SKRIPSI

PERANCANGAN DAN PEMBUATAN BOR OTOMATIS BERBASIS MIKROKONTROLLER AT89S8252

Disusun oleh :

ARDIAN DWI YUNANTO

NIM: 02.17.070

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK ELEKTRONIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

SEPTEMBER 2007

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN BOR OTOMATIS BERBASIS MIKROKONTROLLER AT89S8252

SKRIPSI

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

Oleh:
ARDIAN DWI YUNANTO
NIM: 02.17.070

Diperiksa dan disetujui

Dosen Pembimbing I	Dosen Pembimbing II
Ir. Eko Nurcahyo	**I Komang Samawirata ST, MT**
NIP. Y. 1028700172	**NIP.P. 103010036**

Ketua Jurusan Teknik Elektro S -1

Ir. F. Yudi Limpraptono, MT
NIP. Y. 1038900209

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK ELEKTRONIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2007

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama	: ARDIAN DWI YUNANTO
N I M	: 02.17.070
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika S-1
Judul Skripsi	: Perancangan Dan Pembuatan Bor Otomatis Berbasis Mikrokontroller AT89S8252

Dipertahankan dihadapan tim penguji skripsi jenjang strata satu (S–1) pada :
Hari : Kamis
Tanggal : 6 Septembeer 2007
Dengan Nilai : 79 (B+)

Panitia Ujian Skripsi

Ketua

Ir. Mochtar Asroni. MSME
NIP.Y. 1018100036

Sekretaris

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

Anggota Penguji

Penguji I

Ir. Teguh Herbasuki, MT
NIP. Y. 1038900209

Penguji II

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

ABSTRAKSI

"PERANCANGAN DAN PEMBUATAN BOR OTOMATIS BERBASIS MIKROKONTROLLER AT89S8252"Ardian Dwi Yunanto,02.17.070.Teknik Elektro S-1/ Konsentrasi Teknik Elektronika S-1, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.
Dosen Pembimbing : I. Ir. Eko Nurcahyo
II. I Komang Samawirata ST, MT

Kata Kunci : Bor, *PC, Microcontroller AT89S8252, RS 232*

Permasalahan yang sering terjadi pada suatu *system* pengeboran sangat beragam.Dari keberagaman masalah yang sederhana hingga masalah kompleks tersebut bisa berdampak pada hasil suatu pengeboran maupun operator jika tidak mengetahui bagaimana cara mengatasinya.Berangkat dari permasalahan ini,munculah suatu gagasan untuk membuat suatu alat (bor otomatis) yang bertujuan untuk memberikan kemudahan, efisiensi waktu dan meminimalisasi kesalahan- kesalahan pada suatu pengeboran sehingga mendapatkan suatu hasil pengeboran dengan mutu yang baik dan layak.

Metode pembuatan bor otomatis ini terdiri dari dua bagian yaitu *PC* sebagai input dan penentuan titik pengeboran dan alat sebagai peraga system pengeboran ini.*Interface* alat dihubungkan ke *PC* melalui *COM1* (port komunikasi serial).Tentukan titik yang akan dibor melalui tampilan board pada *PC*.Setelah adanya penandaan dan target pengeboran berwarna merah dan dilanjutkan dengan eksekusi, proses selanjutnya dkirim melalui *RS 232* ke *microcontroller.Microcontroller* mengaktifkan mekanik-mekanik dalam sistem untuk melakukan pengeboran.Setelah proses selesai, *microcontroller* mengirimkan data hasil progress ke *PC* untuk ditampilkan kembali.

Dalam pengujian alat ini,tidak pernah mengalami kesalahan dalam pengolahan data, baik data yang diproses maupun data yang ditampilkan.Pada rangkaian *opto interupt* memiliki kesalahan (error) sebesar 2,36 %, untuk rangkaian *limit switch* sebesar 1,6 %.

pengembangan dari pembaca semuanya akan bisa menyempurnakan alat dan penyusunan skripsi ini menjadi lebih sempurna.

Penyusun berharap agar laporan skripsi ini dapat berguna bagi pembaca pada umumnya dan penyusun pada khususnya.

Malang, Agustus 2007

Penyusun

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pengeboran merupakan salah satu kegiatan yang dilakukan banyak orang, dan masih banyak yang menggunakan peralatan yang dikerjakan secara manual atau dikontrol oleh pekerja (manusia) sehingga faktor *human error* cukup besar, salah satunya adalah peralatan yang digunakan untuk membuat lubang-lubang atau pengeboran.Pengebor tentunya menginginkan mutu dari pengeboran itu baik, maka dengan memanfaatkan *devices* dalam bidang elektronika untuk mengurangi *human error* dan memperingan pekerjaan dibuatlah alat ini sehingga kontrol dalam pengeboran tidak lagi dilakukan oleh pekerja tapi dilakukan oleh *devices* elektronika.

1.2. Rumusan Masalah

Mengacu pada permasalahan yang diuraikan pada latar belakang, maka rumusan masalah dapat ditekankan pada:

1. Apakah mikrokontroller AT89S8252 dapat dimanfaatkan sebagai pengontrol sistem ini?
2. Bagaimana merencanakan dan membuat perangkat keras dan perangkat lunak dapat mengendalikan system dengan mikrokontroller AT89S8252?.

1.3. Batasan masalah

1. Dimensi maksimum yang akan dibor mempunyai ukuran tertentu (270 mm x 210 mm x 2 mm)
2. Media yang akan dibor adalah kayu atau papan.
3. Titik-titik yang akan dibor sudah disediakan dan tinggal memilih.

1.4. Tujuan

Tujuan penulisan skripsi :

1. Merancang dan membuat bor otomatis dengan mengaplikasikan mikrokontroller AT89S8252 sebagai pengontrol sistem pengeboran.
2. Memperingan suatu proses pengeboran.

1.5. Metodelogi Perencanaan

Metode yang digunakan dalam penyusunan laporan akhir ini adalah sebagai berikut:

1. Metode kepustakaan

   Metode ini dilakukan dengan cara mencari literatur yang mendukung sebagai landasan teori dari rangkaian yang akan dibuat.

2. Metode Bimbingan

   Metode ini sebagai petunjuk dan pengarahan dari para pembimbing untuk membantu hal-hal yang belum dimengerti dan sebagai sarana untuk mempertimbangkan segala sesuatu yang dihadapi sehubungan dengan proses pembuatan alat ini.

1.6. Kontribusi

Diharapkan,alat ini mempunyai kontribusi yang cukup besar guna memperingan suatu proses pekerjaan yang masih dilakukan kebanyakan orang. Mungkin dalam perkembangannya, alat ini dapat digunakan pada industri – industri walaupun masih dalam taraf kecil.

1.7. Sistematika Penulisan

Bab I : Menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, metode,dan sistematika penulisan.

Bab II : Menjelaskan tentang Teori dasar yang berisi tentang prinsip dasar dari mikrokontroller dan rangkaian pendukung.

Bab III : Menjelaskan tentang perencanaan dan pembuatan alat.

Bab IV : Menjelaskan tentang pengujian alat yang telah direalisasikan.

Bab V : Menjelaskan tentang kesimpulan dan saran.

Bab VI : Berisi tentang gambar rangkaian. daftar komponen dan data IC yang digunakan.

BAB II

LANDASAN TEORI

2.1. Pendahuluan

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. Bagian-bagian yang akan dibahas diantaranya :

- Mikrokontroller AT89S8252
- Motor (DC dan Stepper)
- RS MAX 232
- Optocoupler
- Transistor
- Limit Switch
- Dioda
- Driver push-pull L 293 D dan L 298 N

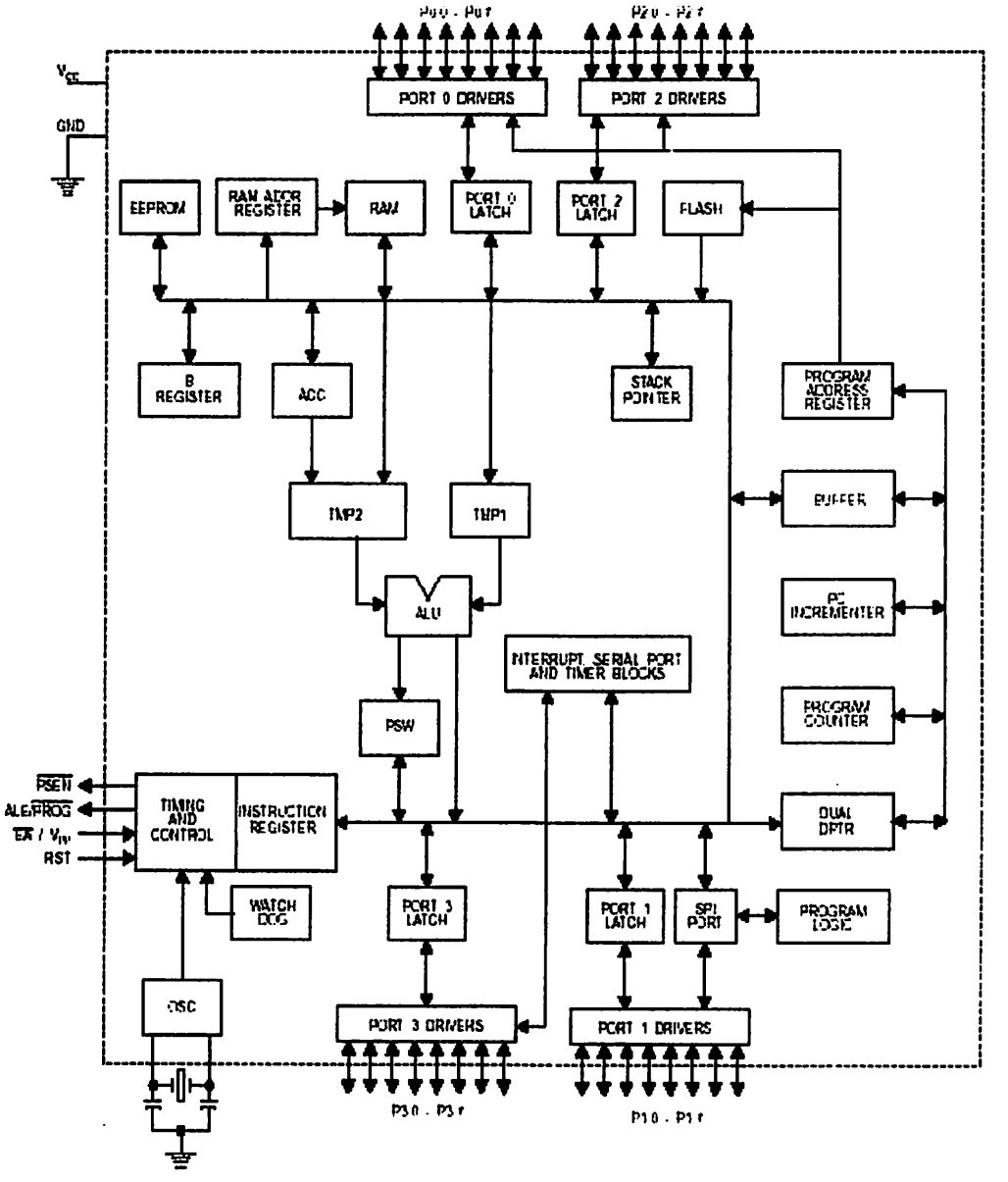
2.2. Mikrokontroller AT89S8252

2.2.1. Teori Umum

SCM (*single on-chip*) adalah suatu mikrokontroler lengkap yang dibuat dalam sebuah IC yang mempunyai struktur seperti CPU, Osilator, Timer, RAM, EPROM dan Buffer (alamat, data dan input-output). Mikrokontroler AT89S8252 merupakan mikrokontroler 8-bit kompatibel dengan standar industri *MCS-51™* baik dari segi pemrogaman maupun kaki tiap pin. Mikrokontroler AT89S8252 mempunyai 8 Kbyte PEROM (*Programmable and Erasable Read Only Memory*).

Pada dasarnya mikrokontroler terdiri atas mikroprosesor, timer, counter, perangkat I/O dan internal memori. Mikrokontroler termasuk perangkat yang mudah didesain dalam bentuk chip tunggal (*single chip*). Pada dasarnya mikrokontroler mempunyai fungsi yang sama dengan mikroprosesor yaitu untuk mengontrol suatu kerja sistem.

Diagram blok dari IC AT89S8252 seperti gambar 2.1 dibawah ini.



Gambar 2.1 Diagram Blok AT89S8252[11]

Di dalam mikrokontroller juga terdapat CPU, ALU, PC, SP dan register seperti dalam mikroprosesor, tetapi juga ditambah dengan perangkat-perangkat lain seperti RAM, ROM, PIO, SIO, Counter dan sebuah rangkaian Clock.

RAM pada dasarnya merupakan suatu flip-flop yang dapat diset/direset, sifat ini membuat RAM dapat dibaca atau ditulis. Karena transistor yang menyusun flip-flop membutuhkan suatu tegangan DC agar tetap aktif, maka sel RAM akan kehilangan datanya bila power dimatikan. Hal ini dalam dunia komputer disebut bersifat *volatile*. Sedangkan ROM memiliki beberapa tipe diantaranya yaitu *Mask Programmable* ROM , *Fusible Link* PROM, *UV Light Erasable* PROM (EPROM) dan EEPROM. Tidak seperti RAM, data yang ada didalam ROM tidak akan hilang bila power dimatikan. Hal ini disebut bersifat *non-volatile*, suatu pemrogram khusus yang diperlukan untuk menulis data ke ROM. Karena non-volatilitasnya maka ROM sering dipetakan ke alamat reset dari mikrokontroler, dalam hal ini sangat diperlukan pada saat melakukan *booting*.

Mikrokontroler didesain dengan instruksi-instruksi lebih luas dan 8-bit instruksi yang digunakan untuk 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 memiliki banyak fitur yang menguntungkan. Dipakainya *downloadable flash memory* memungkinkan mikrokontroler ini bekerja sendiri tanpa diperlukan tambahan chip lainnya. Sementara flash memorinya 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. Di samping itu pula mikrokontroler AT89S8252 membutuhkan daya rendah dan memiliki performen yang tinggi.

AT89S8252 juga mempunyai 2 buah *Power Saving Mode* yang dapat diatur melalui *software*, yaitu IDE Mode yang akan menghentikan CPU sebagai RAM, dimana Timer/Counter, Serial Port dan Interrupt Sistem tetap berfungsi. Sedangkan *Power Down Mode* yang akan menyimpan data di RAM dan akan menahan osilator untuk tidak mengaktifkan chip yang lain sampai terjadi reset secara *hardware*.

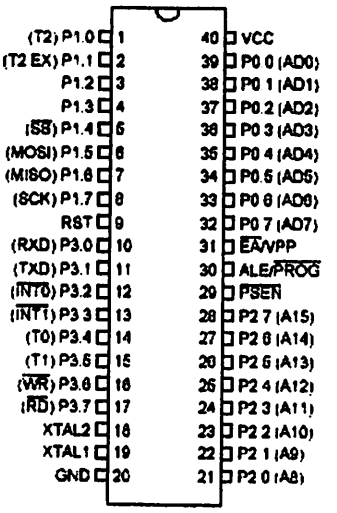
Adapun secara umum, konfigurasi yang dimiliki mikrokontroler AT89S8252 adalah sebagai berikut :

- Sebuah CPU 8-bit dengan menggunakan teknologi dari Atmel
- 8 Kbyte *Downloadable Flash Memory*
- 2 Kbyte EEPROM
- Sebuah port serial dengan kontrol *full duplex* UART (*Universal Asynchronous Receiver Transmitter*)
- 256 byte RAM internal
- 32 I/O yang dapat dipakai semuanya
- 3 buah Timer/Counter 16-bit
- 6 Sumber Interrupt
- SPI Serial Interface
- *Programmable Watchdog Timer*
- *Dual data pointer*
- Frekuensi kerja 0 – 24 MHz
- Tegangan operasi 2,7 V sampai 6 V
- *Power-of flag*

- Kemampuan melaksanakan operasi perkalian, pembagian dan operasi Boolean (bit)

2.2.2. Konfigurasi Pin Pada Mikrokontroller AT89S8252

Konfigurasi kaki-kaki mikrokontroler AT89S8252 terdiri dari 40 pin seperti yang terlihat pada gambar berikut ini.



Gambar 2.2 Konfigurasi Pin-Pin AT89S8252[11]

Adapun fungsi dari tiap pin akan dijelaskan sebagai berikut :

- Pin 1 – 8

  Port 1 yang terdiri atas pin 1 – 8 merupakan saluran masukan/keluaran dua arah dengan internal *pull-up* dan mempunyai fungsi khusus seperti yang terlihat pada tabel

Tabel 2.1 Fungsi Alternatif Port 1[1]

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

- Pin 9

  RST merupakan saluran 2 masukan untuk mereset mikrokontroler dengan cara memberi masukan logika tinggi

- Pin 10 – 17

  Port 3 yang terdiri atas pin 10 – 17 merupakan saluran masukan/keluaran dua arah dengan internal *pull-up* dan mempunyai fungsi khusus seperti yang terlihat pada tabel

Tabel 2.2 Fungsi Alternatif Port 3[1]

Port Pin	Fungsi
P3.0	RXD (port serial input)
P3.1	TXD (port serial output)
P3.2	INT0 (interrupt eksternal 0)
P3.3	INT1 (interrupt eksternal 1)
P3.4	T0 (input eksternal timer 0)
P3.5	T1 (input eksternal timer 1)
P3.6	WR (menulis data ke memori eksternal)
P3.7	RD (membaca data dari memory eksternal)

- Pin 18 – 19

  $XTAL_1$ dan $XTAL_2$ merupakan saluran untuk mengatur pewaktuan sistem. Untuk pewaktuan dapat menggunakan pewaktuan internal maupun eksternal. Pin ini dihubungkan dengan kristal bila menggunakan osilator internal. $XTAL_1$ merupakan masukan ke rangkaian osilator internal sedangkan $XTAL_2$ merupakan keluaran dari rangkaian osilator internal

- Pin 20

  V_{SS} merupakan hubungan ke ground dari rangkaian

- Pin 21 – 28

  Port 2 yang terdiri atas pin 21 – 28 merupakan saluran masukan/keluaran dua arah dengan internal *pull-up*. Port ini mengeluarkan 8-bit bagian alamat tinggi ($A_8 - A_{15}$) selama pengambilan instruksi dari memori program eksternal dan pengambilan data memori eksternal 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 mengeluarkan alamat rendah ($A_0 - A_7$) dalam port 0, selama proses baca/tulis memori eksternal. Frekuensi ALE adalah 1/6 kali frekuensi osilator dan dapat digunakan sebagai pewaktu. Pin ini juga berfungsi sebagai saluran program selama dilakukan pemrograman jika menggunakan memori program internal

- 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 V jika menggunakan memori program internal. Dapat diberikan logika rendah (ground) atau logika tinggi (+5V), jika diberikan logika tinggi maka mikrokontroler akan mengakses program dari ROM internal (EEPROM/*Flash Memory*), dan jika diberikan logika rendah maka mikrokontroler akan mengakses program dari memori eksternal

- Pin 32 – 39

  Port 0 yang terdiri atas pin 32 – 39 merupakan saluran masukan/keluaran dua arah tanpa internal *pull-up*. Port 0 merupakan saluran alamat rendah ($A_0 - A_7$) yang dimultipleks dengan saluran bus data ($D_0 - D_7$)

- Pin 40

  V_{CC} merupakan saluran masukan untuk catu daya positif sebesar 5 volt DC dengan toleransi kurang lebih 1 %

2.2.3. Masukan dan Keluaran

I. Untuk saluran masukan dan keluaran terdapat 4 buah port yang masing-masing 8-bit. Saluran ini bersifat dua arah (*bidirectional*) yang berarti dapat difungsikan sebagai masukan/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 2.2. AT89S8252 juga memiliki fitur tambahan yang terdapat pada port 1 seperti dalam tabel 2.1.

2.2.4. Organisasi Memory

2.2.4.1. Data Memori (EEPROM) dan RAM

Berbeda dengan mikrokontroler standar MCS-51, untuk AT89S8252 terdapat 2 Kbytes dalam EEPROM untuk penyimpanan data dan 256 byte untuk RAM. Dibagian atas 128 byte RAM ditempati paralel untuk SFR. Bagian atas 128 byte mempunyai alamat sama dengan SFR, tetapi secara fisik terpisah dari SFR.

EEPROM on-chip ini diakses dengan mengeset bit EEMEN pada register WMCON pada alamat 96_H. Alamat EEPROM ini adalah 000_H sampai $7FF_H$. Dan selama EEPROM memprogram, yang dibaca dari EEPROM akan mengambil byte yang sedang ditulis dengan melengkapi MSB. Instruksi *movx* digunakan untuk mengakses EEPROM internal ini. Bit EEMWE pada register WMCON harus diset ke "1" sebelum sembarang lokasi pada EEPROM dapat ditulisi. Program pengguna harus mereset 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, tapi jika bit ini berlogika tinggi berarti penulisan sudah selesai dan penulisan lain dapat dimulai lagi.

Sedangkan RAM yang ada pada mikrokontroler AT89S8252 berkapasitas 256 byte dan kompatibel dengan RAM yang ada pada mikrokontroler standar MCS-51. Pada lower 128-bit lokasi memori dapat dibagi menjadi 3 bagian, yaitu :

1. Empat Bank Register

   Setiap bank terdiri dari 8 register (R0-R7), sehingga jumlah register untuk keempat bank register (bank 0 – bank 3) menjadi 32 buah register yang menempati ruang alamat 00_H – $1F_H$. Untuk mengaktifkan salah satu bank register dapat dilakukan dengan mengatur RS0-RS1 melalui pengaturan pada PSW (*Program Status Word*)

2. Bit Addressable

   Terdiri dari 16-bit yang berada pada alamat 20_H – $2F_H$. Masing- masing dari 128-bit lokasi ini dapat dialamati secara langsung yaitu dari 00_H – $7F_H$

3. Scratch Pad Area

   Terdiri dari 80-byte yang menempati alamat 30_H – $7F_H$ yang dapat diaiamati secara langsung dan dapat digunakan untuk keperluan umum (*General Purpose* RAM). Misalnya digunakan untuk lokasi *stack.*

Tabel 2.3 Pengaturan RS0-RS1 Untuk Select Register Bank[1]

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

2.2.4.2. SFR (*Special Function Register*)

Area memori AT89S8252 disebut dengan SFR (*Special Function Register*) yang merupakan register dengan tugas khusus. Tidak semua address digunakan sebagai SFR, hanya addres tertentu seperti yang dijelaskan oleh tabel 2.4 berikut ini.

Tabel 2.4 128 Byte Special Function Register[1]

SYMBOL	NAME	ADDRESS
ACC	ACCUMULATOR	0E0H
B	B REGISTER	0F0H
PSW	PROGRAM STATUS WORD	0D0H
SP	STACK POINTER	81H
DPTR	DATA POINTER 2 BYTE	
DPL	LOW BYTE	82H
DPH	HIGH BYTE	83H
P0	PORT 0	80H
P1	PORT 1	90H
P2	PORT 2	0A0H
P3	PORT 3	080H
IP	INTERRUPT PERIORITY CONTROL	088H
IE	INTERRUPT ENABLE CONTROL	0ABH
TMOD	TIMER/COUNTER MODE CONTROL	89H
TCON	TIMER/COUNTER CONTROL	88H
+TCON	TIMER/COUNTER 2 CONTROL	0CBH
TH0	TIMER/COUNTER 0 HIGH CONTROL	8CH
TL0	TIMER/COUNTER 0 LOW CONTROL	8DH

TH1	TIMER/COUNTER 1 HIGH CONTROL	8DH
TL1	TIMER/COUNTER 1 LOW CONTROL	8CH
TH2	TIMER/COUNTER 2 HIGH CONTROL	0CDH
TL2	TIMER/COUNTER 2 LOW CONTROL	0CCH
RCAP2H	T/C 2 CAPTURE REG. HIGH BYTE	0CBH
+RCAP2L	T/C 2 CAPTURE REG. LOW BYTE	0CAH
SCON	SERIAL CONTROL	98H
SBUF	SERIAL DATA BUFFER	99H
PCON	POWER CONTROL	87H

Akses pembacaan dari semua address akan diwujudkan dalam bentuk random data dan penulisan akses diwujudkan dalam bentuk (efek) tidak tentu. SFR pada mikrokontroler AT89S8252 kompatibel dengan mikrokontroler keluarga MCS-51 dan memiliki alamat 80_H sampai FF_H sehingga terdapat 128-bit lokasi alamat untuk SFR. Namun demikian, pada mikrokontroler ini tidak berarti memiliki SFR sebanyak 128 buah.

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 Control dengan alamat $0C8_H$), T2MOD (Timer 2 Mode dengan alamat $0C9_H$), WMCON (*Watchdog and Memory Control* Register dengan alamat 96_H), SPCR (SPI Control Register dengan alamat $D5_H$), SPSR (SPI Status Register dengan alamat AA_H), SPDR (SPI Data Register dengan alamat 86_H). Gambar berikut akan menjelaskan letak masing-masing SFR.

Gambar 2.3. Peta Letak SFR Pada Mikrokontroller AT89S8252[1]

0F8H								0FFH
0F0H	B 00000000							0F7H
0E8H								0EFH
0E0H	ACC 00000000							0E7H
0D8H								0DF H
0D0H	PSW 00000000					SPCR 000001XX		0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000		0CF H
0C0H								0C7H
0B8H	IP XX000000							0BFH
0B0H	P3 11111111							0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX					0AFH
0A0H	P2 11111111							0A7H
98H	SCON 00000000	SBUF XXXXXXXX						9FH
90H	P1 11111111						WMCON 00000010	97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000		8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

2.2.4.2.1. SFR untuk Timer 2

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

2.2.4.2.2. SFR untuk Watchdog dan Memori

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

2.2.4.2.3.SFR pengontrol SPI

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

mikrokontroler AT89S8252. Fitur ini meliputi :

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

b. Operasi Master atau Slave

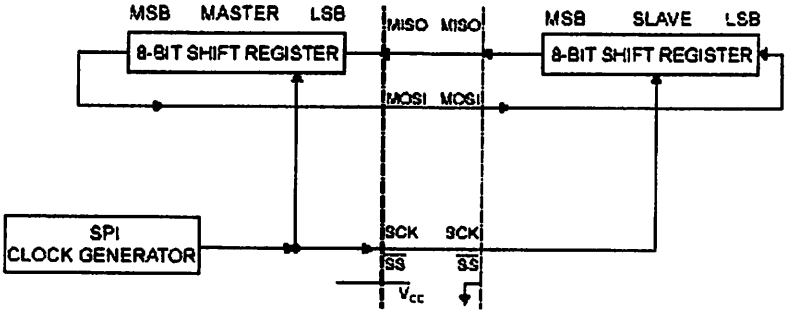
c. Frekuensi maksimum 6 MHz

d. 4-bit rate terprogram

e. Sistem data transfer MSB dahulu atau LSB

f. Write Collision Flag Protection

Gambar berikut menunjukkan hubungan antara CPU master dan slave.



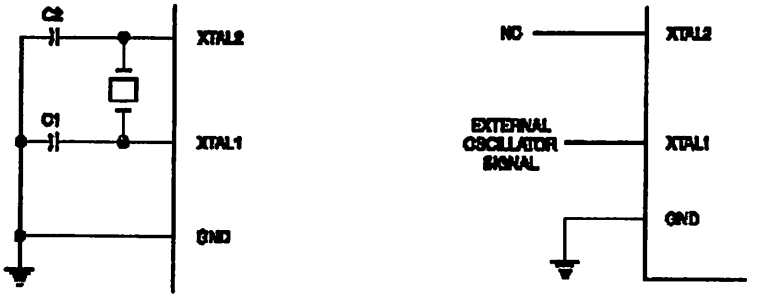
Gambar 2.4. Koneksi SPI Master dan Slave[1]

2.2.5. Osilator

Jantung dari AT89S8252 adalah rangkaian yang membangkitkan pulsa clock yang mensinkronkan semua operasi internal. Mikrokontroler AT89S8252 memiliki osilator internal (*on-chip oscilator*) yang dapat digunakan sebagai sumber pewaktu (clock) bagi CPU. Untuk menggunakan osilator internal diperlukan sebuah kristal atau resonator keramik antara pin $XTAL_1$ dan $XTAL_2$ dan sebuah kapasitor ke ground. $XTAL_2$ dan $XTAL_1$ secara berurutan merupakan input dan output dari sebuah inverting amplifier yang dapat dikonfigurasikan penggunaannya sebagai *on-chip oscillator* seperti yang ditunjukkan pada gambar 2.5a.

Untuk memberikan IC AT89S8252 sumber clock eksternal, maka pin $XTAL_2$ dibiarkan tidak berhubungan dengan sumber clock eksternal dan $XTAL_1$ dihubungkan

dengan sumber clock eksternal seperti pada gambar 2.5b.



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

Gambar 2.5. Karakteristik Osilator[1]

2.2.6. Timer dan Counter

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

Jika difungsikan sebagai Timer, maka akan menggunakan sistem clock sebagai sumber masukan pulsanya. Jika sebagai Counter (pencacah), maka akan menggunakan pulsa dari luar (eksternal) sebagai masukan pulsanya. Pada port 3 terdapat fungsi khusus yaitu T0 (masukan luar untuk Timer/Counter 0) dan T1 (masukan luar untuk Timer/Counter 1). Pemilihan mode Timer/Counter dikontrol oleh register TMOD. Dengan memberikan nilai tertentu pada regiater TMOD, dapat dipilih mode operasi untuk Timer/Counter 0 dan Timer/Counter 1 seperti terlihat dalam tabel.

Tabel 2.5 Mode Operasi Timer/Counter 0 dan 1[1]

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

Pada mikrokontroler terdapat tambahan Timer 2. Timer yang lain adalah Timer 0 dan Timer 1. Timer 2 ini merupakan Timer/Counter 16-bit dan memiliki 3 mode operasi yaitu *capture, auto-reload* (*up-down counting*) dan *baund 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 (*increament*) tiap siklus mesin. Karena siklus mesin terdiri dari 12 periode osilasi, maka *count rate* menjadi 1/12 dari frekuensi osilator. Sedangkan pada fungsi Counter, register dinaikkan berdasarkan tanggapan adanya transisi tinggi ke rendah pada pin yang bersesuaian (dalam hal ini pin T2 atau P1.0). Tabel berikut menunjukkan mode operasi yang dapat dijalankan pada Timer 2.

Tabel 2.6 Mode Operasi Timer 2[1]

RCLK + TCLK	CP/RL2	TR2	MODE
0	0	1	16-bit auto-reload
0	1	1	16-bit capture
1	X	1	Baund Rate Generator
X	X	0	Off

Keterangan :

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

TCLK = *Transmit Clock Enable*. Jika diset menyebabkan serial port menggunakan pulsa overflow Timer 2 sebagai detak pengiriman. Jika TCLK = 0, maka pulsa *overflow* Timer 1 yang digunakan

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

2.2.7. Sistem Interrupt

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

INT0 = interrupt pada P3.2 (kaki 12)

INT1 = interrupt pada P3.3 (kaki 13)

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

Timer 1 = Timer pada P3.5 (kaki15)

Port serial = jika pengiriman/penerimaan suatu frame telah lengkap

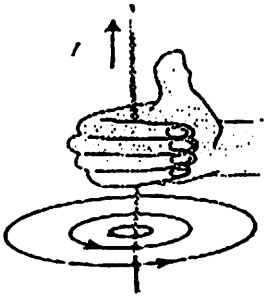
Saat terjadinya interupsi, mikrokontroler secara otomatis akan menuju *subroutine* pada alamat tersebut. Setelah interupsi servis selesai dikerjakan, mikrokontroler akan mengerjakan program semula. Dua sumber eksternal adalah INT0 dan INT1, kedua interupsi eksternal akan aktif, transisi tergantung isi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

dengan set atau *clear bit* pada SFR IP (*interrupt priority*). Register yang akan berperan dalam mengatur aktif tidaknya interupsi adalah Interrupt Enable Register.

2.2.8. Reset

Rangkaian power on reset diperlukan untuk mereset mikrokontroler secara otomatis setiap catu daya. Ketika catu daya diaktifkan, rangkaian reset menahan logika tinggi pin RST dengan jangka waktu yang ditentukan oleh besarnya pengisian muatan C.

2.3. Motor DC



Gambar 2-6. Garis-garis Medan Magnet disekitar Arus Listrik Pada Kawat Lurus [2]

Setiap arus yang mengalir melalui sebuah konduktor akan menimbulkan medan magnet. Arah medan magnet dapat ditentukan dengan kaidah tangan kanan. Ibu jari tangan menunjukkan arah aliran arus listrik sedangkan jari-jari yang lain menunjukkan arah medan magnet yang timbul, seperti yang ditunjukkan oleh gambar 2-6

Kaidah tangan kanan untuk motor menunjukkan arah arus yang mengalir didalam sebuah konduktor yang berada dalam medan magnet. Jari tengah menunjukkan arah arus yang mengalir pada konduktor, jari telunjuk menunjukkan arah medan magnet dan ibu jari menunjukkan arah gaya putar. Adapun besarnya gaya yang bekerja pada konduktor tersebut dapat dirumuskan dengan :

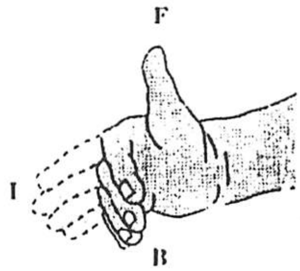
$$F = B.I.L \sin \theta \quad \text{(Newton)} \qquad (2\text{-}7)$$

Dimana : B = kerapatan fluks magnet (weber)

L = panjang konduktor (meter)

I = arus listrik (ampere)

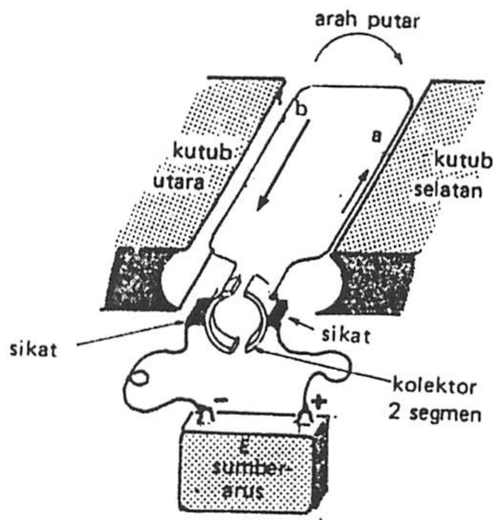
Sin θ = sudut antara antara arus dengan garis-garis medan



Gambar 2-7. Kaidah Tangan Kanan[2]

2.3.1. Cara Kerja Motor DC

Adapun cara kerja motor DC dapat dilihat pada gambar dibawah ini:

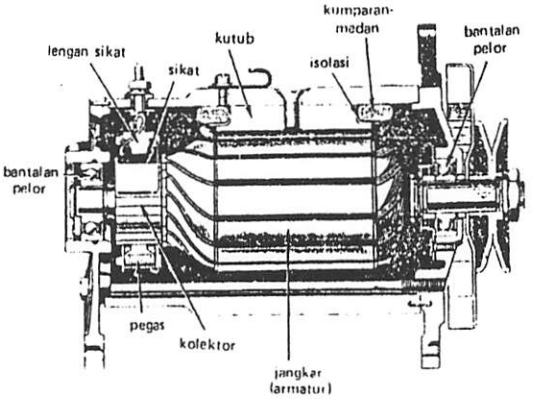


Gambar 2-8. Dasar Kontruksi Motor DC[2]

Ada satu lilit kawat a – b berada di dalam medan magnet. Lilitan ini dapat berputar dengan bebas, lilitan ini biasa disebut dengan jangkat (*armour*).

Pada jangkar dimasukkan arus yang berasal dari sumber (baterai) E. Koneksi baterai dengan jangkar melalui sikat-sikat. Sikat-sikat ini terpasang pada sebuah cincin yang terbelah dua, yang disebut kolektir. Adapun tujuan dari kontruksi ini adalah agar lilitan kawat dapat berputar apabila ada arus listrik yang melewatinya.

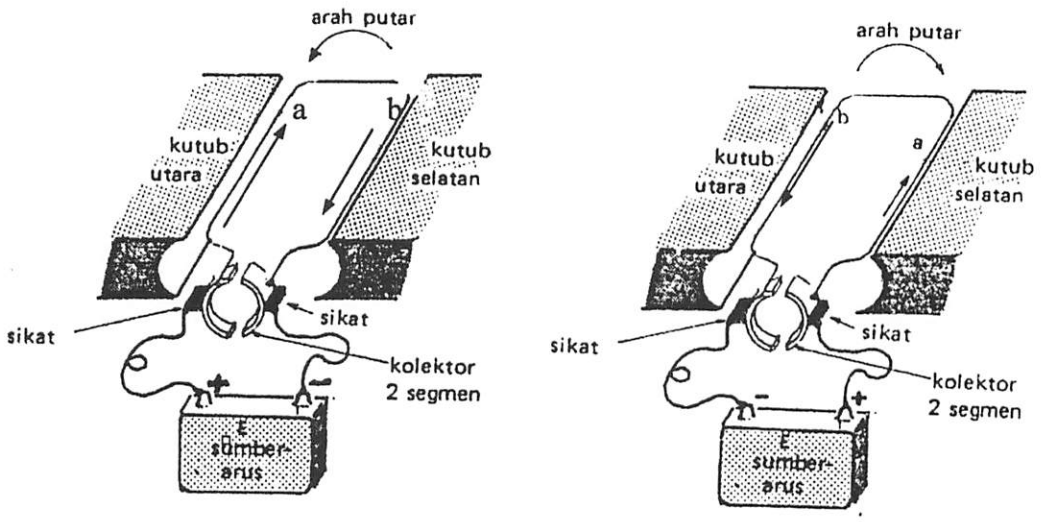
Pada kawat yang berada di kanan arus mengalir dari depan ke belakang dalam kawat yang di kiri, arus mengalir dari belakang ke depan kawat a dan b secara berganti-gantian berada di kiri dan kanan. Karena itu arah arus di a dan arah arus di b selalu membolak balik. Pembalikan arah arus itu terjadi pada saat lilitan kawat melintasi posisi vertikal.



Gambar 2-9. Kontruksi Motor DC[2]

Disini kolektor berfungsi bagaikan penyearah mekanik. *Flux* magnet yang ditimbulkan magnet permanen disebut medan magnetnya motor. Dalam gambar arah fluk magnetik adalah dari kiri ke kanan. Adapun gaya yang bekerja pada penghantar b adalah ke atas, sementara gaya yang bekerja pada penghantar a adalah ke bawah. Gaya-gaya yang bekerja sama kuatnya, jadi ada kopel yang bekerja pada kawat sehingga lilitan pun dapat berputar. Setelah berputar 90^0 arah arus berbalik, pada saat itu penghantar a dan penghantar b bertukar tempat. Akibatnya arah gerak putaran tidak berubah.

2.3.2. Pengendalian Arah Putaran Motor DC



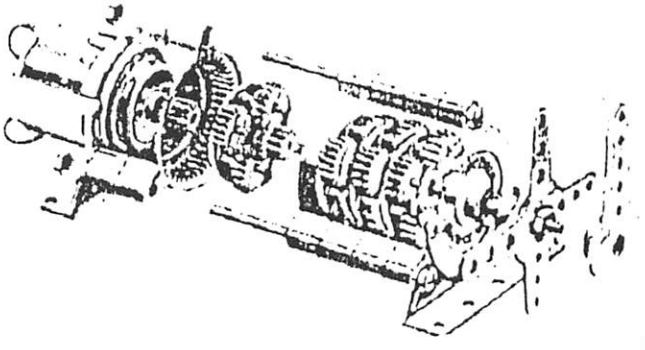
Gambar 2-10. Arah Putaran Motor DC

Sumber : Elektronika dalam Industri

Dari gambar 2-10 diatas, agar arah putaran motor DC berubah, maka polaritas tegangan pada baterai harus dibalik.

2.3.3. Motor DC Gearbox

Prinsip kerja dari motor dc gearbox sama dengan motor dc pada umumnya. Motor dc gearbox bentuk fisiknya adalah motor dc,namun ada penambahan mekanik yang berupa gear (roda gigi).

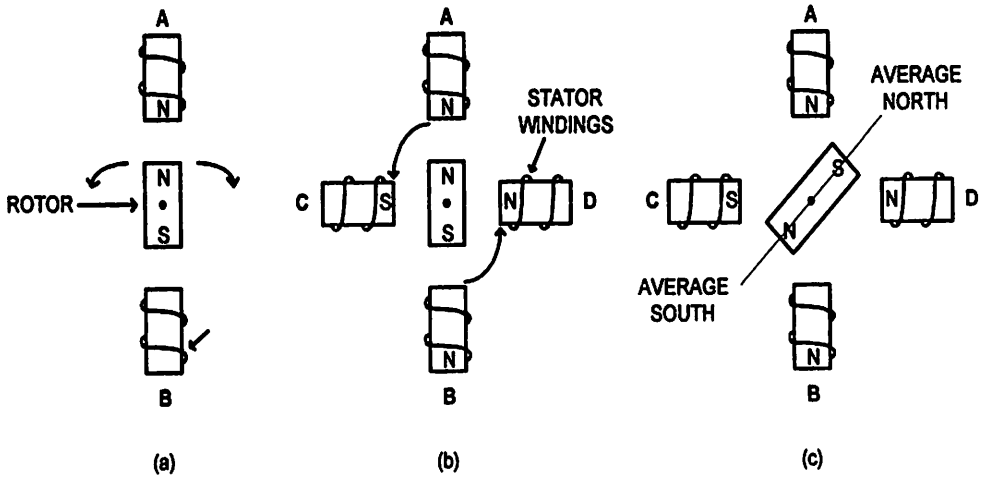


Gambar 2.11. Motor DC Gearbox

2.4. Motor Stepper

Bentuk dasar dari motor *stepper* yang paling sederhana terdiri atas sebuah rotor, yang merupakan magnet permanen, dan sebuah stator, yang dililiti kumparan sehingga dapat membentuk magnet listrik. Jika stator diberi arus listrik, sisi-sisi rotor akan membentuk kutub-kutub magnet. Jika kutub magnet stator dan rotor sama kedua magnet akan saling tolak menolak sehinnga mengakibatkan rotor berputar. Arah perputaran ini dapat dua arah, tergantung dari faktor mekanik motor *stepper* itu sendiri. Besarnya perputaran adalah 180 derajat seperti ditunjukkan Gambar 2.12 a.

Motor *stepper* dapat juga terdiri atas dua buah stator dengan sebuah rotor. Prinsip kerja motor *stepper* ini sama dengan motor *stepper* yang terdiri atas sebuah rotor dan sebuah stator. Jika arah arus listirk dan arah rotor sedemikian rupa seperti membentuk konfigurasi magnet seperti Gambar 2.12 b, rotor akan berputar berlawanan dengan arah jarum jam sebesar 90 derajat. Perputaran ini disebut *full step*.



Gambar 2.12. Dasar-dasar Motor Stepper[2]

Jika magnet permanen dan magnet listrik membentuk konfigurasi seperti Gambar2.12 c, motor akan berputar 45 derajat. Perputaran ini disebut *half step.* Jika motor *stepper* terdiri atas 4 pasang stator, besar *full step* adalah 45 derajat dan *half step* sebesar 22,5 derajat.

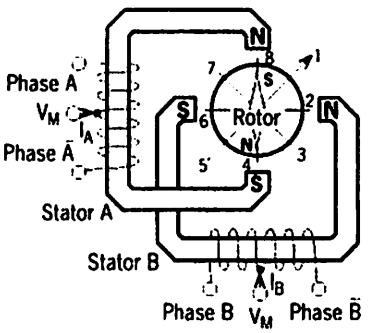
Dengan Kombinasi bit tertentu, arah perputaran dapat searah jarum jam maupun berlawanan arah jarum jam, trergantung konfigurasi bit yang diberikan. Dengan menggunakan *half step* berarti besarnya perputaran semakin kecil.

Motor *stepper* adalah perangkat elektromekanis yang bekerja dengan mengubah pulsa elektronis menjadi gerakan mekanis diskrit. Motor stepper bergerak berdasarkan urutan pulsa yang diberikan kepada motor. Karena itu, untuk menggerakkan motor stepper diperlukan pengendali motor stepper yang membangkitkan pulsa-pulsa periodik. Penggunaan motor stepper memiliki beberapa keunggulan dibandingkan dengan penggunaan motor DC biasa. Keunggulannya antara lain adalah :

- Sudut rotasi motor proporsional dengan pulsa masukan sehingga lebih mudah diatur.
- Motor dapat langsung memberikan torsi penuh pada saat mulai bergerak
- Posisi dan pergerakan dapat ditentukan secara presisi
- Memiliki respon yang sangat baik terhadap mulai, stop dan berbalik (perputaran).
- Frekuensi perputaran dapat ditentukan secara bebas dan mudah pada range yang luas.

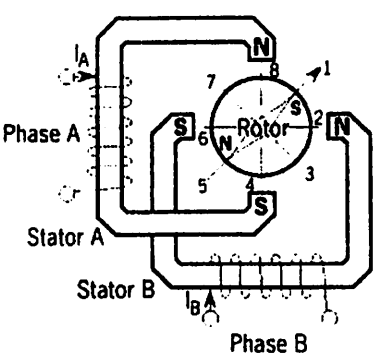
Berdasarkan metode perancangan rangkaian pengendalinya, motor *stepper* dapat dibagi menjadi jenis unipolar dan bipolar. Rangkaian pengendali motor stepper unipolar lebih mudah dirancang karena hanya memerlukan satu switch / transistor setiap lilitannya. Untuk menjalankan dan menghentikan motor ini cukup dengan

menerapkan pulsa digital yang hanya terdiri atas tegangan positif dan nol (*ground*) pada salah satu terminal lilitan (*wound*) motor sementara terminal lainnya dicatu dengan tegangan positif konstan (V_M) pada bagian tengah (*center tap*) dari lilitan (perhatikan gambar 2.13).



Gambar 2. 13. Motor stepper dengan lilitan unipolar[2]

Pada alat ini menggunakan motor stepper dengan lilitan bipolar.Untuk motor stepper dengan lilitan bipolar, diperlukan sinyal pulsa yang berubah-ubah dari positif ke negatif dan sebaliknya. Jadi pada setiap terminal lilitan (A & B) harus dihubungkan dengan sinyal yang mengayun dari positif ke negatif dan sebaliknya (perhatikan gambar 2.14). Karena itu dibutuhkan rangkaian pengendali yang agak lebih kompleks daripada rangkaian pengendali untuk motor unipolar. Motor stepper bipolar memiliki keunggulan dibandingkan dengan motor stepper unipolar dalam hal torsi yang lebih besar untuk ukuran yang sama.



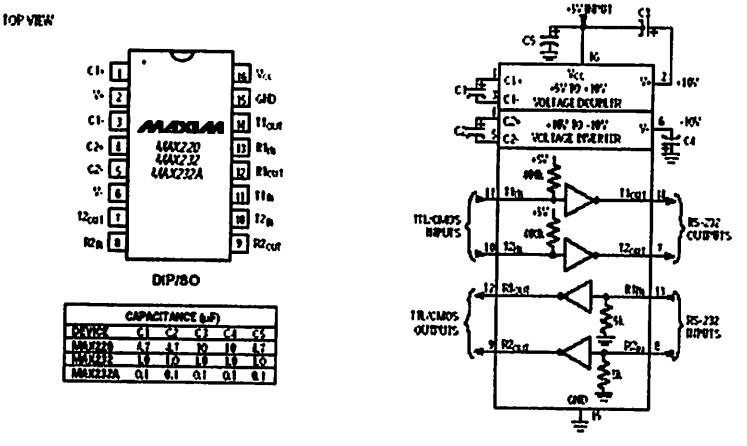
Gambar 2. 14. Motor stepper dengan lilitan bipolar[2]

2.5. RS Max 232

RS 232 merupakan salah satu jenis antar muka (interface) dalam proses transfer data antar komputer dalam bentuk serial transfer. RS 232 merupakan singkatan dari Recommbeded Standard number 232. Alat ini dibuat oleh Elektronik Industri Assosiation, untuk interface antara peralatan terminal data dan peralatan komunikasi data, dengan menggunakan data biner serial sebagai data yang ditransmisikan. IC RS MAX 232 ini mempunyai empat buah bagian konverter yaitu dua buah driver receiver dan dua buah driver transmitter.

Saluran data pada port seri PC menggunakan standard RS 232, dimana logic 0 (low) dinyatakan sebagai tegangan antara +3 volt sampai +10 volt, dan logic 1 (high) dinyatakan sebagai tegangan antara -3 volt sampai -10 volt. Level tegangan ini tidak sesuai dengan level tegangan yang dipakai pada port seri AT89S8252 yang menggunakan standard TTL (Tranastor Transistor Logic), yaitu level tegangan baku dalam rangkaian-rangkaian digital.

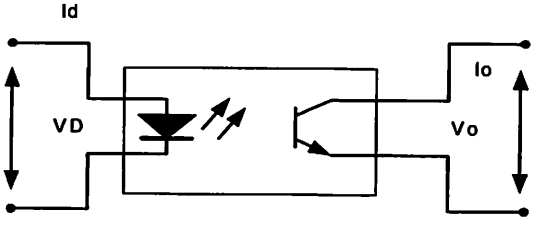
Dalam standard TTL, logic 0 (low) dinyatakan sbagai tegangan antara 0 volt sampai 0,8 volt, dan logic 1 (high) dinyatakan sebagai tegangan antara 3,5 volt sampai 5 volt. Untuk dua MCU yang dihubungkan secara serial pada jarak tertentu maka dibutuhkan IC MAX 232 karena level tegangan TTL terlalu kecil untuk ditrasfer.



Gambar 2.15. IC RS Max 232[3]

2.6. Optocoupler

Optocoupler disebut juga optoisolator atau isolator yang tergandeng optic, menggabungkan LED dan fototransistor dalam satu kemasan. Gambar 2.16. menunjukkan salah satu contoh dari optocoupler. Komponen ini memiliki LED pada sisi masukan dan fototransistor pada sisi keluaran. Keuntungan utama optocoupler adalah pemisah secara listrik antara rangkaian masuk dengan rangkaian keluarnya. Dengan optocoupler, hubungan yang ada antara masukan dan keluaran hanya seberkas cahaya. Karena hal ini dapat memperoleh resistansi penyekatan diantara dua rangkaian tersebut. Optocoupler yang dipakai adalah yang terdiri dari satu LED dan satu transistor foto seperti terlihat dalam gambar berikut ini:



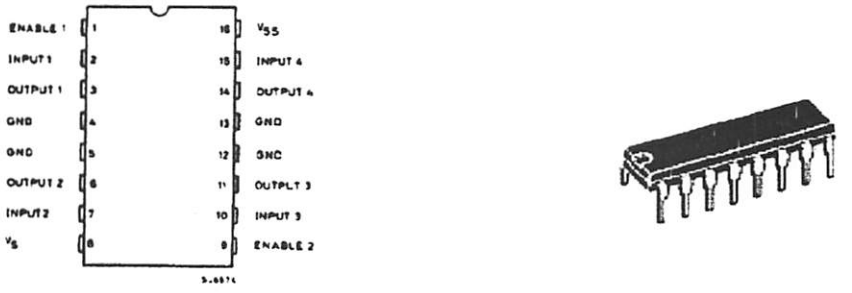
Gambar 2.16. Simbol Optocoupler[4]

2.7. L 293D

L 293D merupakan driver push-pull 4 saluran dengan diode. Komponen ini merupakan driver empat saluran yang terintegrasi pada tegangan tinggi dan arus yang besar.Memenuhi standart tingkat logika DTL atau TTL dan sebagai driver komponen berinduksi seperti relay, solenoid, motor dc maupun motor stepper.

Untuk menjembatani dua saluran terdapat input enable.Yang berfungsi sebagai masukan pada input dan berupa logika.Didalammya terdapat diode-diode internal pada operasi tenaga rendah.

Komponen ini sangat cocok untuk digunakan pada oprasi switching pada frekuensi hingga 5 KHz.



Gambar 2.17. IC L293D[5]

2.8. L 298N

L 298N merupakan driver berjembatan ganda yang memiliki tegangan serta arus yang tinggi. Dua input enable berfungsi mengkondisikan sinyal input pada komponen.Emitor pada transistor terendah di jembatan tergabung bersama- sama dan dikoneksikan pada hambatan luar.

L 298N mampu beroperasi pada tegangan supply hingga 46 V.Total arus DC hingga 4 A.Memiliki tegangan saturasi yang rendah serta tahan pada kondisi bertemperatur sangat tinggi.

Gambar 2.18. IC L 298N(Multiwatt 15)[5]

2.9. Transistor

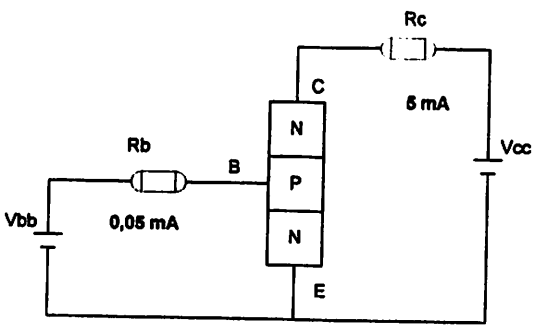
Transistor adalah komponen semikonduktor, Fungsi transistor dapat sebagai penguat (*amplifier*) dari sinyal listrik atau sebagai saklar.

Banyak sistem elektronis yang tergantung pada kemampuan transistor untuk bertindak sebagai saklar. Transistor yang digunakan sebagai saklar mempunyai

keuntungan, yaitu dapat beroperasi ON dan OFF pada kecepatan yang sangat tinggi, memerlukan tegangan dan arus penggerak yang sangat rendah untuk memicu aksi penghubungan.

Gambar 2.19 menggambarkan aksi penghubungan transistor sambungan bipolar NPN (BJT = *Bipolar Junction Transistor*). Emitor yang diberi banyak bahan campuran yang bertindak sebagai sumber utama dari arus elektron. Basis dengan sedikit bahan campuran bertindak untuk mengontrol aliran arus. Kolektor diberi dengan bahan yang cukup dan menerima sebagian besar elektron dari emitor. Arus pada ujung basis disebut arus basis dan arus pada ujung kolektor disebut arus kolektor. Jumlah arus basis (I_b) menentukan jumlah arus kolektor (I_c). Dengan tidak adanya arus basis tidak ada arus kolektor (*normally OFF*). Sedikit kenaikan pada I_b mengakibatkan kenaikan yang besar pada I_c , jadi arus basis bertindak sebagai pengontrol arus kolektor.

Gain arus adalah perbandingan arus kolektor terhadap arus basis, transistor sambungan bipolar mempunyai dua variabel : NPN dan PNP, aksi dari masing-masing adalah sama tetapi polaritasnya terbalik.



Gambar 2.19. Aksi penggabungan transistor[6]

2.9.1 Penentuan Titik Kerja

Pada bias basis mempunyai sifat sebagai berikut :

- Peka terhadap perubahan-perubahan β_{dc}.

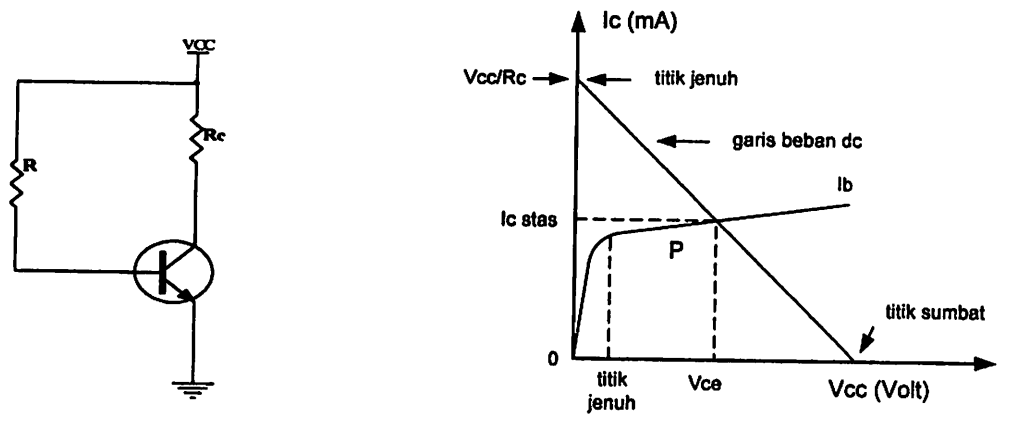
- Untuk switching
- $I_B = \frac{V_{cc} - V_{BE}}{R_B}$
- V_{BE} = 0,6 V (silicon) ; 0,2 V (germanium)
- $I_C = \frac{V_{CC} - V_{CE}}{R_C}$
- Dalam kondisi *cutt of*:

  $V_{CE} = V_{CC}$
- Dalam kondisi jenuh (saturasi) :

  $V_{CE} \approx 0$

  $I_C = \frac{V_{CC}}{R_C}$

  $I_B = \frac{I_C}{\beta_{dc}}$
- Menentukan bias berarti menentukan titik kerja P dalam karakteristik I_C-V_{CE}



(a) (b)

Gambar 2.20. (a) Transistor sebagai saklar[6]
(b) Garis beban dc[6]

2.10. Limit Switch

Limit swicth merupakan sebuah saklar yang bekerja karena ada suatu sentuhan. Limit swicth mempunyai beberapa bagian antara pengungkit dan roda penjulang yang merupakan bagian mekanik yang berfungsi sebagai penentu posisi awal (koordinat 0.0) sumbu X.

Ada beberapa tipe limit switch yaitu limit swicth yang merupakan kontak NC (Normally Closed) dan NO (Normally Open). Limit yang merupakan kontak NO berfungsi sebagai penghubung sedangkan yang kontak NC berfungsi sebagai pemutus.

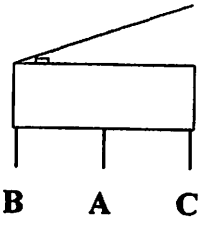
Pushbutton make NO (normally Open) | **Pushbutton break NC (normally Closed)**

Gambar 2.21. Tipe Limit Switch[7]

Adapun simbol dari limit swicth adalah sbb:

Gambar 2.22. Simbol Limit Switch[7]

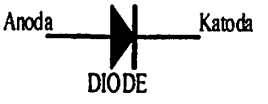
Sedangkan jenis limit switch yang digunakan dalam peralatan ini adalah yang terllihat seperti di bawah ini :



Gambar 2.23. Gambar Komponen dari limit switch[7]

2.11. Dioda

Dioda adalah komponen elektronika yang berfungsi sebagai penyearah arus listrik.Bila bahan jenis P dan jenis N dipertemukan maka akan diperoleh apa yang disebut *dioda*.Dinamakan dioda karena mempunyai 2 elektroda, dimana bahan jenis P- nya disebut *anoda*, sedang bahan jenis N –nya disebut *katoda*.



Gambar 2.24. Simbol Dioda[7]

- Arus dapat mengalir dari arah anoda ke katoda, tetapi tidak mengalir sebaliknya.
- Dioda dikatakan bocor bila arus listrik dapat mengalir dari katoda ke anoda.
- Dioda dikatakan putus bila arus listrik tidak dapat mengalir dari anoda ke katoda.

BAB III

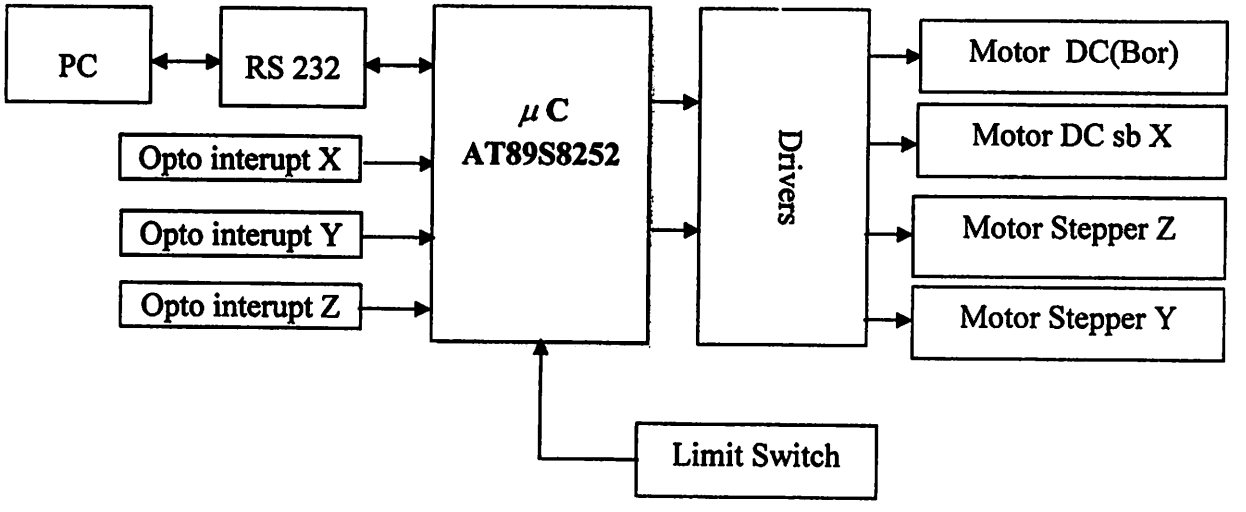
PERANCANGAN DAN PEMBUATAN BOR OTOMATIS BERBASIS MIKROKONTROLLER AT89S8252

Perancangan dan pembuatan dibedakan atas dua aspek yaitu aspek perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.1. Perangkat Keras (*Hardware*)

3.1.1. Miniatur Sistem Dan Blok Diagram

Miniatur siatem dan blok diagram dari perancangan perangkat keras secara keseluruhan adalah sebagai berikut :



Gambar 3.1. Blok Diagram Alat
Sumber : Perancangan dan Pembuatan

Adapun penjelasan dari tiap-tiap blok adalah sebagai berikut :

- Mikrokontroller AT89S8252

  Melakukan komunikasi dengan PC melalui RS MAX 232, pengatur driver motor stepper dan motor DC, menerima sinyal dari limit switch.

- Driver

  Sebagai penggerak mekanik, dalam hal ini adalah motor DC dan motor stepper.

- RS MAX 232

  Sebagai interface antara data masukan dari software PC ke mikrokontroller.

- Limitswitch

  Sebagai penentu posisi awal (koordinat 0,0).

- Motor Stepper

  Sebagai penggerak (dalam hal ini adalah sumbu Y dan sumbu Z)

- Motor DC

  Sebagai pemutar mata bor dan penggerak sumbu X.

- Opto interupt

  Sebagai pendeteksi batas pada sumbu Y dan Z dan penentu jarak lubang pada sumbu X.

3.1.2.Prinsip Kerja Alat

Operator memilih titik atau koordinat yang akan dibor melalui tampilan monitor di PC sebagai inputan, selanjutnya data tersebut dikirim ke mikrokontroller dalam hal ini AT89S8252 melalui RS-232.Setelah menerima dan membaca data dari PC, mikrokontroller akan menyimpan data pada memory sebagai data sementara hingga tidak ada titik lagi yang akan dibor.Setelah tidak ada penandaan titik lagi dan operator

melakukan eksekusi pengeboran, mikrokontroller akan mengolah dan memproses data tersebut (melakukan pengeboran).Setelah proses pengeboran selesai dilakukan, mikrokontroller akan mengirimkan data kembali untuk ditampilkan pada PC.

3.1.3.Mikrokontroller AT89S8252

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

1. Pin 3 – 8 (P1.2 – P1.7)

   Berfungsi untuk menggerakkan motor DC sebagai sumbu X dan pemutar bor.

2. Pin 21- 22 (P2.0- P2.1)

   Merupakan port yang digunakan sebagai inputan tegangan yang berasal dari opto interupt sumbu X dan Y.

3. Pin 12 (P3.2)

   Merupakan port yang digunakan sebagai inputan tegangan yang berasal dari opto interupt sumbu Z.

4. Pin 14 (P3.4)

   Merupakan port yang digunakan sebagai inputan yang berasal dari limit switch (Penentu kondisi awal).

5. Pin 9 berfungsi sebagai Reset.

6. Pin 23– 28 (P2.2 – P2.7)

   Merupakan Port 2 dari mikrokontroller dihubungkan dengan driver sebagai untuk mengendalikan motor stepper sumbu Y.

7. Pin 34- 39 (P0.0-P0.5)

   Merupakan Port 0 dari mikrokontroller dihubungkan dengan driver untuk mengendalikan motor stepper sumbu Z.

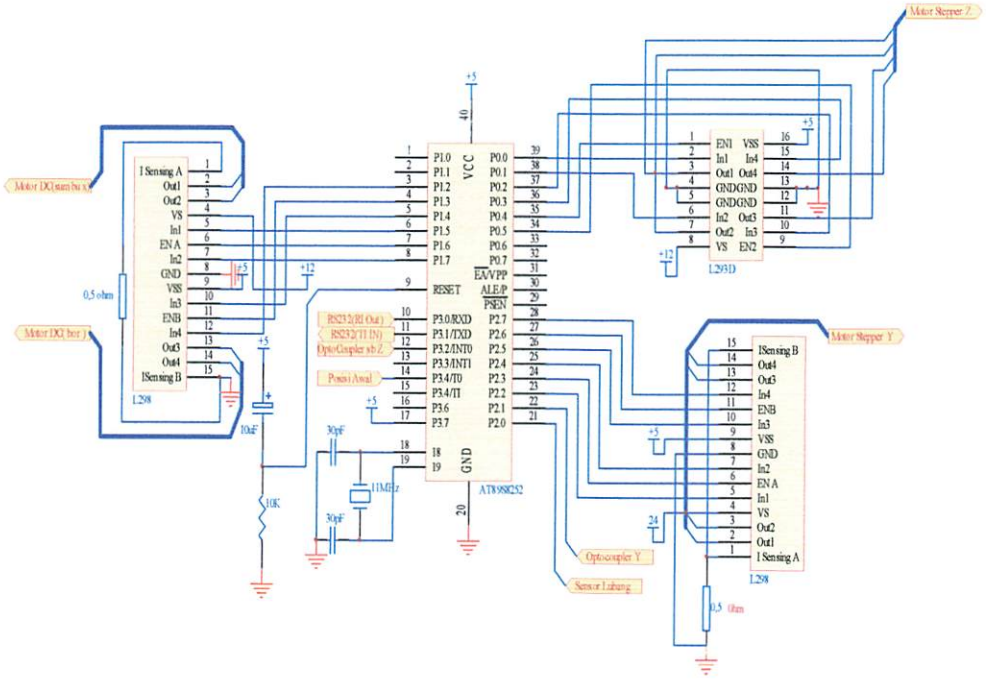
8. Pin 10 dan 11 (P3.0 – P3.1)

   Merupakan Port 3 dari mikrokontroller yang digunakan sebagai kontrol yang terdiri dari:

   - Pin 10 (P3.0), RXD berfungsi sebagai receive data untuk serial port yang dihubungkan dengan data serial komputer.
   - Pin 11 (P3.1), TXD berfungsi sebagai transmit data untuk serial port yang berasal dari data serial komputer.

9. Pin 18 (XTAL 2) sebagai pembangkit ossilator (clock) XTAL 2.
10. Pin 19 (XTAL 1) sebagai pembangkit ossilator (clock) XTAL 1.
11. Pin 31 (EA/VPP) dan Pin 40 berfungsi sebagai VCC +5 Volt.
12. Pin 20 (GND) berfungsi sebagai ground.

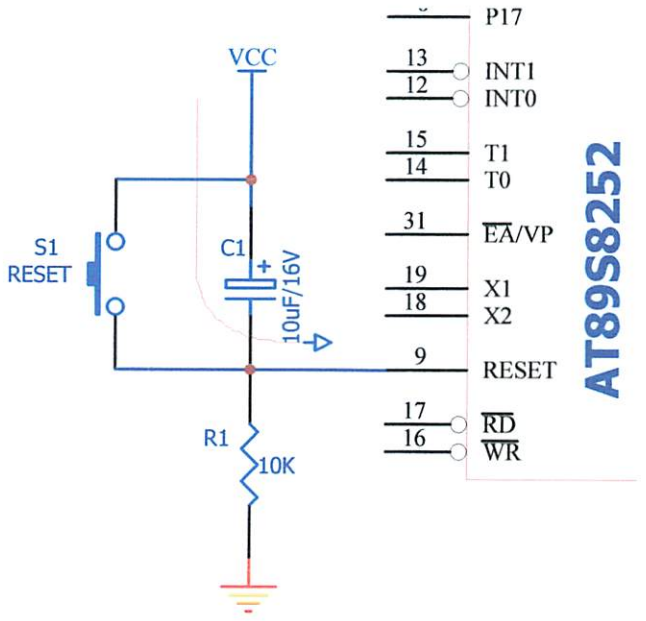
Mikrokontroller pada alat ini tidak dapat bekerja sendiri sehingga masih membutuhkan komponen-komponen pendukung lain, komponen-komponen tersebut saling berhubungan secara hardware dan juga software.Rangkaian mikrokontroller dapat dilihat pada gambar dibawah ini:



Gambar 3.2. Rangkaian Mikrokontroller AT89S8252
Sumber : Perancangan dan Pembuatan

3.1.4. Perancangan Rangkaian Reset

Untuk m*ereset* mikrokontroler AT89S8252, maka pin RST diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal *reset* kapasitor dihubungkan dengan Vcc dan sebuah resistor yang dihubungkan ke *ground*. Rangkaian *reset* ditunjukkan dalam gambar 3-3 sebagai berikut :



Gambar 3-3. Perancangan Rangkaian *Reset*
Sumber : Perancangan dan Pembuatan

Karena kristal yang digunakan mempunyai frekuensi sebesar 11,0592 MHz, maka satu periode membutuhkan waktu sebesar :

$$T = \frac{1}{f_{XTAL}} = \frac{1}{11{,}0592\,MHz}\,S = 9{,}042\times10^{-8}\,s$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk m*ereset* mikrokontroller adalah :

reset(min) = T × periode yang dibutuhkan

= 9,042 × 10^{-8} × 24 = 2,17 μs

Jadi mikrokontroller membutuhkan waktu minimal 2,17 μs untuk m*ereset*. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dengan menentukan nilai R = 8,2 kΩ dan C = 10 μF, maka :

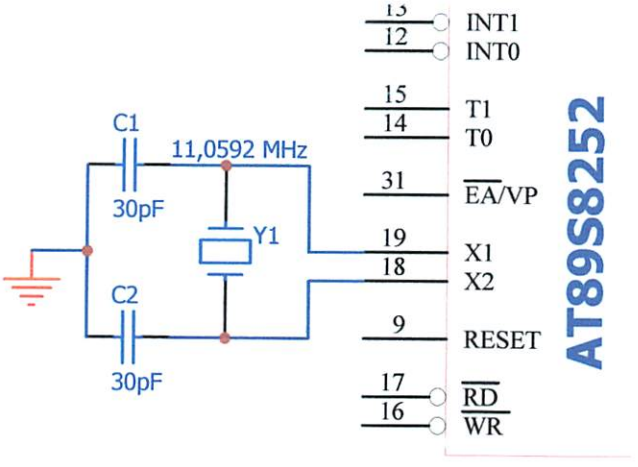
t = 0,357 R.C = 0,357 × 8200Ω × 10.10^{-6} = 29,274 ms

Jadi dengan nilai komponen R = 8,2 kΩ dan C = 10μF dapat memenuhi syarat minimal untuk waktu yang dibutuhkan oleh mikrokontroller.

3.1.5. Perancangan Rangkaian Clock

Kecepatan proses yang dilakukan oleh mikrokontroler ditentukan oleh sumber *clock* yang mengendalikan mikrokontroler tersebut. Sistem yang dirancang ini menggunakan osilator internal yang telah tersedia dalam *chip* AT89S8252. Untuk menentukan frekuensi osilatornya cukup dengan menghubungkan kristal dalam pin 19 (X_1) dan pin 18 (X_2) serta dua buah kapasitor ke *ground.*

Besarnya kapasitansinya disesuaikan dengan spesifikasi dalam lembar data AT89S8252 yaitu 30 pF. Kristal yang digunakan adalah 11,0592 MHz. Gambar 3.4 memperlihatkan rangkaian *clock* yang dirancang.



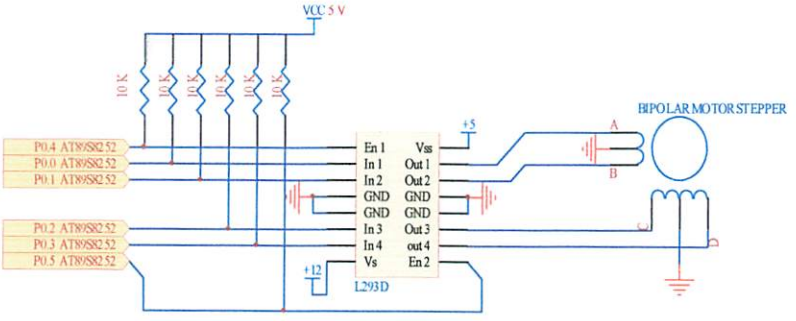
Gambar 3.4. Perancangan Rangkaian *Clock*
Sumber : Perancangan dan Pembuatan

3.1.6.Perancangan Driver

Rangkaian ini berfungsi untuk menyambung dan memutuskan rangkaian yang ada diluar mikrokontroller.

3.1.6.1. Perancangan Driver Motor Stepper Sumbu Z

Rangkaian driver ini digambarkan sebagai berikut :



Gambar 3.5. Driver Motor Stepper
Sumber : Perancangan dan Pembuatan

Diketahui : Vs = 12 V`

Vss = 5 V

Maka, berdasar data sheet.....

Vi = 7 V

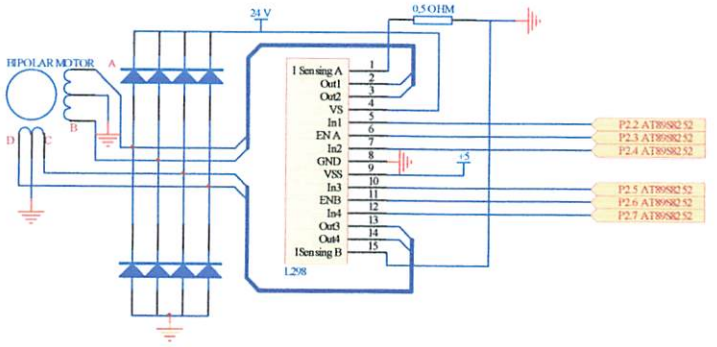
V en = 7 V

Io = 1,2 A

Ptot = 4 Watt

3.1.6.2. Driver Motor Stepper Sumbu Y

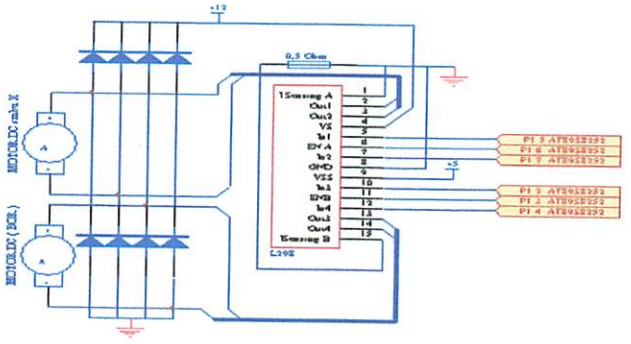
Rangkaian driver ini digambarkan sebagai berikut :



Gambar 3.6. Driver Motor Stepper Y
Sumber : Perancangan dan Pembuatan

3.1.6.3. Perancangan Driver Motor DC

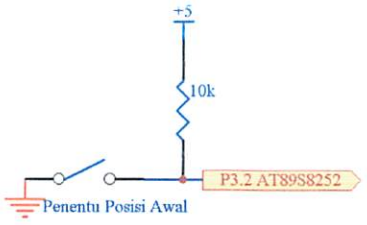
Dalam perancangan ini, driver L298 N yang digunakan untuk menggerakkan motor $12V_{DC}$ dengan memberi inputan logika " 1 "(high) atau" 0 "(Low).Agar timbul beda potensial, input harus berlogika " 1 " dan " 0 ",sedangkan Enable harus selalu berlogika " 1 " (High).



Gambar 3.7. Driver Motor DC
Sumber : Perancangan dan Pembuatan

3.1.7. Perancangan Penentu Posisi Awal

Rangkaian penentu posisi awal terdiri dari limit switch yang dihubungkan pada mikrokontroler seperti pada gambar dibawah ini :



Gambar 3.8. Rangkaian Penentu Posisi Awal
Sumber : Perancangan dan Pembuatan

Dengan tegangan masukan 5 volt dan tahanan respack 10 KΩ maka arus yang mengalir dapat dihitung sebagai berikut ;

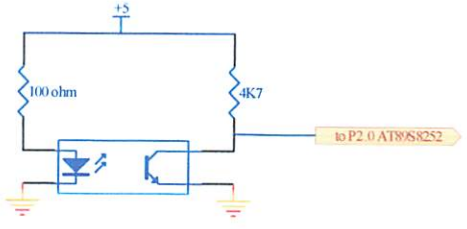
$$IR = \frac{V}{R} \text{ (Ampere)}$$

$$IR = \frac{5}{10\ K}$$

$$IR = 0{,}5\ mA$$

3.1.8. Perancangan Penentu Jarak Lubang Pada Sumbu X

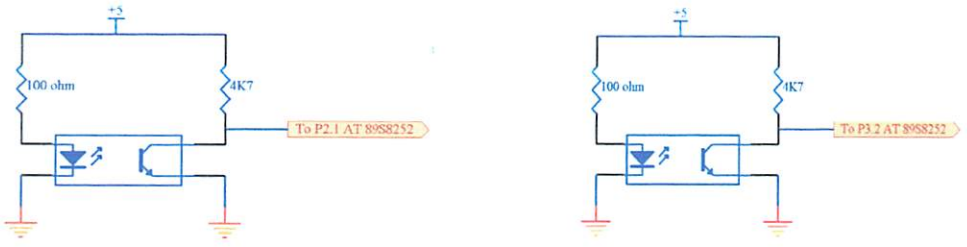
Rangkaian penentu jarak lubang pada sumbu x terdiri dari opto interupt yang dihubungkan pada mikrokontroler seperti pada gambar dibawah ini :



Gambar 3.9. Rangkaian Penentu Jarak Lubang sumbu X
Sumber : Perancangan dan Pembuatan

3.1.9. Perancangan Rangkaian Pendeteksi Pada Sumbu Y dan Z

Rangkaian pendeteksi pada sumbu x dan y terdiri dari opto interupt yang dihubungkan pada mikrokontroller seperti pada gambar dibawah ini :



a. b.

Gambar 3.10. a.) Rangkaian Pendeteksi Pada Sumbu Y
b.) Rangkaian Pendeteksi Pada Sumbu Z
Sumber : Perancangan dan Pembuatan

Mengacu pada gambar rangkaian diatas agar led dapat memancarkan sinar maka diperlukan arus (I_L) sebesar 20 mA dan tegangan (V_L) sebesar 1,5 volt sehingga dapat dihitung nilai R_L sebagai berikut :

$$R_L = \frac{Vcc - V_L}{I_L} \rightarrow R = \frac{5 - 1{,}5}{20.10^{-3}} = 175\ \Omega$$

Dalam hal ini resistor yang digunakan adalah 175 Ω. Karena resistor 175 Ω yang tidak ada dipasaran, maka digunakan resistor 100 Ω. Sedangkan untuk resistor pada phototransistor adalah sebagai berikut :

Diketahui arus yang diperlukan pada phototransistor (I_P) sebesar 0,5 mA, maka

$$R_P = \frac{Vcc}{I_P} \rightarrow R = \frac{5}{0{,}5.10^{-3}} = 10\ \text{k}\Omega$$

Namun yang digunakan adalah resistor 4K7 Ω.

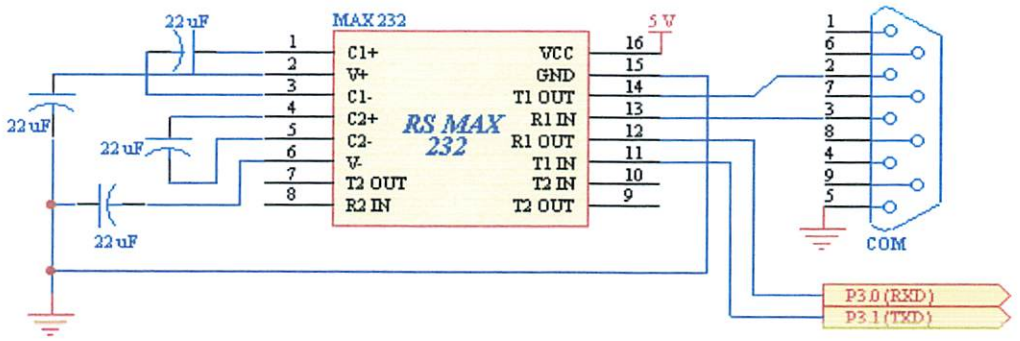
Pada kondisi dari sumber cahaya terhalang maka phototransistor akan cut off, sehingga tidak ada arus kolektor yang mengalir yang mengakibatkan tegangan output $V_0 = V_{CC}$, sebaliknya jika ada cahaya yang mengenai permukaan phototransistor maka

phototransistor akan aktif sehingga ada arus yang mengalir dari tegangan $V_0 = V_{CE\ (Sat)} = 0{,}2$ Volt.

3.2. Perancangan dan Pembuatan Perangkat Lunak (Software)

Untuk mendukung agar perangkat keras berfungsi sesuai dengan perencanaan, maka diperlukan perangkat lunak sebagai penunjangnya. Untuk mengatur dan mengendalikan keseluruhan sistem perangkat keras yang telah dibuat, harus dibantu dengan perangkat lunak.

Sistem aplikasi Mikrokontroller AT89S8252 yang terintegrasi dengan PC ini dapat mengatur dan mengendalikan keseluruhan sistem apabila ada urutan instruksi yang mendefinisikan secara jelas urutan tugas yang harus dikerjakan.



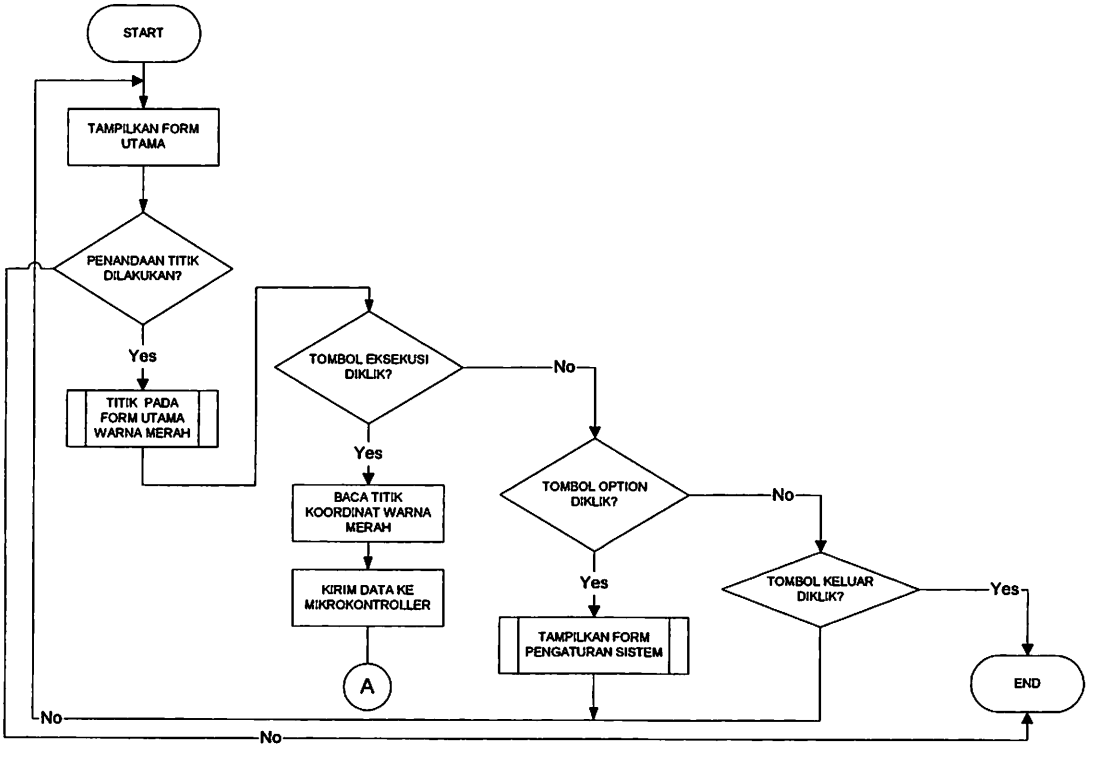
Gambar 3.11. Rangkaian RS 232
Sumber : Perancangan dan Pembuatan

Urutan instruksi ini sangat penting untuk didefinisikan, karena mikrokontroller bekerja secara pasti berdasarkan urutan instruksi ini, susunan logika perancangan yang salah tidak dapat diketahui oleh mikrokontroller. Selama instruksi yang diterima sesuai dengan aturannya, mikrokontroller tetap mengerjakan instruksi tersebut.

Kesalahan seperti ini baru diketahui ketika kerja sistem aplikasi tidak sesuai dengan spesifikasi awal. Sebuah mikrokontroller tidak akan bekerja bila tidak diberikan program kepadanya.

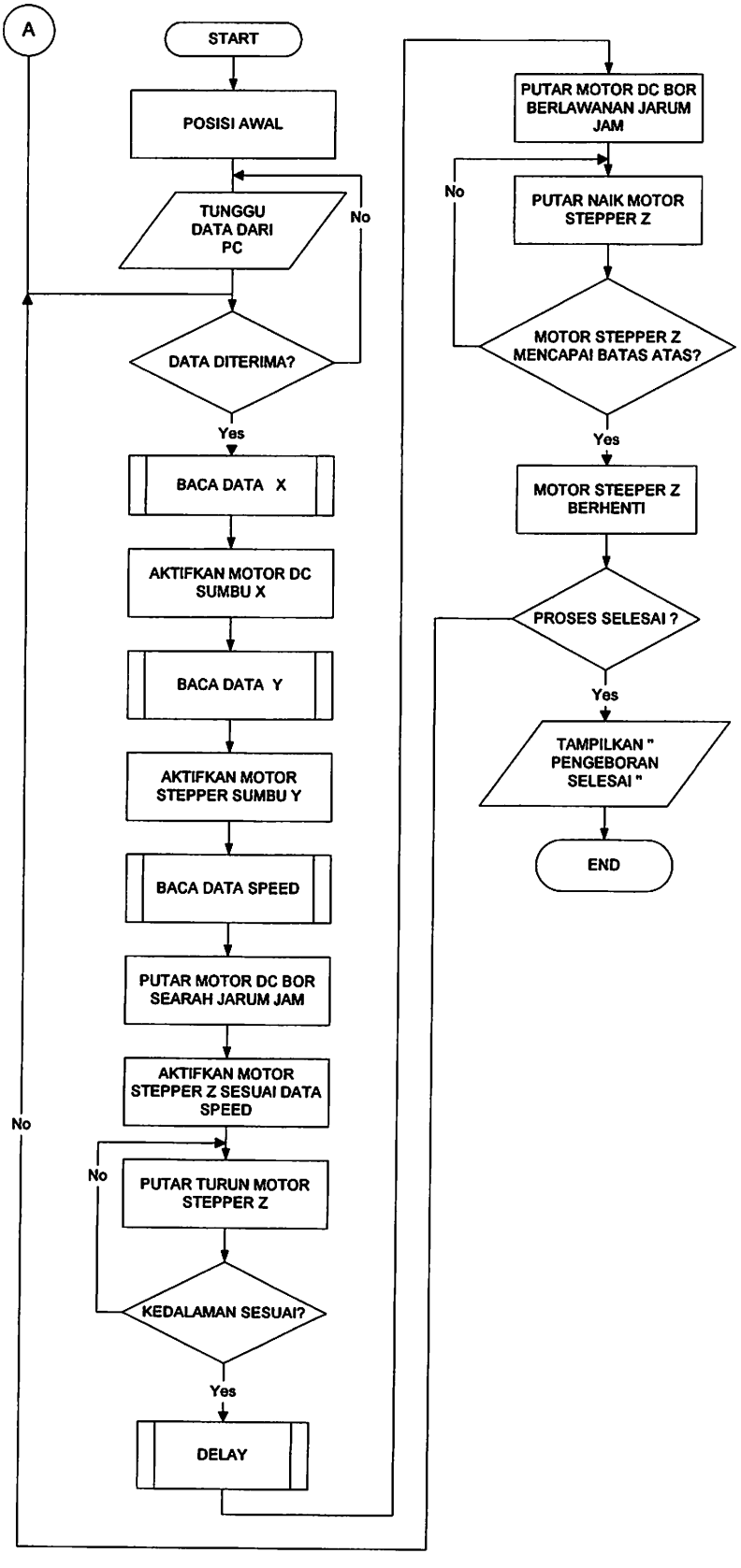
3.2.1 Diagram Alir

3.2.1.1. Diagram Alir Proses Pada PC



Gambar 3.12. Diagram Alir Proses Pada PC

3.2.1.2. Diagram Alir Penerimaan Data Dari PC



Gambar 3.13. Diagram Alir Penerimaan Data Dari PC

BAB IV

PENGUJIAN ALAT

Pengujian alat meliputi pengujian perangkat keras disertai perangkat lunak sistem. Pengujian dilakukan perbagian untuk mempermudah dalam menganalisis hasil perancangan dan pengujian yang dilakukan..

4.1. Pengujian Opto interupt

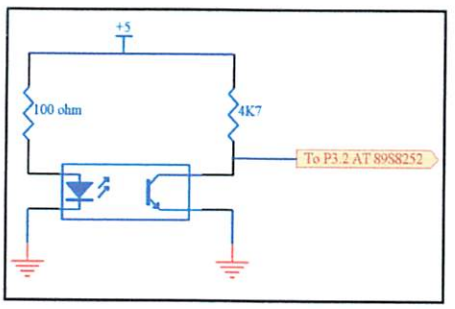
Tujuan dari pengujian ini adalah untuk mengetahui respon rangkaian apabila melewati halangan atau tidak pada saat sistem bekerja.

4.1.1. Alat dan Bahan

1. Catu Daya
2. Multimeter
3. Plat sebagai penghalang

4.1.2.Langkah-langkah Pengujian

1. Merangkai rangkaian
2. Mengukur output dari opto interupt dengan memberi halangan sebuah plat.



Gambar 4.1. Rangkaian Pengujian Opto interupt

Tabel 4.1. Hasil Perbandingan Antara Pengujian dan Perhitungan Rangkaian Opto interupt

Kondisi	Keluaran Logika	Tegangan (Volt)		Error (%)
		Perhitungan	Pengujian	
Tidak Terhalang	1	0	0	0
Terhalang	0	5	4,89	2,20
		5	4,85	3,00
		5	4,89	2,20
		5	4,88	2,40
		5	4,90	2,00

Kondisi tegangan pada opto interupt ketika terhalang :

$$Error\ V_{opto\ interupt} = \frac{\Sigma error}{BanyaknyaPengujian}$$

$$Error\ V_{opto\ interupt} = \frac{11{,}8}{5}\ \%$$

$$Error\ V_{opto\ interupt} = 2{,}36\ \%$$

4.2. Pengujian Rangkaian (Limit Switch)

Tujuan dari pengujian ini adalah untuk mengetahui apakah rangkaian kondisi awal bekerja dengan dengan baik Rangkaian akan aktif apabila saat posisi awal saklar switch tertekan (tertutup), maka limit switch mempunyai nilai " 1 " atau High. Dan sebaliknya, akan bernilai " 0 " atau Low saat posisi running proses pengeboran.

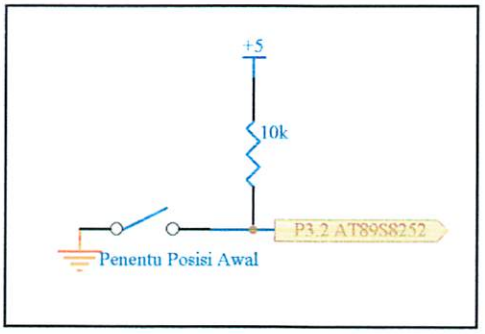
4.2.1. Alat dan Bahan

1. Catu Daya

2. Rangkaian Limit Switch

4.2.2. Langkah-langkah Pengujian

1. Merangkai rangkaian
2. Mengukur nilai tegangan keluaran rangkain saat kondisi switch tertekan atau saat posisi awal.



Gambar 4.2. Rangkaian Pengujian Limit Switch

Tabel 4.2. Hasil Perbandingan Antara Pengujian dan Perhitungan Rangkaian Limit Switch

Kondisi	Keluaran Logika	Tegangan (Volt)		Error (%)
		Perhitungan	Pengujian	
Tidak Tertekan	0	0	0	0
Tertekan	1	5	4,92	1,60
		5	4,90	2,00
		5	4,89	2,20
		5	4,94	1,20
		5	4,90	2,00

Kondisi tegangan pada limit switch saat tertekan :

$$Error\ V_{Limitswitch} = \frac{\Sigma error}{BanyaknyaPercobaan}$$

$$Error\ V_{Limitswitch} = \frac{9}{5}\ \%$$

$$Error\ V_{Limitswitch} = 1{,}8\ \%$$

4.3. Pengujian Rangkaian Driver Motor Stepper

Penggerak motor stepper dalam rangkaian ini adalah driver L293 D untuk motor stepper sumbu Z dan L 298 N untuk motor stepper Y.

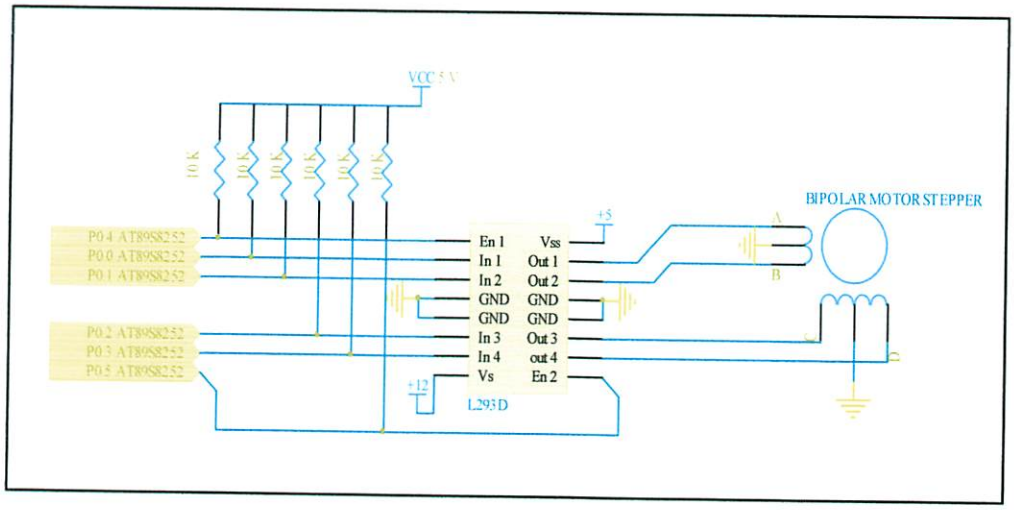
Tujuan pengujian ini adalah untuk mengetahui arah putar motor saat diberi inputan.

4.3.1. Alat dan Bahan

1. Catu Daya
2. Rangkaian Driver Motor Stepper

4.3.2. Langkah- langkah Pengujian

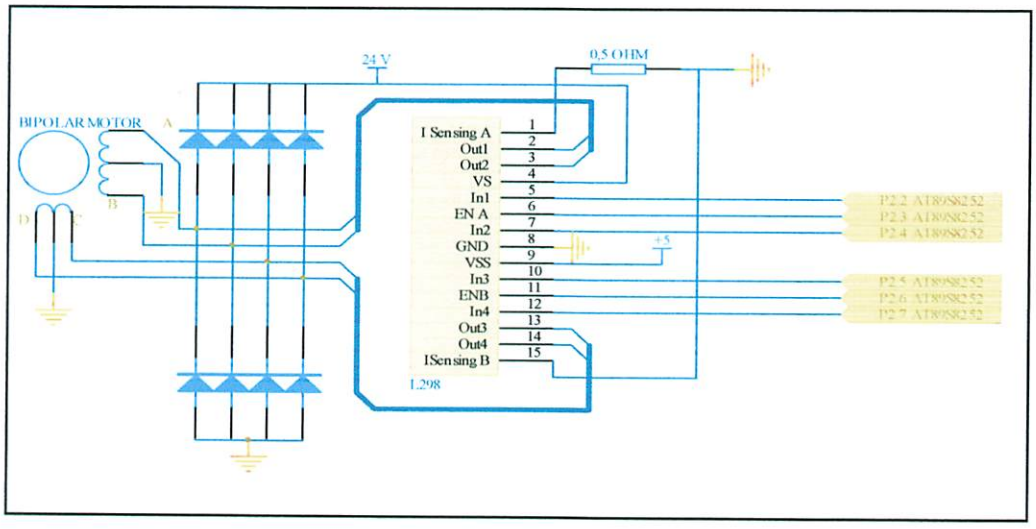
1. Merangkai Rangkaian
2. Mengukur tegangan keluaran dan melihat arah putar motor.



Gambar 4.3. Rangkaian Driver Motor Stepper Z

Tabel 4.3. Tabel Kebenaran Arah Putar Motor Stepper Z

INPUT DATA (LOGIKA)				OUTPUT (ARAH PUTAR MOTOR)
A	B	C	D	
1	1	0	0	KEKANAN
1	0	0	1	
0	0	1	1	
0	1	1	0	
0	1	1	0	KEKIRI
0	0	1	1	
1	0	0	1	
1	1	0	0	



Gambar 4.4 Rangkaian Driver Motor Stepper Y

Tabel 4.4. Tabel Kebenaran Arah Putar Motor Stepper Y

INPUT DATA (LOGIKA)				OUTPUT (ARAH PUTAR MOTOR)
A	B	C	D	
1	1	0	0	KEKANAN
1	0	0	1	
0	0	1	1	
0	1	1	0	
0	1	1	0	KEKIRI
0	0	1	1	
1	0	0	1	
1	1	0	0	

4.4. Pengujian Rangkaian Driver Motor DC

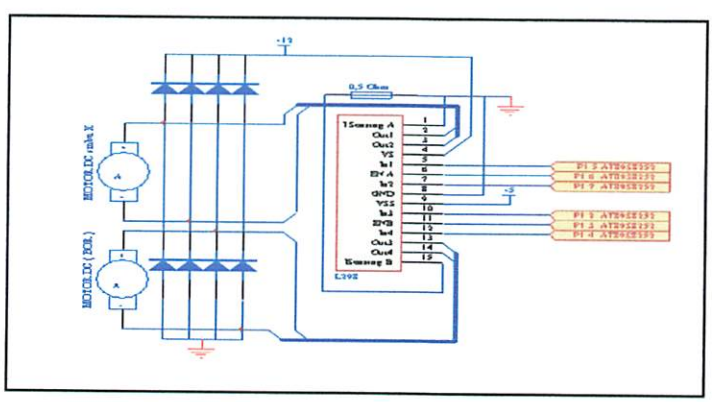
Rangkaian ini menggunakan driver L 298N, dimana berfungsi untuk memutar motor DC 12 Volt dan mengetahui nilai tegangan keluaran.

4.4.1. Alat dan Bahan

1. Catu Daya
2. Rangkaian Driver Motor DC 12 Volt

4.4.2. Langkah percobaan

1. Merangkai Rangkaian
2. Mengukur tegangan keluaran dan melihat arah putar motor.



Gambar 4.7. Rangkaian Driver Motor DC

Tabel 4.5. Tabel Kebenaran Arah Putar Motor DC

Input		Arah Putar	Vout
Ven = H	A = H, B = L	Kekanan	12
	A = L, B = H	kekiri	12
	A = B	Berhenti	0
Ven = L	A = X , B = X	Berhenti	0

H = High L = Low X = Don't care

Tabel 4.6. Pengujian Hasil Pengeboran

Titik Pengeboran	Perhitungan (cm)	Pengujian (cm)	Error (%)
Ke – 1	1	1	0
Ke – 2	1	1,2	20,00
Ke – 3	1	1,1	10,00
Ke – 4	1	1,3	30,00
Ke – 5	1	1,2	20.00
Ke – 6	1	1,1	10,00
Ke – 7	1	1	0
Ke – 8	1	1,1	10,00
Ke – 9	1	1,1	10,00
Ke – 10	1	1,2	20,00

Keterangan :

- Jarak antar titik 1 cm.

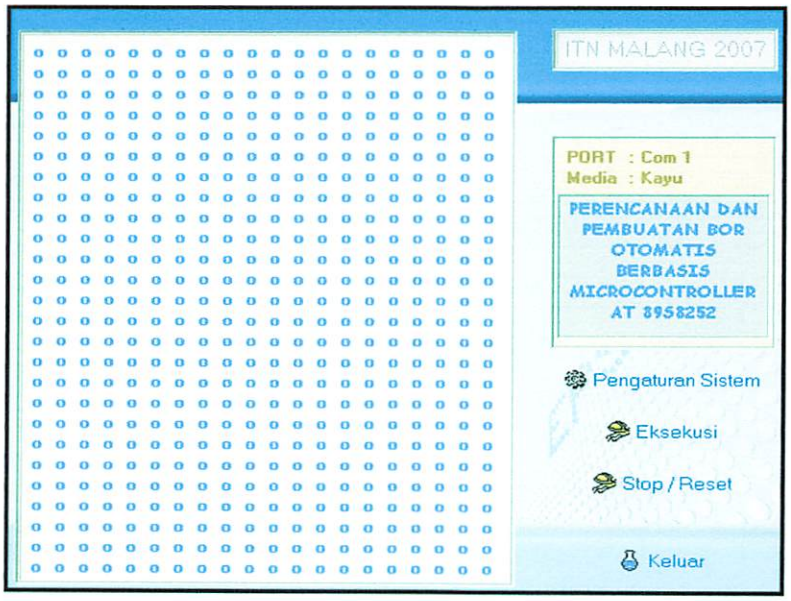
$$\text{Error rata – rata} = \frac{\sum error}{BanyaknyaPercobaan} \times 100\ \%$$

$$= \frac{130}{10} \times 100\ \% = 13\ \%$$

4.5. Pengujian Sistem Secara keseluruhan

4.5.1. Tampilan Umum

- ❖ Memberikan power pada alat.
- ❖ Menghidupkan PC
- ❖ Membuka program " Bor Otomatis "

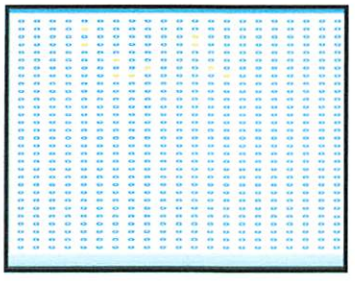


Gambar 4.6. Form Tampilan Utama Sistem

4.5.1.1. Proses Pengeboran

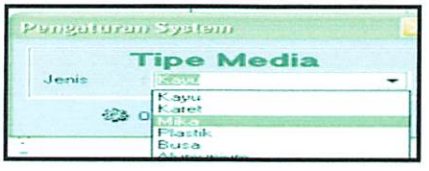
Langkah –langkah :

- Hubungkan dan cek dengan pasti bahwa sistem telah siap atau connect pada COM1
- ***Klik*** ➔ atau pilih titik – titik yang akan dibor pada tampilan utama.Pastikan tanda pada titik menjadi merah.



Gambar 4.7. Form Tampilan Titik- titik Yang Akan Dibor

- Setelah itu,tentukan jenis media yang akan dibor pada menu pengaturan sistem.***Klik*** ➔ ***Pengaturan Sistem***

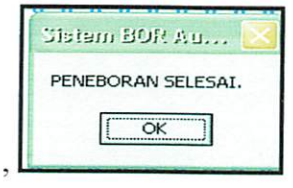


Gambar 4.8. Form Tampilan Pengaturan Sistem

- Setelah melakukan pengaturan sistem dan tidak ada titik lagi yang akan dibor, maka selanjutnya adalah proses pengeboran. ***Klik*** ➔ ***Eksekusi***

Gambar 4.9. Form Tampilan Menu Eksekusi

- Apabila proses telah selesai, maka akan muncul tampilan ***" Pengeboran Selesai"***



Gambar 4.10. Form Tampilan Pengeboran Selesai

- kemudian ***Klik*** ➔ ***Keluar*** jika ingin mengakhirinya.

Gambar 4.11. Form Tampilan Menu Keluar

BAB V

PENUTUP

5.1. Kesimpulan

Selama dalam perancangan dan pengujian sistem bor otomatis berbasis mikrokontroller AT89S8252 yang telah dibuat, maka dapat ditarik kesimpulan antara lain :

1. Pada pengujian rangkaian opto interupt terdapat kesalahan (error) pada hasil perhitungan dan pengujian, yaitu sebesar 2,36 %.
2. Pada pengujian rangkaian limit switch, terdapat kesalahan (error) yaitu, sebesar 1,8 %
3. Hasil pengujian untuk beberapa kali pengeboran terdapat kesalahan (error) sebesar 13 %.

5.2. Saran –saran

1. Penentu jarak pada sumbu X yang berupa rangkaian opto interupts sebaiknya disesuaikan kepresisiannya dengan roda gigi motor, sehingga jarak pengeboran dapat presisi.
2. Kestabilan alat ini akan lebih baik jika menggunakan komponen–komponen dengan mekanik yang berpresisi tinggi dan berkualitas baik.
3. Untuk mendapatkan lubang yang lebih besar, mata bor sebaiknya diperbesar ukurannya.

Daftar Pustaka

[1] www.atmel.com Datasheet of Microcontroller AT89S8252

[2] Frank D. Petruzella, 2002 *Elektronik Industri.* Terjemahan Sumanto. Yogyakarta: Andi

[3] www.senet.com.au/~epeacock Datasheet of RS 232

[4] www.optoelectronics.perkinelmer.com Datasheets of Optocoupler

[5] www.data sheetcatalog.com Datasheets of ST. SGS- THOMSONS Microelectronics

[6] C.M. Prasojo,A. Muljito, Mulyadi W.A.1984.37-56 *Belajar Elektronika (Beltron)*.Jilid Ke-2.Solo:Tiga Serangkai

[7] C.M. Prasojo,A. Muljito, Mulyadi W.A.1984. *Belajar Elektronika (Beltron)*.Jilid ke-1. Solo:Tiga Serangkai

Laporan Praktikum Mikroprosesor. 2005..Bab 4 - 5.Malang: ITN MALANG

Susanto Wibisono Koselan. 2002.*Motor Stepper*. www.alds.stts.edu

LAMPIRAN

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

LEMBAR PERSETUJUAN PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata 1 (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika S-1, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : ARDIAN DWI YUNANTO
N I M : 02.17.070
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika S-1
Judul Skripsi : Perancangan Dan Pembuatan Bor Otomatis Berbasis Mikrokontroller AT89S8252

Perbaikan Meliputi :

No.	Tanggal	Uraian	Paraf Penguji I	Paraf Penguji II
1	6- 9- 2007	- Tujuan dirubah - Ditambahkan gambar rangkaian power supply - Ditambahkan gambar rangkaian lengkap - Pengujian dilakukan 5X - Ditambahkan datasheet LED		
2	6- 9- 2007	- Flowchart diperbaiki		

Disetujui

Penguji I

Ir. Teguh Herbasuki, MT
NIP. Y. 1038900209

Penguji II

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

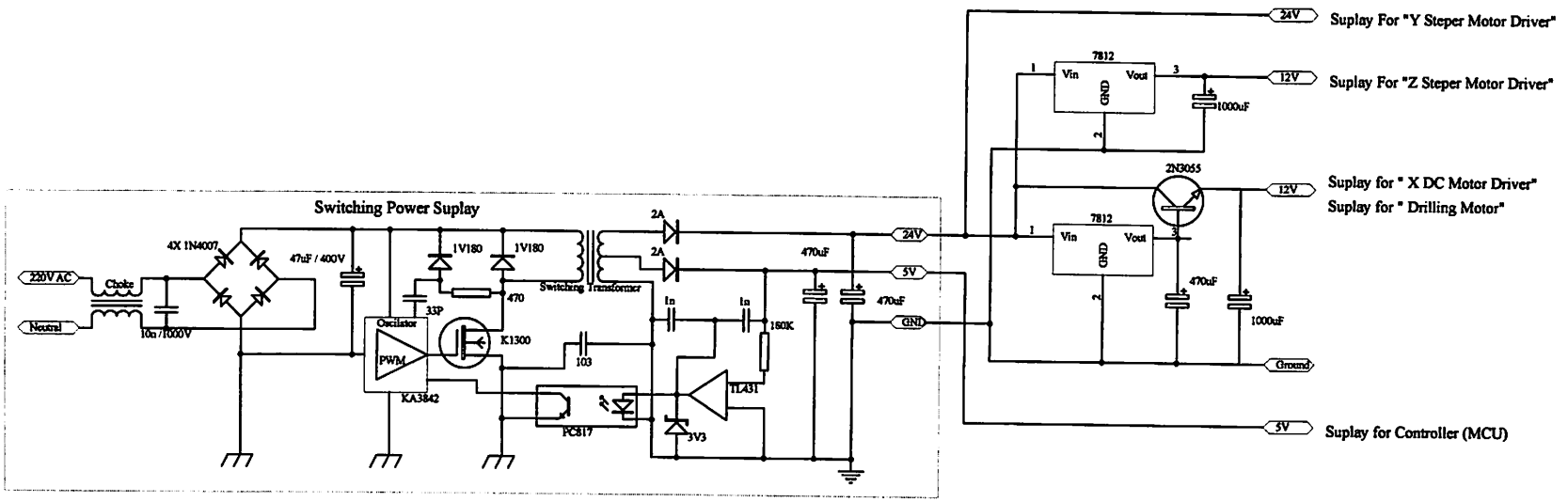
Mengetahui,

Dosen Pembimbing I

Ir. Eko Nurcahyo
NIP. Y. 1028700172

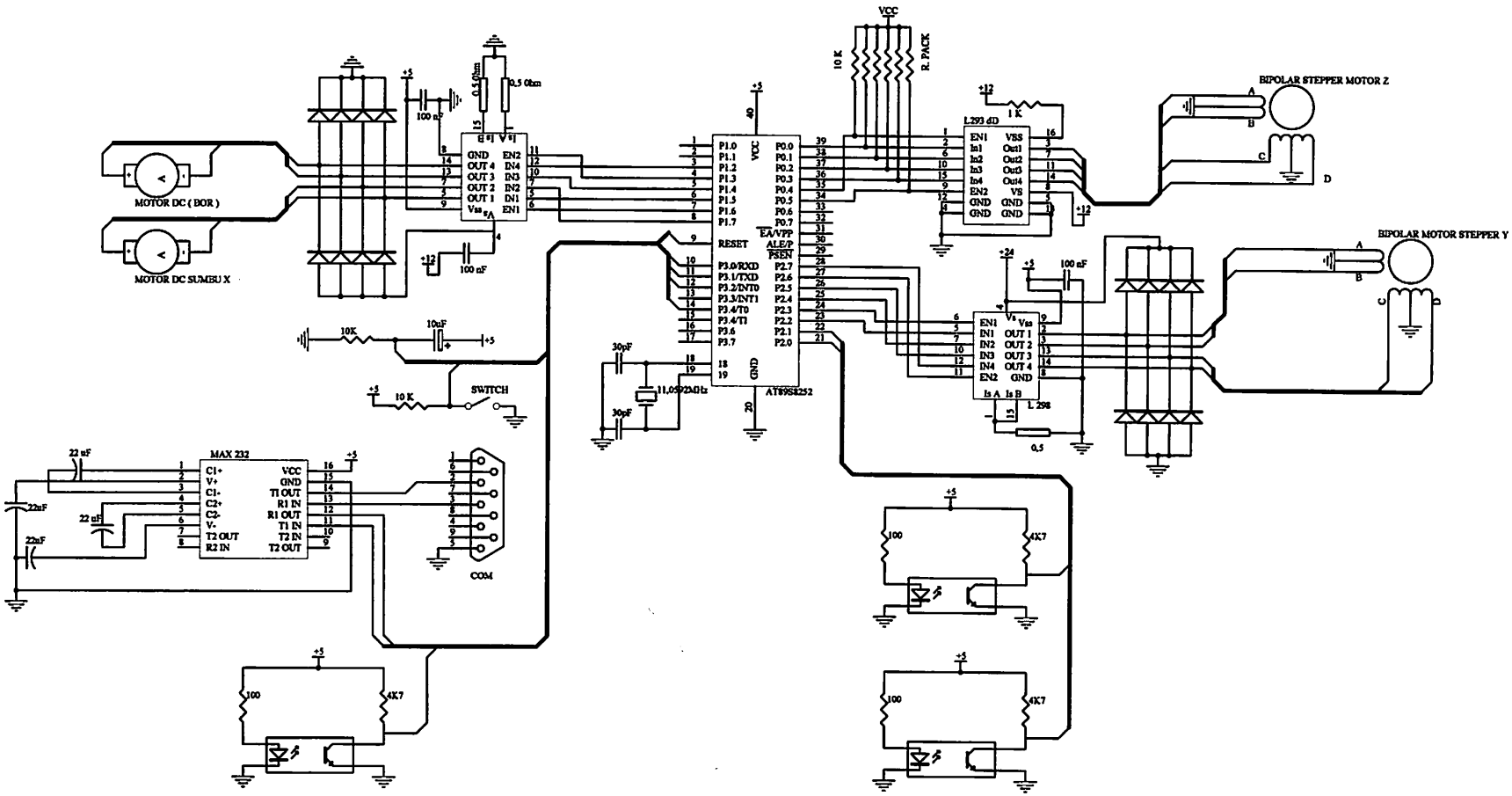
Dosen Pembimbing II

I Komang Somawirata ST, MT
NIP.P. 103010036



Title			
Size: Legal	Number		Revision
Date:	14-Sep-2007	Sheet of	
File:	D:\Program Files\Design Explorer 99 SE\Explorer\Circuit Simulation\ARDIANI.DDB		

KABE-LINA



Title	BOR OTOMATIS BERBASIS MIKROKONTROLLER AT89S8252	
Size: B	Number	Revision
Date: 4-Sep-2007	Sheet of	
File: D:\Program Files\Design Explorer 99 SE\Examples\Tutorial Simulation\Backup of DIAN SCH.DDB		

```
_EN1    BIT     P0.4
_EN2    BIT     P0.5
_IN1    BIT     P0.0
_IN2    BIT     P0.1
_IN3    BIT     P0.2
_IN4    BIT     P0.3

_EN1    BIT     P2.6
_EN2    BIT     P2.3
_IN1    BIT     P2.5
_IN2    BIT     P2.7
_IN3    BIT     P2.4
_IN4    BIT     P2.2

_EN1    BIT     P1.6
_EN2    BIT     P1.3
_IN1    BIT     P1.5
_IN2    BIT     P1.7
_IN3    BIT     P1.4
_IN4    BIT     P1.2


ENSOR_Y         BIT     P2.1
ENSOR_Z         BIT     P3.2
ENSOR_X         BIT     P3.4
ENSOR_JARAK     BIT     P2.0

OR_MAJU         BIT     P0.6
OR_MUNDUR       BIT     P0.7

PEED            EQU     30H
JTARAN          EQU     31H
OUNTER_Y        EQU     32H
EMP             EQU     33H
EMP2            EQU     34H
TEP             EQU     35H
OSISI_X         EQU     36H
OSISI_Y         EQU     37H

ROGRESS         EQU     38H

        ORG     00H
        LJMP    START

        ORG     23H
        LJMP    CEK_SERIAL

        ORG     100H
TART:   SETB    EA
        MOV     TMOD,#20H
        MOV     TH1,#0FDH
        SETB    TR1
        MOV     SCON,#50H
        SETB    ES
        MOV     P0,#0C0H
        MOV     COUNTER_Y,#1
        CALL    PARKING
        JMP     $

EK_SERIAL:
        CLR     ES
        PUSH    ACC
        JNB     RI,$
        MOV     A,SBUF
        CLR     RI
        CJNE    A,#0DH,CEKCOM
        MOV     @R0,A
        CALL    PROCESING
```

```
        MOV     A,#'K'
        MOV     SBUF,A
        JNB     TI,$
        CLR     TI
        JMP     METU
:KCOM:  CJNE    A,#'L',CEKS
        CALL    PARKING
        JMP     METU
:KS:    CJNE    A,#'S',CEKXX
        MOV     A,#'S'
        MOV     SBUF,A
        JNB     TI,$
        CLR     TI
        JMP     METU
:KXX:   CJNE    A,#'X',ISI
        MOV     R0,#PROGRESS
;I:     MOV     @R0,A
        INC     R0
:TU:    POP     ACC
        SETB    ES
        RETI

:ND:    CLR     ES
        MOV     SBUF,A
        JNB     TI,$
        CLR     TI
        SETB    ES
        RET

:OCESING:
        MOV     R0,#PROGRESS
        MOV     A,@R0
        CJNE    A,#'X',WES
        CALL    SCAN_POSISI             ;SCAN POSISI X
:SX:    MOV     A,POSISI_X
        CJNE    A,TEMP,CEKJARAKX
        CALL    MOTOR_X_STOP
        JMP     CEK_Y
:KJARAKX:
        JNC     TURUN_X                 ;JIKA POSISI > SETING
        CALL    X_1CM_MUNDUR            ;POSISI < SETING
        INC     POSISI_X
        JMP     TESX
JRUN_X:CALL     X_1CM_MAJU              ;POSISI > SETING
        DEC     POSISI_X
        JMP     TESX

:K_Y:   CALL    SCAN_POSISI             ;SCAN POSISI Y
:SY:    MOV     A,POSISI_Y
        CJNE    A,TEMP,CEKJARAKY
        JMP     TERUS
:KJARAKY:
        JNC     TURUN_Y                 ;JIKA POSISI > SETING
        CALL    Y_1CM_KIRI              ;POSISI < SETING
        INC     POSISI_Y
        JMP     TESY
JRUN_Y:CALL     Y_1CM_KANAN             ;POSISI > SETING
        DEC     POSISI_Y
        JMP     TESY

:RUS:   CALL    SCAN_POSISI             ;SCAN NILAI SPEED MOTOR STEPPER Z
        MOV     SPEED,TEMP
        CALL    SCAN_POSISI
        CALL    SCAN_POSISI             ;SCAN POSISI STATUS
        MOV     A,TEMP
        CJNE    A,#0,BOR
        JB      SENSOR_X,$
        CALL    MOTOR_X_MAJU
```

```
        JNB     SENSOR_X,$
        CALL    MOTOR_X_STOP
        RET
R:      INC     R0
        MOV     A,@R0
        CJNE    A,#0DH,WES
        CALL    NGEBOR
S:      RET

AN_POSISI:
        INC     R0
        MOV     A,@R0
        CALL    BANDING
        SWAP    A                   ;BALIK KE MSB
        MOV     B,A
        INC     R0
        MOV     A,@R0
        CALL    BANDING
        ORL     A,B
        MOV     TEMP,A
        RET

NDING:  CJNE    A,#'A',CKB
        MOV     A,#0AH
        RET
B:      CJNE    A,#'B',CKC
        MOV     A,#0BH
        RET
C:      CJNE    A,#'C',CKD
        MOV     A,#0CH
        RET
D:      CJNE    A,#'D',CKE
        MOV     A,#0DH
        RET
E:      CJNE    A,#'E',CKF
        MOV     A,#0EH
        RET
F:      CJNE    A,#'F',CK
        MOV     A,#0FH
        RET
:       ANL     A,#0FH
        RET

1CM_MUNDUR:
        CALL    MOTOR_X_MUNDUR
        MOV     STEP,#10
LES:    JNB     SENSOR_JARAK,$
        MOV     R5,#1
        CALL    WAIT
        JB      SENSOR_JARAK,$
        DJNZ    STEP,HOLES
        RET

1CM_MAJU:
        CALL    MOTOR_X_MAJU
        MOV     STEP,#10
LE:     JNB     SENSOR_JARAK,$
        MOV     R5,#1
        CALL    WAIT
        JB      SENSOR_JARAK,$
        DJNZ    STEP,HOLE
        RET

RKING:  MOV     SPEED,#10
        JB      SENSOR_Z,CEKY
OP2:    CALL    STEPER_Z_NAIK
        JNB     SENSOR_Z,LOOP2
```

```
EKY:    djnz    r7,$
        djnz    r6,ceky
n:      djnz    r6,$
        JB      SENSOR_Y,cm2
        jmp     oke
n2:     djnz    r7,cm
        jmp     CEKPOSY

ke:     SETB    Y_IN4           ;DRIVE D
        SETB    Y_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        SETB    Y_IN2           ;drive B
        SETB    Y_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Y_IN2
        CLR     Y_EN1
        CLR     Y_IN4
        CLR     Y_EN2

        MOV     SPEED,#10
OOP4:   CALL    STEPER_Y_KIRI
        JNB     SENSOR_Y,LOOP4
        MOV     SPEED,#80
        CALL    Y_1CM_KIRI
        CALL    Y_1CM_KIRI
        CALL    Y_1CM_KIRI
        CALL    Y_1CM_KIRI
        MOV     STEP,#20
        JMP     CEKX
EKPOSY: CALL    Y_1CM_KANAN
        JMP     CEKY
EKX:    CALL    Y_1CM_KANAN
        DJNZ    STEP,CEKX               ;JUMLAH GESER DALAM SENTI
        JB      SENSOR_X,CEK_JARAK
        CALL    MOTOR_X_MAJU
        JNB     SENSOR_X,$
        CALL    MOTOR_X_STOP

        ;CEK KODISI AWAL SENSOR CM
EK_JARAK:
        JNB     SENSOR_JARAK,ML
        CALL    MOTOR_X_MUNDUR
        JB      SENSOR_JARAK,$
        CALL    MOTOR_X_STOP
L:      MOV     POSISI_X,#1
        MOV     POSISI_Y,#1
        RET

GEBOR:  CALL    MOTOR_BOR_TURUN
        MOV     R3,#50                  ;KEDALAMAN BOR (MOTOR Z)
JDUN:   CALL    STEPER_Z_TURUN
        DJNZ    R3,MUDUN
        MOV     R5,#4
        CALL    WAIT
        CALL    MOTOR_BOR_STOP
        DJNZ    R7,$
        CALL    MOTOR_BOR_NAIK
AGI:    MOV     SPEED,#10
        CALL    STEPER_Z_NAIK
        JNB     SENSOR_Z,LAGI
        CALL    MOTOR_BOR_STOP
        RET

_1CM_KANAN:
        MOV     TEMP2,#22
        MOV     SPEED,#25
EKL:    CALL    STEPER_Y_KANAN
        DJNZ    TEMP2,CEKL
```

```
      RET

1CM_KIRI:
      MOV     TEMP2,#22
      MOV     SPEED,#25
KR:   CALL    STEPER_Y_KIRI
      DJNZ    TEMP2,CEKR
      RET

TOR_BOR_TURUN:
      SETB    X_IN3           ;drive A
      CLR     X_IN4           ;drive B
      SETB    X_EN2           ;ENABLE DRIVE 2
      RET

TOR_BOR_NAIK:
      CLR     X_IN3           ;drive A
      SETB    X_IN4           ;drive B
      SETB    X_EN2           ;ENABLE DRIVE 2
      RET

TOR_BOR_STOP:
      CLR     X_IN3
      CLR     X_IN4
      CLR     X_EN2
:     DJNZ    R7,$
      DJNZ    R6,WT
      RET

TOR_X_MUNDUR:
      SETB    X_IN1           ;drive A
      CLR     X_IN2           ;drive B
      SETB    X_EN1           ;ENABLE DRIVE 1
      RET

TOR_X_MAJU:
      CLR     X_IN1           ;drive A
      SETB    X_IN2           ;drive B
      SETB    X_EN1           ;ENABLE DRIVE 1
      RET

TOR_X_STOP:
      CLR     X_IN1
      CLR     X_IN1
      CLR     X_EN1
      DJNZ    R7,$
      RET

EPER_Z_TURUN:
      SETB    Z_IN1           ;drive A
      SETB    Z_EN1           ;ENABLE DRIVE 1
      CALL    MOTOR_DELAY
      CLR     Z_IN1
      CLR     Z_EN1

      SETB    Z_IN3           ;DRIVE C
      SETB    Z_EN2           ;ENABLE DRIVE 2
      CALL    MOTOR_DELAY
      CLR     Z_IN3
      CLR     Z_EN2

      SETB    Z_IN2           ;drive B
      SETB    Z_EN1           ;ENABLE DRIVE 1
      CALL    MOTOR_DELAY
      CLR     Z_IN2
      CLR     Z_EN1

      SETB    Z_IN4           ;DRIVE D
```

```
        SETB    Z_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Z_IN4
        CLR     Z_EN2
        RET

EPER_Z_NAIK:
        SETB    Z_IN4           ;DRIVE D
        SETB    Z_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Z_IN4
        CLR     Z_EN2

        SETB    Z_IN2           ;drive B
        SETB    Z_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Z_IN2
        CLR     Z_EN1

        SETB    Z_IN3           ;DRIVE C
        SETB    Z_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Z_IN3
        CLR     Z_EN2

        SETB    Z_IN1           ;drive A
        SETB    Z_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Z_IN1
        CLR     Z_EN1
        RET


EPER_Y_KANAN:
        MOV     R4,COUNTER_Y
        CJNE    R4,#1,Y2
        SETB    Y_IN1           ;drive A
        SETB    Y_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Y_IN1
        CLR     Y_EN1
        MOV     COUNTER_Y,#2
        RET

2:      CJNE    R4,#2,Y3
        SETB    Y_IN3           ;DRIVE C
        SETB    Y_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Y_IN3
        CLR     Y_EN2
        MOV     COUNTER_Y,#3
        RET

3:      CJNE    R4,#3,Y4
        SETB    Y_IN2           ;drive B
        SETB    Y_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Y_IN2
        CLR     Y_EN1
        MOV     COUNTER_Y,#4
        RET

4:      CJNE    R4,#4,YOUT
        SETB    Y_IN4           ;DRIVE D
        SETB    Y_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Y_IN4
        CLR     Y_EN2
```

```
        MOV     COUNTER_Y,#1
)UT:    RET

'EPER_Y_KIRI:
        MOV     R4,COUNTER_Y
        CJNE    R4,#1,Y2B
        SETB    Y_IN4           ;DRIVE D
        SETB    Y_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Y_IN4
        CLR     Y_EN2
        MOV     COUNTER_Y,#2
        RET

?B:     CJNE    R4,#2,Y3B
        SETB    Y_IN2           ;drive B
        SETB    Y_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Y_IN2
        CLR     Y_EN1
        MOV     COUNTER_Y,#3
        RET

3B:     CJNE    R4,#3,Y4B
        SETB    Y_IN3           ;DRIVE C
        SETB    Y_EN2           ;ENABLE DRIVE 2
        CALL    MOTOR_DELAY
        CLR     Y_IN3
        CLR     Y_EN2
        MOV     COUNTER_Y,#4
        RET

4B:     CJNE    R4,#4,YBOUT
        SETB    Y_IN1           ;drive A
        SETB    Y_EN1           ;ENABLE DRIVE 1
        CALL    MOTOR_DELAY
        CLR     Y_IN1
        CLR     Y_EN1
        MOV     COUNTER_Y,#1
BOUT:   RET

)TOR_DELAY:
        MOV     R6,SPEED
ELM:    DJNZ    R7,$
        DJNZ    R6,DELM
        RET

ELAY:   MOV     R5,#5
AIT:    DJNZ    R7,$
        DJNZ    R6,WAIT
        DJNZ    R5,WAIT
        RET
        END
```

```
program Project1;

uses
  Forms,
  Unit1 in 'Unit1.pas' {Form1},
  Unit2 in 'Unit2.pas' {Form2};

{$R *.res}

begin
  Application.Initialize;
  Application.Title := 'Sistem BOR Automatis';
  Application.CreateForm(TForm1, Form1);
  Application.CreateForm(TForm2, Form2);
  Application.Run;
end.

program Project1;

uses
  Forms,
  Unit1 in 'Unit1.pas' {Form1},
  Unit2 in 'Unit2.pas' {Form2};

{$R *.res}

begin
  Application.Initialize;
  Application.CreateForm(TForm1, Form1);
  Application.CreateForm(TForm2, Form2);
  Application.Run;
end.
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, QuickRpt, QRCtrls, ExtCtrls, StdCtrls, AppEvnts, Buttons, CPort;

type
  TForm1 = class(TForm)
    Memo1: TMemo;
    SpeedButton1: TSpeedButton;
    SpeedButton2: TSpeedButton;
    SpeedButton3: TSpeedButton;
```

```
  Panel1: TPanel;
  x0y0: TLabel;
  x2y0: TLabel;
  x3y0: TLabel;
  x4y0: TLabel;
  x5y0: TLabel;
  x6y0: TLabel;
  x7y0: TLabel;
  x8y0: TLabel;
  x9y0: TLabel;
  x10y0: TLabel;
  x11y0: TLabel;
  x12y0: TLabel;
  x13y0: TLabel;
  x14y0: TLabel;
  x16y0: TLabel;
  x17y0: TLabel;
  x18y0: TLabel;
  x19y0: TLabel;
  x20y0: TLabel;
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
  x10y29: TLabel;
  x11y29: TLabel;
  x12y29: TLabel;
  x9y29: TLabel;
  x8y29: TLabel;
  x7y29: TLabel;
  x18y26: TLabel;
  x19y26: TLabel;
  x12y28: TLabel;
  x17y26: TLabel;
  x16y26: TLabel;
  x15y26: TLabel;
  x20y26: TLabel;

  ComPort1: TComPort;
  Timer1: TTimer;
  Timer2: TTimer;
  Image1: TImage;
  Image2: TImage;
  StaticText1: TStaticText;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Panel3: TPanel;
```

```
  procedure X1Y1StartDock(Sender: TObject;
    var DragObject: TDragDockObject);
  procedure x10y1Click(Sender: TObject);
  procedure x2y1Click(Sender: TObject);
  procedure SpeedButton3Click(Sender: TObject);
  procedure SpeedButton1Click(Sender: TObject);
  procedure x1y1Click(Sender: TObject);
  procedure x4y1Click(Sender: TObject);
  procedure x2y0Click(Sender: TObject);
  procedure x6y1Click(Sender: TObject);
  procedure x3y0Click(Sender: TObject);
  procedure x8y1Click(Sender: TObject);
  procedure x4y0Click(Sender: TObject);
  procedure x5y0Click(Sender: TObject);
  procedure x12y1Click(Sender: TObject);
  procedure x6y0Click(Sender: TObject);
  procedure x14y1Click(Sender: TObject);
  procedure x7y0Click(Sender: TObject);
  procedure x16y1Click(Sender: TObject);
  procedure x8y0Click(Sender: TObject);
  procedure x18y1Click(Sender: TObject);
  procedure x9y0Click(Sender: TObject);
  procedure Panel1GetSiteInfo(Sender: TObject; DockClient: TControl;
    var InfluenceRect: TRect; MousePos: TPoint; var CanDock: Boolean);
  procedure SpeedButton2Click(Sender: TObject);
  procedure x12y0Click(Sender: TObject);
  procedure x13y0Click(Sender: TObject);
  procedure x14y0Click(Sender: TObject);
  procedure x0y0Click(Sender: TObject);
  procedure x10y0Click(Sender: TObject);
  procedure x15y0Click(Sender: TObject);
  procedure x16y0Click(Sender: TObject);
  procedure x17y0Click(Sender: TObject);
  procedure x18y0Click(Sender: TObject);
  procedure x19y0Click(Sender: TObject);
  procedure x20y0Click(Sender: TObject);
```

```
  procedure x0y29Click(Sender: TObject);
  procedure x1y29Click(Sender: TObject);
  procedure x2y29Click(Sender: TObject);
  procedure x3y29Click(Sender: TObject);
  procedure x4y29Click(Sender: TObject);
  procedure x5y29Click(Sender: TObject);
  procedure x6y29Click(Sender: TObject);
  procedure x7y29Click(Sender: TObject);
```

```
    procedure x8y29Click(Sender: TObject);
    procedure x9y29Click(Sender: TObject);
    procedure x10y29Click(Sender: TObject);
    procedure x11y29Click(Sender: TObject);
    procedure x12y29Click(Sender: TObject);
    procedure x13y29Click(Sender: TObject);
    procedure x14y29Click(Sender: TObject);
    procedure x15y29Click(Sender: TObject);
    procedure x16y29Click(Sender: TObject);
    procedure x17y29Click(Sender: TObject);
    procedure x18y29Click(Sender: TObject);
    procedure x19y29Click(Sender: TObject);
    procedure x20y29Click(Sender: TObject);
    procedure ComPort1RxChar(Sender: TObject; Count: Integer);
    procedure x1y0Click(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure x11y0Click(Sender: TObject);
    procedure x4Y4Click(Sender: TObject);
    procedure Timer2Timer(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
  DATA_SERIAL,seting,nilai_x,SPEED:STRING;
  FIRST:BOOLEAN;
  HITUNGAN,timeout:INTEGER;
implementation

uses Unit2;

{$R *.dfm}

procedure TForm1.X1Y1StartDock(Sender: TObject;
  var DragObject: TDragDockObject);
begin
MEMO1.Text:='PO';
end;

procedure TForm1.x10y1Click(Sender: TObject);
begin
```

```
IF x10y1.Font.Color=CLBLUE then
begin
 x10y1.Font.Color:=CLred;
 exit;
end else
if x10y1.Font.Color=CLred then
begin
   x10y1.Font.Color:=CLblue;
end;

end;

procedure TForm1.x2y1Click(Sender: TObject);
begin
IF x2y1.Font.Color=CLBLUE then
begin
 x2y1.Font.Color:=CLred;
 exit;
end else
if x2y1.Font.Color=CLred then
begin
   x2y1.Font.Color:=CLblue;
end;
end;

procedure TForm1.SpeedButton3Click(Sender: TObject);
begin
     close;
end;

procedure TForm1.SpeedButton1Click(Sender: TObject);
begin
     form2.show;
end;

procedure TForm1.x1y1Click(Sender: TObject);
begin
IF x1y1.Font.Color=CLBLUE then
begin
 x1y1.Font.Color:=CLred;
 exit;
end else
if x1y1.Font.Color=CLred then
begin
 x1y1.Font.Color:=CLblue;
end;
```

```
end;

procedure TForm1.x4y1Click(Sender: TObject);
begin
IF x4y1.Font.Color=CLBLUE then
begin
 x4y1.Font.Color:=CLred;
 exit;
end else
if x4y1.Font.Color=CLred then
begin
   x4y1.Font.Color:=CLblue;
end;
end;

end;

end;

procedure TForm1.Panel1GetSiteInfo(Sender: TObject; DockClient: TControl;
 var InfluenceRect: TRect; MousePos: TPoint; var CanDock: Boolean);
begin
memo1.Lines.Add(inttostr(mousepos.X));
end;

procedure TForm1.SpeedButton2Click(Sender: TObject);
begin
      memo1.Text:='X01:';
      if x0y0.Font.Color=clred  then memo1.Text:=memo1.Text+'01:';
      if x1y0.Font.Color=clred  then memo1.Text:=memo1.Text+'02:';
      if x2y0.Font.Color=clred  then memo1.Text:=memo1.Text+'03:';
      if x3y0.Font.Color=clred  then memo1.Text:=memo1.Text+'04:';
      if x4y0.Font.Color=clred  then memo1.Text:=memo1.Text+'05:';
      if x5y0.Font.Color=clred  then memo1.Text:=memo1.Text+'06:';
      if x6y0.Font.Color=clred  then memo1.Text:=memo1.Text+'07:';
      if x7y0.Font.Color=clred  then memo1.Text:=memo1.Text+'08:';
      if x8y0.Font.Color=clred  then memo1.Text:=memo1.Text+'09:';
      if x9y0.Font.Color=clred then memo1.Text:=memo1.Text+'0A:';
      if x10Y0.Font.Color=clred then memo1.Text:=memo1.Text+'0B:';
      if x11y0.Font.Color=clred then memo1.Text:=memo1.Text+'0C:';
      if x12y0.Font.Color=clred then memo1.Text:=memo1.Text+'0D:';
      if x13y0.Font.Color=clred then memo1.Text:=memo1.Text+'0E:';
      if x14y0.Font.Color=clred then memo1.Text:=memo1.Text+'0F:';
      if x15y0.Font.Color=clred then memo1.Text:=memo1.Text+'10:';
      if x16y0.Font.Color=clred then memo1.Text:=memo1.Text+'11:';
      if x17y0.Font.Color=clred then memo1.Text:=memo1.Text+'12:';
```

```
      if x18y0.Font.Color=clred then memo1.Text:=memo1.Text+'13:';
      if x19y0.Font.Color=clred then memo1.Text:=memo1.Text+'14:';
      if x20y0.Font.Color=clred then memo1.Text:=memo1.Text+'15:';

      memo1.Text:=memo1.Text+'X02:';
      if x0y1.Font.Color=clred  then memo1.Text:=memo1.Text+'01:';
      if x1y1.Font.Color=clred  then memo1.Text:=memo1.Text+'02:';
      if x2y1.Font.Color=clred  then memo1.Text:=memo1.Text+'03:';
      if x3y1.Font.Color=clred  then memo1.Text:=memo1.Text+'04:';
      if x4y1.Font.Color=clred  then memo1.Text:=memo1.Text+'05:';
      if x5y1.Font.Color=clred  then memo1.Text:=memo1.Text+'06:';
      if x6y1.Font.Color=clred  then memo1.Text:=memo1.Text+'07:';
      if x7y1.Font.Color=clred  then memo1.Text:=memo1.Text+'08:';
      if x8y1.Font.Color=clred  then memo1.Text:=memo1.Text+'09:';
      if x9y1.Font.Color=clred then memo1.Text:=memo1.Text+'0A:';
      if x10Y1.Font.Color=clred then memo1.Text:=memo1.Text+'0B:';
      if x11y1.Font.Color=clred then memo1.Text:=memo1.Text+'0C:';
      if x12y1.Font.Color=clred then memo1.Text:=memo1.Text+'0D:';
      if x13y1.Font.Color=clred then memo1.Text:=memo1.Text+'0E:';
      if x14y1.Font.Color=clred then memo1.Text:=memo1.Text+'0F:';
      if x15y1.Font.Color=clred then memo1.Text:=memo1.Text+'10:';
      if x16y1.Font.Color=clred then memo1.Text:=memo1.Text+'11:';
      if x17y1.Font.Color=clred then memo1.Text:=memo1.Text+'12:';
      if x18y1.Font.Color=clred then memo1.Text:=memo1.Text+'13:';
      if x19y1.Font.Color=clred then memo1.Text:=memo1.Text+'14:';
      if x20y1.Font.Color=clred then memo1.Text:=memo1.Text+'15:';

        memo1.Text:=memo1.Text+'END.';
     seting:=memo1.Text;
     hitungan:=1;
     TIMER1.Enabled:=TRUE;
     speedbutton1.Enabled:=false;
     speedbutton2.Enabled:=false;
     FIRST:=TRUE;
end;


procedure TForm1.x0y0Click(Sender: TObject);
begin
IF x0y0.Font.Color=CLBLUE then
begin
 x0y0.Font.Color:=CLred;
 exit;
end else
if x0y0.Font.Color=CLred then
begin
```

```
    x0y0.Font.Color:=CLblue;
 end;
 end;

 procedure TForm1.x10y0Click(Sender: TObject);
 begin
 IF x10y0.Font.Color=CLBLUE then
 begin
  x10y0.Font.Color:=CLred;
  exit;
 end else
 if x10y0.Font.Color=CLred then
 begin
    x10y0.Font.Color:=CLblue;
 end;
 end;

 procedure TForm1.x15y0Click(Sender: TObject);
 begin
 IF x15y0.Font.Color=CLBLUE then
 begin
  x15y0.Font.Color:=CLred;
  exit;
 end else
 if x15y0.Font.Color=CLred then
 begin
    x15y0.Font.Color:=CLblue;
 end;
 end;

 end else
 if x3y17.Font.Color=CLred then
 begin
    x3y17.Font.Color:=CLblue;
 end;

 end;

 procedure TForm1.x4y17Click(Sender: TObject);
 begin
 IF x4y17.Font.Color=CLBLUE then
 begin
  x4y17.Font.Color:=CLred;
  exit;
 end else
 if x4y17.Font.Color=CLred then
```

```
procedure TForm1.x17y29Click(Sender: TObject);
begin
IF x17y29.Font.Color=CLBLUE then
begin
  x17y29.Font.Color:=CLred;
  exit;
end else
if x17y29.Font.Color=CLred then
begin
   x17y29.Font.Color:=CLblue;
end;

end;

procedure TForm1.x18y29Click(Sender: TObject);
begin
IF x18y29.Font.Color=CLBLUE then
begin
  x18y29.Font.Color:=CLred;
  exit;
end else
if x18y29.Font.Color=CLred then
begin
   x18y29.Font.Color:=CLblue;
end;

end;

procedure TForm1.x19y29Click(Sender: TObject);
begin
IF x19y29.Font.Color=CLBLUE then
begin
  x19y29.Font.Color:=CLred;
  exit;
end else
if x19y29.Font.Color=CLred then
begin
   x19y29.Font.Color:=CLblue;
end;

end;

procedure TForm1.x20y29Click(Sender: TObject);
begin
IF x20y29.Font.Color=CLBLUE then
begin
```

```
  x20y29.Font.Color:=CLred;
  exit;
end else
if x20y29.Font.Color=CLred then
begin
   x20y29.Font.Color:=CLblue;
end;

end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
begin
      COMPORT1.ReadStr(DATA_SERIAL,COUNT);
      IF DATA_SERIAL='S' Then
      begin
            label1.Caption:='Status : Connect.';
            timeout:=0;
      end;
end;

procedure TForm1.x1y0Click(Sender: TObject);
begin
IF x1y0.Font.Color=CLBLUE then
begin
 x1y0.Font.Color:=CLred;
 exit;
end else
if x1y0.Font.Color=CLred then
begin
   x1y0.Font.Color:=CLblue;
end;

end;

procedure TForm1.Timer1Timer(Sender: TObject);
var format,temp:string;
begin
IF (FIRST=TRUE) OR(DATA_SERIAL='K') THEN
begin
      if seting='' then exit;
      temp:=seting[1];
      if temp='X' then
      begin
            nilai_x:=copy(seting,1,3);
            delete(seting,1,4);
      end else
```

```
      if SETING='END.' then
      begin
           timer1.Enabled:=false;
           DATA_SERIAL:='';
           speedbutton1.Enabled:=true;
           speedbutton2.Enabled:=true;
           COMPORT1.WriteStr('L');
           SHOWMESSAGE('PENEBORAN SELESAI.');
      end else
      begin
           FIRST:=FALSE;
           temp:=copy(seting,1,2);  //ambil nilai Y
           delete(seting,1,3);
           format:=nilai_x+temp;
           DATA_SERIAL:='';
          COMPORT1.WriteStr(FORMAT+speed+'0100'+#13);
      end;
end;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
      MEMO1.Hide;
      FIRST:=FALSE;
      HITUNGAN:=0;
      timeout:=0;
end;

procedure TForm1.x11y0Click(Sender: TObject);
begin
IF x11y0.Font.Color=CLBLUE then
begin
 x11y0.Font.Color:=CLred;
 exit;
end else
if x11y0.Font.Color=CLred then
begin
   x11y0.Font.Color:=CLblue;
end;
end;

procedure TForm1.x4Y4Click(Sender: TObject);
begin
    IF x4y4.Font.Color=CLBLUE then
begin
 x4y4.Font.Color:=CLred;
```

```
  exit;
end else
if x4y4.Font.Color=CLred then
begin
   x4y4.Font.Color:=CLblue;
end;
end;

procedure TForm1.Timer2Timer(Sender: TObject);
begin
      timeout:=timeout+1;
      if timeout=10 then label1.Caption:='Status : DisConnect.';
      label2.Caption:='PORT  : Com 1';
      label3.Caption:='Media  : '+form2.ComboBox1.Text;
      If form2.ComboBox1.ItemIndex=0 then speed:='20';       // seting speed kayu
      If form2.ComboBox1.ItemIndex=1 then speed:='30';       // seting speed karet
      If form2.ComboBox1.ItemIndex=2 then speed:='35';       // seting speed mika
      If form2.ComboBox1.ItemIndex=3 then speed:='25';       // seting speed plastic
      If form2.ComboBox1.ItemIndex=4 then speed:='04';       // seting speed busa
      If form2.ComboBox1.ItemIndex=5 then speed:='60';       // seting speed aluminium


end;

end.
```

8-bit Microcontroller with 8K Bytes Flash

AT89S8252

Not Recommended for New Designs. Use AT89S8253.

Features

- Compatible with MCS®51 Products
- 8K Bytes of In-System Reprogrammable Downloadable Flash Memory
  - SPI Serial Interface for Program Downloading
  - Endurance: 1,000 Write/Erase Cycles
- 2K Bytes EEPROM
  - Endurance: 100,000 Write/Erase Cycles
- 4V to 6V Operating Range
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Nine Interrupt Sources
- Programmable UART Serial Channel
- SPI Serial Interface
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down
- Programmable Watchdog Timer
- Dual Data Pointer
- Power-off Flag

Description

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

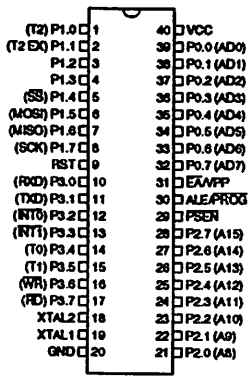
he AT89S8252 provides the following standard features: 8K bytes of downloadable ash, 2K bytes of EEPROM, 256 bytes of RAM, 32 I/O lines, programmable watchdog ner, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt chitecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, e AT89S8252 is designed with static logic for operation down to zero frequency and pports two software selectable power saving modes. The Idle Mode stops the CPU hile allowing the RAM, timer/counters, serial port, and interrupt system to continue nctioning. The Power-down mode saves the RAM contents but freezes the oscillator, sabling all other chip functions until the next external interrupt or hardware reset.

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

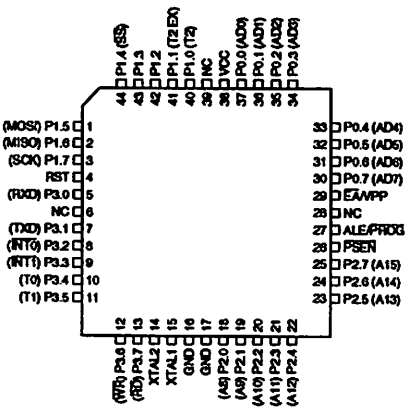
0401G–MICRO–3/06

Pin Configurations

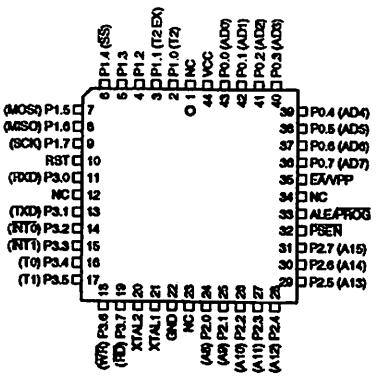
PDIP



TQFP



PLCC



Pin Description

VCC Supply voltage.

GND Ground.

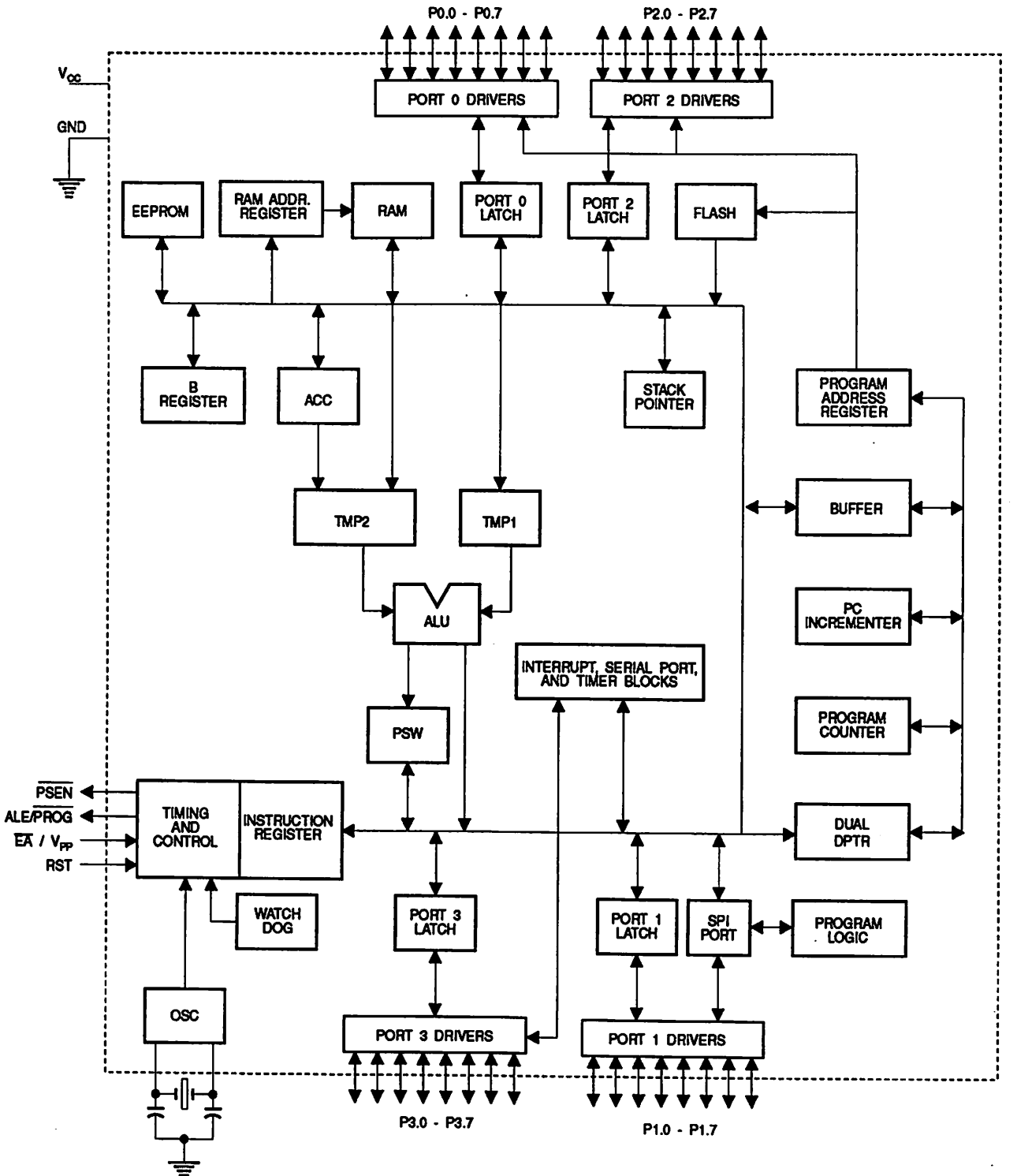
Port 0 Port 0 is an 8-bit open drain bi-didirectional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

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

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

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

Block Diagram



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

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

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

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

ort 2

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

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

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

ort 3

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

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

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

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

ST

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

LE/$\overline{PROG}$

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.

$\overline{SEN}$

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.

$\overline{A}$/VPP

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

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

TAL1

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

TAL2

Output from the inverting oscillator amplifier.

Special Function Registers

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

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

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

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

Table 1. AT89S8252 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000					SPCR 000001XX			0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX						0AFH
0A0H	P2 11111111								0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111						WMCON 00000010		97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

Table 2. T2CON – Timer/Counter 2 Control Register

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

Bit Addressable

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

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

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

Table 3. WMCON—Watchdog and Memory Control Register

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

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

Symbol	Function
PS2 PS1 PS0	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.
EEMWE	EEPROM Data Memory Write Enable Bit. Set this bit to "1" before initiating byte write to on-chip EEPROM with the MOVX instruction. User software should set this bit to "0" after EEPROM write is completed.
EEMEN	Internal EEPROM Access Enable. When EEMEN = 1, the MOVX instruction with DPTR will access on-chip EEPROM instead of external data memory. When EEMEN = 0, MOVX with DPTR accesses external data memory.
DPS	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1
WDTRST RDY/$\overline{BSY}$	Watchdog Timer Reset and EEPROM Ready/$\overline{Busy}$ 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.
WDTEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

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

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

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

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

able 4. SPCR – SPI Control Register

SPCR Address = D5H — Reset Value = 0000 01XXB

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

Symbol	Function
SPIE	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.
SPE	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.
DORD	Data Order. DORD = 1 selects LSB first data transmission. DORD = 0 selects MSB first data transmission.
MSTR	Master/Slave Select. MSTR = 1 selects Master SPI mode. MSTR = 0 selects Slave SPI mode.
CPOL	Clock Polarity. When CPOL = 1, SCK is high when idle. When CPOL = 0, SCK of the master device is low when not transmitting. Please refer to figure on SPI Clock Phase and Polarity Control.
CPHA	Clock Phase. The CPHA bit together with the CPOL bit controls the clock and data relationship between master and slave. Please refer to figure on SPI Clock Phase and Polarity Control.
SPR0 SPR1	SPI Clock Rate Select. These two bits control the SCK rate of the device configured as master. SPR1 and SPR0 have no effect on the slave. The relationship between SCK and the oscillator frequency, F_{osc}, is as follows: SPR1 SPR0 SCK = F_{osc} divided by 0 0 4 0 1 16 1 0 64 1 1 128

able 5. SPSR – SPI Status Register

SPSR Address = AAH — Reset Value = 00XX XXXXB

Bit	7	6	5	4	3	2	1	0
	SPIF	WCOL	–	–	–	–	–	–

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

able 6. SPDR – SPI Data Register

SPDR Address = 86H — Reset Value = unchanged

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

)ata Memory – :EPROM and RAM

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

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

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

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

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

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

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

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

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

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

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

Programmable Watchdog Timer

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

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

Table 7. Watchdog Timer Period Selection

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

imer 0 and 1

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

imer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 8.

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

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

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

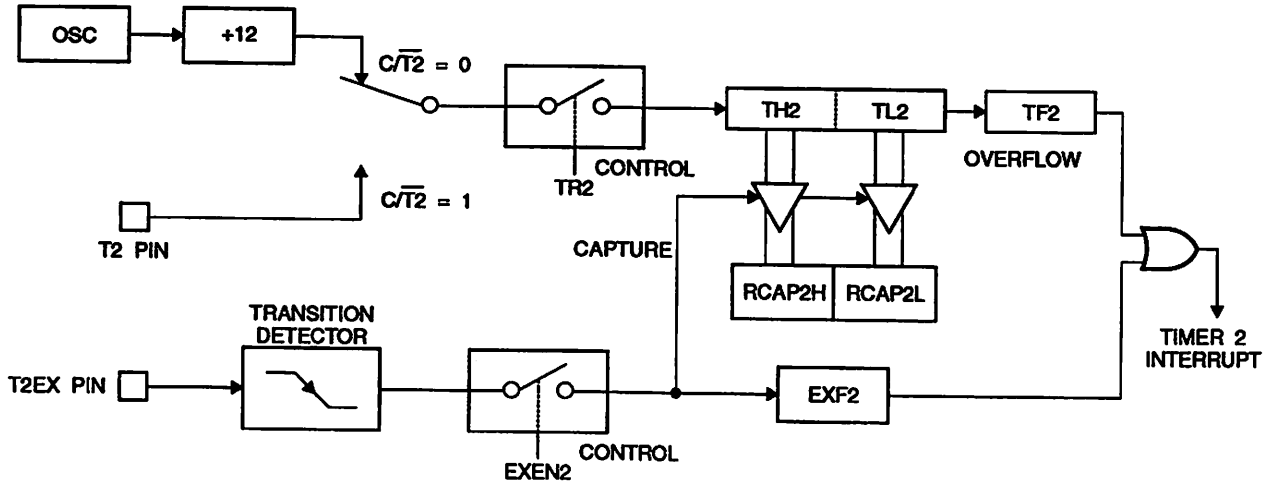
Table 8. Timer 2 Operating Modes

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

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

Figure 1. Timer 2 in Capture Mode



Auto-reload (Up or Down Counter)

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

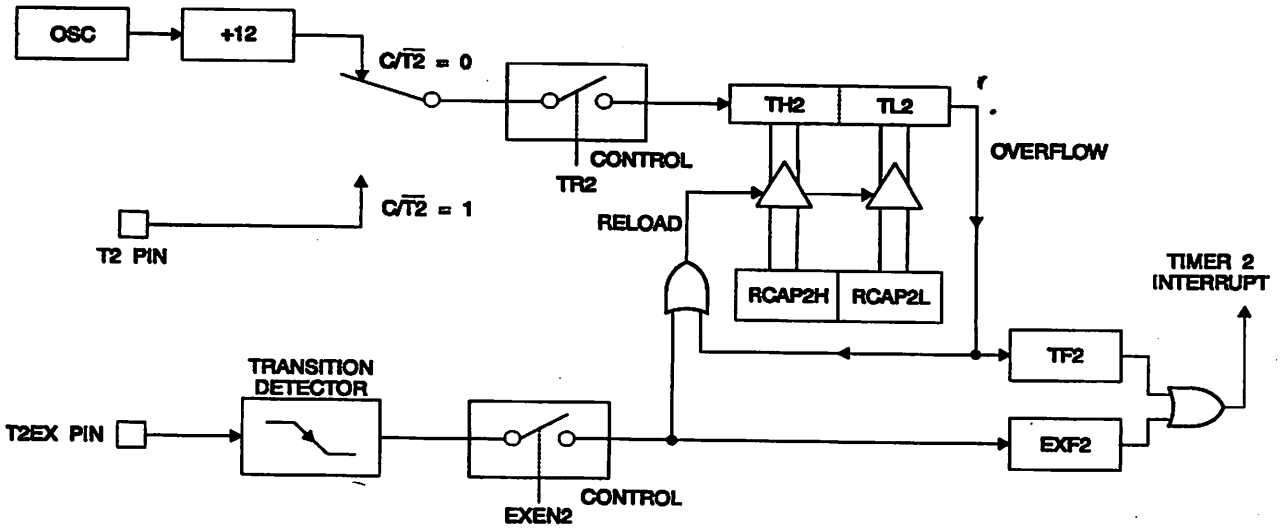
Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H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

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

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

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

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



able 9. T2MOD – Timer 2 Mode Control Register

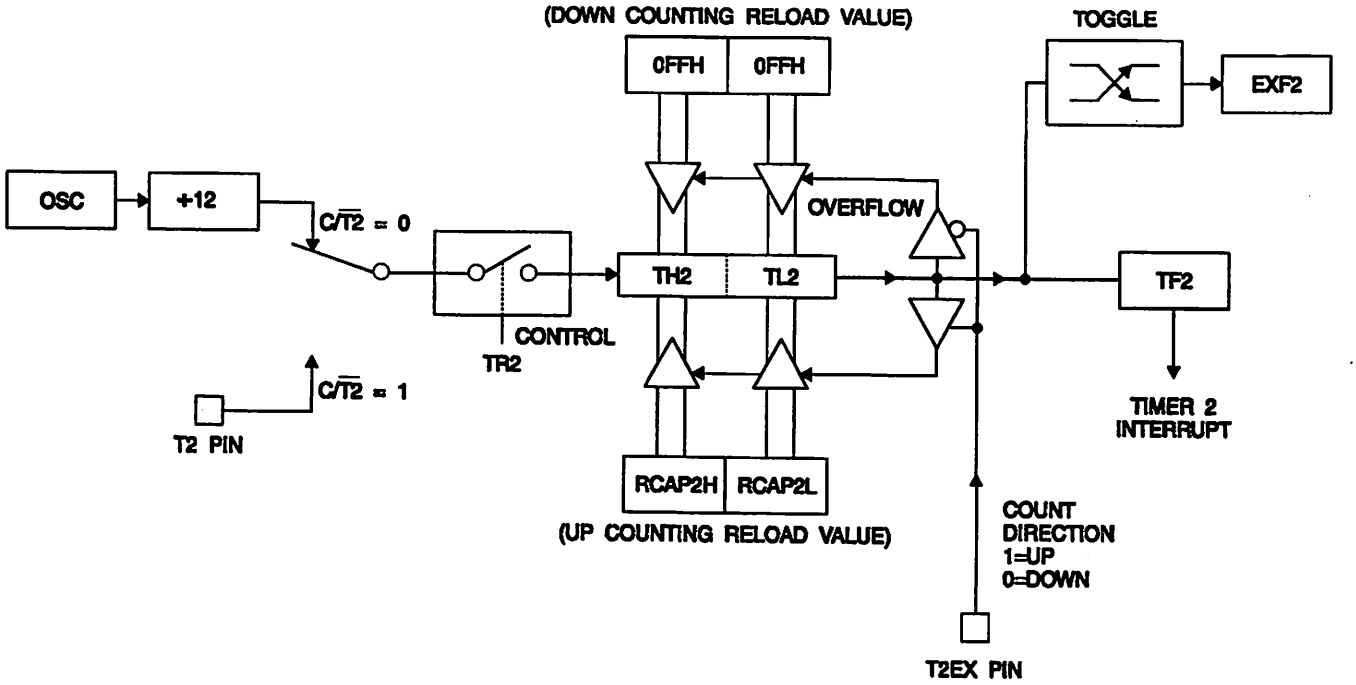
T2MOD Address = 0C9H — Reset Value = XXXX XX00B

Not Bit Addressable

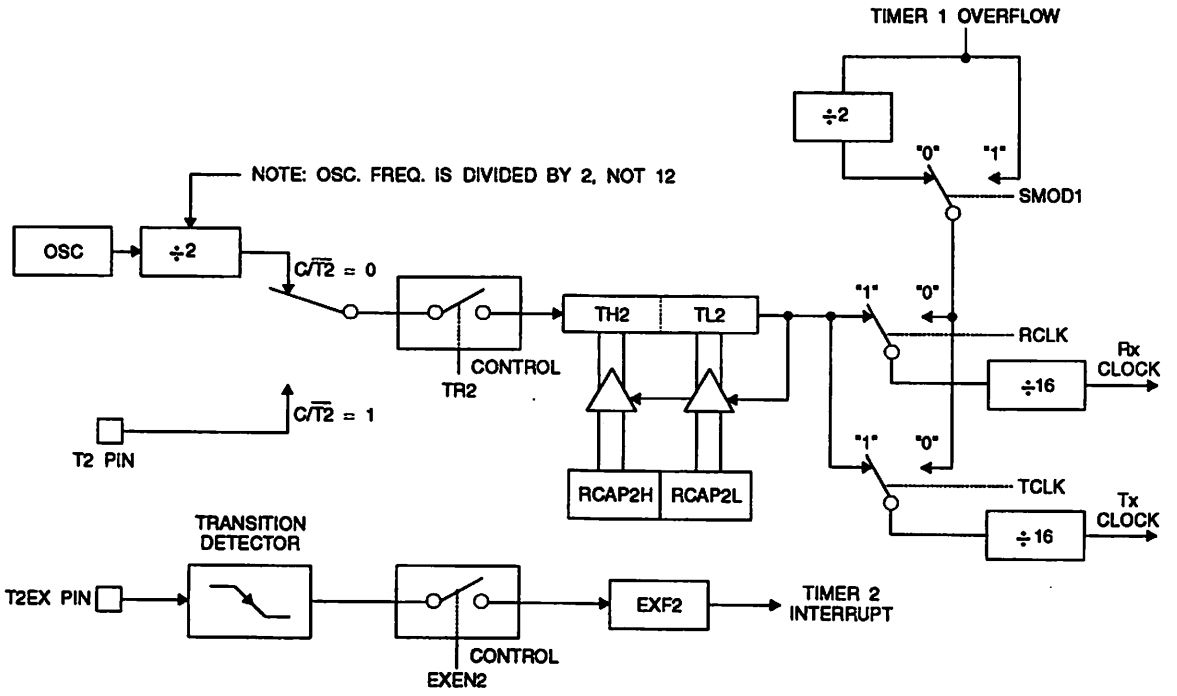
Bit	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	T2OE	DCEN

Symbol	Function
–	Not implemented, reserved for future use.
T2OE	Timer 2 Output Enable bit.
DCEN	When set, this bit allows Timer 2 to be configured as an up/down counter.

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



igure 4. Timer 2 in Baud Rate Generator Mode



]aud Rate Generator

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

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

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

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

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

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

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

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

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

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 (for a 16-MHz operating frequency).

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

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

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

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

Figure 5. Timer 2 in Clock-out Mode

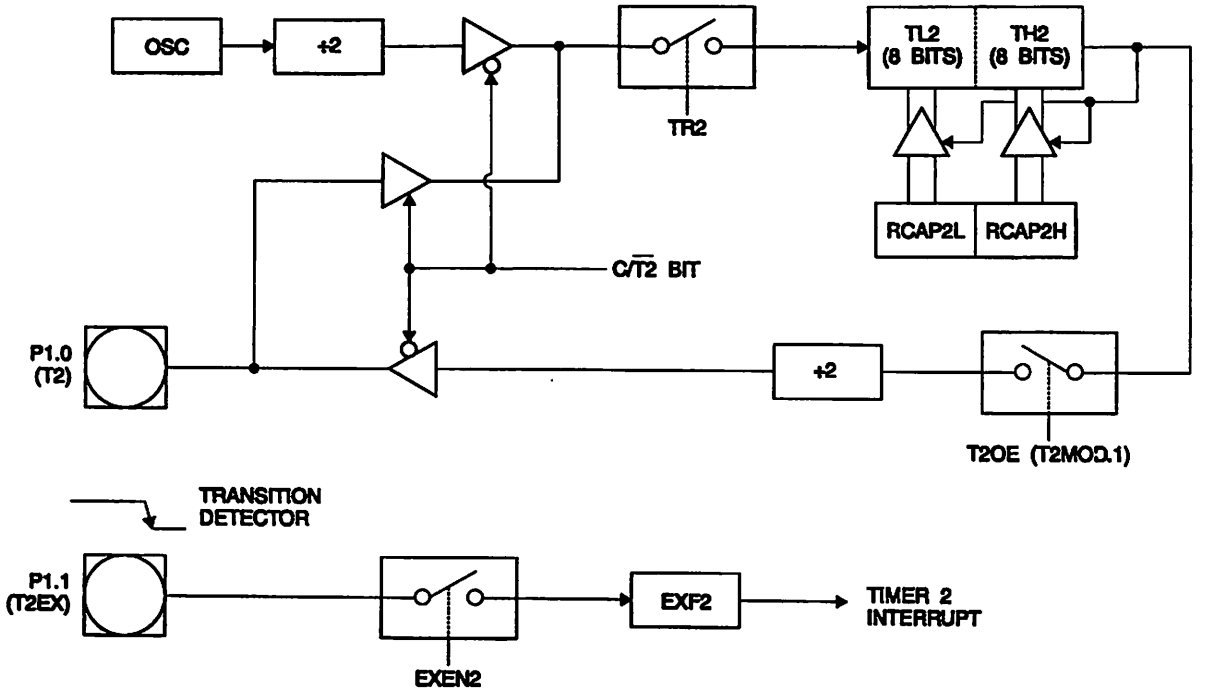
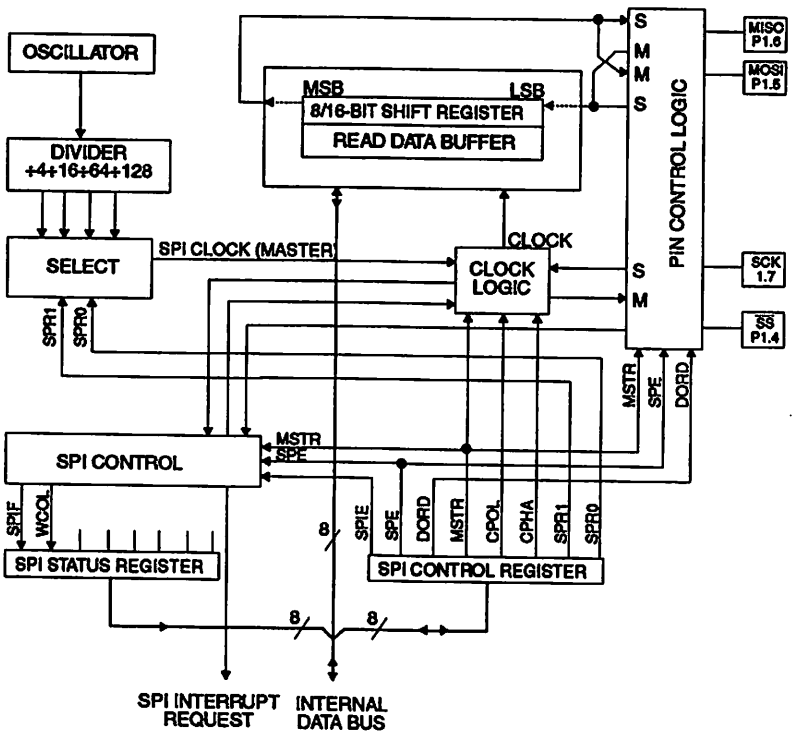


Figure 6. SPI Block Diagram



JART

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

Serial Peripheral nterface

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

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

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

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

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

igure 7. SPI Master-slave Interconnection

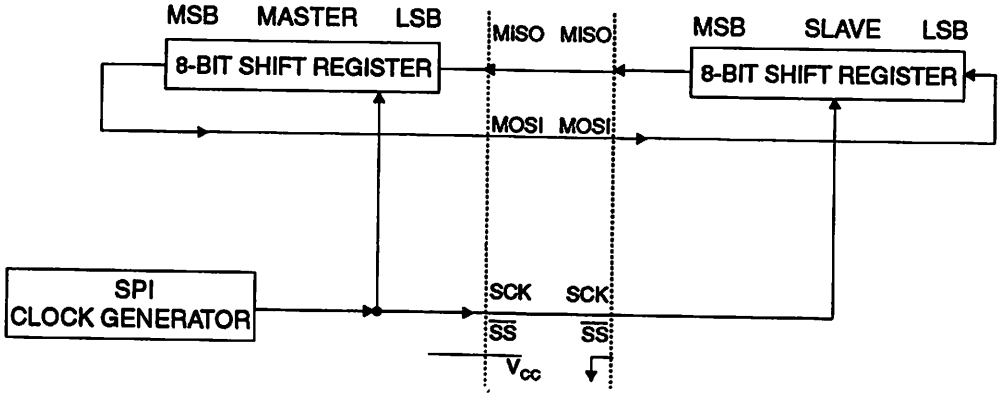
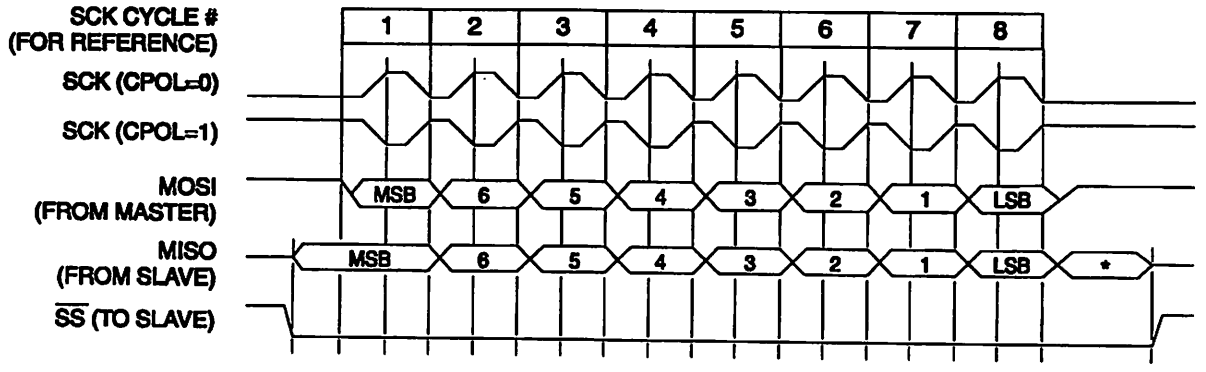
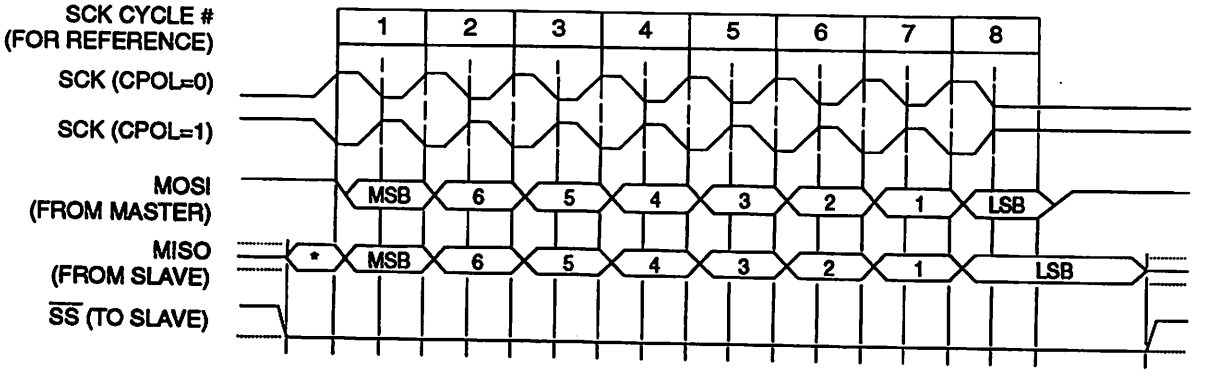


Figure 8. SPI transfer Format with CPHA = 0



Note: *Not defined but normally MSB of character just received

Figure 9. SPI Transfer Format with CPHA = 1



Note: *Not defined but normally LSB of previously transmitted character.

Interrupts

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

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

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

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

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

Table 10. Interrupt Enable (IE) Register

(MSB)(LSB)

EA	–	ET2	ES	ET1	EX1	ET0	EX0

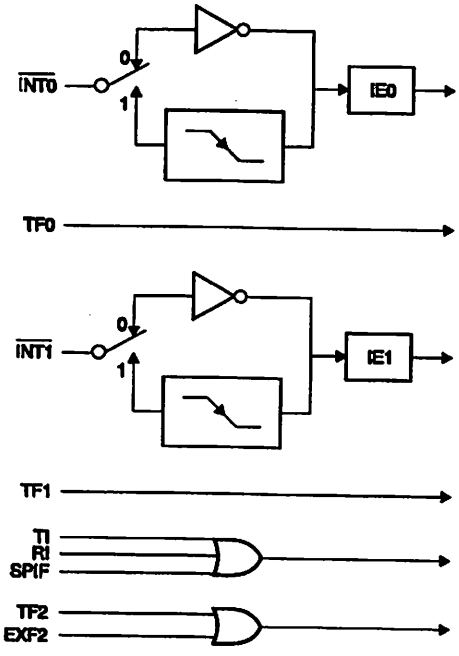
Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

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

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

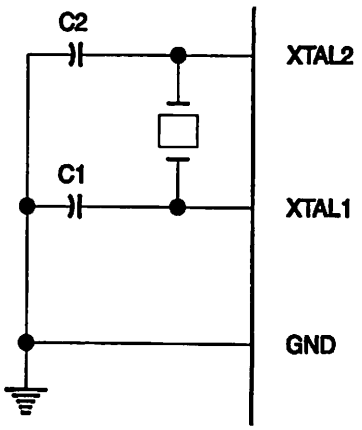
Figure 10. Interrupt Sources



Oscillator Characteristics

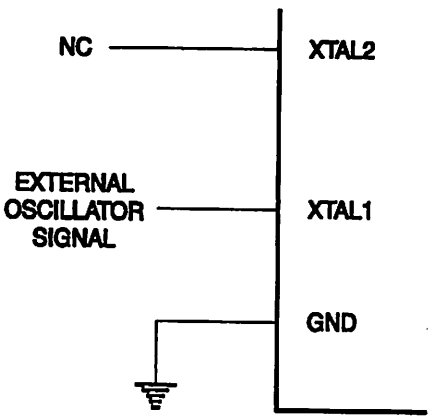
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected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

Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration



Idle Mode

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

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

Status of External Pins During Idle and Power-down Modes

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

Power-down Mode

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

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

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.

Lock Bit Protection Modes[1][2]

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

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

Programming the Flash and EEPROM

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

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

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

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

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

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

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

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

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

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

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

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

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

The AT89S8252 is shipped with the Serial Programming Mode enabled.

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

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

rogramming
ıterface

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

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

ierial Downloading

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

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

The Code and Data memory arrays have separate address spaces:

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

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

ierial Programming lgorithm

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

1. Power-up sequence:

   Apply power between VCC and GND pins.

   Set RST pin to "H".

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

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

   Set RST to "L".

   Turn V_{CC} power off.

Serial Programming Instruction

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

nstruction Set

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

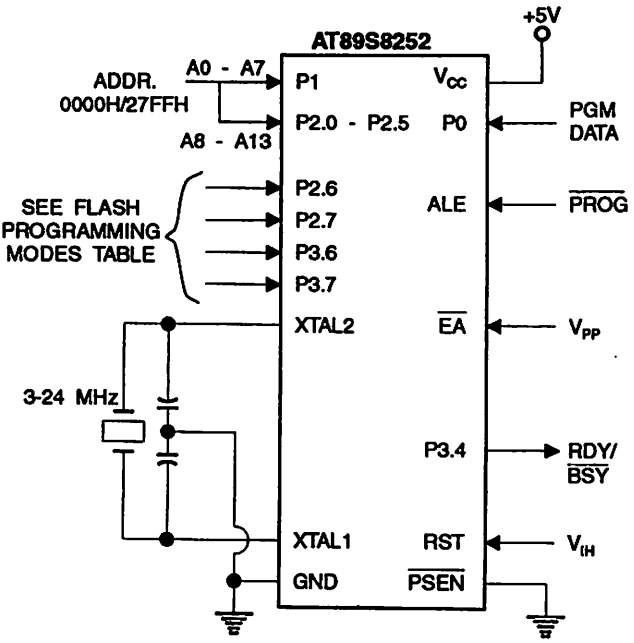
Notes:
1. DATA polling is used to indicate the end of a byte write cycle which typically takes less than 2.5 ms at 5V.
2. "aaaaa" = high order address.
3. "x" = don't care.

Flash and EEPROM Parallel Programming Modes

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

Notes:
1. "h" = weakly pulled "High" internally.
2. Chip Erase and Serial Programming Fuse require a 10 ms $\overline{PROG}$ 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 RDY/$\overline{BSY}$.
4. "X" = don't care

gure 13. Programming the Flash/EEPROM Memory



gure 14. Verifying the Flash/EEPROM Memory

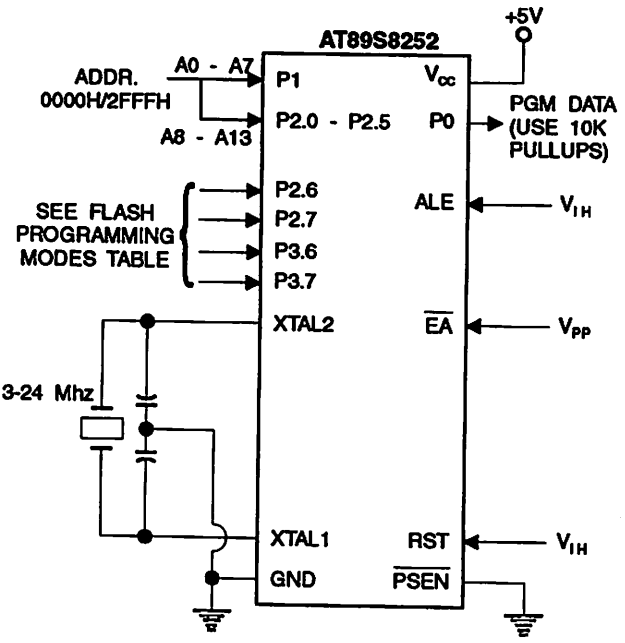
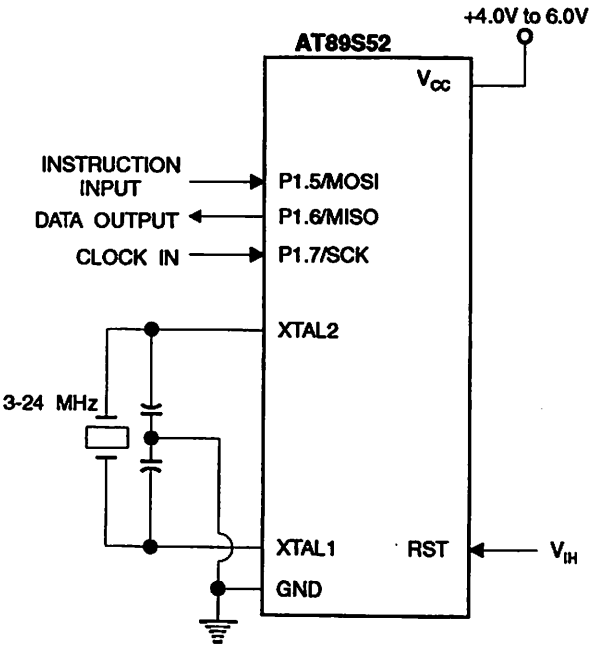


Figure 15. Flash/EEPROM Serial Downloading

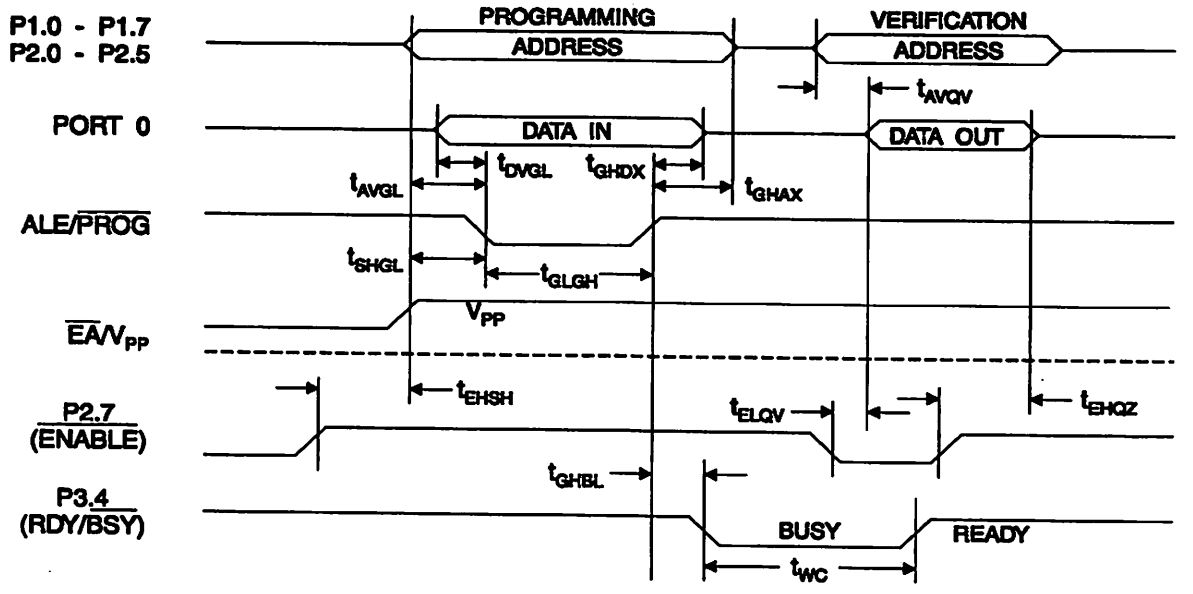


lash Programming and Verification Characteristics – Parallel Mode

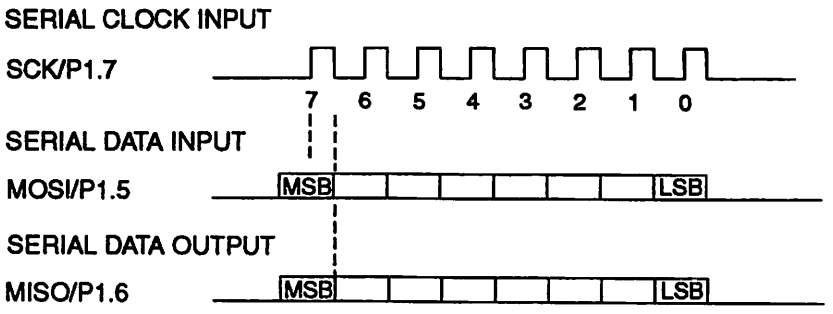
$_A$ = 0°C to 70°C, V_{CC} = 5.0V ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{EHSH}	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	Data Float after $\overline{ENABLE}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
t_{WC}	Byte Write Cycle Time		2.0	ms

lash/EEPROM Programming and Verification Waveforms – Parallel Mode

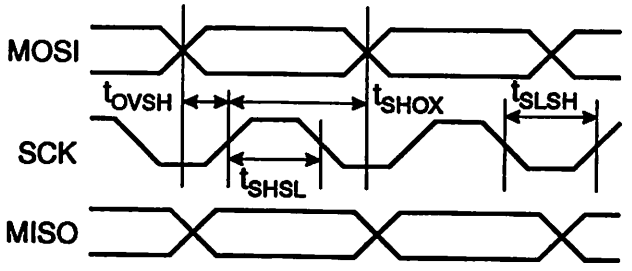


erial Downloading Waveforms



erial Programming Characteristics

gure 16. Serial Programming Timing



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

ymbol	Parameter	Min	Typ	Max	Units
$/t_{CLCL}$	Oscillator Frequency	0		24	MHz
$_{CLCL}$	Oscillator Period	41.6			ns
$_{SHSL}$	SCK Pulse Width High	24 t_{CLCL}			ns
$_{SLSH}$	SCK Pulse Width Low	24 t_{CLCL}			ns
$_{OVSH}$	MOSI Setup to SCK High	t_{CLCL}			ns
$_{SHOX}$	MOSI Hold after SCK High	2 t_{CLCL}			ns

Absolute Maximum Ratings*

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

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

DC Characteristics

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

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC} - 0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC} - 0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC} + 0.5	V
V_{OL}	Output Low-voltage (1) (Ports 1,2,3)	I_{OL} = 1.6 mA		0.5	V
V_{OL1}	Output Low-voltage (1) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.5	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pull-down Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode (2)	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

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

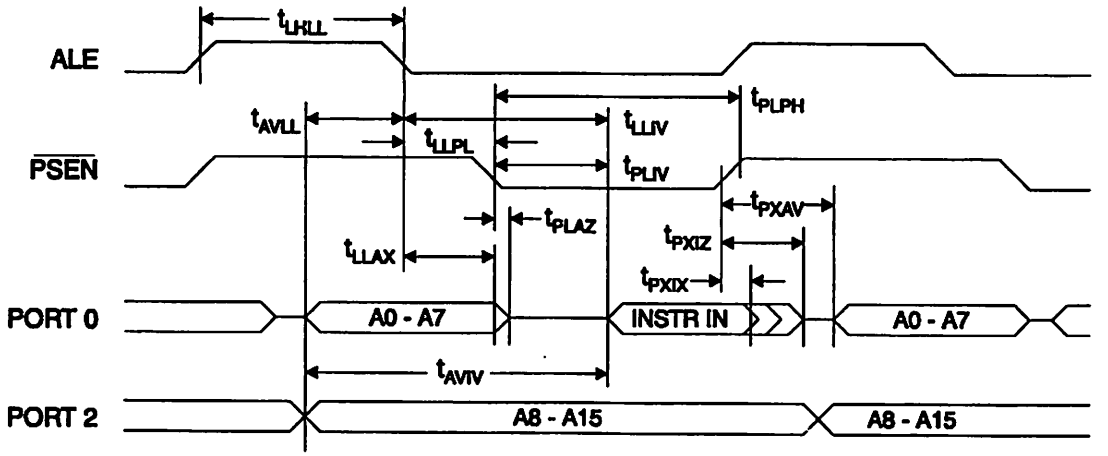
AC Characteristics

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

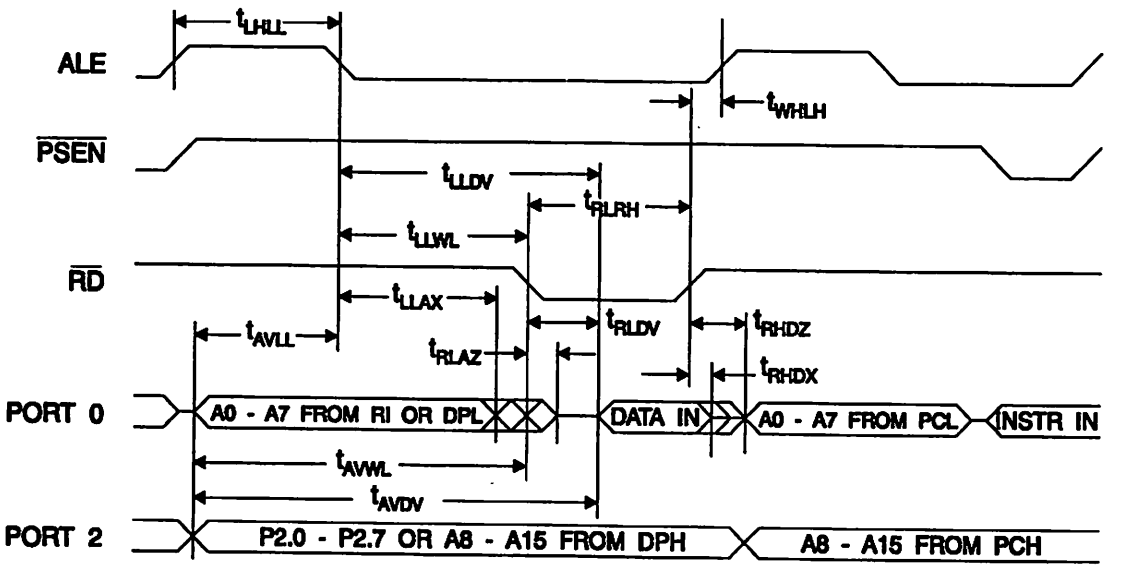
External Program and Data Memory Characteristics

Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{LHLL}	ALE Pulse Width	$2t_{CLCL} - 40$		ns
t_{AVLL}	Address Valid to ALE Low	$t_{CLCL} - 13$		ns
t_{LLAX}	Address Hold after ALE Low	$t_{CLCL} - 20$		ns
t_{LLIV}	ALE Low to Valid Instruction In		$4t_{CLCL} - 65$	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	$t_{CLCL} - 13$		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	$3i_{CLCL} - 20$		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		$3t_{CLCL} - 45$	ns
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		ns
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		$t_{CLCL} - 10$	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	$t_{CLCL} - 8$		ns
t_{AVIV}	Address to Valid Instruction In		$5t_{CLCL} - 55$	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	$6t_{CLCL} - 100$		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	$6t_{CLCL} - 100$		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		$5t_{CLCL} - 90$	ns
t_{RHDX}	Data Hold after $\overline{RD}$	0		ns
t_{RHDZ}	Data Float after $\overline{RD}$		$2t_{CLCL} - 28$	ns
t_{LLDV}	ALE Low to Valid Data In		$8t_{CLCL} - 150$	ns
t_{AVDV}	Address to Valid Data In		$9t_{CLCL} - 165$	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	$3t_{CLCL} - 50$	$3t_{CLCL} + 50$	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	$4t_{CLCL} - 75$		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	$t_{CLCL} - 20$		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	$7t_{CLCL} - 120$		ns
t_{WHQX}	Data Hold after $\overline{WR}$	$t_{CLCL} - 20$		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	$t_{CLCL} - 20$	$t_{CLCL} + 25$	ns

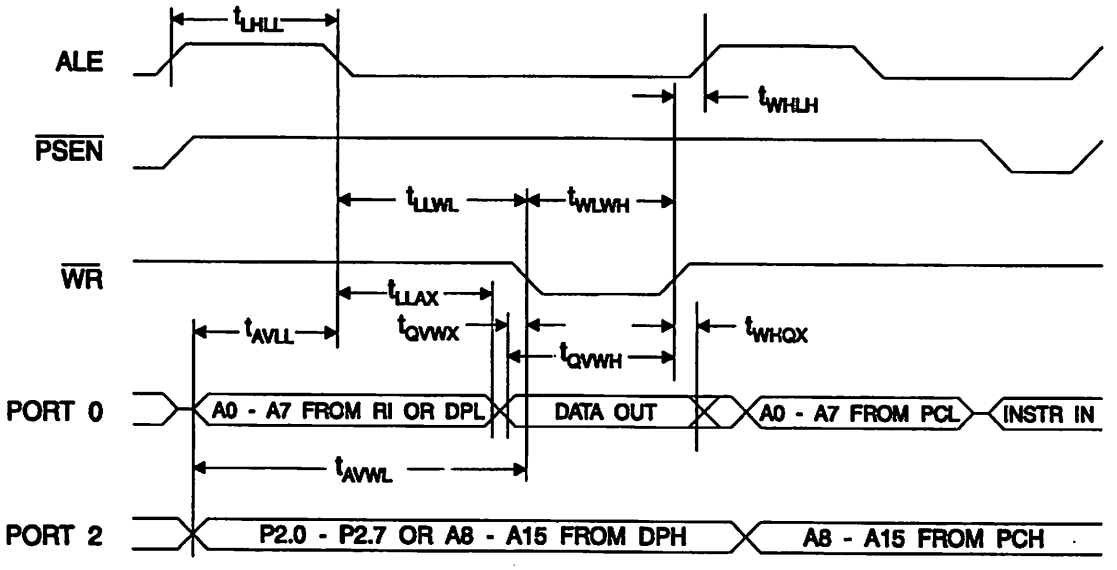
External Program Memory Read Cycle



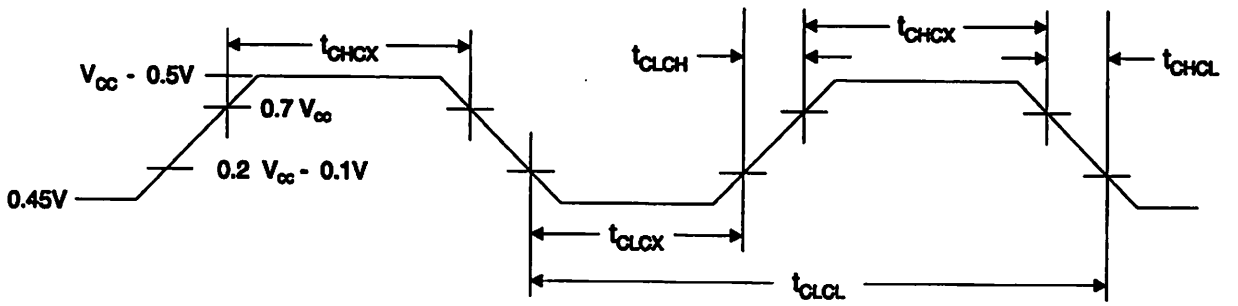
External Data Memory Read Cycle



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

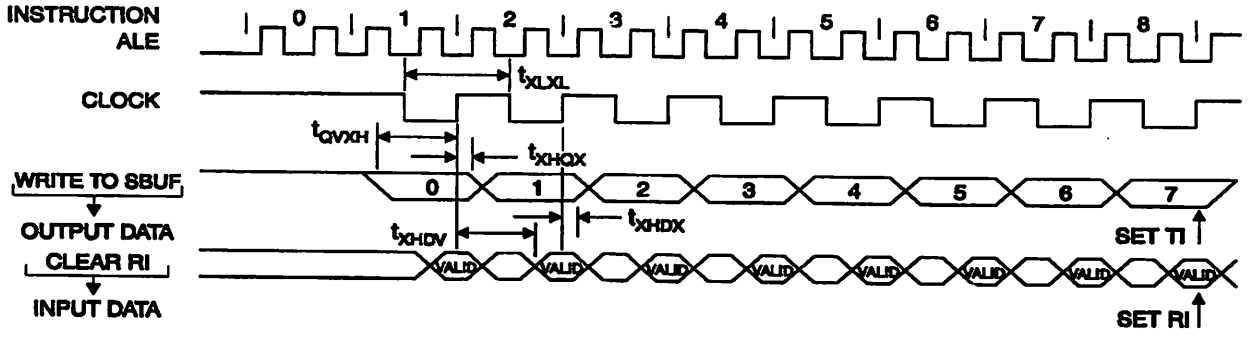
Symbol	Parameter	V_{CC} = 4.0V to 6.0V		Units
		Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{CLCL}	Clock Period	41.6		ns
t_{CHCX}	High Time	15		ns
t_{CLCX}	Low Time	15		ns
t_{CLCH}	Rise Time		20	ns
t_{CHCL}	Fall Time		20	ns

Serial Port Timing: Shift Register Mode Test Conditions

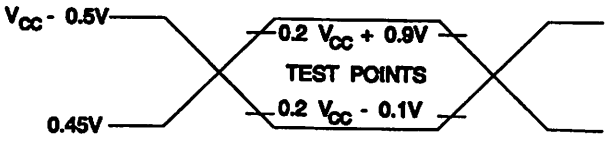
The values in this table are valid for V_{CC} = 4.0V to 6V and Load Capacitance = 80 pF.

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

Shift Register Mode Timing Waveforms

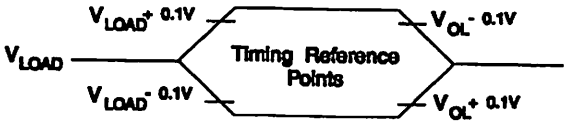


AC Testing Input/Output Waveforms(1)



Note: 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.

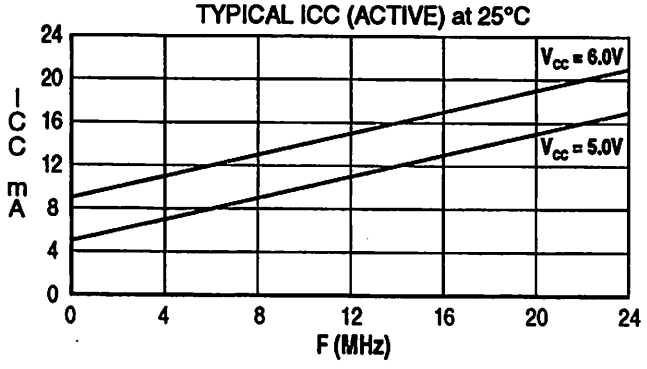
Float Waveforms(1)



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

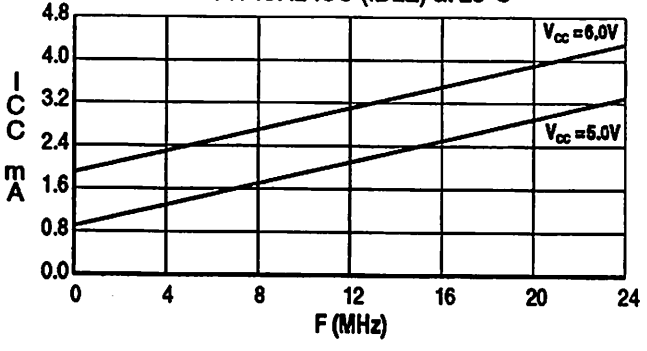
AT89S8252

TYPICAL ICC (ACTIVE) at 25°C



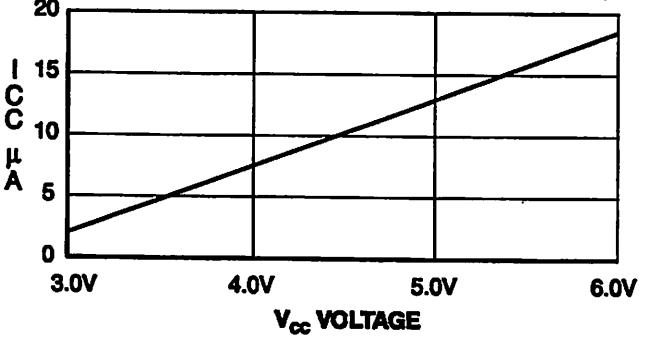
AT89S8252

TYPICAL ICC (IDLE) at 25°C



AT89S8252

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



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

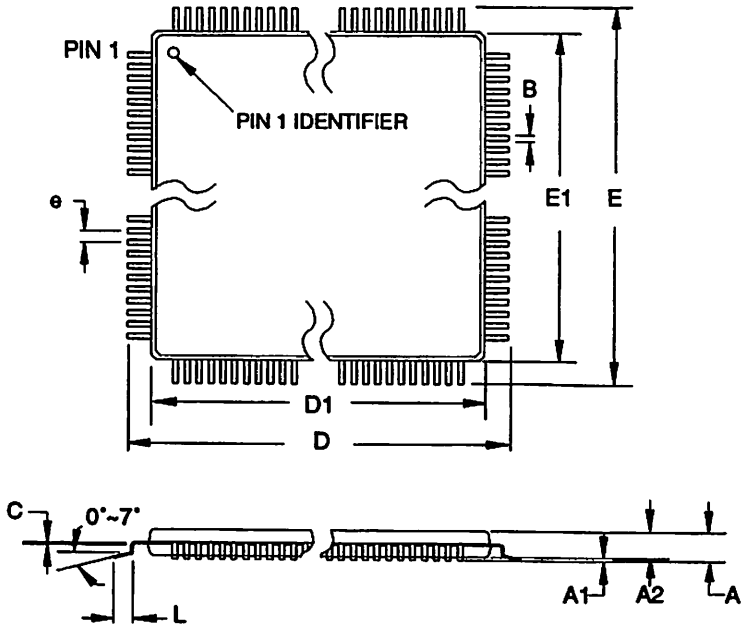
Ordering Information

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

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

Packaging Information

44A – TQFP



COMMON DIMENSIONS
(Unit of Measure = mm)

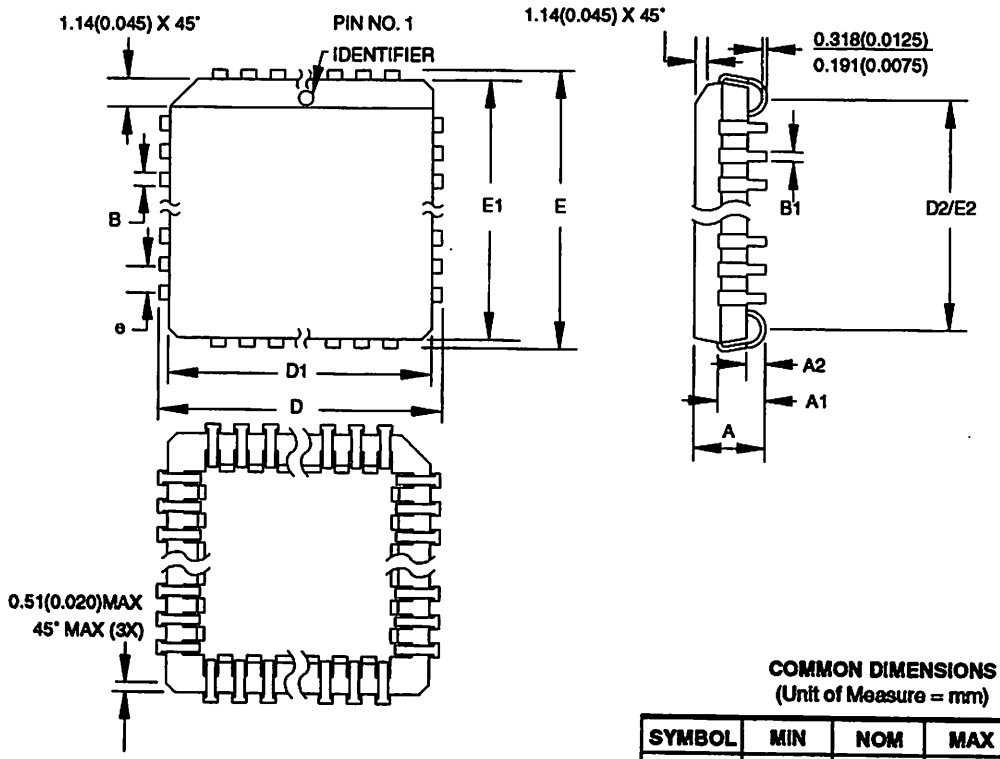
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

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

10/5/2001

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

4J – PLCC



COMMON DIMENSIONS
(Unit of Measure = mm)

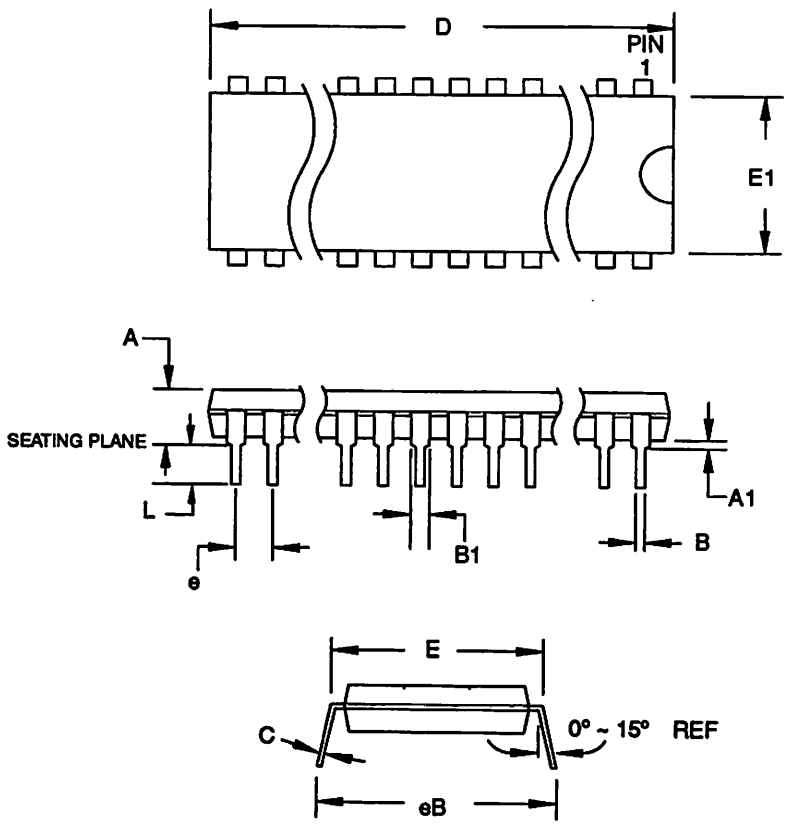
SYMBOL	MIN	NOM	MAX	NOTE
A	4.191	–	4.572	
A1	2.286	–	3.048	
A2	0.508	–	–	
D	17.399	–	17.653	
D1	16.510	–	16.662	Note 2
E	17.399	–	17.653	
E1	16.510	–	16.662	Note 2
D2/E2	14.986	–	16.002	
B	0.660	–	0.813	
B1	0.330	–	0.533	
e	1.270 TYP			

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

10/04/01

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

I0P6 – PDIP



COMMON DIMENSIONS
(Unit of Measure = mm)

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

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

09/28/01

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

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

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

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

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

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

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

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

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

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

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

Literature Requests
www.atmel.com/literature

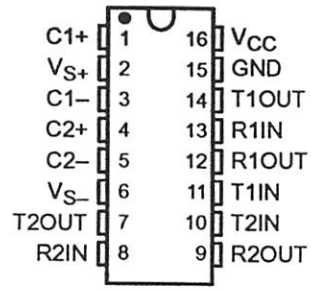
0401G–MICRO–3/06 xM

MAX232, MAX232I
DUAL EIA-232 DRIVER/RECEIVER

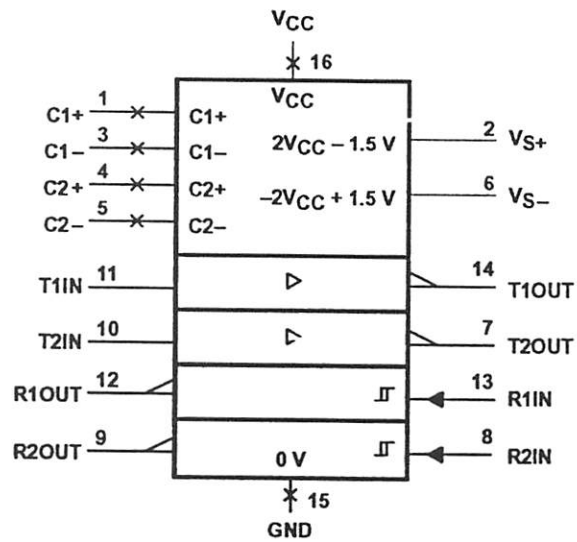
SLLS047G – FEBRUARY 1989 – REVISED AUGUST 1998

- Operates With Single 5-V Power Supply
- LinBiCMOS™ Process Technology
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Designed to be Interchangeable With Maxim MAX232
- Applications
  - TIA/EIA-232-F
  - Battery-Powered Systems
  - Terminals
  - Modems
  - Computers
- ESD Protection Exceeds 2000 V Per MIL-STD-883, Method 3015
- Package Options Include Plastic Small-Outline (D, DW) Packages and Standard Plastic (N) DIPs

D, DW, OR N PACKAGE
(TOP VIEW)



logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

description

The MAX232 device is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

The MAX232 is characterized for operation from 0°C to 70°C. The MAX232I is characterized for operation from –40°C to 85°C.

AVAILABLE OPTIONS

T_A	PACKAGED DEVICES		
	SMALL OUTLINE (D)	SMALL OUTLINE (DW)	PLASTIC DIP (N)
0°C to 70°C	MAX232D‡	MAX232DW‡	MAX232N
–40°C to 85°C	MAX232ID‡	MAX232IDW‡	MAX232IN

‡ This device is available taped and reeled by adding an R to the part number (i.e., MAX232DR).

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

LinASIC and LinBiCMOS are trademarks of Texas Instruments Incorporated.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1) ..... −0.3 V to 6 V
Positive output supply voltage range, V_{S+} ..... V_{CC} −0.3 V to 15 V
Negative output supply voltage range, V_{S-} ..... −0.3 V to −15 V
Input voltage range, V_I: Driver ..... −0.3 V to V_{CC} + 0.3 V
Receiver ..... ±30 V
Output voltage range, V_O: T1OUT, T2OUT ..... V_{S-} −0.3 V to V_{S+} + 0.3 V
R1OUT, R2OUT ..... −0.3 V to V_{CC} + 0.3 V
Short-circuit duration: T1OUT, T2OUT ..... Unlimited
Package thermal impedance, θ_{JA} (see Note 2): D package ..... 113°C/W
DW package ..... 105°C/W
N package ..... 78°C/W
Storage temperature range, T_{stg} ..... −65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds ..... 260°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to network ground terminal.

2. The package thermal impedance is calculated in accordance with JESD 51, except for through-hole packages, which use a trace length of zero.

recommended operating conditions

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.5	5	5.5	V
High-level input voltage, V_{IH} (T1IN,T2IN)		2			V
Low-level input voltage, V_{IL} (T1IN, T2IN)				0.8	V
Receiver input voltage, R1IN, R2IN				±30	V
Operating free-air temperature, T_A	MAX232	0		70	°C
	MAX232I	−40		85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (unless otherwise noted)

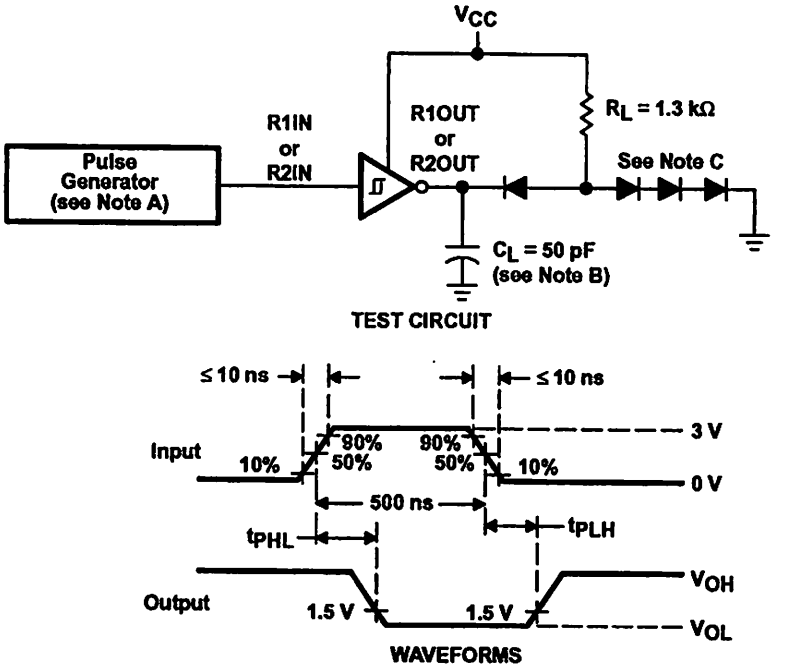
PARAMETER			TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	T1OUT, T2OUT	R_L = 3 kΩ to GND		5	7		V
		R1OUT, R2OUT	I_{OH} = –1 mA		3.5			
V_{OL}	Low-level output voltage‡	T1OUT, T2OUT	R_L = 3 kΩ to GND			–7	–5	V
		R1OUT, R2OUT	I_{OL} = 3.2 mA				0.4	
V_{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C		1.7	2.4	V
V_{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C	0.8	1.2		V
V_{hys}	Input hysteresis voltage	R1IN, R2IN	V_{CC} = 5 V		0.2	0.5	1	V
r_i	Receiver input resistance	R1IN, R2IN	V_{CC} = 5,	T_A = 25°C	3	5	7	kΩ
r_o	Output resistance	T1OUT, T2OUT	V_{S+} = V_{S-} = 0,	V_O = ±2 V	300			Ω
I_{OS}§	Short-circuit output current	T1OUT, T2OUT	V_{CC} = 5.5 V,	V_O = 0		±10		mA
I_{IS}	Short-circuit input current	T1IN, T2IN	V_I = 0				200	µA
I_{CC}	Supply current		V_{CC} = 5.5 V, T_A = 25°C	All outputs open,		8	10	mA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
§ Not more than one output should be shorted at a time.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{PLH(R)}$	Receiver propagation delay time, low- to high-level output	See Figure 1		500		ns
$t_{PHL(R)}$	Receiver propagation delay time, high- to low-level output	See Figure 1		500		ns
SR	Driver slew rate	R_L = 3 kΩ to 7 kΩ, See Figure 2			30	V/µs
SR(tr)	Driver transition region slew rate	See Figure 3		3		V/µs

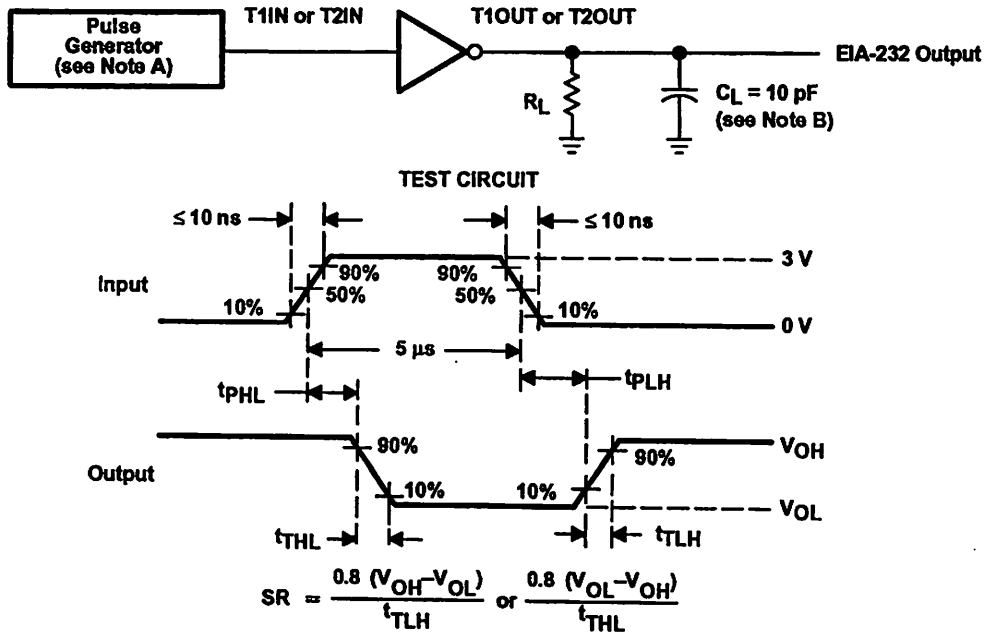
PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.
B. C_L includes probe and jig capacitance.
C. All diodes are 1N3064 or equivalent.

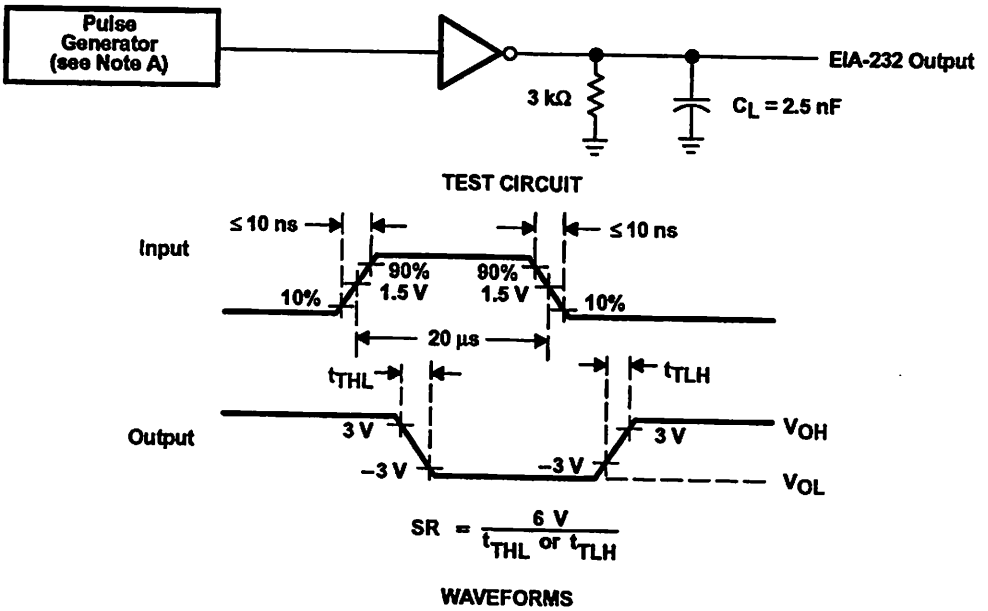
Figure 1. Receiver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements

PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.
B. C_L includes probe and jig capacitance.

Figure 2. Driver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements (5-µs Input)



NOTE A: The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.

Figure 3. Test Circuit and Waveforms for t_{THL} and t_{TLH} Measurements (20-µs Input)

APPLICATION INFORMATION

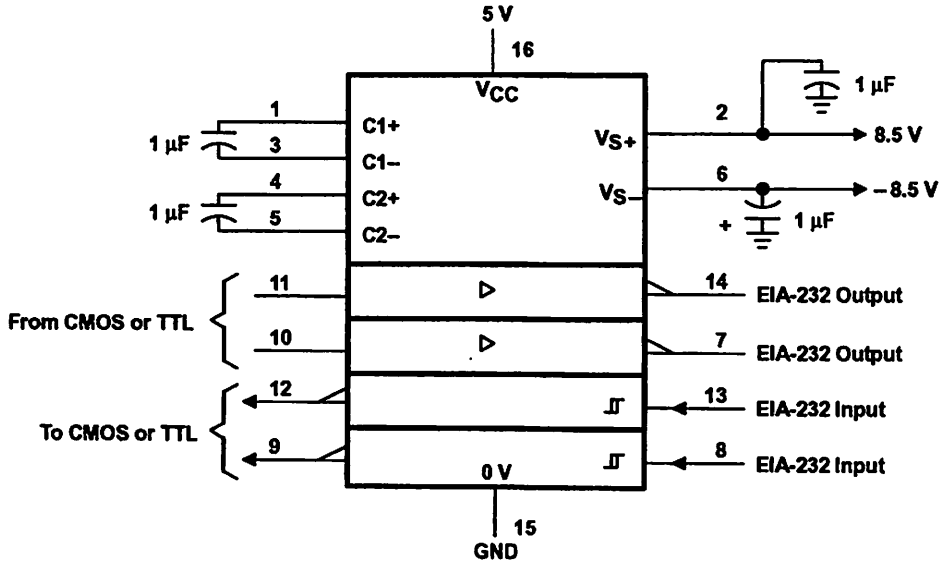


Figure 4. Typical Operating Circuit

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

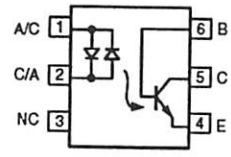
In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Optocoupler, Phototransistor Output, AC input, With Base Connection

Features

- AC or Polarity Insensitive Input
- Built-in Reverse Polarity Input Protection
- I/O Compatible with Integrated Circuits
- Industry Standard DIP Package
- Isolation Test Voltage 5300 V_{RMS}



n7901o

Agency Approvals

- UL File #E52744 System Code H or J
- CSA 93751
- BSI IEC60950 IEC60965
- DIN EN 60747-5-2(VDE0884)
  DIN EN 60747-5-5 pending
  Available with Option 1
- FIMKO

Applications

Telephone line detection
AC line motor
PLC
Instrumentation

Description

The H11AA1 is a bi-directional input optically coupled isolator consisting of two inverse parallel Gallium Arsenide infrared LEDs coupled to a silicon NPN phototransistor in a 6-pin DIP package. The H11AA1 has a minimum CTR of 20 %, a CTR symmetry of 1:3 and is designed for applications requiring detection or monitoring of AC signals.

Order Information

Part	Remarks
H11AA1	CTR > 20 %, DIP-6
H11AA1-X006	CTR > 20 %, DIP-6 400 mil (option 6)
H11AA1-X007	CTR > 20 %, SMD-6 (option 7)
H11AA1-X009	CTR > 20 %, SMD-6 (option 9)

For additional information on the available options refer to Option Information.

Absolute Maximum Ratings

T_{amb} = 25 °C, unless otherwise specified
Stresses in excess of the absolute Maximum Ratings can cause permanent damage to the device. Functional operation of the device is not implied at these or any other conditions in excess of those given in the operational sections of this document. Exposure to absolute Maximum Rating for extended periods of the time can adversely affect reliability.

Input

Parameter	Test condition	Symbol	Value	Unit
Forward continuous current		I_F	60	mA
Power dissipation		P_{diss}	100	mW
Derate linearly from 25 °C			1.3	mW/°C

Output

Parameter	Test condition	Symbol	Value	Unit
Power dissipation		P_{diss}	200	mW
Derate linearly from 25 °C			2.6	mW/°C
Collector-emitter breakdown voltage		BV_{CEO}	30	V
Emitter-base breakdown voltage		BV_{EBO}	5.0	V
Collector-base breakdown voltage		BV_{CBO}	70	V

Coupler

Parameter	Test condition	Symbol	Value	Unit
Isolation test voltage (between emitter and detector)	Referred to standard climate 23 °C/50%RH, DIN 50014	V_{ISO}	5300	V_{RMS}
Creepage			≥ 7.0	mm
Clearance			≥ 7.0	mm
Comparative tracking index per DIN IEC 112/VDE 0303, part 1			175	
Isolation resistance	V_{IO} = 500 V, T_{amb} = 25 °C	R_{IO}	≥ 10^{12}	Ω
	V_{IO} = 500 V, T_{amb} = 100 °C	R_{IO}	≥ 10^{11}	Ω
Storage temperature		T_{stg}	- 55 to + 150	°C
Operating temperature		T_{amb}	- 55 to + 100	°C
Lead soldering time at 260 °C		T_{sld}	10	sec.

Electrical Characteristics

T_{amb} = 25 °C, unless otherwise specified

Minimum and maximum values are testing requirements. Typical values are characteristics of the device and are the result of engineering evaluation. Typical values are for information only and are not part of the testing requirements.

Input

Parameter	Test condition	Symbol	Min	Typ.	Max	Unit
Forward voltage	I_F = ± 10 mA	V_F		1.2	1.5	V

Output

Parameter	Test condition	Symbol	Min	Typ.	Max	Unit
Collector-emitter breakdown voltage	I_C = 1.0 mA	BV_{CEO}	30			V
Emitter-base breakdown voltage	I_E = 100 µA	BV_{EBO}	5.0			V
Collector-base breakdown voltage	I_C = 100 µA	BV_{CBO}	70			V
Collector-emitter leakage current	V_{CE} = 10 V	I_{CEO}		5.0	100	nA

Coupler

Parameter	Test condition	Symbol	Min	Typ.	Max	Unit
Collector-emitter saturation voltage	I_F = ± 10 mA, I_C = 0.5 mA	V_{CEsat}			0.4	V

Current Transfer Ratio

Parameter	Test condition	Symbol	Min	Typ.	Max	Unit
DC Current Transfer Ratio	I_F = ± 10 mA, V_{CE} = 10 V	CTR_{DC}	20			%
Symmetry (CTR at + 10 mA) /(CTR at - 10 mA)			0.33	1.0	3.0	

Typical Characteristics (T_{amb} = 25 °C unless otherwise specified)

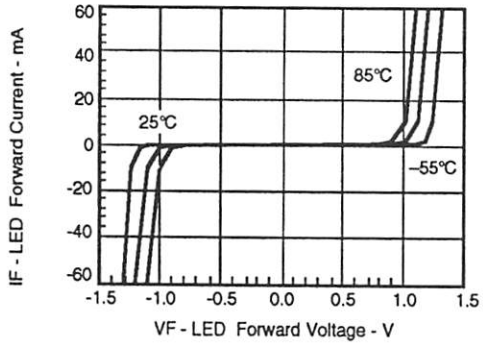


Fig. 1 LED Forward Current vs.Forward Voltage

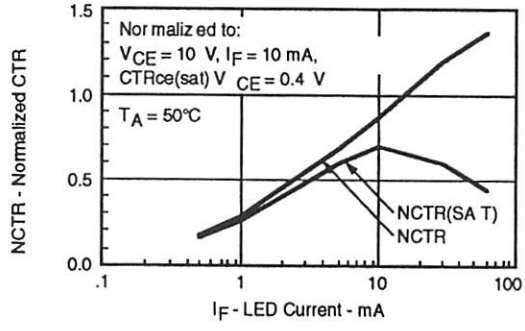


Fig. 3 Normalized Non-saturated and Saturated CTR vs. LED Current

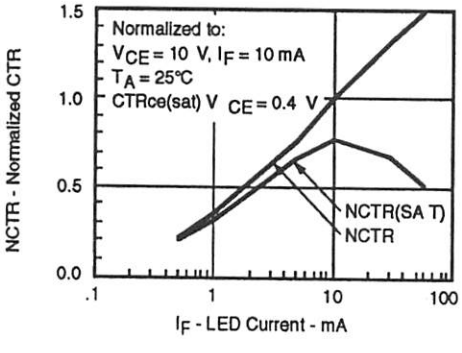


Fig. 2 Normalized Non-Saturated and Saturated CTR vs. LED Current

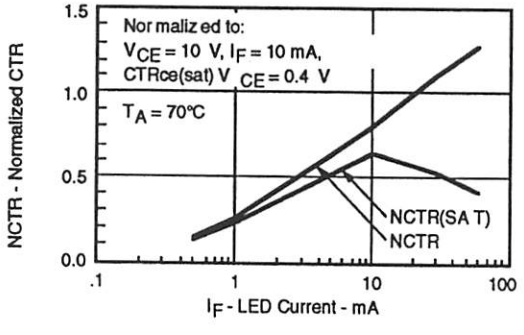


Fig. 4 Normalized Non-saturated and saturated CTR vs. LED Current

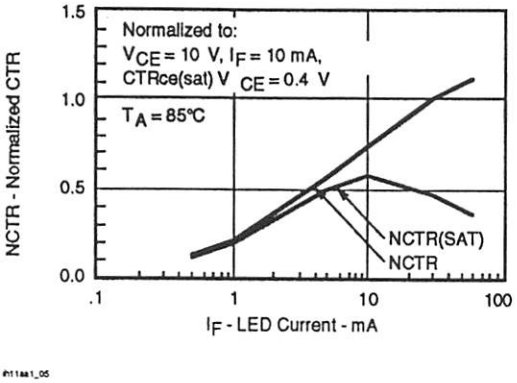


Fig. 5 Normalized Non-saturated and saturated CTR vs. LED Current

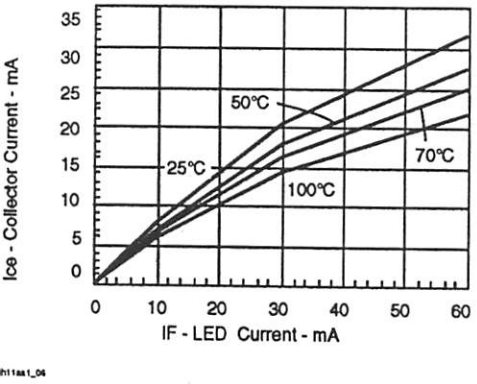


Fig. 6 Collector-Emitter Current vs. Temperature and LED Current

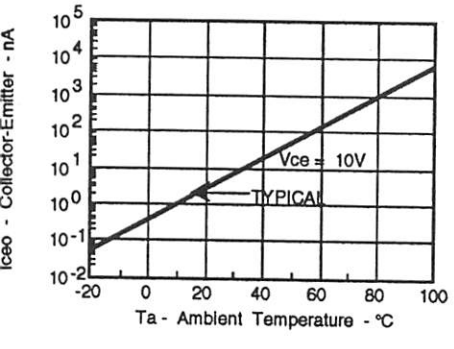


Fig. 7 Collector-Emitter Leakage Current vs.Temp.

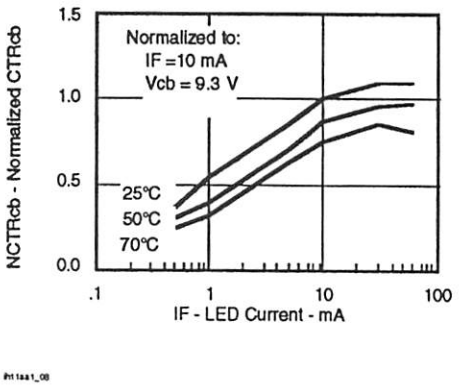


Fig. 8 Normalized CTRcb vs. LED Current and Temp.

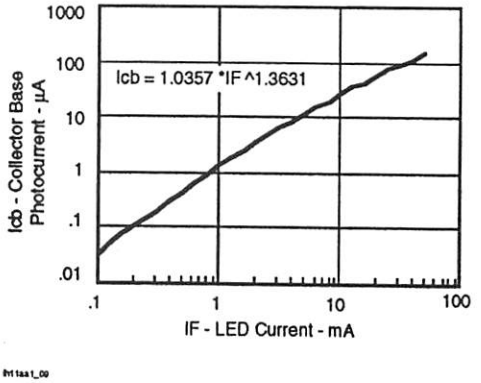


Fig. 9 Collector-Base Photocurrent vs. LED Current

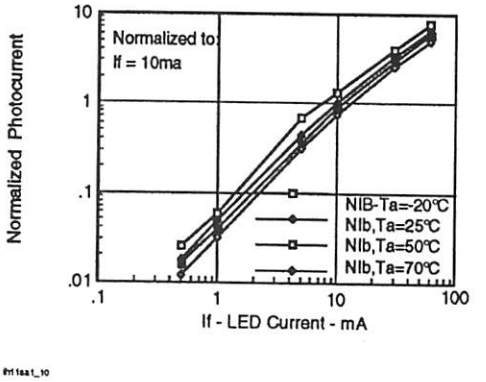


Fig. 10 Normalized Photocurrent vs. LED Current

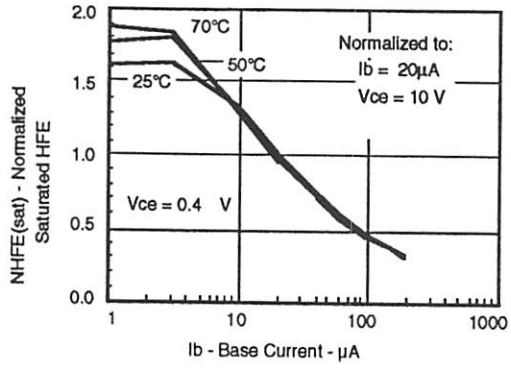


Fig. 11 Normalized Saturated HFE vs. Base Current and Temperature

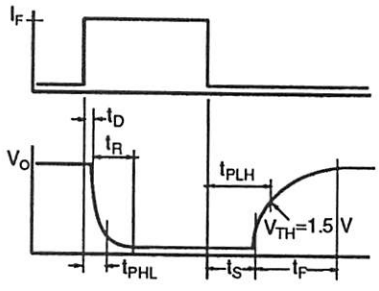


Fig. 14 Switching Waveform

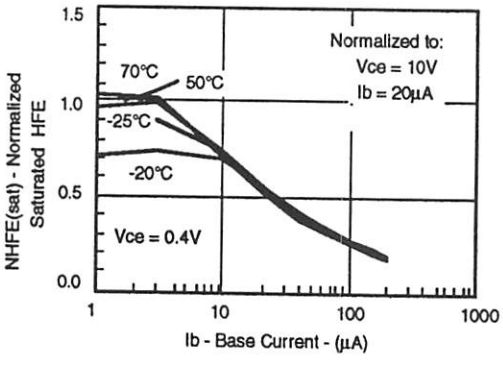


Fig. 12 Normalized Saturated HFE vs. Base Current and Temperature

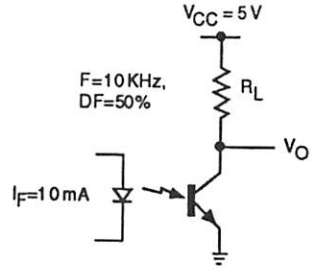


Fig. 15 Switching Schematic

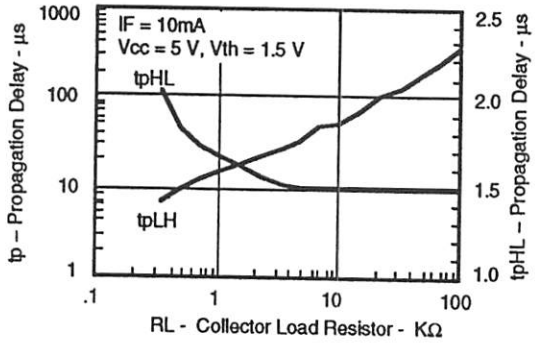
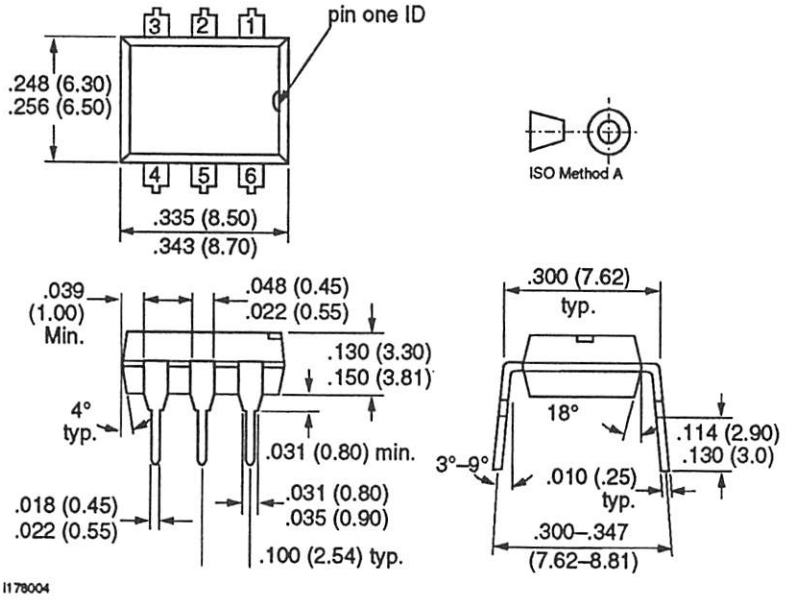
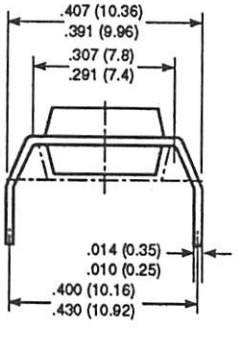


Fig. 13 Propagation Delay vs. Collector Load Resistor

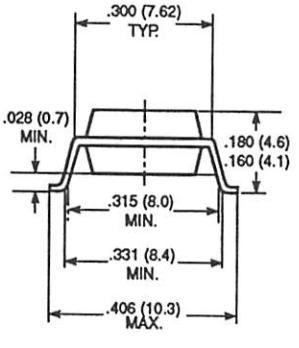
Package Dimensions in Inches (mm)



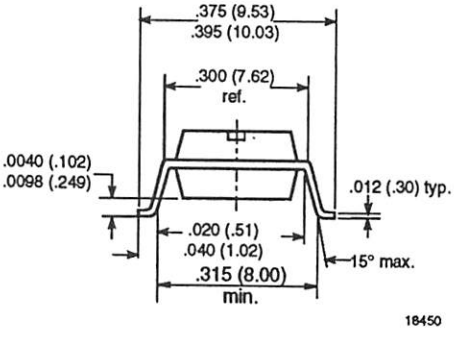
Option 6



Option 7



Option 9



Ozone Depleting Substances Policy Statement

It is the policy of **Vishay Semiconductor GmbH** to

1. Meet all present and future national and international statutory requirements.
2. Regularly and continuously improve the performance of our products, processes, distribution and operatingsystems with respect to their impact on the health and safety of our employees and the public, as well as their impact on the environment.

It is particular concern to control or eliminate releases of those substances into the atmosphere which are known as ozone depleting substances (ODSs).

The Montreal Protocol (1987) and its London Amendments (1990) intend to severely restrict the use of ODSs and forbid their use within the next ten years. Various national and international initiatives are pressing for an earlier ban on these substances.

Vishay Semiconductor GmbH has been able to use its policy of continuous improvements to eliminate the use of ODSs listed in the following documents..

1. Annex A, B and list of transitional substances of the Montreal Protocol and the London Amendments respectively
2. Class I and II ozone depleting substances in the Clean Air Act Amendments of 1990 by the Environmental Protection Agency (EPA) in the USA
3. Council Decision 88/540/EEC and 91/690/EEC Annex A, B and C (transitional substances) respectively.

Vishay Semiconductor GmbH can certify that our semiconductors are not manufactured with ozone depleting substances and do not contain such substances.

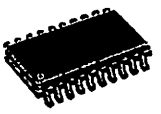
Vishay Semiconductor GmbH, P.O.B. 3535, D-74025 Heilbronn, Germany
Telephone: 49 (0)7131 67 2831, Fax number: 49 (0)7131 67 2423

SGS-THOMSON MICROELECTRONICS

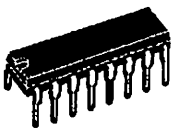
L293D
L293DD

PUSH-PULL FOUR CHANNEL DRIVER WITH DIODES

- 600mA OUTPUT CURRENT CAPABILITY PER CHANNEL
- 1.2A PEAK OUTPUT CURRENT (non repetitive) PER CHANNEL
- ENABLE FACILITY
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)
- INTERNAL CLAMP DIODES



SO(12+4+4)



Powerdip (12+2+2)

ORDERING NUMBERS:

L293DD

L293D

DESCRIPTION

The Device is a monolithic integrated high voltage, high current four channel driver designed to accept standard DTL or TTL logic levels and drive inductive loads (such as relays solenoides, DC and stepping motors) and switching power transistors.

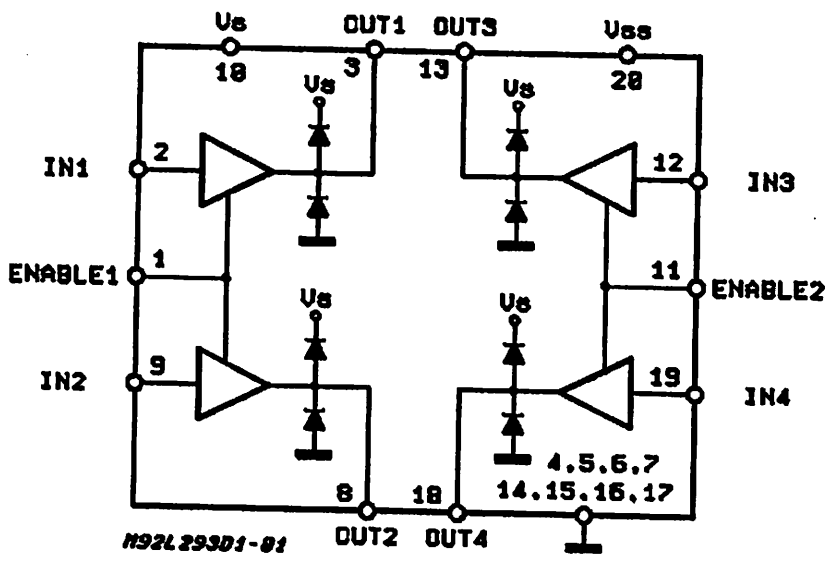
To simplify use as two bridges each pair of channels is equipped with an enable input. A separate supply input is provided for the logic, allowing operation at a lower voltage and internal clamp diodes are included.

This device is suitable for use in switching applications at frequencies up to 5 kHz.

The L293D is assembled in a 16 lead plastic packaage which has 4 center pins connected together and used for heatsinking

The L293DD is assembled in a 20 lead surface mount which has 8 center pins connected together and used for heatsinking.

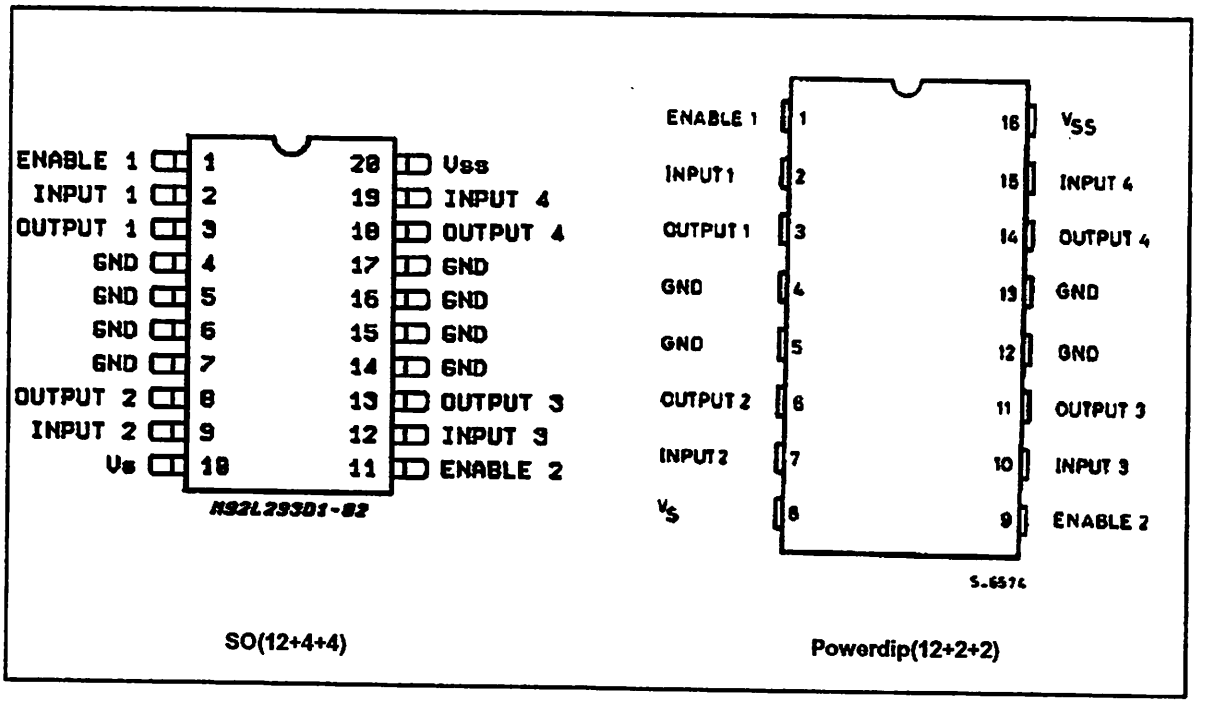
BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Supply Voltage	36	V
V_{SS}	Logic Supply Voltage	36	V
V_i	Input Voltage	7	V
V_{en}	Enable Voltage	7	V
I_o	Peak Output Current (100 µs non repetitive)	1.2	A
P_{tot}	Total Power Dissipation at T_{pins} = 90 °C	4	W
T_{stg}, T_j	Storage and Junction Temperature	– 40 to 150	°C

PIN CONNECTIONS (Top view)



SO(12+4+4)

Powerdip(12+2+2)

THERMAL DATA

Symbol	Decription		DIP	SO	Unit
$R_{th\ j\text{-}pins}$	Thermal Resistance Junction-pins	max.	–	14	°C/W
$R_{th\ j\text{-}amb}$	Thermal Resistance junction-ambient	max.	80	50 (*)	°C/W
$R_{th\ j\text{-}case}$	Thermal Resistance Junction-case	max.	14	–	

(*) With 6sq. cm on board heatsink.

ELECTRICAL CHARACTERISTICS (for each channel, V_S = 24 V, V_{SS} = 5 V, T_{amb} = 25 °C, unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_S	Supply Voltage (pin 10)		V_{SS}		36	V
V_{SS}	Logic Supply Voltage (pin 20)		4.5		36	V
I_S	Total Quiescent Supply Current (pin 10)	V_i = L ; I_O = 0 ; V_{en} = H		2	6	mA
		V_i = H ; I_O = 0 ; V_{en} = H		16	24	mA
		V_{en} = L			4	mA
I_{SS}	Total Quiescent Logic Supply Current (pin 20)	V_i = L ; I_O = 0 ; V_{en} = H		44	60	mA
		V_i = H ; I_O = 0 ; V_{en} = H		16	22	mA
		V_{en} = L		16	24	mA
V_{IL}	Input Low Voltage (pin 2, 9, 12, 19)		− 0.3		1.5	V
V_{IH}	Input High Voltage (pin 2, 9, 12, 19)	$V_{SS} \leq 7$ V	2.3		V_{SS}	V
		$V_{SS} > 7$ V	2.3		7	V
I_{IL}	Low Voltage input Current (pin 2, 9, 12, 19)	V_{IL} = 1.5 V			− 10	µA
I_{IH}	High Voltage Input Current (pin 2, 9, 12, 19)	2.3 V ≤ V_{IH} ≤ V_{SS} − 0.6 V		30	100	µA
$V_{en\,L}$	Enable Low Voltage (pin 1, 11)		− 0.3		1.5	V
$V_{en\,H}$	Enable High Voltage (pin 1, 11)	$V_{SS} \leq 7$ V	2.3		V_{SS}	V
		$V_{SS} > 7$ V	2.3		7	V
$I_{en\,L}$	Low Voltage Enable Current (pin 1, 11)	$V_{en\,L}$ = 1.5 V		− 30	− 100	µA
$I_{en\,H}$	High Voltage Enable Current (pin 1, 11)	2.3 V ≤ $V_{en\,H}$ ≤ V_{SS} − 0.6 V			± 10	µA
$V_{CE(sat)H}$	Source Output Saturation Voltage (pins 3, 8, 13, 18)	I_O = − 0.6 A		1.4	1.8	V
$V_{CE(sat)L}$	Sink Output Saturation Voltage (pins 3, 8, 13, 18)	I_O = + 0.6 A		1.2	1.8	V
V_F	Clamp Diode Forward Voltage	I_O = 600nA		1.3		V
t_r	Rise Time (*)	0.1 to 0.9 V_O		250		ns
t_f	Fall Time (*)	0.9 to 0.1 V_O		250		ns
t_{on}	Turn-on Delay (*)	0.5 V_i to 0.5 V_O		750		ns
t_{off}	Turn-off Delay (*)	0.5 V_i to 0.5 V_O		200		ns

(*) See fig. 1.

TRUTH TABLE (one channel)

Input	Enable (*)	Output
H	H	H
L	H	L
H	L	Z
L	L	Z

Z = High output impedance
(*) Relative to the considered channel

Figure 1: Switching Times

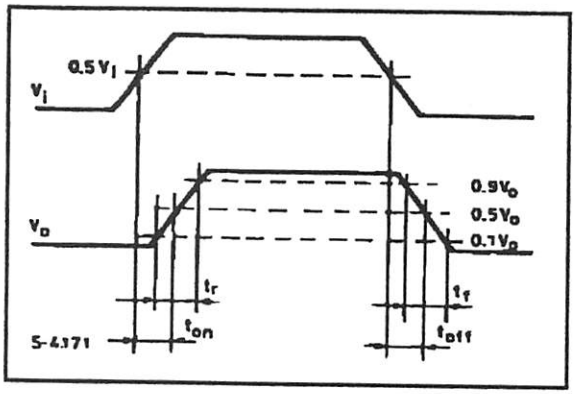
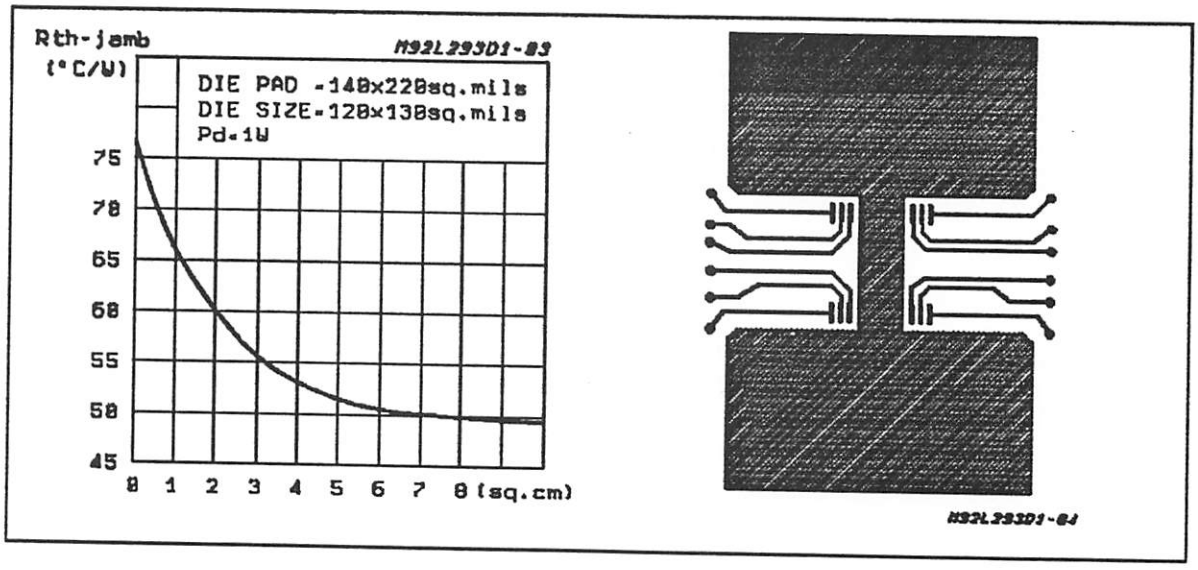
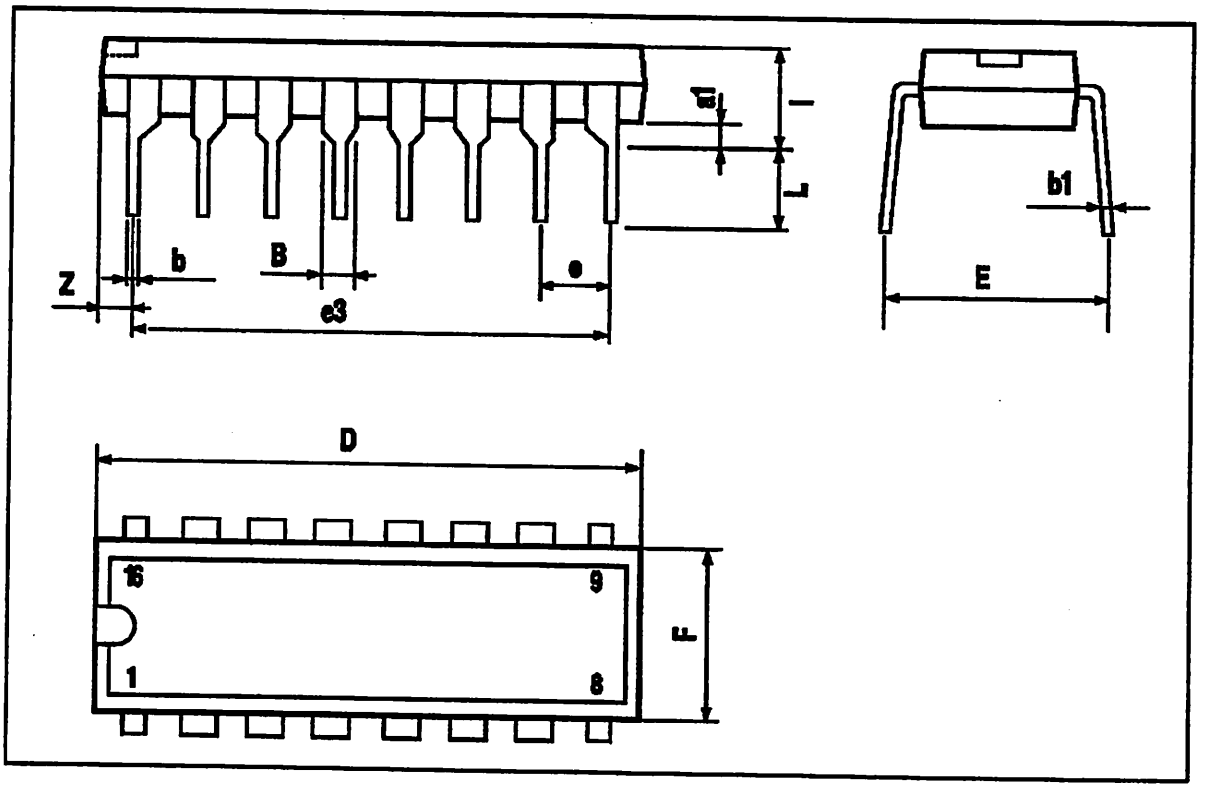


Figure 2: Junction to ambient thermal resistance vs. area on board heatsink (SO12+4+4 package)



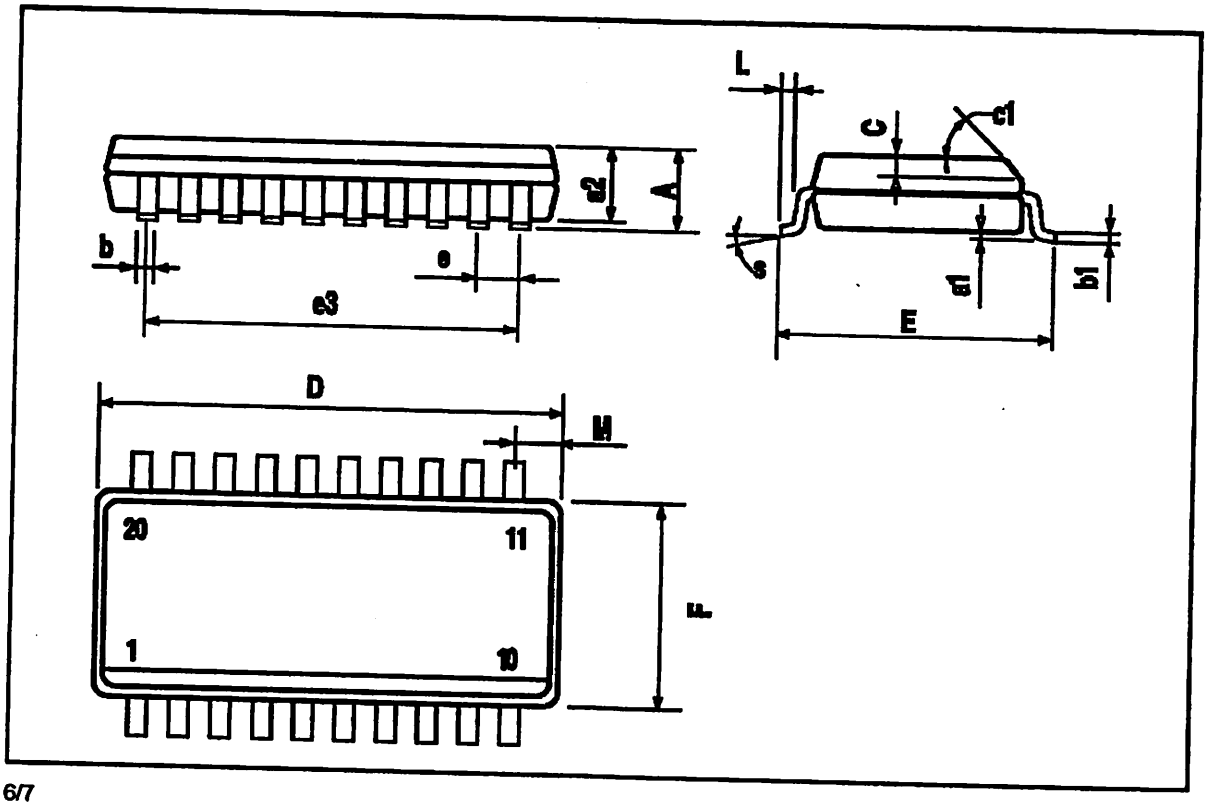
POWERDIP16 PACKAGE MECHANICAL DATA

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
a1	0.51			0.020		
B	0.85		1.40	0.033		0.055
b		0.50			0.020	
b1	0.38		0.50	0.015		0.020
D			20.0			0.787
E		8.80			0.346	
e		2.54			0.100	
e3		17.78			0.700	
F			7.10			0.280
I			5.10			0.201
L		3.30			0.130	
Z			1.27			0.050



SO20 PACKAGE MECHANICAL DATA

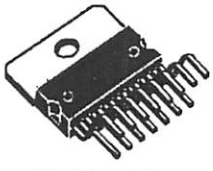
DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			2.65			0.104
a1	0.1		0.2	0.004		0.008
a2			2.45			0.096
b	0.35		0.49	0.014		0.019
b1	0.23		0.32	0.009		0.013
C		0.5			0.020	
c1		45			1.772	
D		1	12.6		0.039	0.496
E	10		10.65	0.394		0.419
e		1.27			0.050	
e3		11.43			0.450	
F		1	7.4		0.039	0.291
G	8.8		9.15	0.346		0.360
L	0.5		1.27	0.020		0.050
M			0.75			0.030
S	8° (max.)					



L298

DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)



Multiwatt15

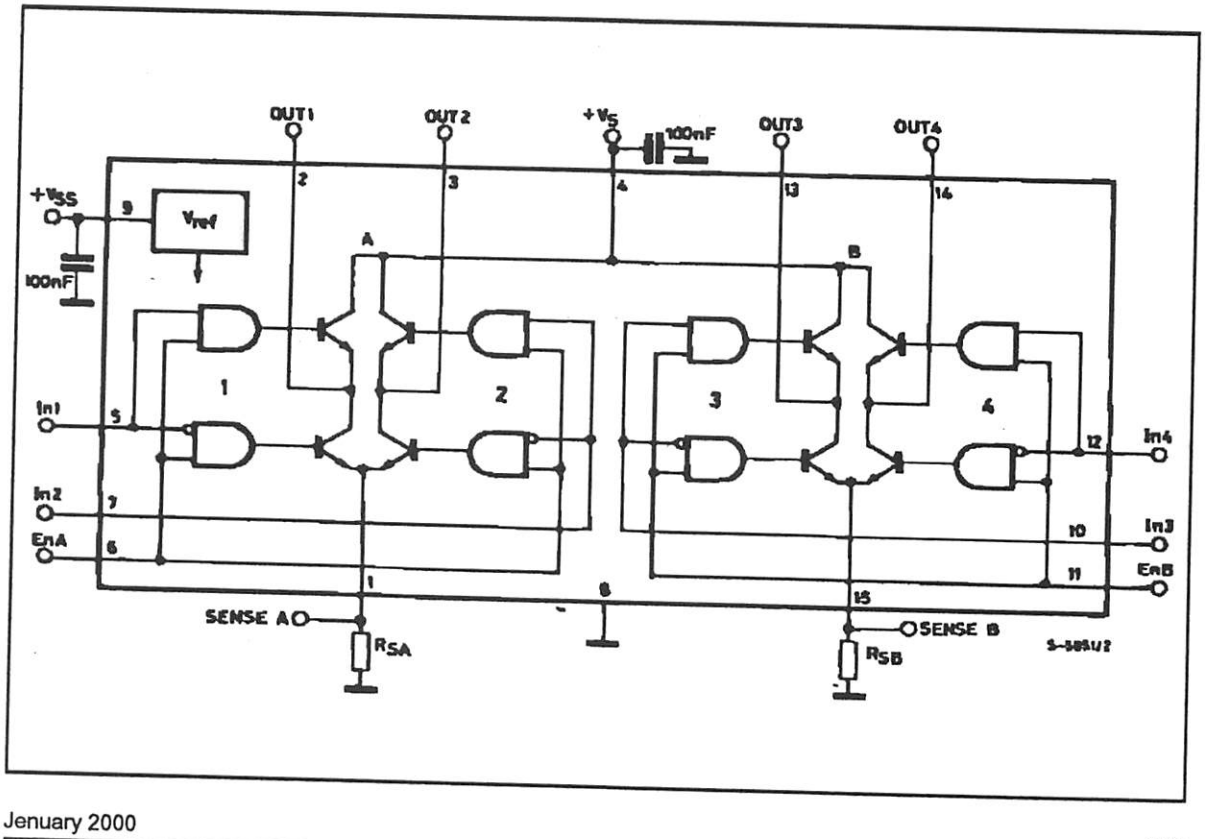
PowerSO20

ORDERING NUMBERS : L298N (Multiwatt Vert.)
L298HN (Multiwatt Horiz.)
L298P (PowerSO20)

DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

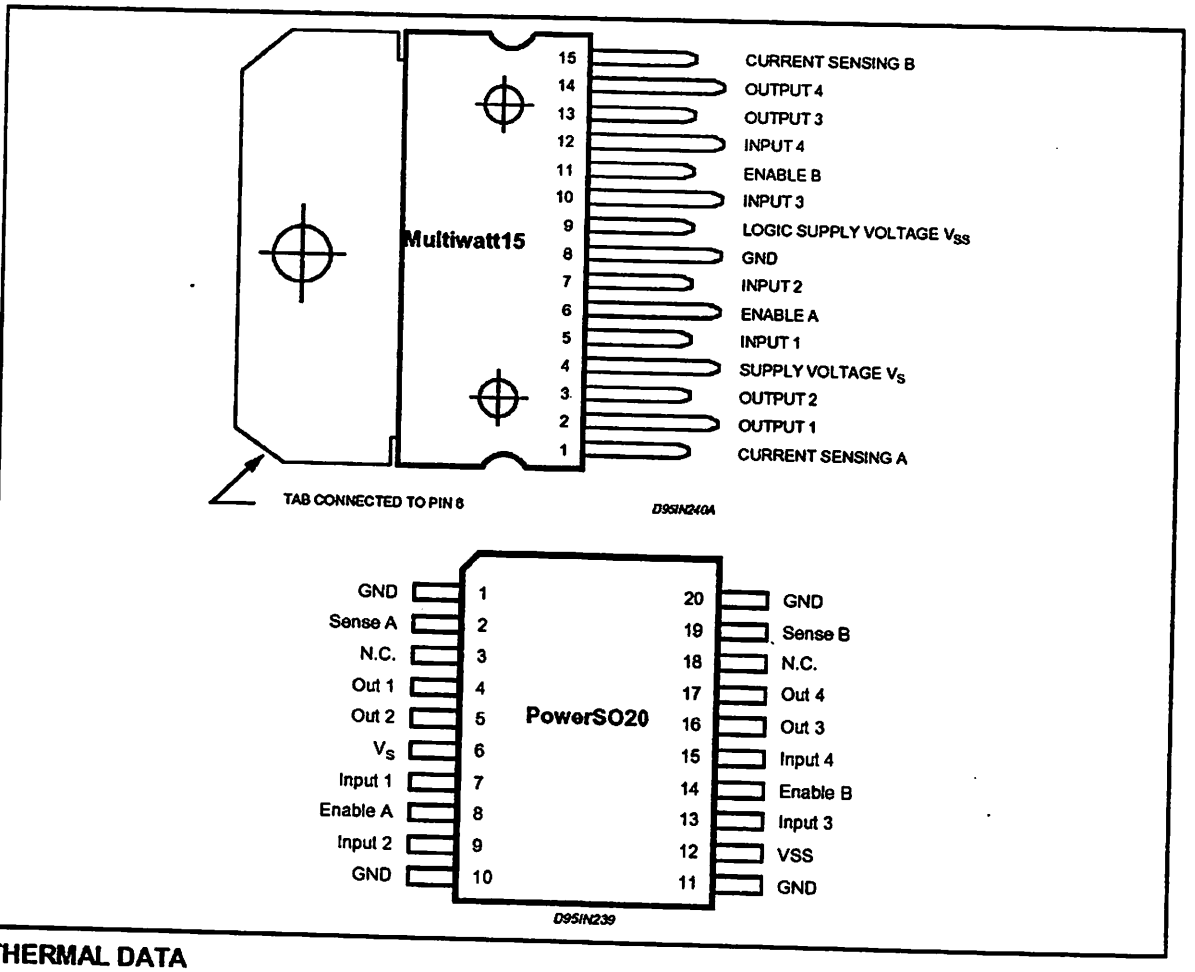
BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Power Supply	50	V
V_{SS}	Logic Supply Voltage	7	V
V_I, V_{en}	Input and Enable Voltage	–0.3 to 7	V
I_O	Peak Output Current (each Channel) – Non Repetitive (t = 100µs) –Repetitive (80% on –20% off; t_{on} = 10ms) –DC Operation	 3 2.5 2	 A A A
V_{sens}	Sensing Voltage	–1 to 2.3	V
P_{tot}	Total Power Dissipation (T_{case} = 75°C)	25	W
T_{op}	Junction Operating Temperature	–25 to 130	°C
T_{stg}, T_j	Storage and Junction Temperature	–40 to 150	°C

PIN CONNECTIONS (top view)



THERMAL DATA

Symbol	Parameter		PowerSO20	Multiwatt15	Unit
$R_{th\ j\text{-}case}$	Thermal Resistance Junction-case	Max.	–	3	°C/W
$R_{th\ j\text{-}amb}$	Thermal Resistance Junction-ambient	Max.	13 (*)	35	°C/W

(*) Mounted on aluminum substrate

PIN FUNCTIONS (refer to the block diagram)

MW.15	PowerSO	Name	Function
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V_S	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1,10,11,20	GND	Ground.
9	12	VSS	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground.
10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
–	3;18	N.C.	Not Connected

ELECTRICAL CHARACTERISTICS (V_S = 42V; V_{SS} = 5V, T_j = 25°C; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_S	Supply Voltage (pin 4)	Operative Condition	V_{IH} +2.5		46	V
V_{SS}	Logic Supply Voltage (pin 9)		4.5	5	7	V
I_S	Quiescent Supply Current (pin 4)	V_{en} = H; I_L = 0 V_i = L		13	22	mA
		V_i = H		50	70	mA
		V_{en} = L V_i = X			4	mA
I_{SS}	Quiescent Current from V_{SS} (pin 9)	V_{en} = H; I_L = 0 V_i = L		24	36	mA
		V_i = H		7	12	mA
		V_{en} = L V_i = X			6	mA
V_{iL}	Input Low Voltage (pins 5, 7, 10, 12)		–0.3		1.5	V
V_{iH}	Input High Voltage (pins 5, 7, 10, 12)		2.3		VSS	V
I_{iL}	Low Voltage Input Current (pins 5, 7, 10, 12)	V_i = L			–10	µA
I_{iH}	High Voltage Input Current (pins 5, 7, 10, 12)	V_i = H ≤ V_{SS} –0.6V		30	100	µA
V_{en} = L	Enable Low Voltage (pins 6, 11)		–0.3		1.5	V
V_{en} = H	Enable High Voltage (pins 6, 11)		2.3		V_{SS}	V
I_{en} = L	Low Voltage Enable Current (pins 6, 11)	V_{en} = L			–10	µA
I_{en} = H	High Voltage Enable Current (pins 6, 11)	V_{en} = H ≤ V_{SS} –0.6V		30	100	µA
$V_{CEsat(H)}$	Source Saturation Voltage	I_L = 1A	0.95	1.35	1.7	V
		I_L = 2A		2	2.7	V
$V_{CEsat(L)}$	Sink Saturation Voltage	I_L = 1A (5)	0.85	1.2	1.6	V
		I_L = 2A (5)		1.7	2.3	V
V_{CEsat}	Total Drop	I_L = 1A (5)	1.80		3.2	V
		I_L = 2A (5)			4.9	V
V_{sens}	Sensing Voltage (pins 1, 15)		–1 (1)		2	V

ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
T_1 (V_i)	Source Current Turn-off Delay	0.5 V_i to 0.9 I_L (2); (4)		1.5		µs
T_2 (V_i)	Source Current Fall Time	0.9 I_L to 0.1 I_L (2); (4)		0.2		µs
T_3 (V_i)	Source Current Turn-on Delay	0.5 V_i to 0.1 I_L (2); (4)		2		µs
T_4 (V_i)	Source Current Rise Time	0.1 I_L to 0.9 I_L (2); (4)		0.7		µs
T_5 (V_i)	Sink Current Turn-off Delay	0.5 V_i to 0.9 I_L (3); (4)		0.7		µs
T_6 (V_i)	Sink Current Fall Time	0.9 I_L to 0.1 I_L (3); (4)		0.25		µs
T_7 (V_i)	Sink Current Turn-on Delay	0.5 V_i to 0.9 I_L (3); (4)		1.6		µs
T_8 (V_i)	Sink Current Rise Time	0.1 I_L to 0.9 I_L (3); (4)		0.2		µs
fc (V_i)	Commutation Frequency	I_L = 2A		25	40	KHz
T_1 (V_{en})	Source Current Turn-off Delay	0.5 V_{en} to 0.9 I_L (2); (4)		3		µs
T_2 (V_{en})	Source Current Fall Time	0.9 I_L to 0.1 I_L (2); (4)		1		µs
T_3 (V_{en})	Source Current Turn-on Delay	0.5 V_{en} to 0.1 I_L (2); (4)		0.3		µs
T_4 (V_{en})	Source Current Rise Time	0.1 I_L to 0.9 I_L (2); (4)		0.4		µs
T_5 (V_{en})	Sink Current Turn-off Delay	0.5 V_{en} to 0.9 I_L (3); (4)		2.2		µs
T_6 (V_{en})	Sink Current Fall Time	0.9 I_L to 0.1 I_L (3); (4)		0.35		µs
T_7 (V_{en})	Sink Current Turn-on Delay	0.5 V_{en} to 0.9 I_L (3); (4)		0.25		µs
T_8 (V_{en})	Sink Current Rise Time	0.1 I_L to 0.9 I_L (3); (4)		0.1		µs

1) 1)Sensing voltage can be –1 V for t ≤ 50 µsec; in steady state V_{sens} min ≥ –0.5 V.
2) See fig. 2.
3) See fig. 4.
4) The load must be a pure resistor.

Figure 1 : Typical Saturation Voltage vs. Output Current.

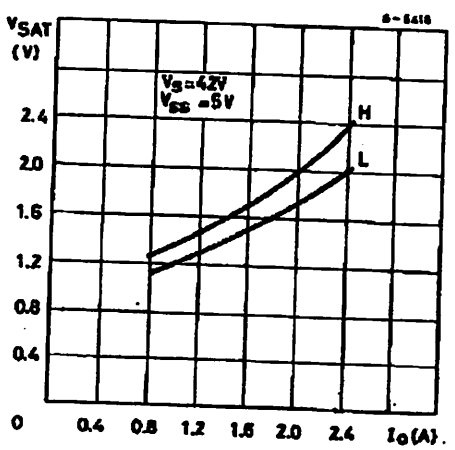
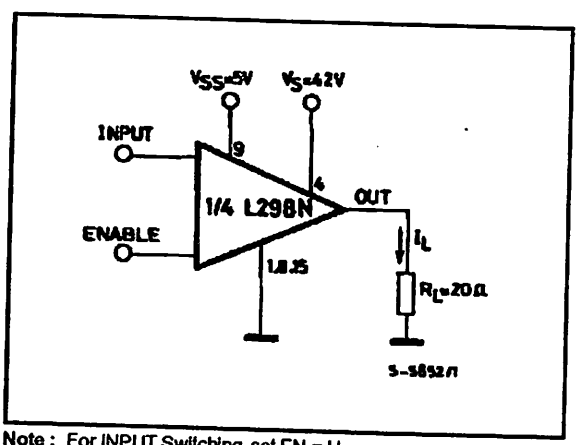


Figure 2 : Switching Times Test Circuits.



Note : For INPUT Switching, set EN = H
For ENABLE Switching, set IN = H

Figure 3 : Source Current Delay Times vs. Input or Enable Switching.

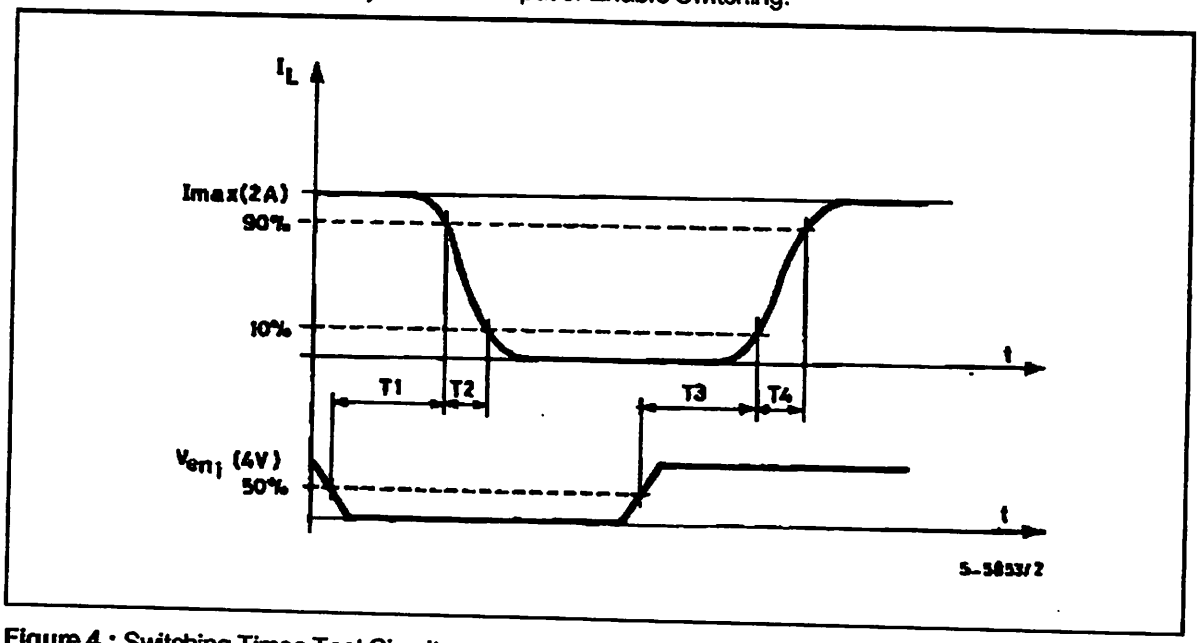
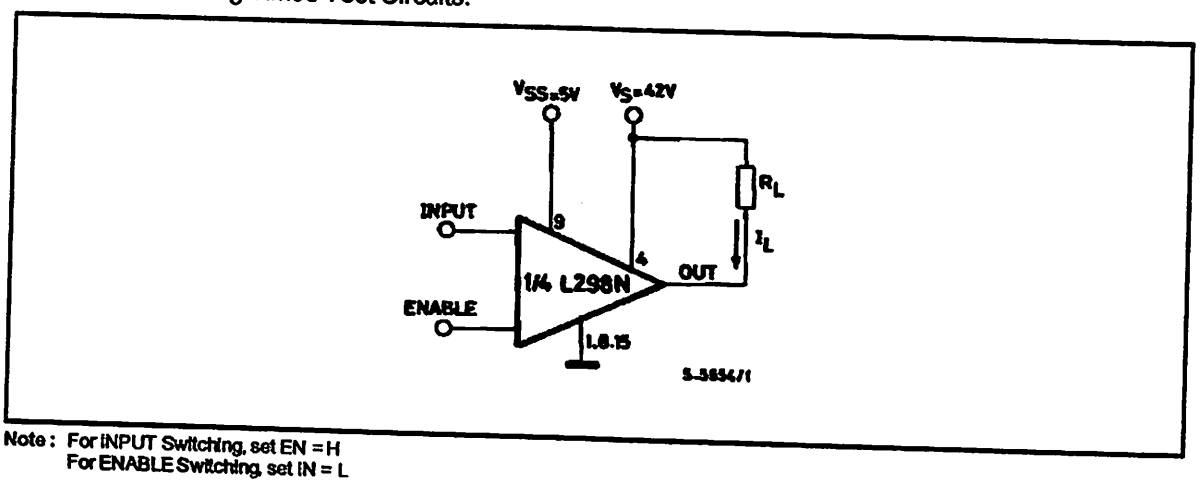


Figure 4 : Switching Times Test Circuits.



Note : For INPUT Switching, set EN = H
For ENABLE Switching, set IN = L

Figure 5 : Sink Current Delay Times vs. Input 0 V Enable Switching.

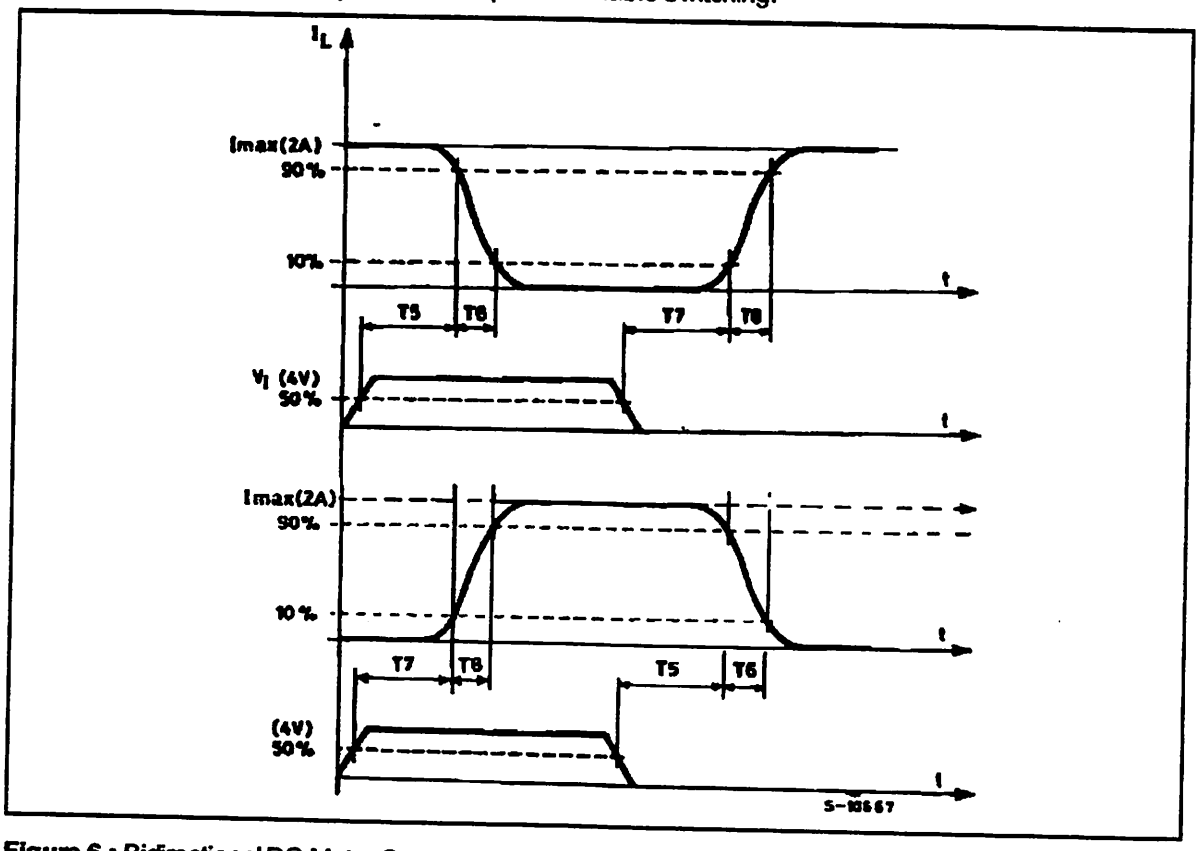
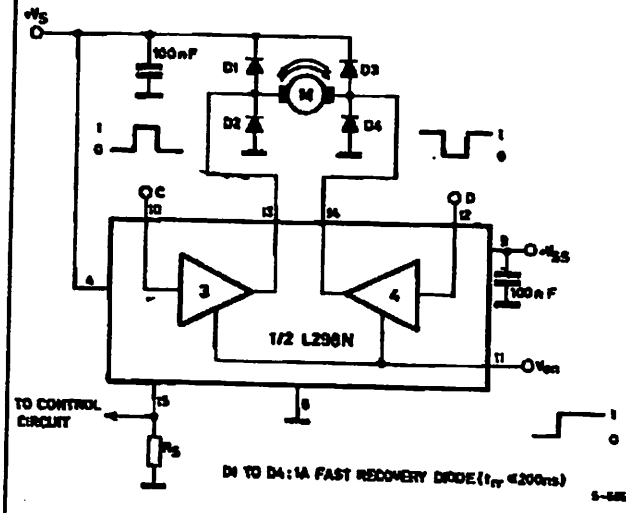


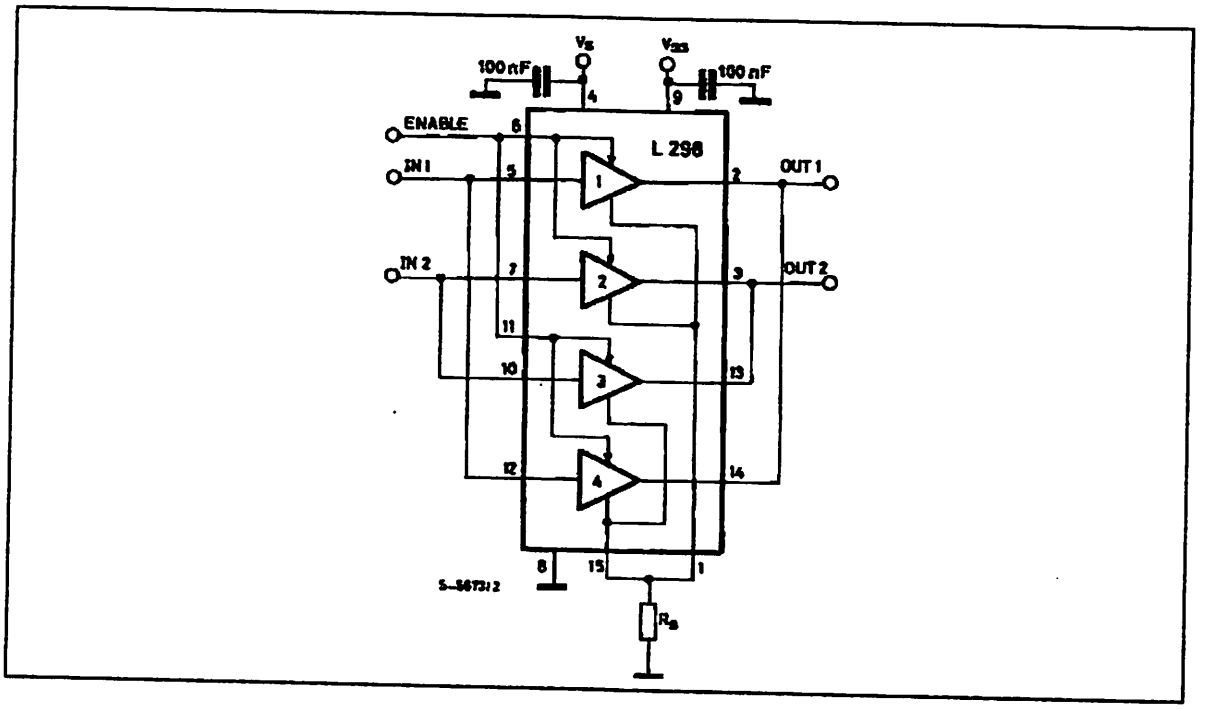
Figure 6 : Bidirectional DC Motor Control.



Inputs		Function
V_{en} = H	C = H ; D = L	Forward
	C = L ; D = H	Reverse
	C = D	Fast Motor Stop
V_{en} = L	C = X ; D = X	Free Running Motor Stop

L = Low H = High X = Don't care

Figure 7 : For higher currents, outputs can be paralleled. Take care to parallel channel 1 with channel 4 and channel 2 with channel 3.



APPLICATION INFORMATION (Refer to the block diagram)

1.1. POWER OUTPUT STAGE

The L298 integrates two power output stages (A ; B). The power output stage is a bridge configuration and its outputs can drive an inductive load in common or differenzial mode, depending on the state of the inputs. The current that flows through the load comes out from the bridge at the sense output : an external resistor (R_{SA} ; R_{SB}.) allows to detect the intensity of this current.

1.2. INPUT STAGE

Each bridge is driven by means of four gates the input of which are In1 ; In2 ; EnA and In3 ; In4 ; EnB. The In inputs set the bridge state when The En input is high ; a low state of the En input inhibits the bridge. All the inputs are TTL compatible.

2. SUGGESTIONS

A non inductive capacitor, usually of 100 nF, must be foreseen between both Vs and Vss, to ground, as near as possible to GND pin. When the large capacitor of the power supply is too far from the IC, a second smaller one must be foreseen near the L298.

The sense resistor, not of a wire wound type, must be grounded near the negative pole of Vs that must be near the GND pin of the I.C.

Each input must be connected to the source of the driving signals by means of a very short path.

Turn-On and Turn-Off : Before to Turn-ON the Supply Voltage and before to Turn it OFF, the Enable input must be driven to the Low state.

3. APPLICATIONS

Fig 6 shows a bidirectional DC motor control Schematic Diagram for which only one bridge is needed. The external bridge of diodes D1 to D4 is made by four fast recovery elements ($t_{rr} \leq 200$ nsec) that must be chosen of a VF as low as possible at the worst case of the load current.

The sense output voltage can be used to control the current amplitude by chopping the inputs, or to provide overcurrent protection by switching low the enable input.

The brake function (Fast motor stop) requires that the Absolute Maximum Rating of 2 Amps must never be overcome.

When the repetitive peak current needed from the load is higher than 2 Amps, a paralleled configuration can be chosen (See Fig.7).

An external bridge of diodes are required when inductive loads are driven and when the inputs of the IC are chopped ; Shottky diodes would be preferred.

This solution can drive until 3 Amps In DC operation and until 3.5 Amps of a repetitive peak current.

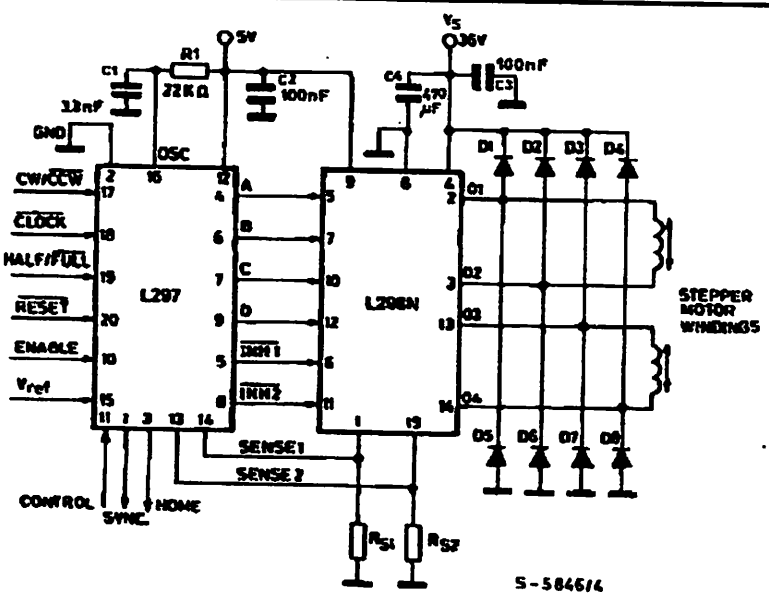
On Fig 8 it is shown the driving of a two phase bipolar stepper motor ; the needed signals to drive the inputs of the L298 are generated, in this example, from the IC L297.

Fig 9 shows an example of P.C.B. designed for the application of Fig 8.

Fig 10 shows a second two phase bipolar stepper motor control circuit where the current is controlled by the I.C. L6506.

Figure 8 : Two Phase Bipolar Stepper Motor Circuit.

This circuit drives bipolar stepper motors with winding currents up to 2 A. The diodes are fast 2 A types.



$R_{S1} = R_{S2} = 0.5\,\Omega$

D1 to D8 = 2 A Fast diodes { $V_F \leq 1.2$ V @ I = 2 A ; $t_{rr} \leq 200$ ns

Figure 9 : Suggested Printed Circuit Board Layout for the Circuit of fig. 8 (1:1 scale).

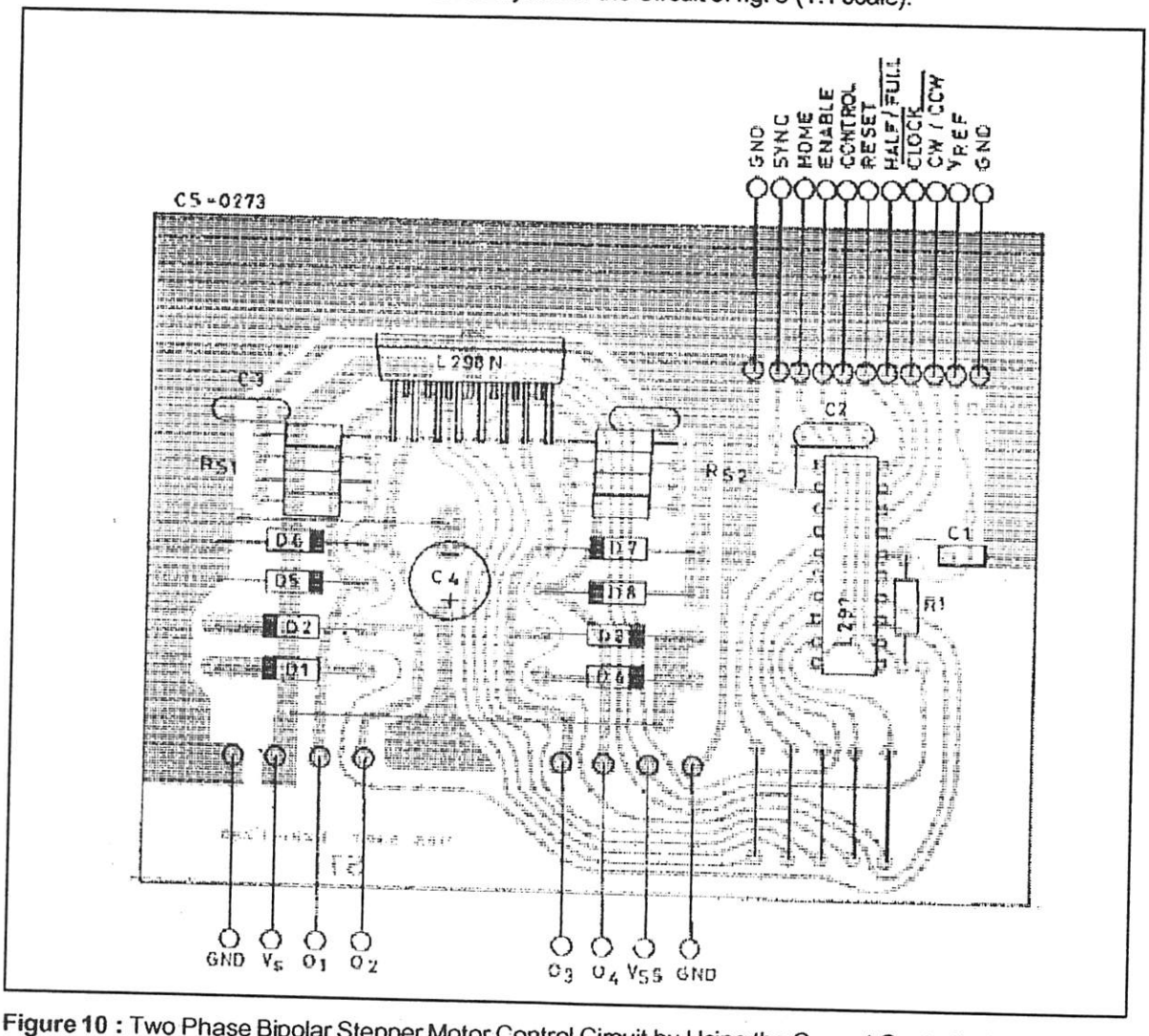
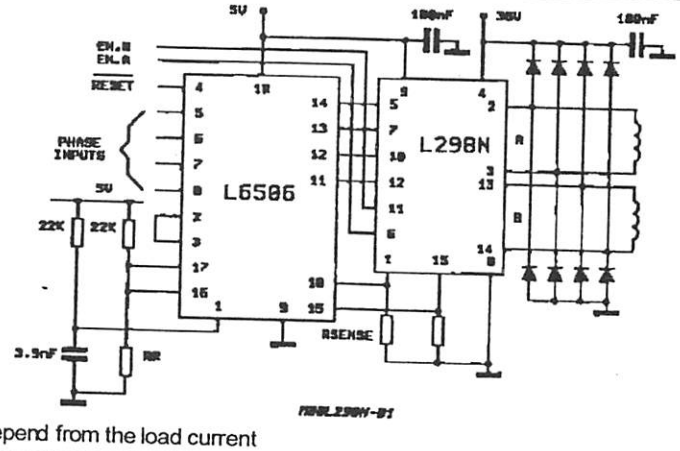


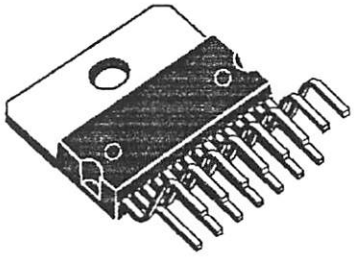
Figure 10 : Two Phase Bipolar Stepper Motor Control Circuit by Using the Current Controller L6506.



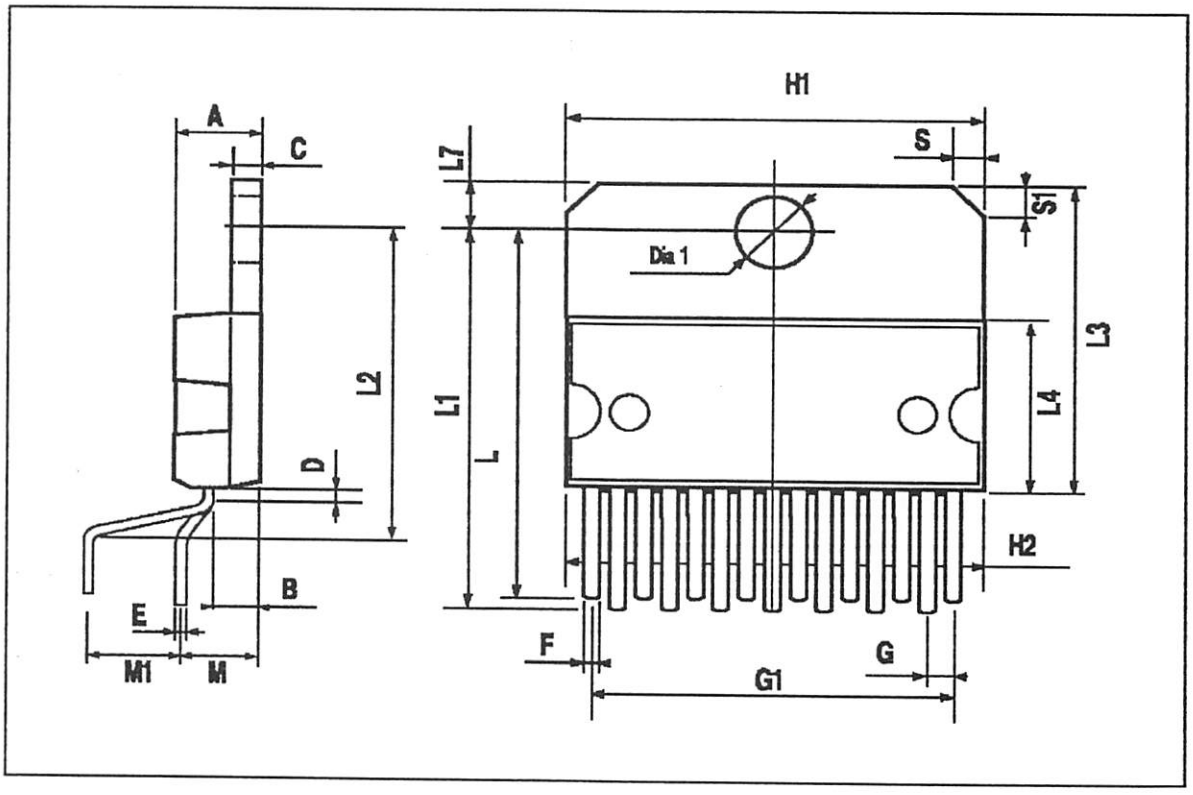
R_R and R_{sense} depend from the load current

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
D		1			0.039	
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.02	1.27	1.52	0.040	0.050	0.060
G1	17.53	17.78	18.03	0.690	0.700	0.710
H1	19.6			0.772		
H2			20.2			0.795
L	21.9	22.2	22.5	0.862	0.874	0.886
L1	21.7	22.1	22.5	0.854	0.870	0.886
L2	17.65		18.1	0.695		0.713
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L7	2.65		2.9	0.104		0.114
M	4.25	4.55	4.85	0.167	0.179	0.191
M1	4.63	5.08	5.53	0.182	0.200	0.218
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

OUTLINE AND MECHANICAL DATA

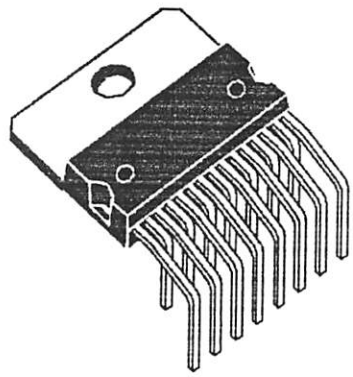


Multiwatt15 V

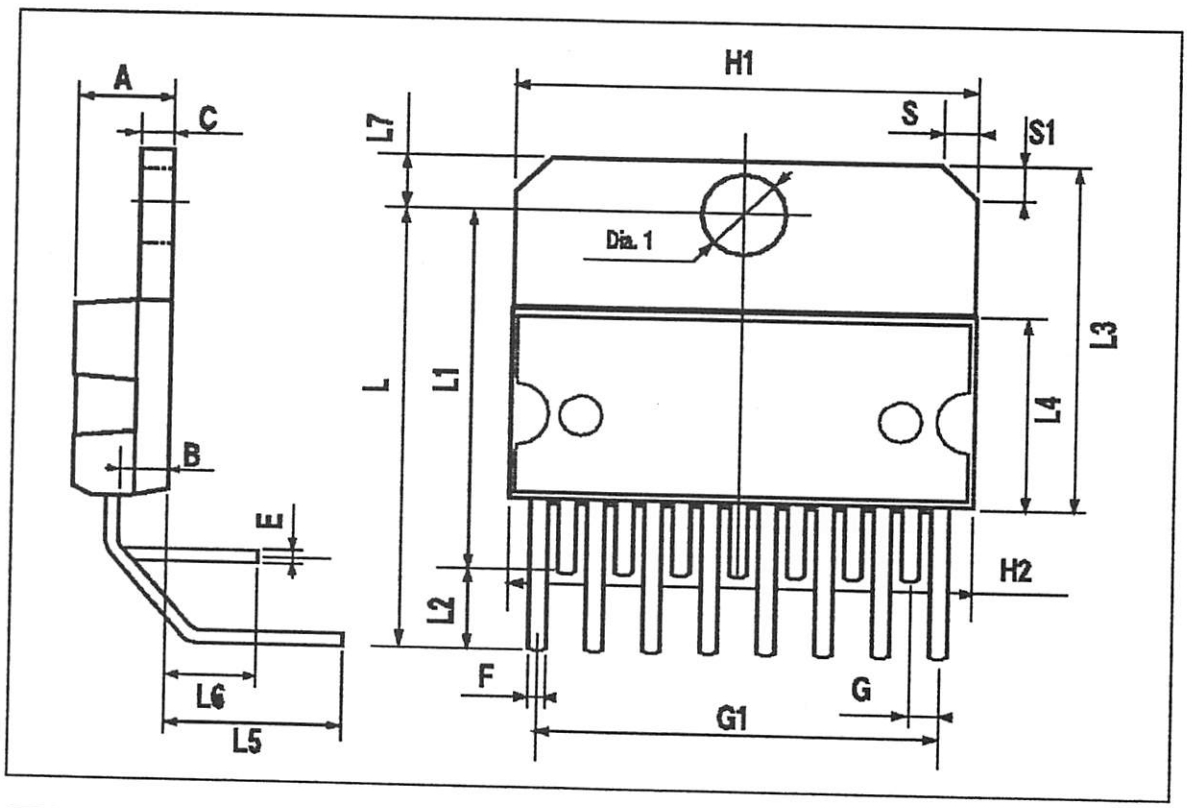


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.14	1.27	1.4	0.045	0.050	0.055
G1	17.57	17.78	17.91	0.692	0.700	0.705
H1	19.6			0.772		
H2			20.2			0.795
L		20.57			0.810	
L1		18.03			0.710	
L2		2.54			0.100	
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L5		5.28			0.208	
L6		2.38			0.094	
L7	2.65		2.9	0.104		0.114
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

OUTLINE AND MECHANICAL DATA



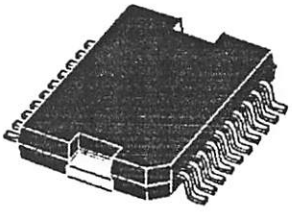
Multiwatt15 H



DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			3.6			0.142
a1	0.1		0.3	0.004		0.012
a2			3.3			0.130
a3	0		0.1	0.000		0.004
b	0.4		0.53	0.016		0.021
c	0.23		0.32	0.009		0.013
D (1)	15.8		16	0.622		0.630
D1	9.4		9.8	0.370		0.386
E	13.9		14.5	0.547		0.570
e		1.27			0.050	
e3		11.43			0.450	
E1 (1)	10.9		11.1	0.429		0.437
E2			2.9			0.114
E3	5.8		6.2	0.228		0.244
G	0		0.1	0.000		0.004
H	15.5		15.9	0.610		0.626
h			1.1			0.043
L	0.8		1.1	0.031		0.043
N	10° (max.)					
S	8° (max.)					
T		10			0.394	

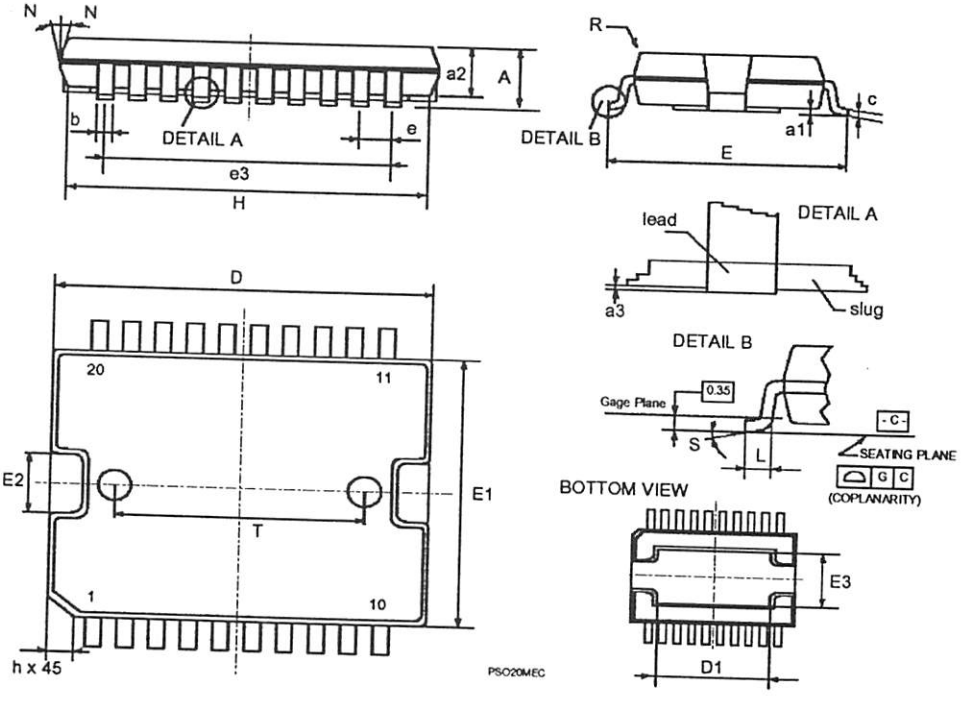
(1) "D and F" do not include mold flash or protrusions.
- Mold flash or protrusions shall not exceed 0.15 mm (0.006").
- Critical dimensions: "E", "G" and "a3"

OUTLINE AND MECHANICAL DATA



JEDEC MO-166

PowerSO20



SGS-THOMSON MICROELECTRONICS

APPLICATION NOTE

STEPPER MOTOR DRIVER CONSIDERATIONS COMMON PROBLEMS & SOLUTIONS

by Thomas L. Hopkins

This note explains how to avoid same of the more common pitfalls in motor drive design. It is based on the author's experience in responding to enquiries from the field.

INTRODUCTION

Over the years while working with stepper motor users, many of the same questions keep occurring from novice as well as experienced users of stepper motors. This application note is intended as a collection of answers to commonly asked questions about stepper motors and driver design. In addition the reference list contains a number of other application notes, books and articles that a designer may find useful in applying stepper motors.

Throughout the course of this discussion the reader will find references to the L6201, L6202 and L6203. Since these devices are the same die and differ only in package, any reference to one of the devices should be considered to mean any of the three devices.

Motor Selection (Unipolar vs Bipolar)

Stepper motors in common use can be divided into general classes, Unipolar driven motors and Bipolar driven motors. In the past unipolar motors were common and preferred for their simple drive configurations. However, with the advent of cost effective integrated drivers, bipolar motors are now more common. These bipolar motors typically produce a higher torque in a given form factor [1].

Drive Topology Selection

Depending on the torque and speed required from a stepper motor there are several motor drive topologies available [5, chapter3]. At low speeds a simple direct voltage drive, giving the motor just sufficient voltage so that the internal resistance of the motor limits the current to the allowed value as shown in Figure 1A, may be sufficient. However at higher rotational speeds there is a significant fall off of torque since the winding inductance limits the rate of change of the current and the current can no longer reach it's full value in each step, as shown in Figure 2.

Figure 1: Simple direct voltage unipolar motors drive.

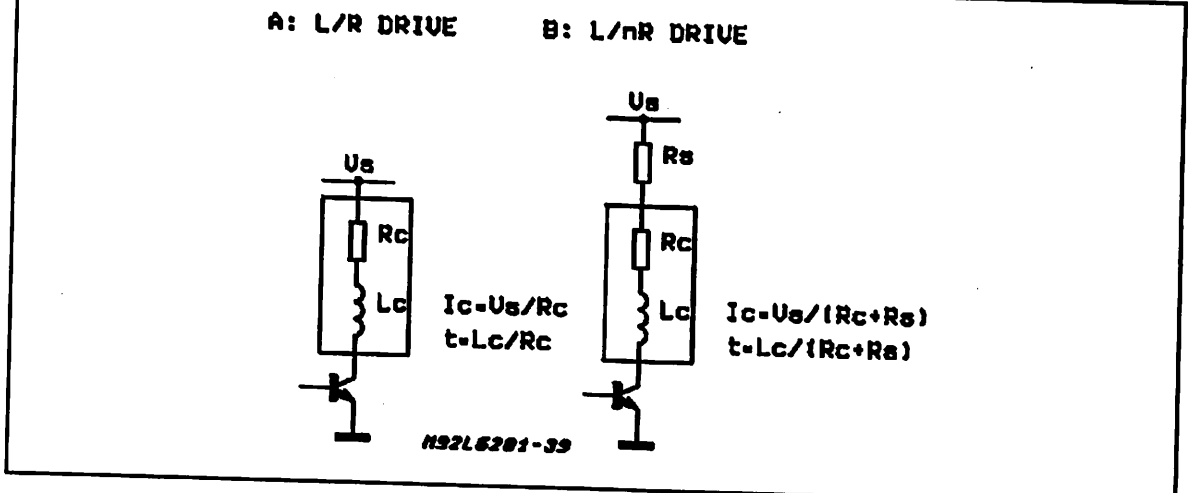
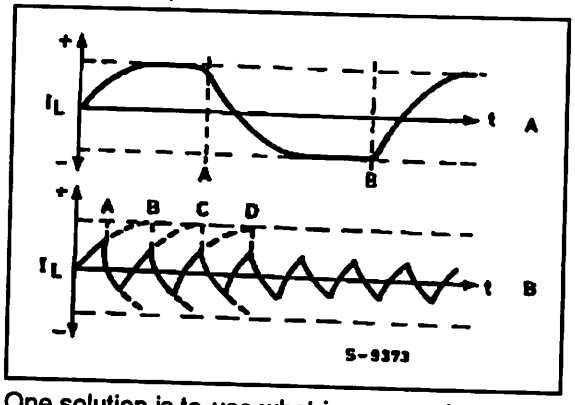


Figure 2: Direct voltage drive.
A - low speed;
B - too high speed generates fall of torque.



One solution is to use what is commonly referred to as an L/nR drive (Fig. 1B). In this topology a higher voltage is used and the current limit is set by an external resistor in series with the motor winding such that the sum of the external resistance and the internal winding resistance limits the current to the allowed value. This drive technique increases the current slew rate and typically provides better torque at high rotational speed. However there is a significant penalty paid in additional dissipation in the external resistances.

To avoid the additional dissipation a chopping controlled current drive may be employed, as shown in Figure 3. In this technique the current through the motor is sensed and controlled by a chopping control circuit so that it is maintained within the rated level. Devices like the L297, L6506 and PBL3717A implement this type of control. This technique improves the current rise time in the motor and improves the torque at high speeds while maintaining a high efficiency in the drive [2]. Figure 4 shows a comparison between the winding current wave forms for the same motor driven in these three techniques.

Figure 3: Chopper drive provides better performance.

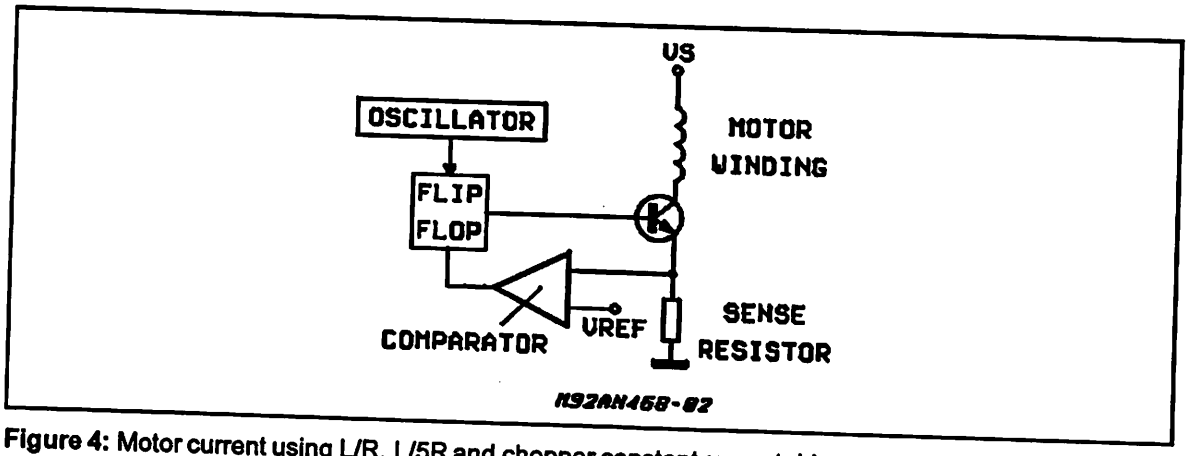
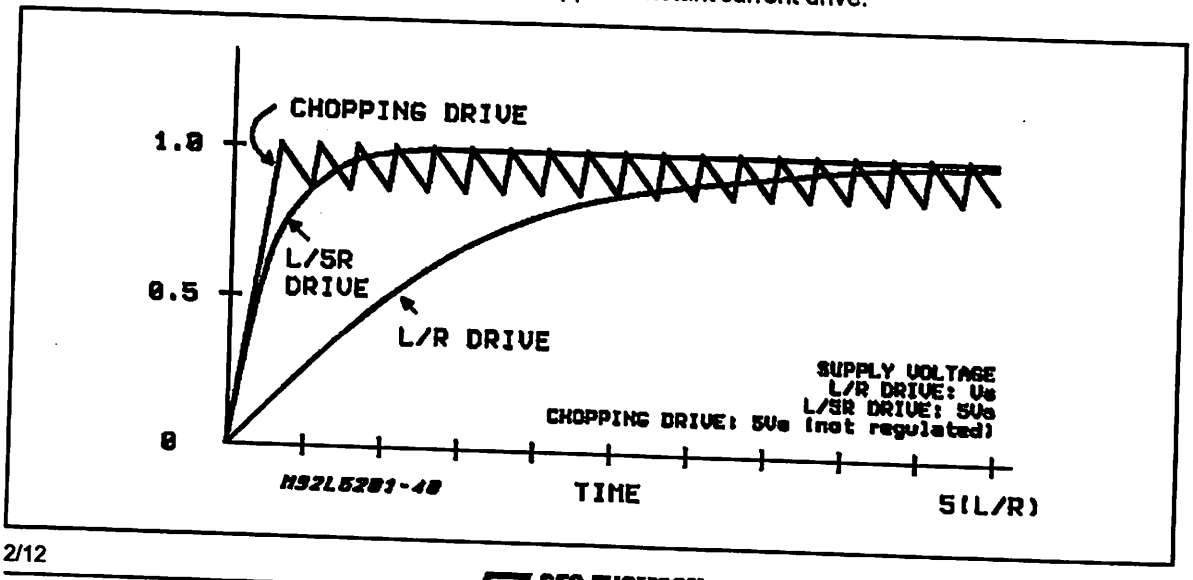


Figure 4: Motor current using L/R, L/5R and chopper constant current drive.



In general the best performance, in terms of torque, is achieved using the chopping current control technique [2]. This technique also allows easy implementation of multiple current level drive techniques to improve the motor performance. [1]

Driving a Unipolar Motor with the L298N or L6202

Although it is not the optimal solution, design constraints sometimes limit the motor selection. In the case where the designer is looking for a highly integrated drive stage with improved performance over previous designs but is constrained to drive a unipolar wound (6 leaded) motor it is possible to drive the motor with H-Bridge drivers like the L298N or L6202. To drive such a motor the center tap of the motor should be left unconnected and the two ends of the common windings are connected to the bridge outputs, as shown in Figure 5. In this configuration the user should notice a marked improvement in torque for the same coil current, or put another way, the same torque output will be achieved with a lower coil current.

A solution where the L298N or L6202 is used to drive a unipolar motor while keeping the center connection of each coil connected to the supply will not work. First, the protection diodes needed from collector to emitter (drain to source) of the bridge transistors will be forward biased by the transformer action of the motor windings, providing an effective short circuit across the supply. Secondly the L298N, even though it has split supply voltages, may not be used without a high voltage supply on the chip since a portion of the drive current for the output bridge is derived from this supply.

Selecting Enable or Phase chopping

When implementing chopping control of the current in a stepper motor, there are several ways in which the current control can be implemented. A bridge output, like the L6202 or L298N, may be driven in enable chopping, one phase chopping or two phase chopping, as shown in Figure 6. The L297 implements enable chopping or one phase chopping, selected by the control input. The L6506 implements one phase chopping, with the recirculation path around the lower half of the bridge, if the four outputs are connected to the 4 inputs of the bridge or enable chopping if the odd numbered outputs are connected to the enable inputs of the bridge. Selecting the correct chopping mode is an important consideration that affects the stability of the system as well as the dissipation. Table 1 shows a relative comparison of the different chopping modes, for a fixed chopping frequency, motor current and motor inductance.

Figure 5: Driving a unipolar wound motor with a bipolar drive

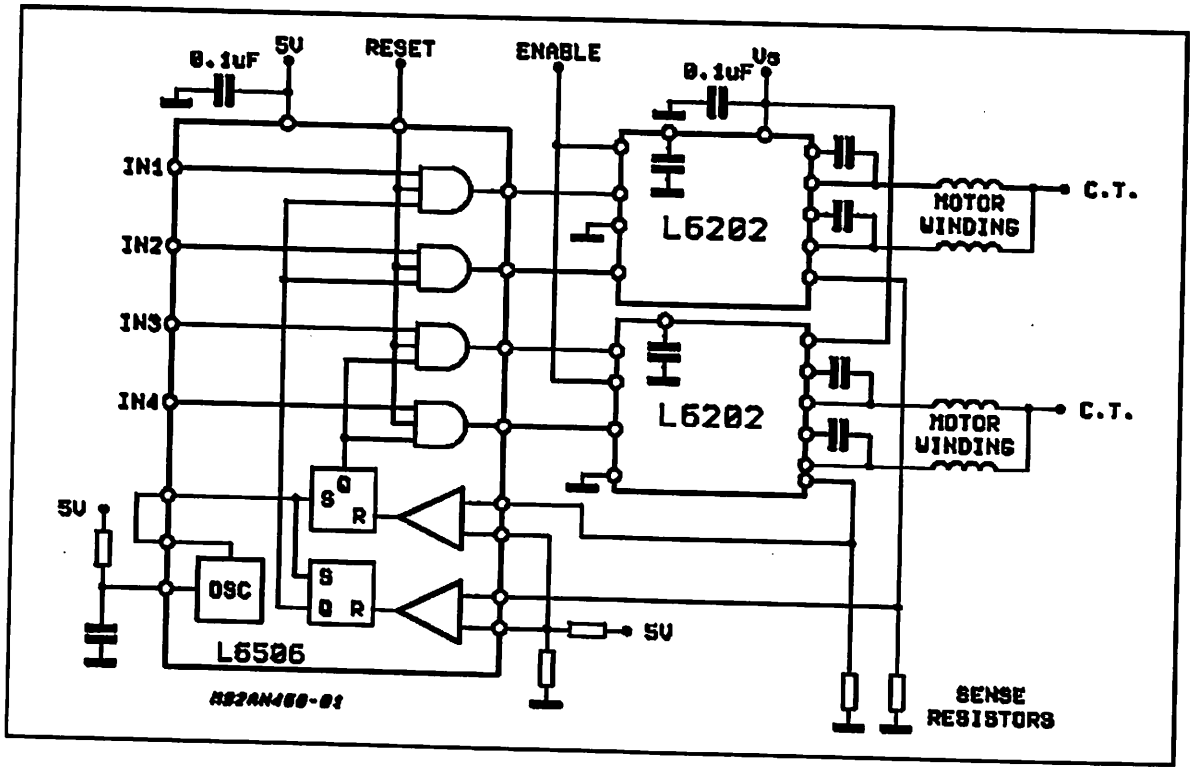


Table 1: Comparative advantages of chopping modes

Chopping Mode	Ripple Current	Motor Dissipation	Bridge Dissipation *	Minimum Current
ENABLE	HIGH	HIGH	HIGH	LOWER
ONE PHASE	LOW	LOW	LOWEST	LOW
TWO PHASE	HIGH	LOW	LOW	Ipp/2

(*) As related to L298N, L6203 or L6202.

Figure 6a: Two Phase Chopping.

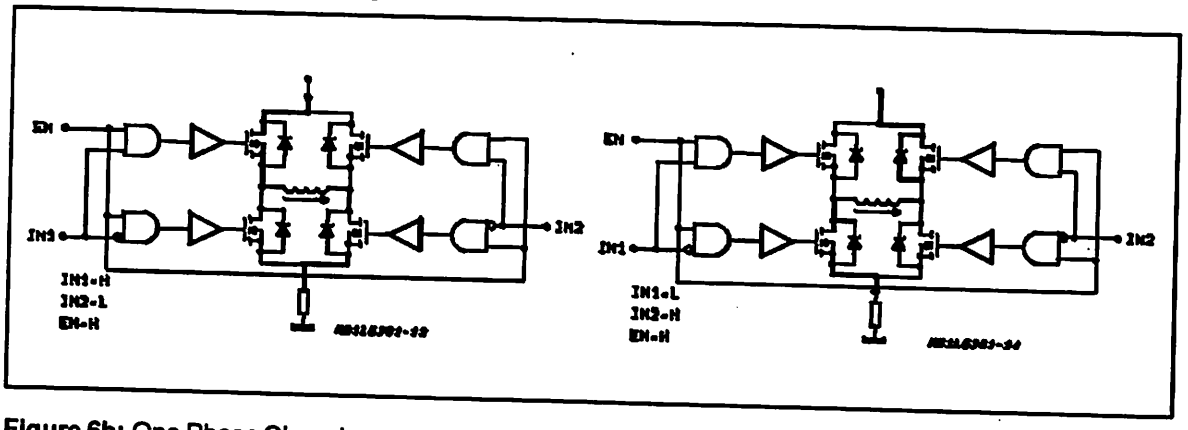


Figure 6b: One Phase Chopping.

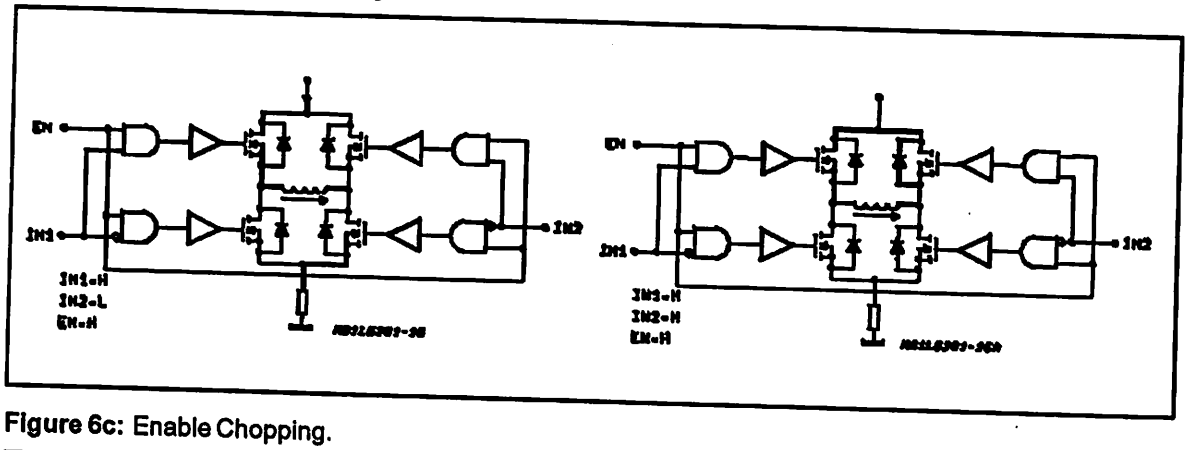
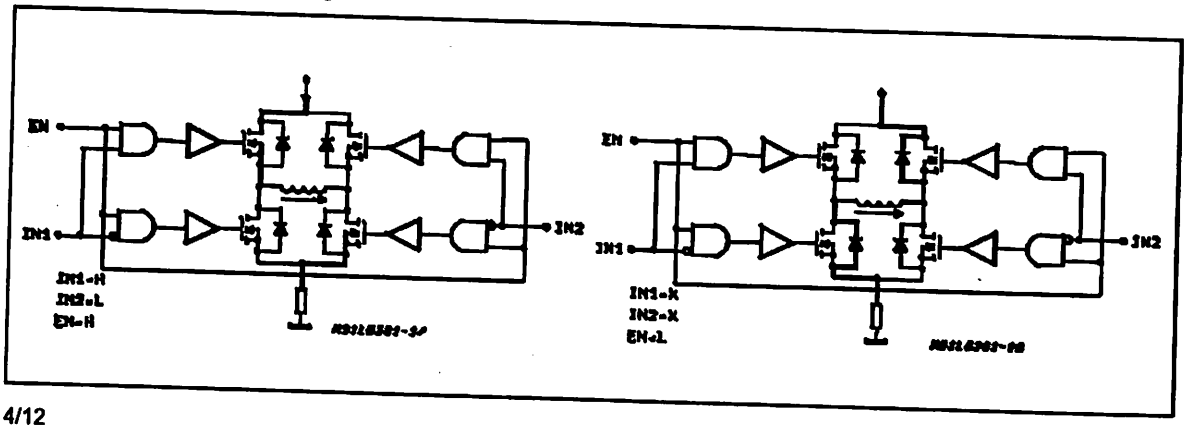


Figure 6c: Enable Chopping.



RIPPLE CURRENT

Since the rate of current change is related directly to the voltage applied across the coil by the equation:

$$V = L\frac{di}{dt}$$

the ripple current will be determined primarily by the chopping frequency and the voltage across the coil. When the coil is driven on, the voltage across the coil is fixed by the power supply minus the saturation voltages of the driver. On the other hand the voltage across the coil during the recirculation time depends on the chopping mode chosen.

When enable chopping or two phase chopping is selected, the voltage across the coil during recirculation is the supply voltage plus either the V_F of the diodes or the RI voltage of the DMOS devices (when using the L6202 in two phase chopping). In this case the slope of the current rise and decay are nearly the same and the ripple current can be large.

When one phase chopping is used, the voltage across the coil during recirculation is V_{on} (V_{sat} for Bipolar devices or $I \cdot R_{DSon}$ for DMOS) of the transistor that remains on plus V_F of one diode plus the voltage drop across the sense resistor, if it is in the recirculation path. In this case the current decays much slower than it rises and the ripple current is much smaller than in the previous case. The effect will be much more noticeable at higher supply voltages.

MOTOR LOSSES

The losses in the motor include the resistive losses (I^2R) in the motor winding and parasitic losses like eddie current losses. The latter group of parasitic losses generally increases with increased ripple currents and frequency. Chopping techniques that have a high ripple current will have higher losses in the motor. Enable or two phase chopping will cause higher losses in the motor with the effect of raising motor temperature. Generally lower motor losses are achieved using phase chopping.

POWER DISSIPATION IN THE BRIDGE IC.

In the L298N, the internal drive circuitry provides active turn off for the output devices when the outputs are switched in response to the 4 phase inputs. However when the outputs are switched off in response to the enable inputs all base drive is removed from output devices but no active element is present to remove the stored charge in the base. When enable chopping is used the fall time of the current in the power devices will be longer and the device will have higher switching losses than if phase chopping is used.

In the L6202 and L6203, the internal gate drive circuit works the same in response to either the input or the enable so the switching losses are the same using enable or two phase chopping, but would be lower using one phase chopping. However, the losses due to the voltage drops across the device are not the same. During enable chopping all four of the output DMOS devices are turned off and the current recirculates through the body to drain diodes of the DMOS output transistors. When phase chopping the DMOS devices in the recirculation path are driven on and conduct current in the reverse direction. Since the voltage drop across the DMOS device is less than the forward voltage drop of the diode for currents less than 2A, the DMOS take a significant amount of the current and the power dissipation is much lower using phase chopping than enable chopping, as can be seen in the power dissipation graphs in the data sheet.
With these two devices, phase chopping will always provide lower dissipation in the device. For discrete bridges the switching loss and saturation losses should be evaluated to determine which is lower.

MINIMUM CURRENT

The minimum current that can be regulated is important when implementing microstepping, when implementing multilevel current controls, or anytime when attempting to regulate a current that is very small compared to the peak current that would flow if the motor were connected directly to the supply voltage used.

With enable chopping or one phase chopping the only problem is loss of regulation for currents below a minimum value. Figure 7 shows a typical response curve for output current as a function of the set reference. This minimum value is set by the motor characteristics, primarily the motor resistance, the supply voltage and the minimum duty cycle achievable by the control circuit. The minimum current that can be supplied is the current that flows through the winding when driven by the minimum duty cycle. Below this value current regulation is not possible. With enable chopping the current through the coil in response to the minimum duty cycle can return completely to zero during each cycle, as shown in figure 8. When using one phase chopping the current may or may not return completely to zero and there may be some residual DC component.

When using a constant frequency control like the L297 or L6506, the minimum duty cycle is basically the duty cycle of the oscillator (sync) since the set dominance of the flip-flop maintains the output on during the time the sync is active. In constant off time regulators, like the PBL3717A, the minimum output time is set by the propagation delay through the circuit and it's ratio to the selected off time.

Figure 7: The transfer function of peak detect current control is nonlinear for low current values.

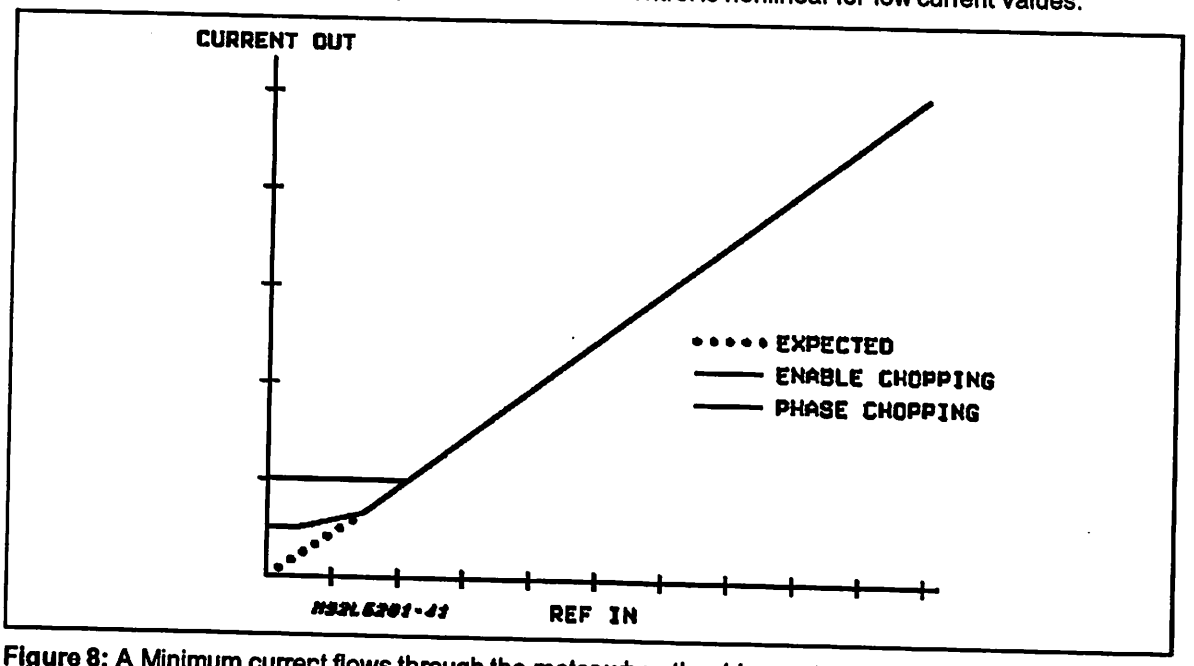
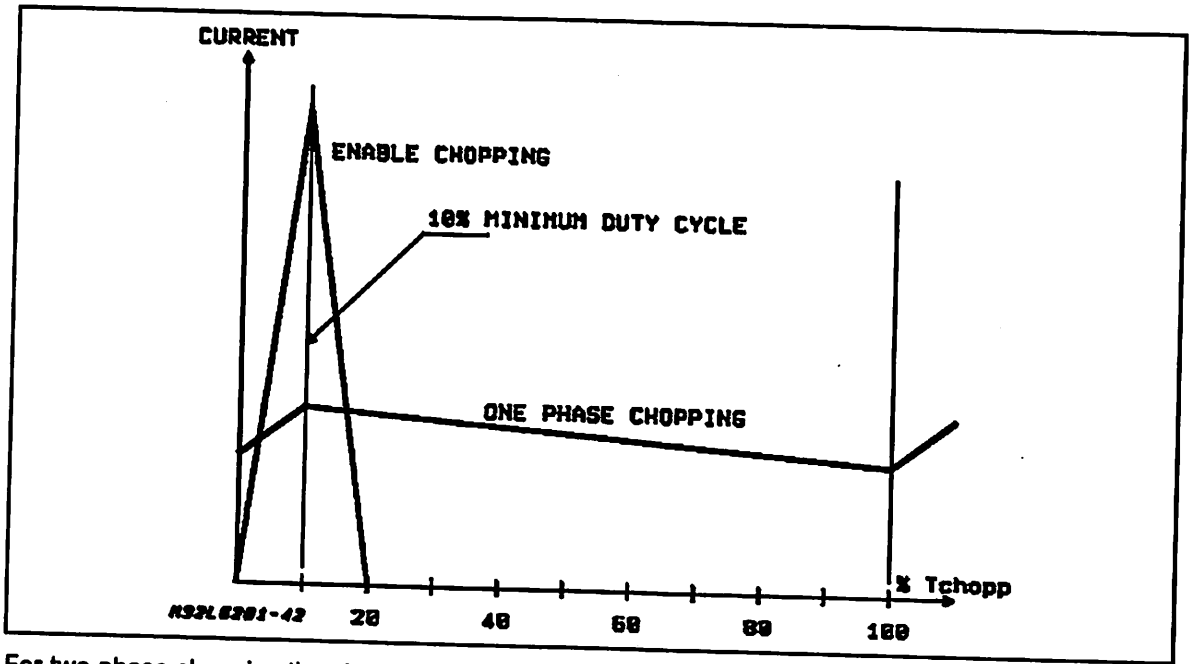
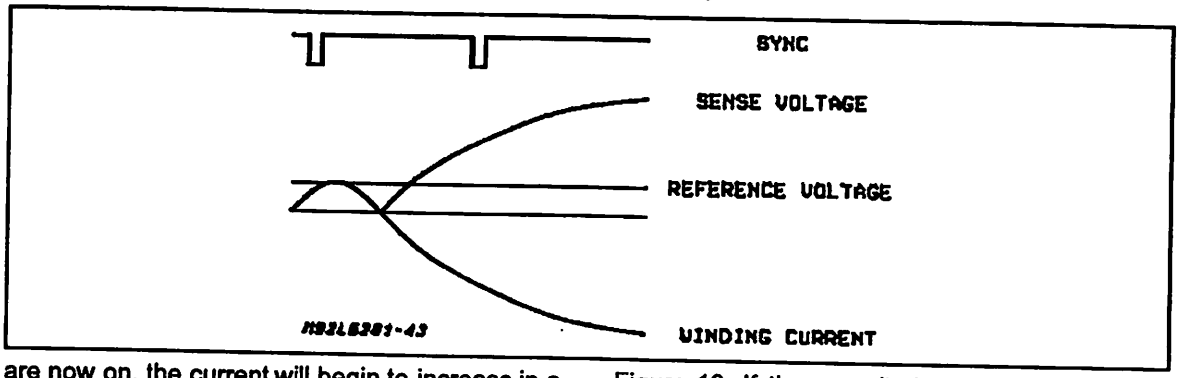


Figure 8: A Minimum current flows through the motor when the driver outputs the minimum duty cycle that is achievable.



For two phase chopping the situation is quite different. Although none of the available control chips implement this mode it is discussed here since it is easy to generate currents that can be catastrophic if two phase chopping is used with peak detecting control techniques. When the peak current is less than 1/2 of the ripple (I_{pp}) current two phase chopping can be especially dangerous. In this case the reverse drive ability of the two phase chopping technique can cause the current in the motor winding to reverse and the control circuit to lose control. Figure 9 shows the current wave form in this case. When the current reaches the peak set by the reference both sides of the bridge are switched and the current decays until it reaches zero. Since the power transistors

Figure 9: Two phase chopping can loose control of the winding current..



are now on, the current will begin to increase in a negative direction. When the oscillator again sets the flip-flop the inputs will then switch again and the current will begin to become more positive. However, the effect of a single sense resistor used with a bridge is to rectify current and the comparator sees only the magnitude and not the sign of the current. If the absolute value of the current in the negative direction is above the set value the comparator will be fooled and reset the flip-flop. The current will continue to become more negative and will not be controlled by the regulation circuit.

For this reason two phase chopping is not recommended with bridge circuits like the L298N or L6203 and is not implemented in any of the currently available driver IC's. The problem can be avoided by more complex current sense techniques that do not rectify the current feedback.

Chopper Stability and Audio Noise.

One problem commonly encountered when using chopping current control is audio noise from the motor which is typically a high pitch squeal. In constant frequency PWM circuits this occurrence is usually traced to a stability problem in the current control circuit where the effective chopping frequency has shifted to a sub-harmonic of the desired frequency set by the oscillator. In constant off time circuits the off time is shifted to a multiple of the off time set by the monostable. There are two common causes for this occurrence.

The first cause is related to the electrical noise and current spikes in the application that can fool the current control circuit. In peak detect PWM circuits, like the L297 and L6506, the motor current is sensed by monitoring the voltage across the sense resistor connected to ground. When the oscillator sets the internal flip flop causing the bridge output to turn on, there is typically a voltage spike developed across this resistor. This spike is caused by noise in the system plus the reverse recovery current of the recirculating diode that flows through the sense resistor, as shown in Figure 10. If the magnitude of this spike is high enough to exceed the reference voltage, the comparator can be fooled into resetting the flip-flop prematurely as shown in Figure 11. When this occurs the output is turned off and the current continues to decay. The result is that the fundamental frequency of the current wave form delivered to the motor is reduced to a sub-harmonic of the oscillator frequency, which is usually in the audio range. In practice it is not uncommon to encounter instances where the period of the current wave form is two, three or even four times the period of the oscillator. This problem is more pronounced in breadboard implementations where the ground is not well laid out and ground noise contributes makes the spike larger.

When using the L6506 and L298N, the magnitude of the spike should be, in theory, smaller since the diode reverse recovery current flows to ground and not through the sense resistor. How-

Figure 10: Reverse recovery current of the recirculation diode flows through the sense resistor causing a spike on the sense resistor.

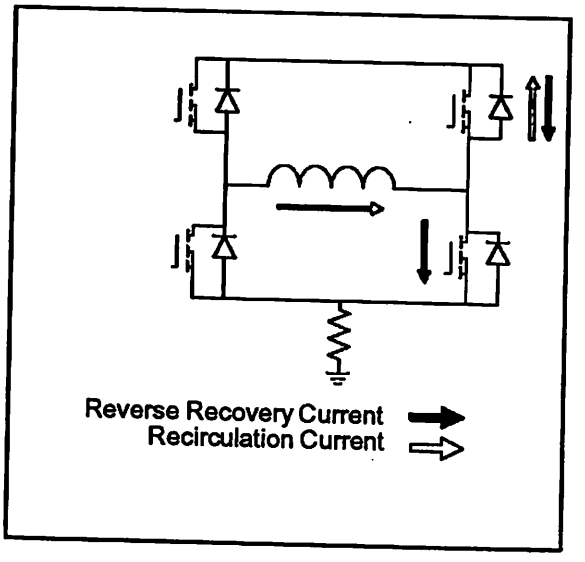
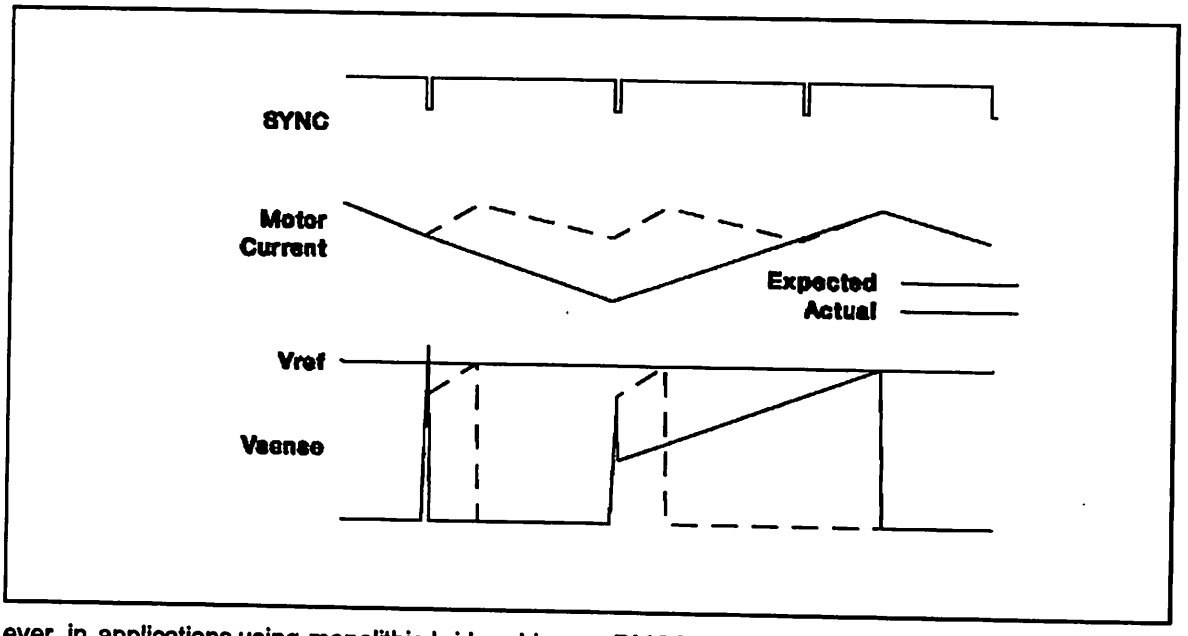


Figure 11: Spikes on the sense resistor caused by reverse recovery currents and noise can trick the current sensing comparator.



ever, in applications using monolithic bridge drivers, like the L298N, internal parasitic structures often produce recovery current spikes similar in nature to the diode reverse recovery current and these may flow through the emitter lead of the device and hence the sense resistor. When using DMOS drivers, like the L6202, the reverse recovery current always flows through the sense resistor since the internal diode in parallel with the lower transistor is connected to the source of the DMOS device and not to ground.

In constant off time FM control circuits, like the

CALCULATING POWER DISSIPATION IN BRIDGE DRIVER IC'S

The power dissipated in a monolithic driver IC like the L298N or L6202 is the sum of three elements: 1) the quiescent dissipation, 2) the saturation losses and 3) the switching losses.

The quiescent dissipation is basically the dissipation of the bias circuitry in the device and can be calculated as Vs · Is where Vs is the power supply voltage and Is is the bias current or quiscent current from the supply. When a device has two supply voltages, like the L298N, the dissipation for each must be calcualted then added to get the total quiescent dissipation. Generally the quiescent current for most monolithic IC's is constant over a vide range of input voltages and the maximum value given on the data sheet can be used for most supply voltages within the allowable range.

The saturation loss is basically the sum of the voltage drops times the current in each of the output transistors. For Bipolar devices, L298N, this is Vsat · I. For DMOS power devices this is $I^2 \cdot R_{DSon}$.

The third main component of dissipation is the switching loss associated with the output devices. In general the switching loss can be calculated as:

$$Vsupply \cdot Iload \cdot tcross \cdot fswitch$$

To calculate the total power dissipation these three compnents are each calculated, multipled by their respective duty cycle then added togther. Obviously the duty cycle for the quiescent current is equal to 100%.

PBL3717A, the noise spike fools the comparator and retriggers the monostable effectively multiplying the set off time by some integer value.

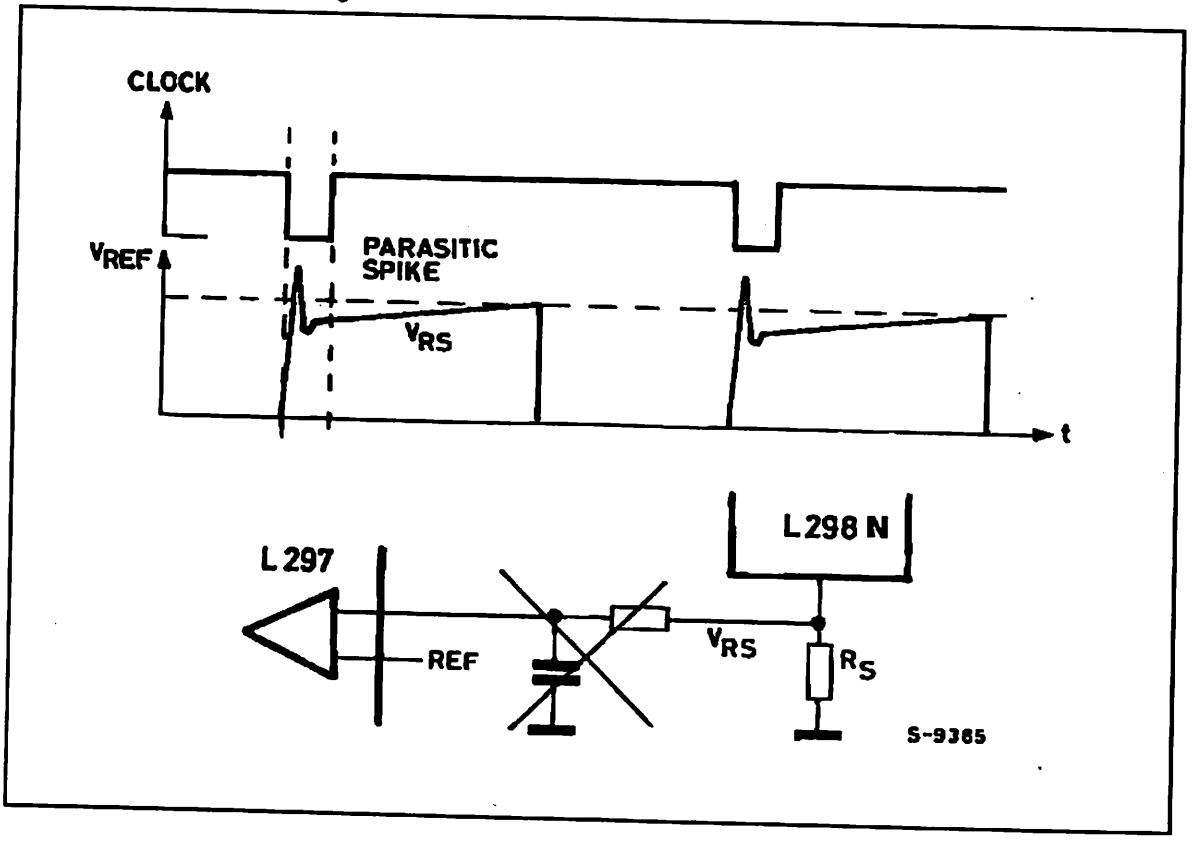
Two easy solutions to this problem are possible. The first is to put a simple RC low pass filter between the sense resistor and the sense input of the comparator. The filter attenuates the spike so it is not detected by the comparator. This obviously requires the addition of 4 additional components for a typical stepper motor. The second solution is to use the inherent set dominance of the internal flip-flop in the L297 or L6506 [1][3] to mask out the spike. To do this the width of the oscillator sync pulse is set to be longer than the sum of the propagation delay (typically 2 to 3µs for the L298N) plus the duration of the spike (usually in the range of 100ns for acceptable fast recovery diodes), as shown in figure 12. When this pulse is applied to the flip-flop set input, any signal applied to the reset input by the comparator is ignored. After the set input has been removed the comparator can properly reset the flip-flop at the correct point.

The corresponding solution in frequency modulated circuits, is to fix a blanking time during which the monostable may not be retriggered.

The best way to evaluate the stability of the chopping circuit is to stop the motor movement (hold the clock of the L297 low or hold the four inputs constant with the L6506) and look at the current wave forms without any effects of the phase changes. This evaluation should be done for each level of current that will be regulated. A DC current probe, like the Tektronix AM503 system, provides the most accurate representation of the motor current. If the circuit is operating stability, the current wave form will be synchronized to the sync signal of the control circuit. Since the spikes discussed previously are extremely short, in the range of 50 to 150 ns, a high frequency scope with a bandwidth of at least 200 MHz is required to evaluate the circuit. The sync signal to the L297 or L6506 provides the best trigger for the scope.

The other issue that affects the stability of the constant frequency PWM circuits is the chopping mode selected. With the L297 the chopping signal may be applied to either the enable inputs or the four phase inputs. When chopping is done using the enable inputs the recirculation path for the current is from ground through the lower recirculation diode, the load, the upper recirculation diode and back to the supply, as shown in Figure 6c. This same recirculation path is achieved using two phase chopping, although this may not be im-

Figure 12: The set-dominanct latch in the L297 may be used to mask spikes on the sense resistor that occur at switching.



plemented directly using the L297 or L6506. In this mode, ignoring back EMF, the voltage across the coil during the on time (t_1) when current is increasing and the recirculation time (t_2), are:

$$V_1 = V_s - 2\,V_{sat} - V_{Rsense}$$

and

$$V_2 = V_{ss} + 2\,V_F$$

The rate of current change is given by (ignoring the series resistance):

$$V = L\frac{di}{dt}$$

Since the voltage across the coil (V_2) during the recirculation time is more than the voltage (V_1) across the coil during the on time the duty cycle will, by definition, be greater than 50% because t_1 must be greater than t_2. When the back EMF of the motor is considered the duty cycle becomes even greater since the back EMF opposes the increase of current during the on time and aides the decay of current.

In this condition the control circuit may be content to operate stability at one half of the oscillator frequency, as shown in Figure 13. As in normal operation, the output is turned off when the current reaches the desired peak value and decays until the oscillator sets the flip-flop and the current again starts to increase. However since t_1 is longer than t_2 the current has not yet reached the peak value before the second oscillator pulse occurs. The second oscillator pulse then has no effect and current continues to increase until the set peak value is reached and the flip-flop is reset by the comparator. The current control circuit is completely content to keep operating in this condition. In fact the circuit may operate on one of two stable conditions depending on the random time when the peak current is first reached relative to the oscillator period.

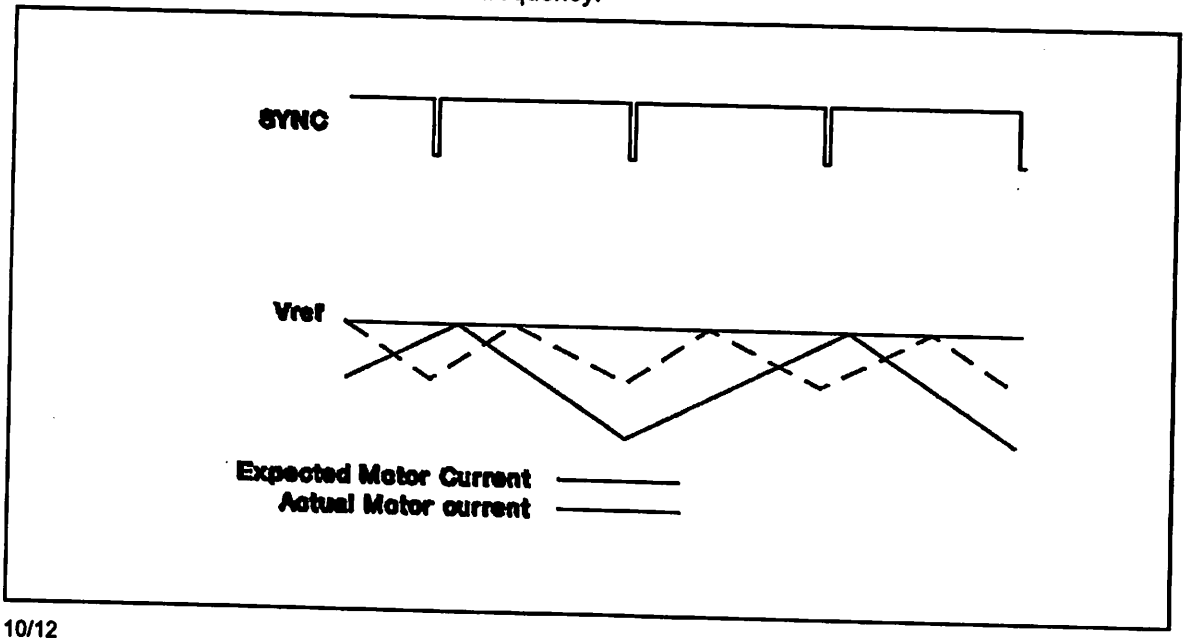
The easiest, and recommended, solution is to apply the chopping signal to only one of the phase inputs, as implemented with the L297, in the phase chopping mode, or the L6506.

Another solution that works, in some cases, is to fix a large minimum duty cycle, in the range of 30%, by applying an external clock signal to the sync input of the L297 or L6506. In this configuration the circuit must output at least the minimum duty cycle during each clock period. This forces the point where the peak current is detected to be later in each cycle and the chopping frequency to lock on the fundamental. The main disadvantage of this approach is that it sets a higher minimum current that can be controlled. The current in the motor also tends to overshoot during the first few chopping cycles since the actual peak current is not be sensed during the minimum duty cycle.

EFFECTS OF BACK EMF

As mentioned earlier, the back EMF in a stepper motor tends to increase the duty cycle of the chopping drive circuits since it opposes current increased and aids current decay. In extreme, cases where the power supply voltage is low compared to the peak back EMF of the motor, the duty cycle required when using the phase chopping may exceed 50% and the problem with the

Figure 13: When the output duty cycle exceeds 50% the chopping circuit may sinchronize of a sub-harmonic of the oscillator frequency.



stability of the operating frequency discussed above can occur. At this point the constant frequency chopping technique becomes impractical to implement and a chopping technique that uses constant off time frequency modulation like implemented in the PBL3717A, TEA3717, TEA3718, and L6219 is more useful.

Why Won't the motor move

Many first time users of chopping control drives first find that the motor does not move when the circuit is enabled. Simply put the motor is not generating sufficient torque to turn. Provided that the motor is capable of producing the required torque at the set speed, the problem usually lies in the current control circuit. As discussed in the previous section the current sensing circuit can be fooled. In extreme cases the noise is so large that the actual current through the motor is essentially zero and the motor is producing no torque. Another symptom of this is that the current being drawn from the power supply is very low.

Avoid Destroying the Driver

Many users have first ask why the device failed in the application. In almost every case the failure was caused by electrical overstress to the device, specifically voltages or currents that are outside of the device ratings. Whenever a driver fails, a careful evaluation of the operating conditions in the application is in order.

The most common failure encountered is the result of voltage transients generated by the inductance in the motor. A correctly designed application will keep the peak voltage on the power supply, across the collector to emitter of the output devices and, for monolithic drivers, from one output to the other within the maximum rating of the device. A proper design includes power supply filtering and clamp diodes and/or snubber networks on the output [6].

Selecting the correct clamp diodes for the application is essential. The proper diode is matched to the speed of the switching device and maintains a V_F that limits the peak voltage within the allowable limits. When the diodes are not integrated they must be provided externally. The diodes should have switching characteristics that are the same or better than the switching time of the output transistors. Usually diodes that have a reverse recovery time of less than 150 ns are sufficient when used with bipolar output devices like the L298N. The 1N4001 series of devices, for example, **is not** a good selection because it is a slow diode.

Although it occurs less frequently, excess current can also destroy the device. In most applications the excess current is the result of short circuits in the load. If the application is pron to have shorted loads the designer may consider implementing some external short circuit protection [7].

Shoot through current, the current that flows from supply to ground due to the simultaneous conduction of upper and lower transistors in the bridge output, is another concern. The design of the L298N, L293 and L6202 all include circuitry specifically to prevent this phenomena. The user should not mistake the reverse recovery current of the diodes or the parasitic structures in the output stage as shoot through current.

SELECTED REFERENCES

[1]Sax, Herbert., "Stepper Motor Driving" (AN235)

[2]"Constant Current Chopper Drive Ups Stepper-Motor Performance" (AN468)

[3]Hopkins, Thomas. "Unsing the L6506 for Current Control of Stepping Motors" (AN469)

[4]"The L297 Steper Motor Controller" (AN470)

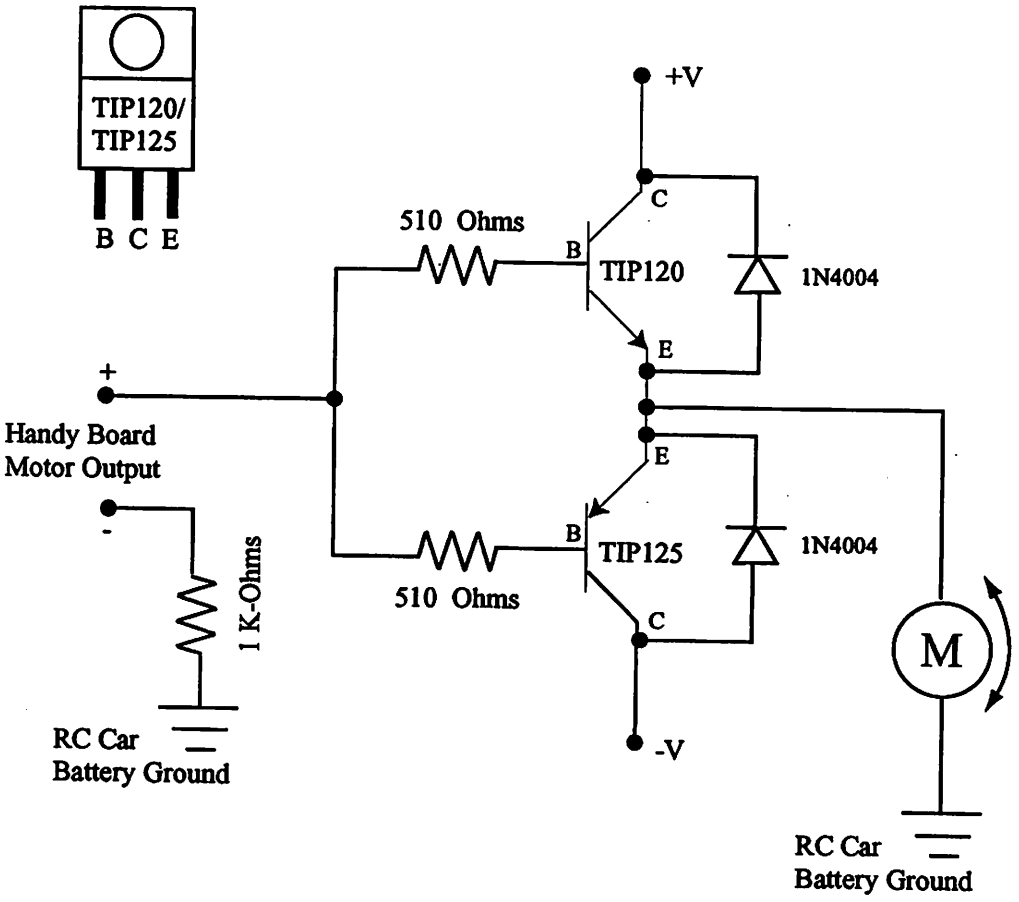
[5]Leenouts, Albert. The Art and Practice of Step Motor Control. Ventura CA: Intertec Communications Inc. (805) 658-0933. 1987

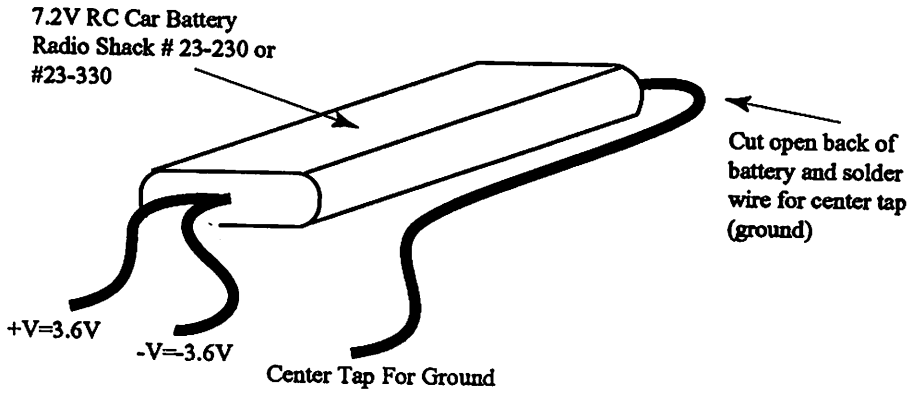
[6]Hopkins, Thomas. "Controlling Voltage Transisnts in Full Bridge Drivers" (AN280)

[7]Scrocchi G. and Fusaroli G. "Short Circuit Protection on L6203". (AN279)

Bipolar "Push-Pull" Motor Driver Circuit for Handy Board Microcontroller

Kam Leang 1998 kleang@eng.utah.edu





EVERLIGHT ELECTRONICS CO., LTD.

DEVICE NUMBER : DIH-033-001 REV : 2.0
ECN : PAGE : 1/8

ım Infrared LED

)DEL NO : HIR333

Features :

- High radiant intensity
- Peak wavelength λp=850nm
- View angle 17°
- High reliability
- 2.54mm Lead spacing

Description :

- EVERLIGHT's Infrared Emitting Diode (HIR333) is a high intensity diode, molded in a yellow transparent plastic package.
  The device is spectrally matched with phototransistor, photodiode and infrared receiver module.

Applications :

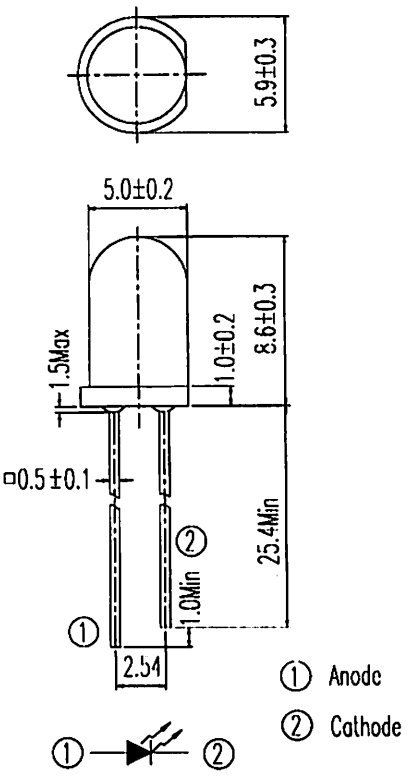
- Free air transmission system
- Optoelectronic switch
- Infrared remote control units with high power requirement
- Floppy disk drive
- Smoke detector

PART NO.	CHIP MATERIAL	LENS COLOR
HIR	GaAlAs	Yellow

Infrared LED

L NO : HIR333

ackage Dimensions :



lotes :

imensions are in millimeter.
uded resin under flange 1.5 mm Max.
l spacing is measured where the lead emerge from the package.
color : Yellow transparent.
/e specification may be changed without notice. EVERLIGHT will reserve authority
naterial change for above specification.
se specification sheets include materials protected under copyright of EVERLIGHT
oration . Please don't reproduce or cause anyone to reproduce them without
:RLIGHT's consent.
n using this product , please observe the absolute maximum ratings and the
uctions for use outlined in these specification sheets. EVERIGHT assumes no
onsibility for any damage resulting from use of the product which does not comply
the absolute maximum ratings and the instructions included in these specification
ets.

m Infrared LED

DEL NO : HIR333

■ Absolute Maximum Ratings at T_A = 25℃

Parameter	Symbol	Rating	Unit	Notice
ontinuous Forward Current	I_F	50	mA	
eak Forward Current ulse width=100 μs,Duty cycle=1%	I_{FP}	1.0	A	
everse Voltage	V_R	5	V	
perating Temperature	Topr	-25 ~ +85	℃	
torage Temperature	Tstg	-40 ~ +85	℃	
oldering Temperature	Tsol	260	℃	4mm from mold body less than 5 seconds
ower Dissipation at(or below) 5℃ Free Air Temperature	Pd	100	mW	

■ Electronic Optical Characteristics :

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
adiant Intensity	Ee	7.8	15.0	----	mW/sr	I_F=20mA
		----	140	----		I_F=100mA,tp=100 μs, t_p/T=0.01
		----	980	----		I_F=1A,tp=100 μs, t_p/T=0.01
eak Wavelength	λ_P	----	850	----	nm	I_F=20mA
pectral andwidth	Δλ	---	45	---	nm	I_F=20mA
orward Voltage	V_F	---	1.45	1.65	V	I_F=20mA
		---	1.80	2.40		I_F=100mA,tp=100 μs, t_p/T=0.01
		---	4.10	5.25		I_F=1A,tp=100 μs, t_p/T=0.01
everse Current	I_R	---	---	10	μA	V_R=5V
ew Angle	2Θ1/2	----	17	----	deg	I_F=20mA

ım Infrared LED

)DEL NO : HIR333

Typical Electrical/Optical/Characteristics Curves

Fig. 1 Forward Current vs. Ambient Temperature

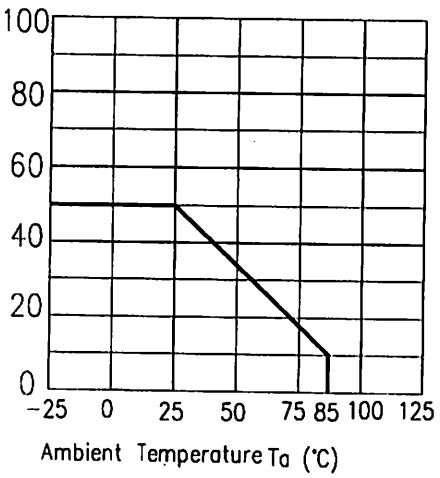


Fig. 2 Spectral Distribution

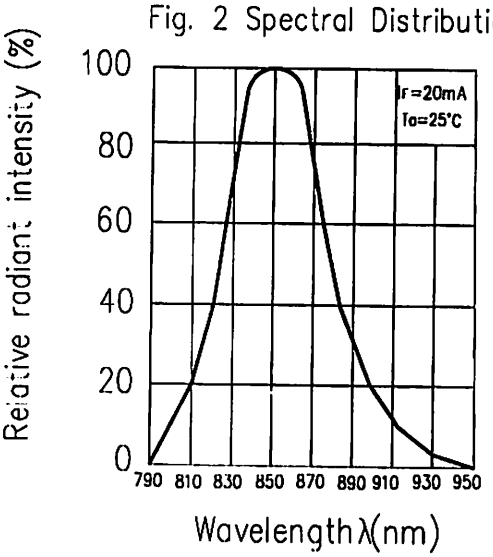


Fig. 3 Peak Emission Wavelength vs. Ambient Temperature

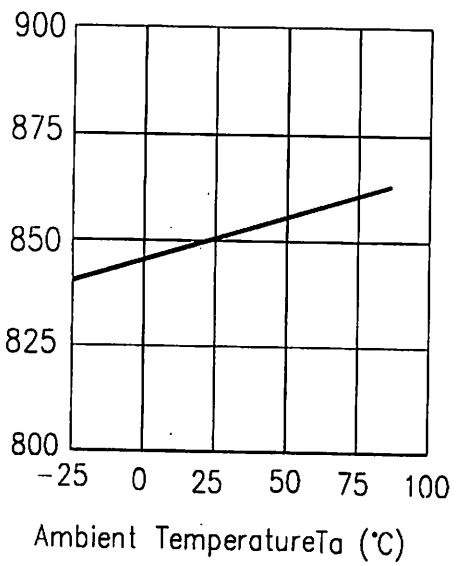
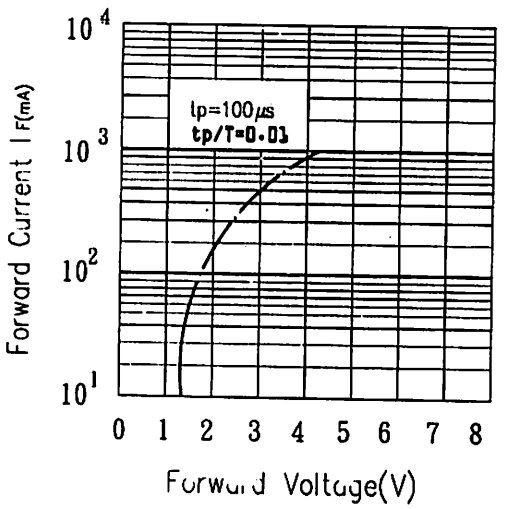


Fig. 4 Forward Current vs. Forward Voltage



m Infrared LED

DEL NO : HIR333

Typical Electrical/Optical/Characteristics Curves

Fig. 5 Relative Intensity vs. Forward Current

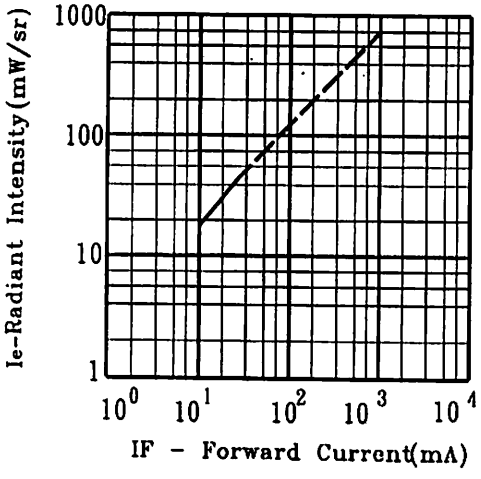


Fig. 6 Relative Radiant Intensity vs. Angular Displacement

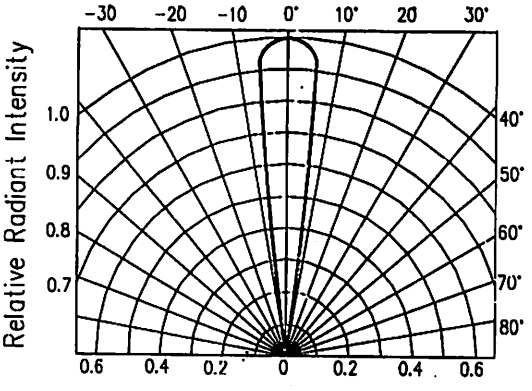


Fig. 7 Relative Intensity vs. Ambient Temperature (℃)

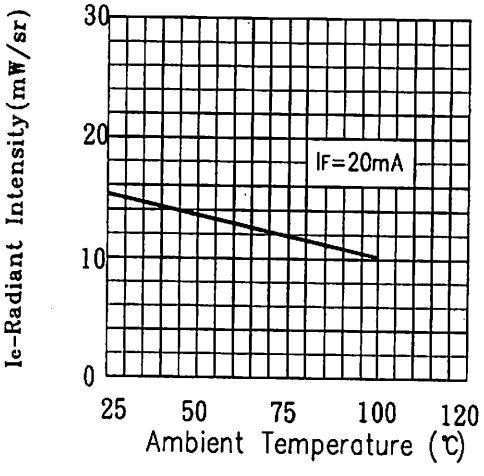
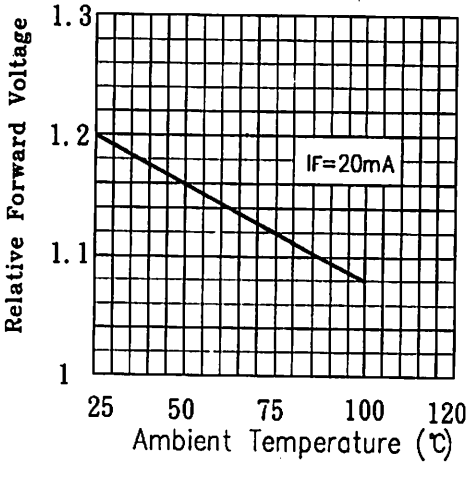


Fig. 8 Forward Current vs. Ambient Temperature (℃)



m Infrared LED

DEL NO : HIR333

■ Reliability Test Item And Condition

The reliability of products shall be satisfied with items listed below.
Confidence level:90%
LTPD:10%

NO.	Item	Test Conditions	Test Hours/ Cycles	Sample Size	Failure Judgement Criteria	Ac/Re
1	**Solder Heat**	TEMP : 260℃ ±5 ℃	5 secs	22 pcs		0/1
2	**Temperature Cycle**	H : +85℃ 30 mins ↕ 5 mins L : -55℃ 30 mins	50 cycles	22 pcs	$I_R \geqq Ux\ 2$ $Ee \leqq Lx\ 0.8$ $V_F \geqq Ux\ 1.2$	0/1
3	**Thermal Shock**	H : +100℃ 5 mins ↕ 10 secs L : -10℃ 5 mins	50 cycles	22 pcs	U :Upper specification limit L :Lower specification limit	0/1
4	**High Temperature Storage**	TEMP. : +100℃	1000 hrs	22 pcs		0/1
5	**Low Temperature Storage**	TEMP. : -55℃	1000 hrs	22 pcs		0/1
6	**DC Operating Life**	I_F=20mA	1000 hrs	22 pcs		0/1
7	**High Temperature / High Humidity**	85℃ / 85% R.H.	1000 hrs	22 pcs		0/1

nm Infrared LED

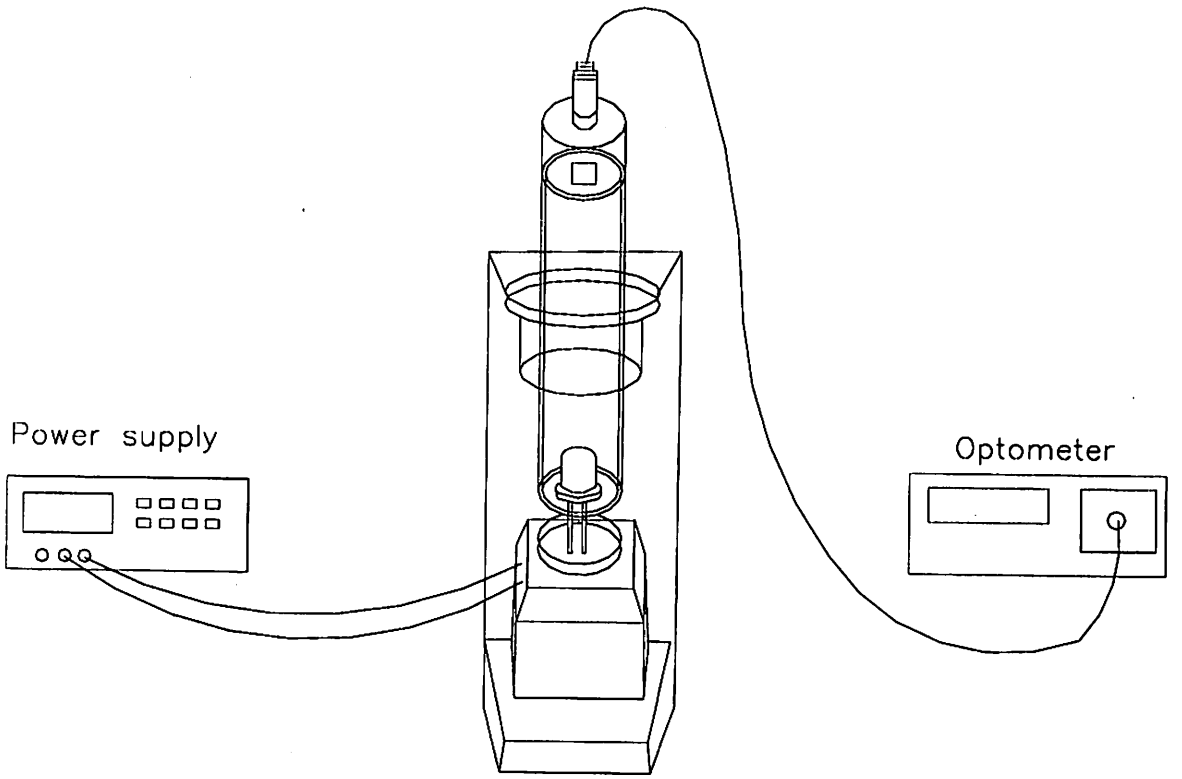
ODEL NO : HIR333

Test Method For Power :

Condition : I_F=20 mA

Test Item : Radiant Intensity

Unit : mW/sr



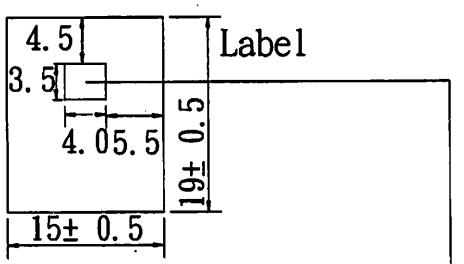
Bin Number	M	N	P	Q	R
Min	7.8	11.0	15.0	21.0	30
Max	12.5	17.6	24.0	34.0	48

ım Infrared LED

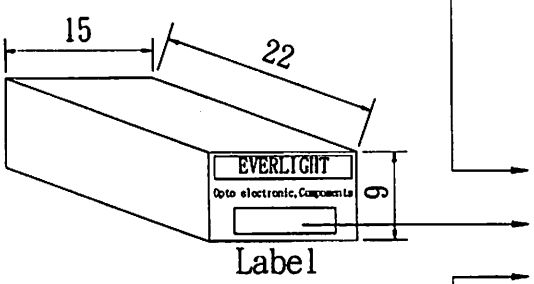
)DEL NO : HIR333

Packing Specifications

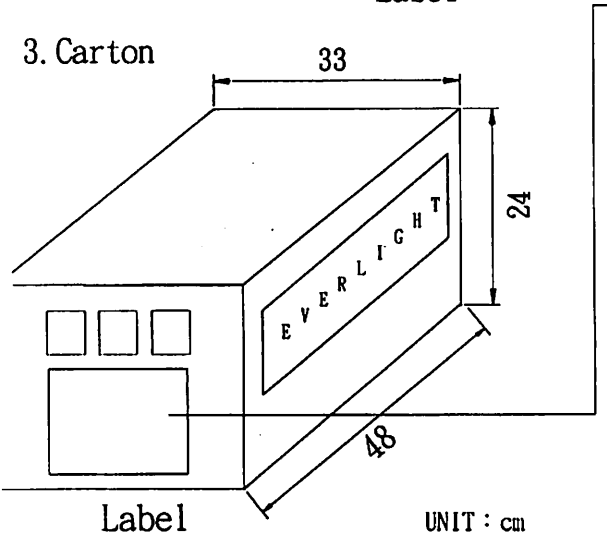
1. Bag



2. Box



3. Carton



EVERLIGHT

CPN:

P/N: 3403830003

IR383

QTY: 500 CAT:

HUE:

REF:

LOT NO:

MADE IN TAIWAN

CPN : Customer's Production Number
P/N : Production Number
QTY : Packing Quantity
CAT : Ranks
HUE : Peak Wavelength
REF : Reference
LOT NO : Lot Number
MADE IN TAIWAN : Production place

Packing Quantity Specification

500Pcs/1Bag , 6 Bags/1Box
10 Boxes/1Carton