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

SKRIPSI

PERANCANGAN DAN PEMBUATAN ALAT PENAMPIL INFORMASI, JAM DAN KALENDER DIGITAL PADA TEMPAT UMUM BERBASIS MIKROKONTROLLER AT89S8252

Disusun Oleh :
ANDRIE DIAN RINALDY
00.17.078

MARET 2006

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN ALAT PENAMPIL INFORMASI, JAM DAN KALENDER DIGITAL PADA TEMPAT UMUM BERBASIS MIKROKONTROLLER AT89S8252

SKRIPSI

Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik Elektronika Strata Satu (S-1)

Disusun Oleh :

ANDRIE DIAN RINALDY
00.17.078

Mengetahui	Disetujui
Ketua Jurusan Teknik Elektro S-1	Dosen Pembimbing
(Ir. F. Yudi Limpraptono, MT)	(Ir. Yusuf Ismail Nakhoda, MT)
NIP.Y.1039500274	NIP.Y.1018800189

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2006

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSRTI
JURUSAN TEKNIK ELEKTRO S – 1
KONSENTRASI ELEKTRONIKA

BERITA ACARA UJIAN SKRIPSI

Nama : Andrie Dian Rinaldy

Nim : 00.17.078

Judul Skripsi : Perancangan dan Pembuatan Alat Penampil Informasi, Jam Dan Kalender Digital Pada Tempat Umum Berbasis Mikrokontroller AT89S8252

Dipertahankan dihadapan tim penguji skripsi jenjang Strata Satu (S-1) pada :

Hari : Kamis

Tanggal : 23 Maret 2006

Dengan Nilai : 85,4

Panitia Ujian Skripsi :

Ir. Mochtar Asroni, MSME	Ir.F. Yudi Limpraptono, MT
Ketua	Sekretaris

Anggota Penguji :

Ir.F. Yudi Limpraptono, MT	Ir. Eko Nurcahyo
Dosen Penguji I	Dosen Penguji II

LEMBAR PERSEMBAHAN

[illegible] Alamin segala puja dan puji penulis panjatkan kehadirat Allah SWT yang [illegible] kemudahan dalam penyusunan skripsi ini. Pada kesempatan ini penulis persembahkan skripsi ini kepada :

ALLAH SWT

"[illegible]ASIH YA ALLAH TELAH MELIMPAHKAN RAHMAT DAN HIDAYAH-NYA SEHINGGA PENULIS DAPAT MENUNTASKAN SKRIPSI INI"

Tak lupa juga ucapan terima kasih ananda persembahkan untuk :

"AYAHANDA IMAM SOEBEKTI DAN IBUNDA NINUK DIAH PURWANTINI"

Berkat do[illegible] dan restumu [illegible] ananda [illegible] tanggung jawab ananda. [illegible]natmu, kepercayaanmu [illegible] seorang sarjana [illegible]. Dengan [illegible] ananda menghaturkan [illegible] ayahanda dan ibunda [illegible].

Tak lupa juga kupersembahkan terima kasihku kepada kakakku tercinta :

"IVA DIAH LISTYANINGRUM"

Selamat telah diterima jadi pegawai negeri semoga apa yang telah kita capai dapat membahagiakan kedua orang tua kita

Dan tak lupa kupersembahkan skripsi ini..........

For my Soulmate :

"ANGGIND FRESHDIKA TANJUNG KENCANA"

Makasih ya yank buat segalanya baik itu dukungan moril maupun materinya. sorry banget aq selalu ngerepotin kamu, ngecewain kamu 'n bikin marah kamu. Aq cuman pengen bilang.........."I LOVE U NEVER............"

Buat Mertua-q :

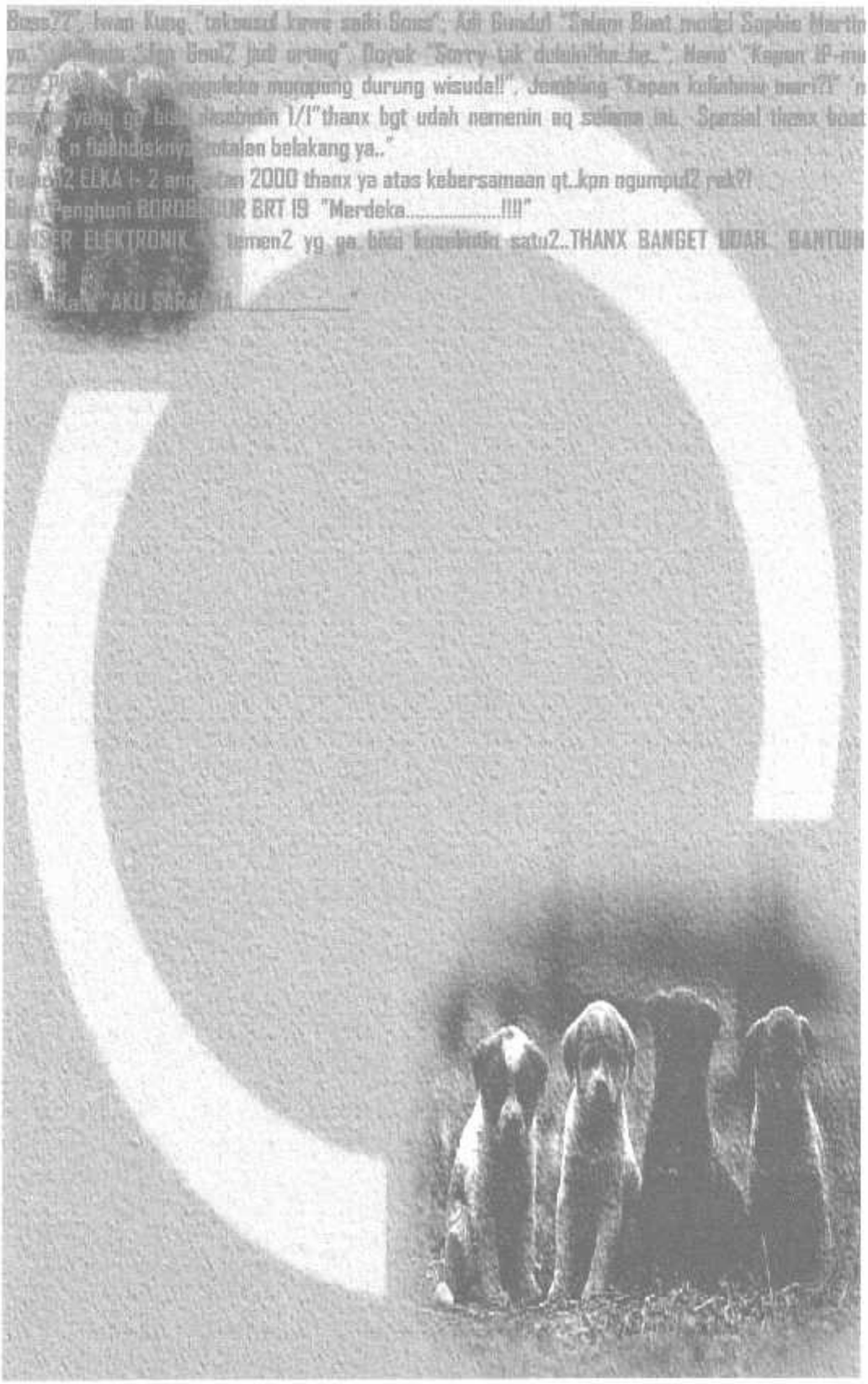
"[illegible]NDA ANGGRA"

Makasih banget buat semuanya, ucap[illegible] cukup untuk membalas kebaikan [illegible]nda [illegible] menjaga dengan baik putri anda selam[illegible]

Thanx to All My Friend's :

Sohibq (Govin "BANJAR" thanx buat [illegible] wisuda ya, jangan [illegible] dalam masalah jodoh, pastiin satu [illegible] wis sarjana [illegible] bareng trus nikah bareng..., Irawan "LEMOT" [illegible] trus [illegible] sojomu [illegible]!!!!!! [illegible] "BAKRIE" thanx bgt buat semangatnya, ingat [illegible] anda..ndang rabio [illegible] Candra "Kak Cen" ngenteni opo wis sarjana [illegible] kancane ndang2 [illegible] kuras awakmu..!!, Rani "Imutz" thanx bgt buat [illegible]nya.....Kadek "yang tegas [illegible] orang"

Buat penghuni Gentong House "Cas "capetan [illegible]" Memble "Semangat Mble ojo mbojo thok neil", Oppie [illegible]



ABSTRAKSI

PERANCANGAN DAN PEMBUATAN ALAT PENAMPIL INFORMASI, JAM DAN KALENDER DIGITAL PADA TEMPAT UMUM BERBASIS MIKROKONTROLLER AT89S8252

(Andrie Dian Rinaldy, 00.17.078, Jurusan Teknik Elektro S-1/Elektronika)
(Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT)

Kata Kunci : Mikrokontroller AT89S8252, *Keyboard*, *Seven Segment*, *Dot Matrik*.

Kemajuan teknologi mendorong berkembangnya banyak teknologi baru terutama di bidang informasi. Salah satunya adalah alat yang berfungsi untuik menampilkan informasi yang dapat dinikmati banyak orang, khususnya di tempat umum. Seperti halnya alat ini dibuat khusus untuk ditempatkan pada tempat keramaian seperti pasar, alun-alun, atau bahkan mungkin di tempat pertokoan. Fokus pembahasan alat ini adalah untuk men-*display*-kan informasi, jam dan kalender digital dimana informasi tersebut dapat berubah sesuai dengan keinginan kita dengan menggunakan *keyboard*. Prinsip kerja alat ini adalah pada saat *keyboard* ditekan maka data tersebut akan diubah dalam bentuk sinyal yang akan dikirim oleh *transmitter* ke *display*. Sinyal diterima oleh *receiver* kemudian diolah oleh mikrokontroller dalam hal ini adalah mikrokontroller AT89S8252 dan diubah dalam bentuk karakter huruf atau angka. Sedangkan jam berubah sendiri sesuai dengan memori yang tersimpan dalam mikrokontroller AT89S8252. Data yang diubah tersebut akan ditampilkan dalam bentuk huruf atau angka melalui *display seven segment* dan *dot matrik*.

KATA PENGANTAR

Atas Berkat Rahmat Allah Yang Maha Kuasa, schingga penulis dapat menyelesaikan laporan Skripsi dengan judul :

"PERANCANGAN DAN PEMBUATAN ALAT PENAMPIL INFORMASI, JAM DAN KALENDER DIGITAL PADA TEMPAT UMUM BERBASIS MIKROKONTROLLER AT89S8252"

Pembuatan Skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata-1 di Institut Teknologi Nasional Malang. Laporan Skripsi ini merupakan tanggung jawab tertulis atas ilmu pengetahuan yang didapat selama penyusun mengikuti kuliah.

Atas terselesaikannya Skripsi ini, penulis mengucapkan terima kasih kepada :

- Bapak Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
- Bapak Ir. Mochtar Asroni, MSME, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang
- Bapak Ir.F.Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro S-1.
- Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Dosen Pembimbing yang telah memberikan bimbingan, pengarahan, serta ilmu-ilmu yang sangat berharga sehingga Skripsi ini dapat terselesaikan.
- Ibu Ir. Mimien Mustikawati, selaku Sekretaris Jurusan Teknik Elektro S1.
- Serta pihak-pihak yang tidak dapat disebutkan satu persatu yang telah banyak membantu dalam pengerjaan skripsi ini sehingga dapat terselesaikan.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohan maaf kepada semua pihak bilamana selama penyusunan Skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga Skripsi ini dapat bermanfaat bagi kita semua.

Malang, Maret 2006

Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Gambar **Halaman**

DAFTAR TABEL

Tabel **Halaman**

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada masa ini, perkembangan teknologi maju sangat pesat, terutama pada teknologi elektronika. Penerapan teknologi elektronika sudah merambah ke segala macam bidang. Hampir semua peralatan sudah memanfaatkan teknologi elektronika sebagai kontrol atau kendali.

Perkembangan teknologi elektronika saat ini banyak dipengaruhi oleh keberadaan mikroprosesor/mikrokontroller, yaitu suatu istilah yang umum digunakan untuk menggambarkan komputer/ atau program yang mampu berpikir dan memecahkan masalah seperti yang dilakukan manusia.

Melalui berbagai keunggulan mikrokontroller sebagai sistem pengontrol di dalam pengendalian, penyusun mencoba mengaplikasikannya pada pembuatan alat penampil data dan penunjuk waktu. Saat ini banyak terdapat alat penampil data yang bagus. Di sini penyusun mencoba memilih dot matrik dan *seven segment* sebagai penampil data dan penunjuk waktu, dengan pengendali mikrokontroller AT89S8252.

Display dot matrik dan *seven segment* sangat baik untuk dijadikan papan reklame/pengumuman dan penunjuk waktu, dimana dengan menggunakan *display* dot matrik dan *seven segment*, data tampilan yang diinginkan dapat berubah-ubah sesuai dengan keinginan penggunanya. Akan tetapi bagi *display* yang besar dan

ditempatkan di tempat yang lumayan tinggi akan memiliki masalah besar ketika tampilan yang sudah ada harus segera diganti dengan tampilan yang lain.

Hal ini membutuhkan banyak energi untuk proses penggantiannya. Proses penggantian dilakukan dengan alat bantu baik itu berupa tangga kemudian melepaskan mikrokontroller atau yang lainnya karena harus me-*reset* tampilan yang sudah ada sebelumnya. Selanjutnya rangkaian mikrokontroller dimasukkan oleh yang baru lalu dikendalikan pada tempatnya. Disamping memiliki faktor kecelakaan yang cukup tinggi, pengendalioan rangkaian itu membutuhkan energi yang besar.

Jika proses penggantian dilakukan dengan menggunakan remote kontrol, maka pengeluaran energi tidak akan terjadi. Jadi perubahan tampilan *display* dot matrik dan *seven segment* cukup menggunakan remote ketika mengganti isi tampilan *display* dot matrik dan *seven segment* dari isi sebelumnya.

1.2. Batasan Masalah

Sesuai dengan permasalahan yang difokuskan pada perencanaan dan pembuatan alat penampil informasi, jam dan kalender digital pada tempat umum berbasis mikrokontroller AT89S8252, maka diberikan batasan masalah sebagai berikut :

- Tidak membahas masalah *power supply*.
- *Keyboard* dibahas secara umum
- Tidak melakukan analisa bentuk gelombang dari *keyboard*.
- Menggunakan mikrokontroller AT89S8252 sebagai sistem kendali.

1.3. Tujuan Penulisan

Tujuan dari perencanaan dan pembuatan alat penampil informasi, jam dan kalender digital pada tempat umum berbasis mikrokontroller AT89S8252 ini adalah untuk mempermudah penggantian tampilan dari *display* dot matrik dan *seven segment* yang sudah ada sesuai dengan keinginan pengguna.

1.4. Rumusan Masalah

Dalam perencanaan dan pembuatan alat penampil informasi, jam dan kalender digital pada tempat umum berbasis mikrokontroller AT89S8252 dapat dirumuskan beberapa masalah yang akan dibahas yaitu :

1. Bagaimana merancang dan membuat alat ini agar dapat memberikan informasi, jam dan kalender digital pada tempat umum secara tepat dan praktis..
2. Bagaimana merancang dan membuat *software* agar mikrokontroller AT89S8252 dapat dioperasikan sebagai sistem kendali.

1.5. Metodologi Penulisan

Untuk mencapai sasaran yang diinginkan dan sesuai dengan tujuan penyusuanan, maka digunakan metode-metode sebagai berikut :

1. Studi literatur

   Metoda yang digunakan diawali dengan melakukan studi literatur terhadap data – data yang diperlukan guna menunjang kelancaran di dalam proses pembuatan alat. Studi ini akan dilaksanakan dengan cara mencari buku – buku

yang menjadi landasan dari tiap- tiap teori maupun mencari data lewat sarana internet. Kemudian data ini akan diproses untuk mencari yang sesuai dengan data yang kita kehendaki.

2. Perencanaan dan Pembuatan Alat

   Melaksanakan perencanaan dan pembuatan alat baik *hardware* maupun *software* sesuai dengan rancangan yang disusun.

3. Pengujian Alat

   Pengujian alat dilakukan saat mencapai tahap akhir untuk menemukan kesalahan atau kekurangan pada alat tersebut untuk kemudian dilakukan perbaikan.

4. Penulisan laporan

   Bertujuan untuk penyusunan data laporan berpedoman pada alat yang selesai dibuat beserta cara kerja alat.

1.6 Sistematika Penulisan

Sistematika penulisan dari tugas akhir ini adalah sebagai berikut :

- BAB I : Pendahuluan

  Membahas tentang latar belakang permasalahan, tujuan, ruang lingkup metoda yang digunakan, serta susunan penulisan dari buku tugas akhir ini.

- BAB II : Teori Penunjang

  Bab ini berisikan teori-teori penunjang yang digunakan dalam perencanaan dan pembuatan tugas akhir.

- BAB III : Perencanaan Sistem

  Pada bab ini akan di bahas perencanaan dan realisasi keras serta perangkat lunak yang digunakan.

- BAB IV : Pengujian Sistem

  Bab ini berisi tentang pengujian sistem dan hasil berupa analisa data.

- BAB V : Kesimpulan

  Berisikan kesimpulan yang diperoleh dari perencanaan, realisasi, dan pengetesan yang dilakukan.

BAB II

TEORI PENUNJANG

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

2.1 Mikrokontroler AT89S8252

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

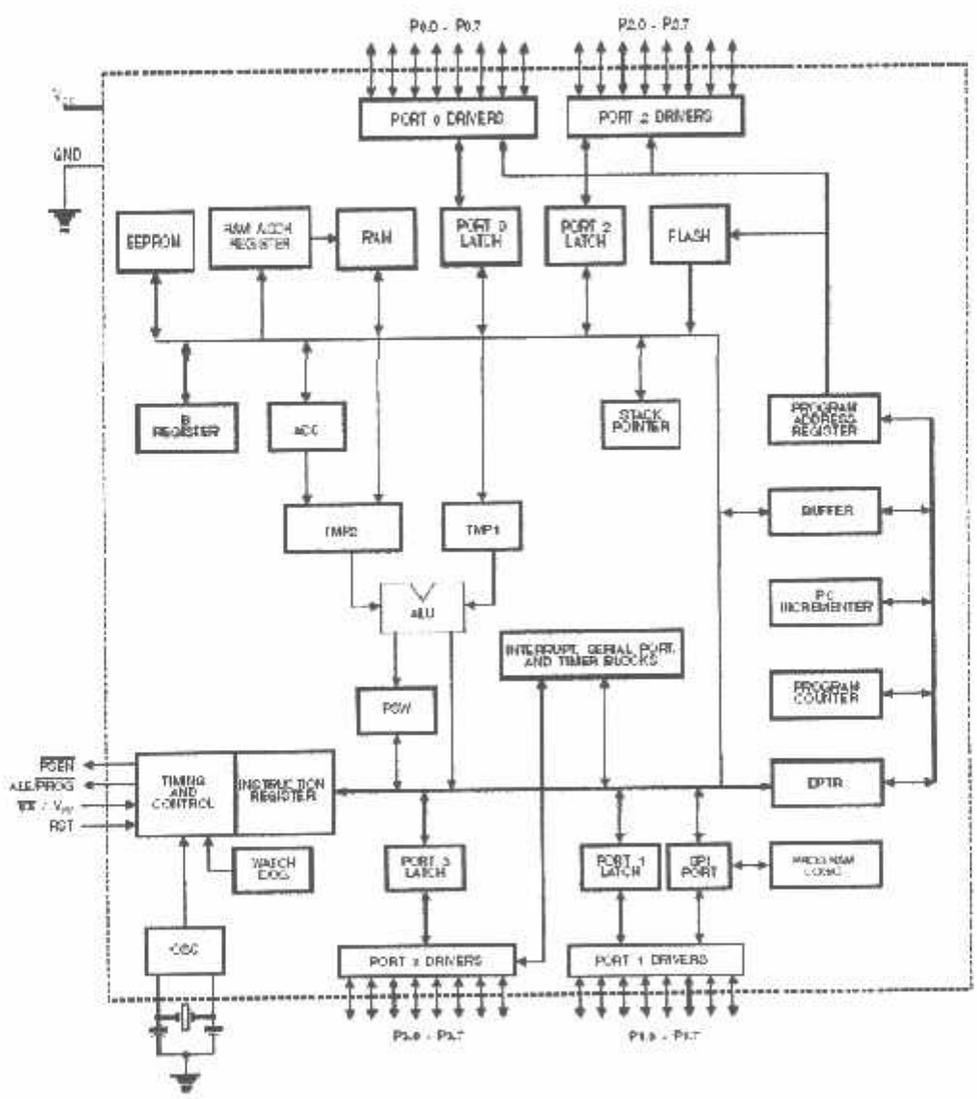
Pada dasarnya mikrokontroler mempunyai fungsi yang sama dengan mikroprosesor yaitu untuk mengontrol suatu kerja sistem. Selain itu mikrokontroler juga dikemas dalam satu *chip* (*single chip*). Didalam mikrokontroler juga terdapat CPU, ALU, PC, SP, dan *register* seperti dalam mikroprosesor, tetapi juga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

kontrol mikrokontroler bila dibandingkan dengan mikroprosesor memiliki kemampuan dan segi ekonomis yang bisa diandalkan karena dalam mikrokontroler sudah terdapat RAM dan ROM. Sedangkan mikroprosesor didalamnya tidak terdapat keduanya. Terlihat bahwa mikrokontroler Atmel AT89S8252 memilki banyak fitur yang menguntungkan. Dipakainya *Downloadable flash memory* memungkinkan mikrokontroler ini bekerja sendiri tanpa diperlukan tambahan *chip* lainnya. Sementara *Flash memory-nya* mampu diprogram hingga seribu kali. Hal lain yang menguntungkan adalah sistem pemrograman menjadi lebih sederhana dan tidak memerlukan rangkaian yang rumit seperti rangkaian untuk memprogram produk Atmel lainnya. Secara umum konfigurasi yang dimiliki mikrokontroler AT89S8252 adalah sebagai berikut : *(Atmel, 1997:4-105)*

- Sebuah CPU 8 bit dengan menggunakan teknologi dari Atmel.
- 8K byte *Downloadable Flash Memory*.
- 2K byte EEPROM
- Sebuah *port* serial dengan kontrol *full duplex* UART (*Universal Asynchronous Receiver Transmiter*).
- 256 byte RAM *internal*.
- 32 I/O yang dapat dipakai semua.
- 3 buah *Timer/Counter* 16 bit .
- SPI Serial *Interface*.
- *Programmable Watchdog Timer* .
- *Dual Data Pointer*.

- Frekuensi kerja 0 sampai 24 MHz
- Tegangan operasi 2,7 Volt sampai 6Volt.
- Kemampuan melaksanakan operasi perkalian, pembagian, dan operasi *Bolean* (bit)

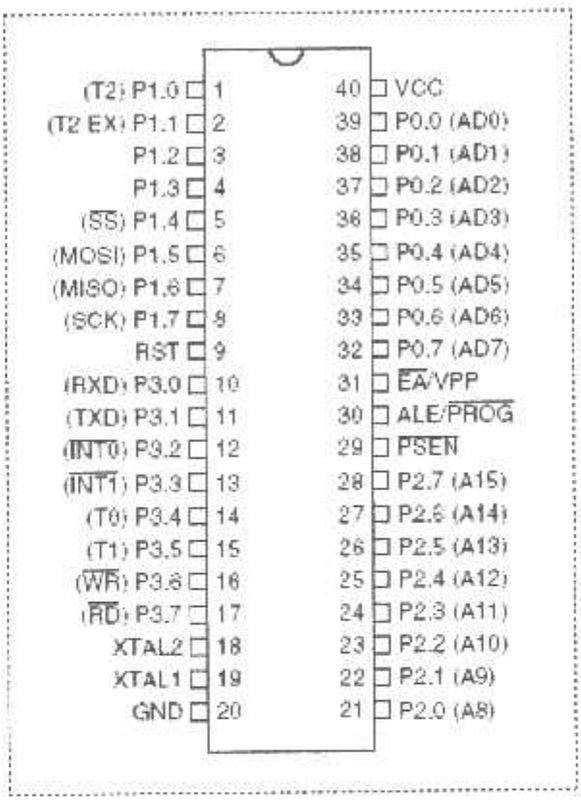
Sedangkan untuk blok diagram AT89S8252 diperlihatkan dalam gambar 2-1



Gambar 2-1 Blok Diagram MikrokontrollerAT89S8252[1]

2.1.1. Penjelasan Fungsi Pin AT89S8252

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



Gambar 2.2 Susunan Pin Mikrokontroller AT89S8252[11]

- Pin 1 sampai 8

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

- Pin 9

  RST merupakan saluran dua masukan untuk me-*reset* mikrokontroler dengan cara memberi masukan logika tinggi.

- Pin 10 sampai 17

  Port 3 yang terdiri atas pin 10 sampai pin 17 merupakan saluran masukan/keluaran dua arah dan mempunyai fungsi khusus seperti yang terlihat dalam tabel 2-1

- Pin 18 dan 19

  $XTAL_1$ dan $XTAL_2$ merupakan saluran untuk mengatur pewaktuan sistem. Untuk pewaktuan dapat menggunakan pewaktuan internal maupun eksternal .

- Pin 20

  Vss merupakan hubungan ke *ground* dari rangkaian.

- Pin 21 sampai 28

  Port 2 yang terdiri atas pin 21 sampai 28 merupakan saluran masukan/keluaran dua arah. *Port* ini mengeluarkan 8 bit bagian alamat tinggi (A8-A15) selama pengambilan instruksi dari memori program eksternal dan pengambilan data memori eksternal yang menggunakan mode pengalamatan 16 bit.

- Pin 29

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

- Pin 30

  ALE/PROG (*Address Latch Enable*) merupakan pulsa yang berfungsi untuk menahan alamat rendah (A0-A7) dalam *port* 0, selama proses baca/tulis memori eksternal. Frekuensi ALE adalah 1/6 kali frekuensi

osilator, dan digunakan sebagai pewaktu. Pin ini juga berfungsi sebagai saluran program selama dilakukan pemrograman jika menggunakan memori program eksternal.

- Pin 31

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

- Pin 32 sampai 39

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

- Pin 40

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

2.1.2. Masukan dan Keluaran

Untuk saluran dan keluaran terdapat 4 buah *port* yang masing-masing 8 bit. Saluran ini bersifat dua arah (*bidirectional*) yang berarti dapat difungsikan sebagai masukan atau keluaran, serta dapat dialamati per bit. *Port* 3 selain digunakan sebagai *port* masukan dan keluaran juga dapat digunakan sebagai fungsi pengganti sebagaimana yang terdapat dalam tabel 2-1. Sedangkan

AT89S8252 memiliki fitur tambahan yang terdapat pada *port 1* seperti pada tabel 2-2.

Tabel 2-1 Fungsi Pengganti *Port 3*[1]

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)

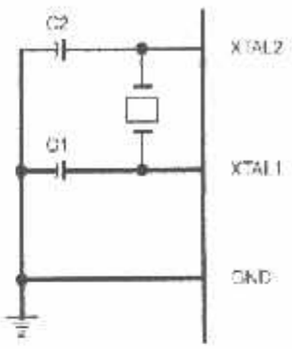
Tabel 2-2 Fungsi khusus pada *port 1* AT89S8252[1]

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)

2.1.3. Osilator

Jantung dari AT89S8252 adalah rangkaian yang membangkitkan pulsa *clock* yang mesinkronkan semua operasi internal. Mikrokontroler AT89S8252 memiliki osilator internal (*on chip oscillator*) yang dapat digunakan sebagai

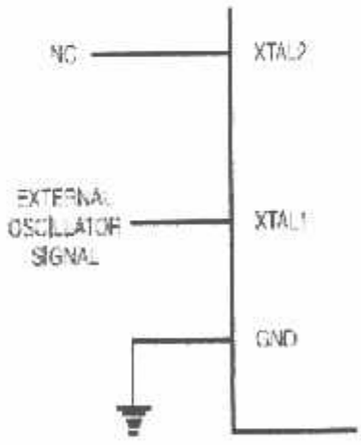
sumber pewaktu (*clock*) bagi CPU. Untuk menggunakan internal diperlukan sebuah kristal atau resonator keramik antara pin XTAL1 dan pin XTAL2 dan sebuah kapasitor ke *ground*. Konfigurasinya dapat dilihat pada gambar berikut.



Gambar 2-3 Konfigurasi Osilator Menggunakan Kristal[11]

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

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

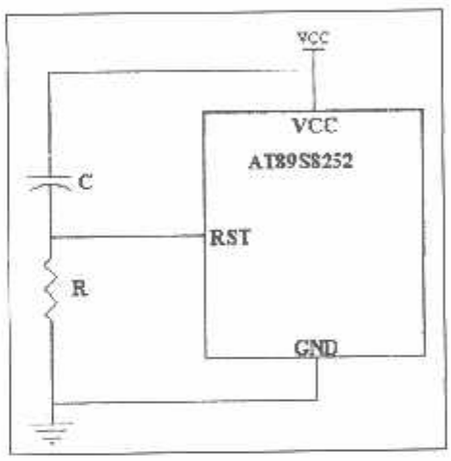


Gambar 2-4 Konfigurasi Osilator Menggunakan *External Oscilator Signal*[11]

2.1.4. *Reset*

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

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



Gambar 2-5 Rangkaian *Power On Reset*[11]

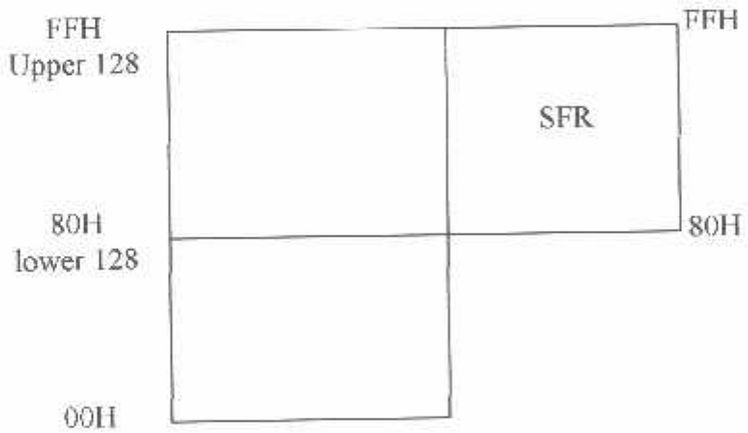
2.1.5. Data Memori (EEPROM) Dan RAM

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

EEPROM *on-chip* ini diakses dengan mengeset bit EEMEN pada *register* WMCON pada alamat 96H. Alamat EEPROM ini adalah 000H sampai 7FFH. Intruksi *move* digunakan untuk mengakses EEPROM internal ini. Bit EEMWE

pada *register* WMCON harus diset ke-1 sebelum seberang lokasi pada EEPROM dapat ditulisi. Program pengguna harus *me-reset* bit EEMWE ke '0'jika proses penulisan ke EEPROM tidak diperlukan lagi. Proses penulisan ke EEPROM dapat dilihat dengan membaca bit RDY/BSY pada SFR WMCON. Jika bit ini berlogika rendah maka berarti penulisan EEPROM sedang berlangsung, jika bit ini berlogika tinggi berarti penulisan sudah selesai dan penulisan lain dapat dimulai lagi.

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



Gambar 2-6 Memori Data Eksternal[11]

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

1. *Register* bank 0 – 3

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

2. Bit *Addressing*

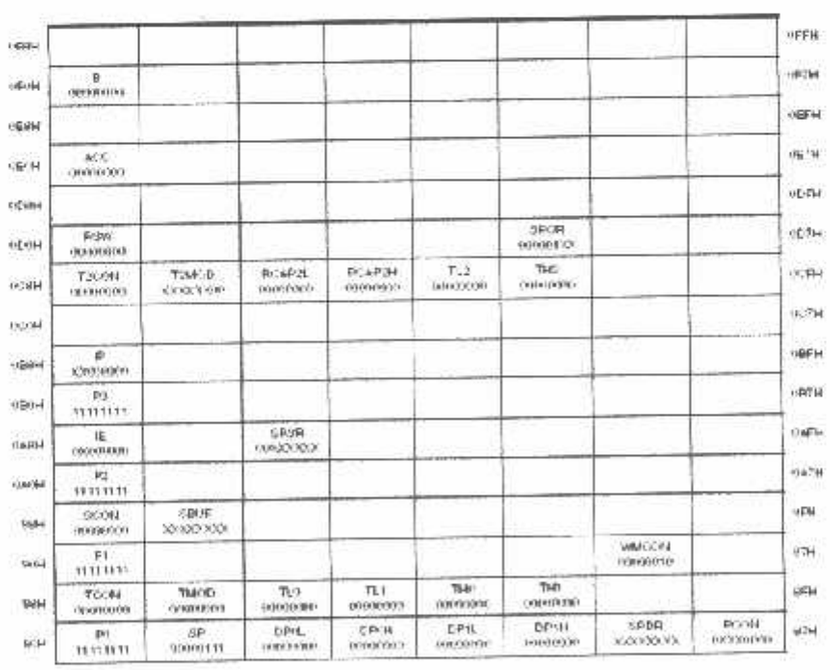
Terdiri dari 16 *bytes* yang dimulai dari 20H – 2FH. Masing-masing dari 128 bit lokasi ini dapat dialamati secara langsung yaitu dari 00H sampai 7FH.

3. *Scratch Pad Area*

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

2.1.6. *Special Function Register* (SFR)

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



Gambar 2-7 AT89S8252 SFR Map dan Reset Value[1]

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

- SFR untuk Timer 2

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

- SFR untuk *Watchdog* Memori.

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

- SFR pengontrol SPI

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

a. *Full Duplex*, 3 kawat dengan *transfer* data secara sinkron .

b. Operasi *Master* atau *Slave*.

c. Frekuensi maksimum 6 MHz.

d. 4 bit rate terprogram.

2.1.7. *Timer* dan *Counter*

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

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

Tabel 2-3 Mode Operasi *Timer/Counter* 0 dan 1[11]

Mode	*Timer/Counter* 0	*Timer/Counter* 1
0	13 bit *Timer*	13 bit *Timer*
1	16 bit *Timer*	16 bit *Timer*
2	8 bit *auto-reload*	8 bit *auto-reload*
3	28 bit *timer*	Tidak bekerja

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

Tabel 2-4 Mode Operasi *Timer* 2

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)

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

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

CP/RL2 = Pemilihan *capture/Reload*. Jika diset maka proses *capture* yang terjadi sedangkan jika bit ini di-*clear* maka proses *reload*.

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

2.1.8. *Idle Mode*

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

2.1.9. Sistem Interupt

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

Saat terjadinya *interrupt*, mikrokontroler secara otomatis akan menuju ke *sub routin* pada alamat tersebut. Setelah *interrupt service* selesai dikerjakan, mikrokontroler akan mengerjakan program semula. Dua sumber *interrupt external*

adalah INT0 dan INT1, dimana kedua interupsi *eksternal* akan aktif atau aktif transisi tergantung isi dari IT0 dan IT1 pada register TCON. Interupsi T0, T1, T2 aktif pada saat *timer* yang sesuai mengalami *roll over*, interupsi serial dibangkitkan dengan melakukan operasi OR pada R1 dan T1. Tiap-tiap sumber interupsi dapat *enable* atau *disable* secara otomatis.

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

Tabel 2-5 Alamat Sumber Interupsi[11]

(MSB)(LSB)

EA	–	ET2	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

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

2.2 Keyboard

2.2.1. Tata Kerja *Keyboard*

Setiap kali salah satu tombol *keyboard* ditekan atau dilepas, *keyboard* akan mengirim kode ke *host* (*host* adalah komputer kalau dihubungkan ke *PC*, atau

berupa mikrokontroller kalau *keyboard* dihubungkan ke peralatan berbasis mikrokontroller).

Komunikasi antar *keyboard* dan *host* adalah komunikasi dua arah, *keyboard* mengirim *scan code* ke *host*, *host* bisa mengirim perintah untuk mengatur kerja dari *keyboard*. Kode perintah untuk *keyboard* tidak sebanyak *scan code*, berikut ini daftar perintah untuk *keyboard* (dalam heksadesimal) adalah:

- **ED**, perintah untuk menyalakan lampu indikator di *keyboard*, setelah menerima perintah **ED** dari *host*, *keyboard* akan menjawab dengan **FA** sebagai tanda perintah itu telah dikenali (**ACK** – *acknowledge*) dan menunggu 1 byte perintah lagi dari *host* untuk menentukan lampu indikator mana yang perlu di-nyala/padam-kan. 1 byte perintah susulan tersebut akan diartikan sebagai berikut : bit 0 dipakai untuk mengatur lampu indikator *Scrool Lock*, bit 1 untuk *Num Lock* dan bit 2 untuk *Caps* Lock, bit-bit lainnya diabaikan.
- **EE**, dipakai *host* untuk memeriksa apakah *keyboard* masih aktif. Setelah menerima perintah **EE** dari *host*, *keyboard* akan menjawab dengan **EE** pula, menandakan dirinya masih aktif.
- **F0,** ada *keyboard* yang dilengkapi 3 set *scan code*, perintah ini dipakai untuk memilih *scan code* yang ingin dipakai. Setelah menerima perintah **F0** dari *host*, *keyboard* akan menjawab dengan **FA** sebagai tanda perintah itu telah dikenali (**ACK** – *acknowledge*) dan *host* menjawab 1 byte lagi (nilainya 1,, 2, atau 3) untuk memilih set *scan code*. Jika bit yang dikirimkan nilainya 0, *keyboard* akan menjawab dengan nomor set *scan code* yang saat itu dipakai.

- **F3**, dipakai untuk mengatur kecepatan tanggapan *keyboard* (*Typematic Repeat Rate*), setelah menerima perintah **F3** dari *host*, *keyboard* akan menjawab dengan **FA** sebagai tanda perintah itu telah dikenali (**ACK** – *acknowledge*) dan *host* menjawab 1 byte nilai kecepatan tanggapan *keyboard* yang dikehendaki.
- **F4**, dipakai untuk mengaktifkan kembali *keyboard*, setelah menerima perintah ini *keyboard* akan menjawab dengan **FA** (**ACK** – *acknowledge*).
- **F5**, dipakai untuk menonaktifkan *keyboard*, setelah menerima perintah ini *keyboard* akan menjawab dengan **FA** (**ACK** – *acknowledge*).
- **FE**, dipakai meminta *keyboard* mengirim ulang *scan code* terakhir yang dikirim.
- **FF**, perintah untuk me-*reset keyboard*

Selain perintah dari *host*, keyboard juga mempunyai kode-kode lain selain *scan code* yang dikirimkan ke , *sebagai berikut :*

- **FA**, berarti **ACK** (*acknowledge*), yaitu jawaban dari *keyboard* bahwa perintah dari *host* sudah dikenali dengan baik.
- **AA**, berarti *keyboard* selesai memeriksa diri dan siap bekerja setelah diberi catu daya.
- **EE**, identik dengan perintah **EE diatas.**
- **FE**, artinya minta *host* mengulang perintah terakhir yang dikirim.
- **FF / 00**, berarti terjadi kesalahan di *keyboard.*

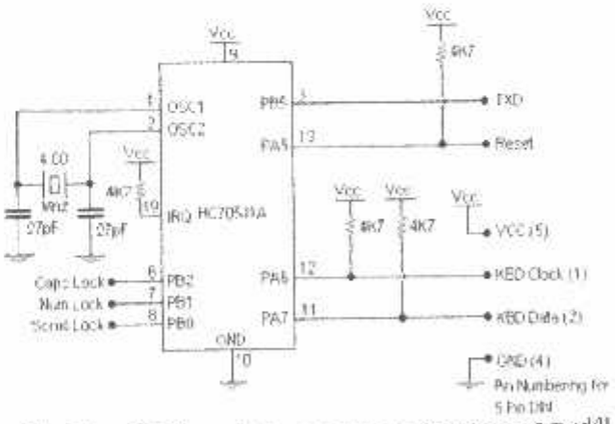
Gambar 2.8 menjelaskan *scan mode* masing-masing tombol *keyboard* PC. Terlihat pada gambar tersebut, *scan mode* tidak berupa mode ASCII yang biasa

dipakai memakai huruf, dan ditentukan secara acak juga. Sehingga setelah diterima *host, scan code* harus dirubah menjadi kode ASCII dengan memakai cara "penerimaan tabel".

Gambar 2.8 *Keyboard* PC dan *Scan Code*[4]

2.2.2. Rangkaian Penghubung

Keyboard PC dibangun dengan mikrokontoler MCS48 yang merupakan saudara tua MCS51 tapi jauh lebih sederhana. Untuk keperluan membentuk rangkaian penghubung tidak perlu diketahui bagaimana cara kerja mikrokontroler dalam *keyboard*, tapi cukup meninjau rangkaian pada gambar 2.9 .



Gambar 2.9 Penghubung dalam *Keyboard PC*[4]

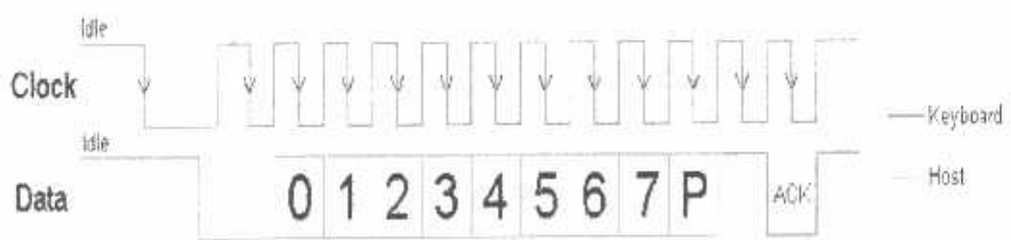
Kbd *Clock* dibangkitkan oleh MCS48, merupakan sinyal pendorong **Kbd Data** yang bisa bersumber dari *keyboard* maupun bersumber dari PC. Level

tegangan pada kedua sinyal ini memenuhi standart sinyal TTL biasa, jadi bisa langsung dihubungkan mikrokontroller. Sumber data dari *keyboard* dicatu dari luar, harus diperhatikan kebutuhan arusnya cukup besar bisa sampai 300 mA.

2.2.3. sinyal pengiriman data dari *Keyboard*

Saat tidak ada pengiriman data, sinyal Kbd *clock* dan Kbd Data dalam keadaan '1'. Sinyal pengiriman data dari *keyboard* dalam gambar 2.10 dijelaskan sebagai berikut :

- Data mulai dikirimkan dengan me-nol-kan **Kbd Data** sebagai data mulai pengiriman (*start bit*), berapa saat kemudian setelah **Kbd Data** stabil disusul **Kbd *Clock*** berubah menjadi '0' dan kembali '1' lagi, ini berarti selesai mengirimkan data 1 bit.
- Setelah mengirim *'start bit'*, dikirimkan bit 0, bit 1 dan seterusnya sampai bit 7.
- Menyusul dikirim *'parity bit'*, yaitu bit kontrol yang berguna bagi *host* penerima data untuk memastikan data yang diterima tidak ada kesalahan. Jika banyaknya bit '1' yang terdapat di bit 0 sampai bit 7 ganjil, *'parity bit'* akan bernilai '1'.
- Sebagai penutup (*stop bit*) **Kbd Data** dikembalikan ke keadaan normalnya, yaitu '1'.



Gambar 2.10 Sinyal Komunikasi Data Seri *Keyboard*[41]

2.2.4. Komunikasi Serial

Pengiriman data serial adalah pengiriman data 1 bit pada suatu saat tertentu, sehingga untuk pengiriman data 1 *byte* dibutuhkan suatu urutan. Frekuensi transmisi di dalam sistem komunikasi serial disebut dengan *Baud-rate*. *Baud-Rate* didefinisikan sebagai jumlah bit yang dtransmisikan tiap detik. *Typical Baud-Rate* adalah : 50, 75, 110, 150, 300, 1200, 2400, 4800, dan 9600. Satuan *Baud-Rate* adalah Bps (*Bit per Second*).

Komunikasi data serial dibagi menjadi dua :

1. Komunikasi Sinkron USRT (*Universal Synchronous Receiver Transmitter*)
2. Komunikasi Asinkron UART (*Universal Asynchronous Receiver Transmitter*)

Disini hanya akan membahas komunikasi sinkron sesuai dengan serial *port* yang dimiliki MCS 48.

- START BIT

  Data yang dtransmisikan secara serial harus dapat diterima kemudian diinterpretasikan kembali menjadi data 1 *byte*. Data yang ditransmisikan secara serial selalu diawali dengan *start bit* pada aliran data tersebut. Fungsi dari ini untuk mengetahui data baru yang diterima. Pada saat saluran transmisi tidak mengirimkan informasi, hal ini disebut dengan *marking*, atau sering disebut juga dengan *Idle State* (saat nganggur). Jika kondisi *marking* dari saluran berada pada logika 1, maka *start bit* yang ditambahkan pada aliran data tersebut adalah merupakan kebalikan dari kondisi *marking*, dalam hal ini start bit berlogika 0. *Start bit* ditambahkan pada awal aliran data dengan panjang 1 bit.

- STOP BIT

Bit terakhir yang ditambahkan pada aliran data adalah *stop bit*. *Receiver* akan menerima *stop bit* ini untuk mendeteksi pada akhir aliran bit. Ada beberapa jenis stop bit :1, 1.5, dan 2 satuan bps.

Mode Transmisi Serial

Ada 3 mode transmisi serial antara lain :

1. *Simplex serial connection*
2. *Half Duplex serial connection*
3. *Full Duplex serial connection*

Dalam hal ini mode transmisi yang digunakan adalah *Simplex serial connection* dimana mode *Simplex serial connection* dapat dilihat pada gambar 2.11

Gambar 2.11 Mode Transmisi *Simplex Serial Connection*

2.3 LED Sebagai Dot Matrik

Tampilan matrik titik (Dot Matrik) terdiri dari sejumlah LED (*Light Emitting Diode*) yang disusun secara baris dan kolom. Susunan yang paling sering digunakan adalah matrik 8 x 8, yaitu delapan kolom dan delapan baris. Susunan X kali Y ini memungkinkan pengamatan X dan Y, susunan LED di dalam dot matrik tersebut dapat digunakan sebagai penampil karakter yang diinginkan.

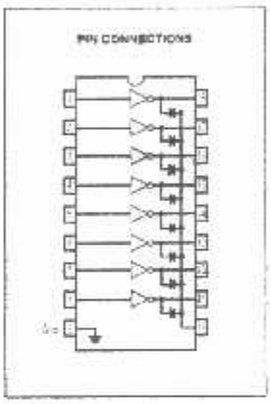
Matrik 8 x 8 dapat digunakan untuk menyajikan karakter *alfanumerik* yang lengkap. Proses pembangkitan karakter anatara lain melibatkan proses *scanning* baris atau kolom, proses *scanning* tersebut akan memilih susunan (baris atau kolom) LED yang tepat untuk dinyalakan atau dipadamkan oleh *driver* dot matrik. Proses ini diulang untuk baris atau kolom berikutnya dipilih dengan urutan tertentu, proses diatas diulang mulai baris paling atas (atau kolom pertama) hingga terakhir yang berulang terus dengan frekuensi yang sama. Frekuensi *driving* LED diatas frekuensi kemampuan mata untuk membedakan padam dan nyala sehingga proses *scanning* dot matrik tidak terlihat. Rangkaian skema dot matrik 8x8 ditunjukkan dalam Gambar 2.11

Gambar 2.12 Rangkaian Dot Matrik 8 x 8[15]

Jika frekuensi *scanning* diatas kemampuan mata manusia untuk melihat kedipan tersebut akan diperoleh bebas kedip. Jika matrik di *scan* dari kiri ke kanan kolom demi kolom disebut vertikal *scanning*.

2.4 *Driver* LED (IC ULN2803)

ULN2803 sebagai penggerak perangkat LED dalam dot matrik merupakan rangkaian transistor yang disusun *Darlington* 7-bit input-output. Input dari ULN2803 kompatibel dengan TTL dan +5v CMOS. Setiap saluran dari 7-bit output mampu melewatkan arus maksimal 500 mA. Dengan karakteristik yang dimiliki, ULN2803 akan dirangkai sebagai penggerak perangkat LED dalam dot matrik (*driver*). Dalam Gambar 2.12 diperlihatkan diagram ULN2803.

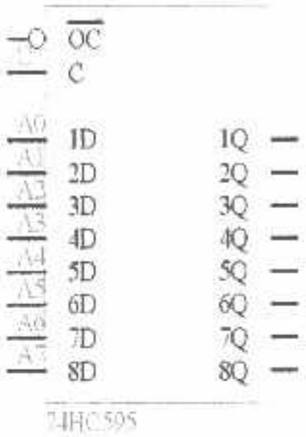


Gambar 2.13 Pin Koneksi ULN 2803[5]

2.5 Register Geser (74HC595)

IC 74HC595 terdiri atas enam belas buah *flip-flop* tipe D. Register geser ini disusun dari enam belas buah *flip-flop* tipe D. Register ini disebut register geser 8-bit karena mempunyai delapan tempat untuk menyimpan data Qa sampai Qd. Dari ke-enam belas *flip-flop* tersebut delapan untuk data serial dan delapan untuk data paralel. Selain enam belas *flip-flop* juga terdapat delapan buah *tristate* yang jika diilustrasikan berfungsi sebagai saklar, yaitu apabila kontrol aktif, keluaran akan sama dengan masukan, sedangkan bila kontrol dalam kondisi tidak

aktif, maka keluaran akan berada dalam kondisi *Hi-Z*. Rangkaian *internal* 74HC595 ditunjukkan dalam Gambar 2.13



Gambar 2.14 Rangkaian Internal IC 74HC595[5]

Fungsi dari masing-masing pin 74HC595 adalah :

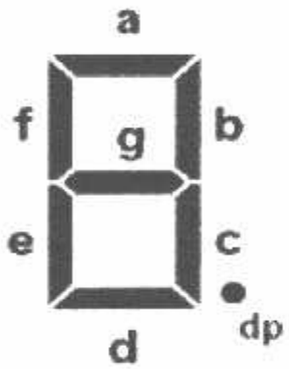
- Pin SER (14) berfungsi sebagai masukan secara serial.
- Pin SRCLK (11) berfungsi sebagi *clock register* geser serial.
- Pin SRCLR (10) berfungsi sebagai *clear register* serial.
- Pin RCLK (12) berfungsi sebagai *clock* bersama *register lacth.*
- Pin E (3) berfungsi sebagai *enable tri state output register* geser.
- Pin Q0 – Q7 dan Q'7 (1-7, 9, 15) berfungsi sebagai keluaran *register lacth.*

2.6 *Display Seven Segment*

Penampil-penampil biner dari sandi BCD menjadi bilangan desimal selain dalam tabung angka (*nixe tube*) yang sudah berbentuk angka –angka desimal dari 0 sampai 9 dapat di tampilkan dengan mengatur penyelaan dari 7 ruas tersebut.

Perlu di ketahui bahwa LED (*Light Emmiting Diode*) adalah suatu diode yang bersifat mengala bila mendapat suatu arus maju (*forward* bias)

Dengan sifatnya yang demikian, LED banyak di pakai pada lampu-lampu penunjuk kecil yang serba guna. Misalnya untuk menunjuk keadaan ON dari suatu peralatan atau lampu-lampu tes, penggunaannya sebagai *seven segmen display* adalah susunannya sesuai gambar:



Gambar 2-15. *Display seven segment*[4]

Pada gambar diatas *segment*-nya adalah a,b,c,d,e,f,g (tujuh buah). Bila semua segmen menyala, maka dapat di baca sebagai angka desimal.

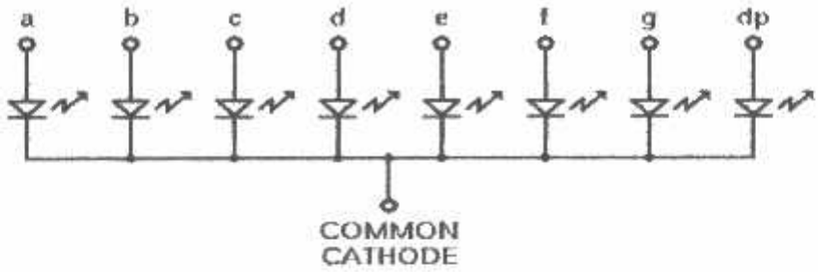
Tabel 2-6

Tabel kebenaran BCD to *seven segment*

Desimal	BCD	Seven segmen						
	8421	A	b	c	d	e	f	g
0	0000	1	1	1	1	1	1	0
1	0001	0	1	1	0	0	0	0
2	0010	1	1	0	1	1	0	1
3	0011	1	1	1	1	0	0	1
4	0100	0	1	1	0	0	1	1
5	0101	1	0	1	1	0	1	1
6	0110	1	0	1	1	1	1	1
7	0111	1	1	1	0	0	0	0
8	1000	1	1	1	1	1	1	1
9	1001	1	1	1	1	0	1	1

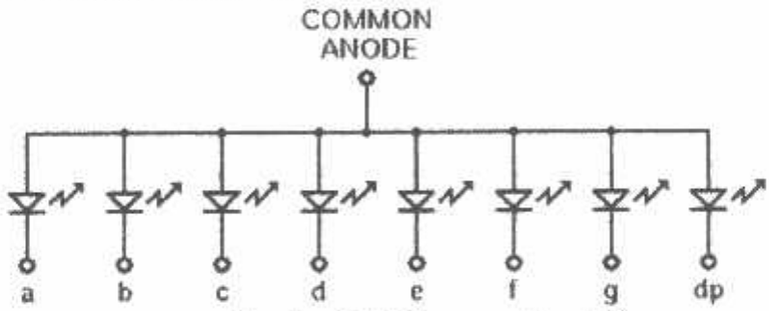
Ada 2 jenis seven segmen:

1. *Seven segment common cathode*:



Gambar 2.16 *Common Cathoda*[4]

2. *Seven segment common Anode*:



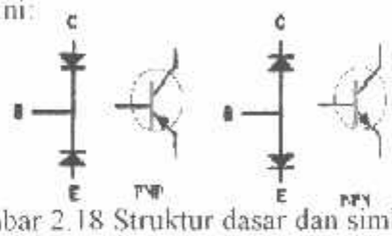
Gambar 2.17 Common Anode[4]

2.7 Transistor

Transistor adalah komponen aktif dengan arus, tegangan atau daya keluarannya dikendalikan oleh arus masukan, merupakan serpihan kristal yang terdiri dari 3 daerah, yaitu : emitor, kolektor, dan basis. Emitor berfungsi menyalurkan atau menyuntikkan emisi ke dalam basis. Basis bertugas meneruskan sebagian besar elektron suntikan dari emitor ke kolektor. Kolektor merupakan daerah terbesar dari ketiga daerah tersebut, karena harus menangani disipasi energi yang lebih besar dari dua daerah lainya.

Simbol sirkuit kedua jenis bahan tersebut hampir sama, perbedaan hanya terletak pada arah panah di ujung emitter. Arah panah ini menunjukkan arah aliran aris konvensional yang berlawanan arah dalam kedua jenis tadi.

Adapun lambang komponen transistor PNP maupun NPN dapat di lihat pada gambar di bawah ini:



Gambar 2.18 Struktur dasar dan simbol transistor[11]

2.7.1 Titik Kerja Transistor

Pemilihan titik kerja bertujuan agar transistor bekerja di daerah yang diinginkan Pada untai penguat, transistor dirancang untuk bekerja di daerah aktif', sehingga sinyal keluaran (tegangan atau arus kolektor) merupakan reproduksi sinyal masukan yang diperkuat.

Dalam pemilihan titik kerja, Vcc, Rb, dan Rc dipilih agar transistor tidak melampaui batas jangkauan (rating)nya yaitu :

a. Lesapan (disipasi) kolektor maksimum , $P_{c(max)}$.

b. Tegangan kolektor-emitor maksimum , $V_{ce(max)}$.

c. Arus kolektor maksimum $I_{c(max)}$.

d. Tegangan basis-emiter maksimum $V_{be(max)}$.

BAB III

PERENCANAAN SISTEM

3.1 Pendahuluan

Pada bab ini akan dibahas mengenai peralatan yang direncanakan dan akan direalisasikan sebagaimana fungsinya. Adapun perencanaan dan pembuatan alat secara garis besar terdapat 2 (dua) perangkat yang ada yaitu:

1. Perencanaan dan pembuatan perangkat keras (*Hardware*)

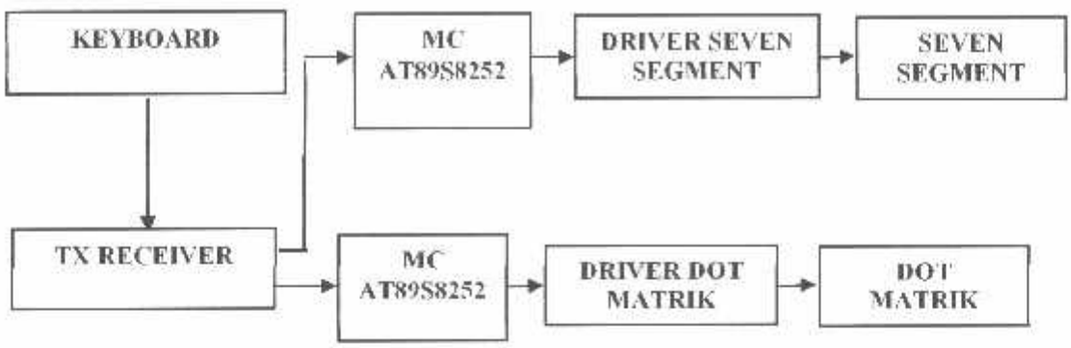
   Terdiri dari 2 (dua) macam sistem, yaitu sistem utama yang terdiri dari : mikrokontroller AT89S8252, *Keyboard*. Sedangkan untuk sistem *display* terdiri dari : mikrokontroller AT89S8252, Dot Matrik, *driver* LED, register geser, *seven segment*.

2. Perencanaan dan pembuatan perangkat lunak (*Software*)

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

3.2 Perencanaan Blok Diagram

Secara garis besar, prinsip kerja dari Alat Penampil Informasi, Jam, dan Kalender Digital dapat dijelaskan melalui diagram blok seperti berikut ini :



Gambar 3.1 Blok Diagram Alat Penampil Informasi, Jam Dan Kalender Digital Pada Tempat Umum

Fungsi dari tiap-tiap blok dapat dijelaskan sebagai berikut :

1. *Keyboard* : Sebagai piranti masukan baik itu berupa data ataupun perintah yang berasal dari perangkat *keyboard PC* yang dihubungkan langsung dengan AT89S8252 dengan jenis komunikasi serial asinkron.
2. TX *Receiver* : Menerima sinyal pulsa dari *transmitter*.
3. MC AT89S8252 : Sebagai pengolah data dan operasi I/O untuk mengendalikan *driver* Dot Matrik serta *Seven Segment*.
4. *Driver Seven Segment* : Mengendalikan *Seven Segment*
5. *Driver* Dot Matrik : Mengendalikan Dot Matrik
6. *Seven Segment* : Tempat untuk menampilkan waktu.
7. Dot Matrik : Tempat untuk menampilkan informasi.

3.3 Prinsip Kerja Alat

Diawali dengan pengaturan waktu yang meliputi jam, menit, tanggal, bulan, dan tahun yang dapat dilakukan dengan menggunakan *keyboard* dengan

menekan *alt*. Kemudian data akan terkirim ke minimum sistem dan akan ditampilkan langsung melalui *seven segment*, kemudian pengesetan dapat dilakukan untuik informasi dengan menekan *ctrl+enter* kemudian kita dapat memilih menu yang ada yang akan ditampilkan sesuai dengan keinginan pengguna. Data informasi yang ada juga akan dikirim ke minimum sistem yang kemudian diolah untuk ditampilkan dalam dot matrik.

3.4 Perancangan Perangkat Keras (*Hardware*)

Dalam Tugas Akhir ini, perancangan alat penampil informasi, jam, dan kalender digital menggunakan Mikrokontroller AT89S8252 sebagai pengontrol utama, dan menggunakan dot matrik dan *seven segment* sebagai komponen pendukungnya.

3.4.1 Mikrokontroller AT89S8252

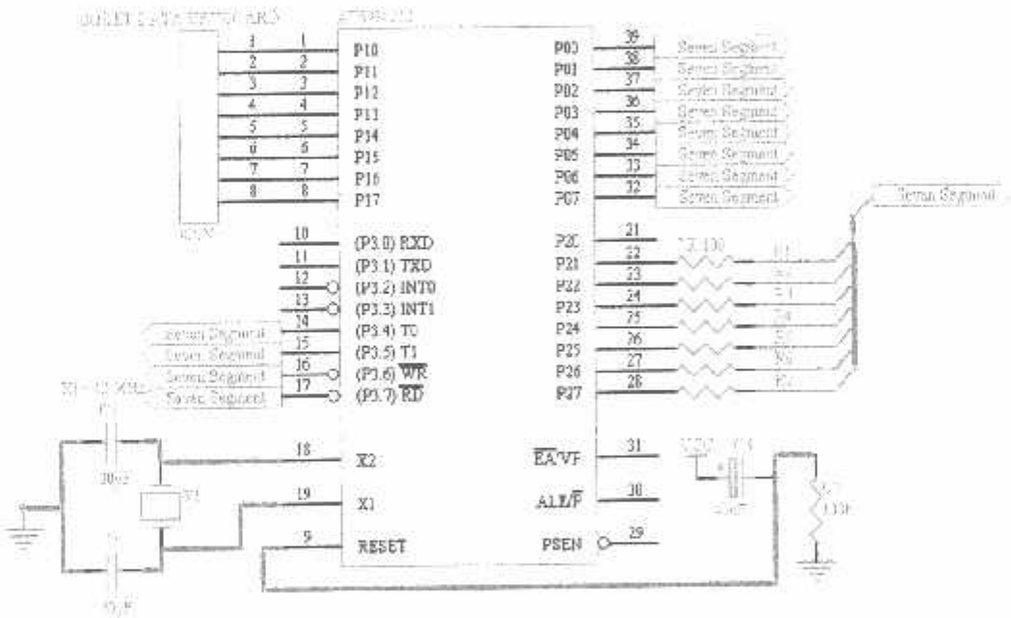
Mikrokontroller AT89S8252 adalah sebuah *chip* IC yang terdiri dari 40 pin. Dalam perencanaan sistem ini pin – pin yang digunakan adalah sebagai berikut:

Adapun fungsi dari masing – masing pin tersebut adalah:

1. Pin 1- 8 (P1.0 – P1.7) merupakan *port 1* yang digunakan sebagai *input*-an dari keyboard.
2. Pin 32-39 (P0.0 – P0.7) merupakan *port 0* dari mikrokontroller, digunakan sebagai *output*-an ke driver seven segment.
3. Pin 21-28 (P2.0 P2.7) merupakan *port 2* dari mikrokontroller, digunakan sebagai *output*-an ke seven segment.

4. Pin 14-17 (P3.4 - P3.7) merupakan *port 0* dari mikrokontroller, digunakan sebagai *output*-an ke driver seven segment.
5. Pin 18 (XTAL 2) sebagai pembangkit ossilator (clock) XTAL 2
6. Pin 19 (XTAL 1) sebagai Pembangkit Ossilator (clock) XTAL 1
7. Pin 9 berfungsi sebagai *Reset*.

Mikrokontroller pada alat ini tidak dapat bekerja sendiri sehingga masih membutuhkan komponen – komponen pendukung lain, komponen – komponen tersebut saling berhubungan secara *hardware* dan juga *software*. Rangkaian mikrokontroller dapat dilihat pada gambar 3.2 dibawah ini:



Gambar 3.2. Rangkaian Mikrokontroller AT89S8252

3.4.2. Minimum Sistem AT89S8252

Mikrokontroller AT89S8252 dirancang untuk dapat berdiri sendiri, karena sudah terdapat 2 Kbytes EEPROM, 256 bit RAM *internal*, 32 *Programmable I/Ollines*, dan terdapat dua 16 bit *timer/counters*. Dari fasilitas-fasilitas tersebut

dapat memfungsikan mikrokontroller AT89S8252 untuk bekerja dalam *single chip*, maksudnya dengan sebuah mikrokontroller saja sudah dapat mengontrol kerja dari keseluruhan sistem. Dalam perancangan ini digunakan 1 buah mikrokontroller AT89S8252.

- ALE/PROG

  Pulsa ALE (*Address Latch Enable*) hanya digunakan untuk akses ke eksternal memori. Sedangkan PROG digunakan untuk input program selama program *Flash* memori. Karena dalam perancangan ini menggunakan *single chip* maka untuk pin ALE/PROG tidak digunakan.

- PSEN

  Program Strobe Enable berfungsi membaca *strobe* ke program memori eksternal. Untuk itu PSEN tidak digunakan.

- EA/Vpp

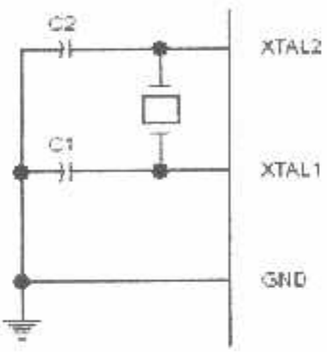
  Logika yang diberikan pada pin ini menunjukkan kerja dari mikrokontroller AT89S51. Pin ini dihubungkan ke Vcc karena pada perancangan ini tidak memanfaatkan memori eksternal.

- XTAL1 dan XTAL2

  Kecepatan proses yang dilakukan oleh mikrokontroller ditentukan oleh sumber *clock* (pewaktu) yang dikendalikan oleh mikrokontroller tersebut. Untuk mendapatkan *clock* pada mikrokontroller, maka digunakan pin XTAL1 dan XTAL2 yang dihubungkan dengan sebuah kristal yang sudah terancang dan tersedia di dalam chip AT89S8252. Besar XTAL yang digunakan adalah 12 MHz dan kapasitansinya sebesar 33 pf yang sesuai

dengan spesifikasi pada data sheet AT89S8252 dengan frekuensi pewaktu berdasarkan kebutuhan dan kecepatan waktu pelaksaan intruksi.

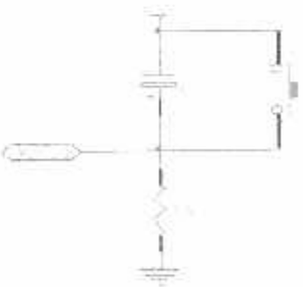
Untuk menjaga kestabilan *clock*, maka ditambah 2 buah kapasitor seperti pada gambar 3.3 dibawah ini:



Gambar 3.3. Rangkaian *Clock*

- *Reset*

Untuk melakukan reset sistem pada mikrokontroller yaitu untuk mengawali eksekusi program pada alamat paling rendah yang dapat dimanfaatkan pin reset yang ada pada mikrokontroller. Pin 9 dihubungkan dengan rangkaian *reset* rangkaian ini diharapkan agar dapat mempunyai kemampuan *power ON Reset*, yaitu Reset terjadi saat *power* diaktifkan. Dibawah ini adalah adalah gambar 3.4 rangkaian reset :



Gambar 3.4. Rangkaian *Reset*

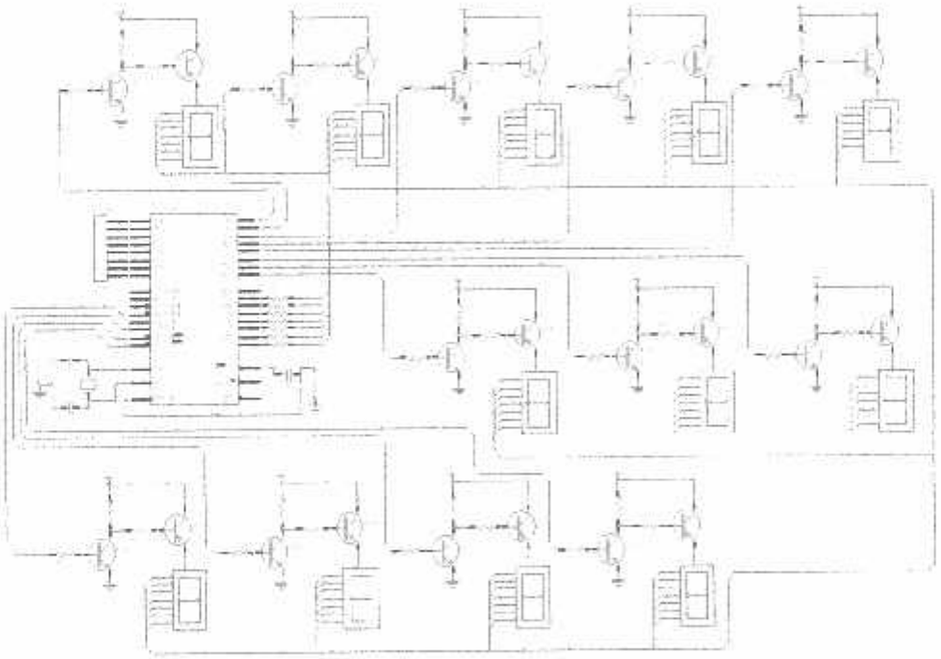
Sehingga dengan komponen resistor dengan nilai 10 K serta kapasitor dengan nilai 47μF akan dihasilkan :

$$T = R.C\ Ln\ 2$$

$$= 0{,}32\mu s$$

3.5 Perencanaan Rangkaian *Display Seven Segment*

Antara dua kaki yang dimiliki oleh LED tersebut salah satu kaki dari ketujuh LED tersebut saling dihubungkan satu sama yang lainnya atau lebih dikenal dengan commom. Commom pada *sevent segment* yang digunakan dalam perancangan ini adalah jenis commom anoda.



Gambar 3.5. Rangkaian *Display Seven Segment*

Asas kerja dari transistor adalah akan ada arus diantara terminal-terminal kolektor-emitor (Ic) hanya apabila ada arus yang mengalir diantara terminal basis-emitor (Ib). jadi transistor harus dioperasikan di daerah linier agar diperoleh sinyal

keluaran yang tidak cacat (distorsi) untuk dapat mengoperasikan secara tepat maka pengertian tentang karakteristik, titik kerja, disipasi daya transistor, dan rangkaian bias (ada yang menyebutnya dengan pra tegangan, tegangan kerja awal) amatlah penting.

$$Rled = \frac{Vled}{Iled}$$

$$Rled = \frac{1.7V}{20mA}$$

$$Ic - 85 = 100\Omega$$

Dari data sheet diketahui nilai β dc dari transistor 9013 =250, maka nilai Ib dapat dicari:

$$Ib = \frac{Ic}{\beta dc}$$

$$Ib = \frac{50mA}{250}$$

$$Ib = 2.10^{-4} A$$

Setelah nilai Ic dan Ib sudah dicari maka kita dapat mencari nilai Rb:

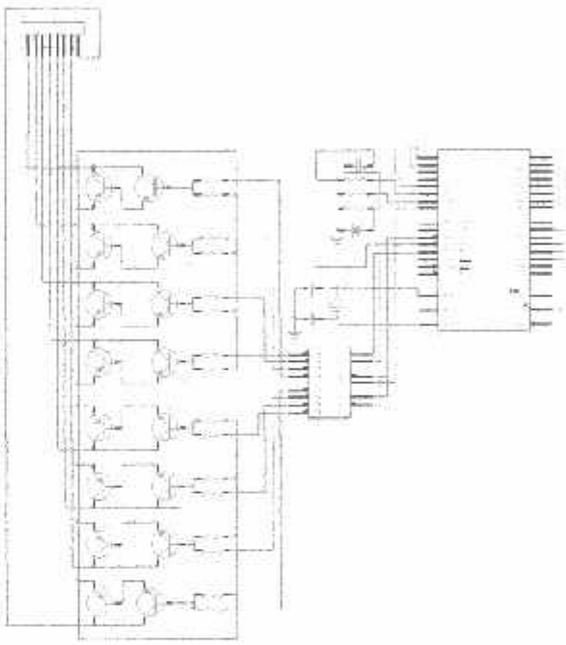
$$Rb = \frac{Vin - VBe}{Ib}$$

$$Rb = \frac{5 - 0.7}{2.10^{-4}}$$

$$Rb = 22K$$

3.6. Perencanaan Rangkaian *Display* Dot Matrik

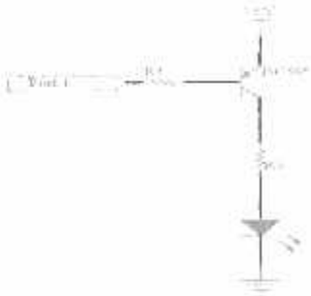
Dot matrik 8x8 (8 kolom dan 8 baris) tersusun dari 64 buah LED yang dirangkai dalam 8 kolom dan 8 baris. *Driver* setiap baris dot matrik dihubungkan dengan port 1 dari *MCU* modul *display*. Sedangkan kutub katoda dari LED akan dihubungkan dengan port dari ULN 2803 yang dirangkai dengan *driver scan* kolom 74HC595.



Gambar 3.6. Rangkaian *Display* Dot Matrik

Setiap *port* dalam P1 akan menangani satu baris dot matrik, dan arena sistem *scan* yang digunakan adalah *scanning* kolom, maka setiap port 1 hanya akan men-*driver* satu LED pada setiap barisnya, sedangkan arus untuk mengaktifkan sebuah LED dot matrik berdasarkan *datasheet* sebesar 30 mA. Apabila dot matrik dioperasikan dalam keadaan 1/10 *duty cycle*, dapat dialirkan arus maksimal

sebesar 100mA dan dalam perencanaan alat *duty cycle* lebih kecil dari nilai tersebut.



Gambar 3.7. *Driver* Setiap Baris *Dot Matrik*

Dari gambar 3.7 kondisi *low* dalam port 1 akan mengaktifkan rangkaian *driver* dot matrik tersebut. Kondisi tersebut akan berkaitan dengan karakteristik port MCU ketika logika *low*, yaitu : Iol (max) = 1.6 mA, Vol (max) = 0,45 V

Arus yang diberikan ke dot matrik (Ic) sebesar 60 mA dengan tegangan bias 4 V. daopat diperoleh nilai R5 (Rc) dari persamaan 3.4 dengan Vce = 0,4 V sebagai berikut :

$$R_1 = \frac{V_{cc} - (V_{EC} + V_{LED})}{I_{LED}}$$

$$R_5 = \frac{5 - (0,4 + 0,4)}{60x10^{-3}} = 10\,\Omega$$

Nilai minimum resistor R_4 dapat diketahui dari persamaan berikut

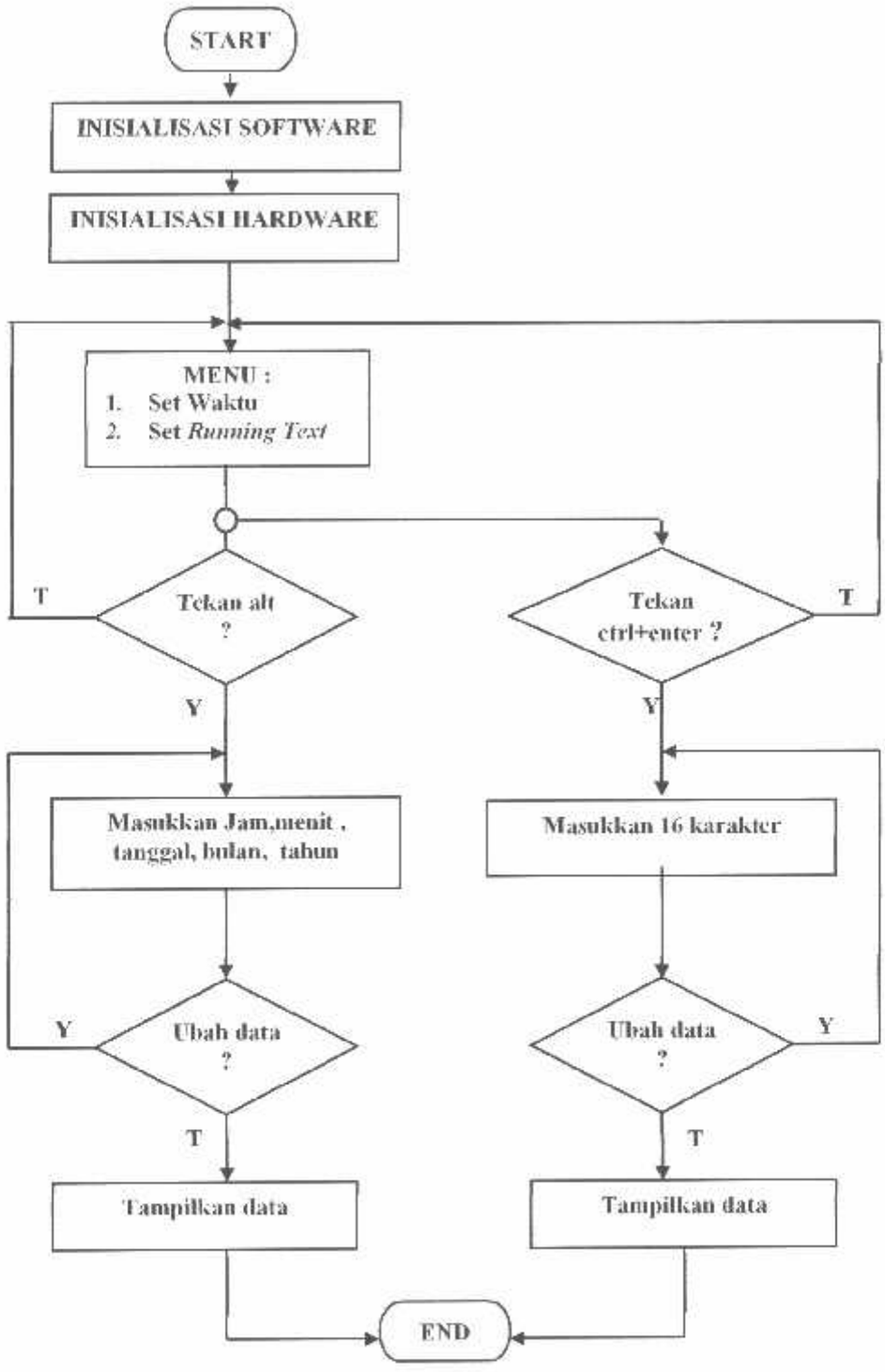
$$R_4 = \frac{V_{BB} - (V_{BE} + V_{OL})}{I_{OL}}$$

$$R_4 = \frac{5 - (0,7 + 0,45)}{1,6x10^{-3}} = 2406,25\,\Omega$$

Digunakan resistor 12 kΩ untuk R_4, dengan nilai tersebut arus Ib (Io1) dapat dihitung kembali. Dari persamaan diatas diperoleh arus sebesar 0,32 mA, tidak melebihi arus maksimal Io1. Dari persamaan diatas dapat diperoleh Ic baru sebesar 64 mA.

3.7 Perencanaan Perangkat Lunak pada Mikrokontroller

Untuk mendukung agar *hardware* berfungsi sesuai dengan perencanaan, maka diperlukan perangkat lunak sebagai penunjangnya. Untuk mengatur dan mengendalikan keseluruhan *hardware* yang telah dibuat, harus dibantu dengan *software*. Sistem aplikasi Mikrokontroller AT89S8252 ini dapat mengatur dan mengendalikan keseluruhan sistem apabila ada urutan instruksi yang mendefinisikan secara jelas urutan tugas yang harus dikerjakan. Cara kerja dari *software* pada Mikrokontroller secara umum sebagai berikut :



Gambar 3.8. *Flowchart Software* pada Mikrokontroller

BAB IV

PENGUJIAN ALAT

4.1. Umum

Pengujian alat ini dilakukan untuk mengetahui kinerja dari keseluruhan sistem rangkaian. Jadi pada tahap ini akan diketahui nilai-nilai serta parameter-parameter dari setiap bagian yang menyusun sistem secara keseluruhan.

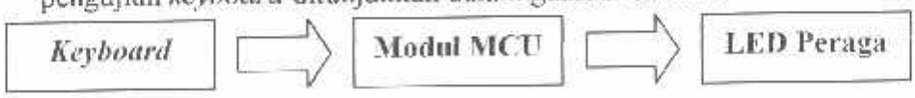
4.2 Pengujian *Hardware*

4.2.1 Pengujian Sistem Mikrokontroller

Tujuan pengujian ini adalah untuk mengetahui apakah sistem minimum modul utama yang dihubungkan dengan *keyboard* dapat bekerja sesuai dengan yang diberikan. Peralatan yang digunakan dalam pengujian ini adalah :

1. Seperangkat komputer
2. Modul program *writer*
3. Modul minimum sistem
4. LED peraga
5. *Keyboard PC*
6. Catu daya 5 Volt

Blok diagram pengujian minimum sistem yang meliputi pengujian *keyboard* ditunjukkan dalam gambar berikut.



Gambar 4.1 Pengujian AT89S8252 dengan antar muka keyboard

Hasil pengujian dari program modul utama dapat dilihat pada tabel 4.1

TABEL 4.1 Hasil Pengujian Program

Tombol	Data *Keyboard* (*Scan Code*)	Kode ASCII	Keluaran LED Peraga
1	16_H	$(31_h)0011\ 0001_b$	$0011\ 0001_b$
2	$1E_H$	$(32_H)1100\ 0010_b$	$1100\ 0010_b$
3	26_H	$(33_H)0011\ 0011_b$	$0011\ 0011_b$
A	$1C_H$	$(41_h)0100\ 0001_b$	$0100\ 0001_b$
B	32_H	$(42_H)0100\ 0010_b$	$0100\ 0010_b$
C	21_H	$(43_H)0100\ 0011_b$	$0100\ 0011_b$

Dari hasil pengujian diketahui ketika ada penekanan tombol *keyboard*, sistem dalam *keyboard* akan mengirimkan data berupa *scan code* dari tombol yang ditekan ke modul MCU. Selanjutnya data tersebut akan dikonversi oleh program MCU menjadi kode ASCII yang ditampilkan dalam *port* 0 berupa data biner. Dari tabel tersebut keluaran pada LED peraga sudah sesuai dengan kode3 ASCII dari tombol *keyboard* yang ditekan. Tidak terjadi kesalahan konversi dari *scan code* ke ASCII. Hal ini menunjukkan bahwa rangkaian dan program antarmuka *keyboard* dengan Mikrokontroller bekerja sesuai dengan yang diharapkan.

4.2.2 Pengujian *Seven Segment*

1. Tujuan

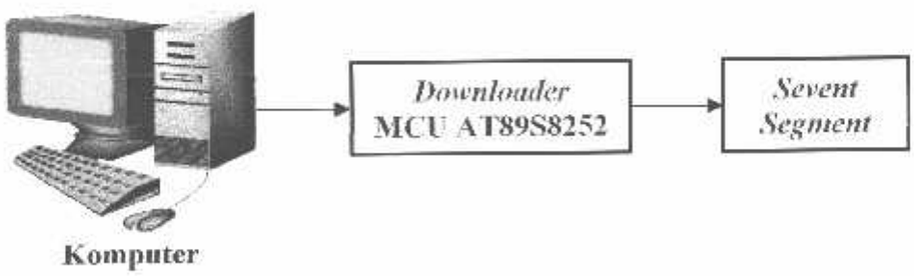
Untuk mengetahui apakah *seven segment* yang dibuat tersebut dapat menampilkan angka.

2. Peralatan yang digunakan

- *PC (Personal Computer)*
- Mikrokontroler dan *Downloader*-nya
- *Seven Segment*
- Catu daya 5 volt

3. Langkah-langkah pengujian

- Merangkai peralatan yang digunakan sesuai gambar 4.2.
- Memberikan catu daya 5 volt pada rangkaian mikrokontroler dan *seven segment*.
- Memberikan data ke mikrokontroler.
- Memberikan data ke *seven segment*.
- Mengamati tampilan *seven segment*.



Gambar 4.2 Pengujian *Sevent Segment*

4. Hasil pengujian

Hasil pengujian *seven segment* ditunjukkan dalam tabel 4.2 berikut ini:

Tabel 4.2 hasil pengujian *Seven Segment*

Data	Tampilan Seven Segmen
00111111b	0
00000110b	1
01011011b	2
01001111b	3
01100110b	4
01101101b	5
01111101b	6
00000111b	7
01111111b	8
01101111b	9

4.2.3 Pengujian Modul *Display* Dot Matrik

1. Tujuan

Untuk mengetahui apakah seluruh LED dot matrik dapat menyala sesuai data yang diberikan oleh mikrokontroller.

2. Peralatan yang digunakan

- Modul *display* dot matrik
- Rangkaian *driver* kolom *display* dot matrik
- Modul minimum sistem AT89S8252
- Catu daya 5 volt

3. Langkah-langkah pengujian

- Merangkai peralatan yang digunakan sesuai gambar 4.3.
- Membuat program tampilan dot matrik dan menulis data yang akan ditampilkan oleh dot matrik
- Memberikan data ke mikrokontroler
- Mengaktifkan catu daya rangkaian.
- Mengamati tampilan dot matrik

Gambar 4.3 Pengujian Dot Matrik

4. Hasil Pengujian

Percobaan *display* dot matrik ini untuk menguji apakah seluruh LED dot matrik dan rangkaian transistor sebagai *driver* baris dot matrik bekerja dengan baik atau tidak. Rangkaian dot matrik disambungkan secara berurutan melalui *shift register* pada sisi kolomnya, sedangkan pada sisi baris dihubungkan dengan *driver* kolom. Apabila logika 1 diberikan pada data baris dan kolom aktif tersebut akan menyala. Sebaliknya LED dot matrik akan padam pada semua kolom yang tidak aktif, sehingga sebenarnya hanya satu kolom saja yang tiap periode *scan* kolomnya. Karena frekuensi *scan* kolom diatas 24 Hz, maka mata normal tidak bisa untuk mengikuti gerak *scan* tersebut sehingga seluruh kolom terlihat aktif semua. Hasil pengujian rangkaian modul MCU, modul *driver*, modul seven segment dan modul dot matrik dapat bekerja sesuai dengan tujuan yang diharapkan.

4.3 Pengujian Alat Secara Keseluruhan

Pada pengujian secara keseluruhan, alat akan dijalankan seluruhnya sebagaimana fungsinya. Tujuan dari pengujian alat secara keseluruhan ini yaitu

untuk mengetahui kinerja alat apakah sudah dapat dijalankan dengan baik atau tidak. Berikut ini langkah-langkah pengujian :

- Menghidupkan catu daya yang telah terhubung dengan *hardware.*
- Memasukkan *scan code* melalui *keyboard* dan mengamati tampilan *seven segment* dan dot matrik.

Setelah penekanan tombol perintah keyboard dimana disini berupa *alt* untuk *setting* waktu dan *ctrl enter* untuk *setting* informasi,selanjutnya program akan menunggu perintah berikutnya. Apabila tombol perintah *setting* waktu (*alt*) ditekan maka modul utama akan mengirim kode pada modul *seven segment.* Sebagai kode prosedur "*setting* waktu" untuk diaktifkan. Selanjutnya modul utama akan mengirim data sesuai penekanan tombol *keyboard* untuk dikirim ke modul *display seven segment,* kemudian *display* akan menampilkan data tersebut sampai ada perintah untuk mengakhiri. Hal yang sama juga dapat dilakukan untuk menampilkan informasi pada *display* dot matrik. Jangkauan *keyboard* sendiri dalam pengesetan, baik itu waktu maupun informasi kurang lebih sekitar 3 meter, lebih dari itu *transfer* data mulai terganggu bahkan tidak bisa sama sekali.

4.4 Spesifikasi Alat

Alat ini terdiri dari dua bagian, yaitu *keyboard* dan *display* informasi yang akan ditampilkan. Alat ini berdimensi cukup besar supaya informasi yang ditampilkan dapat dilihat dengan mudah. *Keyboard* sendiri digunakan untuk mengatur waktu dan mengeset jam dan kalender dengan mudah.

Spesifikasi Alat Secara Umum

Karakter untuk jam dan kalender disini berupa *display seven segment* berukuran 4"inch tiap digitnya dan berjumlah 12 buah yang antara lain :

- 2 digit untuk jam
- 2 digit untuk menit
- 2 digit untuk tanggal
- 2 digit untuk bulan
- 4 digit untuk tahun

Sedangkan untuk menampilkan informasi digunakan *display* dot matrik 8 x 8. *Display* ini dapat menampilkan karakter huruf besar, huruf kecil, angka dan tanda baca umum yang sering digunakan. *Display* ini dapat menampilkan sampai 200 karakter dan dapat memilih mode tampilan dan mode *erase* (penghapusan).

Semua data yang ditampilkan tidak akan hilang walaupun tegangan dimatikan karena disimpan dalam mikrokontroller. *Display* ini hanya dapat menampilkan 16 karakter.

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil perancangan dan pembuatan serta pengukuran alat yang telah dilakukan dapat disimpulkan bahwa :

1. a. Pada perencanaan rangkaian *seven segment* dari hasil perhitungan didapatkan nilai resistansi sebesar 100 Ω .
   b. Alat ini dirancang untuk menampilkan waktu dan tampilan informasi 16 karakter
2. a. Pada pengujian mikrokontroller didapatkan hasil bahwa mikrokontroller berjalan sesuai dengan yang direncanakan yaitu masukan data dari keyboard dan mengirimkannya pada *driver display* yang selanjutnya ditampilkan.
   b. Pada pengujian rangkaian *seven segment* pengiriman data tidak bisa dilakukan dengan cepat dikarenakan adanya delay.
   c. Pengesetan informasi lebih dari 16 karakter tidak dapat dilakukan karena melebihi kapasitas dot matrik

5.2. Saran – saran

1. Dengan segala keterbatasan yang dimiliki penulis maka, terbatasnya kinerja alat ini seperi jangkauan keyboard yang terbatas, sebaiknya alat ini dikembangkan lagi
2. Hendaknya alat ini dipergunakan sebagaimana fungsinya yaitu unutk menampilkan informasi, jam dan kalender digital pada tempat umum
3. Sebaiknya alat ini diterapkan dan digunakan sesuai dengan prosedur dan ketentuan yang telah disebutkan.

DAFTAR PUSTAKA

[1] ATMEL, *Datasheet Book*

[2] Agfianto Eko Putra, "*Belajar Mikrokontroller AT89S8252 Teori dan Aplikasi*", Gava Media, Yogyakarta

[3] *http://www.atmel.com*

[4] *http://www.fairchildsemiconductor.com*

[5] *http://www.howstuffworks.com*

[6] *http://www.Petra.ac.id.*

[7] Malvino, "*Prinsip-Prinsip Elektronika*", edisi kedua, diterjemahkan oleh Hanapi Gunawan, Penerbit Erlangga

[8] Roger L. Tokheim, "*Prinsip-Prinsip Digital*", edisi kedua, diterjemahkan oleh Sutisna, Penerbit Erlangga, Jakarta,1996

[9] Soedarto Gatot, "*Teknik Komputer Dasar-Dasar Digital*, Penerbit Usaha Nasional

[10] Warsito S, "*Vademekum Elektronika*", PT. Gramedia, Jakarta, 1989

LAMPIRAN

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSRTI
JURUSAN TEKNIK ELEKTRO S – I
KONSENTRASI ELEKTRONIKA

LEMBAR BIMBINGAN SKRIPSI

1. Nama : Andrie Dian Rinaldy
2. Nim : 00.17.078
3. Judul Skripsi : Perancangan Dan Pembuatan Alat Penampil Informasi, Jam Dan Kalender Digital Pada Tempat Umum Berbasis Mikrokontroller AT89S8252
4. Tanggal Pengajuan : 15 September 2005
5. Selesai Menulis : 18 Maret 2006
6. Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT
7. Telah Dievaluasi Dengan Nilai : 92

Mengetahui
Ketua Jurusan

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

Diperiksa dan Disetujui
Dosen Pembimbing

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

INSTITUT TEKNOLOGI NASIONAL
JL. BENDUNGAN SIGURA-GURA 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Andrie Dian Rinaldy
Nim : 0017078
Masa Bimbingan : 10-Sep-2005 s/d 13-Mar-2006
Judul Skripsi : Perancangan dan pembuatan alat penampil informasi, jam dan kalender digital pada tempat umum berbasis mikrokontroller AT89S8252

NO	Tanggal	Uraian	Paraf Pembimbing
1.	01-02-2006	Revisi penulisan pada Bab I	
2.	03-02-2006	Revisi penulisan letak judul tabel 2.3 pada Bab II	
3.	08-02-2006	Revisi gambar dan lebar *paragraph* penulisan pada tabel 3.3 Bab III	
4.	11-02-2006	Revisi hasil *error* pengujian alat pada Bab IV	
5.	15-02-2006	Revisi isi kesimpulan dan saran pada Bab V	
6.	19-02-2006	Revisi lebar *paragraph* penulisan Pada daftar pustaka	
7.	22-02-2006	Revisi makalah seminar hasil skripsi	
8.	14- 03-2006	ACC makalah seminar hasil Skripsi	
9.	19-03-2006	ACC Laporan Skripsi	

Malang, 17 Maret 2006
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT

Form S-4a

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

FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Elektronika yang diselenggarakan pada :

Hari : Kamis
Tanggal : 23 Maret 2006
Telah dilakukan perbaikan Skripsi oleh :
Nama : Andric Dian Rinaldy
NIM : 00.17.078
Jurusan : Teknik Elektrto S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : Perancangan dan Pembuatan Alat Penampil Informasi, Jam Dan Kalender Digital Pada Tempat Umum Berbasis Mikrokontroller AT89S8252

No.	Materi Perbaikan	Paraf
1.	Perbaikan gambar (Skematik)	
2.	Kata Pengantar (Nama Dekan)	

Mengetahui
Dosen Pembimbing

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

Disetujui
Penguji

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

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

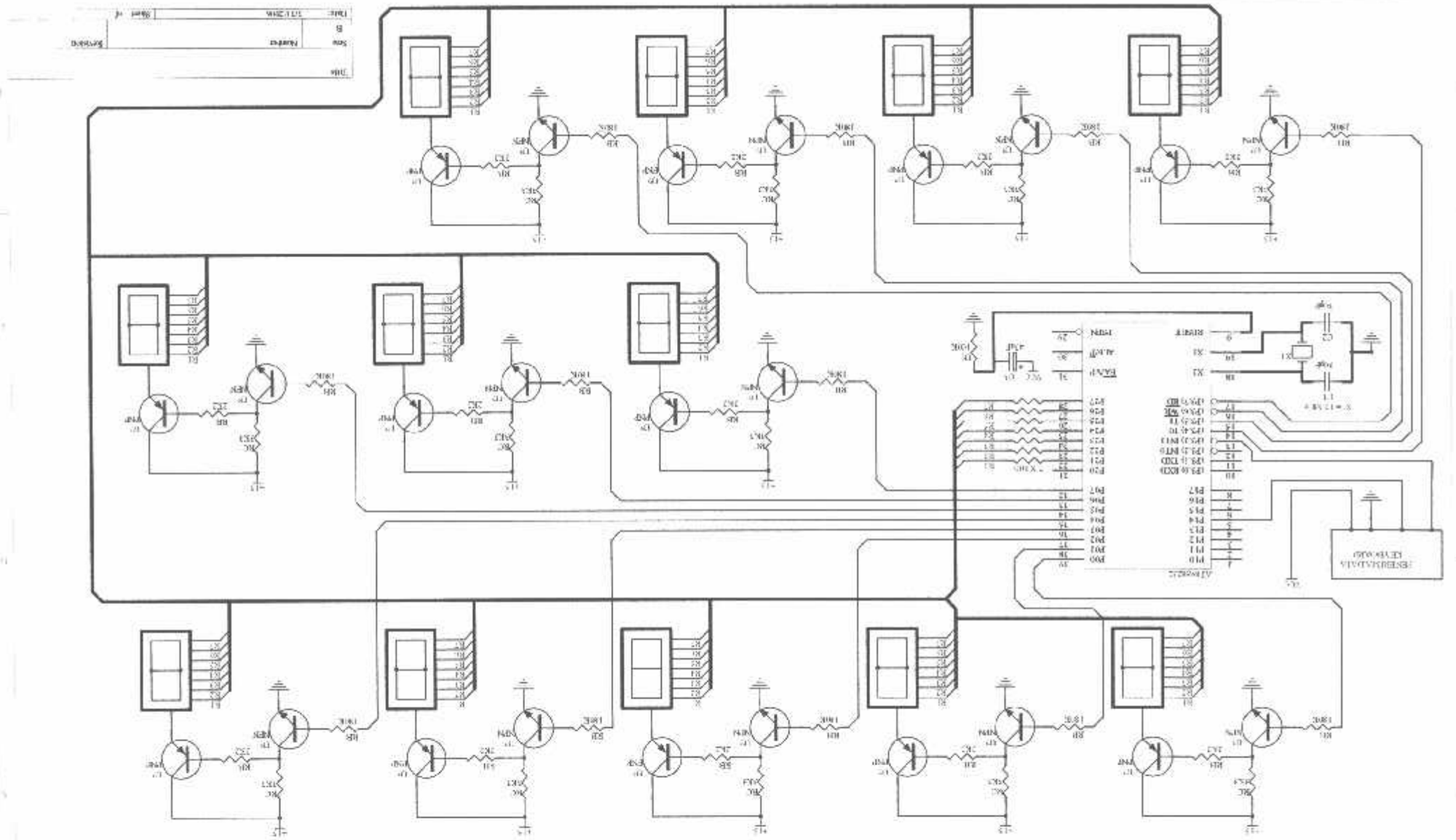
Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentasi T. Energi Listrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

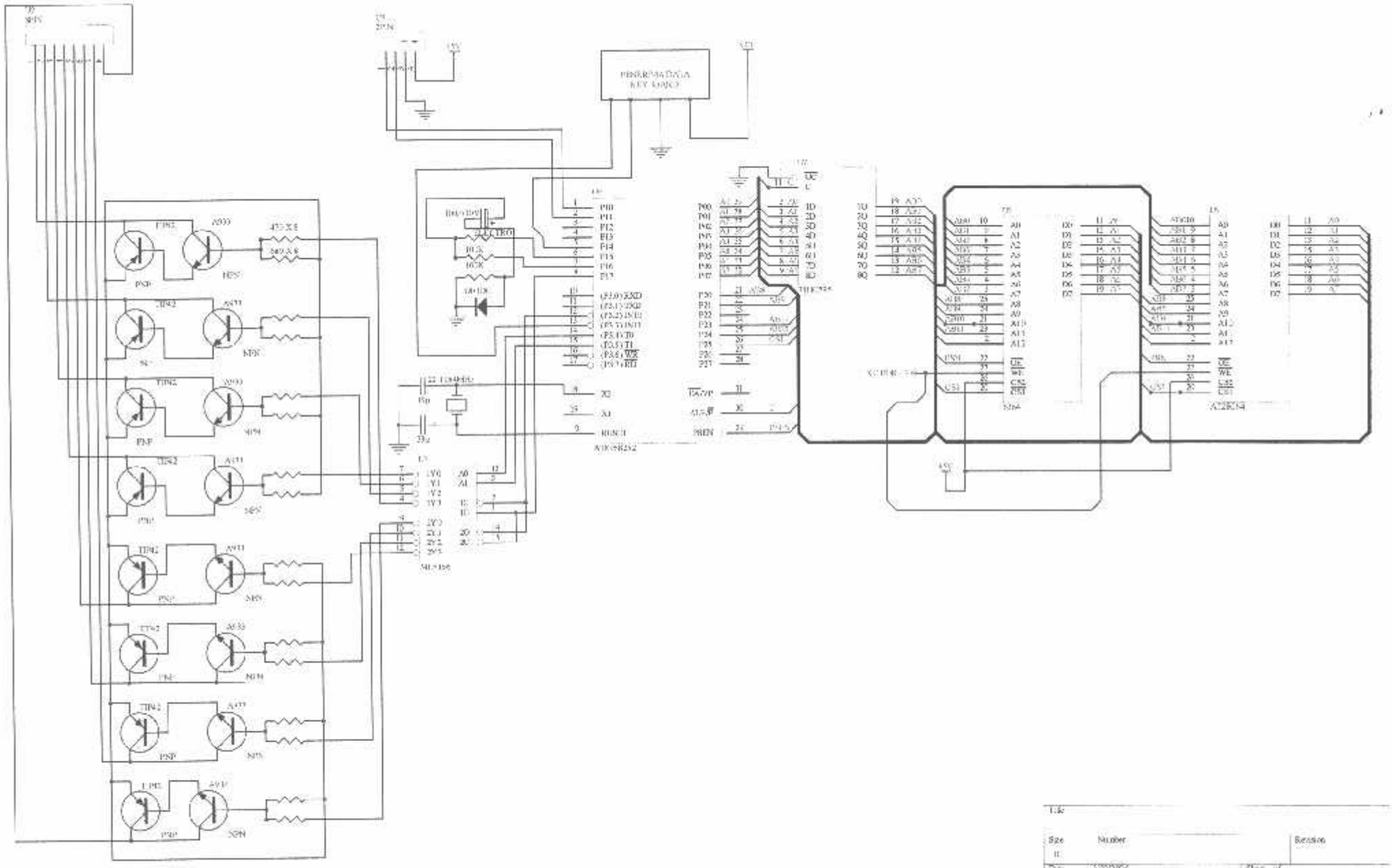
NAMA : ANDRIE DIAN R.
N I M : 0017078.
Perbaikan meliputi :

(1) Perbaiki gambar (interface ke keyboard)
(2) Dehan, → Bukan hal Maka

Malang,

( )





Size	Number	Revision
B		
Date:	5/22/2002	Sheet of
File:	C:\DATA\...\SCANKEY.SCH	Drawn By:

```
;=============================================
;               PROGRAM TA-2006
;             ANDRIE DIAN RINALDY
;                 00.17.078
;             T. ELEKTRONIKA S-I
;=============================================
;     PENAMPIL INFORMASI DAN KALENDER
;           PADA TEMPAT UMUM
;=============================================
RAMKAR              EQU   2000H
RAMEDT              EQU   2010H
RAMSIMP             EQU   2200H
RM_TMPL             EQU   2400H

DPL0        EQU   30H
DPH0        EQU   31H
DPL1        EQU   32H
DPH1        EQU   33H
DPL2        EQU   34H
DPH2        EQU   35H
DPL3        EQU   36H
DPH3        EQU   37H
KLM         EQU   38H
CONT        EQU   39H
DT_RMT              EQU   3AH

DT_MEN              EQU   01H
DT_DON              EQU   02H
DT_GSKN             EQU   03H
DT_CAPS             EQU   04H
DT_SPC              EQU   05H
DT_GSKR             EQU   06H
DT_ENTR             EQU   07H
DT_ADTM             EQU   08H
DT_CLR              EQU   09H

TKN0        DATA  20H
MN_ACTV             BIT   TKN0.0
CAPS_ACTV   BIT   TKN0.1
TM_ACTV             BIT   TKN0.2
CLR_ACTV    BIT   TKN0.3
CHK_ACTV    BIT   TKN0.4
EDT_ACTV    BIT   TKN0.5

OUT_PORT    DATA  P0

BARC        BIT   P1.3
BAR1        BIT   P1.1
SCAN        BIT   P1.4
CLOCK       BIT   P1.5

            ORG   00H
            LJMP  Start
            ORG   23H
;=============================================
;     DATA TAMPILAN
```

```
;=========  ------------------------------

Menu:       DB    'Program Page    1'
            DB    'Program Page    2'
            DB    'Program Page    3'
            DB    'Program Page    4'
            DB    'Program Page    5'
            DB    'Program Page    7'
            DB    'Program Page    8'
            DB    'Run Page  :     1'
            DB    'Run Page  :     2'
            DB    'Run Page  :     3'
            DB    'Run Page  :     4'
            DB    'Run Page  :     5'
            DB    'Run Page  :     6'
            DB    'Run Page  :     7'
            DB    'Run Page  :     8'
            DB    'T00 Disp Mode:  1'
            DB    'T00 Disp Mode:  2'
            DB    'T00 Disp Mode:  3'
            DB    'T00 Disp Mode:  4'
            DB    'T00 Disp Mode:  5'
            DB    'T00 Disp Mode:  6'
            DB    'T00 Disp Mode:  7'
            DB    'T00 Disp Mode:  8'
            DB    'T00 Disp Mode:  9'
            DB    'T00 Disp Mode:  0'
            DB    'T00 Disp Mode: 11'
            DB    'T00 Disp Mode: 12'
            DB    'T00 Disp Mode: 13'
            DB    'T00 Disp Mode: 14'
            DB    'T00 Disp Mode: 15'
            DB    'T00 Disp Mode: 16'
            DB    'T01 Disp Mode:  1'
            DB    'T01 Disp Mode:  2'
            DB    'T01 Disp Mode:  3'
            DB    'T01 Disp Mode:  4'
            DB    'T01 Disp Mode:  5'
            DB    'T01 Disp Mode:  6'
            DB    'T01 Disp Mode:  7'
            DB    'T01 Disp Mode:  8'
            DB    'T01 Disp Mode:  9'
            DB    'T01 Disp Mode: 10'
            DB    'T01 Disp Mode: 11'
            DB    'T01 Disp Mode: 12'
            DB    'T01 Disp Mode: 13'
            DB    'T01 Disp Mode: 14'
            DB    'T01 Disp Mode: 15'
            DB    'T01 Disp Mode: 16'
            DB    'T02 Disp Mode:  1'
            DB    'T02 Disp Mode:  2'
            DB    'T02 Disp Mode:  3'
            DB    'T02 Disp Mode:  4'
            DB    'T02 Disp Mode:  5'
            DB    'T02 Disp Mode:  6'
            DB    'T02 Disp Mode:  7'
```

```
DB    'T02 Disp Mode:  8'
DB    'T02 Disp Mode:  9'
DB    'T02 Disp Mode: 10'
DB    'T02 Disp Mode: 11'
DB    'T02 Disp Mode: 12'
DB    'T02 Disp Mode: 13'
DB    'T02 Disp Mode: 14'
DB    'T02 Disp Mode: 15'
DB    'T02 Disp Mode: 16'
DB    'T03 Disp Mode:  1'
DB    'T03 Disp Mode:  2'
DB    'T03 Disp Mode:  3'
DB    'T03 Disp Mode:  4'
DB    'T03 Disp Mode:  5'
DB    'T03 Disp Mode:  6'
DB    'T03 Disp Mode:  7'
DB    'T03 Disp Mode:  8'
DB    'T03 Disp Mode:  9'
DB    'T03 Disp Mode: 10'
DB    'T03 Disp Mode: 11'
DB    'T03 Disp Mode: 12'
DB    'T03 Disp Mode: 13'
DB    'T03 Disp Mode: 14'
DB    'T03 Disp Mode: 15'
DB    'T03 Disp Mode: 16'
DB    'T04 Disp Mode:  1'
DB    'T04 Disp Mode:  2'
DB    'T04 Disp Mode:  3'
DB    'T04 Disp Mode:  4'
DB    'T04 Disp Mode:  5'
DB    'T04 Disp Mode:  6'
DB    'T04 Disp Mode:  7'
DB    'T04 Disp Mode:  8'
DB    'T04 Disp Mode:  9'
DB    'T04 Disp Mode: 10'
DB    'T04 Disp Mode: 11'
DB    'T04 Disp Mode: 12'
DB    'T04 Disp Mode: 13'
DB    'T04 Disp Mode: 14'
DB    'T04 Disp Mode: 15'
DB    'T04 Disp Mode: 16'
DB    'T05 Disp Mode:  1'
DB    'T05 Disp Mode:  2'
DB    'T05 Disp Mode:  3'
DB    'T05 Disp Mode:  4'
DB    'T05 Disp Mode:  5'
DB    'T05 Disp Mode:  6'
DB    'T05 Disp Mode:  7'
DB    'T05 Disp Mode:  8'
DB    'T05 Disp Mode:  9'
DB    'T05 Disp Mode: 10'
DB    'T05 Disp Mode: 11'
DB    'T05 Disp Mode: 12'
DB    'T05 Disp Mode: 13'
DB    'T05 Disp Mode: 14'
DB    'T05 Disp Mode: 15'
```

```
DB    'T05 Disp Mode: 16'
DB    'T06 Disp Mode:  1'
DB    'T06 Disp Mode:  2'
DB    'T06 Disp Mode:  3'
DB    'T06 Disp Mode:  4'
DB    'T06 Disp Mode:  5'
DB    'T06 Disp Mode:  6'
DB    'T06 Disp Mode:  7'
DB    'T06 Disp Mode:  8'
DB    'T06 Disp Mode:  9'
DB    'T06 Disp Mode: 10'
DB    'T06 Disp Mode: 11'
DB    'T06 Disp Mode: 12'
DB    'T06 Disp Mode: 13'
DB    'T06 Disp Mode: 14'
DB    'T06 Disp Mode: 15'
DB    'T06 Disp Mode: 16'
DB    'T07 Disp Mode:  1'
DB    'T07 Disp Mode:  2'
DB    'T07 Disp Mode:  3'
DB    'T07 Disp Mode:  4'
DB    'T07 Disp Mode:  5'
DB    'T07 Disp Mode:  6'
DB    'T07 Disp Mode:  7'
DB    'T07 Disp Mode:  8'
DB    'T07 Disp Mode:  9'
DB    'T07 Disp Mode: 10'
DB    'T07 Disp Mode: 11'
DB    'T07 Disp Mode: 12'
DB    'T07 Disp Mode: 13'
DB    'T07 Disp Mode: 14'
DB    'T07 Disp Mode: 15'
DB    'T07 Disp Mode: 16'
DB    'T08 Disp Mode:  1'
DB    'T08 Disp Mode:  2'
DB    'T08 Disp Mode:  3'
DB    'T08 Disp Mode:  4'
DB    'T08 Disp Mode:  5'
DB    'T08 Disp Mode:  6'
DB    'T08 Disp Mode:  7'
DB    'T08 Disp Mode:  8'
DB    'T08 Disp Mode:  9'
DB    'T08 Disp Mode: 10'
DB    'T08 Disp Mode: 11'
DB    'T08 Disp Mode: 12'
DB    'T08 Disp Mode: 13'
DB    'T08 Disp Mode: 14'
DB    'T08 Disp Mode: 15'
DB    'T08 Disp Mode: 16'
DB    'T09 Disp Mode:  1'
DB    'T09 Disp Mode:  2'
DB    'T09 Disp Mode:  3'
DB    'T09 Disp Mode:  4'
DB    'T09 Disp Mode:  5'
DB    'T09 Disp Mode:  6'
DB    'T09 Disp Mode:  7'
```

```
DB    'T09 Disp Mode:  8'
DB    'T09 Disp Mode:  9'
DB    'T09 Disp Mode: 10'
DB    'T09 Disp Mode: 11'
DB    'T09 Disp Mode: 12'
DB    'T09 Disp Mode: 13'
DB    'T09 Disp Mode: 14'
DB    'T09 Disp Mode: 15'
DB    'T09 Disp Mode: 16'
DB    'T10 Disp Mode:  1'
DB    'T10 Disp Mode:  2'
DB    'T10 Disp Mode:  3'
DB    'T10 Disp Mode:  4'
DB    'T10 Disp Mode:  5'
DB    'T10 Disp Mode:  6'
DB    'T10 Disp Mode:  7'
DB    'T10 Disp Mode:  8'
DB    'T10 Disp Mode:  9'
DB    'T10 Disp Mode: 10'
DB    'T10 Disp Mode: 11'
DB    'T10 Disp Mode: 12'
DB    'T10 Disp Mode: 13'
DB    'T10 Disp Mode: 14'
DB    'T10 Disp Mode: 15'
DB    'T10 Disp Mode: 16'
DB    'T00 D. Speed :  1'
DB    'T00 D. Speed :  2'
DB    'T00 D. Speed :  3'
DB    'T00 D. Speed :  4'
DB    'T00 D. Speed :  5'
DB    'T00 D. Speed :  6'
DB    'T00 D. Speed :  7'
DB    'T00 D. Speed :  8'
DB    'T00 D. Speed :  9'
DB    'T00 D. Speed : 10'
DB    'T01 D. Speed :  1'
DB    'T01 D. Speed :  2'
DB    'T01 D. Speed :  3'
DB    'T01 D. Speed :  4'
DB    'T01 D. Speed :  5'
DB    'T01 D. Speed :  6'
DB    'T01 D. Speed :  7'
DB    'T01 D. Speed :  8'
DB    'T01 D. Speed :  9'
DB    'T01 D. Speed : 10'
DB    'T02 D. Speed :  1'
DB    'T02 D. Speed :  2'
DB    'T02 D. Speed :  3'
DB    'T02 D. Speed :  4'
DB    'T02 D. Speed :  5'
DB    'T02 D. Speed :  6'
DB    'T02 D. Speed :  7'
DB    'T02 D. Speed :  8'
DB    'T02 D. Speed :  9'
DB    'T02 D. Speed : 10'
DB    'T03 D. Speed :  1'
```

```
DB    'T03 D. Speed :  2'
DB    'T03 D. Speed :  3'
DB    'T03 D. Speed :  4'
DB    'T03 D. Speed :  5'
DB    'T03 D. Speed :  6'
DB    'T03 D. Speed :  7'
DB    'T03 D. Speed :  8'
DB    'T03 D. Speed :  9'
DB    'T03 D. Speed : 10'
DB    'T04 D. Speed :  1'
DB    'T04 D. Speed :  2'
DB    'T04 D. Speed :  3'
DB    'T04 D. Speed :  4'
DB    'T04 D. Speed :  5'
DB    'T04 D. Speed :  6'
DB    'T04 D. Speed :  7'
DB    'T04 D. Speed :  8'
DB    'T04 D. Speed :  9'
DB    'T04 D. Speed : 10'
DB    'T05 D. Speed :  1'
DB    'T05 D. Speed :  2'
DB    'T05 D. Speed :  3'
DB    'T05 D. Speed :  4'
DB    'T05 D. Speed :  5'
DB    'T05 D. Speed :  6'
DB    'T05 D. Speed :  7'
DB    'T05 D. Speed :  8'
DB    'T05 D. Speed :  9'
DB    'T05 D. Speed : 10'
DB    'T06 D. Speed :  1'
DB    'T06 D. Speed :  2'
DB    'T06 D. Speed :  3'
DB    'T06 D. Speed :  4'
DB    'T06 D. Speed :  5'
DB    'T06 D. Speed :  6'
DB    'T06 D. Speed :  7'
DB    'T06 D. Speed :  8'
DB    'T06 D. Speed :  9'
DB    'T06 D. Speed : 10'
DB    'T07 D. Speed :  1'
DB    'T07 D. Speed :  2'
DB    'T07 D. Speed :  3'
DB    'T07 D. Speed :  4'
DB    'T07 D. Speed :  5'
DB    'T07 D. Speed :  6'
DB    'T07 D. Speed :  7'
DB    'T07 D. Speed :  8'
DB    'T07 D. Speed :  9'
DB    'T07 D. Speed : 10'
DB    'T08 D. Speed :  1'
DB    'T08 D. Speed :  2'
DB    'T08 D. Speed :  3'
DB    'T08 D. Speed :  4'
DB    'T08 D. Speed :  5'
DB    'T09 D. Speed :  6'
DB    'T09 D. Speed :  7'
```

```
DB     'T08 D. Speed :  8'
DB     'T08 D. Speed :  9'
DB     'T08 D. Speed : 10'
DB     'T09 D. Speed :  1'
DB     'T09 D. Speed :  2'
DB     'T09 D. Speed :  3'
DB     'T09 D. Speed :  4'
DB     'T09 D. Speed :  5'
DB     'T09 D. Speed :  6'
DB     'T09 D. Speed :  7'
DB     'T09 D. Speed :  8'
DB     'T09 D. Speed :  9'
DB     'T09 D. Speed : 10'
DB     'T10 D. Speed :  1'
DB     'T10 D. Speed :  2'
DB     'T10 D. Speed :  3'
DB     'T10 D. Speed :  4'
DB     'T10 D. Speed :  5'
DB     'T10 D. Speed :  6'
DB     'T10 D. Speed :  7'
DB     'T10 D. Speed :  8'
DB     'T10 D. Speed :  9'
DB     'T10 D. Speed : 10'
DB     'T00 Hold Time:  1'
DB     'T00 D. Speed :  2'
DB     'T00 D. Speed :  3'
DB     'T00 D. Speed :  4'
DB     'T00 D. Speed :  5'
DB     'T00 D. Speed :  6'
DB     'T00 D. Speed :  7'
DB     'T00 D. Speed :  8'
DB     'T00 D. Speed :  9'
DB     'T00 D. Speed : 10'
DB     'T01 D. Speed :  1'
DB     'T01 D. Speed :  2'
DB     'T01 D. Speed :  3'
DB     'T01 D. Speed :  4'
DB     'T01 D. Speed :  5'
DB     'T01 D. Speed :  6'
DB     'T01 D. Speed :  7'
DB     'T01 D. Speed :  8'
DB     'T01 D. Speed :  9'
DB     'T01 D. Speed : 10'
DB     'T02 D. Speed :  1'
DB     'T02 D. Speed :  2'
DB     'T02 D. Speed :  3'
DB     'T02 D. Speed :  4'
DB     'T02 D. Speed :  5'
DB     'T02 D. Speed :  6'
DB     'T02 D. Speed :  7'
DB     'T02 D. Speed :  8'
DB     'T02 D. Speed :  9'
DB     'T02 D. Speed : 10'
DB     'T03 D. Speed :  1'
DB     'T03 D. Speed :  2'
DB     'T03 D. Speed :  3'
```

```
DB     'T03 D. Speed :  4'
DB     'T03 D. Speed :  5'
DB     'T03 D. Speed :  6'
DB     'T03 D. Speed :  7'
DB     'T03 D. Speed :  8'
DB     'T03 D. Speed :  9'
DB     'T03 D. Speed : 10'
DB     'T04 D. Speed :  1'
DB     'T04 D. Speed :  2'
DB     'T04 D. Speed :  3'
DB     'T04 D. Speed :  4'
DB     'T04 D. Speed :  5'
DB     'T04 D. Speed :  6'
DB     'T04 D. Speed :  7'
DB     'T04 D. Speed :  8'
DB     'T04 D. Speed :  9'
DB     'T04 D. Speed : 10'
DB     'T05 D. Speed :  1'
DB     'T05 D. Speed :  2'
DB     'T05 D. Speed :  3'
DB     'T05 D. Speed :  4'
DB     'T05 D. Speed :  5'
DB     'T05 D. Speed :  6'
DB     'T05 D. Speed :  7'
DB     'T05 D. Speed :  8'
DB     'T05 D. Speed :  9'
DB     'T05 D. Speed : 10'
DB     'T06 D. Speed :  1'
DB     'T06 D. Speed :  2'
DB     'T06 D. Speed :  3'
DB     'T06 D. Speed :  4'
DB     'T06 D. Speed :  5'
DB     'T06 D. Speed :  6'
DB     'T06 D. Speed :  7'
DB     'T06 D. Speed :  8'
DB     'T06 D. Speed :  9'
DB     'T06 D. Speed : 10'
DB     'T07 D. Speed :  1'
DB     'T07 D. Speed :  2'
DB     'T07 D. Speed :  3'
DB     'T07 D. Speed :  4'
DB     'T07 D. Speed :  5'
DB     'T07 D. Speed :  6'
DB     'T07 D. Speed :  7'
DB     'T07 D. Speed :  8'
DB     'T07 D. Speed :  9'
DB     'T07 D. Speed : 10'
DB     'T08 D. Speed :  1'
DB     'T08 D. Speed :  2'
DB     'T08 D. Speed :  3'
DB     'T08 D. Speed :  4'
DB     'T08 D. Speed :  5'
DB     'T08 D. Speed :  6'
DB     'T08 D. Speed :  7'
DB     'T08 D. Speed :  8'
DB     'T08 D. Speed :  9'
```

```
        DB    'T08 D. Speed : 10'
        DB    'T09 D. Speed :  1'
        DB    'T09 D. Speed :  2'
        DB    'T09 D. Speed :  3'
        DB    'T09 D. Speed :  4'
        DB    'T09 D. Speed :  5'
        DB    'T09 D. Speed :  6'
        DB    'T09 D. Speed :  7'
        DB    'T09 D. Speed :  8'
        DB    'T09 D. Speed :  9'
        DB    'T09 D. Speed : 10'
        DB    'T10 D. Speed :  1'
        DB    'T10 D. Speed :  2'
        DB    'T10 D. Speed :  3'
        DB    'T10 D. Speed :  4'
        DB    'T10 D. Speed :  5'
        DB    'T10 D. Speed :  6'
        DB    'T10 D. Speed :  7'
        DB    'T10 D. Speed :  8'
        DB    'T10 D. Speed :  9'
        DB    'T10 D. Speed : 10'

Save:   DB    '                    '
        DB    'Save T00 ?     Yes  '
        DB    '                    ',0H

;-------------------------------------------
;     TRANSFER DATA
;-------------------------------------------
Simpan: MOV   DPTR,#RAMKAR
        MOV   DPH0,DPH
        MOV   DPL0,DPL
        MOV   DPTR,#RAMSIMP
        MOV   DPH1,DPH
        MOV   DPL1,DPL
        ACALL Isi_RAM
        RET

Kmslkn: MOV   DPTR,#RAMSIMP
        ACALL Set_Almt
        ACALL Isi_RAM
        ACALL Isi_Konv
        RET

Isi_Menu: MOV   DPTR,#MENU
        ACALL Set_Almt
        ACALL Fill
        ACALL Isi_Konv
        RET
```

```
Isi_Sur:    MOV   DPTR,#Sure
            ACALL Set_Almt
            ACALL Fill
            ACALL Isi_Konv
            RET

Isi_RAM:    MOV   DPH,DPH0
            MOV   DPL,DPL0
            MOVX  A,@DPTR             ;AMBIL DATA
            INC   DPTR
            MOV   DPL0,DPL
            MOV   DPH0,DPH
            CJNE  A,#0H,Put_It        ;DATA=0?
            MOV   DPL,DPL1
            MOV   DPH,DPH1
            MOVX  @DPTR,A
            RET

Put_It:     MOV   DPL,DPL1
            MOV   DPH,DPH1
            MOVX  @DPTR,A
            INC   DPTR
            MOV   DPL1,DPL
            MOV   DPH1,DPH
            SJMP  Isi_RAM

Fill:       MOV   DPH,DPH0
            MOV   DPL,DPL0
            CLR   A
            MOVC  A,@A+DPTR    ;AMBIL DATA
            INC   DPTR
            MOV   DPL0,DPL
            MOV   DPH0,DPH
            CJNE  A,#0H,Fill_It       ;DATA=0?
            MOV   DPL,DPL1
            MOV   DPH,DPH1
            MOVX  @DPTR,A             ;RAMKAR=0
            RET

Fill_It:    MOV   DPL,DPL1     ;DPTR=RAMKAR
            MOV   DPH,DPH1
            MOVX  @DPTR,A
            INC   DPTR
            MOV   DPL1,DPL
            MOV   DPH1,DPH
            SJMP  Fill

Empt_RAM:   MOV   DPTR,#RAMSIMP
Kong:       MOV   A,#20H
            MOVX  @DPTR,A
            INC   DPTR
            MOV   A,DPH
            CJNE  A,#23H,Kong
            MOV   A,DPL
            CJNE  A,#0FFH,Kong
```

```
            CLR   A
            MOVX  @DPTR,A
            RET

Isi_Konv:   MOV   DPTR,#RAMKAR
            MOV   DPH0,DPH     ;CODE BYTE
            MOV   DPL0,DPL
Konvrs:           MOV   DPTR,#RM TMPL
            MOV   DPH1,DPH     ;RAM
            MOV   DPL1,DPL

Isikan:     MOV   DPH,DPH0     ;DPTR=RAMKAR
            MOV   DPL,DPL0
            MOVX  A,@DPTR             ;AMBIL DATA
            INC   DPTR
            MOV   DPH0,DPH
            MOV   DPL0,DPL
            CJNE  A,#0H,Isilah        ;BILA TDK 0 ISI RAM
            MOV   DPL,DPL1     ;BILA 0 ISI RAM #24H
            MOV   DPH,DPH1
            MOV   A,#24H
            MOVX  @DPTR,A
            RET

Isilah:     ACALL Banding             ;ALMT KARAKTER
            MOV   DPH2,DPH
            MOV   DPL2,DPL
Ganti:      CLR   A
            MOVC  A,@A+DPTR    ;AMBIL DATA
            INC   DPTR
            MOV   DPH2,DPH
            MOV   DPL2,DPL
            CJNE  A,#24H,Lanjut       ;DATA-AKHIR KARAKTER
            ACALL Jarak        ;JARAK
            SJMP  Isikan              ;ISI KAR SERIKUT

Lanjut:     MOV   DPH,DPH1     ;RAM
            MOV   DPL,DPL1
            MOVX  @DPTR,A
            INC   DPTR
            MOV   DPH1,DPH
            MOV   DPL1,DPL
            MOV   DPH,DPH2
            MOV   DPL,DPL2
            SJMP  Ganti

Jarak:      MOV   DPH,DPH1
            MOV   DPL,DPL1
            MOV   A,#0FFH
            MOVX  @DPTR,A
            INC   DPTR
            MOVX  @DPTR,A
            INC   DPTR
            MOV   DPH1,DPH
            MOV   DPL1,DPL
            RET
```

```
;-------------------------------------------------
;     SUBROUTINE
;-------------------------------------------------
Delay:          MOV   7FH,#10
                DJNZ  7FH,$
                RET
Wait:           MOV   7FH,#2
                DJNZ  7FH,$
                RET

;-------------------------------------------------
;     DRIVER DOT MATRIX
;-------------------------------------------------
Siap:           ACALL Lock0
                ACALL Lock1
                MOV   KLM,#180
Blnk:           ACALL Gesr
                DJNZ  KLM,Blnk
PULSA:          SETB  SCAN
                ACALL Gesr
                CLR   SCAN
                RET

Gesr:           SETB  CLOCK
                ACALL Wait
                CLR   CLOCK
                ACALL Wait
                RET

Lock0:          SETB  BAR0
                ACALL Wait
                CLR   BAR0
                ACALL Wait
                RET

Lock1:          SETB  BAR1
                ACALL Wait
                CLR   BAR1
                ACALL Wait
                RET

Dot_Off:        MOV   OUT_PORT,#0FFH
                ACALL Lock0
                ACALL Lock1
                RET

Dot_On:         MOV   OUT_PORT,#00H
                ACALL Lock0
                ACALL Lock1
                RET
;-------------------------------------------------
;     CHECK DISPLAY
;-------------------------------------------------
Cek_Disp:       MOV   DT_RMT,#0FFH
Ck_Dsp:         ACALL Siap
                MOV   KLM,#180
```

```
Dsc_On:     ACALL Dot_On
            ACALL Gesr
            MOV   CONT,#10
Priks:      ACALL Delay
            MOV   A,DT_RMT
            CJNE  A,#DT_DON,Nxt_Dot
            MOV   DT_RMT,#0FFH
            CLR   CHK_ACTV
            RET
Nxt_Dot:    DJNZ  CONT,Priks
            ACALL Dot_Off
            DJNZ  KLM,Dsp_On
            SJMP  Ck_Dsp
;-------------------------------------------
;     EDIT TEXT
;-------------------------------------------
Edt_Txt:    ACALL Kmb_kn
            MOV   DPTR,#RAMEDT
            ACALL Sim_Al
Kon:        ACALL Konvrs
Dsp_Kar:    SJMP  Tombl
Tampil:     MOV   KLM,#180
            MOV   DPTR,#RM_TMPL
Shw_Txt:    MOVX  A,@DPTR
            MOV   OUT_PORT,A
            ACALL Lock0
            INC   DPTR
            MOVX  A,@DPTR
            MOV   OUT_PORT,A
            ACALL Lock1
            INC   DPTR
            ACALL Delay
            ACALL Dot_Off
            ACALL Gesr
            DJNZ  KLM,Shw_Txt
            ACALL Pulsa
            SJMP  Dsp_Kar
;-------------------------------------------
Tombl:      MOV   A,DT_RMT
            CJNE  A,#DT_ENTR,Bc_Tombl
            MOV   DT_RMT,#0FFH
            CLR   EDT_ACTV
            ACALL Simpan
            RET

Bc_Tombl:   CJNE  A,#DT_MEN,Tbl_Don
            MOV   DT_RMT,#0FFH
Tbl_Don:    CJNE  A,#DT_DON,Tbl_Tm
            MOV   DT_RMT,#0FFH
Tbl_Tm:     CJNE  A,#DT_ADTM,Tbl_Clr
            MOV   DT_RMT,#0FFH
Tbl_Clr:    CJNE  A,#DT_CLR,Tbl_Cps
            MOV   DT_RMT,#0FFH

Tbl_Cps:    CJNE  A,#DT_CAPS,Tbl_Spc
            MOV   DT_RMT,#0FFH
```

```
                JNB     CAPS_ACTV,Aktiv
                CLR     CAPS_ACTV
                SJMP    Tbl_Spc
Aktiv:          SETB    CAPS_ACTV

Tbl_Spc:        CJNE    A,#DT_SPC,Tbl_Kn
                MOV     DT_RMT,#0FFH
                MOV     DPH,DPH3
                MOV     DPL,DPL3
                MOV     A,DPH
                CJNE    A,#21H,Belm_Mntk
                MOV     A,DPL
                CJNE    A,#0E8H,Belm_Mntk
                SJMP    Oke
Belm_Mntk:      MOV     A,#20H
                MOVX    @DPTR,A
                INC     DPTR
Oke:            ACALL   Sim_Al
                AJMP    Kon

Tbl_Kn:         CJNE    A,#DT_GSKN,Tbl_Kr
                MOV     DT_RMT,#0FFH
                MOV     DPH,DPH3
                MOV     DPL,DPL3
                MOV     A,DPH
                CJNE    A,#21H,Blm_Mntk
                MOV     A,DPL
                CJNE    A,#0E8H,Blm_Mntk
                SJMP    Mentok
Blm_Mntk:       INC     DPTR
Mentok:         ACALL   Sim_Al
                AJMP    Kon

Tbl_Kr:         CJNE    A,#DT_GSKR,In_Char
                MOV     DT_RMT,#0FFH
                MOV     DPH,DPH3
                MOV     DPL,DPL3
                MOV     A,DPH
                CJNE    A,#20H,Blum_Mntk
                MOV     A,DPL
                CJNE    A,#18H,Blum_Mntk
                SJMP    Mntk
Blum_Mntk:      MOV     A,DPL
                DEC     A
                MOV     DPL,A
                CJNE    A,#00H,Mntk
                DEC     DPH
Mntk:           ACALL   Sim_Al
                AJMP    Kon

In_Char:        CJNE    A,#0FFH,In_Kar
                AJMP    Tampil
In_Kar:         MOV     DPH,DPH3
                MOV     DPL,DPL3
                JB      CAPS_ACTV,CapsLock
                ADD     A,#20H
```

```
Capslock:   MOVX  @DPTR,A
            ACALL Sim_Al
            MOV   DT_RMT,#0FFH
            AJMP  Kon

Sim_Al:     MOV   DPH0,DPH
            MOV   DPL0,DPL
            MOV   DPH3,DPH
            MOV   DPL3,DPL
            RET
;-------------------------------------------
;           PROGRAM UTAMA
;-------------------------------------------
Start:            LCALL Inis_Seri
            SETB  EA
            SETB  ES
            ACALL SIAP
            ACALL TST_SLM
            MOV   TKK0,#00H
            MOV   DT_RMT,#0FFH
;-----------  ----------  ----------
;     DISPLAY KARAKTER
;-----------  ----------  ----------
Dspy:       MOV   DPTR,#RM_TMPL
            MOV   DPL0,DPL          ; ADA RAM
            MOV   DPH0,DPH
Baca:       SJMP  Sc_Tmbl

Dsp_Kan:    MOV   DPL,DPL0
            MOV   DPH,DPH0
            MOV   KLM,#100
Shw_Nxt:    MOVX  A,@DPTR
            CJNE  A,#240,Disp
            SJMP  Stop
Disp:       MOV   OUT_PORT,A
            ACALL Lock0
            INC   DPTR
            MOVX  A,@DPTR
            MOV   OUT_PORT,A
            ACALL Lock1
            ACALL Delay
            ACALL Dot_Off
            ACALL Gesr
            INC   DPTR
            DJNZ  KLM,Shw_Nxt
            ACALL Pulsa
            MOV   DPL,DPL0
            MOV   DPH,DPH0
            INC   DPTR          ;TAMPILAN GESER 1KOLOM
            INC   DPTR
            MOV   DPL0,DPL
            MOV   DPH0,DPH
            SJMP  Baca
Stop:       ACALL Pulsa
            SJMP  Dspy
;-----  --------  --------  --------
```

```
;      DETEKSI TOMBOL
;-----------------------------------------
Sc_Tmbl:    MOV   A,DT_RMT
            CJNE  A,#DT_MEN,Option
            MOV   DT_RMT,#0FFH
            JB    MN_ACTV,Option
            JB    EDT_ACTV,Option
            JB    CHK_ACTV,Option
            JB    TM_ACTV,Option
            JB    CLR_ACTV,Option
            SETB  MN_ACTV
            ACALL Simpan
Mn_Prog:    ACALL Tsl_Mnu
            SJMP  Dspy

Option:     CJNE  A,#'1',Dua
            MOV   DT_RMT,#0FFH
            JNB   MN_ACTV,Dua
            JB    EDT_ACTV,Dua
            JB    CHK_ACTV,Dua
            JB    CLR_ACTV,Clear
            SETB  TM_ACTV
            ACALL Isi_Tim
            SJMP  Dspy

Clear:      ACALL Empt_RAM
            CLR   CLR_ACTV
            AJMP  Mn_Prog

Dua:        CJNE  A,#'2',Tiga
            MOV   DT_RMT,#0FFH
            JNB   MN_ACTV,Tiga
            JB    TM_ACTV,Tiga
            JB    CHK_ACTV,Tiga
            JB    CLR_ACTV,Cler
            SETB  EDT_ACTV
            ACALL Edt_Txt
            AJMP  Mn_Prog

Cler:       CLR   CLR_ACTV
            AJMP  Mn_Prog

Tiga:       CJNE  A,#'3',Empat
            MOV   DT_RMT,#0FFH
            JNB   MN_ACTV,Empat
            JB    TM_ACTV,Empat
            JB    EDT_ACTV,Empat
            SETB  CHK_ACTV
            ACALL Chk_Disp
            AJMP  Mn_Prog

Empat:      CJNE  A,#'4',Sc_Don
            MOV   DT_RMT,#0FFH
            JNB   MN_ACTV,Sc_Don
            MOV   CKN0,#00H
            ACALL Kmbl'an
```

```
                AJMP    Dspy

Bc_Don:         CJNE    A,#DT_DON,Bc_Clr
                MOV     DT_RMT,#0FFH
                JNB     MN_ACTV,Bc_Clr
                CLR     TM_ACTV
                AJMP    Mn_Prog

Bc_Clr:         CJNE    A,#DT_CLR,Dsbl_Clr
                MOV     DT_RMT,#0FFH
                JNB     MN_ACTV,Dsbl_Clr
                JB      TM_ACTV,Dsbl_Clr
                JB      CHK_ACTV,Dsbl_Clr
                JB      EDT_ACTV,Dsbl_Clr
                SETB    CLR_ACTV
                ACALL   Isi_Sur
                AJMP    Dspy

Dsbl_Clr:       MOV     DT_RMT,#0FFH
                AJMP    Dsp_Kar
;---------- ------------------------------------
;       PEMBANDING KARAKTER
;------ ------------------------------------
BANDING:
KARA:           CJNE    A,#'A',KARB
                MOV     DPTR,#DT_A
                RET
KARB:           CJNE    A,#'B',KARC
                MOV     DPTR,#DT_B
                RET
KARC:           CJNE    A,#'C',KARD
                MOV     DPTR,#DT_C
                RET
KARD:           CJNE    A,#'D',KARE
                MOV     DPTR,#DT_D
                RET
KARE:           CJNE    A,#'E',KARF
                MOV     DPTR,#DT_E
                RET
KARF:           CJNE    A,#'F',KARG
                MOV     DPTR,#DT_F
                RET
KARG:           CJNE    A,#'G',KARH
                MOV     DPTR,#DT_G
                RET
KARH:           CJNE    A,#'H',KARI
                MOV     DPTR,#DT_H
                RET
KARI:           CJNE    A,#'I',KARJ
                MOV     DPTR,#DT_I
                RET
KARJ:           CJNE    A,#'J',KARK
                MOV     DPTR,#DT_J
                RET
KARK:           CJNE    A,#'K',KARL
                MOV     DPTR,#DT_K
```

```
                RET
KARL:           CJNE    A,#'L',KARM
                MOV     DPTR,#DT_L
                RET
KARM:           CJNE    A,#'M',KARN
                MOV     DPTR,#DT_M
                RET
KARN:           CJNE    A,#'N',KARO
                MOV     DPTR,#DT_N
                RET
KARO:           CJNE    A,#'O',KARP
                MOV     DPTR,#DT_O
                RET
KARP:           CJNE    A,#'P',KARQ
                MOV     DPTR,#DT_P
                RET
KARQ:           CJNE    A,#'Q',KARR
                MOV     DPTR,#DT_Q
                RET
KARR:           CJNE    A,#'R',KARS
                MOV     DPTR,#DT_R
                RET
KARS:           CJNE    A,#'S',KART
                MOV     DPTR,#DT_S
                RET
KART:           CJNE    A,#'T',KARU
                MOV     DPTR,#DT_T
                RET
KARU:           CJNE    A,#'U',KARV
                MOV     DPTR,#DT_U
                RET
KARV:           CJNE    A,#'V',KARW
                MOV     DPTR,#DT_V
                RET
KARW:           CJNE    A,#'W',KARX
                MOV     DPTR,#DT_W
                RET
KARX:           CJNE    A,#'X',KARY
                MOV     DPTR,#DT_X
                RET
KARY:           CJNE    A,#'Y',KARZ
                MOV     DPTR,#DT_Y
                RET
KARZ:           CJNE    A,#'Z',KAR_AA
                MOV     DPTR,#DT_Z
                RET
;---------- ---------- ---------- ----------
KAR_AA:         CJNE    A,#'a',KAR_BB
                MOV     DPTR,#DT_AA
                RET
KAR_BB:         CJNE    A,#'b',KAR_CC
                MOV     DPTR,#DT_BB
                RET
KAR_CC:         CJNE    A,#'c',KAR_DD
                MOV     DPTR,#DT_CC
                RET
```

```
KAR_DD:     CJNE  A,#'d',KAR_EE
            MOV   DPTR,#DT_DD
            RET
KAR_EE:     CJNE  A,#'e',KAR_FF
            MOV   DPTR,#DT_EE
            RET
KAR_FF:     CJNE  A,#'f',KAR_GG
            MOV   DPTR,#DT_FF
            RET
KAR_GG:     CJNE  A,#'g',KAR_HH
            MOV   DPTR,#DT_GG
            RET
KAR_HH:     CJNE  A,#'h',KAR_II
            MOV   DPTR,#DT_HH
            RET
KAR_II:     CJNE  A,#'i',KAR_JJ
            MOV   DPTR,#DT_II
            RET
KAR_JJ:     CJNE  A,#'j',KAR_KK
            MOV   DPTR,#DT_JJ
            RET
KAR_KK:     CJNE  A,#'k',KAR_LL
            MOV   DPTR,#DT_KK
            RET
KAR_LL:     CJNE  A,#'l',KAR_MM
            MOV   DPTR,#DT_LL
            RET
KAR_MM:     CJNE  A,#'m',KAR_NN
            MOV   DPTR,#DT_MM
            RET
KAR_NN:     CJNE  A,#'n',KAR_OO
            MOV   DPTR,#DT_NN
            RET
KAR_OO:     CJNE  A,#'o',KAR_PP
            MOV   DPTR,#DT_OO
            RET
KAR_PP:     CJNE  A,#'p',KAR_QQ
            MOV   DPTR,#DT_PP
            RET
KAR_QQ:     CJNE  A,#'q',KAR_RR
            MOV   DPTR,#DT_QQ
            RET
KAR_RR:     CJNE  A,#'r',KAR_SS
            MOV   DPTR,#DT_RR
            RET
KAR_SS:     CJNE  A,#'s',KAR_TT
            MOV   DPTR,#DT_SS
            RET
KAR_TT:     CJNE  A,#'t',KAR_UU
            MOV   DPTR,#DT_TT
            RET
KAR_UU:     CJNE  A,#'u',KAR_VV
            MOV   DPTR,#DT_UU
            RET
KAR_VV:     CJNE  A,#'v',KAR_WW
            MOV   DPTR,#DT_VV
```

```
                RET
KAR_WW:         CJNE    A,#'w',KAR_XX
                MOV     DPTR,#DT_WW
                RET
KAR_XX:         CJNE    A,#'x',KAR_YY
                MOV     DPTR,#DT_XX
                RET
KAR_YY:         CJNE    A,#'y',KAR_ZZ
                MOV     DPTR,#DT_YY
                RET
KAR_ZZ:         CJNE    A,#'z',KAR0
                MOV     DPTR,#DT_ZZ
                RET
;---------- ---------- ----------
KAR0:           CJNE    A,#'0',KAR1
                MOV     DPTR,#DT_0
                RET
KAR1:           CJNE    A,#'1',KAR2
                MOV     DPTR,#DT_1
                RET
KAR2:           CJNE    A,#'2',KAR3
                MOV     DPTR,#DT_2
                RET
KAR3:           CJNE    A,#'3',KAR4
                MOV     DPTR,#DT_3
                RET
KAR4:           CJNE    A,#'4',KAR5
                MOV     DPTR,#DT_4
                RET
KAR5:           CJNE    A,#'5',KAR6
                MOV     DPTR,#DT_5
                RET
KAR6:           CJNE    A,#'6',KAR7
                MOV     DPTR,#DT_6
                RET
KAR7:           CJNE    A,#'7',KAR8
                MOV     DPTR,#DT_7
                RET
KAR8:           CJNE    A,#'8',KAR9
                MOV     DPTR,#DT_8
                RET
KAR9:           CJNE    A,#'9',KARSE
                MOV     DPTR,#DT_9
                RET
KARSE:          CJNE    A,#'!',KARPLUS
                MOV     DPTR,#K_SE
                RET
KARPLUS:        CJNE    A,#'+',KARMIN
                MOV     DPTR,#K_PLS
                RET
KARMIN:         CJNE    A,#'-',KARGR
                MOV     DPTR,#K_MIN
                RET
KARGR:          CJNE    A,#' ',KARBT
                MOV     DPTR,#K_GRB
                RET
```

```
KARBT:      CJNE  A,#'*',KARTK
            MOV   DPTR,#K_BNTNG
            RET
KARTK:      CJNE  A,#'.',KART2
            MOV   DPTR,#K_TTK
            RET
KART2:      CJNE  A,#':',KARPG
            MOV   DPTR,#K_TTK2
            RET
KARPG:      CJNE  A,#'{',BUKUR
            MOV   DPTR,#K_PGR
            RET
BUKUR:      CJNE  A,#'[',TUKUR
            MOV   DPTR,#K_BUKUR
            RET
TUKUR:      CJNE  A,#']',BUKURA
            MOV   DPTR,#K_TUKUR
            RET
BUKURA:     CJNE  A,#'{',TUKURA
            MOV   DPTR,#K_BUKURA
            RET
TUKURA:     CJNE  A,#'|',LESAR
            MOV   DPTR,#K_TUKURA
            RET
LESAR:      CJNE  A,#'>',LECIL
            MOV   DPTR,#K_LESAR
            RET
LECIL:      CJNE  A,#'<',GRNG
            MOV   DPTR,#K_LECIL
            RET
GRNG:       CJNE  A,#'/',SLASH
            MOV   DPTR,#K_GRNG
            RET
SLASH:      CJNE  A,#'\',ASSTR
            MOV   DPTR,#K_SLSH
            RET
ASSTR:      CJNE  A,#'@',DOLAR
            MOV   DPTR,#K_ASSTR
            RET
DOLAR:      CJNE  A,#'$',PETIK
            MOV   DPTR,#K_DLR
            RET
PETIK:      CJNE  A,#'^',DAN
            MOV   DPTR,#K_PTK
            RET
DAN:        CJNE  A,#'&',PERSEN
            MOV   DPTR,#K_DAN
            RET
PERSEN:     CJNE  A,#25H,GAR2
            MOV   DPTR,#K_PRSN
            RET
GAR2:       CJNE  A,#'|',KOMA
            MOV   DPTR,#K_GR2
            RET
KOMA:       CJNE  A,#2CH,KMA_A
            MOV   DPTR,#K_KMA
```

```
            RET
KMA_A:      CJNE    A,#27H,NOKAR
            MOV     DPTR,#K_KMA_A
            RET
NOKAR:      MOV     DPTR,#K_SPC
            RET
;-----------------------------------------
;     DATA UNTUK MEMBENTUK KARAKTER
;-----------------------------------------
K_BUKUR:    DW      0F007H          ;1111 0000 0000|0111
            DW      0EFFBH          ;1110|1111|1111|1011
            DW      0DFFDH          ;1101|1111|1111|1101
            DW      0BFFEH          ;1011|1111|1111|1110
            DB      24H

K_TUKUR:    DW      0BFFEH          ;1011|1111|1111|1110
            DW      0DFFDH          ;1101|1111|1111|1101
            DW      0EFFBH          ;1110|1111|1111|1011
            DW      0F007H          ;1111|0000|0000|0111
            DB      24H

K_BUKURA:   DW      0FF7FH          ;1111 1111|0111|1111
            DW      0FEBFH          ;1111 1110|1011|1111
            DW      0E1C3H          ;1110|0001|1100|0011
            DW      0DFFDH          ;1101111111|1111|1101
            DW      0BFFEH          ;1011|1111|1111|1110
            DW      0BFFEH          ;1011|1111|1111|1110
            DB      24H

K_TUKURA:   DW      0BFFEH          ;1011|1111|1111|1110
            DW      0BFFEH          ;1011|1111 1111|1110
            DW      0DFFDH          ;1101 1111|1111|1101
            DW      0E1C3H          ;1110|0001|1100|0011
            DW      0FEBFH          ;1111|1110|1011|1111
            DW      0FF7FH          ;1111|1111|0111|1111
            DB      24H

K_DECIL:    DW      0FE3FH          ;1111|1110|0011|1111
            DW      0FDDFH          ;1111|1101|1101 1111
            DW      0FBEFH          ;1111|1011 1110|1111
            DW      0F7F7H          ;1111 0111|1111|0111
            DW      0EFFBH          ;1110 1111|1111|1011
            DW      0DFFDH          ;1101|1111|1111|1101
            DW      0BFFEH          ;1011|1111|1111|1110
            DB      24H

K_IPSAR:    DW      0BFFEH          ;1011|1111|1111 1110
            DW      0DFFDH          ;1101|1111|1111|1101
            DW      0EFFBH          ;1110|1111 1111|1011
            DW      0F7F7H          ;1111 0111|1111|0111
            DW      0FBEFH          ;1111|1011|1110|1111
            DW      0FDDFH          ;1111|1101|1101|1111
            DW      0FE3FH          ;1111|1110|0011|1111
            DB      24H

K_TTK2:     DW      0FFFFH          ;1111|1111|1111|1111
```

```
                DW      0CFF9H          ;1100|1111|1111|1001
                DW      0CFF9H          ;1100|1111|1111|1001
                DW      0FFFFH          ;1111|1111|1111|1111
                DB      24H

K_GRNG:         DW      0BFFFH          ;1011 1111 1111|1111
                DW      0DFFFH          ;1101 1111 1111|1111
                DW      0EFFFH          ;1110|1111|1111|1111
                DW      0F7FFH          ;1111|0111|1111|1111
                DW      0FBFFH          ;1111|1011|1111|1111
                DW      0FDFFH          ;1111|1101|1111|1111
                DW      0FEFFH          ;1111|1110|1111|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FFBFH          ;1111|1111|1011|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFEFH          ;1111|1111|1110 1111
                DW      0FFF7H          ;1111|1111|1111|0111
                DW      0FFFBH          ;1111|1111|1111|1011
                DW      0FFFDH          ;1111|1111|1111|1101
                DW      0FFFEH          ;1111|1111|1111|1110
                DB      24H

K_SLSH:         DW      0FFFEH          ;1111|1111|1111|1110
                DW      0FFFDH          ;1111|1111|1111|1101
                DW      0FFFBH          ;1111|1111|1111|1011
                DW      0FFF7H          ;1111|1111|1111|0111
                DW      0FFEFH          ;1111|1111|1110|1111
                DW      0FFDFH          ;1111 1111 1101|1111
                DW      0FFBFH          ;1111|1111|1011|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FEFFH          ;1111|1110|1111|1111
                DW      0FDFFH          ;1111|1101|1111|1111
                DW      0FBFFH          ;1111|1011|1111|1111
                DW      0F7FFH          ;1111|0111|1111|1111
                DW      0EFFFH          ;1110|1111|1111 1111
                DW      0DFFFH          ;1101|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DB      24H

K_GRB:          DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DW      0FFFFH          ;1011|1111|1111|1111
                DB      24H

K_SPC:          DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
                DW      0FFFFH          ;1111|1111|1111|1111
```

```
                DB      24H

K_SR:           DW      0FFFFH          ;1111 1111 1111|1111
                DW      09C00H          ;1001 1100 0000|0000
                DW      0FFFFH          ;1111|1111|1111|1111
                DB      24H

K_PGR:          DW      0FBBFH          ;1111|1011|1011|1111
                DW      0FBBFH          ;1111|1011|1011|1111
                DW      0CCC7H          ;1100|0000|0000|0111
                DW      0FBBFH          ;1111|1011|1011|1111
                DW      0FBBFH          ;1111|1011|1011|1111
                DW      0FBBFH          ;1111|1011|1011|1111
                DW      0C007H          ;1100|0000|0000|0111
                DW      0FBBFH          ;1111|1011|1011|1111
                DW      0FBBFH          ;1111|1011|1011|1111
                DB      24H

K_BNTNG:        DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FB6FH          ;1111|1011|0110|1111
                DW      0FD5FH          ;1111|1101|0101|1111
                DW      0FE3FH          ;1111|1110|0011|1111
                DW      0E003H          ;1110|0000|0000|0011
                DW      0FE3FH          ;1111|1110|0011|1111
                DW      0FD5FH          ;1111|1101|0101|1111
                DW      0FB6FH          ;1111|1011|0110|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DB      24H

K_PLS:          DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0E003H          ;1110|0000|0000|0011
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DB      24H

K_PTK:          DW      0FFF7H          ;1111|1111|1111|0111
                DW      0FFFBH          ;1111|1111|1111|1011
                DW      0FFFDH          ;1111|1111|1111|1101
                DW      0FFFFH          ;1111|1111|1111|1110
                DW      0FFFDH          ;1111|1111|1111|1101
                DW      0FFFBH          ;1111|1111|1111|1011
                DW      0FFF7H          ;1111|1111|1111|0111
                DB      24H

K_ASSTR:        DW      0E003H          ;1110|0000|0000|0011
                DW      0DFFDH          ;1101|1111|1111|1101
                DW      0BFFEH          ;1011|1111|1111|1110
                DW      0BFFEH          ;1011|1111|1111|1110
                DW      0B81EH          ;1011|1000|0001|1110
                DW      0B99EH          ;1011|0000|0000|1110
                DW      0B00EH          ;1011|0000|0000|1110
                DW      0D10DH          ;1101|1000|0000|1101
                DW      0FC03H          ;1111|1100|0000|0011
```

```
                DB      24H

K_DLR:          DW      0F88FH          ;1111|1011|1000|1111
                DW      0F777H          ;1111|0111|0111|0111
                DW      0F777H          ;1111|0111|0111|0111
                DW      0F777H          ;1111|0111|0111|0111
                DW      08770H          ;1000|0111|0111|0000
                DW      0F777H          ;1111|0111|0111|0111
                DW      0F777H          ;1111|0111|0111|0111
                DW      0F777H          ;1111|0111|0111|0111
                DW      0F96FH          ;1111|1000|1110|1111
                DB      24H

K_DAN:          DW      0E1FFH          ;1110|0001|1111|1111
                DW      0DEC3H          ;1101|1110|1100|0011
                DW      0BF3DH          ;1011|1111|0011|1101
                DW      0BF7EH          ;1011|1111|0111|1110
                DW      0BE7EH          ;1011|1110|0111|1110
                DW      0BD7EH          ;1011|1101|0111|1110
                DW      0BB7EH          ;1011|1011|0111|1110
                DW      0D7BDH          ;1101|0111|1011|1101
                DW      0C3C3H          ;1100|0011|1100|0011
                DW      0B7FFH          ;1011|0111|1111|1111
                DB      24H

K_PRSN:         DW      0DFFFH          ;1011|1111|1111|1111
                DW      0DFF0H          ;1101|1111|1111|0000
                DW      0EF60H          ;1110|1111|1111|0110
                DW      0F760H          ;1111|0111|1111|0110
                DW      0FB00H          ;1111|1011|1111|0000
                DW      0FDFFH          ;1111|1101|1111|1111
                DW      0FEFFH          ;1111|1110|1111|1111
                DW      0FF7FH          ;1111|1111|0111|1111
                DW      0FFBFH          ;1111|1111|1011|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      08EEFH          ;1000|0111|1110|1111
                DW      0BEF7H          ;1011|0111|1111|0111
                DW      0BEFBH          ;1011|0111|1111|1011
                DW      08EFDH          ;1000|0111|1111|1101
                DW      0FFFEH          ;1111|1111|1111|1110
                DB      24H

K_GRZ:          DW      0FFFFH          ;1111|1111|1111|1111
                DW      081C0H          ;1000|0001|1100|0000
                DW      0FFFFH          ;1111|1111|1111|1111
                DB      24H

K_KMA:          DW      0A7FFH          ;1010|0111|1111|1111
                DW      0A7EFH          ;1010|0111|1111|1110
                DW      0C7EFH          ;1100|0111|1111|1110
                DB      24H

K_KMA_A:        DW      0FFFCH          ;1111|1111|1111|1100
                DW      0FFF4H          ;1111|1111|1111|0100
                DW      0FFF8H          ;1111|1111|1111|1000
                DB      24H
```

```
K_MIN:      DW    0FF7FH          ;1111|1111|0111|1111
            DW    0FF7FH          ;1111|1111|0111|1111
            DW    0FF7FH          ;1111|1111|0111|1111
            DW    0FF7FH          ;1111|1111|0111|1111
            DW    0FF7FH          ;1111|1111|0111|1111
            DB    24H

K_TIK:      DW    09FFFH          ;1001|1111|1111|1111
            DW    09FFFH          ;1001|1111|1111|1111
            DB    24H
;-----------------------------------------------
;     KARAKTER HURUF
;-----------------------------------------------
DT_A:       DW    C8003H          ;1000|0000|0000|0011    ;A
            DW    CFF7DH          ;1111|1111|0111|1101
            DW    CFF7EH          ;1111|1111|0111|1110
            DW    CFF7EH          ;1111|1111|0111|1110
            DW    CFF7EH          ;1111|1111|0111|1110
            DW    CFF7EH          ;1111|1111|0111|1110
            DW    CFF7EH          ;1111|1111|0111|1110
            DW    CFF7EH          ;1111|1111|0111|1110
            DW    CFF7DH          ;1111|1111|0111|1101
            DW    C8003H          ;1000|0000|0000|0011
            DB    24H

DT_B:       DW    C8000H          ;1000|0000|0000|0000    ;B
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CBF7EH          ;1011|1111|0111|1110
            DW    CDEBDH          ;1101|1110|1011|1101
            DW    CE1C3H          ;1110|0001|1100|0011
            DB    24H

DT_C:       DW    CF007H          ;1111|0000|0000|0111    ;C
            DW    CEFFBH          ;1110|1111|1111|1011
            DW    CDFFDH          ;1101|1111|1111|1101
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CDFFDH          ;1101|1111|1111|1101
            DW    CEFFBH          ;1110|1111|1111|1011
            DB    24H

DT_D:       DW    C8000H          ;1000|0000|0000|0000    ;D
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
            DW    CBFFEH          ;1011|1111|1111|1110
```

```
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0DFFDH          ;1101|1111|1111|1101
        DW      0EFFBH          ;1110|1111|1111|1011
        DW      0F007H          ;1111|0000|0000|0111
        DB      24H

DT_E:   DW      08000H          ;1000|0000|0000|0000    ;E
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DB      24H

DT_F:   DW      08000H          ;1000|0000|0000|0000
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DB      24H

DT_G:   DW      0F007H          ;1111|0000|0000|0111
        DW      0EFFBH          ;1110|1111|1111|1011
        DW      0DFFDH          ;1101|1111|1111|1101
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF73H          ;1011|1111|0111|1110
        DW      0BF73H          ;1011|1111|0111|1110
        DW      0DF73H          ;1101|1111|0111|1101
        DW      0E073H          ;1110|0000|0111|0011
        DB      24H

DT_H:   DW      08000H          ;1000|0000|0000|0000
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      08000H          ;1000|0000|0000|0000
        DB      24H

DT_I:   DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      08000H          ;1000|0000|0000|0000
        DW      0BFFEH          ;1011|1111|1111|1110
```

```
        DW      0BFFBH          ;1011|1111|1111|1110
        DB      24H

DT_J:   DW      CFFFFH          ;1110|1111|1111|1111
        DW      0DFFEH          ;1101|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0DFFEH          ;1101|1111|1111|1110
        DW      0E000H          ;1110|0000|0000|0000
        DB      24H

DT_K:   DW      C0000H          ;1000|0000|0000|0000
        DW      CFF7FH          ;1111|1111|0111|1111
        DW      CFEBFH          ;1111|1110|1011|1111
        DW      CFDDFH          ;1111|1101|1101|1111
        DW      CFBEFH          ;1111|1011|1110|1111
        DW      CF7FFH          ;1111|0111|1111|1111
        DW      CEFFBH          ;1110|1111|1111|1011
        DW      CDFFDH          ;1101|1111|1111|1101
        DW      CBFFEH          ;1011|1111|1111|1110
        DB      24H

DT_L:   DW      00000H          ;0000|0000|0000|0000
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DB      24H

DT_M:   DW      C0000H          ;1000|0000|0000|0000
        DW      0FFFDH          ;1111|1111|1111|1101
        DW      0FFFBH          ;1111|1111|1111|1011
        DW      0FFF7H          ;1111|1111|1111|0111
        DW      0FFEFH          ;1111|1111|1110|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFBFH          ;1111|1111|1011|1111
        DW      0FF7FH          ;1111|1111|0111|1111
        DW      0FEFFH          ;1111|1110|1111|1111
        DW      0FDFFH          ;1111|1101|1111|1111
        DW      0FBFFH          ;1111|1011|1111|1111
        DW      0F7FFH          ;1111|0111|1111|1111
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DB      24H

DT_N:   DW      C0000H          ;1000|0000|0000|0000
        DW      0FFFDH          ;1111|1111|1111|1101
        DW      0FFFBH          ;1111|1111|1111|1011
        DW      0FFF7H          ;1111|1111|1111|0111
        DW      0FFEFH          ;1111|1111|1110|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFBFH          ;1111|1111|1011|1111
        DW      0FF7FH          ;1111|1111|0111|1111
        DW      0FEFFH          ;1111|1110|1111|1111
```

```
        DW      08000H          ;1000|0000|0000|0000
        DB      24H

DT_O:   DW      0F007H          ;1111|0000|0000|0111
        DW      0EFFBH          ;1110|1111|1111|1011
        DW      0DFFDH          ;1101|1111|1111|1101
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0DFFDH          ;1101|1111|1111|1101
        DW      0EFFBH          ;1110|1111|1111|1011
        DW      0F007H          ;1111|0000|0000|0111
        DB      24H

DT_P:   DW      08000H          ;1000|0000|0000|0000
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FFBEH          ;1111|1111|1011|1110
        DW      0FFDDH          ;1111|1111|1011|1101
        DW      0FFE3H          ;1111|1111|1100|0011
        DB      24H

DT_Q:   DW      0F007H          ;1111|0000|0000|0111
        DW      0EFFBH          ;1110|1111|1111|1011
        DW      0DFFDH          ;1101|1111|1111|1101
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BEFEH          ;1011|1011|1111|1110
        DW      0D7FDH          ;1101|0111|1111|1101
        DW      0EFFBH          ;1110|1111|1111|1011
        DW      0D007H          ;1101|0000|0000|0111
        DB      24H

DT_R:   DW      08000H          ;1000|0000|0000|0000
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FF7EH          ;1111|1111|0111|1110
        DW      0FE7EH          ;1111|1110|0111|1110
        DW      0FD7EH          ;1111|1101|0111|1110
        DW      0FB7EH          ;1111|1011|0111|1110
        DW      0F77EH          ;1111|0111|0111|1110
        DW      0EFBEH          ;1110|1111|1011|1110
        DW      0DFDDH          ;1101|1111|1101|1101
        DW      0BFE3H          ;1011|1111|1100|0011
        DB      24H

DT_S:   DW      0DFC3H          ;1101|1111|1100|0011
        DW      0BFBDH          ;1011|1111|1011|1101
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
```

```
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0DEFEH          ;1101|1110|1111|1110
        DW      0E1FDH          ;1110|0001|1111|1101
        DB      24H

DI_V:   DW      0FFFEH          ;1111|1111|1111|1110
        DW      0FFFEH          ;1111|1111|1111|1110
        DW      0FFFEH          ;1111|1111|1111|1110
        DW      0FFFEH          ;1111|1111|1111|1110
        DW      08000H          ;1000|0000|0000|0000
        DW      0FFFFH          ;1111|1111|1111|1111
        DW      0FFFFH          ;1111|1111|1111|1111
        DW      0FFFEH          ;1111|1111|1111|1110
        DW      0FFFEH          ;1111|1111|1111|1110
        DB      24H

DI_U:   DW      0F000H          ;1111|0000|0000|0000
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0F000H          ;1111|0000|0000|0000
        DB      24H

DI_V:   DW      0FC00H          ;1111|1100|0000|0000
        DW      0FBFFH          ;1111|1011|1111|1111
        DW      0F7FFH          ;1111|0111|1111|1111
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0F7FFH          ;1111|0111|1111|1111
        DW      0F800H          ;1111|1000|0000|0000
        DB      24H

DI_W:   DW      08000H          ;1000|0000|0000|0000
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0F7FFH          ;1111|0111|1111|1111
        DW      0FBFFH          ;1111|1011|1111|1111
        DW      0FDFFH          ;1111|1101|1111|1111
        DW      0FBFFH          ;1111|1011|1111|1111
        DW      0F7FFH          ;1111|0111|1111|1111
        DW      0EFFFH          ;1110|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      08000H          ;1000|0000|0000|0000
        DB      24H
```

```
DT_X:   DW   08FF8H          ;1000|1111|1111|1000
        DW   0F7F7H          ;1111|0111|1111|0111
        DW   0FBEFH          ;1111|1011|1110|1111
        DW   0FDDFH          ;1111|1101|1101|1111
        DW   0FEBFH          ;1111|1110|1011|1111
        DW   0FF7FH          ;1111|1111|0111|1111
        DW   0FEBFH          ;1111|1110|1011|1111
        DW   0FDDFH          ;1111|1101|1101|1111
        DW   0FBEFH          ;1111|1011|1110|1111
        DW   0F7F7H          ;1111|0111|1111|0111
        DW   08FF8H          ;1000|1111|1111|1000
        DB   24H

DT_Y:   DW   0FFF8H          ;1111|1111|1111|1000
        DW   0FFF7H          ;1111|1111|1111|0111
        DW   0FFEFH          ;1111|1111|1110|1111
        DW   0FFDFH          ;1111|1111|1101|1111
        DW   0FFBFH          ;1111|1111|1011|1111
        DW   0807FH          ;1000|0000|0111 1111
        DW   0FFBFH          ;1111|1111|1011|1111
        DW   0FFDFH          ;1111 1111 1101|1111
        DW   0FFEFH          ;1111|1111|1110|1111
        DW   0FFF7H          ;1111|1111|1111|0111
        DW   0FFF8H          ;1111|1111|1111|1000
        DB   24H

DT_Z:   DW   08FFFH          ;1000|1111|1111 1110
        DW   0B7FFH          ;1011|0111|1111|1110
        DW   0BBFFH          ;1011 1011|1111|1110
        DW   0BDFFH          ;1011|1101|1111|1110
        DW   0BEFFH          ;1011|1110|1111|1110
        DW   0BF7FH          ;1011|1111|0111|1110
        DW   0BFBFH          ;1011|1111|1011|1110
        DW   0BFDFH          ;1011|1111|1101|1110
        DW   0BFEFH          ;1011|1111|1110|1110
        DW   0BFF7H          ;1011 1111 1111|0110
        DW   0BFF8H          ;1011|1111|1111|1000
        DB   24H
;-------- -------- -------- ---------- --
DT_AA:  DW   0C3BFH          ;1100|0011|1011|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0BDDFH          ;1011 1101 1101|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0BDDFH          ;1011|1101|1101|1111
        DW   0C03FH          ;1100|0000|0011 1111
        DB   24H

DT_BB:  DW   08000H          ;1000|0000|0000|0000
        DW   0BFDFH          ;1011|1111|1101|1111
        DW   0BFDFH          ;1011|1111|1101|1111
        DW   0BFDFH          ;1011|1111|1101|1111
        DW   0BFDFH          ;1011|1111|1101|1111
```

```
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0DFBFH          ;1101|1111|1011|1111
        DW      0E07FH          ;1110|0000|0111|1111
        DB      24H

DT_CC:  DW      0E07FH          ;1110|0000|0111|1111
        DW      0DFBFH          ;1101|1111|1011|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0C03FH          ;1100|0000|0011|1111
        DB      24H

DT_DD:  DW      0E07FH          ;1110|0000|0111|1111
        DW      0DFBFH          ;1101|1111|1011|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      0BFDFH          ;1011|1111|1101|1111
        DW      08000H          ;1000|0000|0000|0000
        DB      24H

DT_EE:  DW      0C03FH          ;1100|0000|0011|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0DE3FH          ;1101|1110|0011|1111
        DB      24H

DT_FF:  DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0800IH          ;1000|0000|0000|0001
        DW      0FFDEH          ;1111|1111|1101|1110
        DW      0FFDEH          ;1111|1111|1101|1110
        DW      0FFDEH          ;1111|1111|1101|1110
        DW      0FFDEH          ;1111|1111|1101|1110
        DB      24H

DT_GG:  DW      0DE3FH          ;1101|1110|0011|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDEH          ;1011|1101|1101|1111
```

```
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0BDDFH          ;1011|1101|1101|1111
        DW      0C03FH          ;1100|0000|0011|1111
        DB      24H

DT_HH:  DW      C80C0H          ;1000|0000|0000|0000
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFBFH          ;1111|1111|1011|1111
        DW      C807FH          ;1000|0000|0111|1111
        DB      24H

DT_II:  DW      0BFDCH          ;1011|1111|1101|1100
        DW      0801CH          ;1000|0000|0001|1100
        DW      0BFFFH          ;1011|1111|1111|1111
        DB      24H

DT_JJ:  DW      0EFFFH          ;1110|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0DFFFH          ;1101|1111|1111|1111
        DW      0E000H          ;1110|0000|0000|0000
        DB      24H

DT_KK:  DW      08000H          ;1000|0000|0000|0000
        DW      0FDFFH          ;1111|1101|1111|1111
        DW      0FAFFH          ;1111|1010|1111|1111
        DW      0F77FH          ;1111|0111|0111|1111
        DW      0EFBFH          ;1110|1111|1011|1111
        DW      0DFDFH          ;1101|1111|1101|1111
        DW      0BFEFH          ;1011|1111|1110|1111
        DB      24H

DT_LL:  DW      08000H          ;1000|0000|0000|0000
        DW      0BFFFH          ;1011|1111|1111|1111
        DW      0BFFFH          ;1011|1111|1111|1111
        DB      24H

DT_MM:  DW      0801FH          ;1000|0000|0001|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      C803FH          ;1000|0000|0011|1111
        DW      0FFDFH          ;1111|1111|1101|1111
        DW      0FFDFH          ;1111|1111|1101|1111
```

```
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0803FH          ;1000|0000|0011|1111
                DB      24H

DT_NN:          DW      0801FH          ;1000|0000|0001|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0803FH          ;1000|0000|0011|1111
                DB      24H

DT_OO:          DW      0E07FH          ;1110|0000|0111|1111
                DW      0DF3FH          ;1101|1111|1011|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0DFBFH          ;1101|1111|1011|1111
                DW      0E07FH          ;1110|0000|0111|1111
                DB      24H

DT_PP:          DW      0801FH          ;1000|0000|0001|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FBDFH          ;1111|1011|1101|1111
                DW      0FDBFH          ;1111|1101|1011|1111
                DW      0FE7FH          ;1111|1110|0111|1111
                DB      24H

DT_QQ:          DW      0E07FH          ;1110|0000|0111|1111
                DW      0FBBFH          ;1111|1011|1011|1111
                DW      0F7DFH          ;1111|0111|1101|1111
                DW      0F7DFH          ;1111|0111|1101|1111
                DW      0F7DFH          ;1111|0111|1101|1111
                DW      0F7DFH          ;1111|0111|1101|1111
                DW      0F7DFH          ;1111|0111|1101|1111
                DW      0F7DFH          ;1111|0111|1101|1111
                DW      0F7BFH          ;1111|0111|1011|1111
                DW      0807FH          ;1000|0000|0111|1111
                DB      24H

DT_RR:          DW      0801FH          ;1000|0000|0001|1111
                DW      0FE7FH          ;1111|1110|0111|1111
                DW      0FFBFH          ;1111|1111|1011|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
```

```
                DW      0FFDFH          ;1111 1111|1101|1111
                DB      24H

DT_SS:          DW      0DE3FH          ;1101|1110|0011|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0C3BFH          ;1100|0011|1011|1111
                DB      24H

DT_TT:          DW      0FFDFH          ;1111|1111|1101|1111
                DW      0FFDFH          ;1111|1111|1101|1111
                DW      0C000H          ;1100|0000|0000|0000
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DB      24H

DT_UU:          DW      0E01FH          ;1110|0000|0001|1111
                DW      0DFFFH          ;1101|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0DFFFH          ;1101|1111|1111|1111
                DW      0E01FH          ;1110|0000|0001|1111
                DB      24H

DT_VV:          DW      0F81FH          ;1111|1000|0001|1111
                DW      0F7EFH          ;1111|0111|1110|1111
                DW      0EFF7H          ;1110|1111|1111|0111
                DW      0DFFBH          ;1101|1111|1111|1011
                DW      0BFFDH          ;1011|1111|1111|1101
                DW      0DFFBH          ;1101|1111|1111|1011
                DW      0EFF7H          ;1110|1111|1111|0111
                DW      0F7EFH          ;1111|0111|1110|1111
                DW      0F81FH          ;1111|1000|0001|1111
                DB      24H

DT_WW:          DW      0E01FH          ;1110|0000|0001|1111
                DW      0DFFFH          ;1101|1111|1111|1111
                DW      0BFF7H          ;1011|1111|1111|0111
                DW      0DFEFH          ;1101|1111|1110|1111
                DW      0E03FH          ;1110|0000|0011|1111
                DW      0DFEFH          ;1101|1111|1110|1111
                DW      0BFF7H          ;1011|1111|1111|0111
                DW      0DFFFH          ;1101|1111|1111|1111
                DW      0E01FH          ;1110|0000|0001|1111
```

```
                DB      24H

DT_XX:          DW      0BF0FH          ;1011|1111|1101|1111
                DW      0DFBFH          ;1101|1111|1011|1111
                DW      0EF7FH          ;1110|1111|0111|1111
                DW      0F6FFH          ;1111|0110|1111|1111
                DW      0F9FFH          ;1111|1001|1111|1111
                DW      0F6FFH          ;1111|0110|1111|1111
                DW      0EF7FH          ;1110|1111|0111|1111
                DW      0D7BFH          ;1101|1111|1011|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DB      24H

DT_YY:          DW      0DE1FH          ;1101|1110|0001|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0BDFFH          ;1011|1101|1111|1111
                DW      0C01FH          ;1100|0000|0001|1111
                DB      24H

DT_ZZ:          DW      0BFDFH          ;1011|1111|1101|1111
                DW      09FDFH          ;1001|1111|1101|1111
                DW      0AFDFH          ;1010|1111|1101|1111
                DW      0B7DFH          ;1011|0111|1101|1111
                DW      0BBDFH          ;1011|1011|1101|1111
                DW      0BDDFH          ;1011|1101|1101|1111
                DW      0BEDFH          ;1011|1110|1101|1111
                DW      0BF5FH          ;1011|1111|0101|1111
                DW      0BF9FH          ;1011|1111|1001|1111
                DW      0BFDFH          ;1011|1111|1101|1111
                DB      24H
;---------  ---------  ----------  ------
;       KARAKTER ANGKA
;---------  ---------  ----------  ------
DT_0:           DW      0F007H          ;1111|0000|0000|0111
                DW      0E0BFH          ;1110|1011|1111|1011
                DW      0DDFDH          ;1101|1101|1111|1101
                DW      0BEFEH          ;1011|1110|1111|1110
                DW      0BF7FH          ;1011|1111|0111|1110
                DW      0BFBFH          ;1011|1111|1011|1110
                DW      0BFDFH          ;1011|1111|1101|1110
                DW      0BFEFH          ;1011|1111|1110|1110
                DW      0D1E5H          ;1101|0001|1110|0101
                DW      0EFFBH          ;1110|1111|1111|1011
                DW      0F007H          ;1111|0000|0000|0111
                DB      24H

DT_1:           DW      0BFFBH          ;1011|1111|1111|1011
                DW      0BFFDH          ;1011|1111|1111|1101
                DW      08000H          ;1000|0000|0000|0000
                DW      0BFFFH          ;1011|1111|1111|1111
                DW      0BFFFH          ;1011|1111|1111|1111
```

```
        DB      24H

DT_2:   DW      09FF7H          ;1001 1111|1111|0111
        DW      0AFFBH          ;1010|1111|1111|1011
        DW      0B7FDH          ;1011|0111|1111|1101
        DW      0BBFEH          ;1011|1011|1111|1110
        DW      0BDFBH          ;1011|1101|1111|1110
        DW      0BEFBH          ;1011|1110|1111|1110
        DW      0BF73H          ;1011|1111|0111|1110
        DW      0BF3DH          ;1011|1111|1011|1101
        DW      0BFC3H          ;1011|1111|1100|0011
        DB      24H

DT_3:   DW      0BFFEH          ;1101|1111|1111|1101
        DW      0BFFEH          ;1011|1111|1111|1110
        DW      0BFFBH          ;1011|1111|1111|1110
        DW      0BF7BH          ;1011|1111|0111|1110
        DW      0BF7BH          ;1011|1111|0111|1110
        DW      0BF7BH          ;1011|1111|0111|1110
        DW      0BF7BH          ;1011|1111|0111|1110
        DW      0BEBBH          ;1011|1110|1011|1110
        DW      0DDDDH          ;1101|1101|1101|1101
        DW      0E3E3H          ;1110|0011|1110|0011
        DB      24H

DT_4:   DW      0FE7FH          ;1111|1110|0111|1111
        DW      0FEBFH          ;1111|1110|1011|1111
        DW      0FEDFH          ;1111|1110|1101|1111
        DW      0FEEFH          ;1111|1110|1110|1111
        DW      0FEF7H          ;1111|1110|1111|0111
        DW      0FEFBH          ;1111|1110|1111|1011
        DW      0FEFDH          ;1111|1110|1111|1101
        DW      0FEFEH          ;1111|1110|1111|1110
        DW      0800CH          ;1000|0000|0000|0000
        DW      0FE7FH          ;1111|1110|1111|1___
        DB      24H

DT_5:   DW      0DF00H          ;1101|1111|0000|0000
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0___|1110
        DW      0BF7EH          ;1011|1111|0___|1110
        DW      0BF7EH          ;1011 1111 0111 1110
        DW      0BF7EH          ;1011|1111 0111 1110
        DW      0DF7EH          ;1101|1111 0111|1110
        DW      0EEFEH          ;1110|1110 1111|1110
        DW      0F1FEH          ;1111|0001|1111|1110
        DB      24H

DT_6:   DW      0F007H          ;1111 0000|0000|0111
        DW      0EF7BH          ;1110 1111|0111 1011
        DW      0DF7DH          ;1101 1111|0111|1101
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
        DW      0BF7EH          ;1011|1111|0111|1110
```

```
        DW      0B7EH                   ;1011|0111|1111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0DEFDH                  ;1101|1110|1111|1101
        DW      0F1F3H                  ;1111|0001|1111|0011
        DB      24H

DT_7:   DW      087FEH                  ;1000|0111|1111|1110
        DW      0FBFEH                  ;1111|1011|1111|1110
        DW      0FDFEH                  ;1111|1101|1111|1110
        DW      0FEFEH                  ;1111|1110|1111|1110
        DW      0FF7EH                  ;1111|1111|0111|1110
        DW      0FFBEH                  ;1111|1111|1011|1110
        DW      0FFDEH                  ;1111|1111|1101|1110
        DW      0FFEEH                  ;1111|1111|1110|1110
        DW      0FFF6H                  ;1111|1111|1111|0110
        DW      0FFF8H                  ;1111|1111|1111|1000
        DB      24H

DT_8:   DW      0F1C7H                  ;1111|0001|1100|0111
        DW      0EEB5H                  ;1110|1110|1011|1011
        DW      0DF7DH                  ;1101|1111|0111|1101
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0DF7DH                  ;1101|1111|0111|1101
        DW      0EEB5H                  ;1110|1110|1011|1011
        DW      0F1C7H                  ;1111|0001|1100|0111
        DB      24H

DT_9:   DW      0E3C7H                  ;1110|0011|1100|0111
        DW      0DF7DH                  ;1101|1111|0111|1101
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0BF7EH                  ;1011|1111|0111|1110
        DW      0DF7DH                  ;1101|1111|0111|1101
        DW      0E0C7H                  ;1110|0000|1100|0111
        DB      24H
;-------------------------------------------------------------------------------
        END
```

ıtures

mpatible 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
to 6V Operating Range
lly Static Operation: 0 Hz to 24 MHz
ee-level Program Memory Lock
3 x 8-bit Internal RAM
Programmable I/O Lines
ree 16-bit Timer/Counters
ıe Interrupt Sources
grammable UART Serial Channel
I Serial Interface
w-power Idle and Power-down Modes
errupt Recovery From Power-down
ıgrammable Watchdog Timer
al Data Pointer
wer-off Flag

scription

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

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

downloadable Flash can be changed a single byte at a time and is accessible gh the SPI serial interface. Holding RESET active forces the SPI bus into a serial amming interface and allows the program memory to be written to or read from s Lock Bit 2 has been activated.

ATMEL

8-bit Microcontroller with 8K Bytes Flash

AT89S8252

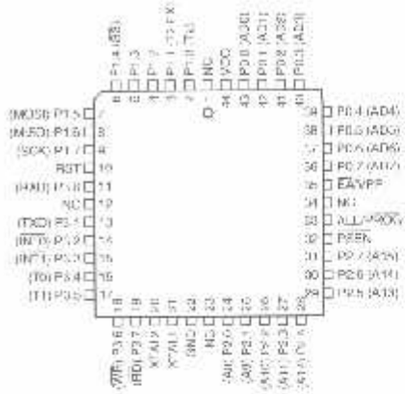
Rev. 0401E–02/00

ATMEL

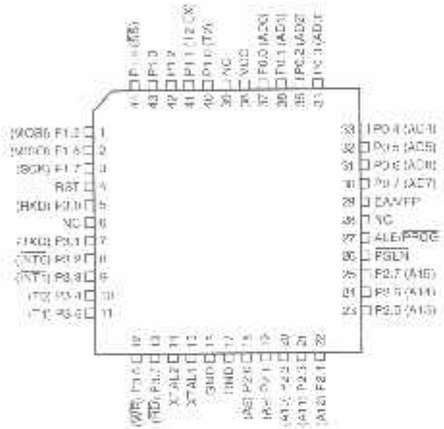
Configurations

PDIP

PLCC



PQFP/TQFP



Description

ly voltage.

nd.

0

) is an 8-bit open drain bbi-didirectional I/O port. As an it port, each pin can sink eight TTL inputs. When 1s /ritten to port 0 pins, the pins can be used as high-dance inputs.

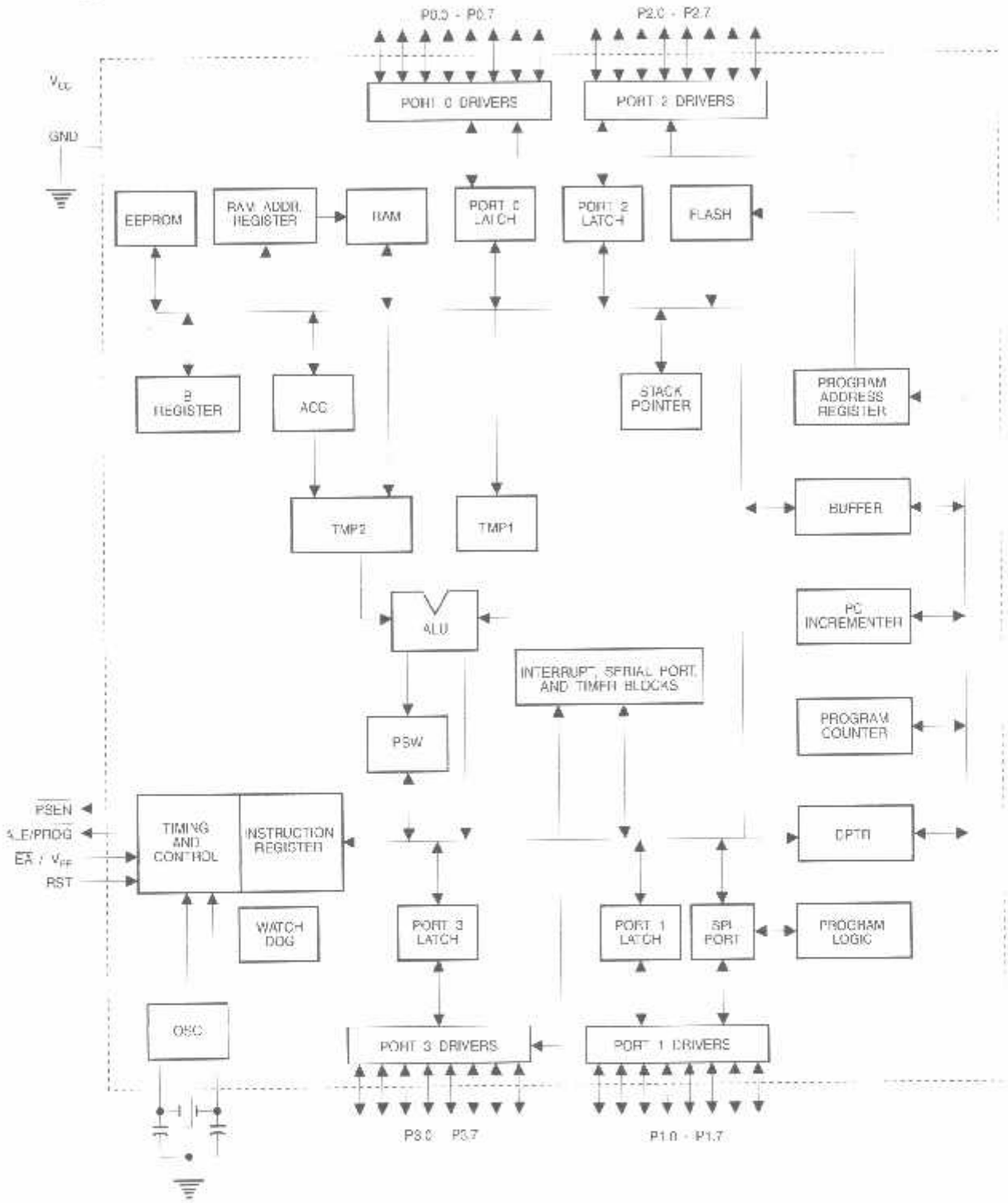
) can also be configured to be the multiplexed low-r address/data bus during accesses to external program and data memory. In this mode, P0 has internal pullups.

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

Port 1

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

ıck 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
BUFFER
TMP2
TMP1
ALU
PC INCREMENTER
INTERRUPT, SERIAL PORT, AND TIMER BLOCKS
PROGRAM COUNTER
PSW
PSEN
ALE/PROG
EA / V_{PP}
RST
TIMING AND CONTROL
INSTRUCTION REGISTER
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

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

Description

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

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

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

2

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

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

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

3

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

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

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

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

RST

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

ALE/$\overline{PROG}$

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.

$\overline{PSEN}$

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

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

$\overline{EA}$/VPP

External Access Enable. $\overline{EA}$ must be strapped to GND in order to enable the device to fetch code from externa pro-

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

hould be strapped to V_{CC} for nternal program execu- s. This pin also receives the 12-volt programming ble voltage (V_{PP}) during Flash programming when 12- programming is selected.

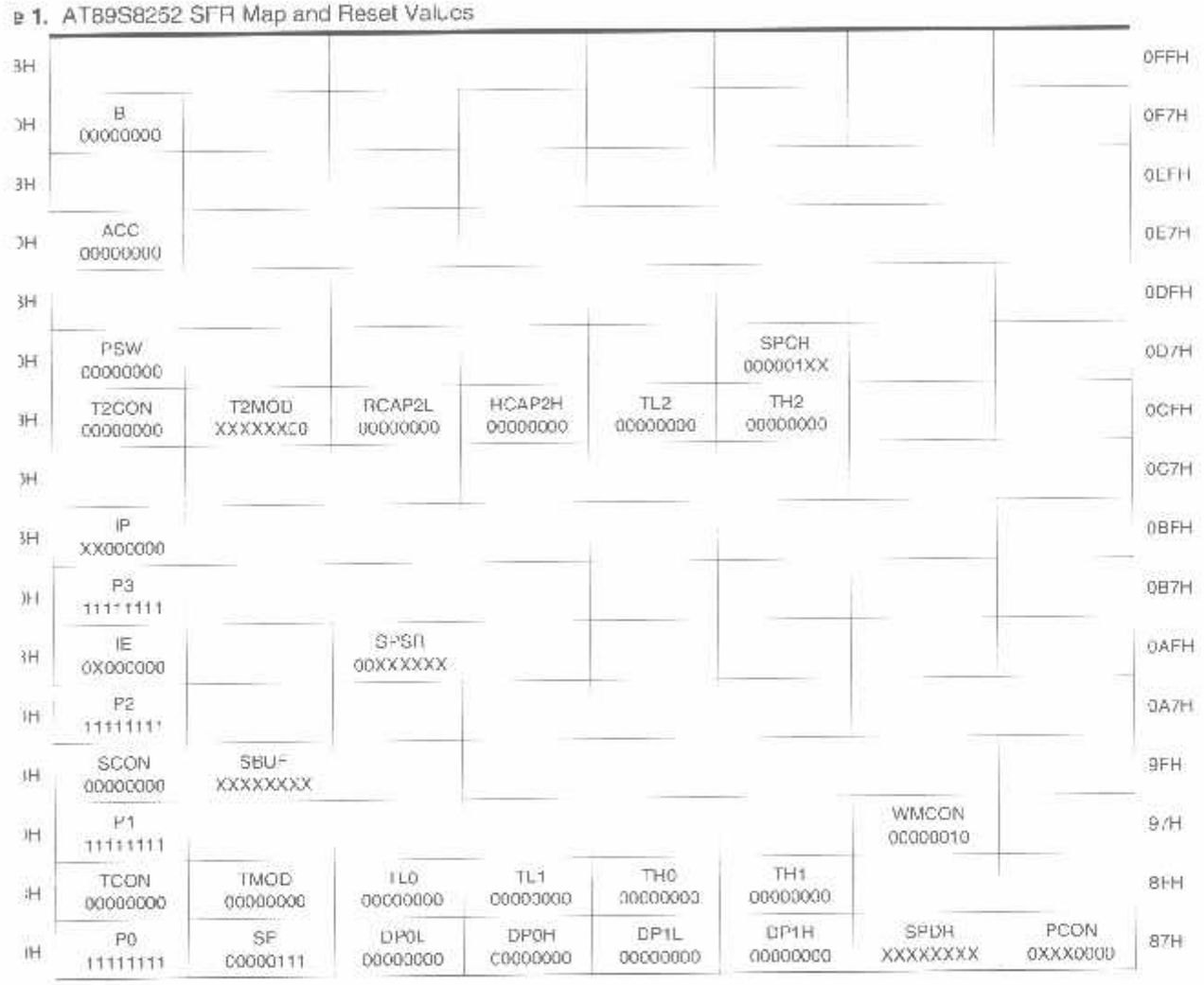
XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

e 1. AT89S8252 SFR Map and Reset Values



ecial Function Registers

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

that not all of the addresses are occupied, and unoc- ed addresses may not be implemented on the chip. l accesses to these addresses will in general return om data, and write accesses will have an indeterminate t.

· 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 containec 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.

e 2. T2CON—Timer/Counter 2 Control Register

:ON Address = 0C8H — Reset Value = 0000 0000B

Addressable

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

ıbol	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.
2	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).
RL2	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

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

e 3. WMCON—Watchdog and Memory Control Register

CON Address = 96H — Reset Value = 0000 0010B

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

ıbol	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.
MWE	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 EEPHOM write is completed.
MEN	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
TRST /BSY	Watchdog Timer Reset and EEPROM Ready/Busy Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The 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/$\overline{BSY}$ flag in a Read-Only mode during EEPROM write. RDY/$\overline{BSY}$ = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/$\overline{BSY}$ bit equals "0" and is automatically reset to "1" when programming is completed.
TEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

legisters Control and status bits for the Serial Periph- nterface are contained in registers SPCR (shown in : 4) and SPSR (shown in Table 5). The SPI data bits ontained in the SPDR register. Writing the SPI data ter during serial data transfer sets the Write Collision /COL, in the SPSR register. The SPDR is double buff- for writing and the values in SPDR are not changed by L

·upt Registers The global interrupt enable bit and the dual interrupt enable bits are in the IE register. In on, the individual interrupt enable bit for the SPI is in PCR register. Two priorities can be set for each of the terrupt 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.

4. SPCR—SPI Control Register

R Address = D5H — Reset Value = 0000 01XXB

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

bol	Function
E	SPI Interrupt Enable. This bit, in conjunction with the ES bit in the IE register, enables SPI interrupts: SPIE = 1 and ES = 1 enable SPI interrupts. SPIE = 0 disables SPI interrupts.
E	SPI Enable. SPI = 1 enables the SPI channel and connects $\overline{SS}$, MOSI, MISO and SCK to pins P1.4, P1.5, P1.6, and P1.7. SPI = 0 disables the SPI channel.
RD	Data Order. DORD = 1 selects LSB first data transmission. DORD = 0 selects MSB first data transmission.
TR	Master/Slave Select. MSTR = 1 selects Master SPI mode. MSTR = 0 selects Slave SPI mode.
OL	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.
HA	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.
R0 R1	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

5. SPSR – SPI Status Register

R Address = AAH — Reset Value = 00XX XXXXB

SPIF	WCOL	–	–	–	–	–	–
7	6	5	4	3	2	1	0

bol	Function
F	SPI Interrupt Flag. When a serial transfer is complete, the SPIF bit is set and an interrupt is generated if SPIE = 1 and ES = 1. The SPIF bit is cleared by reading the SPI status register with SPIF and WCOL bits set, and then accessing the SPI data register.
OL	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.

6. SPDR – SPI Data Register

R Address = 86H — Reset Value = unchanged

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

a Memory – EEPROM and RAM

AT89S8252 implements 2K bytes of on-chip EEPROM ata storage and 256 bytes of RAM. The upper 128 of RAM occupy a parallel space to the Special Func- Registers. That means the upper 128 bytes have the addresses as the SFR space but are physically sepa- rom SFR space.

n an instruction accesses an internal location above ess 7FH, the address mode used in the instruction ifies whether the CPU accesses the upper 128 bytes AM or the SFR space. Instructions that use direct essing access SFR space.

xample, the following direct addressing instruction sses the SFR at location 0A0H (which is P2).

```
mov 0A0H, #data
```

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

```
mov @R0, #data
```

that stack operations are examples of indirect essing, so the upper 128 bytes of data RAM are avail- as stack space.

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

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

dition, during EEPROM programming, an attempted from the EEPROM will fetch the byte being written he MSB complemented. Once the write cycle is com- d, true data are valid at all bit locations.

grammable Watchdog Timer

rogrammable Watchdog Timer (WDT) operates from dependent oscillator. The prescaler bits, PS0, PS1 PS2 in SFR WMCON are used to set the period of the hdog Timer from 16 ms to 2048 ms. The available 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

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S8252 operate the same way as Timer 0 and Timer 1 in the AT89C51, AT89C52 and AT89C55. For further information, see the October 1995 Microcontroller Data Book, page 2-45, section titled, "Timer/Counters."

Timer 2

Timer 2 is a 16 bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ 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

ransition was detected. Since two machine cycles (24 ator periods) are required to recognize a 1-to-0 transi- the maximum count rate is 1/24 of the oscillator ency. To ensure that a given level is sampled at least before it changes, the level should be held for at least ull machine cycle.

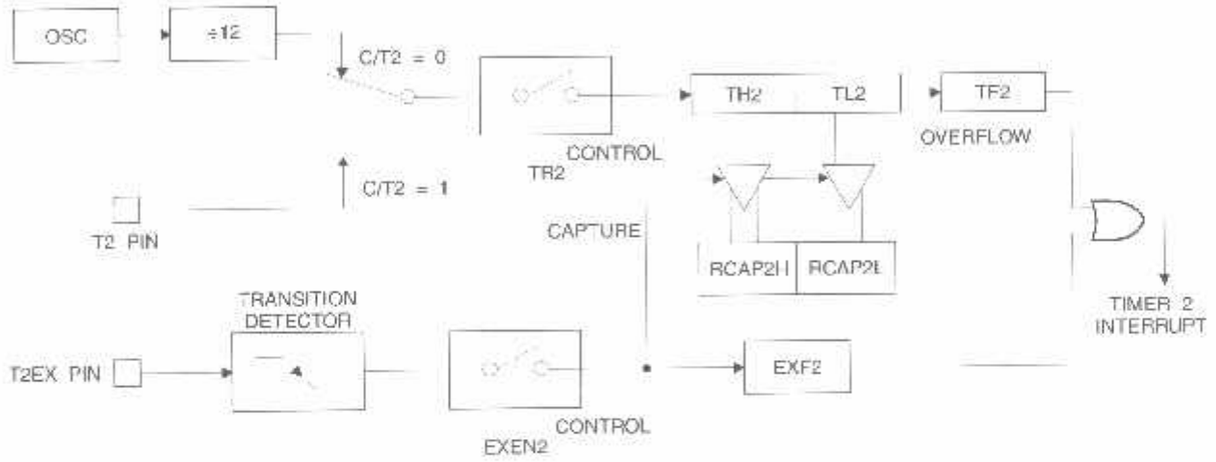
e 8. Timer 2 Operating Modes

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

Capture Mode

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

re 1. Timer 2 in Capture Mode



ɔ-reload (Up or Down Counter)

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

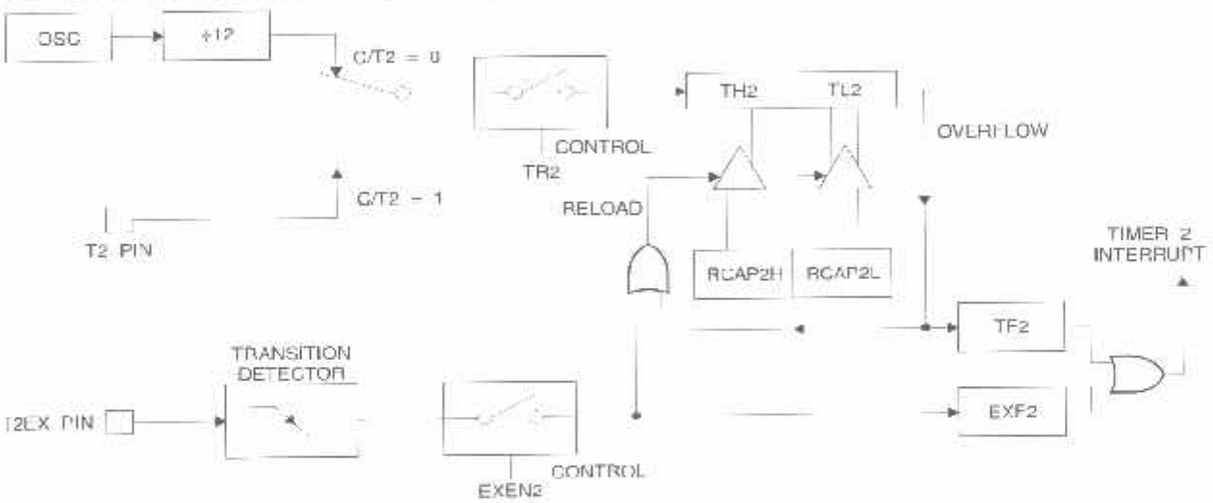
:e 2 shows Timer 2 automatically counting up when N = 0. In this mode, two options are selected by bit N2 in T2CON. If EXEN2 = 0, Timer 2 counts up to FFH and then sets the TF2 bit upon overflow. The low also causes the timer registers to be reloaded with 6 bit value in RCAP2H and RCAP2L. The values in P2H and RCAP2L are preset by software. If EXEN2 = 6 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.

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



9. T2MOD – Timer 2 Mode Control Register

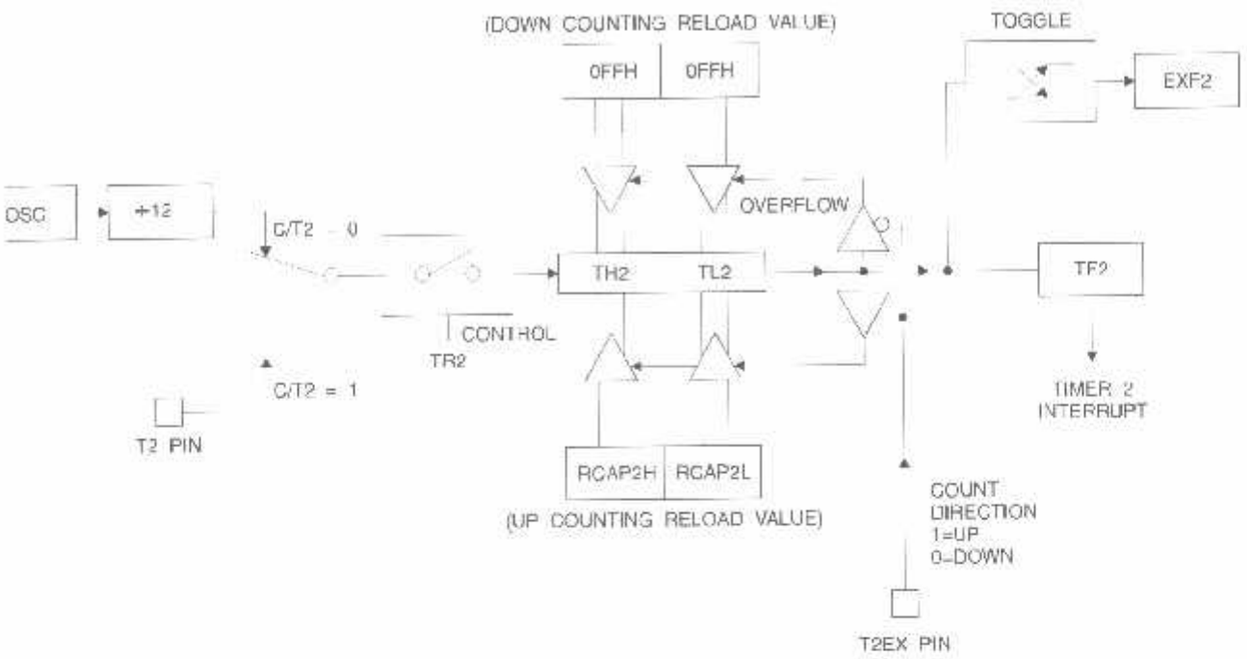
OD Address = 0C9H — Reset Value = XXXX XX00B

Bit Addressable

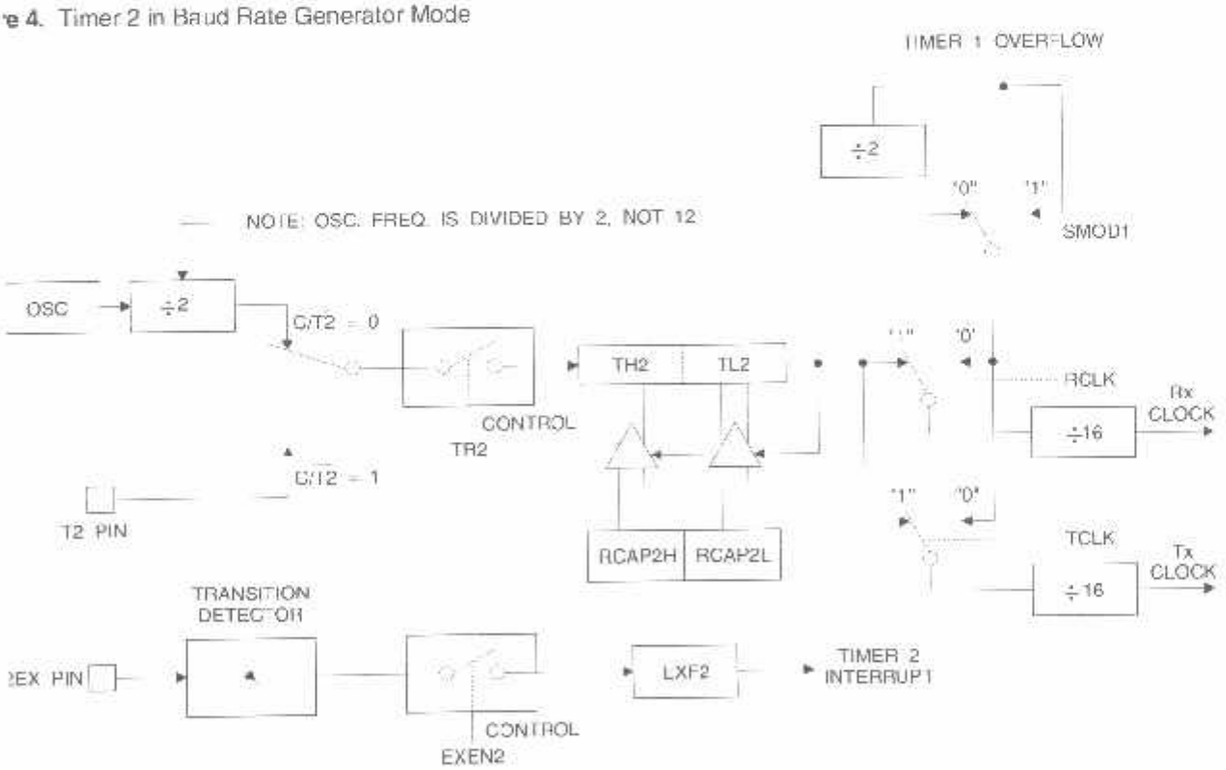
–	–	–	–	–	–	T2OE	DCEN
7	6	5	4	3	2	1	0

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

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



e 4. Timer 2 in Baud Rate Generator Mode



ıd Rate Generator

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

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

ɔaud rates in Modes 1 and 3 are determined by Timer
/erflow rate according to the following equation

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

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

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

ɜ (RCAP2H, RCAP2L) is the content of RCAP2H and
P2L taken as a 16 bit unsigned integer.

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

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

Programmable Clock Out

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

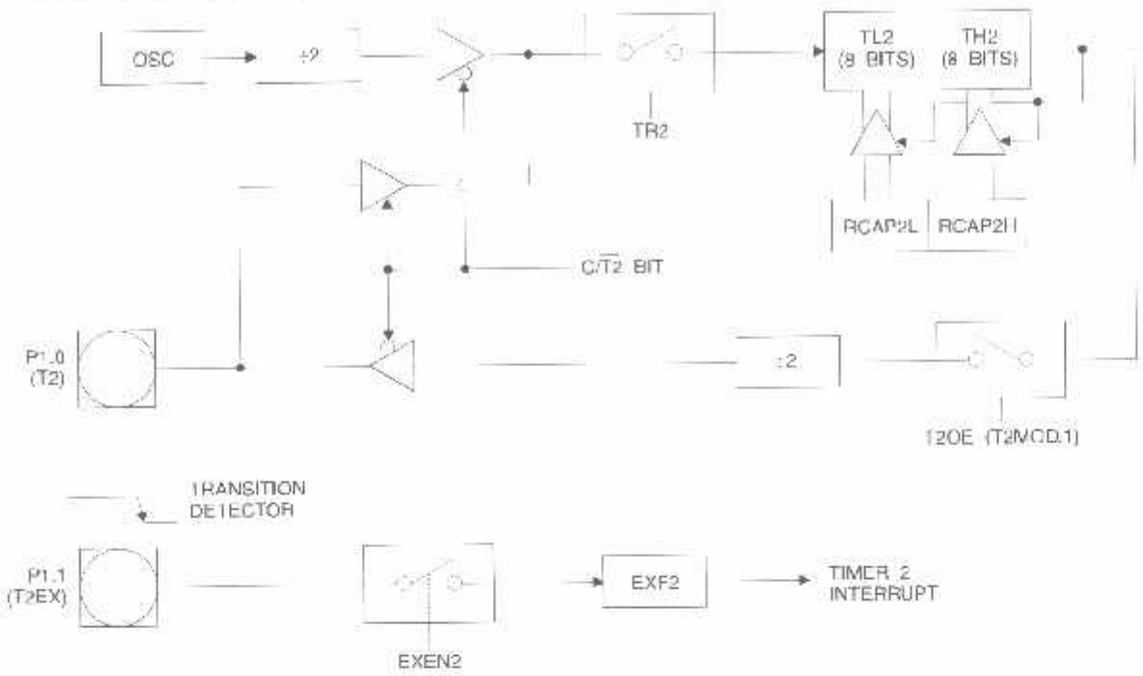
To configure the Timer/Counter 2 as a clock generator, bit C/$\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

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.

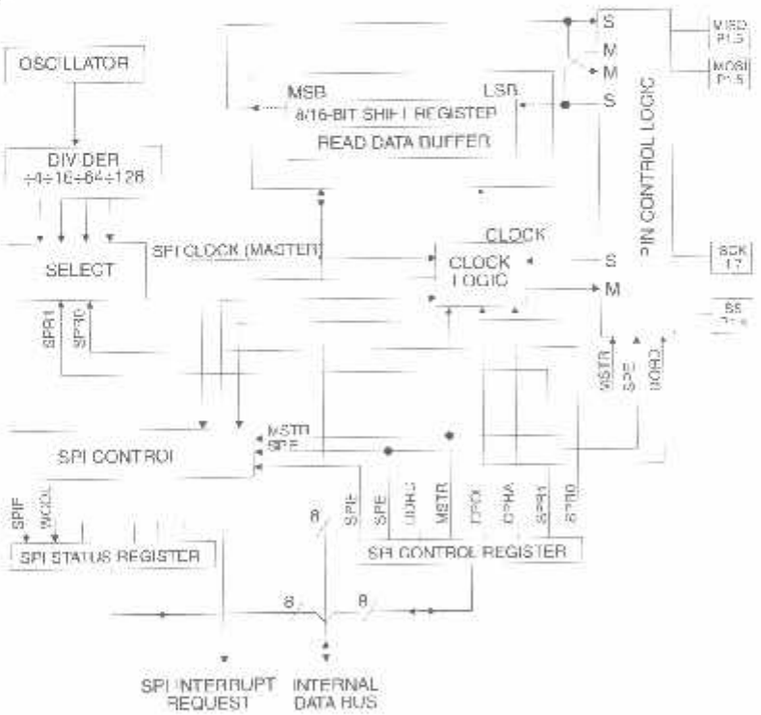
$$\text{Clock Out Frequency} = \frac{\text{Oscillator Frequency}}{4 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

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.

e 5. Timer 2 in Clock-out Mode



e 6. SPI Block Diagram



RT

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

ial Peripheral Interface

erial peripheral interface (SPI) allows high-speed syn- nous data transfer between the AT89S8252 and heral devices or between several AT89S8252 ces. The AT89S8252 SPI features include the ng:

l-Duplex, 3-Wire Synchronous Data Transfer

ster or Slave Operation

MHz Bit Frequency (max.)

B First or MSB First Data Transfer

ur Programmable Bit Rates

d of Transmission Interrupt Flag

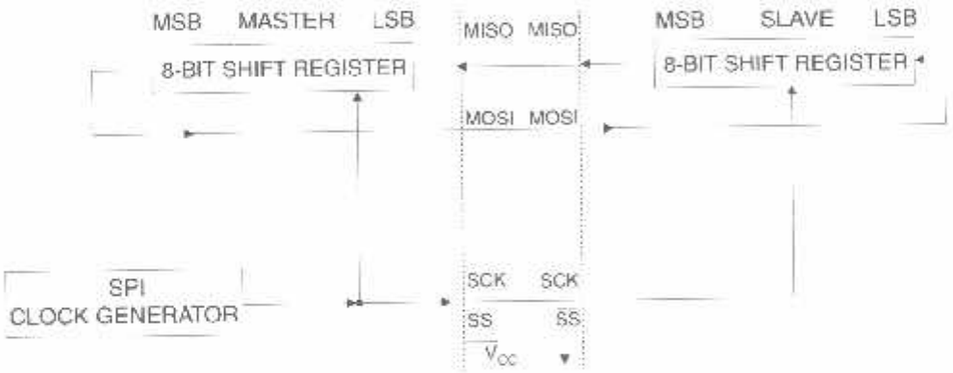
- Write Collision Flag Protection
- Wakeup from Idle Mode (Slave Mode Only)

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 nto 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.

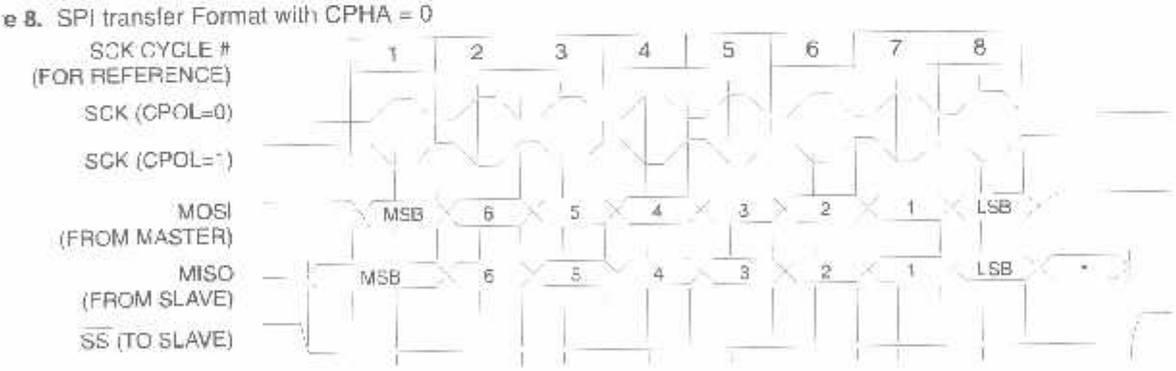
The Slave Select input, $\overline{SS}$/P1.4, is set low to select an individual SPI device as a slave. When $\overline{SS}$/P1.4 is set high, the SPI port is deactivated and the MOSI/P1.5 pin can be used as an input.

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.

e 7. SPI Master-slave Interconnection

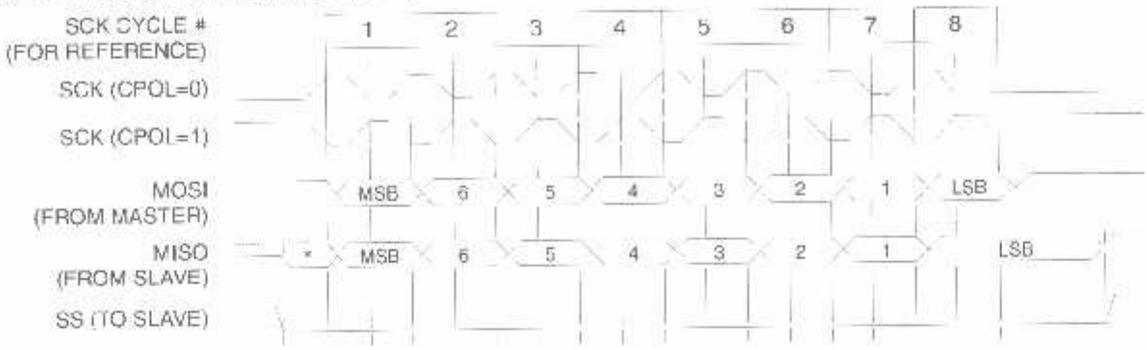


e 8. SPI transfer Format with CPHA = 0



defined but normally MSB of character just received

e 9. SPI Transfer Format with CPHA = 1



defined but normally LSB of previously transmitted character

rrupts

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

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

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

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

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

Table 10. Interrupt Enable (IE) Register

(MSB)(LSB)

EA	–	ET2	ES	ET1	EX1	ET0	EX0

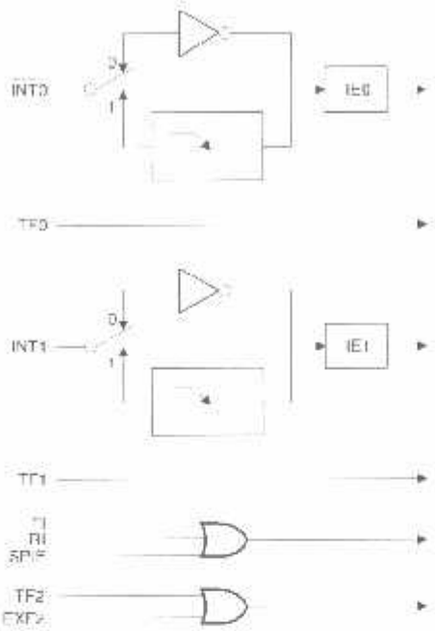
Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

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

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

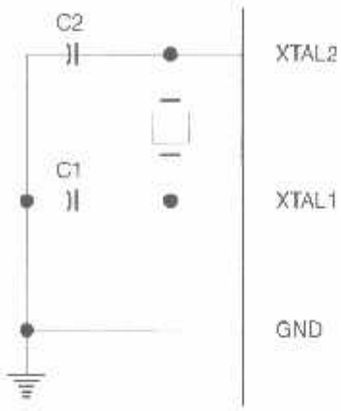
re 10. Interrupt Sources



:illator Characteristics

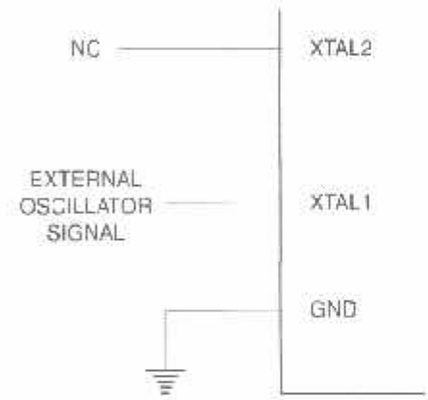
.1 and XTAL2 are the input and output, respectively, inverting amplifier that can be configured for use as -chip oscillator, as shown in Figure 11. Either a quartz al or ceramic resonator may be used. To drive the e from an external clock source, XTAL2 should be left nnected while XTAL1 is driven, as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

Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration



Mode

e mode, the CPU puts itself to sleep while all the on-peripherals remain active. The mode is invoked by are. The content of the on-chip RAM and all the spe-unctions registers remain unchanged during this . The idle mode can be terminated by any enabled ipt or by a hardware reset.

that when idle mode is terminated by a hardware 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.

us of External Pins During Idle and Power-down Modes

e	Program Memory	ALE	$\overline{PSEN}$	PORT0	PORT1	PORT2	PORT3
	Internal	1	1	Data	Data	Data	Data
	External	1	1	Float	Data	Address	Data
r-down	Internal	0	0	Data	Data	Data	Data
r-down	External	0	0	Float	Data	Data	Data

er-down Mode

power-down mode, the oscillator is stopped and the ction that invokes power-down is the last instruction ited. The on-chip RAM and Special Function Regis-etain their values until the power-down mode is iated. Exit from power-down can be initiated either by dware reset or by an enabled external interrupt. Reset nes the SFRs but does not change the on-chip RAM. eset should not be activated before V_{CC} is restored to rmal operating level and must be held active long gh to allow the oscillator to restart and stabilize.

it power-down via an interrupt, the external interrupt be enabled as level sensitive before entering power-The interrupt service routine starts at 16 ms (nomi-lter the enabled interrupt pin is activated.

Program Memory Lock Bits

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

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

k Bit Protection Modes[1][2]

rogram 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

gramming the Flash and EEPROM

's AT89S8252 Flash Microcontroller offers 8K bytes ystem reprogrammable Flash Code memory and 2K of EEPROM Data memory.

T89S8252 is normally shipped with the on-chip Flash and EEPROM Data memory arrays in the erased (i.e. contents = FFH) and ready to be programmed. device supports a High-voltage (12V) Parallel programming mode and a Low-voltage (5V) Serial amming mode. The serial programming mode pro- a convenient way to download the AT89S8252 inside er's system. The parallel programming mode is com- e with conventional third party Flash or EPROM ammers.

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

ode and Data memory arrays on the AT89S8252 are ammed byte-by-byte in either programming mode. An rase cycle is provided with the self-timed program- operation in the serial programming mode. There is ed to perform the Chip Erase operation to reprogram emory location in the serial programming mode any of the lock bits have been programmed.

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

el Programming Algorithm: To program and verify T89S8252 in the parallel programming mode, the fol- sequence is recommended:

ower-up sequence:

pply power between V_{CC} and GND pins.

et RST pin to "H".

pply a 3 MHz to 24 MHz clock to XTAL1 pin and wait r at least 10 milliseconds.

et $\overline{PSEN}$ pin to "L"

LE pin to "H"

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

pply the appropriate combination of "H" or "L" logic vels to pins P2.6, P2.7, P3.6, P3.7 to select one of e programming operations shown in the Flash rogramming Modes table.

pply the desired byte address to pins P1.0 to P1.7 nd P2.0 to P2.5.

pply data to pins P0.0 to P0.7 for Write Code peration.

5. Raise 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 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.

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

Ready/$\overline{Busy}$: The progress of byte programming in the parallel programming mode can also be monitored by the RDY/$\overline{BSY}$ output signal. Pin P3.4 is pulled Low after ALE goes High during programming to indicate $\overline{BUSY}$. 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/$\overline{PROG}$ low for 10 ms. The Code and Data arrays are written with all "1"s in the Chip Erase operation.

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

ig chip erase, a serial read from any address location turn 00H at the data outputs.

l Programming Fuse: A programmable fuse is avail- to disable Serial Programming if the user needs mum system security. The Serial Programming Fuse nly be programmed or erased in the Parallel Program- Mode.

AT89S8252 is shipped with the Serial Programming enabled.

ling the Signature Bytes: The signature bytes are by the same procedure as a norma verification of ons 030H and 031H, except that P3.6 and P3.7 must lled to a logic low. The values returned are as follows:

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

gramming Interface

/ code byte in the Flash and EEPROM arrays can be n, and the entire array can be erased, by using the priate combination of control signals. The write oper- cycle is self-timec and once initiated, will natically time itself to completion.

ajor programming vendors offer worldwide support for tmel microcontroller series. Please contact your local amming vendor for the appropriate software revision.

ial Downloading

the Code and Data memory arrays can be pro- med using the serial SPI bus while RST is pulled to he serial interface consists of pins SCK, MOSI (input) MISO (output). After RST is set high, the Programming le instruction needs to be executed first before pro- erase operations can be executed.

ito-erase cycle is built into the self-timed programming tion (in the serial mode ONLY) and there is no need t execute the Chip Erase instruction unless any of the bits have been programmed. The Chip Erase opera- urns the content of every memory location in both the and Data arrays into FFH.

Code and Data memory arrays have separate address es:

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

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

Serial Programming Algorithm

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

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

Power-off sequence (if needed):
Set XTAL1 to "L" (if a crystal is not used).
Set RST to "L".
Turn V_{CC} power off.

Serial Programming Instruction

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

:ruction Set

ruction	Input Format			Operation
	Byte 1	Byte 2	Byte 3	
ıramming Enable	1010 1100	01C1 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.
ı Code Memory	aaaa a001	low addr	xxxx xxxx	Reac 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.
ə Coce Memory	aaaa a010	low addr	data in	Write data :o Code memory location at selectec address. The address bits are the 5 MSBs of the first byte togelher w th the second byte.
ı 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.
ə Data Memory	00aa a110	low addr	cata in	Write data to Data memory location at selected address.
ə Lock Bits	1010 1100	x x111	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

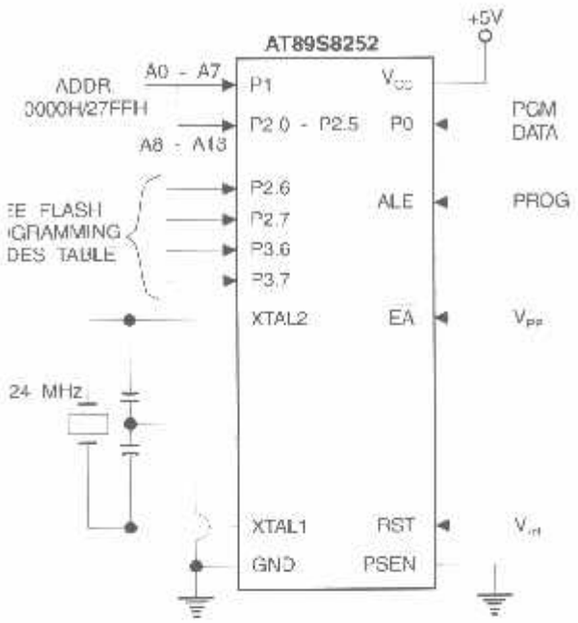
1. DATA polling is used to ind cate the end of a write cycle which typically takes less than 2.5 ms at 5V.
2. "aaaaa" = high order address.
3. "x" = don't care.

sh and EEPROM Parallel Programming Modes

e	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						
Erase	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 $\overline{PROG}$ 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 $\overline{RDY/BSY}$.
4. "X" = don't care

e 13. Programming the Flash/EEPROM Memory



e 14. Verifying the Flash/EEPROM Memory

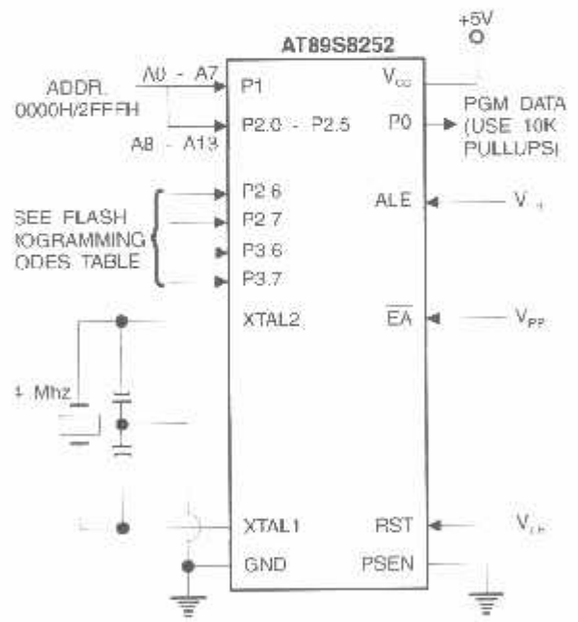
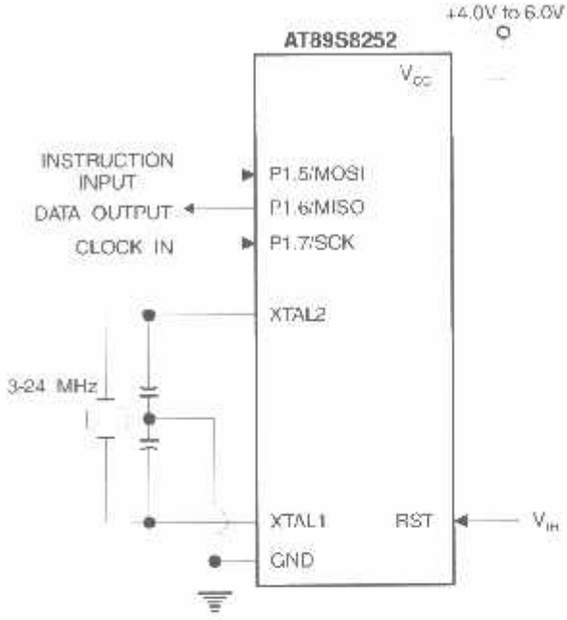


Figure 15. Flash/EEPROM Serial Downloading

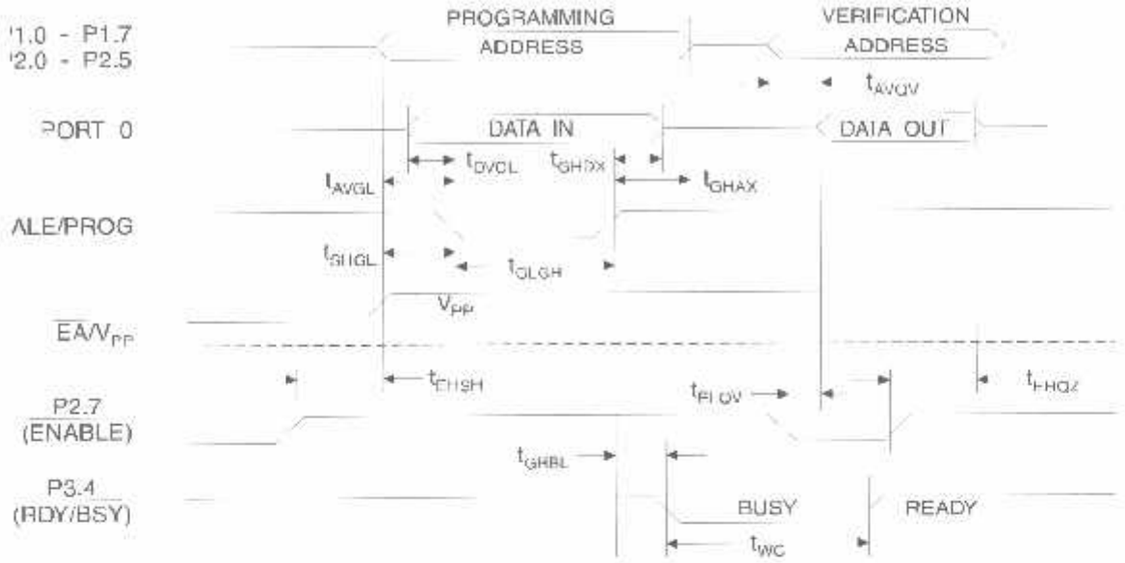


sh Programming and Verification Characteristics – Parallel Mode

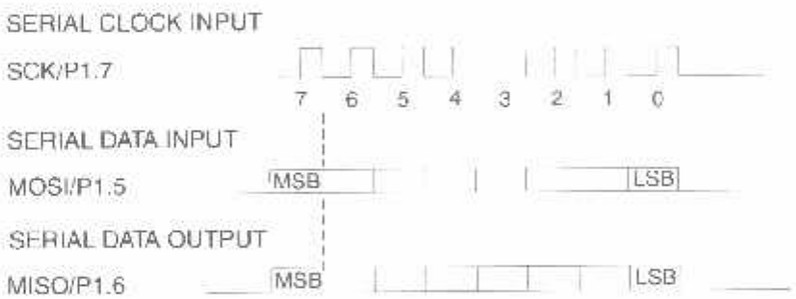
)°C to 70°C, V_{CC} = 5.0V ± 10%

bol	Parameter	Min	Max	Units
	Programming Enable Voltage	11.5	12.5	V
	Programming Enable Current		1.0	mA
CL	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 BUSY Low		1.0	µs
	Byte Write Cycle Time		2.0	ms

sh/EEPROM Programming and Verification Waveforms – Parallel Mode



ial Downloading Waveforms



;olute Maximum Ratings*

rating Temperature	-55°C to +125°C
age Temperature	-65°C to +150°C
ıge on Any Pin Respect to Ground	-1.0V to +7.0V
imum Operating Voltage	6.6V
Output Current	15.0 mA

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

Characteristics

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

bol	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 Hifh-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
	Input Hifh-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 Hifh-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 Hifh-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, EA)	0.45 < V_{IN} < V_{CC}		±10	µA
T	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

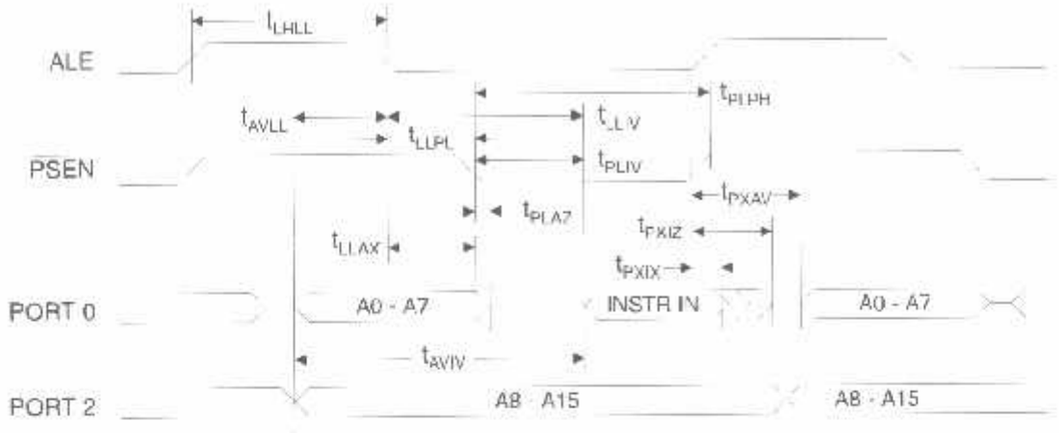
Characteristics

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

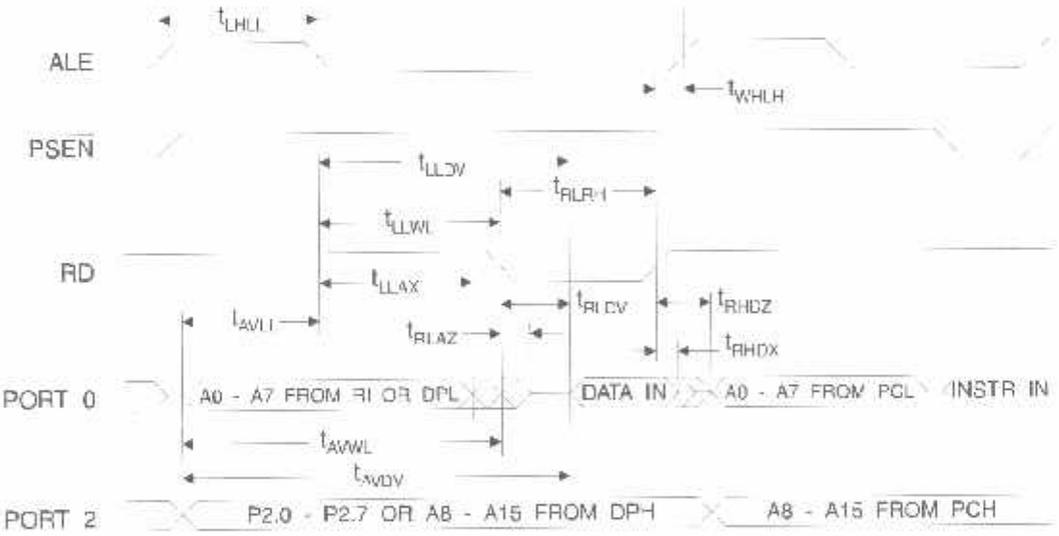
ernal Program and Data Memory Characteristics

ibol	Parameter	Variable Oscillator Min	Variable Oscillator Max	Units
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
t	$\overline{WR}$ 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 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 RD or $\overline{WR}$ Low	$4t_{CLCL}$ - 75		ns
c	Data Valid to $\overline{WR}$ Transition	t_{CLCL} - 20		ns
t	Data Valid to $\overline{WR}$ High	$7t_{CLCL}$ - 120		ns
t	Data Hold after $\overline{WR}$	t_{CLCL} - 20		ns
	$\overline{RD}$ Low to Address Float		0	ns
t	RD or $\overline{WR}$ High to ALE High	t_{CLCL} - 20	t_{CLCL} + 25	ns

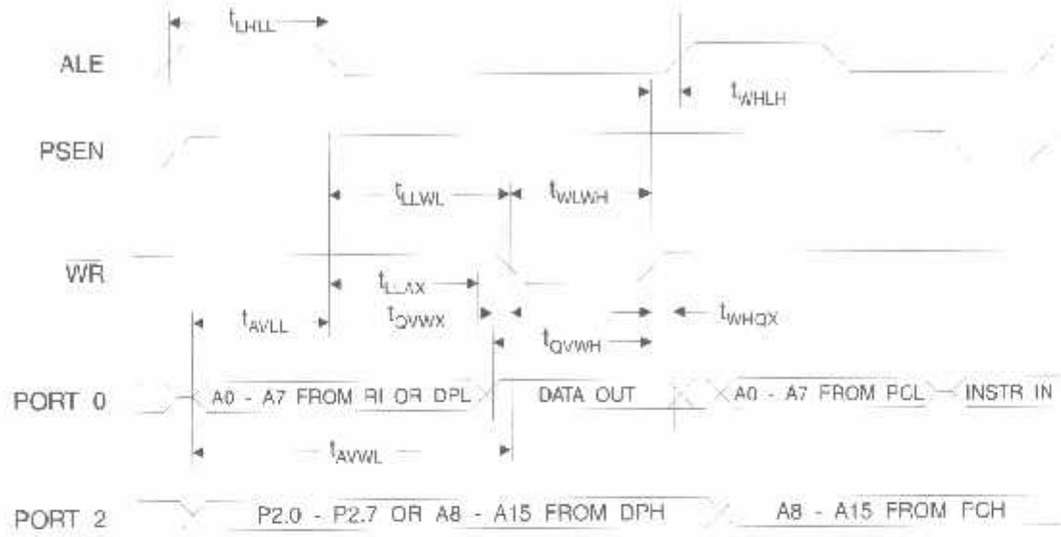
ernal Program Memory Read Cycle



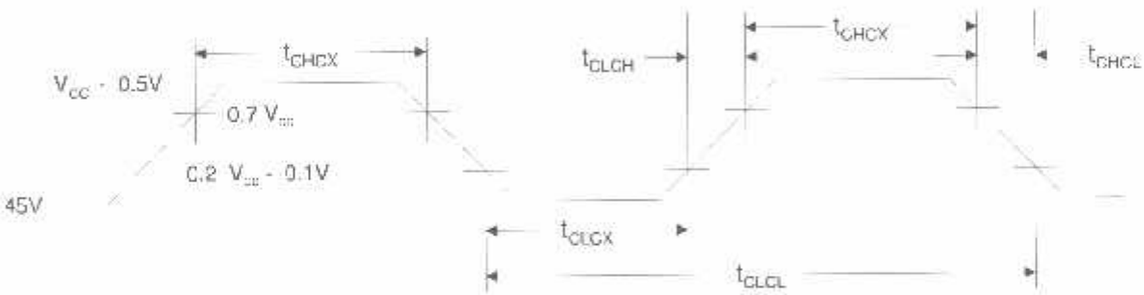
ernal Data Memory Read Cycle



ernal Data Memory Write Cycle



ernal Clock Drive Waveforms



ernal Clock Drive

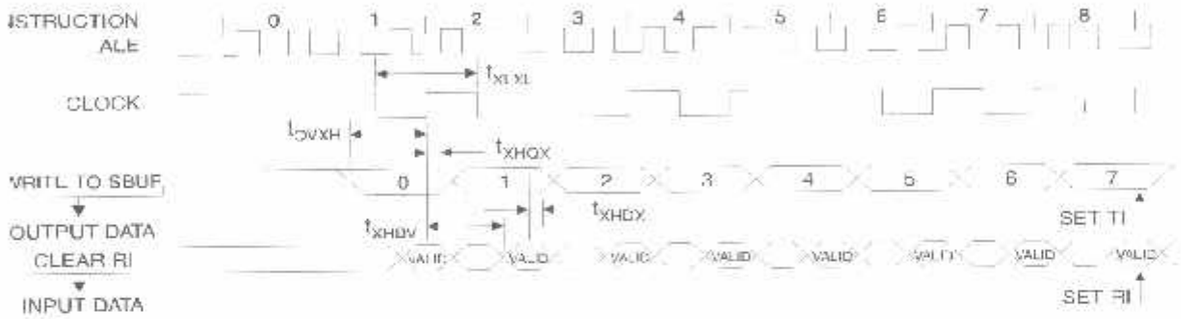
bol	Parameter	V_{CC} = 4.0V to 6.0V		Units
		Min	Max	
	Oscillator Frequency	0	24	MHz
	Clock Period	41.6		ns
	High Time	15		ns
	Low Time	15		ns
	Rise Time		20	ns
	Fall Time		20	ns

al Port Timing: Shift Register Mode Test Conditions

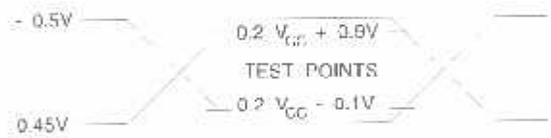
alues in this table are valid for V_{CC} = 4.0V to 6V and Load Capacitance = 80 pF.

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]



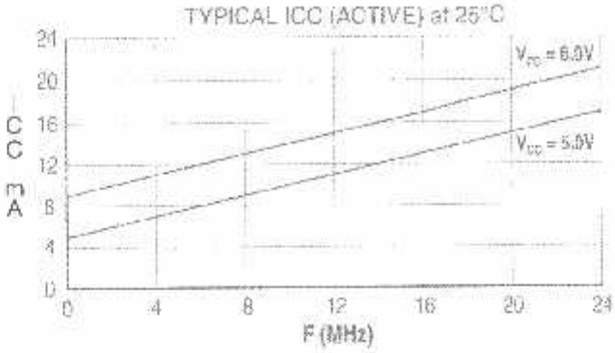
1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

Float Waveforms[1]

Notes: 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.

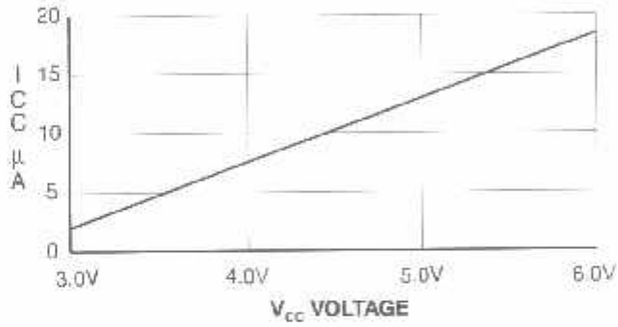
AT89S8252

TYPICAL ICC (ACTIVE) at 25°C



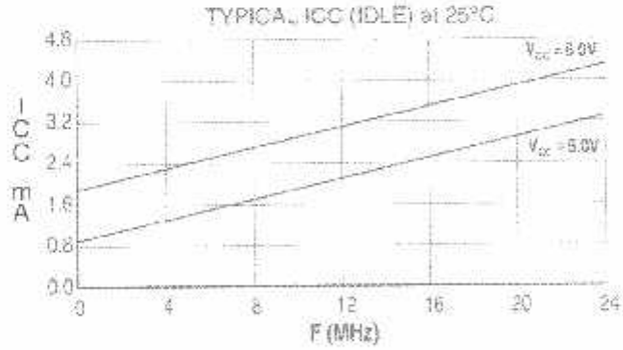
AT89S8252

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



AT89S8252

TYPICAL ICC (IDLE) at 25°C



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

ering Information

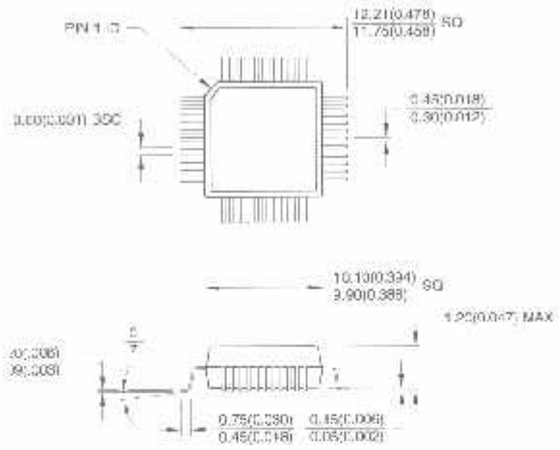
eed Hz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 6.0V	AT89S8252-24AC	44A	Commercial (0°C to 70°C)
		AT89S8252-24JC	44J	
		AT89S8252-24PC	40P6	
		AT89S8252-24QC	44Q	
	4.0V to 6.0V	AT89S8252 24AI	44A	Industrial (-40°C to 85°C)
		AT89S8252-24JI	44J	
		AT89S8252-24PI	40P6	
		AT89S8252-24QI	44Q	
33	4.5V to 5.5V	AT89S8252-33AC	44A	Commercial (0°C to 70°C)
		AT89S8252-33JC	44J	
		AT89S8252-33PC	40P6	
		AT89S8252-33QC	44Q	

= Preliminary Information

	Package Type
	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)
	44-lead, Plastic Gull Wing Quad Flatpack (PQFP)

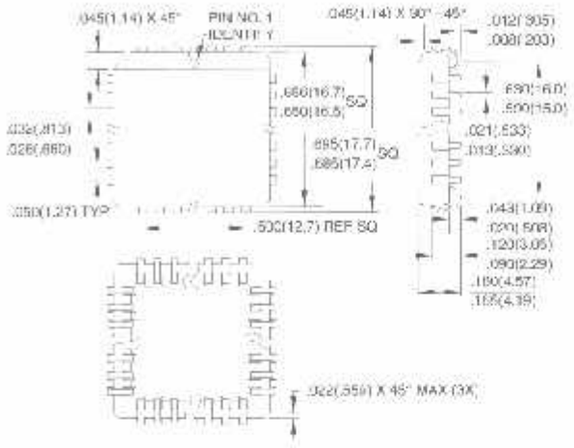
kaging Information

4A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad latpack (TQFP)
imensions in Millimeters and (Inches)*
EDEC STANDARD MS-026 ACB

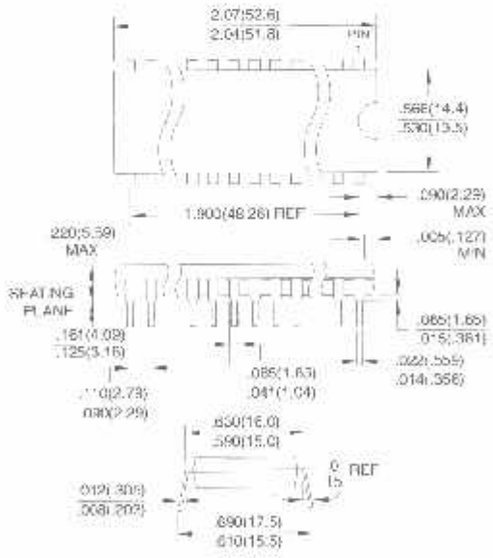


ontrolling dimension: millimeters

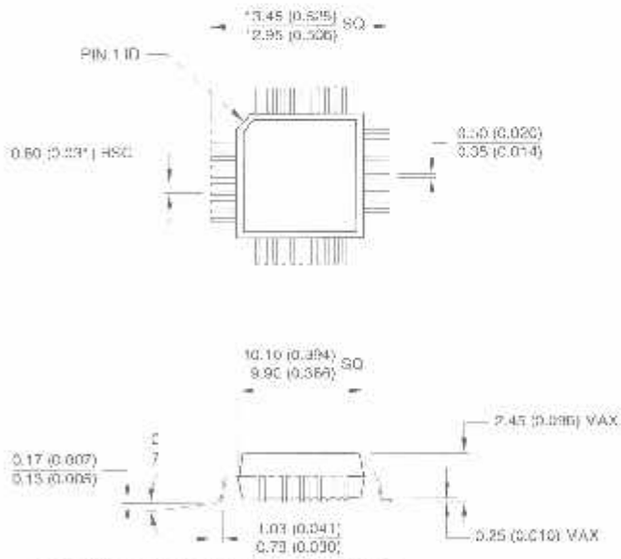
44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-018 AC



IP6, 40-lead, 0.600" Wide, Plastic Dual Inline ckage (PDIP)
mensions in Inches and (Millimeters)



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



Controlling dimension: millimeters

nel Headquarters

porate Headquarters
325 Orchard Parkway
an Jose, CA 95131
EL (408) 441-0311
AX (408) 487-2600

ope
mel U.K., Ltc.
oliseum Business Centre
verside Way
amberley, Surrey GU15 3YL
ngland
EL (44) 1276-686-677
AX (44) 1276-686-697

t
mel Asia, Ltd.
oom 1219
inachem Golden Plaza
Mody Road Tsimhatsui
st Kowloon
ng Kong
EL (852) 2721-9778
AX (852) 2722-1369

an
mel Japan K.K.
, Tonetsu Shinkawa Bldg.
24-8 Shinkawa
uo-ku, Tokyo 104-0033
pan
L (81) 3-3523-3551
AX (81) 3-3523-7581

Atmel Operations

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

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

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

0101E–02/00/xM

DATA SHEET

For a complete data sheet, please also download:

- The IC06 74HC/HCT/HCU/HCMOS Logic Family Specifications
- The IC06 74HC/HCT/HCU/HCMOS Logic Package Information
- The IC06 74HC/HCT/HCU/HCMOS Logic Package Outlines

74HC/HCT595

8-bit serial-in/serial or parallel-out shift register with output latches; 3-state

Product specification
Supersedes data of September 1993
File under Integrated Circuits, IC06

1998 Jun 04

Philips Semiconductors

PHILIPS

8-bit serial-in/serial or parallel-out shift register with output latches; 3-state

74HC/HCT595

EATURES

8-bit serial input

8-bit serial or parallel output

Storage register with 3-state outputs

Shift register with direct clear

100 MHz (typ) shift out frequency

Output capability:

- parallel outputs; bus driver
- serial output; standard

I_{CC} category: MSI.

PPLICATIONS

Serial-to-parallel data conversion

Remote control holding register.

DESCRIPTION

The 74HC/HCT595 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). They are specified in compliance with JEDEC standard no. 7A.

The "595" is an 8-stage serial shift register with a storage register and 3-state outputs. The shift register and storage register have separate clocks.

Data is shifted on the positive-going transitions of the SH_{CP} input. The data in each register is transferred to the storage register on a positive-going transition of the ST_{CP} input. If both clocks are connected together, the shift register will always be one clock pulse ahead of the storage register.

The shift register has a serial input (D_S) and a serial standard output (Q_7') for cascading. It is also provided with asynchronous reset (active LOW) for all 8 shift register stages. The storage register has 8 parallel 3-state bus driver outputs. Data in the storage register appears at the output whenever the output enable input (OE) is LOW.

JICK REFERENCE DATA

ND = 0 V; T_{amb} = 25 °C; $t_r = t_f$ = 6 ns.

YMBOL	PARAMETER	CONDITIONS	TYP.		UNIT
			HC	HCT	
HL/tPLH	propagation delay	C_L = 15 pF; V_{CC} = 5 V			
	SH_{CP} to Q_7'		16	21	ns
	ST_{CP} to Q_n		17	20	ns
	$\overline{MR}$ to Q_7'		14	19	ns
ax	maximum clock frequency SH_{CP}, ST_{CP}		100	57	MHz
	input capacitance		3.5	3.5	pF
PD	power dissipation capacitance per package	notes 1 and 2	115	130	pF

tes

C_{PD} is used to determine the dynamic power dissipation (P_D in µW):

$P_D = C_{PD} \times V_{CC}^2 \times f_i + \Sigma (C_L \times V_{CC}^2 \times f_o)$ where:

f_i = input frequency in MHz

f_o = output frequency in MHz

$\Sigma(C_L \times V_{CC}^2 \times f_o)$ = sum of outputs

C_L = output load capacitance in pF

V_{CC} = supply voltage in V

For HC the condition is V_I = GND to V_{CC}; for HCT the condition is V_I = GND to V_{CC} – 1.5 V.

RDERING INFORMATION

TYPE NUMBER	PACKAGE		
	NAME	DESCRIPTION	VERSION
4HC595N	DIP16	plastic dual in-line package; 16 leads (300 mil); long body	SOT38-1
4HC595D	SO16	plastic small outline package; 16 leads; body width 3.9 mm	SOT109-1
4HC595DB	SSOP16	plastic shrink small outline package; 16 leads; body width 5.3 mm	SOT338-1
4HC595PW	TSSOP16	plastic thin shrink small outline package; 16 leads; body width 4.4 mm	SOT403-1
4HCT595N	DIP16	plastic dual in-line package; 16 leads (300 mil); long body	SOT38-1
4HCT595D	SO16	plastic small outline package; 16 leads; body width 3.9 mm	SOT109-1

NNING

SYMBOL	PIN	DESCRIPTION
$_0$ to Q_7	15, 1 to 7	parallel data output
ND	8	ground (0 V)
$_7$'	9	serial data output
$\overline{IR}$	10	master reset (active LOW)
H_{CP}	11	shift register clock input
T_{CP}	12	storage register clock input
$\overline{E}$	13	output enable (active LOW)
S	14	serial data input
CC	16	positive supply voltage

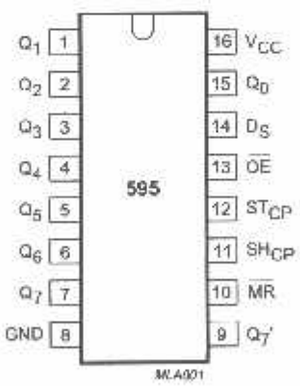


Fig.1 Pin configuration.

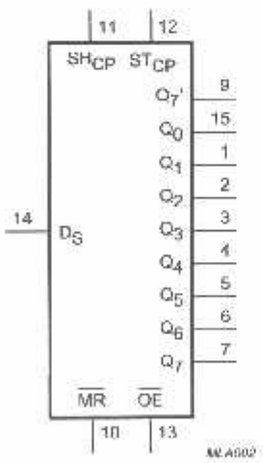


Fig.2 Logic symbol.

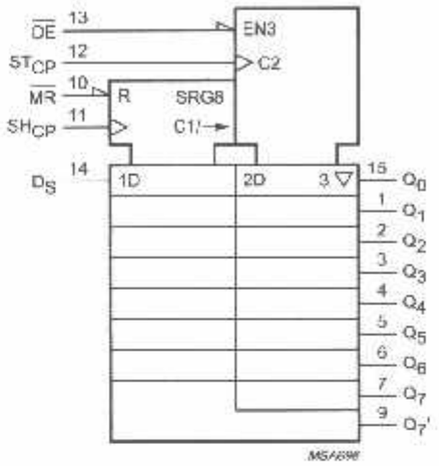


Fig.3 IEC logic symbol.

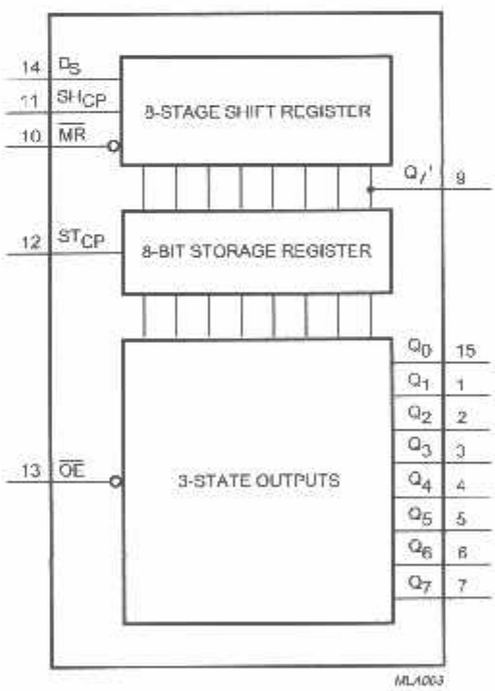


Fig.4 Functional diagram.

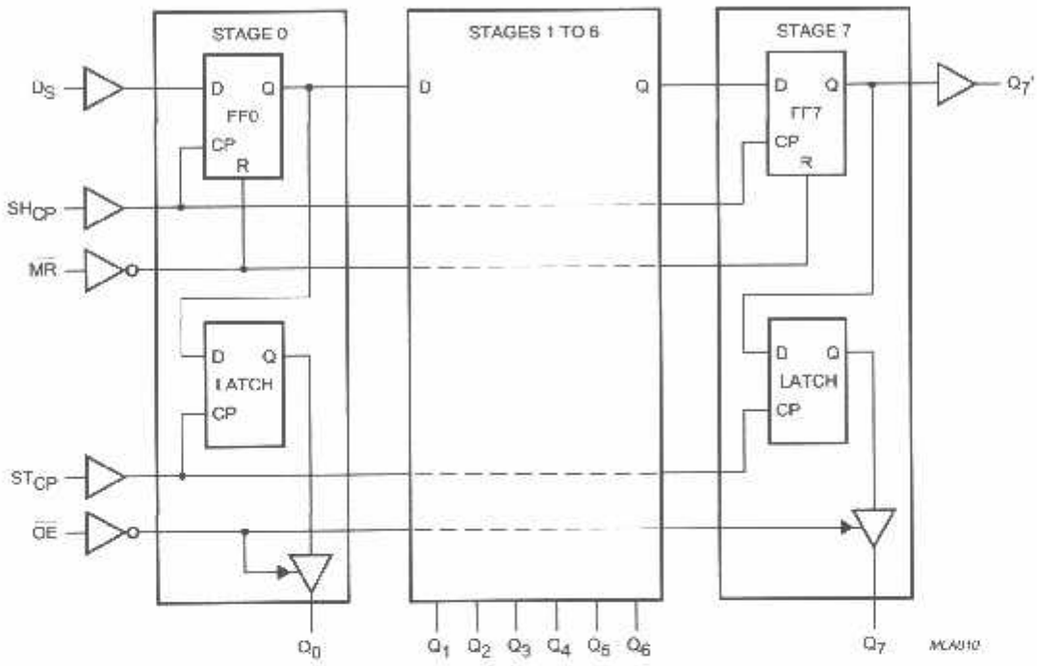


Fig.5 Logic diagram.

JNCTION TABLE

INPUTS					OUTPUTS		FUNCTON
SH_{CP}	ST_{CP}	$\overline{OE}$	$\overline{MR}$	D_S	Q_7'	Q_N	
X	X	L	L	X	L	NC	a LOW level on $\overline{MR}$ only affects the shift registers
X	↑	L	L	X	L	L	empty shift register loaded into storage register
X	X	H	L	X	L	Z	shift register clear. Parallel outputs in high-impedance OFF-state
↑	X	L	H	H	Q_6'	NC	logic high level shifted into shift register stage 0. Contents of all shift register stages shifted through, e.g. previous state of stage 6 (internal Q_6') appears on the serial output (Q_7')
X	↑	L	H	X	NC	Q_n'	contents of shift register stages (internal Q_n') are transferred to the storage register and parallel output stages
↑	↑	L	H	X	Q_6'	Q_n'	contents of shift register shifted through. Previous contents of the shift register is transferred to the storage register and the parallel output stages.

ıtes

H = HIGH voltage level; L = LOW voltage level
↑ = LOW-to-HIGH transition; ↓ = HIGH-to-LOW transition
Z = high-impedance OFF-state; NC = no change
X = don't care.

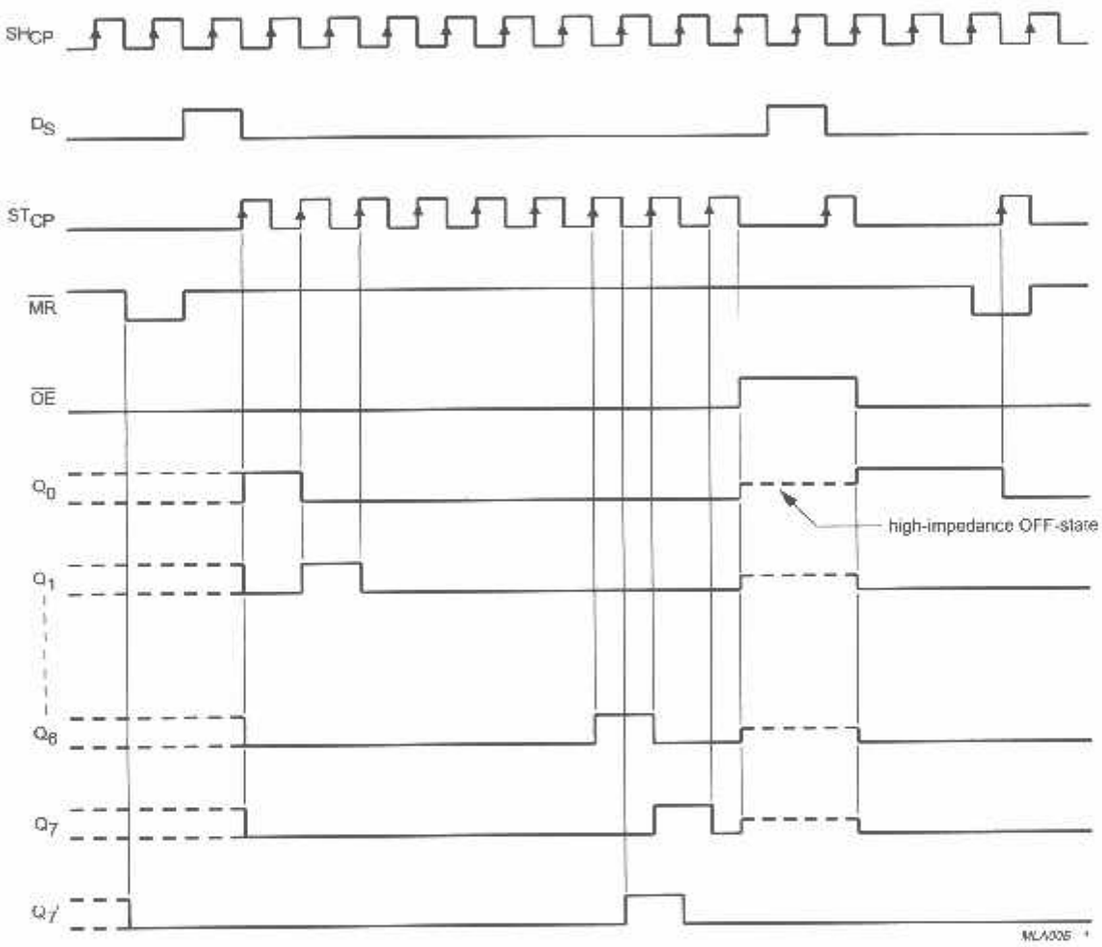


Fig.6 Timing diagram.

: CHARACTERISTICS FOR 74HC

r the DC characteristics see chapter *"74HC/HCT/HCU/HCMOS Logic Family Specifications"*.

itput capability: parallel outputs, bus driver, serial output, standard I_{CC} category: MSI.

: CHARACTERISTICS FOR 74HC

ND = 0 V; $t_r = t_f$ = 6 ns; C_L = 50 pF.

YMBOL	PARAMETER	T_{amb} (°C)							UNIT	TEST CONDITION	
		+25			−40 to +85		−40 to +125			V_{CC} (V)	WAVEFORMS
		min	typ	max	min	max	min	max			
HL/tPLH	propagation delay SH_{CP} to Q_7'	–	52	160	–	200	–	240	ns	2.0	Fig.7
		–	19	32	–	40	–	48		4.5	
		–	15	27	–	34	–	41		6.0	
HL/tPLH	propagation delay ST_{CP} to Q_n	–	55	175	–	220	–	265	ns	2.0	Fig.8
		–	20	35	–	44	–	53		4.5	
		–	16	30	–	37	–	45		6.0	
HL	propagation delay $\overline{MR}$ to Q_7'	–	47	175	–	220	–	265	ns	2.0	Fig.10
		–	17	35	–	44	–	53		4.5	
		–	14	30	–	37	–	45		6.0	
ZH/tPZL	3-state output enable time $\overline{OE}$ to Q_n	–	47	150	–	190	–	225	ns	2.0	Fig.11
		–	17	30	–	38	–	45		4.5	
		–	14	26		33	–	38		6.0	
HZ/tPLZ	3-state output disable time $\overline{OE}$ to Q_n	–	41	150	–	190	–	225	ns	2.0	Fig.11
		–	15	30	–	38	–	45		4.5	
		–	12	26	–	33	–	38		6.0	
	shift clock pulse width HIGH or LOW	75	17	–	95	–	110	–	ns	2.0	Fig.7
		15	6	–	19	–	22	–		4.5	
		13	5	–	16	–	19	–		6.0	
	storage clock pulse width HIGH or LOW	75	11	–	95	–	110	–	ns	2.0	Fig.8
		15	4		19	–	22	–		4.5	
		13	3	–	16	–	19	–		6.0	
	master reset pulse width LOW	75	17	–	95	–	110	–	ns	2.0	Fig.10
		15	6.0	–	19	–	22	–		4.5	
		13	5.0	–	16	–	19	–		6.0	
	set-up time D_S to SH_{CP}	50	11	–	65	–	75	–	ns	2.0	Fig.9
		10	4.0	–	13	–	15	–		4.5	
		9.0	3.0	–	11	–	13	–		6.0	
	set-up time SH_{CP} to ST_{CP}	75	22	–	95	–	110	–	ns	2.0	Fig.8
		15	8	–	19	–	22	–		4.5	
		13	7	–	16	–	19	–		6.0	

SYMBOL	PARAMETER	T_{amb} (°C)							UNIT	TEST CONDITION	
		+25			−40 to +85		−40 to +125			V_{CC} (V)	WAVEFORMS
		min	typ	max	min	max	min	max			
	hold time D_S to SH_{CP}	3	−6	–	3	–	3	–	ns	2.0	Fig.9
		3	−2	–	3	–	3	–		4.5	
		3	2	–	3	–	3	–		6.0	
em	removal time $\overline{MR}$ to SH_{CP}	50	−19	–	65	–	75	–	ns	2.0	Fig.10
		10	−7	–	13	–	15	–		4.5	
		9	−6		11	–	13	–		6.0	
nax	maximum clock pulse frequency SH_{CP} or ST_{CP}	9	30	–	4.8	–	4	–	MHz	2.0	Figs 7 and 8
		30	91	–	24	–	20	–		4.5	
		35	108	–	28	–	24			6.0	

: CHARACTERISTICS FOR 74HCT

ır the DC characteristics see chapter *"74HC/HCT/HCU/HCMOS Logic Family Specifications"*.

ıtput capability: parallel outputs, bus driver; serial output, standard I_{CC} category: MSI.

ıte to HCT types

ıe value of additional quiescent supply current (ΔI_{CC}) for a unit load of 1 is given in the family specifications. ı determine ΔI_{CC} per input, multiply this value by the unit load coefficient shown in the table below.

ND = 0 V; $t_r = t_f$ = 6 ns; C_L = 50 pF.

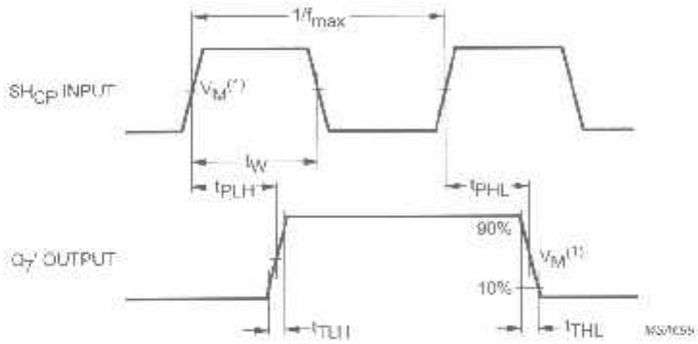
IPUT	UNIT LOAD COEFFICIENT
D_S	0.25
$\overline{MR}$	1.50
SH_{CP}	1.50
ST_{CP}	1.50
$\overline{OE}$	1.50

C CHARACTERISTICS FOR 74HCT

ND = 0 V; $t_r = t_f = 6$ ns; $C_L = 50$ pF.

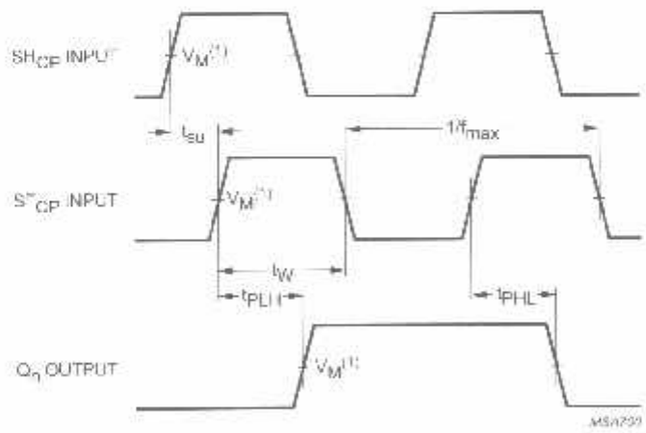
YMBOL	PARAMETER	T_{amb} (°C)							UNIT	TEST CONDITION	
		+25			−40 to −85		−40 to +125			V_{CC} (V)	WAVEFORMS
		min	typ	max	min	max	min	max			
HL/ t_{PLH}	propagation delay SH_{CP} to Q_7'	–	25	42	–	53	–	63	ns	4.5	Fig.7
HL/ t_{PLH}	propagation delay ST_{CP} to Q_n	–	24	40	–	50	–	60	ns	4.5	Fig.8
HL	propagation delay $\overline{MR}$ to Q_7'	–	23	40	–	50	–	60	ns	4.5	Fig.10
ZH/ t_{PZL}	3-state output enable time $\overline{OE}$ to Q_n	–	21	35	–	44	–	53	ns	4.5	Fig.11
HZ/ t_{PLZ}	3-state output disable time $\overline{OE}$ to Q_n	–	18	30	–	38	–	45	ns	4.5	Fig.11
/	shift clock pulse width HIGH or LOW	16	6		20	–	24	–	ns	4.5	Fig.7
/	storage clock pulse width HIGH or LOW	16	5		20	–	24	–	ns	4.5	Fig.8
	master reset pulse width LOW	20	8	–	25	–	30	–	ns	4.5	Fig.10
	set-up time D_S to SH_{CP}	16	5	–	20	–	24		ns	4.5	Fig.9
	set-up time SH_{CP} to ST_{CP}	16	8	–	20	–	24		ns	4.5	Fig.8
	hold time D_S to SH_{CP}	3	−2	–	3	–	3		ns	4.5	Fig.9
m	removal time $\overline{MR}$ to SH_{CP}	10	−7	–	13	–	15	–	ns	4.5	Fig.10
ax	maximum clock pulse frecuency SH_{CP} or ST_{CP}	30	52	–	24	–	20	–	MHz	4.5	Figs 7 and 8

: WAVEFORMS



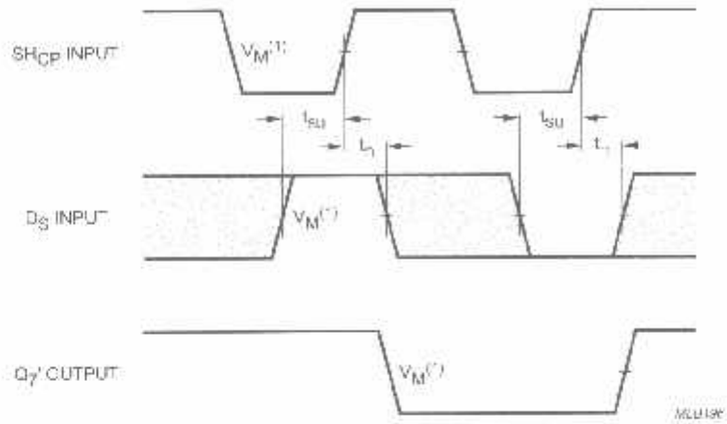
(1) HC: V_M = 50%; V_I = GND to V_{CC}.
HCT: V_M = 1.3 V; V_I = GND to 3 V.

Fig.7 Waveforms showing the clock (SH_{CP}) to output (Q_7') propagation delays, the shift clock pulse width and maximum shift clock frequency.



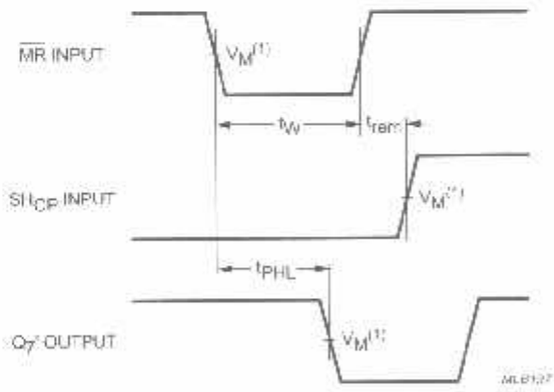
(1) HC: V_M = 50%; V_I = GND to V_{CC}.
HCT: V_M = 1.3 V; V_I = GND to 3 V.

Fig.8 Waveforms showing the storage clock (ST_{CP}) to output (Q_n) propagation delays, the storage clock pulse width and the shift clock to storage clock set-up time.



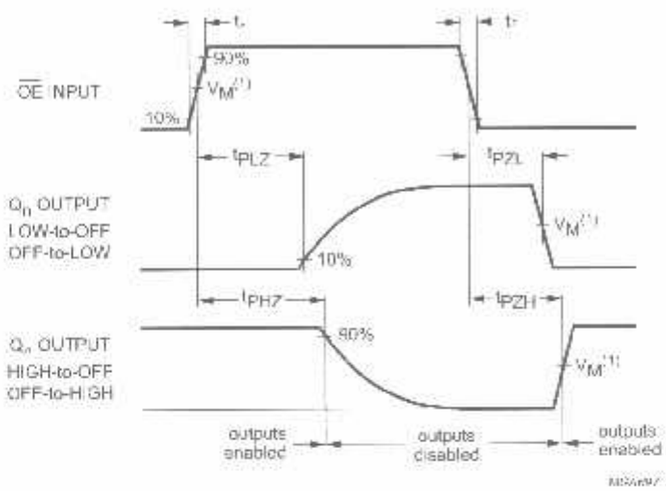
(1) HC: V_M = 50%; V_I = GND to V_{CC};
HCT: V_M = 1.3 V; V_I = GND to 3 V.

Fig.9 Waveforms showing the data set-up and hold times for the D_S input.



(1) HC: V_M = 50%; V_I = GND to V_{CC};
HCT: V_M = 1.3 V; V_I = GND to 3 V.

Fig.10 Waveforms showing the master reset ($\overline{MR}$) pulse width, the master reset to output (Q_7') propagation delay and the master reset to shift clock (SH_{CP}) removal time.

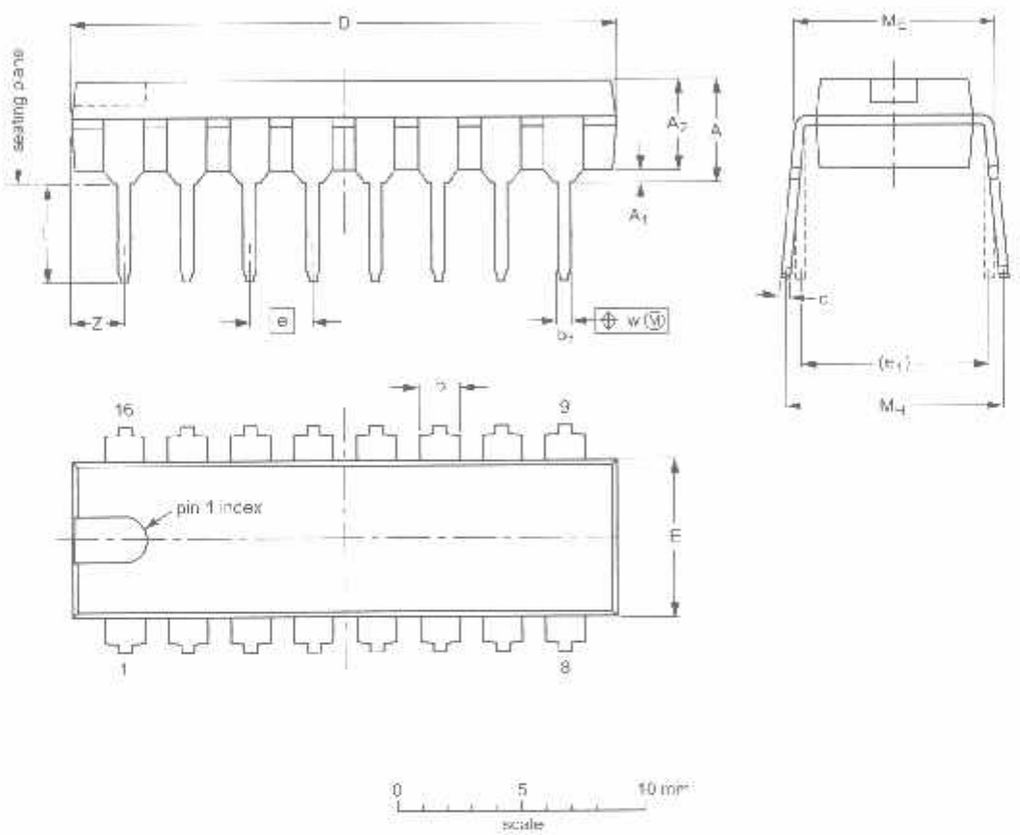


(1) HC: V_M = 50%; V_I = GND to V_{CC}.
HCT: V_M = 1.3 V; V_I = GND to 3 V.

Fig.11 Waveforms showing the 3-state enable and disable times for input $\overline{OE}$.

CKAGE OUTLINES

P16: plastic dual in-line package; 16 leads (300 mil); long body **SOT38-1**



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

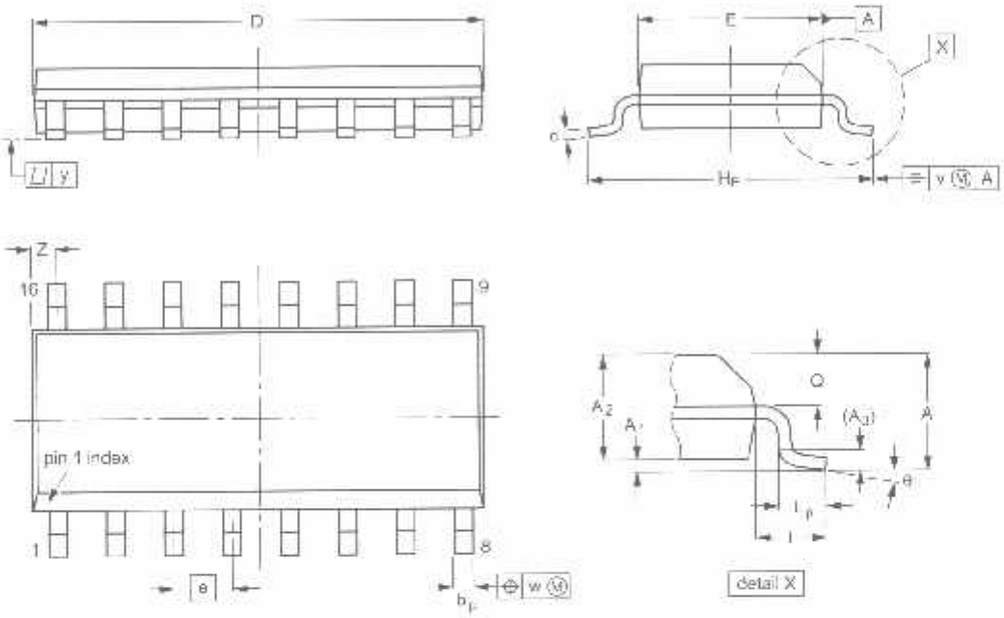
UNIT	A max.	A_1 min.	A_2 max.	b	b_1	c	$D^{(1)}$	$E^{(1)}$	e	e_1	L	M_E	M_H	w	$Z^{(1)}$ max.
mm	4.7	0.51	3.7	1.40 1.14	0.53 0.38	0.32 0.23	21.8 21.4	6.48 6.20	2.54	7.62	3.9 3.4	8.25 7.80	9.5 8.3	0.254	2.2
inches	0.19	0.020	0.15	0.055 0.045	0.021 0.015	0.013 0.009	0.86 0.84	0.26 0.24	0.10	0.30	0.15 0.13	0.32 0.31	0.37 0.33	0.01	0.087

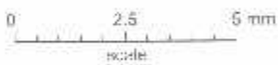
Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT38-1	050G09	MO-001AE				~~92-10-02~~ 95-01-19

)16: plastic small outline package; 16 leads; body width 3.9 mm **SOT109-1**





)IMENSIONS (inch dimensions are derived from the original mm dimensions)

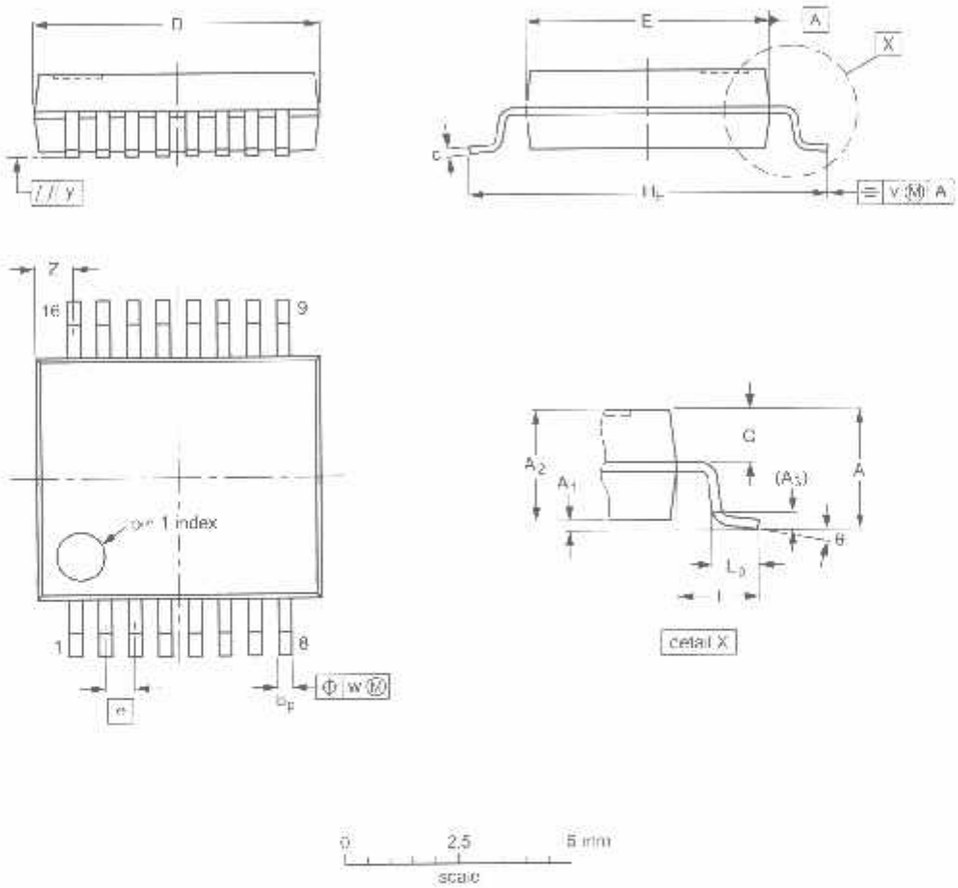
UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_E	L	L_p	Q	v	w	y	$Z^{(1)}$	θ
mm	1.75	0.25 0.10	1.45 1.25	0.25	0.49 0.36	0.25 0.19	10.0 9.8	4.0 3.8	1.27	6.2 5.8	1.05	1.0 0.4	0.7 0.6	0.25	0.25	0.1	0.7 0.3	8° 0°
inches	0.069	0.010 0.004	0.057 0.049	0.01	0.019 0.014	0.0100 0.0075	0.39 0.38	0.16 0.15	0.050	0.244 0.228	0.041	0.039 0.016	0.028 0.020	0.01	0.01	0.004	0.028 0.012	

Note

1. Plastic or meta protrusions of 0.15 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT109-1	076E07S	MS-012AC				~~95-01-23~~ 97-05-22

SOP16: plastic shrink small outline package; 16 leads; body width 5.3 mm | **SOT338-1**



DIMENSIONS (mm are the original dimensions)

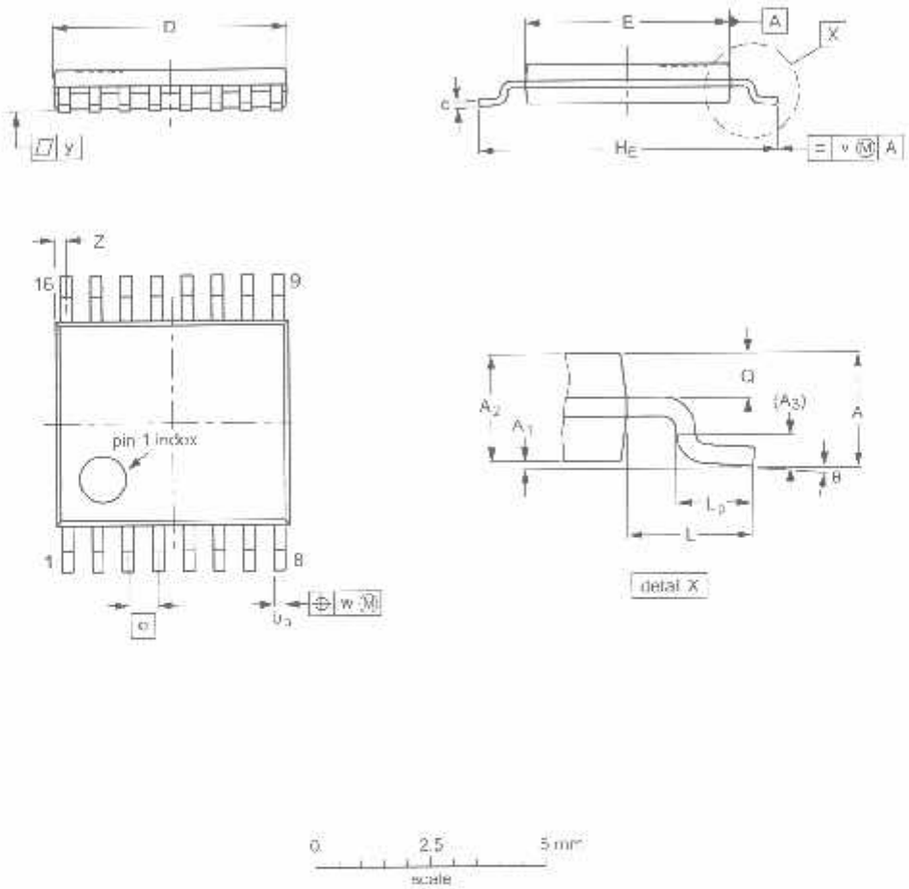
UNIT	A max.	A_1	A_2	A_3	b_p	c	D(1)	E(1)	e	H_E	L	L_p	Q	v	w	y	Z(1)	θ
mm	2.0	0.21 0.05	1.80 1.65	0.25	0.38 0.25	0.20 0.09	6.4 6.0	5.4 5.2	0.65	7.9 7.6	1.25	1.33 0.63	0.9 0.7	0.2	0.13	0.1	1.00 0.55	8° 0°

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT338-1		MO-150AC				~~94-01-11~~ 95-02-04

SOP16: plastic thin shrink small outline package; 16 leads; body width 4.4 mm **SOT403-1**



)IMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	D[1]	E[2]	e	H_L	L	L_p	Q	v	w	y	Z[1]	θ
mm	1.10	0.15 0.05	0.95 0.80	0.25	0.30 0.19	0.2 0.1	5.1 4.9	4.5 4.3	0.65	6.6 6.2	1.0	0.75 0.50	0.4 0.3	0.2	0.13	0.1	0.40 0.06	8° 0°

Notes

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.
2. Plastic interlead protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES			EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ		
SOT403-1		MO-153			~~94-07-12~~ 95-04-04

)LDERING

roduction

ore is no soldering method that is ideal for all IC ckages. Wave soldering is often preferred when ough-hole and surface mounted components are mixed one printed-circuit board. However, wave soldering is always suitable for surface mounted ICs, or for ntec-circuits with high population densities. In these ations reflow soldering is often used.

s text gives a very brief insight to a complex technology. nore in-depth account of soldering ICs can be found in *"Data Handbook IC26; Integrated Circuit Packages"* der code 9398 652 90011).

LDERING BY DIPPING OR BY WAVE

e maximum permissible temperature of the solder is 1 °C; solder at this temperature must not be in contact n the joint for more than 5 seconds. The total contact e of successive solder waves must not exceed econds.

e device may be mounted up to the seating plane, but temperature of the plastic body must not exceed the cified maximum storage temperature ($T_{stg\ max}$). If the ted-circuit board has been pre-heated, forced cooling y be necessary immediately after soldering to keep the perature within the permissible limit.

PAIRING SOLDERED JOINTS

ly a low voltage soldering iron (less than 24 V) to the d(s) of the package, below the seating plane or not e than 2 mm above it. If the temperature of the dering iron bit is less than 300 °C it may remain in tact for up to 10 seconds. If the bit temperature is ween 300 and 400 °C, contact may be up to 5 seconds.

SO, SSOP and TSSOP

REFLOW SOLDERING

Reflow soldering techniques are suitable for all SO, SSOP and TSSOP packages.

Reflow soldering requires solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the printed-circuit board by screen printing, stencilling or pressure-syringe dispensing before package placement.

Several techniques exist for reflowing; for example, thermal conduction by heated belt. Dwell times vary between 50 and 300 seconds depending on heating method.

Typical reflow temperatures range from 215 to 250 °C. Preheating is necessary to dry the paste and evaporate the binding agent. Preheating duration: 45 minutes at 45 °C.

WAVE SOLDERING

Wave soldering can be used for all SO packages. Wave soldering is **not** recommended for SSOP and TSSOP packages, because of the likelihood of solder bridging due to closely-spaced leads and the possibility of incomplete solder penetration in multi-lead devices.

If wave soldering is used - **and cannot be avoided for SSOP and TSSOP packages** - the following conditions must be observed:

- A double-wave (a turbulent wave with high upward pressure followed by a smooth laminar wave) soldering technique should be used.
- The longitudinal axis of the package footprint must be parallel to the solder flow and must incorporate solder thieves at the downstream end.

ren with these conditions:

Only consider wave soldering SSOP packages that have a body width of 4.4 mm, that is SSOP16 (SOT369-1) or SSOP20 (SOT266-1).

Do not consider wave soldering TSSOP packages with 48 leads or more, that is TSSOP48 (SOT362-1) and TSSOP56 (SOT364-1).

ıring placement and before soldering, the package must fixed with a droplet of adhesive. The adhesive can be plied by screen printing, pin transfer or syringe pensing. The package can be soldered after the hesive is cured.

ıximum permissible solder temperature is 260 °C, and ıximum duration of package immersion in solder is seconds, if cooled to less than 150 °C within econds. Typical dwell time is 4 seconds at 250 °C.

nildly-activated flux will eliminate the need for removal corrosive residues in most applications.

REPAIRING SOLDERED JOINTS

Fix the component by first soldering two diagonally-opposite end leads. Use only a low voltage soldering iron (less than 24 V) applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300 °C. When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds between 270 and 320 °C.

FINITIONS

ata sheet status	
ojective specification	This data sheet contains target or goal specifications for product development.
elim nary specification	This data sheet contains preliminary data; supplementary data may be published later.
oduct specification	This data sheet contains final product specifications.

miting values

niting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or ore of the limiting values may cause permanent damage to the device. These are stress ratings only and operation the device at these or at any other conditions above those given in the Characteristics sections of the specification not implied. Exposure to limiting values for extended periods may affect device reliability.

ıplication information

here application information is given, it is advisory and does not form part of the specification.

E SUPPORT APPLICATIONS

ese products are not designed for use in life support appliances, devices, or systems where malfunction of these ducts can reasonably be expected to result in personal injury. Philips customers using or selling these products for ı in such applications do so at their own risk and agree to fully indemnify Philips for any damages resulting from such ıroper use or sale.

Order this document by ULN2803/D

MOTOROLA

ULN2803
ULN2804

Octal High Voltage, High Current Darlington Transistor Arrays

OCTAL PERIPHERAL DRIVER ARRAYS

SEMICONDUCTOR TECHNICAL DATA

The eight NPN Darlington connected transistors in this family of arrays are ideally suited for interfacing between low logic level digital circuitry (such as TTL, CMOS or PMOS/NMOS) and the higher current/voltage requirements of lamps, relays, printer hammers or other similar loads for a broad range of computer, industrial, and consumer applications. All devices feature open collector outputs and free wheeling clamp diodes for transient suppression.

The ULN2803 is designed to be compatible with standard TTL families while the ULN2804 is optimized for 6 to 15 volt high level CMOS or PMOS.

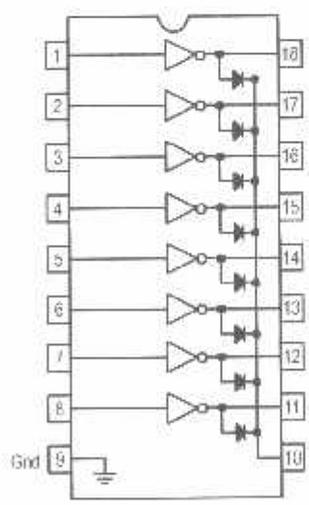
A SUFFIX
PLASTIC PACKAGE
CASE 707

MAXIMUM RATINGS (T_A = 25°C and rating apply to any one device in the package, unless otherwise noted.)

Rating	Symbol	Value	Unit
Output Voltage	V_O	50	V
Input Voltage (Except ULN2801)	V_I	30	V
Collector Current – Continuous	I_C	500	mA
Base Current – Continuous	I_B	25	mA
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C
Junction Temperature	T_J	125	°C

$R_{\theta JA}$ = 55°C/W
Do not exceed maximum current limit per driver.

PIN CONNECTIONS



ORDERING INFORMATION

Device	Characteristics		
	Input Compatibility	V_{CE}(Max)/I_C(Max)	Operating Temperature Range
ULN2803A ULN2804A	TTL, 5.0 V CMOS 6 to 15 V CMOS, PMOS	50 V/500 mA	T_A = 0 to +70°C

ELECTRICAL CHARACTERISTICS (T_A = 25°C, unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Output Leakage Current (Figure 1)		I_{CEX}				µA
(V_O = 50 V, T_A = +70°C)	All Types		–	–	100	
(V_O = 50 V, T_A = +25°C)	All Types		–		50	
(V_O = 50 V, T_A = +70°C, V_I = 6.0 V)	ULN2802			–	500	
(V_O = 50 V, T_A = +70°C, V_I = 1.0 V)	ULN2804		–	–	500	
Collector-Emitter Saturation Voltage (Figure 2)		$V_{CE(sat)}$				V
(I_C = 350 mA, I_B = 500 µA)	All Types		–	1.1	1.6	
(I_C = 200 mA, I_B = 350 µA)	All Types		–	0.95	1.3	
(I_C = 100 mA, I_B = 250 µA)	All Types		–	0.85	1.1	
Input Current – On Condition (Figure 4)		$I_{I(on)}$				mA
(V_I = 17 V)	ULN2802		–	0.82	1.25	
(V_I = 3.85 V)	ULN2803		–	0.93	1.35	
(V_I = 5.0 V)	ULN2804		–	0.35	0.5	
(V_I = 12 V)	ULN2804		–	1.0	1.45	
Input Voltage – On Condition (Figure 5)		$V_{I(on)}$				V
(V_{CE} = 2.0 V, I_C = 300 mA)	ULN2802		–	–	13	
(V_{CE} = 2.0 V, I_C = 200 mA)	ULN2803		–	–	2.4	
(V_{CE} = 2.0 V, I_C = 250 mA)	ULN2803		–	–	2.7	
(V_{CE} = 2.0 V, I_C = 300 mA)	ULN2803		–	–	3.0	
(V_{CE} = 2.0 V, I_C = 125 mA)	ULN2804		–	–	5.0	
(V_{CE} = 2.0 V, I_C = 200 mA)	ULN2804		–	–	6.0	
(V_{CE} = 2.0 V, I_C = 275 mA)	ULN2804		–		7.0	
(V_{CE} = 2.0 V, I_C = 350 mA)	ULN2804		–	–	8.0	
Input Current – Off Condition (Figure 3) (I_C = 500 µA, T_A = +70°C)	All Types	$I_{I(off)}$	50	100	–	µA
DC Current Gain (Figure 2) (V_{CE} = 2.0 V, I_C = 350 mA)	ULN2801	h_{FE}	1000	–	–	–
Input Capacitance		C_I	–	15	25	pF
Turn-On Delay Time (50% E_I to 50% E_O)		t_{on}	–	0.25	1.0	µs
Turn-Off Delay Time (50% E_I to 50% E_O)		t_{off}	–	0.25	1.0	µs
Clamp Diode Leakage Current (Figure 6) (V_R = 50 V)	T_A = +25°C T_A = +70°C	I_R		–	50 100	µA
Clamp Diode Forward Voltage (Figure 7) (I_F = 350 mA)		V_F	–	1.5	2.0	V

TEST FIGURES

(See Figure Numbers in Electrical Characteristics Table)

Figure 1.

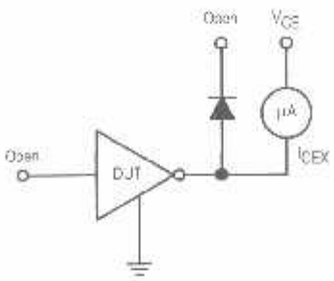


Figure 2.

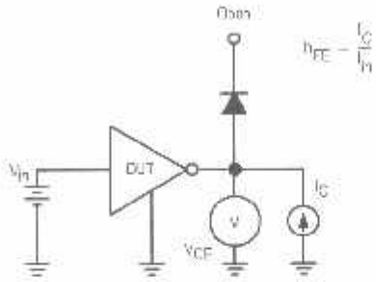


Figure 3.

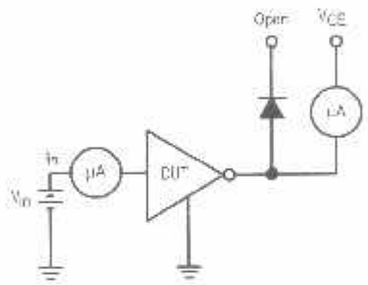


Figure 4.

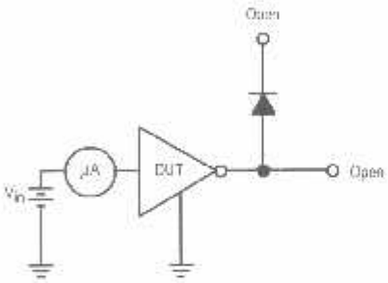


Figure 5.

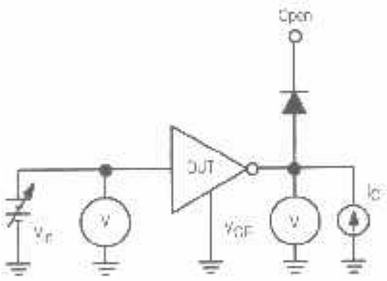


Figure 6.

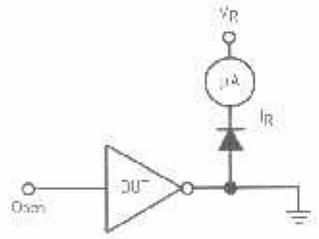
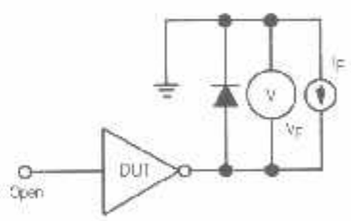


Figure 7.



TYPICAL CHARACTERISTIC CURVES – T_A = 25°C, unless otherwise noted

Output Characteristics

Figure 8. Output Current versus Saturation Voltage

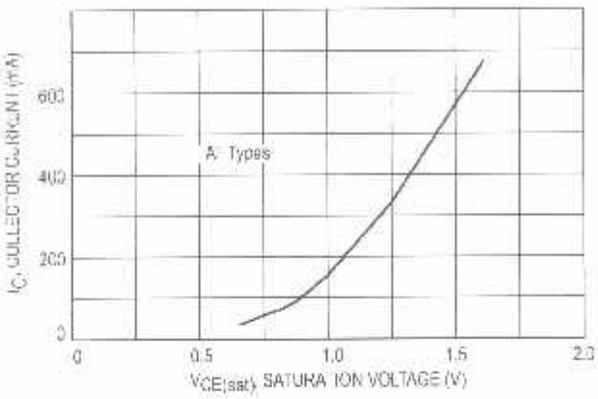
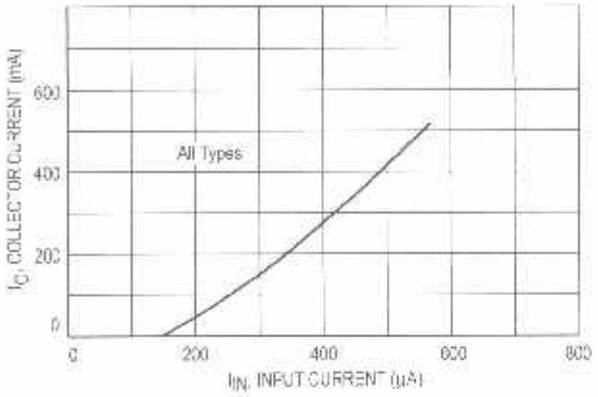


Figure 9. Output Current versus Input Current



Input Characteristics

Figure 10. ULN2803 Input Current versus Input Voltage

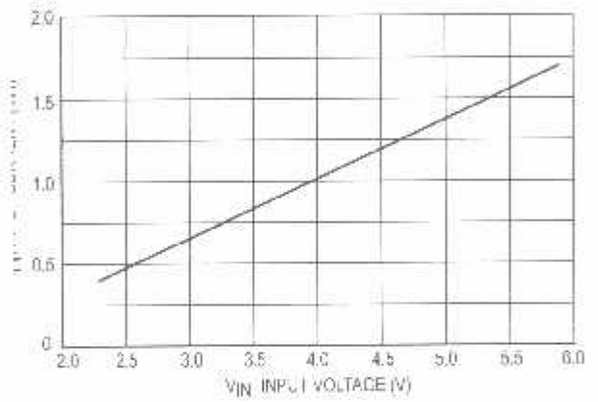


Figure 11. ULN2804 Input Current versus Input Voltage

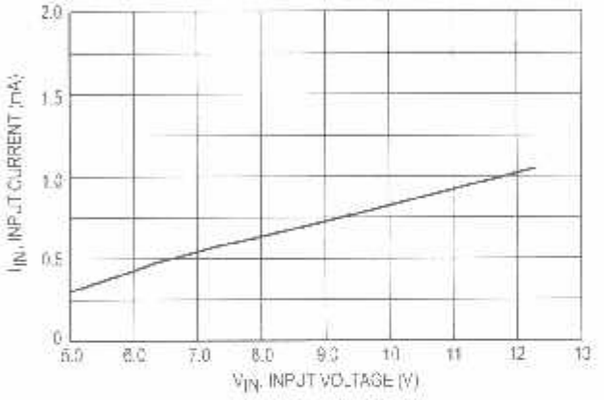
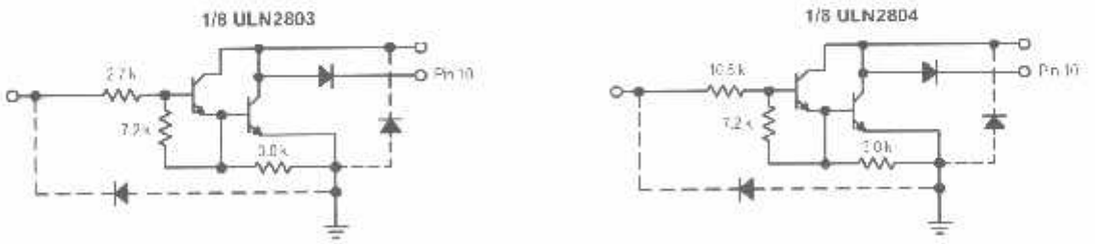
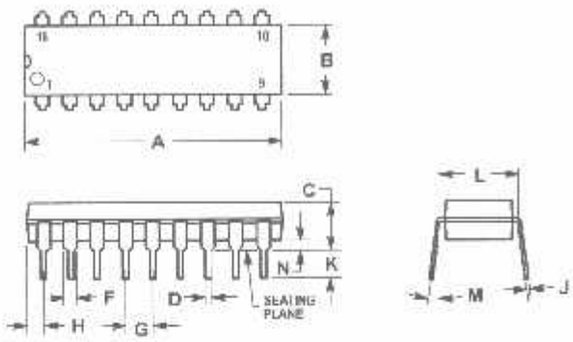


Figure 12. Representative Schematic Diagrams



OUTLINE DIMENSIONS

A SUFFIX
PLASTIC PACKAGE
CASE 707–02
ISSUE C



NOTES
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	22.22	23.24	0.875	0.915
B	6.10	6.60	0.240	0.260
C	3.56	4.57	0.140	0.180
D	0.36	0.56	0.014	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

How to reach us:
USA/EUROPE/Locations Not Listed: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036. 1–800–441–2447 or 602–303–5454

FAX: RMFAX0@email.sps.mot.com – TOUCHTONE 602–244–6609
INTERNET: Http://Design-NET.com

JAPAN: Nippon Motorola Ltd.; Tatsumi–SPD–JLDC, 6F Seibu–Butsuryu–Center, 3–14–2 Tatsumi Koto–Ku, Tokyo 135, Japan. 03–81–3521–8315

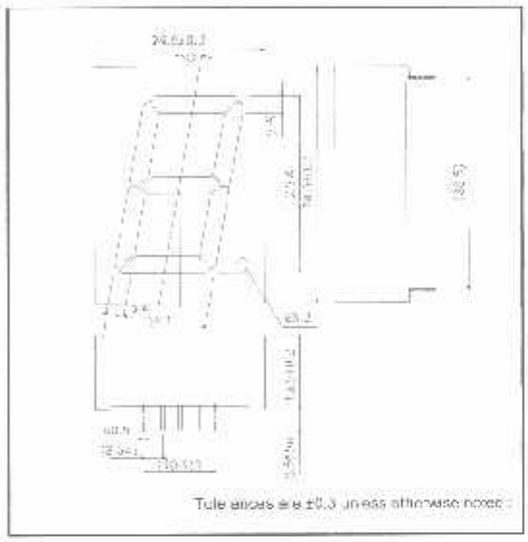
ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852–26629298

High efficiency, single-digit numeric displays

LA-101AK Series

The LA-101AK series are LED numerical displays designed to allow use even in bright locations. The height of the character is 25.4 mm, and two colors are available: red and green. These displays are designed for use in large numerical displays.

●**External Dimensions** (Unit : mm)



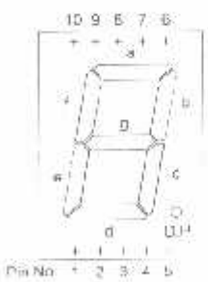
●**Features**

1) Height of character: 25.4 mm
2) Dimensions: 24 x 34 x 10.5 mm
3) A common anode configuration and a common cathode configuration are available for each color.
4) The package surface is painted black and the segments are colored the display color.
5) High luminance, clear display.

●**Selection guide**

Emitting color / Common	Red	Green
Anode	LA-101VA	LA-101MA
Cathode	LA-101VK	LA-101MK

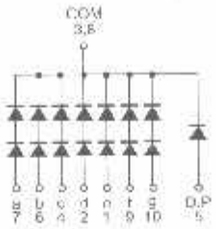
●**Pin assignments**



Pin No.	Function
1	Segment "e"
2	Segment "d"
3	Common
4	Segment "c"
5	D.P
6	Segment "b"
7	Segment "a"
8	Common
9	Segment "f"
10	Segment "g"

ROHM

●**Internal circuit schematic** (example of common cathode)



●**Absolute maximum ratings** (Ta=25°C)

Parameter	Symbol	Red LA-101VA / VK	Green LA-101MA / MK	Unit
Power dissipation	P_D	640	640	mW
Power dissipation	P_D / seg	85 (45)	85 (45)	mW
Forward current	I_F	15	20	mA
Peak forward current	I_{FP}	60*	60*	mA
Reverse voltage	V_R	3	3	V
Operating temperature	Topr	−25 to 75		°C
Storage temperature	Tstg	−30 to 85		°C

* Pulse width 1ms duty 1/5
( ) is D.P value

●**Electrical and optical characteristics** (Ta=25°C)

Parameter	Symbol	Conditions	Elements	Red Min.	Red Typ.	Red Max.	Green Min.	Green Typ.	Green Max.	Unit
Forward voltage	V_F	I_F=10mA	2	–	4.0*1	5.6*1	–	4.2*1	5.6*1	V
			1	–	2.0*2	2.8*2	–	2.1*2	2.8*2	
Reverse current	I_R	V_R=3V	–	–	–	100	–	–	100	μA
Peak wavelength	λ_P	I_F=10mA	–	–	650	–	–	563	–	nm
Spectral line half width	$\Delta\lambda$	I_F=10mA	–	–	40	–	–	40	–	nm

※Not designed for radiation resistance.
*1 The forward voltage and reverse current values are the guaranteed values per element.

●**Luminous intensity**

Color	λ_P	Type	Min.	Typ.	Max.	Unit
Red	650	LA-101VA LA-101VK	3.6	10		mcd
Green	563	LA-101MA LA-101MK	5.0	18	–	mcd

Note : Measured at I_F=10mA

Notes

- No technical content pages of this document may be reproduced in any form or transmitted by any means without prior permission of ROHM CO.,LTD.
- The contents described herein are subject to change without notice. The specifications for the product described in this document are for reference only. Upon actual use, therefore, please request that specifications to be separately delivered.
- Application circuit diagrams and circuit constants contained herein are shown as examples of standard use and operation. Please pay careful attention to the peripheral conditions when designing circuits and deciding upon circuit constants in the set.
- Any data, including, but not limited to application circuit diagrams information, described herein are intended only as illustrations of such devices and not as the specifications for such devices. ROHM CO.,LTD. disclaims any warranty that any use of such devices shall be free from infringement of any third party's intellectual property rights or other proprietary rights, and further, assumes no liability of whatsoever nature in the event of any such infringement, or arising from or connected with or related to the use of such devices.
- Upon the sale of any such devices, other than for buyer's right to use such devices itself, resell or otherwise dispose of the same, no express or implied right or license to practice or commercially exploit any intellectual property rights or other proprietary rights owned or controlled by
- ROHM CO., LTD. is granted to any such buyer
- Products listed in this document are no antiradiation design.

The products listed in this document are designed to be used with ordinary electronic equipment or devices (such as audio visual equipment, office-automation equipment, communications devices, electrical appliances and electronic toys).
Should you intend to use these products with equipment or devices which require an extremely high level of reliability and the malfunction of with would directly endanger human life (such as medical instruments, transportation equipment, aerospace machinery, nuclear-reactor controllers, fuel controllers and other safety devices), please be sure to consult with our sales representative in advance.

About Export Control Order in Japan

Products described herein are the objects of controlled goods in Annex 1 (Item 16) of Export Trade Control Order in Japan.
In case of export from Japan, please confirm if it applies to "objective" criteria or an "informed" (by MITI clause) on the basis of "catch all controls for Non-Proliferation of Weapons of Mass Destruction.

TO—92

1.EMITTER
2.BASE
3.COLLECTOR

1 2 3

ATURES

Power dissipation

P_{CV} : 0.625 W (Tamb=25℃)

Collector current

I_{CM} : -0.5 A

Collector-base voltage

$V_{(BR)CBO}$: -40 V

ECTRICAL CHARACTERISTICS (Tamb=25℃ unless otherwise specified)

Parameter	Symbol	Test conditions	MIN	TYP	MAX	UNIT
llector-base breakdown voltage	$V(BR)_{CBO}$	Ic= -100 μ A, I_E=0	-40			V
llector-emitter breakdown voltage	$V(BR)_{CEO}$	Ic= -0.1 mA, I_B=0	-20			V
itter-base breakdown voltage	$V(BR)_{EBO}$	I_E= -100 μ A, I_C=0	-5			V
llector cut-off current	I_{CBO}	V_{CB}=-40 V I_E=0			-0.1	μ A
llector cut-off current	I_{CEO}	V_{CE}=-20 V I_B=0			-0.2	μ A
itter cut-off current	I_{EBO}	V_{EB}=-5 V, I_C=0			-0.1	μ A
current gain(note)	$H_{FE(1)}$	V_{CE}=-1 V, I_C=-50 mA	64		300	
	$H_{FE(2)}$	V_{CE}=-1V, I_C=-500 mA	40			
llector-emitter saturation voltage	V_{CE}(sat)	I_C=-500 mA, I_B=-50 mA			-0.6	V
se-emitter saturation voltage	V_{BE}(sat)	I_C=-500mA, I_B=-50 mA			-1.2	V
se-emitter voltage	V_{BE}	I_E=-100mA			-1.4	V
ansition frequency	f_T	V_{CE}=-6 V, I_C=-20 mA f =30MHz	150			MHz

LASSIFICATION OF $H_{FE(1)}$

ank	D	E	F	G	H	I
ange	64-91	78-112	96-135	112-166	144-202	190-300

g Shing Computer Components Co., (H.K.) Ltd. Tel:(852)2341 9276 Fax:(852)2797 8153
epage: E-mail: ic@wingshing.com

ATURES

征

Power dissipation （最大耗散功率）

P_{CM} : 0.625 W（Tamb=25℃）

Collector current （最大集电极电流）

I_{CM} : 0.5 A

Collector-base voltage （集电极-基极击穿电压）

$V_{(BR)CBO}$: 45 V

TO 92

1.EMITTER
发 射 极
2.BASE
基 极
3.COLLECTOR
集 电 极

1 2 3

ECTRICAL CHARACTERISTICS（Tamb=25℃ unless otherwise specified）

电 特 性 （环境温度 除 非 另 有 规 定）

Parameter 参 数	Symbol 符 号	Test conditions 测 试 条 件	MIN 最小值	TYP 典型值	MAX 最大值	UNIT 单位
llector-base breakdown voltage 电 极 - - 基 极 击 穿 电 压	$V(BR)_{CBO}$	Ic= 100 μ A, I_E=0	45			V
llector-emitter breakdown voltage 电 极 - - 发 射 极 击 穿 电 压	$V(BR)_{CEO}$	Ic= 0. 1 mA, I_E=0	25			V
itter-base breakdown voltage 射 极 - - 基 极 击 穿 电 压	$V(BR)_{EBO}$	I_E= 100 μ A, I_C=0	5			V
lector cut-off current 电 极 - - 基 极 截 止 电 流	I_{CBO}	V_{CB}= 40 V I_E=0			0.1	μ A
lector cut-off current 电极--发射极截止电流	I_{CEO}	V_{CE}= 20 V I_B=0			0.1	μ A
itter cut-off current 射极--基极截止电流	I_{EBO}	V_{EB}= 5 V, I_C=0			0.1	μ A
current gain(note) 流 电 流 增 益	$H_{FE(1)}$	V_{CE}= 1 V, I_C= 50 mA	64		300	
	$H_{FE(2)}$	V_{CE}= 1V, I_C=500 mA	40			
lector-emitter saturation voltage 电 极 - - 发 射 极 饱 和 压 降	V_{CE}(sat)	I_C= 500 mA, I_B=50 mA			0.6	V
se-emitter saturation voltage 极--发射极饱和压降	V_{BE}(sat)	I_C= 500mA, I_B= 50 mA			1.2	V
se-emitter voltage 极--发射极正向电压	V_{BE}	I_E=100mA			1.4	V
nsition frequency 征 频 率	f_T	V_{CE}= 6 V, I_C= 20 mA f =30MHz	150			MHz

ASSIFICATION OF $H_{FE(1)}$ （分类）

nk 次	D	E	F	G	H	I
nge 围	64-91	78-112	96-135	112-166	144-220	190-300

Shing Computer Components Co., (H.K.)Ltd. Tel:(852)2341 9276 Fax:(852)2797 8153
page: E-mail: ic@wingshing.com

Order this document by MC3361B/D

MOTOROLA

Low Power Narrowband FM IF

MC3361B

The MC3361B includes an Oscillator, Mixer, Limiting Amplifier, Quadrature Discriminator, Active Filter, Squelch, Scan Control and Mute Switch. This device is designed for use in FM dual conversion communications equipment.

- Operates from 2.0 to 8.0 V Supply
- Low Drain Current 3.9 mA Typical @ V_{CC} = 4.0 Vdc
- Excellent Sensitivity: Input Limiting Voltage –3.0 dB = 2.6 μV Typical
- Low Number of External Parts Required

Operating Frequency Up to 60 MHz

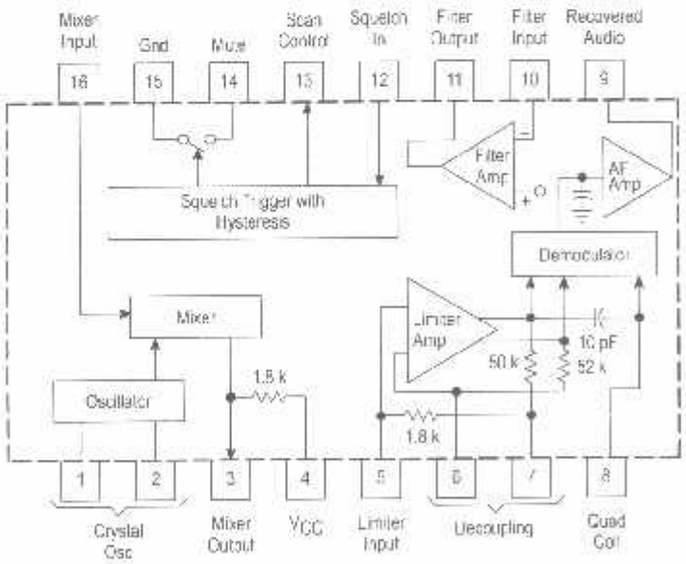
LOW POWER NARROWBAND FM IF

SEMICONDUCTOR TECHNICAL DATA

P SUFFIX
PLASTIC PACKAGE
CASE 648
Not Recommended for New Design

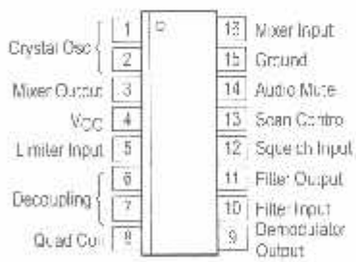
D SUFFIX
PLASTIC PACKAGE
CASE 751B
(SO–16)

Representative Block Diagram



This device contains 92 active transistors.

PIN CONNECTIONS



(Top View)

ORDERING INFORMATION

Device	Operating Temperature Range	Package
MC3361BD	T_A = –30 to 70°C	SO–16
MC3361BP		Plastic DIP

MAXIMUM RATINGS (T_A = 25°C, unless otherwise noted.)

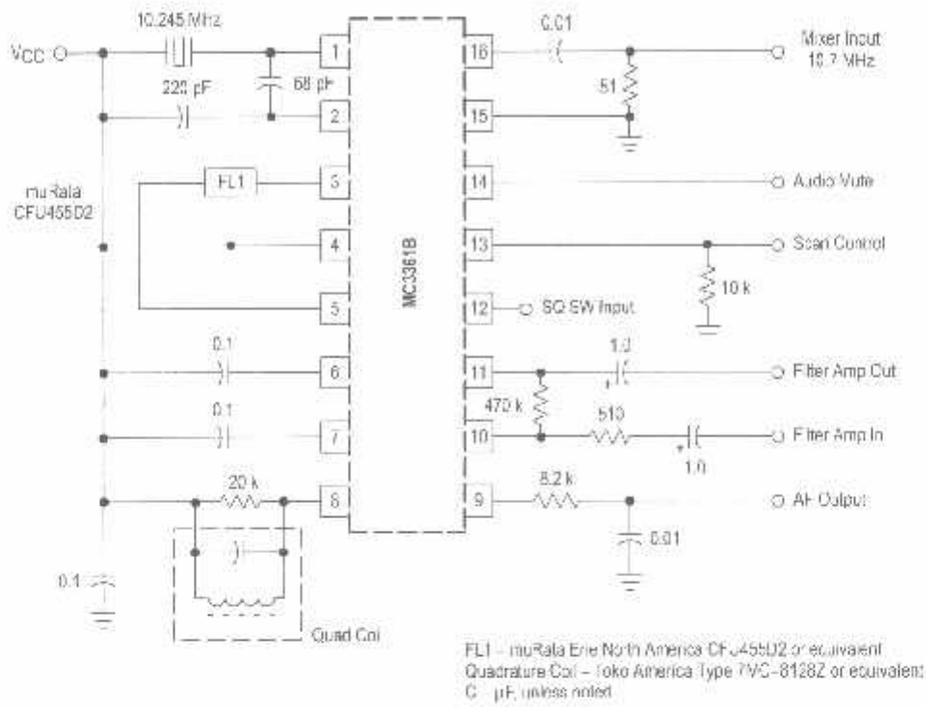
Rating	Pin	Symbol	Value	Unit
Power Supply Voltage	4	V_{CC}(max)	10	Vdc
Operating Supply Voltage Range	4	V_{CC}	2.0 to 8.0	Vdc
Detector Input Voltage	8		1.0	V_{pp}
Input Voltage (V_{CC} = 4.0 V)	16	V_{16}	1.0	V_{rms}
Mute Function	14	V_{14}	–0.5 to 5.0	V_{pk}
Junction Temperature	–	T_J	150	°C
Operating Ambient Temperature Range		T_A	–30 to 70	°C
Storage Temperature Range	–	T_{stg}	–65 to 150	°C

NOTES: 1. Maximum Ratings are those values beyond which damage to the device may occur. Functional operation should be restricted to the limits in the Electrical Characteristics tables or Pin Descriptions section.
2. ESD data available upon request.

ELECTRICAL CHARACTERISTICS (V_{CC} = 4.0 Vdc, f_o = 10.7 MHz, Δf = ± 3.0 kHz, f_{mod} = 1.0 kHz, T_A = 25°C, unless otherwise noted.)

Characteristic	Pin	Min	Typ	Max	Unit
Drain Current (No Signal)	4				mA
Squelch "Off"		2.9	3.9	4.9	
Squelch "On"		4.4	5.4	6.4	
Recovered Audio Output Voltage (V_{in} = 10 mVrms)	9	130	160	200	mVrms
Input Limiting Voltage (–3.0 dB Limiting)	16	–	2.6	6.0	μV
Total Harmonic Distortion	9	–	0.36		%
Recovered Output Voltage (No Input Signal)	9	60	120	250	mVrms
Drop Voltage AF Gain Loss	9	–3.0	–0.6	–	dB
Detector Output Impedance		–	450	–	Ω
Filter Gain (10 kHz) (V_{in} = 0.3 mVrms)	–	40	50		dB
Filter Output Voltage	11	1.0	1.3	1.6	Vdc
Mute Function Low	14	–	30	50	Ω
Mute Function High	14	1.0	11	–	MΩ
Scan Function Low (Mute "Off") (V_{12} = 1.0 Vdc)	13	–	0	0.4	Vdc
Scan Function High (Mute "On") (V_{12} = Gnd)	13	3.0	3.5	–	Vdc
Trigger Hysteresis	–	–	45	100	mV
Mixer Conversion Gain	3	–	28	–	dB
Mixer Input Resistance	16	–	3.3	–	kΩ
Mixer Input Capacitance	16	–	2.2	–	pF

Figure 1. Test Circuit



FL1 – muRata Erie North America CFU455D2 or equivalent
Quadrature Coil – Toko America Type 7VC-8128Z or equivalent
C = µF unless noted

Figure 2. Audio Output, Distortion versus Supply Voltage

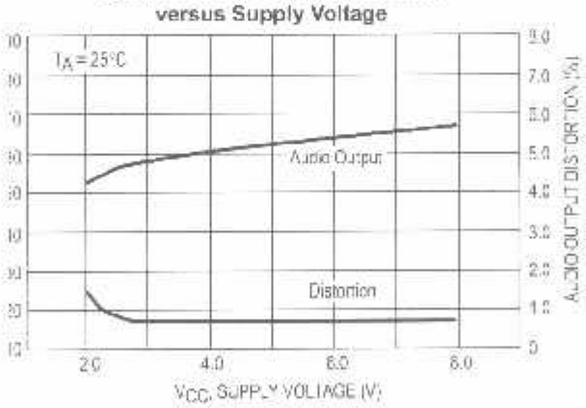


Figure 3. Audio Output, Distortion versus Temperature

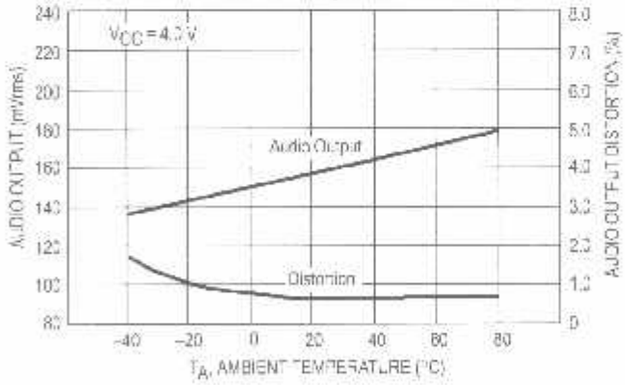


Figure 5. Input Limiting Voltage

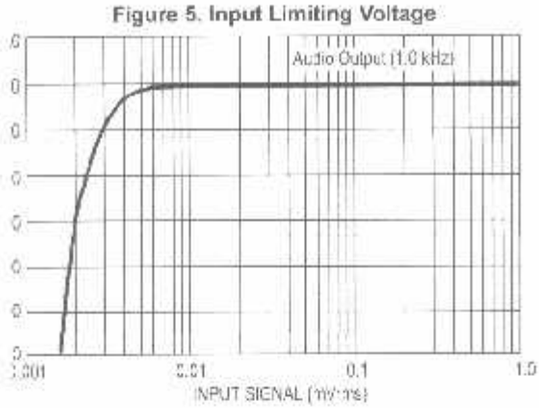


Figure 6. Overall Gain, Noise and AM Rejection

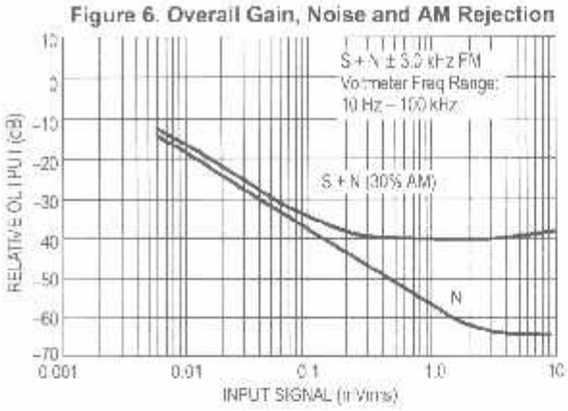


Figure 7. Filter Amp Response

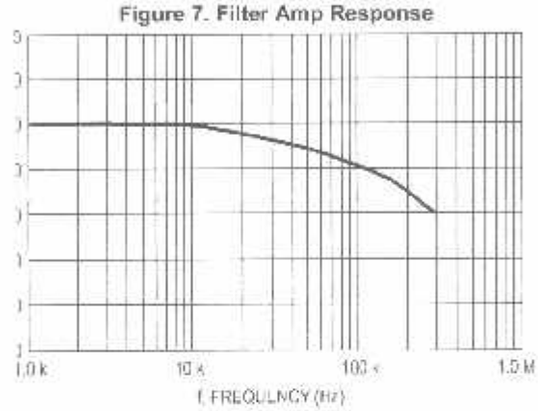


Figure 8. Filter Amp Gain

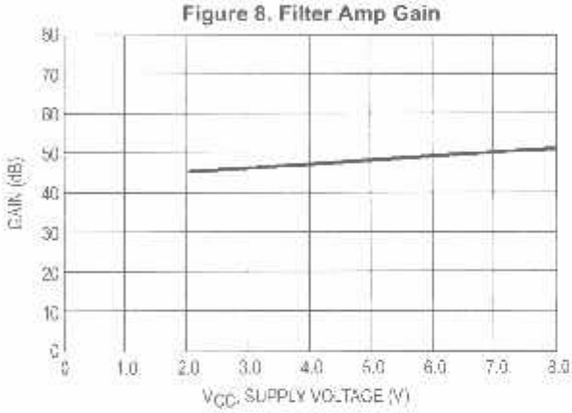


Figure 9. Supply Current

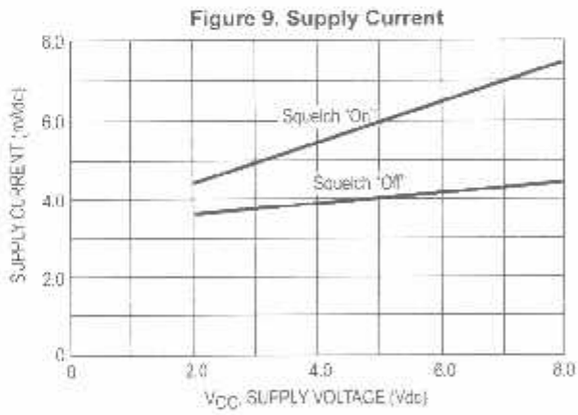


Figure 5. Input Limiting Voltage

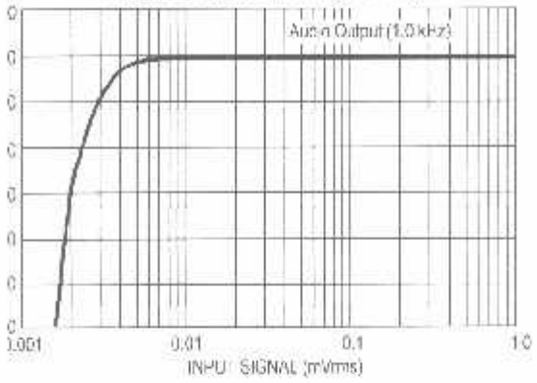


Figure 6. Overall Gain, Noise and AM Rejection

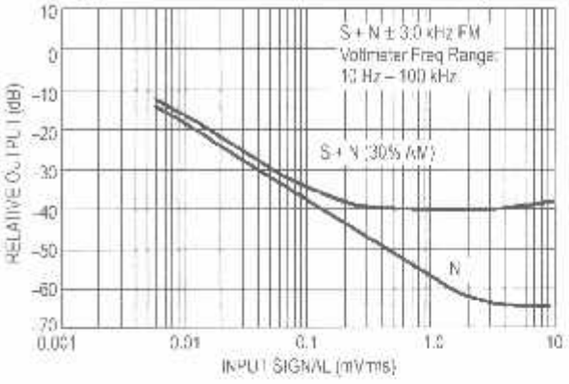


Figure 7. Filter Amp Response

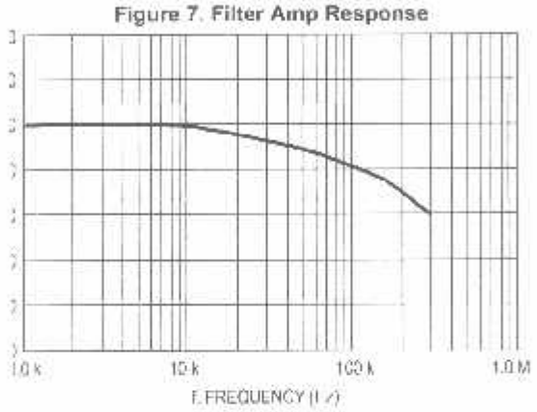


Figure 8. Filter Amp Gain

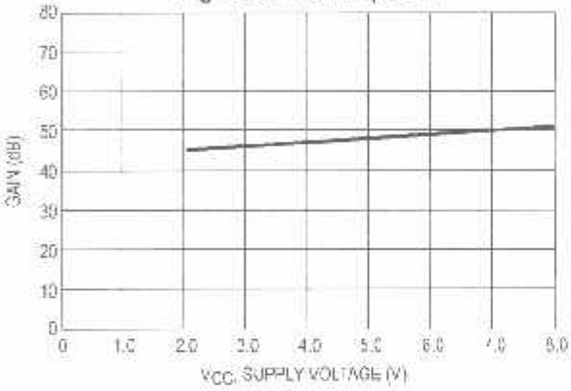


Figure 9. Supply Current

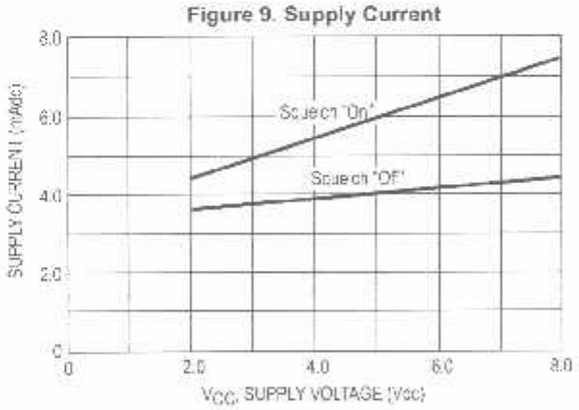
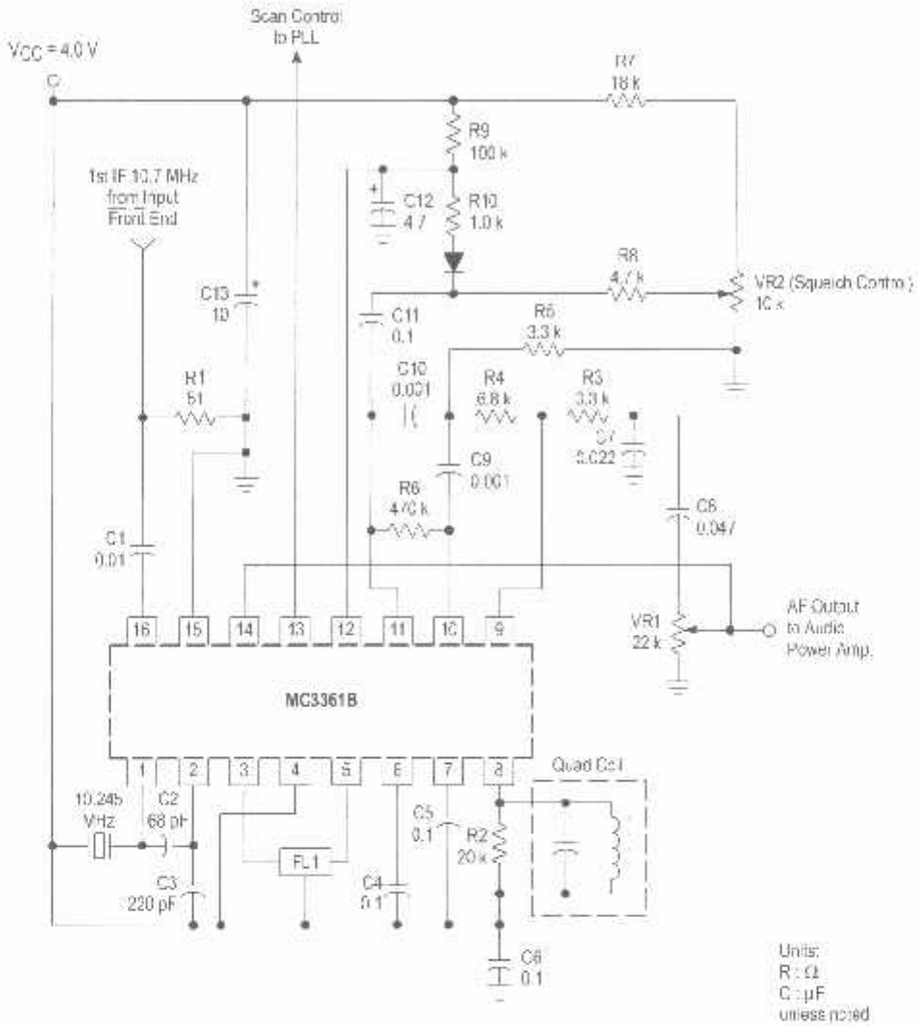


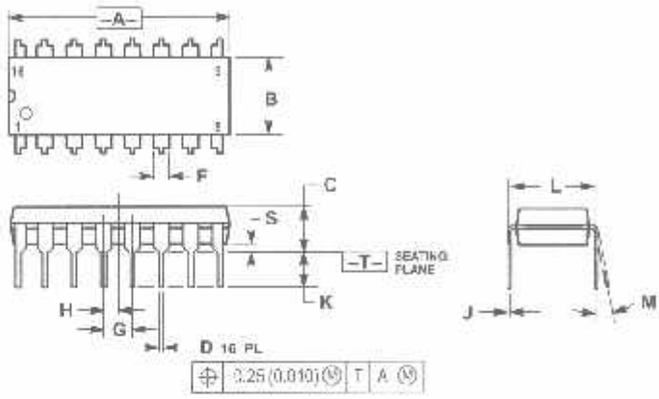
Figure 10. Simplified Application



FL1 – muRata Erie North America Type CFU455D2 or equivalent
Quadrature Coil – Toko America Type 7MC-8126Z or equivalent

OUTLINE DIMENSIONS

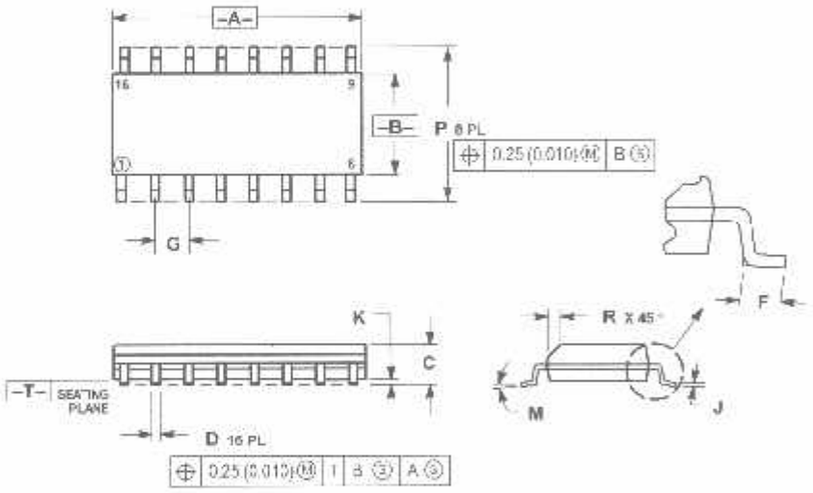
P SUFFIX
PLASTIC PACKAGE
CASE 648–08
ISSUE R



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
4. DIMENSIONS A AND B DO NOT INCLUDE MOLD FLASH.
5. ROUNDED CORNERS OPTIONAL.

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.740	0.770	18.80	19.55
B	0.250	0.270	6.35	6.85
C	0.145	0.175	3.69	4.44
D	0.015	0.021	0.38	0.53
F	0.040	0.70	1.02	1.77
G	0.100 BSC		2.54 BSC	
H	0.050 BSC		1.27 BSC	
J	0.008	0.015	0.21	0.38
K	0.110	0.130	2.80	3.30
L	0.295	0.305	7.50	7.74
M	0°	10°	0°	10°
S	0.020	0.040	0.51	1.01

D SUFFIX
PLASTIC PACKAGE
CASE 751B–05
(SO–16)
ISSUE J



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION.
4. MAXIMUM MOLD PROTRUSION 0.15 (0.006) PER SIDE.
5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.127 (0.005) TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	9.80	10.00	0.386	0.393
B	3.80	4.00	0.150	0.157
C	1.35	1.75	0.054	0.068
D	0.35	0.49	0.014	0.019
F	0.40	1.25	0.016	0.049
G	1.27 BSC		0.050 BSC	
J	0.19	0.25	0.008	0.009
K	0.10	0.25	0.004	0.009
M	0°	7°	0°	7°
P	5.80	6.20	0.228	0.244
R	0.25	0.50	0.010	0.019

How to reach us:
USA/EUROPE/Locations Not Listed: Motorola Literature Distribution;
P.O. Box 5405, Denver, Colorado 80217. 1–303–675–2140 or 1–800–441–2447

JAPAN: Motorola Japan Ltd.; SPD, Strategic Planning Office, 141,
4–32–1 Nishi–Gotanda, Shinagawa–ku, Tokyo, Japan. 81–3–5487–8488

Customer Focus Center: 1–800–521–6274

Mfax™: RMFAX0@email.sps.mot.com – TOUCHTONE 1–602–244–6609
Motorola Fax Back System – US & Canada ONLY 1–800–774–1848
http://sps.motorola.com/mfax/

HOME PAGE: http://motorola.com/sps/

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Centre,
2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.
852–26668334

MOTOROLA

BC440 · BC460
BC441 · BC461

COMPLEMENTARY SILICON AF MEDIUM POWER AMPLIFIERS & SWITCHES

E BC440, BC441, BC460, BC461 ARE SILICON PLANAR ITAXIAL TRANSISTORS FOR AF DRIVERS AND OUTPUTS, WELL AS FOR SWITCHING APPLICATIONS UP TO 1 PERE. THE BC440, BC441 ARE NPN AND ARE COMPLE-NTARY TO THE PNP BC460, BC461 RESPECTIVELY.

CASE TO-39

BSOLUTE MAXIMUM RATINGS (For p-n-p devices, voltage and current values are negative)		BC440(NPN) BC460(PNP)	BC441(NPN) BC461(PNP)
llector-Emitter Voltage ($R_{BE} \leq 100\,\Omega$)	V_{CER}	50V	75V
llector-Emitter Voltage ($I_B=0$)	V_{CEO}	40V	60V
itter-Base Voltage	V_{EBO}	5V	5V
llector Current	I_C	1A	
llector Peak Current	I_{CM}	2A	
tal Power Dissipation ($T_C \leq 25^oC$, $V_{CE} \leq 10V$)	P_{tot}	10W	
($T_A \leq 25^oC$)		1W	
erating Junction & Storage Temperature	T_j, T_{stg}	-55 to 200°C	

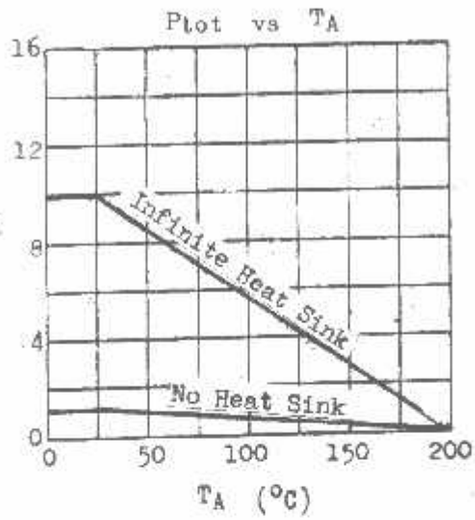
ECTRICAL CHARACTERISTICS ($T_A=25^oC$ unless otherwise noted)

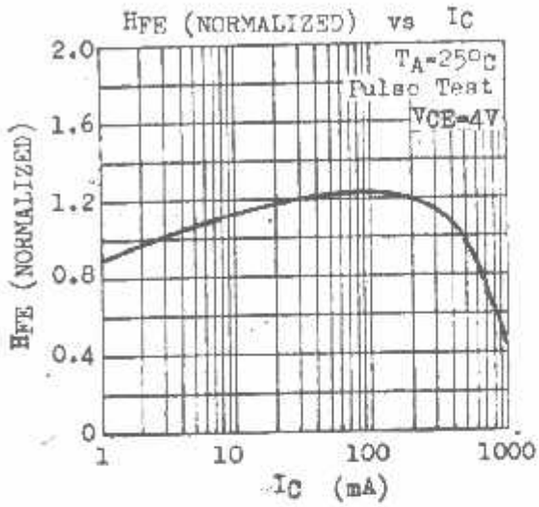
PARAMETER	SYMBOL	BC440 BC460 MIN	BC440 BC460 MAX	BC441 BC461 MIN	BC441 BC461 MAX	UNIT	TEST CONDITIONS
llector-Emitter Breakdown Voltage	LV_{CEO} *	40		60		V	I_C=100mA I_B=0
itter-Base Breakdown Voltage	BV_{EBO}	5		5		V	I_E=0.1mA I_C=0
llector Cutoff Current	I_{CBO}		100		100	nA	V_{CB}=40V I_E=0
llector Cutoff Current	I_{CER}		10			µA	V_{CE}=50V R_{BE}=100Ω
					10	µA	V_{CE}=70V R_{BE}=100Ω
llector-Emitter Saturation Voltage	$V_{CE(sat)}$*		1		1	V	I_C=1A I_B=0.1A
se-Emitter Saturation Voltage	$V_{BE(sat)}$*		1.5		1.5	V	I_C=1A I_B=0.1A
. Current Gain	H_{FE} *	40	250	40	250	V	I_C=500mA V_{CE}=4V
Group 4		40	70	40	70		
Group 5		60	130	60	130		
Group 6		115	250	115	250		
		20					I_C=1A V_{CE}=2V
rrent Gain-Bandwidth Product	f_T	50		50		MHz	I_C=50mA V_{CE}=4V
llector-Base Capacitance	Cob		25		25	pF	V_{CB}=10V I_E=0 f=1MHz

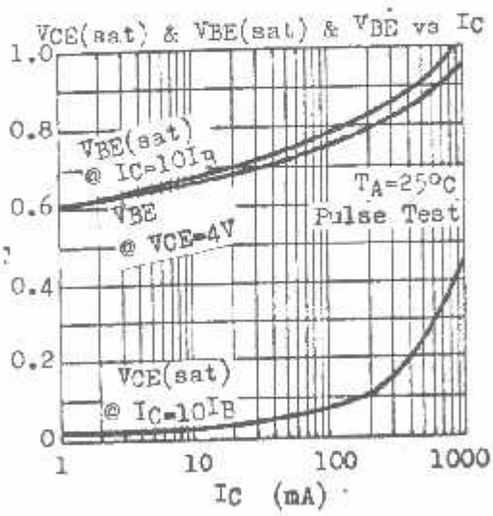
Pulse Test : Pulse Width=0.3mS, Duty Cycle=1%

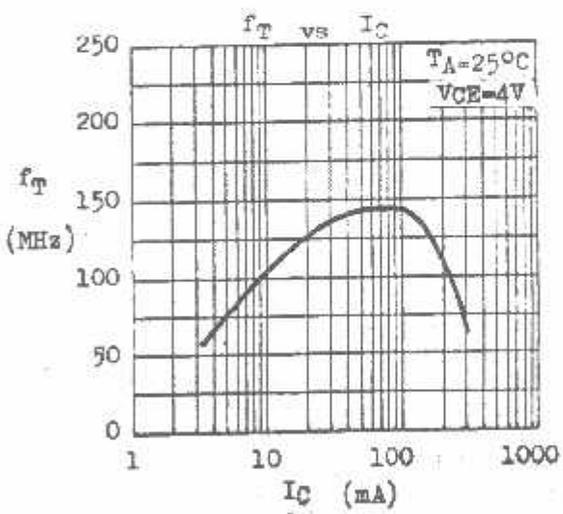
MICRO ELECTRONICS LTD. 38 HUNG TO ROAD, KWUN TONG, HONG KONG. KWUN TONG P. O. BOX69477 CABLE ADDRESS "MICROTRON" TELEX 43510 FAX: 3-410321 TELEPHONE:- 3-430181-6 ~~3-893360, 3-892429~~

TYPICAL CHARACTERISTICS









INSTITUT TEKNOLOGI NASIONAL
JL. BENDUNGAN SIGURA-GURA 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Andrie Dian Rinaldy
Nim : 0017078
Masa Bimbingan : 10-Sep-2005 s/d 13-Mar-2006
Judul Skripsi : Perancangan dan pembuatan alat penampil informasi, jam dan kalender digital pada tempat umum berbasis mikrokontroller AT89S8252

NO	Tanggal	Uraian	Paraf Pembimbing
1.	01-02-2006	Revisi penulisan pada Bab I	
2.	03-02-2006	Revisi penulisan letak judul tabel 2.3 pada Bab II	
3.	08-02-2006	Revisi gambar dan lebar *paragraph* penulisan pada tabel 3.3 Bab III	
4.	11-02-2006	Revisi hasil *error* pengujian alat pada Bab IV	
5.	15-02-2006	Revisi isi kesimpulan dan saran pada Bab V	
6.	19-02-2006	Revisi lebar *paragraph* penulisan Pada daftar pustaka	
7.	22-02-2006	Revisi makalah seminar hasil skripsi	
8.	14- 03-2006	ACC makalah seminar hasil Skripsi	
9.	19-03-2006	ACC Laporan Skripsi	

Malang, 17 Maret 2006
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT

Form S-4a