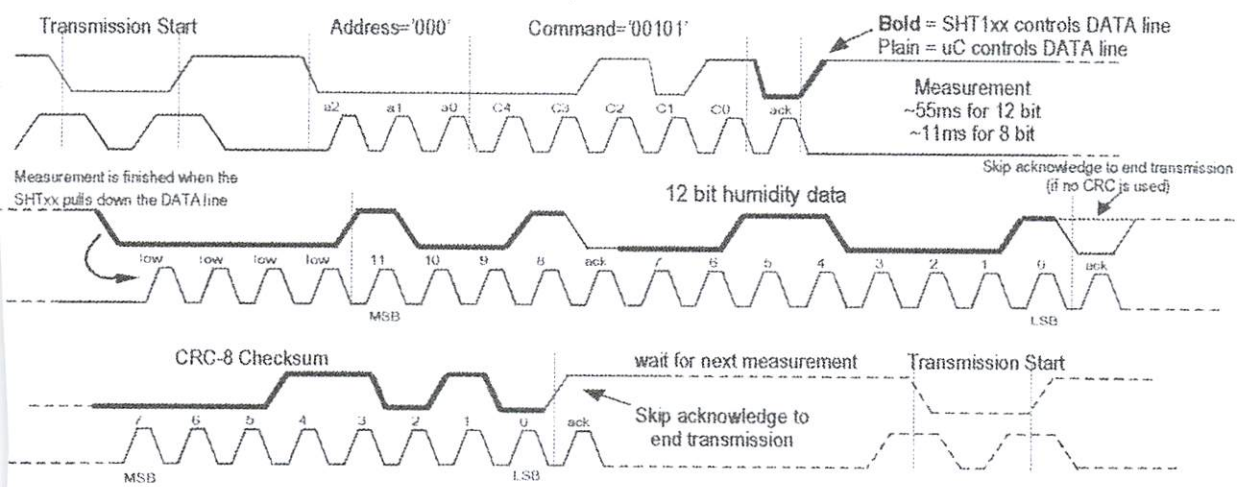


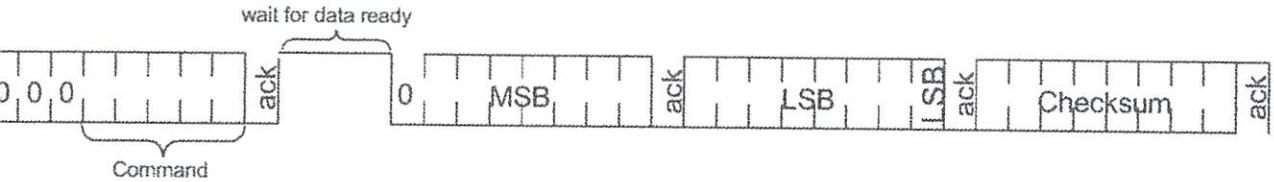
Figure 4 Connection reset sequence

2.2.6 CRC-8 Checksum calculation

The whole digital transmission is secured by a 8 bit checksum. It ensures that any wrong data can be detected and eliminated.
Please consult application note "CRC-8 Checksum Calculation" for information on how to calculate the CRC.



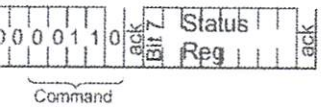
5 Example RH measurement sequence for value "0000'1001 ' 0011'0001"= 2353 = 75.79 %RH (without temperature compensation)



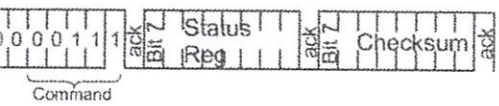
re 6 Overview of Measurement Sequence (TS = Transmission Start)

Status Register

e of the advanced functions of the SHTxx are available gh the status register. The following section gives a overview of these features. A more detailed description ailable in the application note "Status Register"



e 7 Status Register Write



e 8 Status Register Read

ype	Description	Default	
	reserved	0	
	End of Battery (low voltage detection) '0' for Vdd > 2.47 '1' for Vdd < 2.47	X	No default value, bit is only updated after a measurement
	reserved	0	
	reserved	0	
	For Testing only, do not use	0	
W	Heater	0	off
W	no reload from OTP	0	reload
W	'1' = 8bit RH / 12bit Temperature resolution '0' = 12bit RH / 14bit Temperature resolution	0	12bit RH 14bit Temp

3 Status Register Bits

Measurement resolution

default measurement resolution of 14bit (temperature) 2bit (humidity) can be reduced to 12 and 8bit. This is cially useful in high speed or extreme low power cations.

End of Battery

End of Battery" function detects VDD voltages below /. Accuracy is ±0.05 V

Heater

n chip heating element can be switched on. It will se the temperature of the sensor by 5-15 °C (9-27 °F). consumption will increase by ~8 mA @ 5 V.

ations:

mparing temperature and humidity values before and after switching on the heater, proper functionality of both sensors can be verified.

- In high (>95 %RH) RH environments heating the sensor element will prevent condensation, improve response time and accuracy

Warning: While heated the SHTxx will show higher temperatures and a lower relative humidity than with no heating.

2.4 Electrical Characteristics[1]

VDD=5V, Temperature = 25 °C unless otherwise noted

Parameter	Conditions	Min.	Typ.	Max.	Units
Power supply DC		2.4	5	5.5	V
Supply current	measuring		550		µA
	average	2[2]	28[3]		µA
	sleep		0.3	1	µA
Low level output voltage		0		20%	Vdd
High level output voltage		75%		100%	Vdd
Low level input voltage	Negative going	0		20%	Vdd
High level input voltage	Positive going	80%		100%	Vdd
Input current on pads				1	µA
Output peak current	on			4	mA
	Tristated (off)		10		µA

Table 4 SHTxx DC Characteristics

	Parameter	Conditions	Min	Typ.	Max.	Unit
F_{SCK}	SCK frequency	VDD > 4.5 V			10	MHz
		VDD < 4.5 V			1	MHz
T_{RFO}	DATA fall time	Output load 5 pF	3.5	10	20	ns
		Output load 100 pF	30	40	200	ns
T_{CLx}	SCK hi/low time		100			ns
T_V	DATA valid time			250		ns
T_{SU}	DATA set up time		100			ns
T_{HO}	DATA hold time		0	10		ns
T_R/T_F	SCK rise/fall time			200		ns

Table 5 SHTxx I/O Signals Characteristics

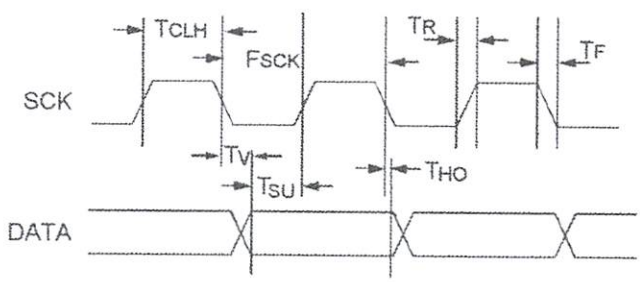


Figure 9 Timing Diagram

meters are periodically sampled and not 100% tested
one measurement of 8 bit accuracy without OTP reload per second
one measurement of 12bit accuracy per second

Converting Output to Physical Values

Relative Humidity

ompensate for the non-linearity of the humidity sensor to obtain the full accuracy it is recommended to convert eadout with the following formula[1]:

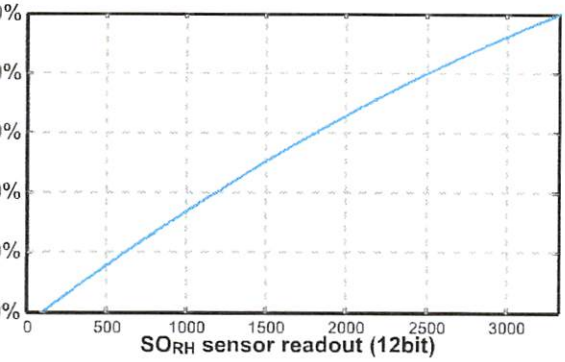
$$_{inear} = c_1 + c_2 \bullet SO_{RH} + c_3 \bullet SO_{RH}^2$$

O_{RH}	c_1	c_2	c_3
bit	-4	0.0405	$-2.8 * 10^{-6}$
bit	-4	0.648	$-7.2 * 10^{-4}$

e 6 Humidity conversion coefficients

implified, less computation intense conversion formulas application note "RH and Temperature Non-Linearity pensation".

humidity sensor has no significant voltage dependency.



e 10 Conversion from SO_{RH} to relative humidity

Humidity Sensor RH/Temperature compensation

emperatures significantly different from 25 °C (~77 °F) emperature coefficient of the RH sensor should be dered:

$$_{ue} = (T_{°C} - 25) \bullet (t_1 + t_2 \bullet SO_{RH}) + RH_{linear}$$

O_{RH}	t_1	t_2
bit	0.01	0.00008
bit	0.01	0.00128

7 Temperature compensation coefficients

equals ~0.12 %RH / °C @ 50 %RH

3.2 Temperature

The bandgap PTAT (Proportional To Absolute Temperature) temperature sensor is very linear by design. Use the following formula to convert from digital readout to temperature:

$$\text{Temperature} = d_1 + d_2 \bullet SO_T$$

VDD	d_1 [°C]	d_1 [°F]
5V	-40.00	-40.00
4V	-39.75	-39.50
3.5V	-39.66	-39.35
3V	-39.60	-39.28
2.5V	-39.55	-39.23

SO_T	d_2 [°C]	d_2 [°F]
14bit	0.01	0.018
12bit	0.04	0.072

Table 8 Temperature conversion coefficients

For improved accuracies in extreme temperatures with more computation intense conversion formulas see application note "RH and Temperature Non-Linearity Compensation".

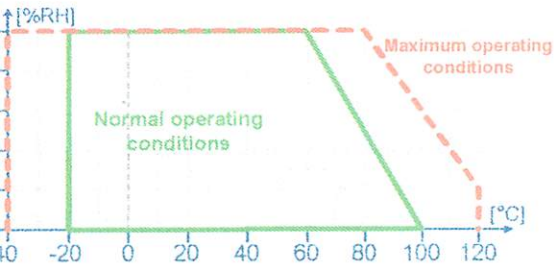
3.3 Dewpoint

Since humidity and temperature are both measured on the same monolithic chip, the SHTxx allows superb dewpoint measurements. See application note "Dewpoint calculation" for more.

[1] SO_{RH} is the sensor output for relative humidity

Applications Information

Operating and Storage Conditions



re 11 Recommended operating conditions

ditions outside the recommended range may temporarily t the RH signal up to ±3 %RH. After return to normal itions it will slowly return towards calibration state by . See 4.3 "Reconditioning Procedure" to accelerate this ess. Prolonged exposure to extreme conditions may lerate ageing.

Exposure to Chemicals

mical vapors may interfere with the polymer layers used apacitive humidity sensors. The diffusion of chemicals the polymer may cause a shift in both offset and itivity. In a clean environment the contaminants will ly outgas. The reconditioning procedure described w will accelerate this process. High levels of pollutants cause permanent damage to the sensing polymer.

Reconditioning Procedure

following reconditioning procedure will bring the sensor to calibration state after exposure to extreme conditions emical vapors.

0 °C (176-194°F) at < 5 %RH for 24h (baking) followed by
0 °C (70-90°F) at > 74 %RH for 48h (re-hydration)

Temperature Effects

relative humidity of a gas strongly depends on its erature. It is therefore essential to keep humidity ors at the same temperature as the air of which the ve humidity is to be measured.

e SHTxx shares a PCB with electronic components that off heat it should be mounted far away and below the source and the housing must remain well ventilated.

reduce heat conduction copper layers between the 1x and the rest of the PCB should be minimized and a ay be milled in between (see figure 13).

Membranes

embrane may be used to prevent dirt from entering the ing and to protect the sensor. It will also reduce peak entrations of chemical vapors. For optimal response air volume behind the membrane must be kept to a num. For the SHT1x package Sensirion recommends F1 filter cap for optimal IP67 protection.

4.6 Light

The SHTxx is not light sensitive. Prolonged direct exposure to sunshine or strong UV radiation may age the housing.

4.7 Materials Used for Sealing / Mounting

Many materials absorb humidity and will act as a buffer, increasing response times and hysteresis. Materials in the vicinity of the sensor must therefore be carefully chosen. Recommended materials are: All Metals, LCP, POM (Delrin), PTFE (Teflon), PE, PEEK, PP, PB, PPS, PSU, PVDF, PVF
For sealing and gluing (use sparingly): High filled epoxy for electronic packaging (e.g. glob top, underfill), and Silicone. Outgassing of these materials may also contaminate the SHTxx (cf. 4.2). Store well ventilated after manufacturing or bake at 50°C for 24h to outgas contaminants before packing.

4.8 Wiring Considerations and Signal Integrity

Carrying the SCK and DATA signal parallel and in close proximity (e.g. in wires) for more than 10cm may result in cross talk and loss of communication. This may be resolved by routing VDD and/or GND between the two data signals.
Please see the application note "ESD, Latchup and EMC" for more information.
Power supply pins (VDD, GND) should be decoupled with a 100 nF capacitor if wires are used.

4.9 Qualifications

Extensive tests were performed in various environments. Please contact SENSIRION for detailed information.

Environment	Norm	Results[1]
Temperature Cycles	JESD22-A104-B **-40 °C / 125 °C, 1000 cy**	Within Specifications
HAST Pressure Cooker	JESD22-A110-B **2.3 bar 125 °C 85 %RH**	Reversible shift by +2 %RH
High Temperature and Humidity	JESD22-A101-B **85 °C 85 %RH 1250h**	Reversible shift by +2 %RH
Salt Atmosphere	DIN-50021ss	Within Spec.
Condensing Air	-	Within Spec.
Freezing cycles fully submerged	**-20 / +90 °C, 100 cy** 30min dwell time	Reversible shift by +2 %RH
Various Automotive Chemicals	DIN 72300-5	Within Specifications

Table 9 Qualification tests (excerpt)

4.10 ESD (Electrostatic Discharge)

ESD immunity is qualified according to MIL STD 883E, method 3015 (Human Body Model at ±2 kV)).
Latch-up immunity is provided at a force current of ±100 mA with T_{amb} = 80 °C according to JEDEC 17. See application note "ESD, Latchup and EMC" for more information.

[1] e temperature sensor passed all tests without any detectable drift. Package and electronics also passed 100%

Package Information

SHT1x (surface mountable)

	Name	Comment
	GND	Ground
	DATA	Serial data, bidirectional
	SCK	Serial clock, input
	VDD	Supply 2.4 - 5.5 V
	NC	Remaining pins must be left unconnected

e 10 SHT1x Pin Description

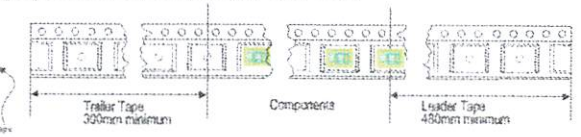
1 Package type

SHT1x is supplied in a surface-mountable LCC dless Chip Carrier) type package. The sensors housing ists of a Liquid Crystal Polymer (LCP) cap with epoxy top on a standard 0.8 mm FR4 substrate. The device is of lead, Cd and Hg.
ce size is 7.42 x 4.88 x 2.5 mm (0.29 x 0.19 x 0.1 inch)
ght 100 mg

production date is printed onto the cap in white numbers e form wwy. e.g. "351" = week 35, 2001.

2 Delivery Conditions

SHT1x are shipped in 12mm tape at 100pcs or 400pcs.. s are individually labelled with barcode and human able labels. The Lot numbers allow full traceability gh production, calibration and test.



e 12 Tape configuration and unit orientation

3 Soldering Information

dard reflow soldering ovens may be used. For details e see application note "soldering procedure".

For manual soldering contact time must be limited to 5 seconds at up to 350 °C.
After soldering the devices should be stored at >74 %RH for at least 24h to allow the polymer to rehydrate.
Please consult the application note "Soldering procedure" for more information.

5.1.4 Mounting Examples

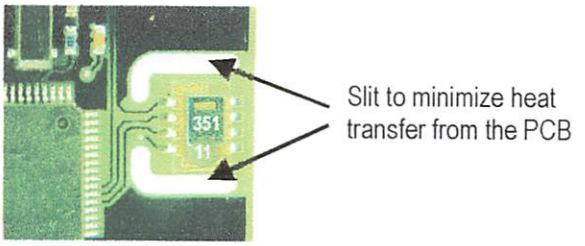


Figure 13 SHT1x PCB Mounting example

The SF1 membrane filter cap is available for optimal IP67 protection. When mounted through a housing the interior can be protected from the environment while still allowing high quality humidity measurements (see example below).

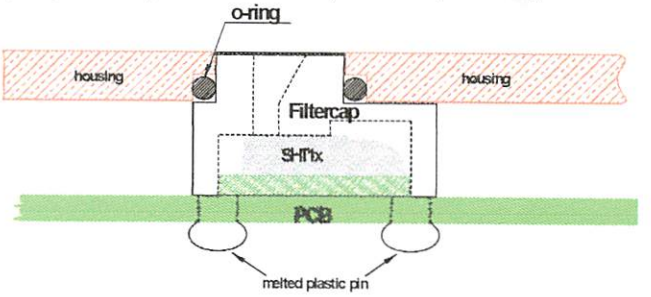
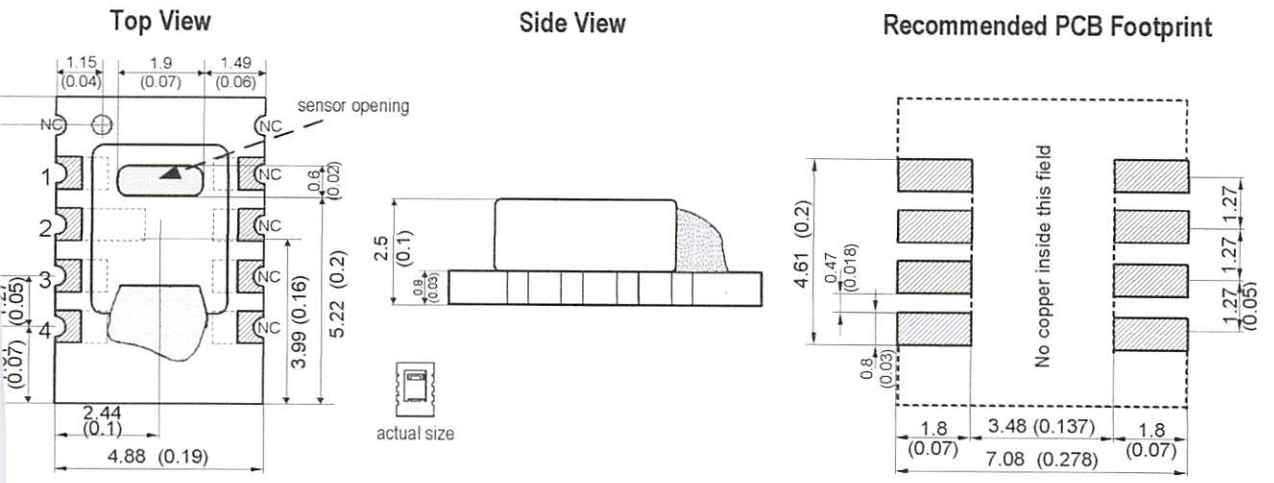


Figure 14 SF1 IP67 filter cap mounting example



15 SHT1x drawing and footprint dimensions in mm (inch)

SHT7x (4-pin single-in-line)

	Name	Comment
	SCK	Serial clock input
	VDD	Supply 2.4 - 5.5 V
	GND	Ground
	DATA	Serial data bidirectional

e 11 SHT7x Pin Description

1 Package type[1]

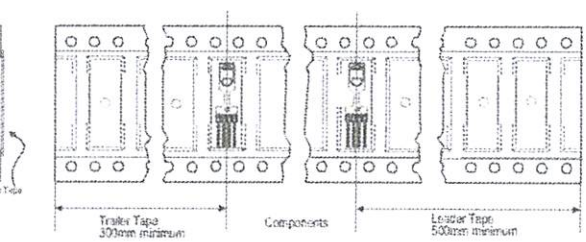
device is supplied in a single-in-line pin type package.
sensor housing consists of a Liquid Crystal Polymer
P) cap with epoxy glob top on a standard 0.6 mm FR4
trate. The device is Cd and Hg free.
sensor head is connected to the pins by a small bridge
inimize heat conduction and response times. The gold
d back side of the sensor head is connected to the GND

00nF capacitor is mounted on the back side between
and GND.
ins are gold plated to avoid corrosion. They can be
ered or mate with most 1.27 mm (0.05") sockets
Preci-dip / Mill-Max 851-93-004-20-001 or similar
weight: 168 mg, weight of sensor head: 73 mg

production date is printed onto the cap in white numbers
e form wwy. e.g. "351" = week 35, 2001.

2 Delivery Conditions

SHT7x are shipped in 32 mm tape. These reeled parts
andard option are shipped with 500 units per 13 inch
eter reel. Reels are individually labelled with barcode
human readable labels.



e 16 Tape configuration and unit orientation

3 Soldering Information[2]

dard wave SHT7x soldering ovens may be used at
mum 235 °C for 20 seconds.
manual soldering contact time must be limited to 5
nds at up to 350 °C.
wave soldering the devices should be stored at
%RH for at least 24 h to allow the polymer to rehydrate.
e consult the application note "Soldering procedure" for
information.

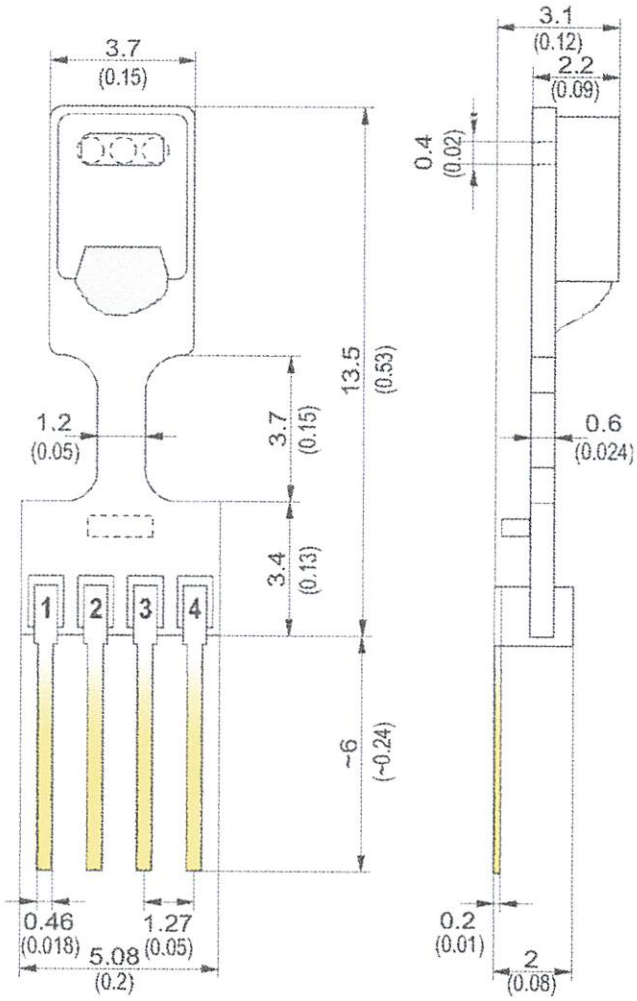


Figure 17 SHT7x dimensions in mm (inch)

r packaging options may be available on request.
naximum accuracy do not solder SHT75!

Revision history

e	Version	Page(s)	Changes
ruary 2002	Preliminary	1-9	First public release
e 2002	Preliminary		Added SHT7x information
ch 2003	Final v2.0	1-9	Major remake, added application information etc. Various small modifications
	V2.01	1-9	Typos, Graph labeling
2004	V2.02	1-9	Improved specifications, added SF1 information, improved wording

latest version of this document and all application notes can be found at:
.sensirion.com/en/download/humiditysensor/SHT1x_SHT7x.htm

Important Notices

Warning, personal injury

ot use this product as safety or emergency stop ces or in any other application where failure of the luct could result in personal injury. Failure to ply with these instructions could result in death or ous injury.

ld buyer purchase or use SENSIRION AG products for any unintended or unauthorized application, Buyer shall nnify and hold SENSIRION AG and its officers, employees, diaries, affiliates and distributors harmless against all claims, , damages and expenses, and reasonable attorney fees g out of, directly or indirectly, any claim of personal injury or associated with such unintended or unauthorized use, even h claim alleges that SENSIRION AG was negligent regarding esign or manufacture of the part.

ESD Precautions

nherent design of this component causes it to be sensitive to ostatic discharge (ESD). To prevent ESD-induced damage r degradation, take normal ESD precautions when handling roduct.

application note "ESD, Latchup and EMC" for more nation.

7.3 Warranty

SENSIRION AG makes no warranty, representation or guarantee regarding the suitability of its product for any particular purpose, nor does SENSIRION AG assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typical" must be validated for each customer applications by customer's technical experts.

SENSIRION AG reserves the right, without further notice, to change the product specifications and/or information in this document and to improve reliability, functions and design.

dquarters and Sales Office

SIRION AG
ühlstr. 14
Box
052 Zürich
erland

Phone: +41 (0)44 306 40 00
Fax: +41 (0)44 306 40 30
e-mail: info@sensirion.com
http://www.sensirion.com/

Sensirion humidity sensors are available from:

find your local representative at:
www.sensirion.com/reps

tures

- npatible with MCS®51 Products
- Bytes of In-System Reprogrammable Downloadable Flash Memory
  - SPI Serial Interface for Program Downloading
  - Endurance: 1,000 Write/Erase Cycles
- Bytes EEPROM
  - Endurance: 100,000 Write/Erase Cycles
- o 6V Operating Range
- y Static Operation: 0 Hz to 24 MHz
- ee-level Program Memory Lock
- x 8-bit Internal RAM
- Programmable I/O Lines
- ee 16-bit Timer/Counters
- e Interrupt Sources
- grammable UART Serial Channel
- Serial Interface
- -power Idle and Power-down Modes
- rrupt Recovery from Power-down
- grammable Watchdog Timer
- l Data Pointer
- er-off Flag

cription

T89S8252 is a low-power, high-performance CMOS 8-bit microcontroller with 8K of downloadable Flash programmable and erasable read-only memory and 2K of EEPROM. The device is manufactured using Atmel's high-density nonvolatile ry technology and is compatible with the industry-standard 80C51 instruction d pinout. The on-chip downloadable Flash allows the program memory to be grammed In-System through an SPI serial interface or by a conventional nonvol- nemory programmer. By combining a versatile 8-bit CPU with downloadable on a monolithic chip, the Atmel AT89S8252 is a powerful microcontroller, which les a highly-flexible and cost-effective solution to many embedded control ations.

T89S8252 provides the following standard features: 8K bytes of downloadable 2K bytes of EEPROM, 256 bytes of RAM, 32 I/O lines, programmable watchdog two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt ecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, 89S8252 is designed with static logic for operation down to zero frequency and rts two software selectable power saving modes. The Idle Mode stops the CPU allowing the RAM, timer/counters, serial port, and interrupt system to continue ning. The Power-down mode saves the RAM contents but freezes the oscillator, ng all other chip functions until the next external interrupt or hardware reset.

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

8-bit Microcontroller with 8K Bytes Flash

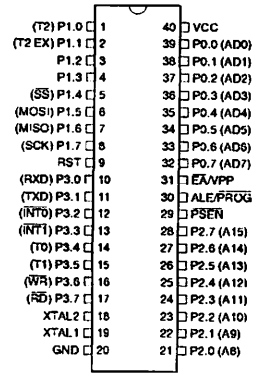
AT89S8252

Not Recommended for New Designs. Use AT89S8253.

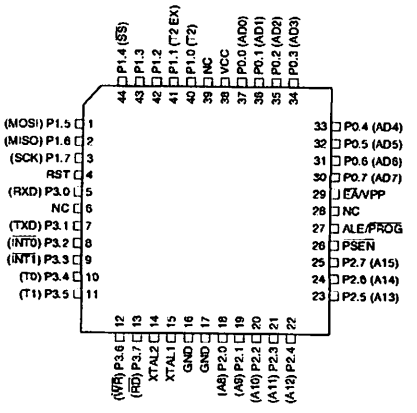
0401G–MICRO–3/06

Configurations

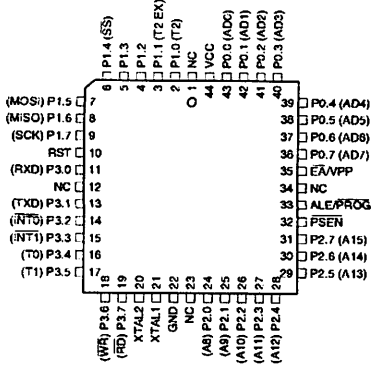
PDIP



TQFP



PLCC



Description

Supply voltage.

Ground.

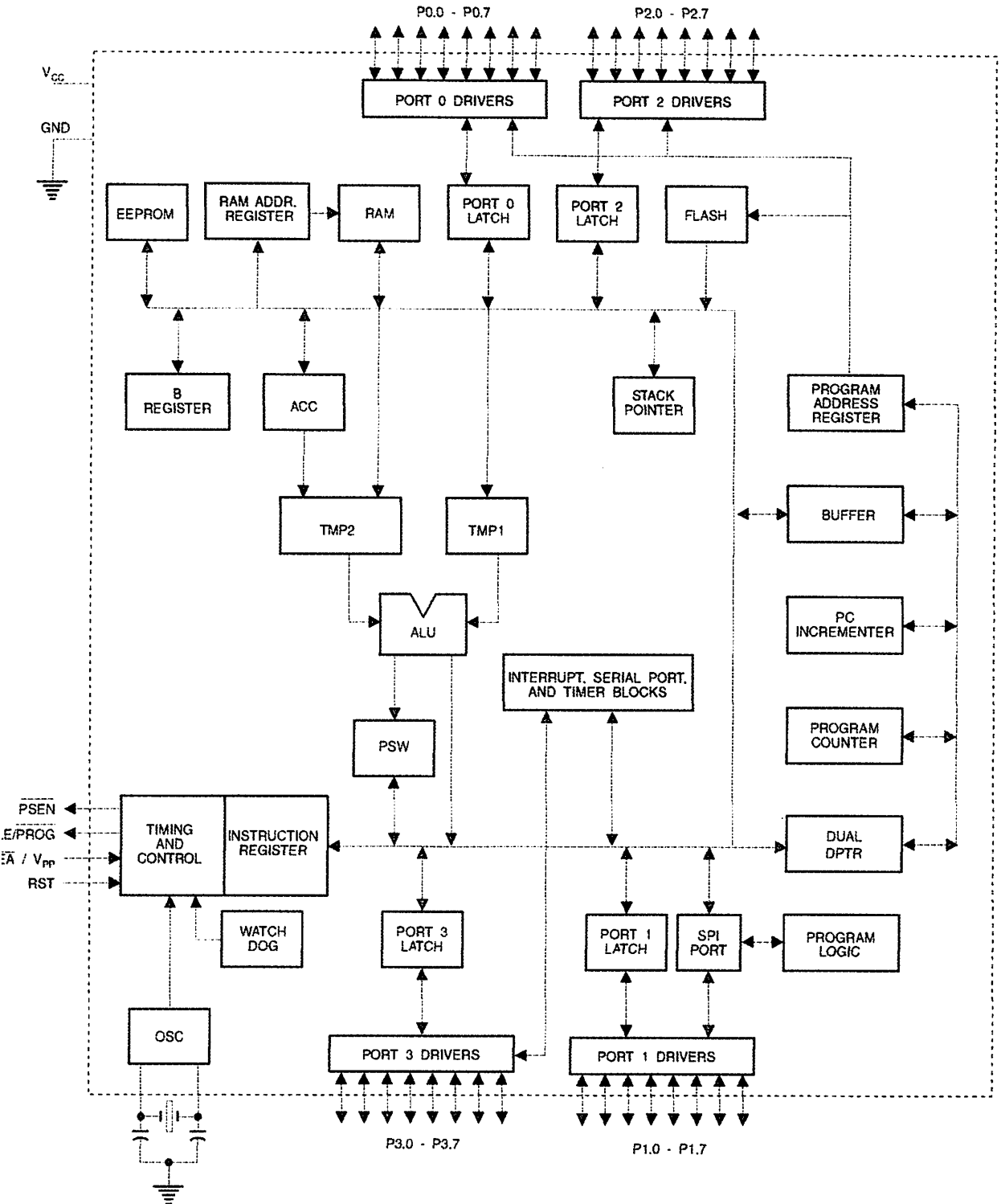
Port 0 is an 8-bit open drain bi-didirectional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

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

ck Diagram


P0.0 - P0.7
P2.0 - P2.7
V_{CC}
GND
PORT 0 DRIVERS
PORT 2 DRIVERS
EEPROM
RAM ADDR. REGISTER
RAM
PORT 0 LATCH
PORT 2 LATCH
FLASH
B REGISTER
ACC
STACK POINTER
PROGRAM ADDRESS REGISTER
TMP2
TMP1
BUFFER
ALU
PC INCREMENTER
INTERRUPT, SERIAL PORT, AND TIMER BLOCKS
PSW
PROGRAM COUNTER
PSEN
E/PROG
EA / V_{PP}
RST
TIMING AND CONTROL
INSTRUCTION REGISTER
DUAL DPTR
WATCH DOG
PORT 3 LATCH
PORT 1 LATCH
SPI PORT
PROGRAM LOGIC
OSC
PORT 3 DRIVERS
PORT 1 DRIVERS
P3.0 - P3.7
P1.0 - P1.7

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

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

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

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

2

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

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

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

3

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

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

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

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (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)

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

PROG

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

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

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

PP

External Access Enable. $\overline{EA}$ 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, $\overline{EA}$ will be internally latched on reset.

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

1

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

2

Output from the inverting oscillator amplifier.

cial Function isters

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

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

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

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

1. AT89S8252 SFR Map and Reset Values

								0FFH
B 00000000								0F7H
								0EFH
ACC 00000000								0E7H
								0DFH
PSW 00000000					SPCR 000001XX			0D7H
T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
								0C7H
IP XX000000								0BFH
P3 11111111								0B7H
IE 0X000000		SPSR 00XXXXXX						0AFH
P2 11111111								0A7H
SCON 00000000	SBUF XXXXXXXX							9FH
P1 11111111						WMCON 00000010		97H
TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

2. T2CON – Timer/Counter 2 Control Register

ON Address = 0C8H

Reset Value = 0000 0000B

ddressable

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

bol	Function
	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.
2	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).
K	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflows to be used for the receive clock.
K	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.
N2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
$\overline{L2}$	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.

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

3. WMCON—Watchdog and Memory Control Register

ON Address = 96H

Reset Value = 0000 0010B

PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN
7	6	5	4	3	2	1	0

ol	Function
	Prescaler Bits for the Watchdog Timer. When all three bits are set to "0", the watchdog timer has a nominal period of 16 ms. When all three bits are set to "1", the nominal period is 2048 ms.
/E	EEPROM Data Memory Write Enable Bit. Set this bit to "1" before initiating byte write to on-chip EEPROM with the MOVX instruction. User software should set this bit to "0" after EEPROM write is completed.
N	Internal EEPROM Access Enable. When EEMEN = 1, the MOVX instruction with DPTR will access on-chip EEPROM instead of external data memory. When EEMEN = 0, MOVX with DPTR accesses external data memory.
	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1
ST SY	Watchdog Timer Reset and EEPROM Ready/$\overline{\text{Busy}}$ Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The WDTRST bit is then automatically reset to "0" in the next instruction cycle. The WDTRST bit is Write-Only. This bit also serves as the RDY/$\overline{\text{BSY}}$ flag in a Read-Only mode during EEPROM write. RDY/$\overline{\text{BSY}}$ = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/$\overline{\text{BSY}}$ bit equals "0" and is automatically reset to "1" when programming is completed.
N	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

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

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

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

Power Off Flag The Power Off Flag (POF) is located at bit_4 (PCON.4) in the PCON SFR. POF is set to "1" during power up. It can be set and reset under software control and is not affected by RESET.

l. SPCR – SPI Control Register

Address = D5H

Reset Value = 0000 01XXB

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

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

. SPSR – SPI Status Register

Address = AAH

Reset Value = 00XX XXXXB

SPIF	WCOL	–	–	–	–	–	–
7	6	5	4	3	2	1	0

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

SPDR – SPI Data Register

Address = 86H

Reset Value = unchanged

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

Memory – OM and RAM

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

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

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

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

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

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

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

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

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

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

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

rammable hdog Timer

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

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

Table 7. Watchdog Timer Period Selection

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

0 and 1

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

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 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 8.

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

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

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

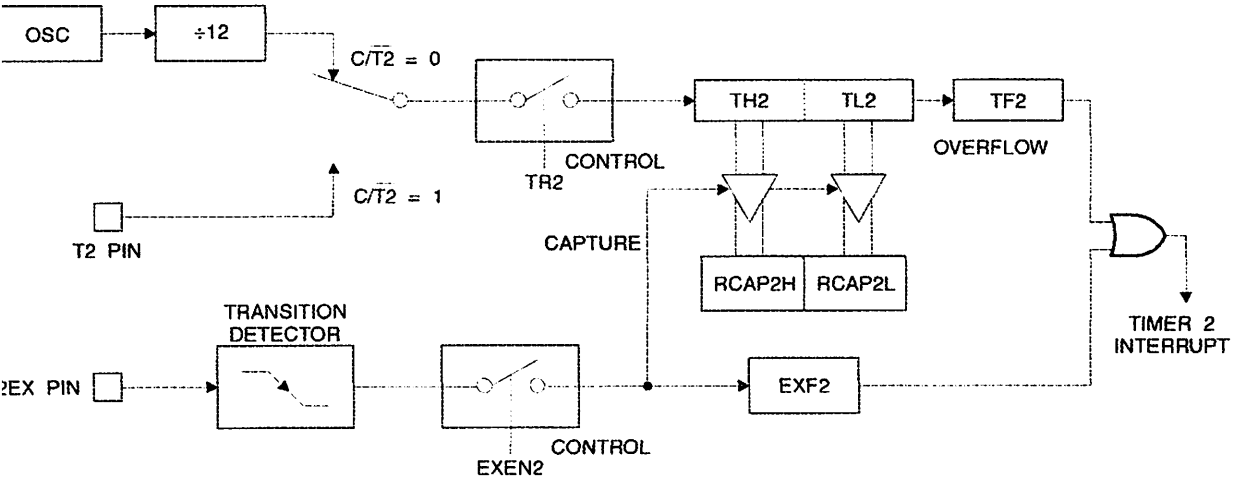
Table 8. Timer 2 Operating Modes

RCLK + TCLK	CP/$\overline{RL2}$	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)

ıre 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 I-to-0 transition at external input T2EX also causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt. The capture mode is illustrated in Figure 1.

1. Timer 2 in Capture Mode



-reload (Up or Down ıter)

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

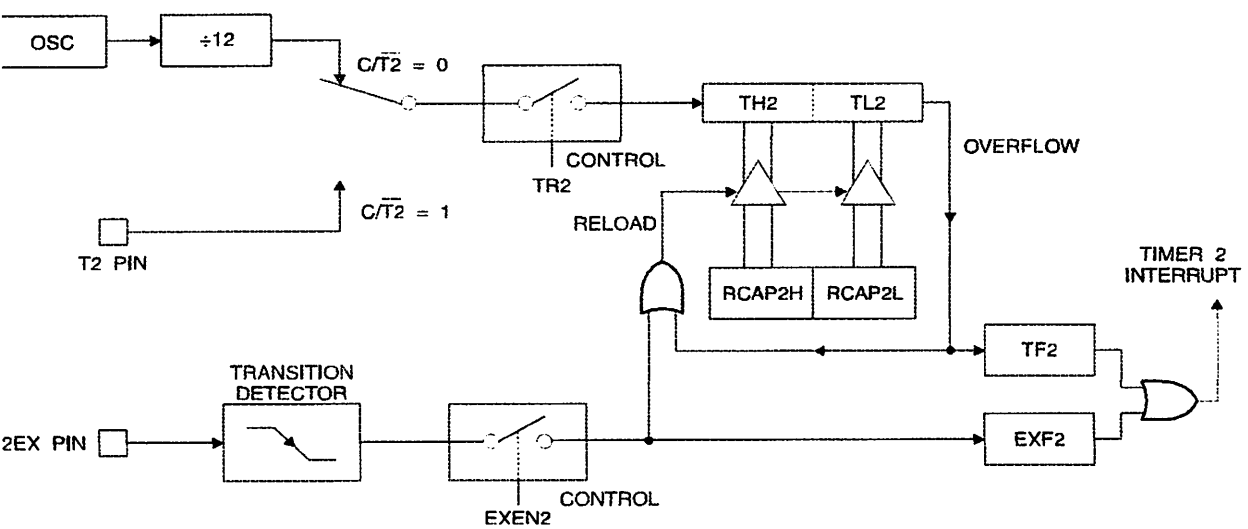
Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H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

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

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

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

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



9. T2MOD – Timer 2 Mode Control Register

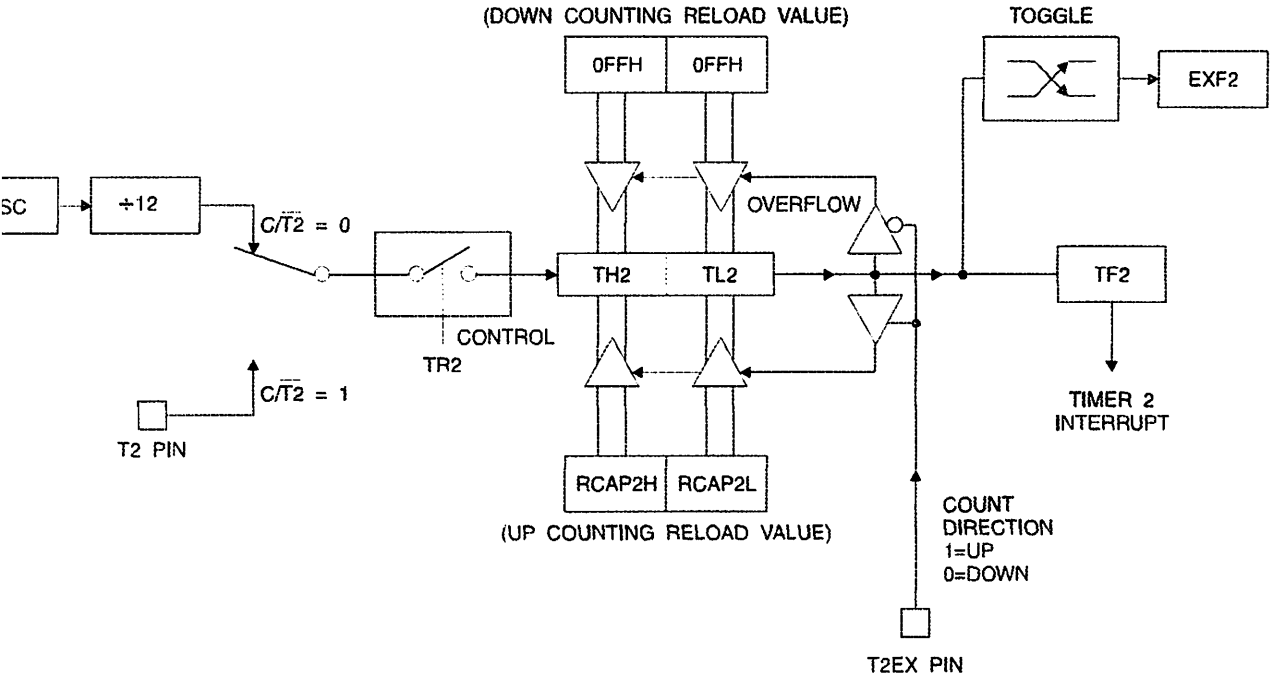
D Address = 0C9H — Reset Value = XXXX XX00B

t Addressable

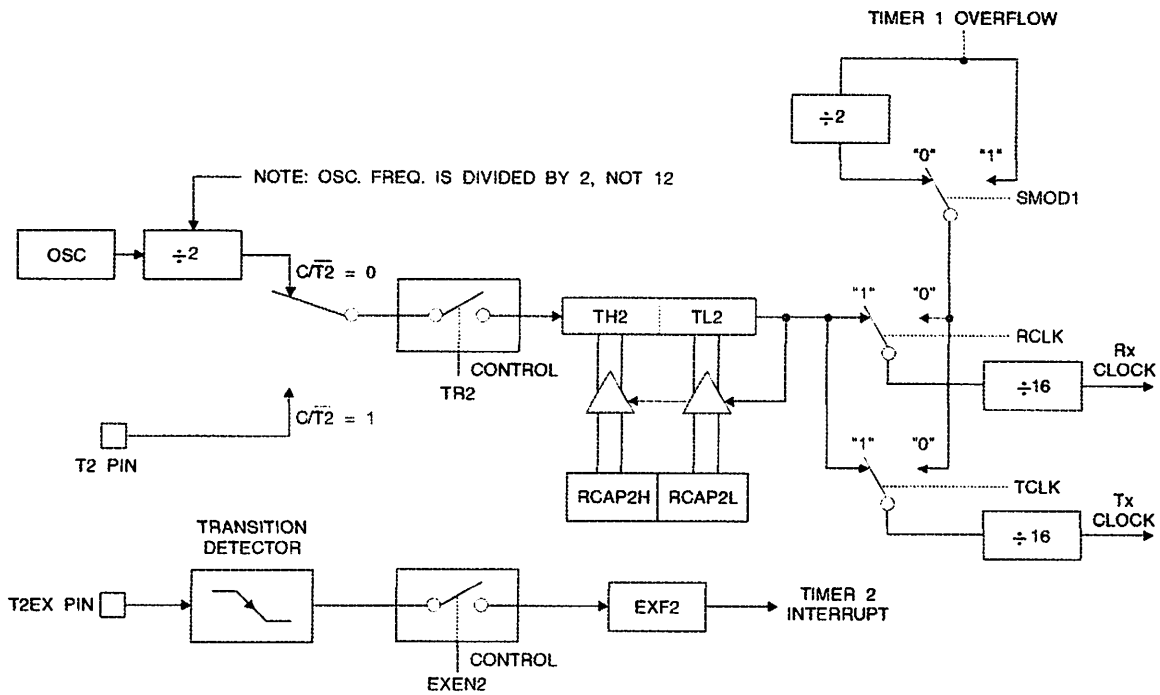
–	–	–	–	–	–	T2OE	DCEN
7	6	5	4	3	2	1	0

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

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



e 4. Timer 2 in Baud Rate Generator Mode



d Rate Generator

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

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

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

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

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

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

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

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

ɟrammable
;k Out

A 50% duty cycle clock can be programmed to come out on P1.0, as shown in Figure 5. This pin, besides being a regular I/0 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).

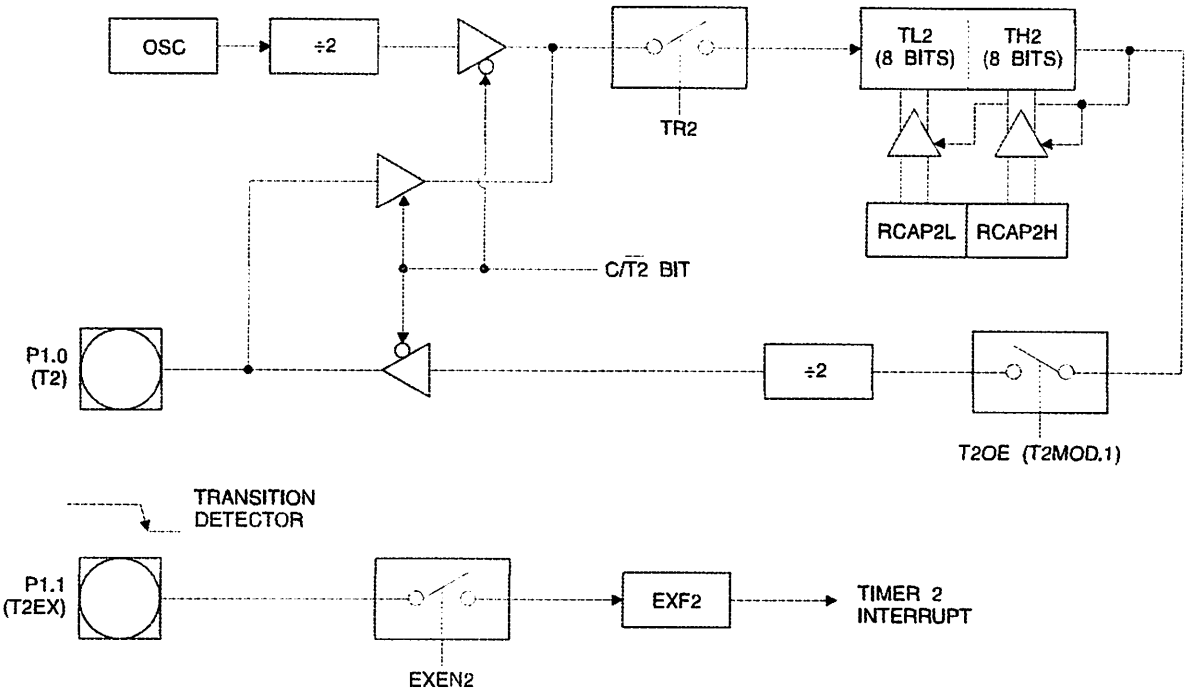
To configure the Timer/Counter 2 as a clock generator, bit C/$\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

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.

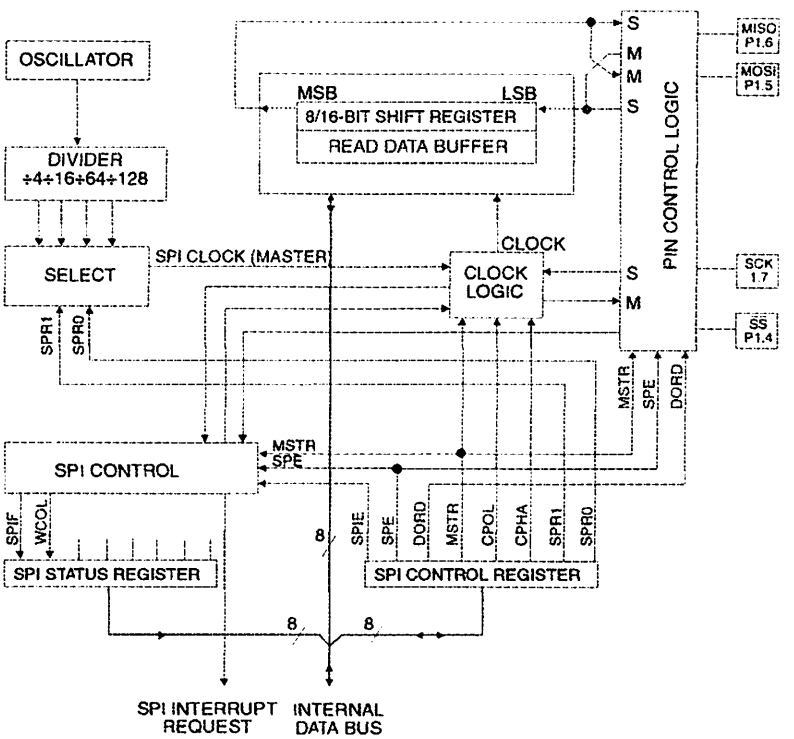
$$\text{Clock Out Frequency} = \frac{\text{Oscillator Frequency}}{4 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

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

5. Timer 2 in Clock-out Mode



6. SPI Block Diagram



T

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

al Peripheral face

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

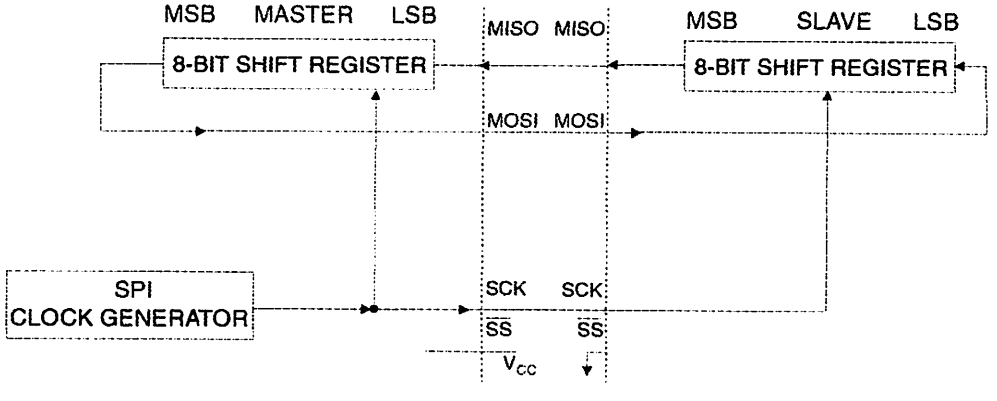
- Full-Duplex, 3-Wire Synchronous Data Transfer
- Master or Slave Operation
- 1.5 MHz Bit Frequency (max.)
- LSB First or MSB First Data Transfer
- Four Programmable Bit Rates
- End of Transmission Interrupt Flag
- Write Collision Flag Protection
- Wakeup from Idle Mode (Slave Mode Only)

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

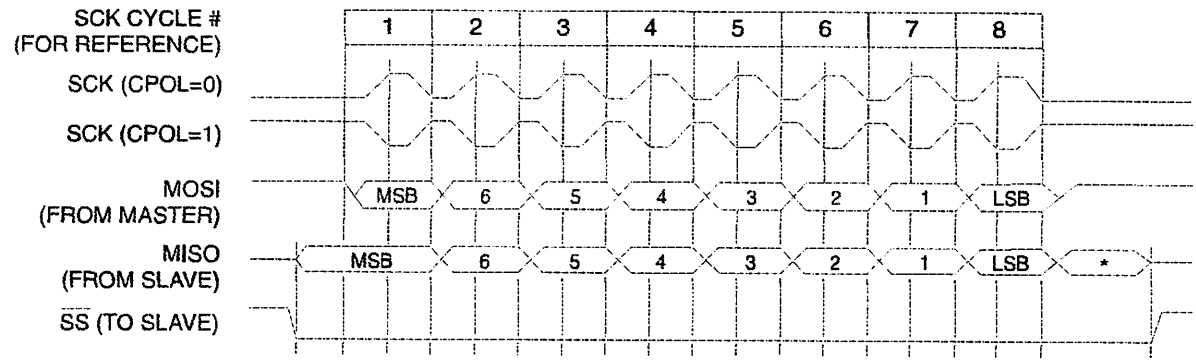
The Slave Select input, $\overline{SS}$/P1.4, is set low to select an individual SPI device as a slave. When $\overline{SS}$/P1.4 is set high, the SPI port is deactivated and the MOSI/P1.5 pin can be used as an input.

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

7. SPI Master-slave Interconnection

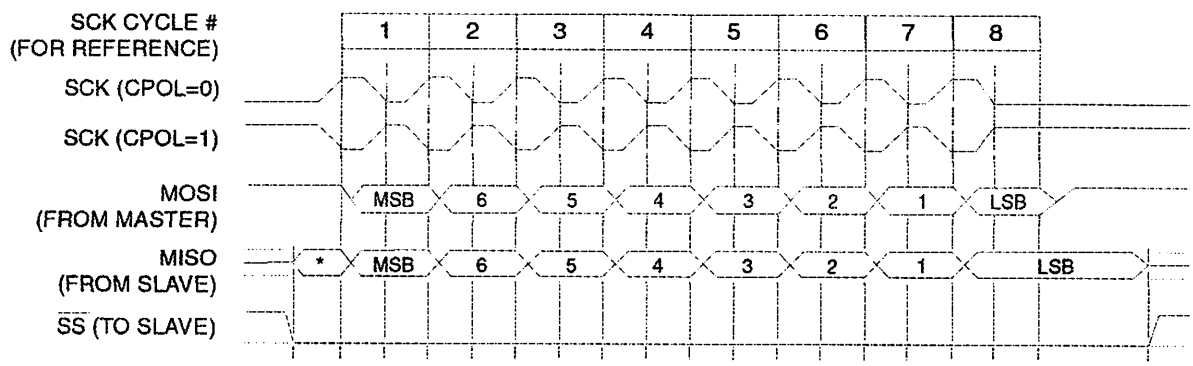


8. SPI transfer Format with CPHA = 0



*Not defined but normally MSB of character just received

9. SPI Transfer Format with CPHA = 1



*Not defined but normally LSB of previously transmitted character.

rupts

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

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

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

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

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

10. Interrupt Enable (IE) Register

3)(LSB)

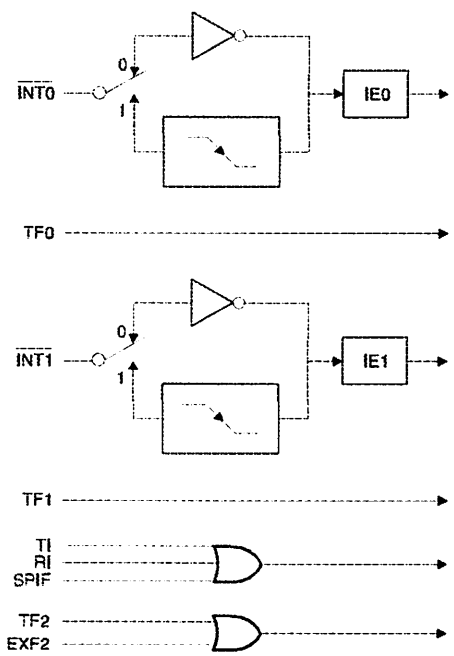
EA	–	ET2	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	SPI and UART interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.
software should never write 1s to unimplemented bits, because they may be used in future AT89 products.		

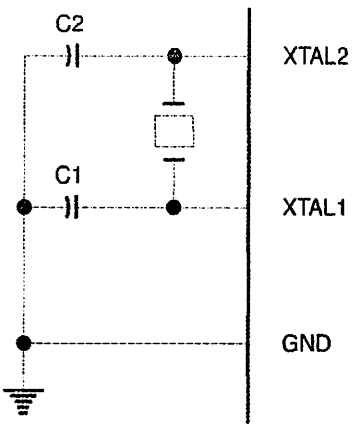
e 10. Interrupt Sources



illator racteristics

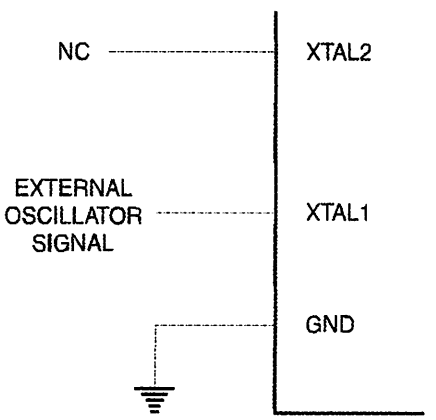
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected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

Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration



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.

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

er-down Mode

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

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

ram Memory Bits

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

When lock bit 1 is programmed, the logic level at the $\overline{EA}$ 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.

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

Bit Protection Modes[1][2]

Program Lock Bits			
LB1	LB2	LB3	Protection Type
U	U	U	No internal memory lock feature.
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 (parallel or serial mode) is disabled.
P	P	U	Same as Mode 2, but parallel or serial verify are also disabled.
P	P	P	Same as Mode 3, but external execution is also disabled.

1. U = Unprogrammed
2. P = Programmed

ɹramming the h and EEPROM

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

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

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

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

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

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

1. Power-up sequence:
   Apply power between V_{CC} and GND pins.
   Set RST pin to "H".
   Apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Set $\overline{PSEN}$ pin to "L"
   ALE pin to "H"
   $\overline{EA}$ pin to "H" and all other pins to "H".
3. Apply the appropriate combination of "H" or "L" logic levels to pins P2.6, P2.7, P3.6, P3.7 to select one of the programming operations shown in the Flash Programming Modes table.
4. Apply the desired byte address to pins P1.0 to P1.7 and P2.0 to P2.5.
   Apply data to pins P0.0 to P0.7 for Write Code operation.
5. Raise $\overline{EA}/V_{PP}$ to 12V to enable Flash programming, erase or verification.
6. Pulse ALE/$\overline{PROG}$ once to program a byte in the Code memory array, the Data memory array or the lock bits. The byte-write cycle is self-timed and typically takes 1.5 ms.
7. To verify the byte just programmed, bring pin P2.7 to "L" and read the programmed data at pins P0.0 to P0.7.
8. Repeat steps 3 through 7 changing the address and data for the entire 2K or 8K bytes array or until the end of the object file is reached.
9. Power-off sequence:
   Set XTAL1 to "L".
   Set RST and $\overline{EA}$ pins to "L".
   Turn V_{CC} power off.

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

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

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

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

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

In the serial programming mode, a chip erase operation is initiated by issuing the Chip Erase instruction. In this mode, chip erase is self-timed and takes about 16 ms.

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

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

The AT89S8252 is shipped with the Serial Programming Mode enabled.

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

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

ramming face

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

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

al Downloading

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

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

The Code and Data memory arrays have separate address spaces:

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

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

ıl Programming rithm

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

1. Power-up sequence:

   Apply power between VCC and GND pins.

   Set RST pin to "H".

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

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

   Set RST to "L".

   Turn V_{CC} power off.

ıl Programming uction

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

ction Set

ıction	Input Format			Operation
	Byte 1	Byte 2	Byte 3	
ramming Enable	1010 1100	0101 0011	xxxx xxxx	Enable serial programming interface after RST goes high.
Erase	1010 1100	xxxx x100	xxxx xxxx	Chip erase both 8K & 2K memory arrays.
Code Memory	aaaa a001	low addr	xxxx xxxx	Read data from Code memory array at the selected address. The 5 MSBs of the first byte are the high order address bits. The low order address bits are in the second byte. Data are available at pin MISO during the third byte.
Code Memory	aaaa a010	low addr	data in	Write data to Code memory location at selected address. The address bits are the 5 MSBs of the first byte together with the second byte.
Data Memory	00aa a101	low addr	xxxx xxxx	Read data from Data memory array at selected address. Data are available at pin MISO during the third byte.
Data Memory	00aa a110	low addr	data in	Write data to Data memory location at selected address.
Lock Bits	1010 1100	LB1 LB2 LB3 x x111	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

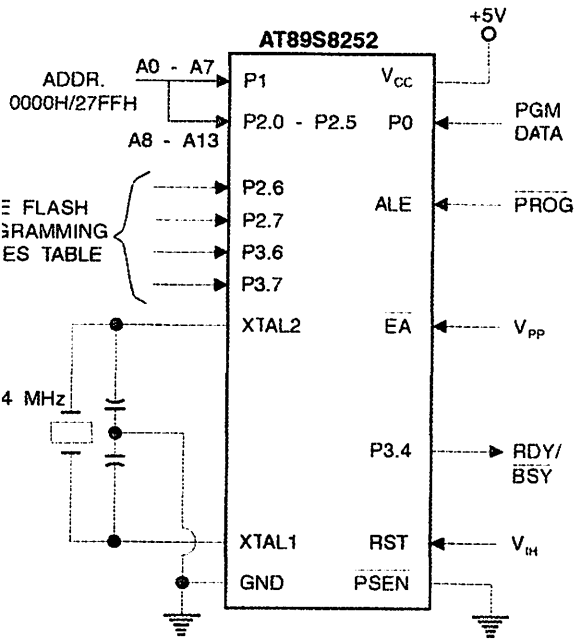
1. DATA polling is used to indicate the end of a byte write cycle which typically takes less than 2.5 ms at 5V.
2. "aaaaa" = high order address.
3. "x" = don't care.

;h and EEPROM Parallel Programming Modes

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

1. "h" = weakly pulled "High" internally.
2. Chip Erase and Serial Programming Fuse require a 10 ms PROG pulse. Chip Erase needs to be performed first before reprogramming any byte with a content other than FFH.
3. P3.4 is pulled Low during programming to indicate RDY/BSY.
4. "X" = don't care

e 13. Programming the Flash/EEPROM Memory



14. Verifying the Flash/EEPROM Memory

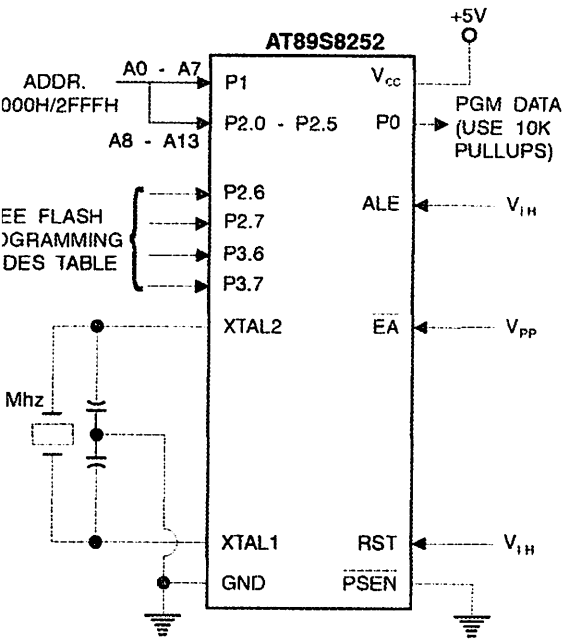
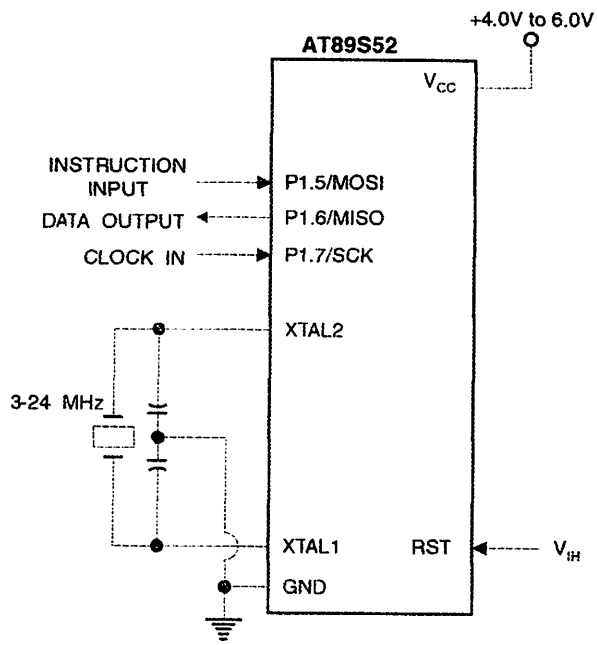


Figure 15. Flash/EEPROM Serial Downloading

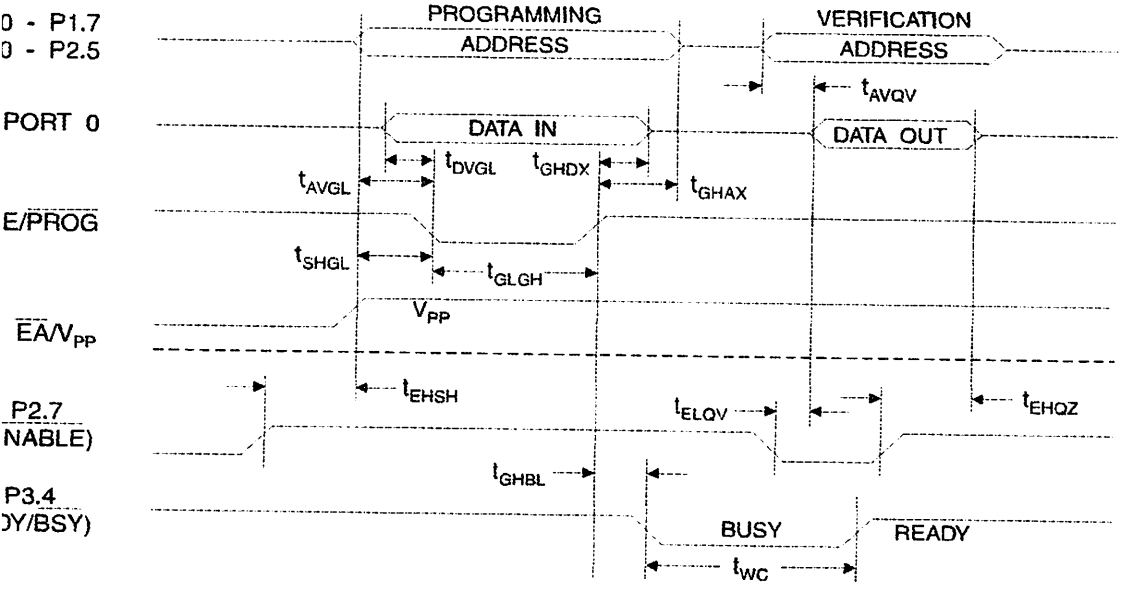


h Programming and Verification Characteristics – Parallel Mode

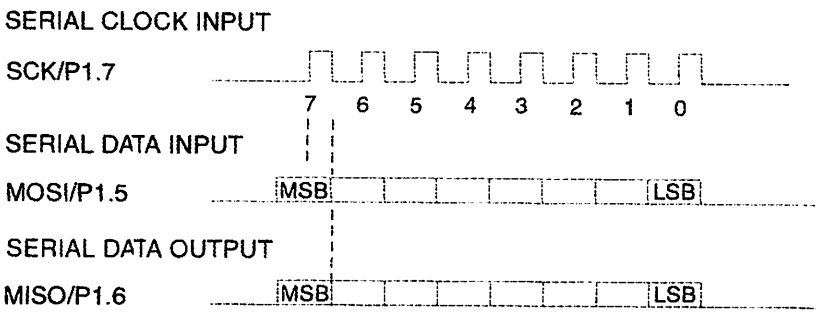
'C to 70°C, V_{CC} = 5.0V ± 10%

ol	Parameter	Min	Max	Units
	Programming Enable Voltage	11.5	12.5	V
	Programming Enable Current		1.0	mA
	Oscillator Frequency	3	24	MHz
	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
	Data Hold after $\overline{PROG}$	$48t_{CLCL}$		
	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
	$\overline{PROG}$ Width	1	110	µs
	Address to Data Valid		$48t_{CLCL}$	
	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
	Data Float after $\overline{ENABLE}$	0	$48t_{CLCL}$	
	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
	Byte Write Cycle Time		2.0	ms

ı/EEPROM Programming and Verification Waveforms – Parallel Mode

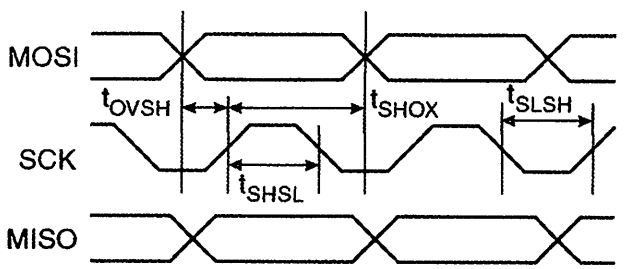


al Downloading Waveforms



l Programming Characteristics

16. Serial Programming Timing



11. Serial Programming Characteristics, T_A = -40°C to 85°C, V_{CC} = 4.0 - 6.0V (Unless Otherwise Noted)

ol	Parameter	Min	Typ	Max	Units
L	Oscillator Frequency	0		24	MHz
	Oscillator Period	41.6			ns
	SCK Pulse Width High	24 t_{CLCL}			ns
	SCK Pulse Width Low	24 t_{CLCL}			ns
	MOSI Setup to SCK High	t_{CLCL}			ns
	MOSI Hold after SCK High	2 t_{CLCL}			ns

olute Maximum Ratings*

ıting Temperature	-55°C to +125°C
ge Temperature	-65°C to +150°C
ıe on Any Pin lespect to Ground	-1.0V to +7.0V
ıum Operating Voltage	6.6V
utput Current	15.0 mA

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

Characteristics

lues shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 5.0V ± 20%, unless otherwise noted.

ol	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		0.5	V
	Output Low-voltage (1) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.5	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%		-650	µA
	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
	Reset Pull-down 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 (2)	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

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

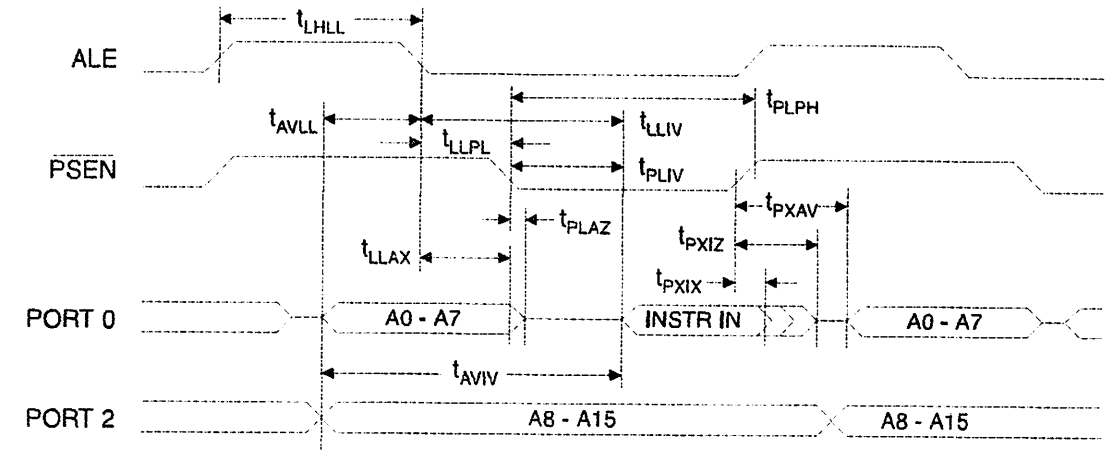
Characteristics

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

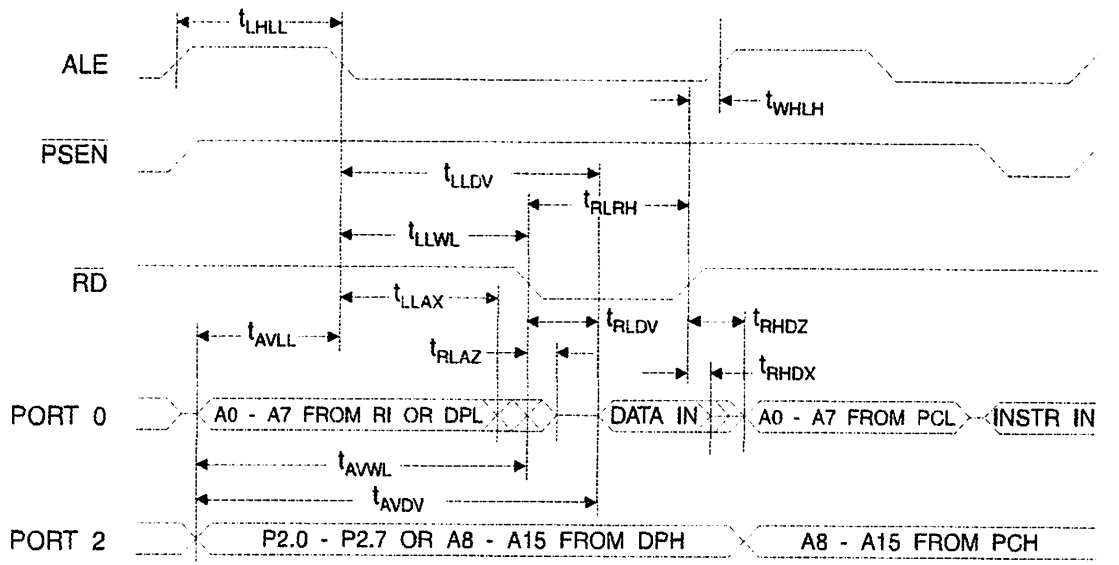
rnal Program and Data Memory Characteristics

bol	Parameter	Variable Oscillator		Units
		Min	Max	
CL	Oscillator Frequency	0	24	MHz
	ALE Pulse Width	$2t_{CLCL}$ - 40		ns
	Address Valid to ALE Low	t_{CLCL} - 13		ns
	Address Hold after ALE Low	t_{CLCL} - 20		ns
	ALE Low to Valid Instruction In		$4t_{CLCL}$ - 65	ns
	ALE Low to $\overline{PSEN}$ Low	t_{CLCL} - 13		ns
	$\overline{PSEN}$ Pulse Width	$3t_{CLCL}$ - 20		ns
	$\overline{PSEN}$ Low to Valid Instruction In		$3t_{CLCL}$ - 45	ns
	Input Instruction Hold after $\overline{PSEN}$	0		ns
	Input Instruction Float after $\overline{PSEN}$		t_{CLCL} - 10	ns
	$\overline{PSEN}$ to Address Valid	t_{CLCL} - 8		ns
	Address to Valid Instruction In		$5t_{CLCL}$ - 55	ns
	$\overline{PSEN}$ Low to Address Float		10	ns
	$\overline{RD}$ Pulse Width	$6t_{CLCL}$ - 100		ns
	$\overline{WR}$ Pulse Width	$6t_{CLCL}$ - 100		ns
	$\overline{RD}$ Low to Valid Data In		$5t_{CLCL}$ - 90	ns
	Data Hold after $\overline{RD}$	0		ns
	Data Float after $\overline{RD}$		$2t_{CLCL}$ - 28	ns
	ALE Low to Valid Data In		$8t_{CLCL}$ - 150	ns
	Address to Valid Data In		$9t_{CLCL}$ - 165	ns
	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	$3t_{CLCL}$ - 50	$3t_{CLCL}$ + 50	ns
	Address to $\overline{RD}$ or $\overline{WR}$ Low	$4t_{CLCL}$ - 75		ns
	Data Valid to $\overline{WR}$ Transition	t_{CLCL} - 20		ns
	Data Valid to $\overline{WR}$ High	$7t_{CLCL}$ - 120		ns
	Data Hold after $\overline{WR}$	t_{CLCL} - 20		ns
	$\overline{RD}$ Low to Address Float		0	ns
	$\overline{RD}$ or $\overline{WR}$ High to ALE High	t_{CLCL} - 20	t_{CLCL} + 25	ns

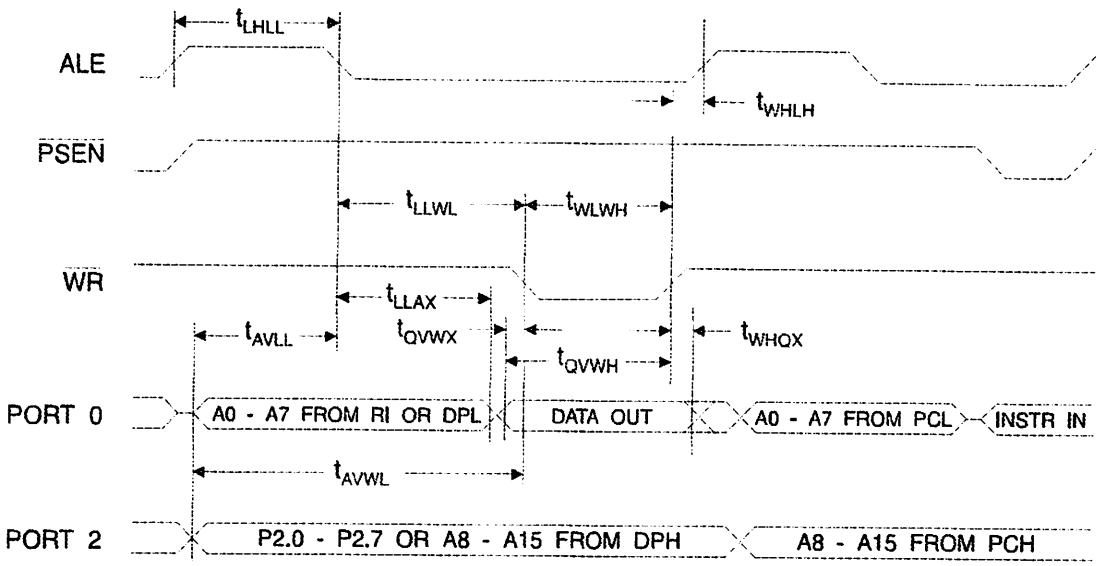
rnal Program Memory Read Cycle



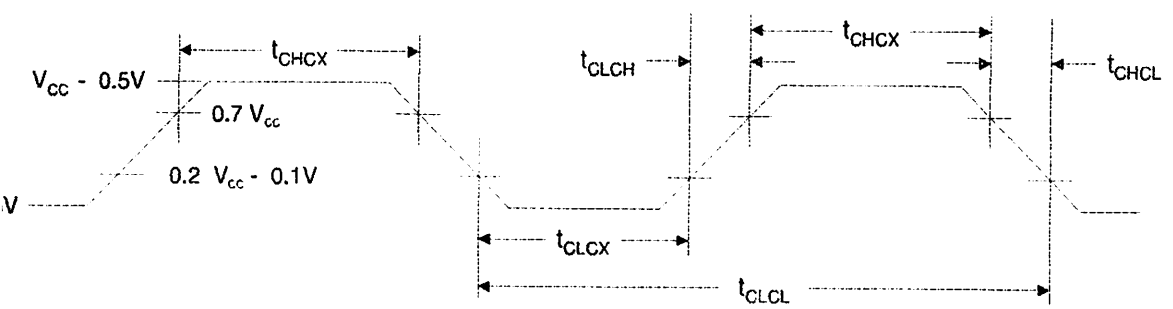
rnal Data Memory Read Cycle



rnal Data Memory Write Cycle



nal Clock Drive Waveforms



nal Clock Drive

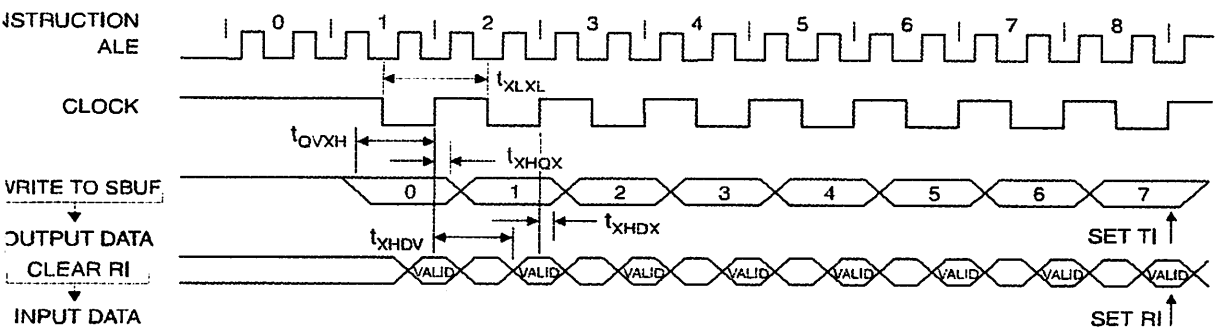
		V_{CC} = 4.0V to 6.0V		
	Parameter	Min	Max	Units
	Oscillator Frequency	0	24	MHz
	Clock Period	41.6		ns
	High Time	15		ns
	Low Time	15		ns
	Rise Time		20	ns
	Fall Time		20	ns

ial Port Timing: Shift Register Mode Test Conditions

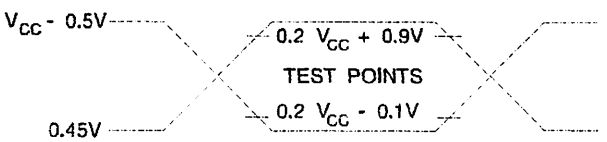
alues in this table are valid for V_{CC} = 4.0V to 6V and Load Capacitance = 80 pF.

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

t Register Mode Timing Waveforms

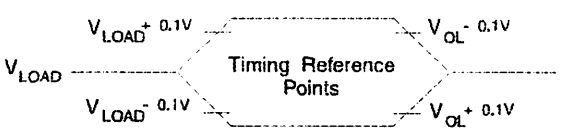


Testing Input/Output Waveforms[1]



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.

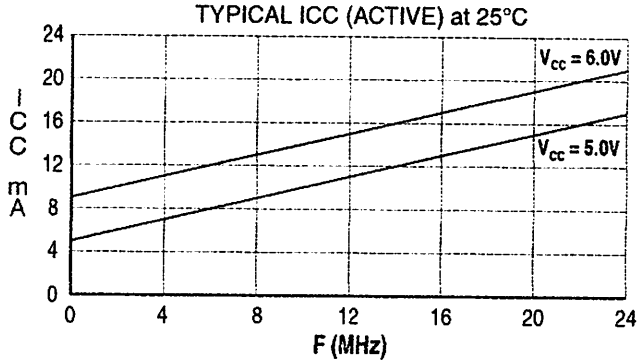
it Waveforms[1]



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.

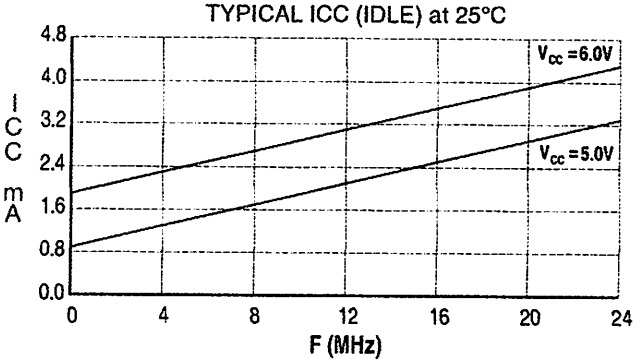
AT89S8252

TYPICAL ICC (ACTIVE) at 25°C



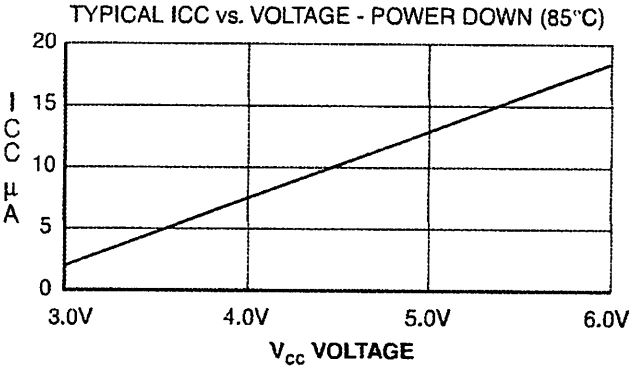
AT89S8252

TYPICAL ICC (IDLE) at 25°C



AT89S8252

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



Notes: 1. XTAL1 tied to GND for Icc (power-down)
2. Lock bits programmed

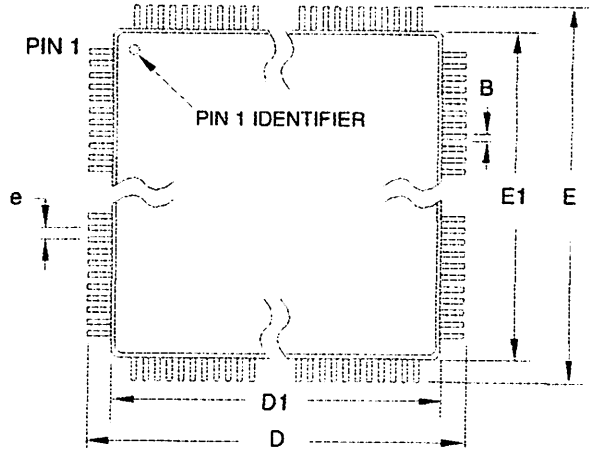
ring Information

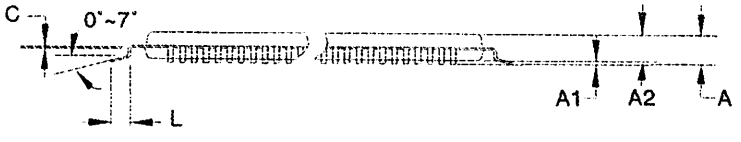
ed z)	Power Supply	Ordering Code	Package	Operation Range
	4.0V to 6.0V	AT89S8252-24AC AT89S8252-24JC AT89S8252-24PC	44A 44J 40P6	Commercial (0°C to 70°C)
	4.0V to 6.0V	AT89S8252-24AI AT89S8252-24JI AT89S8252-24PI	44A 44J 40P6	Industrial (-40°C to 85°C)

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

kaging Information

- TQFP





COMMON DIMENSIONS
(Unit of Measure = mm)

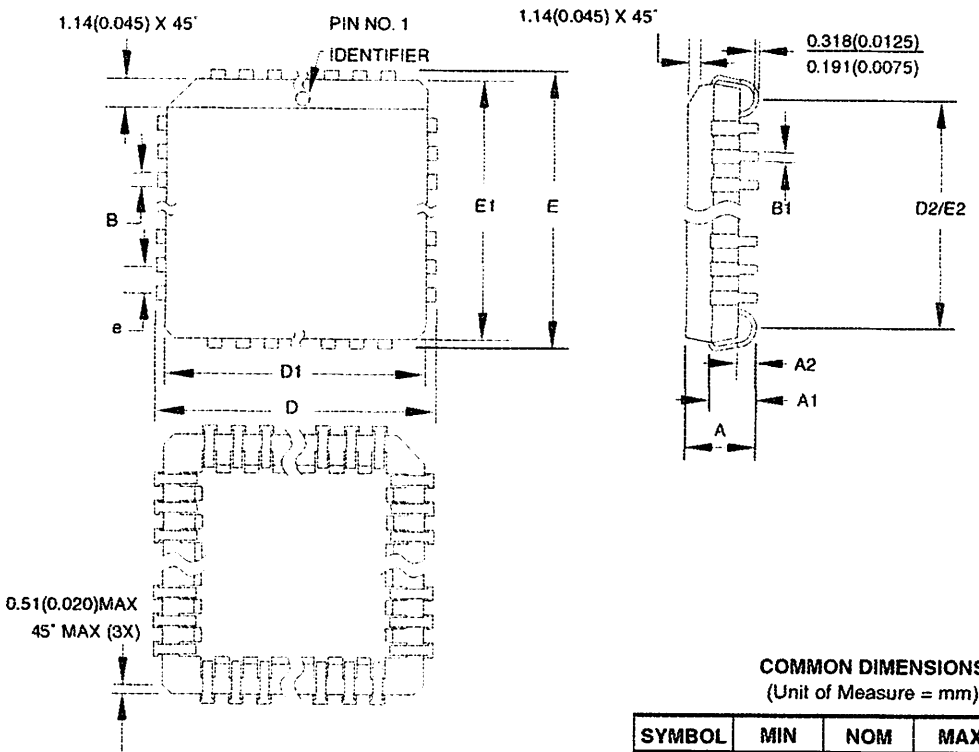
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

tes:
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

PLCC



COMMON DIMENSIONS
(Unit of Measure = mm)

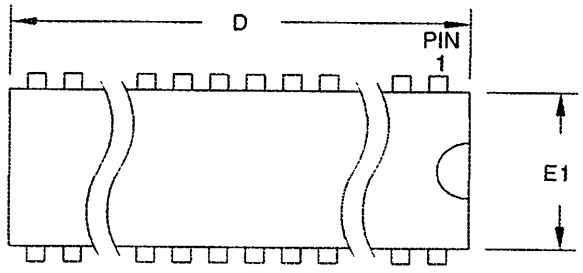
SYMBOL	MIN	NOM	MAX	NOTE
A	4.191	–	4.572	
A1	2.286	–	3.048	
A2	0.508	–	–	
D	17.399	–	17.653	
D1	16.510	–	16.662	Note 2
E	17.399	–	17.653	
E1	16.510	–	16.662	Note 2
D2/E2	14.986	–	16.002	
B	0.660	–	0.813	
B1	0.330	–	0.533	
e	1.270 TYP			

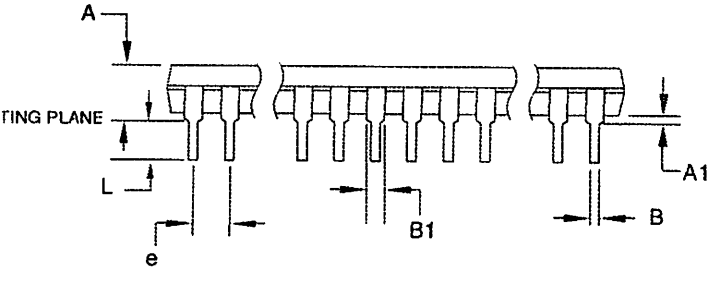
Notes:
1. This package conforms to JEDEC reference MS-018. Variation AC.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is .010"(0.254 mm) per side. Dimension D1 and E1 include mold mismatch and are measured at the extreme material condition at the upper or lower parting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

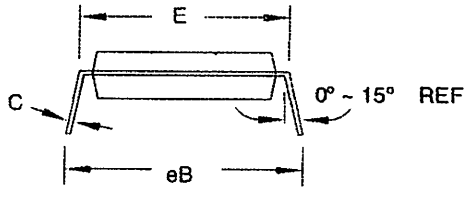
10/04/01

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

– 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			

tes:
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

el Corporation

5 Orchard Parkway
Jose, CA 95131, USA
1(408) 441-0311
: 1(408) 487-2600

onal Headquarters

le
el Sarl
te des Arsenaux 41
e Postale 80
1705 Fribourg
tzerland
(41) 26-426-5555
: (41) 26-426-5500

m 1219
iachem Golden Plaza
Mody Road Tsimshatsui
t Kowloon
g Kong
(852) 2721-9778
(852) 2722-1369

Tonetsu Shinkawa Bldg.
-8 Shinkawa
o-ku, Tokyo 104-0033
an
(81) 3-3523-3551
(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.

0401G–MICRO–3/06 xM

ures

patible with MCS-51® Products
ytes of In-System Programmable (ISP) Flash Memory
Endurance: 1000 Write/Erase Cycles
to 5.5V Operating Range
Static Operation: 0 Hz to 33 MHz
e-level Program Memory Lock
8-bit Internal RAM
ogrammable I/O Lines
16-bit Timer/Counters
terrupt Sources
)uplex UART Serial Channel
power Idle and Power-down Modes
upt Recovery from Power-down Mode
ndog Timer
Data Pointer
r-off Flag
Programming Time
le ISP Programming (Byte and Page Mode)

ription

39S51 is a low-power, high-performance CMOS 8-bit microcontroller with 4K
f in-system programmable Flash memory. The device is manufactured using
high-density nonvolatile memory technology and is compatible with the indus-
dard 80C51 instruction set and pinout. The on-chip Flash allows the program
to be reprogrammed in-system or by a conventional nonvolatile memory pro-
r. By combining a versatile 8-bit CPU with in-system programmable Flash on a
nic chip, the Atmel AT89S51 is a powerful microcontroller which provides a
exible and cost-effective solution to many embedded control applications.

9S51 provides the following standard features: 4K bytes of Flash, 128 bytes of
I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five-
vo-level interrupt architecture, a full duplex serial port, on-chip oscillator, and
cuitry. In addition, the AT89S51 is designed with static logic for operation
zero frequency and supports two software selectable power saving modes.
Mode stops the CPU while allowing the RAM, timer/counters, serial port, and
system to continue functioning. The Power-down mode saves the RAM con-
freezes the oscillator, disabling all other chip functions until the next external
or hardware reset.

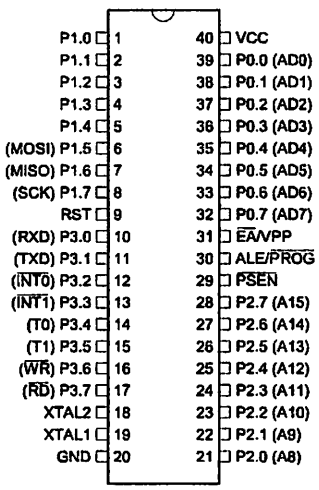
8-bit Microcontroller with 4K Bytes In-System Programmable Flash

AT89S51

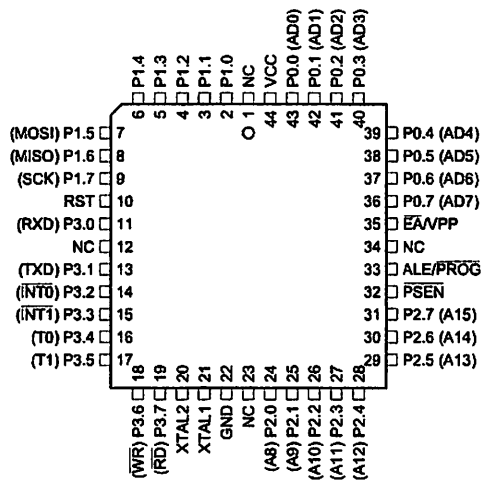
Rev. 2487A–10/01

Configurations

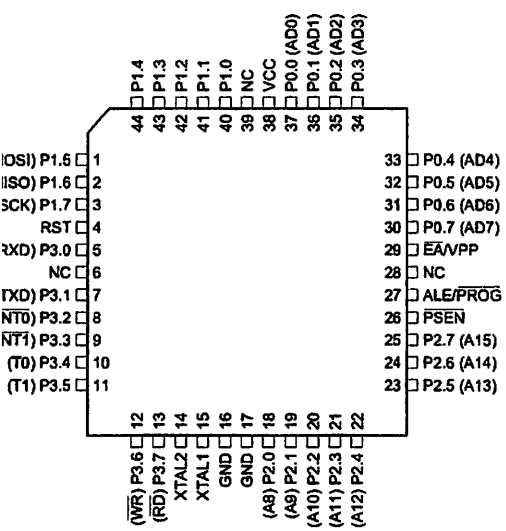
PDIP



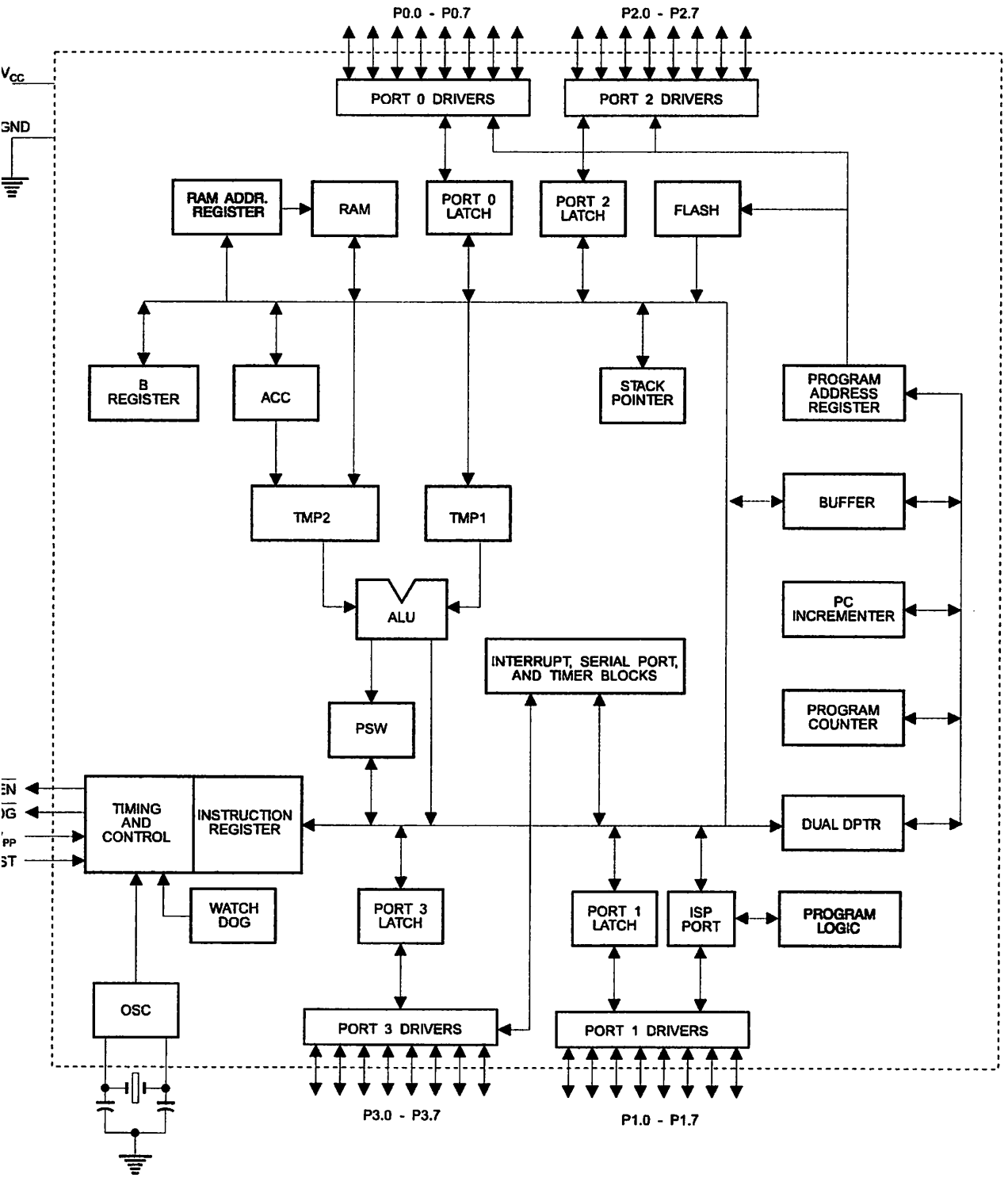
PLCC



TQFP



‹ Diagram


P0.0 - P0.7
P2.0 - P2.7
PORT 0 DRIVERS
PORT 2 DRIVERS
RAM ADDR. REGISTER
RAM
PORT 0 LATCH
PORT 2 LATCH
FLASH
B REGISTER
ACC
STACK POINTER
PROGRAM ADDRESS REGISTER
BUFFER
TMP2
TMP1
ALU
PC INCREMENTER
INTERRUPT, SERIAL PORT, AND TIMER BLOCKS
PSW
PROGRAM COUNTER
TIMING AND CONTROL
INSTRUCTION REGISTER
DUAL DPTR
WATCH DOG
PORT 3 LATCH
PORT 1 LATCH
ISP PORT
PROGRAM LOGIC
OSC
PORT 3 DRIVERS
PORT 1 DRIVERS
P3.0 - P3.7
P1.0 - P1.7

Description

Supply voltage.

Ground.

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

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

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

Port 2 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 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pull-ups.

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

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

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (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)

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.

$\overline{PROG}$

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.

$\overline{}$

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

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

PP

External Access Enable. $\overline{EA}$ 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, $\overline{EA}$ will be internally latched on reset.

$\overline{EA}$ 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

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

.2

Output from the inverting oscillator amplifier

ial tion sters

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

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

I. AT89S51 SFR Map and Reset Values

								0FFH
B 00000000								0F7H
								0EFH
ACC 00000000								0E7H
								0DFH
PSW 00000000								0D7H
								0CFH
								0C7H
IP XX000000								0BFH
P3 11111111								0B7H
IE 0X000000								0AFH
P2 11111111		AUXR1 XXXXXXX0				WDTRST XXXXXXXX		0A7H
SCON 00000000	SBUF XXXXXXXX							9FH
P1 11111111								97H
TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR XXX00XX0		8FH
P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000		PCON 0XXX0000	87H

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

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

Table 2. AUXR: Auxiliary Register

AUXR Address = 8EH Reset Value = XXX00XX0B

Not Bit Addressable

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

– Reserved for future expansion

DISALE Disable/Enable ALE

DISALE Operating Mode

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

1 ALE is active only during a MOVX or MOVC instruction

DISRTO Disable/Enable Reset out

DISRTO

0 Reset pin is driven High after WDT times out

1 Reset pin is input only

WDIDLE Disable/Enable WDT in IDLE mode

WDIDLE

0 WDT continues to count in IDLE mode

1 WDT halts counting in IDLE mode

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

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

Table 3. AUXR1: Auxiliary Register 1

AUXR1

Address = A2H

Reset Value = XXXXXXX0B

Not Bit Addressable

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

– Reserved for future expansion

DPS Data Pointer Register Select

DPS	
0	Selects DPTR Registers DP0L, DP0H
1	Selects DPTR Registers DP1L, DP1H

ıory anization

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.

ram Memory

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

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

Memory

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

chdog er e-time bled with et-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.

ıg 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.

T During er-down Idle

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

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

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

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

RT

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

er 0 and 1

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

rupts

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

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

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

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

Table 4. Interrupt Enable (IE) Register

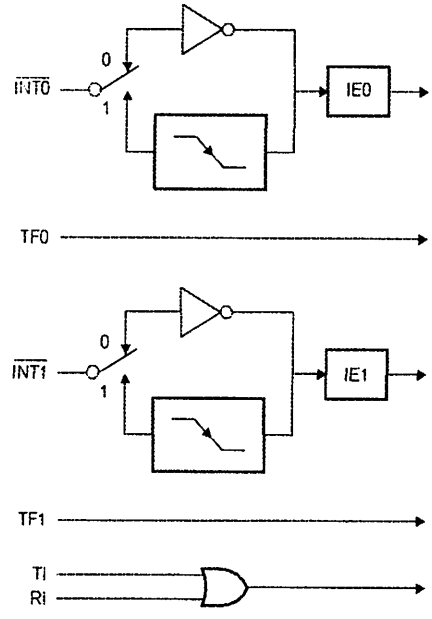
(MSB)							(LSB)
EA	–	–	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	Reserved
–	IE.5	Reserved
ES	IE.4	Serial Port interrupt enable bit
ET1	IE.3	Timer 1 interrupt enable bit
EX1	IE.2	External interrupt 1 enable bit
ET0	IE.1	Timer 0 interrupt enable bit
EX0	IE.0	External interrupt 0 enable bit
User software should never write 1s to reserved bits, because they may be used in future AT89 products.		

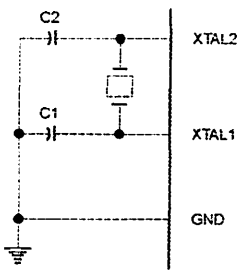
Figure 1. Interrupt Sources



llator racteristics

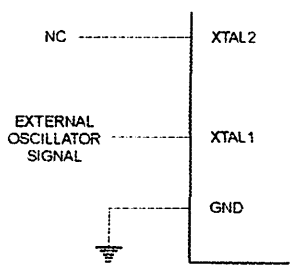
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected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

Figure 2. Oscillator Connections



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

Figure 3. External Clock Drive Configuration



Mode

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

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

r-down

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

Table 5. Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	$\overline{PSEN}$	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

ram ory Lock

The AT89S51 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

Table 6. Lock Bit Protection Modes

Program Lock Bits				
	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.

ramming lash – llel Mode

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

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

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

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise $\overline{EA}/V_{PP}$ to 12V.
5. Pulse ALE/$\overline{PROG}$ once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 50 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 AT89S51 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/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) = 51H indicates 89S51
(200H) = 06H

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

ıramming
⁼lash –
ıl 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.

ı
ramming
ithm

To program and verify the AT89S51 in the serial programming mode, the following sequence is recommended:

1. Power-up sequence:

   Apply power between VCC and GND pins.

   Set RST pin to "H".

   If a crystal is not connected across pins XTAL1 and XTAL2, apply a 3 MHz to 33 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Enable serial programming by sending the Programming Enable serial instruction to pin MOSI/P1.5. The frequency of the shift clock supplied at pin SCK/P1.7 needs to be less than the CPU clock at XTAL1 divided by 16.
3. The Code array is programmed one byte at a time in either the Byte or Page mode. The write cycle is self-timed and typically takes less than 0.5 ms at 5V.
4. Any memory location can be verified by using the Read instruction that returns the content at the selected address at serial output MISO/P1.6.
5. At the end of a programming session, RST can be set low to commence normal device operation.

Power-off sequence (if needed):

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

Set RST to "L".

Turn V_{CC} power off.

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

l ramming uction Set

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

ıramming ·face – ıllel Mode

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

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

7. Flash Programming Modes

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

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

Figure 4. Programmi ; the Flash Memory (Parallel Mode)

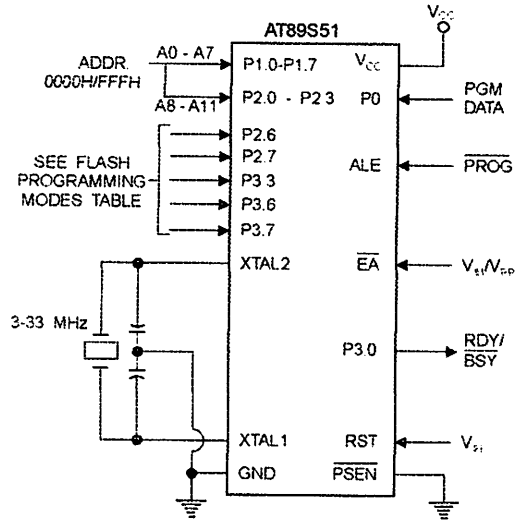
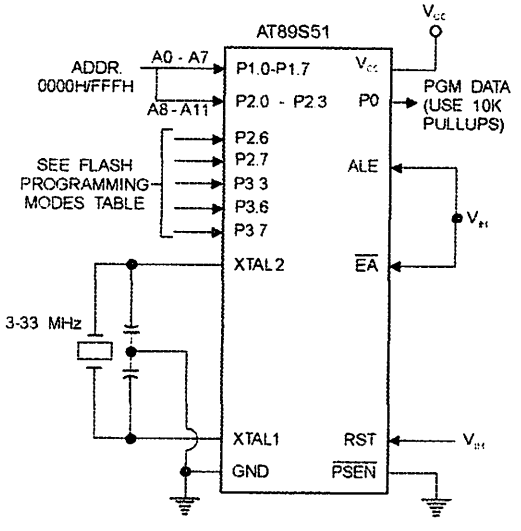


Figure 5. Verifying the Flash Memory (Parallel Mode)

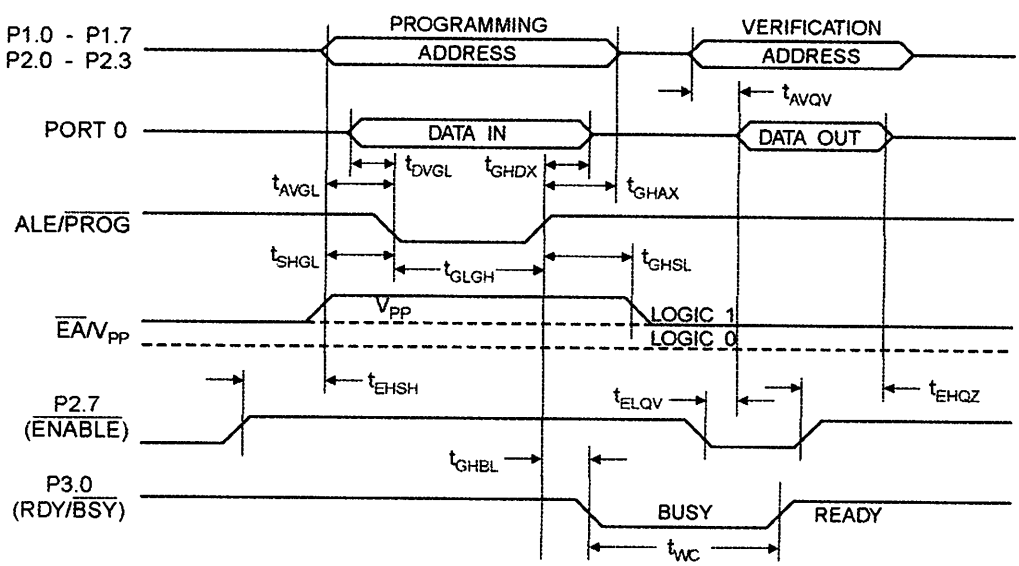


h Programming and Verification Characteristics (Parallel Mode)

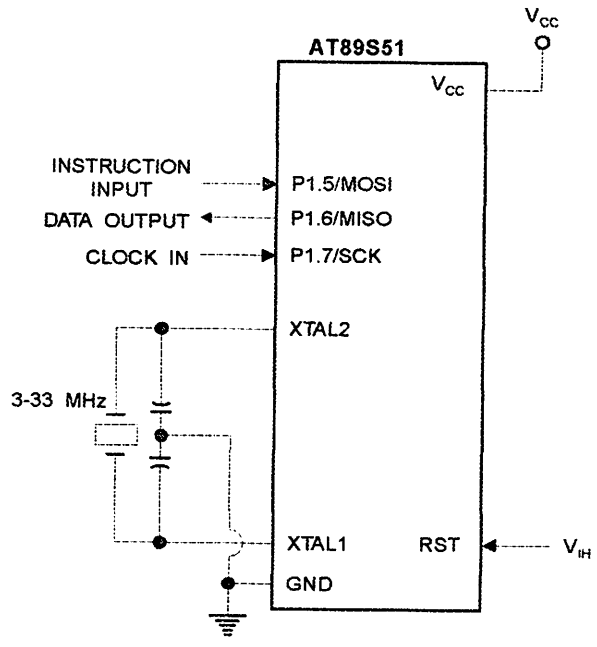
)°C to 30°C, V_{CC} = 4.5 to 5.5V

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

5. Flash Programming and Verification Waveforms – Parallel Mode

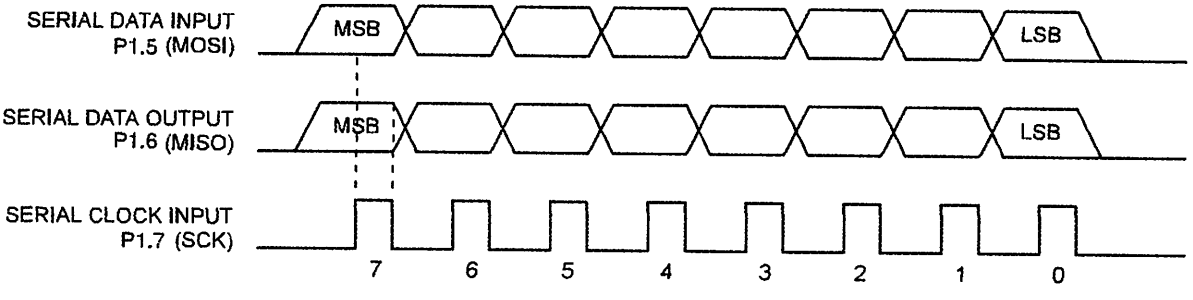


7. Flash Memory Serial Downloading



Programming and Verification Waveforms – Serial Mode

8. Serial Programming Waveforms



3. Serial Programming Instruction Set

ction	Instruction Format Byte 1	Byte 2	Byte 3	Byte 4	Operation
amming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx 0110 1001 (Output)	Enable Serial Programming while RST is high
Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash memory array
Program Memory Mode)	0010 0000	xxxx 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
Program Memory Mode)	0100 0000	xxxx 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
Lock Bits[2]	1010 1100	1110 00 B1 B2	xxxx xxxx	xxxx xxxx	Write Lock bits. See Note (2).
Lock Bits	0010 0100	xxxx xxxx	xxxx xxxx	xx LB3 LB2 LB1 xx	Read back current status of the lock bits (a programmed lock bit reads back as a "1")
Signature Bytes[1]	0010 1000	xxx A5 A4 A3 A2 A1	A0 xxx xxxx	Signature Byte	Read Signature Byte
Program Memory Mode)	0011 0000	xxxx A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Read data from Program memory in the Page Mode (256 bytes)
Program Memory Mode)	0101 0000	xxxx A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Write data to Program memory in the Page Mode (256 bytes)

1. The signature bytes are not readable in Lock Bit Modes 3 and 4.
2. B1 = 0, B2 = 0 →Mode 1, no lock protection
   B1 = 0, B2 = 1 →Mode 2, lock bit 1 activated
   B1 = 1, B2 = 0 →Mode 3, lock bit 2 activated
   B1 = 1, B1 = 1 →Mode 4, lock bit 3 activated

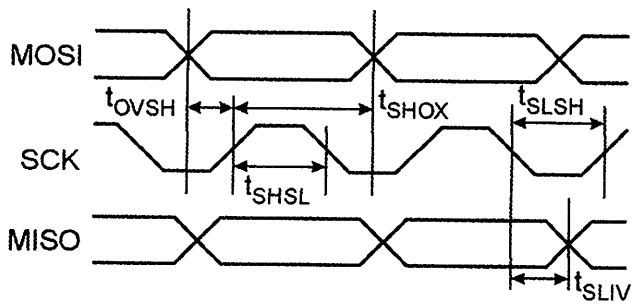
   } Each of the lock bits needs to be activated sequentially before Mode 4 can be executed.

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

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

ial Programming Characteristics

re 9. Serial Programming Timing



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

bol	Parameter	Min	Typ	Max	Units
CL	Oscillator Frequency	0		33	MHz
-	Oscillator Period	30			ns
-	SCK Pulse Width High	8 t_{CLCL}			ns
l	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
	Serial Byte Write Cycle Time			64 t_{CLCL} + 400	µs

olute Maximum Ratings*

ting Temperature	-55°C to +125°C
je Temperature	-65°C to +150°C
e on Any Pin espect to Ground	-1.0V to +7.0V
ium Operating Voltage	6.6V
itput Current	15.0 mA

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

Characteristics

lues 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.

ol	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%		-650	µA
	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
	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(2)	V_{CC} = 5.5V		50	µA

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

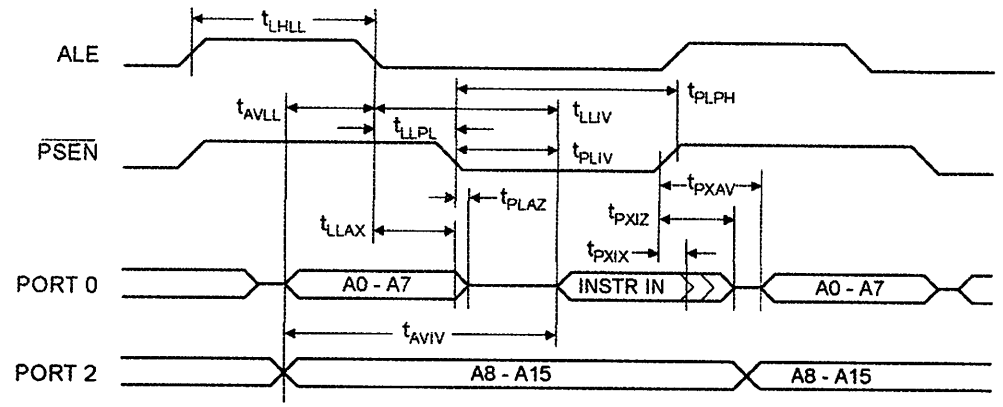
Characteristics

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

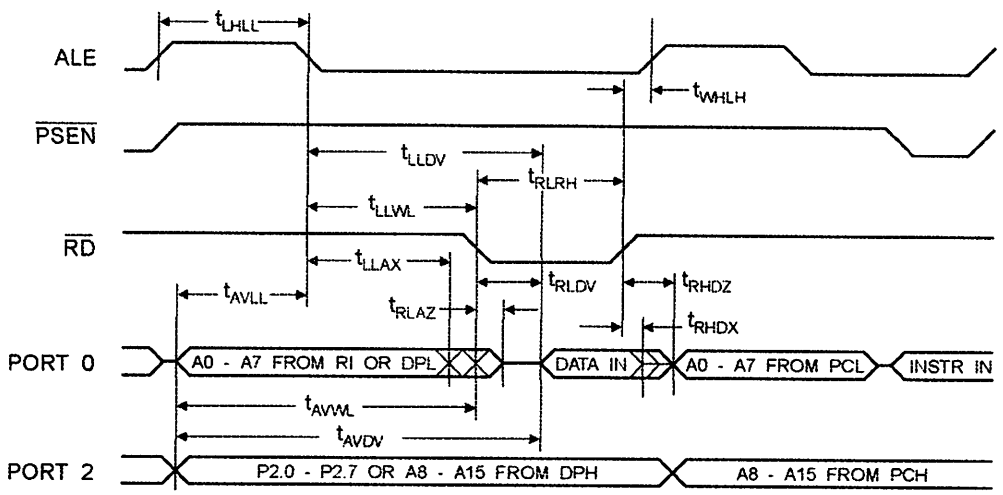
rnal Program and Data Memory Characteristics

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

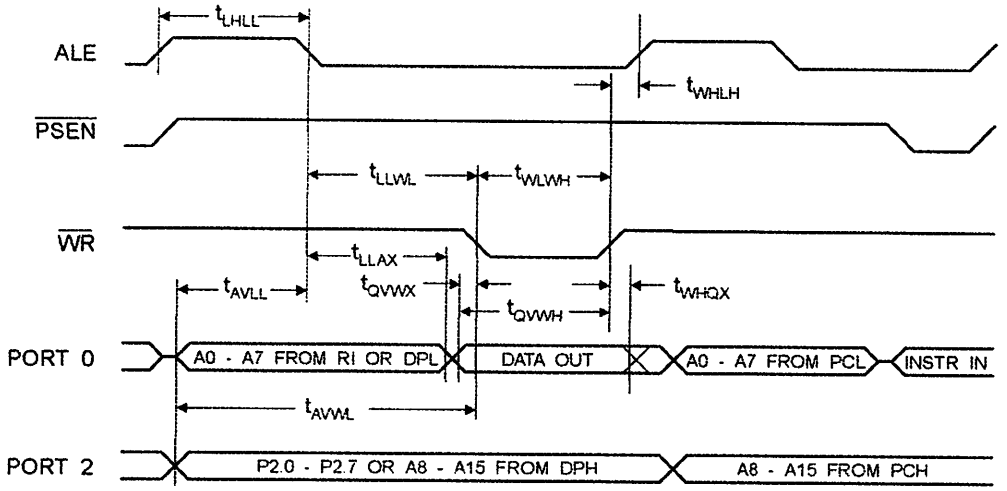
rnal Program Memory Read Cycle



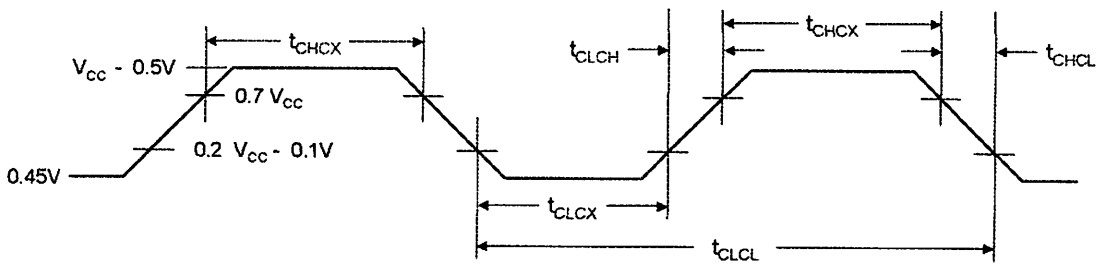
rnal Data Memory Read Cycle



rnal Data Memory Write Cycle



rnal Clock Drive Waveforms



nal Clock Drive

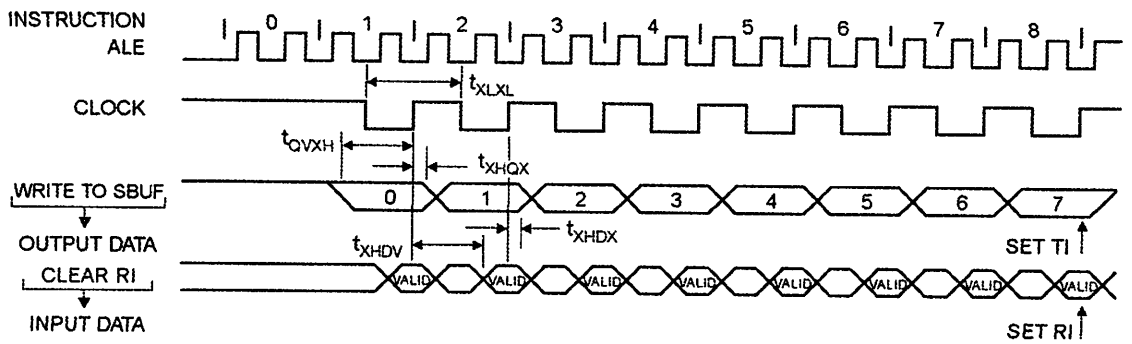
ol	Parameter	Min	Max	Units
	Oscillator Frequency	0	33	MHz
	Clock Period	30		ns
	High Time	12		ns
	Low Time	12		ns
	Rise Time		5	ns
	Fall Time		5	ns

al Port Timing: Shift Register Mode Test Conditions

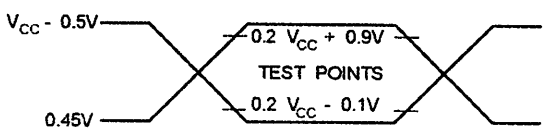
lues in this table are valid for V_{CC} = 4.0V to 5.5V and Load Capacitance = 80 pF.

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

Register Mode Timing Waveforms

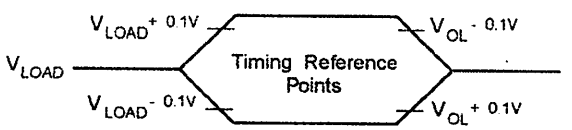


esting Input/Output Waveforms[1]



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.

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.

ering Information

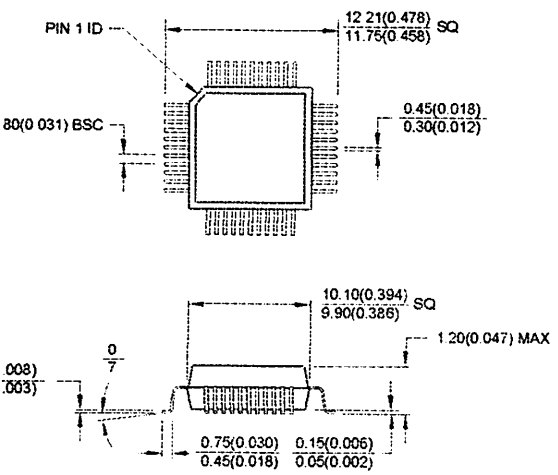
ed Iz)	Power Supply	Ordering Code	Package	Operation Range
	4.0V to 5.5V	AT89S51-24AC AT89S51-24JC AT89S51-24PC	44A 44J 40P6	Commercial (0° C to 70° C)
		AT89S51-24AI AT89S51-24JI AT89S51-24PI	44A 44J 40P6	Industrial (-40° C to 85° C)
	4.5V to 5.5V	AT89S51-33AC AT89S51-33JC AT89S51-33PC	44A 44J 40P6	Commercial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

] = Preliminary Availability

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

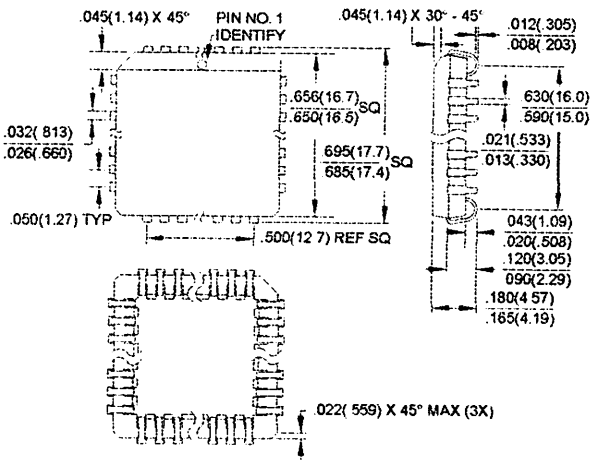
aging Information

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

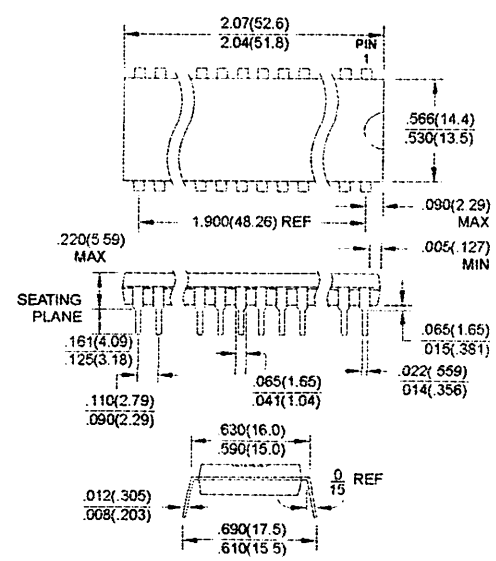


ntrolling dimension: millimeters

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



6, 40-pin, 0.600" Wide, Plastic Dual Inline
kage (PDIP)
ensions in Inches and (Millimeters)
EC STANDARD MS-011 AC



el Headquarters

ɔrate Headquarters
5 Orchard Parkway
ı Jose, CA 95131
. (408) 441-0311
((408) 487-2600

ɔe
el SarL
ıte des Arsenaux 41
a Postale 80
1705 Fribourg
tzerland
. (41) 26-426-5555
((41) 26-426-5500

el Asia, Ltd.
m 1219
ıachem Golden Plaza
Mody Road Tsimhatsui
t Kowloon
g Kong
(852) 2721-9778
(852) 2722-1369

el Japan K.K.
Tonetsu Shinkawa Bldg.
-8 Shinkawa
o-ku, Tokyo 104-0033
ın
(81) 3-3523-3551
(81) 3-3523-7581

Atmel Product Operations

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

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

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

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

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

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

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

2487A–10/01/xM

```
include <AT89s51.h>
include <intrins.h>
include <math.h>
include <stdio.h>

nsigned int i;
loat f;
define SUHU P3_0
define KELEMBABAN P3_1
define DATA P3_2
define SCK P3_3
define noACK 0
define ACK 1
define STATUS_REG_W 0x06 H
define STATUS_REG_R 0x07 H
define MEASURE_TEMP 0x03 H
define MEASURE_HUMI 0x05 H
define RESET        0x03 H

void connectionreset (void)
{
 unsigned char i;
 DATA=1; SCK=0;
 for(i=0;i<9;i++)
 { SCK=1;
   SCK=0;
 }
 transstart();
}

void transstart(void)
{
DATA=1; SCK=0;
_nop_();
SCK=1;
_nop_();
DATA=0;
_nop_();
SCK=0;
_nop_();_nop_();_nop_();
SCK=1;
_nop_();
DATA=1;
_nop_();
SCK=0
}
har tulis_byte(unsigned char value)
 {
  unsigned char i,error=0;
  for (i=0x80;i>0;i/=2)
  { if (i & value) DATA=1;
    else DATA=0;
    SCK=1;
    _nop_();_nop_();_nop_();
    SCK=0;
  }
 DATA=1;
```

```
 SCK=1;
 error=DATA;
 SCK=0;
}



char baca_byte (unsigned char ack)
{
unsigned char i, val=0;
DATA=1;
for (i=0x80;i>0;i/=2)
{ SCK=1;
  if (DATA) val=(val | i);
  SCK=0;
}
 DATA=!ack;
 SCK=1;
 _nop_();_nop_();_nop_();
 SCK=0;
 DATA=1;
 return val;
}

char baca_statusreg(unsigned char *p_value, unsigned char *p_checksum)

{
unsigned char error=0;
trnsstart();
error=s_write_byte(STATUS_REG _R);
*p_value=s_read byte(ACK);
*p_checksum=s_read byte(noACK);
 }


void calc_sth11_suhu(void)
{
  float t_C;
  t_C= - 40 + t*0.04;
  p1 = t_C;
}


 void calc_sth11_humi(void )
{ const float C1=-4.0;
  const float C2= 0.0405;
  const float C3=-0.0000028;
  const float T1= 0.01;
  const float T2= 0.00008;

  float rh_lin;
  float rh_true;
  float t_C;

  t_C=t*0.01 - 40;
  rh_lin=C3*rh*rh + C2*rh + C1;
  rh_true=(t_C-25)*(T1+T2*rh)+rh_lin;
```

```
 p1=rh_true;
}

void main()
 { value humi_val,temp_val;
  float dew_point;
  unsigned char error,checksum;
  unsigned int i;
  humi_val.f=(float)humi_val.i;
  temp_val.f=(float)temp_val.i;
  while(1)
  {
      if (SUHU == 0)
      {connectionreset();
       tulis_byte(MEASURE_TEMP);
       baca_statusreg();
       calc_sth11_suhu();
      }
       if (KELEMBABAN == 0)
       {
       connectionreset();
       tulis_byte(MEASURE_HUMI);
       baca_statusreg();
       calc_sth11_humi();
       }
    }

}
```

LISTING PROGRAM I

```
; LCD CONSTANTA
DISPCLR   EQU 00000001B
BLINK     EQU 00001101B
ENTRMOD   EQU 00000110B
DISPON    EQU 00001100B
CURSOR    EQU 00011100B
FUNCSET   EQU 00111000B
;
;DEVICE ADDRESS [LCD]
RS        BIT P3.2  ;LCD
E         BIT P3.3  ;LCD
WMCON     DATA 96H
EEMEN     EQU  00001000B
EEMWE     EQU  00010000B
WDTRST    EQU  00000010B
address EQU  7ffh

BLN     EQU 2EH
TGL     EQU 2FH
DTK     EQU 30H
MNT     EQU 31H
JAM     EQU 32H
STATDTK  EQU 0
suhu1a  EQU 34H
suhu1b  EQU 35H
suhu2a  EQU 36H
suhu2b  EQU 37H
klb1a   EQU 38H
klb1b   EQU 39H
klb2a   EQU 3AH
klb2b   EQU 3BH
suhu1c  EQU 3CH
suhu2c  EQU 3DH
klb1c   EQU 3Eh
klb2c   EQU 3FH

simpansht2   EQU 40H
simpansht1   EQU 33H
simpanklb1  EQU 41H
simpanklb2  EQU 42H

jam9sht1   EQU 7eH
jam9klb1   EQU 7fH
jam9sht2   EQU 43H
jam9klb2   EQU 44H
```

```
jam10sht1  EQU 45H
jam10klb1  EQU 46H
jam10sht2  EQU 47H
jam10klb2  EQU 48H

jam11sht1  EQU 49H
jam11klb1  EQU 4AH
jam11sht2  EQU 4BH
jam11klb2  EQU 4CH

jam12sht1  EQU 4DH
jam12klb1  EQU 4EH
jam12sht2  EQU 4FH
jam12klb2  EQU 50H

jam13sht1  EQU 51H
jam13klb1  EQU 52H
jam13sht2  EQU 53H
jam13klb2  EQU 54H

jam14sht1  EQU 55H
jam14klb1  EQU 56H
jam14sht2  EQU 57H
jam14klb2  EQU 58H

jam15sht1  EQU 59H
jam15klb1  EQU 5AH
jam15sht2  EQU 5BH
jam15klb2  EQU 5CH

jam16sht1  EQU 5DH
jam16klb1  EQU 5EH
jam16sht2  EQU 5FH
jam16klb2  EQU 60H

jam17sht1  EQU 61H
jam17klb1  EQU 62H
jam17sht2  EQU 63H
jam17klb2  EQU 64H

jam18sht1  EQU 65H
jam18klb1  EQU 66H
jam18sht2  EQU 67H
jam18klb2  EQU 68H

jam19sht1  EQU 69H
jam19klb1  EQU 6AH
jam19sht2  EQU 6BH
```

```
jam19klb2  EQU 6€H

jam20sht1  EQU 6DH
jam20klb1  EQU 6EH
jam20sht2  EQU 6FH
jam20klb2  EQU 70H

jam21sht1  EQU 71H
jam21klb1  EQU 72H
jam21sht2  EQU 73H
jam21klb2  EQU 74H

jam22sht1  EQU 75H
jam22klb1  EQU 76H
jam22sht2  EQU 77H
jam22klb2  EQU 78H

jam8sht1  EQU 79H
jam8klb1  EQU 7AH
jam8sht2  EQU 7BH
jam8klb2  EQU 7CH


ISR_INTO:
        ORG   3H
        RETI
ISR_TIMER0:
        ORG   0BH
        CLR   TR0
        MOV   TL0,#0ECH
;RELOAD TIMER1
        MOV  TH0,#0F0H

        LCALL       HITUNG_WAKTU
        LCALL       DSPJAM
        RETI
ISR_TIMER1:
        ORG   1BH
        SETB  TR0
        CLR   TR0
        RETI

        JMP   MULAI
        ORG   23H
        CLR   ES
        JNB   RI,$
        CLR   RI
        MOV   R1,SBUF
```

```
        MOV  DPTR,#TRANS1
        LCALL        PRINTSTRING1
        MOV  DPTR,#TRANS2
        LCALL        PRINTSTRING2
KIRIM_LAGI:
        MOV  address,#000h
        MOV  A,address
        INC  A
        MOV  address,A
        ORL  WMCOM,#EEMEN
        MOV  DPTR,address
        MOVX         A,@DPTR
        XRL  WMCOM,#EEMEN
        CJNE A,#00H,KIRIM
        CALL delaytaM
        MOV  DPTR,#PILIH3
        LCALL        PRINTSTRING2
        SETB ES
        RETI



KIRIM:
        ;MOV A,A
        ADD  A,#30H
        MOV  B,A
        CLR  ES      ; matikan serial interupt saat mengirim
        MOV  A,B
        MOV  SBUF,A        ; isi serial buffer dengan data yg dikirim
        JNB  TI,$    ; tunggu pengiriman selesai
        CLR  TI      ; clear transmit interupt flag
        JMP  KIRIM_LAGI


delaytam:
        MOV  R7,#50
LOPAA:  MOV  R6,#50
LOPAB:  MOV  R5,#50
        DJNZ R5,$
        DJNZ R6,LOPAB
        DJNZ R7,LOPAA
        RET


MULAI:
        MOV  SCON,#50H
        MOV  TMOD,#20H
        MOV  TL1,#0FDH
```

```
        MOV  TH1,#0FDH
        SETB TR1
        SETB EA             ; enable interupt
        SETB ES             ; enable serial interupt

        LCALL       INIT_LCD


INISIALISASI:

;*************************
;* INISIALISASI LCD    *
;*************************
DELAY_INIT_LCD:
        MOV R6,#20H

DLY_LCD_LP:

        MOV   R7,#0
        DJNZ  R7,$
        DJNZ  R6,DLY_LCD_LP
        RET

INIT_LCD:
        SETB RS
        CLR E
        MOV A,#DISPCLR
        LCALL CONTROLOUT
        LCALL DELAY_INIT_LCD
        MOV    A,#FUNCSET
        LCALL  CONTROLOUT
        MOV    A,#DISPON
        LCALL  CONTROLOUT
        MOV    A,#ENTRMOD
        LCALL  CONTROLOUT
        MOV  DPTR,#NAMA
        LCALL       PRINTSTRING1
        MOV  DPTR,#SEKOLAH
        LCALL       PRINTSTRING2
        MOV  R7,#100
LOOP2:  MOV  R6,#100
LOOP1:  MOV  R5,#100
        DJNZ R5,$
        DJNZ R6,LOOP1
        DJNZ R7,LOOP2

        MOV  R7,#100
LOOP4:  MOV  R6,#100
```

```
LOOP3:      MOV  R5,#100
        DJNZ  R5,$
        DJNZ  R6,LOOP3
        DJNZ  R7,LOOP4

        MOV  DPTR,#PILIH1
        LCALL       PRINTSTRING1
        MOV  DPTR,#PILIH3
        LCALL       PRINTSTRING2

        MOV  R7,#100
LOOP2A:     MOV  R6,#100
LOOP1A:     MOV  R5,#100
        DJNZ  R5,$
        DJNZ  R6,LOOP1A
        DJNZ  R7,LOOP2A


COBA:
WAKTU:      MOV  TMOD,#00100001B
        ;T1 MODE 2, TO MODE 3
        MOV  TH1,#25H
        MOV  TL1,TH1
        MOV  TL0,#0ECH
        MOV  TH0,#0F0H
        MOV  IE,#10001010B
        ;ENEBLE ALL INTRAF: TIMER0&1
        SETB  IP.1
        SETB  TR1    ;T1 RUN
TIMERON:
        MOV  DTK,#00H
        MOV  MNT,#48H
        MOV  JAM,#08H
        MOV  TGL,#06H
        MOV    BLN,#03H
        MOV  address,#000h
putar sensor:
baca_sht:
        PUSH ACC
        SETB  P3.4
        SETB  P3.5
        SETB  P3.6
        SETB  P3.7
        CLR   P3.4
        CLR   P3.4

        NOP
        NOP
```

```
        NOP
        NOP
        NOP
        NOP
;*************************************************************
        MOV  suhu1a,P2
        MOV  A,suhu1a
        MOV  B,#10
        DIV   AB
        MOV  suhu1a,B
        MOV  B,#10
        DIV   AB
        MOV  suhu1b,B
        MOV  suhu1c,A
        MOV A,#1
        LCALL       POSISI2

        MOV  A,suhu1c
        ADD   A,#30H
        LCALL       DATAOUT
        MOV A,#2
        LCALL       POSISI2
        MOV  A,suhu1b
        ADD   A,#30H
        LCALL       DATAOUT
        MOV A,#3
        LCALL       POSISI2
        MOV  A,suhu1a
        ADD   A,#30H
        LCALL       DATAOUT
        MOV A,#4
        MOV  A,#'|'
        LCALL       DATAOUT
        SETB  P3.4
        SETB  P3.5
        SETB  P3.6
        SETB  P3.7
        CLR   P3.5
        CLR   P3.5
        CALL delayyy
;*********************************************
        MOV klb1a,P2
        MOV A,klb1a
        MOV  B,#10
        DIV   AB
        MOV  klb1a,B
        MOV  B,#10
        DIV   AB
```

```
        MOV  klb1b,B
        MOV  klb1c,A

        MOV A,#5
        LCALL       POSISI2
        MOV  A,klb1c
        ADD  A,#30H
        LCALL       DATAOUT
        MOV A,#6
        LCALL       POSISI2
        MOV  A,klb1b
        ADD  A,#30H
        LCALL       DATAOUT
        MOV A,#7
        LCALL       POSISI2
        MOV  A,klb1a
        ADD  A,#30H
        LCALL       DATAOUT

;***********************************************
        SETB P3.4
        SETB P3.5
        SETB P3.6
        SETB P3.7
        CLR  P3.6
        CLR  P3.6
        CALL delayyy
        CALL delayyy
;***********************************************
        MOV suhu2a,P1
        MOV  A,suhu2a
        MOV  B,#10
        DIV   AB
        MOV  suhu2a,B
        MOV  B,#10
        DIV   AB
        MOV  suhu2b,B
        MOV  suhu2c,A
        MOV A,#10
        LCALL       POSISI2

        MOV  A,suhu2c
        ADD  A,#30H
        LCALL       DATAOUT
        MOV A,#11
        LCALL       POSISI2
        MOV  A,suhu2b
        ADD  A,#30H
```

```
        LCALL       DATAOUT
        MOV A,#12
        LCALL       POSISI2
        MOV  A,suhu2a
        ADD  A,#30H
        LCALL       DATAOUT
        MOV A,#13
        MOV  A,#'|'
        LCALL       DATAOUT


        SETB  P3.4
        SETB  P3.5
        SETB  P3.6
        SETB  P3.7
        CLR   P3.7
        CLR   P3.7
        CLR   P3.7
        CALL delayyy
;*****************************************************
        MOV klb2b,P1
        MOV A,klb2b
        MOV  B,#10
        DIV   AB
        MOV  klb2a,B
        MOV  B,#10
        DIV   AB
        MOV  klb2b,B
        MOV  klb2c,A

        MOV A,#14
        LCALL       POSISI2
        MOV  A,klb2c
        ADD  A,#30H
        LCALL       DATAOUT
        MOV A,#15
        LCALL       POSISI2
        MOV  A,klb2b
        ADD  A,#30H
        LCALL       DATAOUT
        MOV A,#16
        LCALL       POSISI2
        MOV  A,klb2a
        ADD  A,#30H
        LCALL       DATAOUT
        CALL delayyy
        POP   ACC
        JMP   putar sensor
```

```
HITUNG_WAKTU:
        PUSH  ACC
        CLR   C
        MOV   A,DTK         ;INC DTK
        INC   A
        DA    A
        MOV   DTK,A
        CJNE  A,#60H,RETURN_HW
        MOV   DTK,#0

        MOV   A,MNT         ;INC MNT
        INC   A
        DA    A
        MOV   MNT,A
        CJNE  A,#60H,RETURN_HW
        MOV   MNT,#0

        MOV   A,JAM;INC JAM
        INC   A
        DA    A
        MOV   JAM,A
        CJNE  A,#24H,RETURN_HW1
        MOV   JAM,#0

        MOV   A,TGL;INC MNT
        INC   A
        DA    A
        MOV   TGL,A
        CJNE  A,#30H,RETURN_HW
        MOV   TGL,#0

        MOV   A,BLN;INC MNT
        INC   A
        DA    A
        MOV   BLN,A
        CJNE  A,#12H,RETURN_HW
        MOV   BLN,#0

RETURN_HW:
        POP   ACC
        RET
;****************************************
delayyy:
        MOV   R7,#34
LP2A:   MOV   R6,#50
LP1A:   MOV   R5,#50
        DJNZ  R5,$
```

```
        DJNZ  R6,LP1A
        DJNZ  R7,LP2A
        RET

RETURN_HW1:

        MOV   A,address
        INC   A
        MOV   address,A
        ORL   WMCOM,#EEMEN
        ORL   WMCOM,#EEMWE
        MOV   DPTR,address
        MOV   A,suhu1c
        MOVX        @DPTR,A
        MOV   A,address
        INC   A
        MOV   address,A
        MOV   DPTR,address
        MOV   A,suhu1b
        MOVX        @DPTR,A
        MOV   A,address
        INC   A
        MOV   address,A
        MOV   DPTR,address
        MOV   A,suhu1a
        MOVX        @DPTR,A
        MOV   A,address
        INC   A
        MOV   address,A
        MOV   DPTR,address
        MOV   A,klb1c
        MOVX        @DPTR,A
        MOV   A,address
        INC   A
        MOV   address,A
        MOV   DPTR,address
        MOV   A,klb1b
        MOVX        @DPTR,A
        MOV   A,address
        INC   A
        MOV   address,A
        MOV   DPTR,address
        MOV   A,klb1a
        MOVX        @DPTR,A
        MOV   A,address
        INC   A
        MOV   address,A
        MOV   DPTR,address
```

```
        MOV  A,suhu2c
        MOVX        @DPTR,A
        MOV  A,address
        INC    A
        MOV  address,A
        MOV  DPTR,address
        MOV  A,suhu2b
        MOVX        @DPTR,A
        MOV  A,address
        INC    A
        MOV  address,A
        MOV  DPTR,address
        MOV  A,suhu2a
        MOVX        @DPTR,A
        MOV  A,address
        INC    A
        MOV  address,A
        MOV  DPTR,address
        MOV  A,klb2c
        MOVX        @DPTR,A
        MOV  A,address
        INC    A
        MOV  address,A
        MOV  DPTR,address
        MOV  A,klb2b
        MOVX        @DPTR,A
        MOV  A,address
        INC    A
        MOV  address,A
        MOV  DPTR,address
        MOV  A,klb2a
        MOVX        @DPTR,A
        POP    ACC
        RET

DSPJAM:
        MOV  DPTR,#PILIH1
        LCALL       PRINTSTRING1
        PUSH ACC
        MOV A,#8
        CALL POSISI1

        MOV  A,JAM
        SWAP A
        ANL    A,#0FH
        ORL    A,#030H
        CALL DATAOUT
        MOV  A,JAM
```

```
        ANL   A,#0FH
        ORL   A,#030H
        CALL DATAOUT
        MOV  A,#':'
        CALL DATAOUT

        MOV  A,MNT
        SWAP A
        ANL   A,#0FH
        ORL   A,#030H
        CALL DATAOUT
        MOV  A,MNT
        ANL   A,#0FH
        ORL   A,#030H
        CALL DATAOUT
        MOV  A,#':'
        CALL DATAOUT

        MOV  A,DTK
        SWAP A
        ANL   A,#0FH
        ORL   A,#030H
        CALL DATAOUT
        MOV  A,DTK
        ANL   A,#0FH
        ORL   A,#030H
        CALL DATAOUT
        POP   ACC
        RET


;*******************************
;* KUMPULAN RUTIN PELAYANAN LCD *
;*******************************

POSISI2_1:
        MOV   A,#1
POSISI2:
        ADD   A,#11000000B
        SJMP  POSISI_SUB

POSISI1_1:
        MOV   A,#1
POSISI1:
        ADD   A,#10000000B
POSISI_SUB:
        DEC   A
        LCALL CONTROLOUT
```

```
        RET

PRINTSTRING2:
        LCALL  POSISI2_1
        SJMP   PRINTSTRING

PRINTSTRING1:
        LCALL  POSISI1_1

PRINTSTRING:
        SJMP   OUTSTRING
PRINTSTRINGLOOP:
        LCALL  DATAOUT
        INC    DPTR

OUTSTRING:
        CLR    A
        MOVC   A,@A+DPTR
        JNZ    PRINTSTRINGLOOP
        RET

CONTROLOUT:
        CPL RS
        CPL E
        MOV P0,A
        CPL E
        CPL RS
        MOV P0,#0FFH
        SJMP LCD_OUT

DATAOUT:
        ;CPL RS
        CPL E
        MOV P0,A
        CPL E
        ;CPL RS

LCD_OUT:
        MOVX   @DPTR,A

DELAY_LCD:
        PUSH ACC
        MOV    A,#250
        DJNZ   ACC,$
        POP ACC
        RET
```

```
NAMA:      DB 'ROMEU RIVELINO  ',0
SEKOLAH:   DB 'NIM: 0017194    ',0
SETWAKTU:  DB 'MASUKKAN WAKTU  ',0
PILIH:     DB 'SET WAKTU :     ',0
PILIH1:    DB 'WAKTU:          ',0
PILIH2:    DB 'SENSOR1  SENSOR2',0
PILIH3:    DB '000|000  000|000',0
trans1:    DB '    PROSES      ',0
trans2:    DB ' TRANSFER DATA  ',0

        END
```