SORTING OF RED, YELLOW AND BLUE
COLORED OBJECTS USING AT89C51 MICROCONTROLLER

(PEMISAH BENDA BERWARNA MERAH, KUNING DAN BIRU
BERBASIS MIKROCONTROLLER AT89C51)

TUGAS AKHIR
Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
Jurusan Teknik Elektro

Disusun Oleh:
STEFANUS HERI TRIYANTO
NIM : 965114061
NIRM : 960051123107120060

PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS SANATA DHARMA
YOGYAKARTA
2004
TUGAS AKHIR

SORTING OF RED, YELLOW AND BLUE
COLORED OBJECTS USING AT89C51 MICROCONTROLLER
(PEMISAH BENDA BERWARNA MERAH, KUNING DAN BIRU
BERBASIS MIKROCONTROLER AT89C51)

Disusun oleh:
STEFANUS HERI TRIYANTO
NIM : 965114061
PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK

Telah disetujui oleh:

Pembimbing I

\[\text{J\text{\textsuperscript{\textregistered}}\text{\textcopyright}}\]
B. DJOKO UNTORO, SSi., MT

Tanggal 11 Januari 2005

Pembimbing II

\[\text{J\text{\textsuperscript{\textregistered}}\text{\textcopyright}}\]
Ir. HENDRO

Tanggal 4 Januari 2005
TUGAS AKHIR

SORTING OF RED, YELLOW AND BLUE
COLORED OBJECTS USING AT89C51 MICROCONTROLLER
(PEMISAH BENDA BERWARNA MERAH, KUNING DAN BIRU
BERBASIS MIKROCONTROLLER AT89C51)

Disusun oleh :
STEFANUS HERI TRIYANTO
NIM : 965114061

Telah dipertahankan di depan panitia pengujian pada tanggal : 5 Nopember 2004
dan dinyatakan memenuhi syarat

Susunan panitia penguji :

Nama lengkap
Ketua : B. DJOKO UNTORO SUWARNO, S.Si.,M.T.
Sekretaris : Ir. T.JENDRO
Anggota : Ir. ISWANJONO, M.T.
Anggota : Ir. TH. PRIMA ARI SETIYANI, M.T.

Tanda tangan :

Yogyakarta, Desember 2004
Fakultas Teknik
Universitas Sanata Dharma

Dekan Fakultas Teknik

Ir. Greg. Heliarko SJ.,SS.,BST.,MA.,MSc
PERNYATAAN KEASLIAN KARYA

Saya menyatakan dengan sesungguhnya bahwa skripsi yang saya tulis ini tidak memuat karya orang lain, kecuali yang telah disebutkan dalam kutipan dan daftar pustaka sebagaimana layaknya karya ilmiah.

Yogyakarta, Nopember 2004

Penulis

Stefanus Heri Triyanto
PERSEMBAHAN

"YESUS KRISTUS"

Karya kecil ini kupersembahkan kepada:

- Bapak, Ibu, kakakku yang ada
dirumah.

- Istriku dan Anakku yang tercinta.
KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa yang telah melimpahkan rahmatNya sehingga penulis berhasil menyelesaikan tulisan Pemisah Benda Berwarna Merah, Kuning, dan Biru ini.

Penulis mengucapkan terima kasih kepada:


2. Bapak Ir. Tjendro, selaku pembimbing II Tugas Akhir, yang telah membantu dan membimbing dalam pembuatan tugas akhir ini.

3. Seluruh dosen dan karyawan Fakultas Teknik Universitas Sanata Dharma yang telah membantu dalam pembuatan Tugas Akhir ini.

4. Buat laboran laboratorium fakultas teknik lab. kendali, lab. elektronika dasar dan lab. perancangan yang telah membantu dalam pengambilan data untuk tugas akhir ini.

5. Bapak, Ibu, Kakak, Istri ku dan anakku tercinta Sekar, yang telah memberikan dukungan material dan spiritual.


7. Rekan-rekan yang tidak bisa penulis sebut satu persatu.
Penulis berharap tulisan ini dapat memberi manfaat bagi pembaca, namun penulis menyadari bahwa tulisan ini masih jauh dari sempurna, untuk itu penulis akan menerima dengan senang hati kritik dan saran yang diberikan untuk perbaikan dan penyempurnaannya. Harapan penulis semoga dimasa yang akan datang dapat tercipta karya yang lebih baik.

Yogyakarta, Nopember 2004

Penulis

[Signature]

Stefanus Heri Triyanto
INTISARI

Proses pemisahan benda banyak dijumpai dalam proses di industri. Pada penelitian ini akan dibahas tentang model (miniatur) proses pemisah benda berdasarkan warna benda menggunakan kontroler berupa mikrokontroler.


Dari hasil pengamatan diperoleh sistem dapat bekerja memisahkan benda berdasarkan warna benda. Sistem bekerja memisahkan benda satu per satu.
SORTING OF RED, YELLOW AND BLUE
COLORED OBJECTS USING AT89C51 MICROCONTROLLER

By
Stefanus Heri Triyanto

ABSTRACT

Objects Sorting and selection are commonly found at industrial processes. This paper discussed about a model of objects sorting and objects selection process of colored object using ATMEL AT 89C51 microcontroller.

In this system, object selection based on its color: red yellow and blue object. The object color is made by colored marble paper. Detection of object color uses LDR while conveyor is used to carry an object tray bin to selected colored object tray bin. The object selector uses stepper motor as actuator.

The result is system able to sort and select the colored objects. Object selection process is done one by one object selection.
DAFTAR ISI

HALAMAN JUDUL ......................................................... i
LEMBAR PERSETUJUAN ................................................ ii
LEMBAR PENGESAHAN ................................................. iii
LEMBAR PERNYATAAN KEASLIAN KARYA ......................... iv
PERSEMBAHAN ........................................................... v
KATA PENGANTAR ..................................................... vi
INTISARI ...................................................................... viii
ABSTRACT ..................................................................... ix
DAFTAR ISI ................................................................... x
DAFTAR GAMBAR ........................................................ xiii
DAFTAR TABEL ............................................................. xv
DAFTAR LAMPIRAN ....................................................... xvi

BAB I PENDAHULUAN

1.1 Latar Belakang ......................................................... 1
1.2 Manfaat ................................................................. 2
1.3 Tujuan .................................................................... 2
1.4 Langkah Penelitian .................................................. 2
1.5 Batasan Masalah ...................................................... 3
1.6 Sistematika Penulisan .............................................. 4

BAB II DASAR TEORI

2.1 Mikrokontroler AT89C51 .......................................... 6
BAB III PERANCANGAN

3.1 Perancangan Perangkat keras ........................................... 24

3.1.1 Pemisah Benda Berwarna Merah, Kuning, Biru .................. 24
3.1.2 Bentuk Benda ............................................................ 28
3.1.3 Mikrokontroler AT89C51 ......................................... 28
3.1.4 Rangkaian Pembeda Warna ...................................... 29
3.1.5 Sensor Keberadaan Benda ......................................... 33
3.1.6 Penggerak Motor Stepper ........................................ 34
3.1.7 Sensor Akhir ......................................................... 36
3.1.8 Penggerak Motor DC ............................................. 38
3.1.9 Limit Switch ....................................................... 40
3.2 Diagram Alir ........................................................... 41

BAB IV PENGAMATAN DAN PEMBAHASAN

4.1 Deteksi Keberadaan dan Warna Benda ................................. 45
4.2 Pemrograman gerak motor Stepper ..................................... 47
4.3 Waktu Tunda .................................................................. 48
4.4 Pemrograman motor DC .................................................... 49
4.5 Respon tegangan pada i.DR .............................................. 49
4.6 Driver motor .................................................................. 51
4.7 Data hasil pengamatan ...................................................... 51
   4.7.1 Waktu pemrosesan benda ......................................... 51
   4.7.2 Kecepatan benda ..................................................... 52
   4.7.3 Waktu pintu pengarah .............................................. 53

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan ................................................................... 55
5.2 Saran ............................................................................. 55

DAFTAR PUSTAKA .................................................................. 57
LAMPIRAN ........................................................................... 58
DAFTAR GAMBAR

Gambar 2.1 Memori program dalam mikrokontroler AT89C51 .................... 7
Gambar 2.2 Memori data internal .................................................. 8
Gambar 2.3 Register Fungsi Khusus (Special Function Register) .......... 10
Gambar 2.4 Register Program Status Word (PSW) ............................ 12
Gambar 2.5 Op-Amp sebagai pembanding ....................................... 15
Gambar 2.6 Pembanding dengan titik pergeseran yang dapat diatur .... 17
Gambar 2.7 Simbol Light Dependent Resistor (LDR) ......................... 18
Gambar 2.8 Rangkaian Dasar Transistor sebagai Saklar ..................... 18
Gambar 2.9 Eksitasi pada motor stepper .......................................... 20
Gambar 2.10 Simbol penyanga (buffer) ............................................. 22
Gambar 2.11 Simbol Foto Transistor .............................................. 22
Gambar 2.12 Skema motor DC ....................................................... 23
Gambar 2.13 Simbol motor DC ....................................................... 23
Gambar 3.1 Diagram Blok pemisah benda berwarna mera, kuning, biru 24
Gambar 3.2 Diagram kotak lokasi komponen pendukung ................... 26
Gambar 3.3 Bentuk fisik pemisah benda berwarna ............................ 27
Gambar 3.4 Bentuk benda ............................................................ 28
Gambar 3.5 Rangkaian sensor pembeda warna ................................. 30
Gambar 3.6 Rangkaian pembanding LM324 ..................................... 32
Gambar 3.7 Rangkaian keberadaan benda ....................................... 33
Gambar 3.8 Rangkaian driver motor stepper ........................................... 35
Gambar 3.9 Rangkaian inframerah dan photo transistor ................................ 36
Gambar 3.10 Rangkaian lengkap sensor akhir ........................................... 38
Gambar 3.11 Rangkaian penggerak motor DC ............................................ 39
Gambar 3.12 Rangkaian limit switch ......................................................... 40
Gambar 3.13 Diagram alir pemisah benda berwarna ................................... 42
Gambar 4.1 Hasil pengukuran kecepatan dengan osiloskop digital ............... 54
DAFTAR TABEL

Tabel 2.1 Urutan perlakuan eksitasi tunggal .......................................................... 21
Tabel 2.2 Urutan perlakuan eksitasi ganda ............................................................ 21
Tabel 3.1 Tegangan LDR ..................................................................................... 30
Tabel 3.2 Masukan sensor warna .......................................................................... 33
Tabel 4.1 Respon tegangan LDR .......................................................................... 50
Tabel 4.2 Data arus motor stepper ........................................................................ 51
Tabel 4.3 Waktu pemrosesan benda .................................................................... 52
Tabel 4.4 kecepatan motor stepper ....................................................................... 53
DAFTAR LAMPIRAN

LAMPIRAN A  Rangakaian pemisah benda berwarna ......................... L.1
LAMPIRAN B  Warna dan jenis kertas yang digunakan ......................... L.2
LAMPIRAN C  Program pemisah benda berwarna ................................ L.3
LAMPIRAN D  Data sheet mikrokontroler AT89C51 .............................. L.7
LAMPIRAN E  Data sheet IC Buffer 74LS541 ..................................... L.25
LAMPIRAN F  Data sheet IC Driver ULN2803 ..................................... L.28
LAMPIRAN G  Data sheet Transistor BD677 ..................................... L.36

xvi
BAB I
PENDAHULUAN

1.1 Latar Belakang

Dalam perkembangan teknologi elektronika yang semakin maju, banyak di tawarkan berbagai macam alat elektronika yang mampu menjalankan program yang dibuat oleh manusia. Dalam dunia industri banyak dibutuhkan komponen-komponen yang bisa diterapkan dalam rangka otomatisasi industri.

Salah satunya adalah mikrokontroler, yaitu sebuah komponen elektronik yang dapat bekerja sesuai dengan program yang diisikan ke dalam memorinya seperti layaknya sebuah komputer sederhana.

Mikrokontroler AT89C51, yaitu mikrokontroler dengan arsitektur MCS51 produksi Atmel yang mempunyai sistem memori, timer, port serial dan 32 bit I/O di dalamnya sehingga sangat memungkinkan untuk membentuk suatu sistem yang hanya terdiri atas single chip (keping tunggal) saja. Mikrokontroler ini mempunyai memori dengan teknologi nonvolatile memory, isi memori tersebut dapat diisi ulang ataupun dihapus berkali-kali, sehingga memudahkan kita dalam pembuatan suatu sistem sesuai dengan keinginan pembuatnya.

Mikrokontroler dirancang dapat mengendalikan mesin dan proses dengan mengimplementasikan fungsi nalar, kendali urutan (sequential), operasi pewaktuan (timming), pencacah (counting), dan aritmatika. Mikrokontroler mampu menjalankan operasi aritmatika yang komplek, mengendalikan motor DC, motor stepper, dan masih banyak aplikasi dari mikrokontroler.
Secara umum mikrokontroler sangat banyak membantu dalam pembuatan suatu sistem dalam industri modern yang memerlukan kecepatan dan kerepatan dalam memproduksi barang. Dalam tugas akhir ini merupakan salah satu dari penerapan mikrokontroler AT89C51.

1.2 Manfaat

Pembaca dapat mengetahui dan menerapkan penggunaan mikrokontroler AT89C51 untuk memilih benda berwarna merah, kuning, atau biru.

1.3 Tujuan

1. Merancang suatu rangkaian elektronika yang bisa digunakan untuk membedakan warna merah, kuning, dan biru.
2. Membuat model mekanik yang digunakan untuk memilih benda berwarna merah, kuning, dan biru.
3. Membuat program dengan diagram alir (flow chart) untuk mengendalikan mikrokontroler AT89C51 agar dapat memilih benda berwarna merah, kuning, dan biru.
4. Memprogram mikrokontroler AT89C51.

1.4 Langkah Penelitian

1. Meneliti pengaruh intensitas cahaya, yang dipantulkan oleh benda berwarna merah, kuning dan biru terhadap Light Dependent Resistor (LDR).
2. Membuat rangkaian elektronis untuk membedakan benda berwarna merah, kuning, dan biru.

3. Membuat rangkaian deteksi benda.

4. Membuat rangkaian sensor akhir.

5. Membuat model mekanik pemilih benda berwarna.

6. Membuat program untuk mengendalikan mikrokontroler AT89C51.

7. Menguji alat.

8. Penulisan laporan.

1.5 Batasan Masalah

Pada tugas akhir ini, mikrokontroler yang digunakan adalah AT89C51, sedangkan benda yang dipisahkan terdiri tiga warna yaitu: benda berwarna merah, kuning, biru. Jenis warna yang digunakan adalah warna dari kertas marmer sehingga dalam pemisah benda berwarna ini benda yang dideteksi harus menggunakan warna dan jenis kertas yang sama. Bentuk benda yang dideteksi berbentuk tabung dengan diameter 4,5 cm dan tinggi 4 cm. Dalam pemisahan benda menggunakan konveyor dan sebagai pengarah benda menggunakan dua motor stepper. Sebagai penggerak konveyor menggunakan motor DC.

Dalam prosesnya alat ini dirancang untuk dapat mendeteksi warna dan mengarahkan benda secara satu-persatu atau dengan kata lain dalam satu kali proses hanya satu benda saja. Jadi untuk benda selanjutnya menunggu sampai benda yang pertama selesai diproses.
Peralatan mekanik yang digunakan untuk memisahkan benda berwarna tersebut bukanlah merupakan sebuah prototipe, melainkan hanya model, sehingga dalam pembuatannya tidak memakai perhitungan-perhitungan yang presisi.

1.6 Sistematika penulisan

Penulisan proposal tugas akhir ini disusun dalam lima bab, yang meliputi:

1. BAB I Pendahuluan

   Pendahuluan berisi tentang latar belakang masalah, tujuan dan manfaat dari penelitian, batasan masalah, dan sistematika penulisan.

2. BAB II, Dasar Teori

   Berisi tentang dasar teori yang berkaitan dengan judul yang dibuat yaitu pemisah benda berwarna merah, kuning dan biru.

3. BAB III, Rancangan Penelitian

   Berisi tentang rancangan perangkat keras dan perangkat lunak dan diagram alir program.

4. BAB IV, Pengamatan dan Pembahasan hasil

   Berisi tentang pembahasan perangkat lunak dan pengamatan dari pembuatan alat pemisah benda berwarna.

5. BAB V, Kesimpulan dan Saran

   Berisi tentang kesimpulan dan saran yang merupakan penutup dari tulisan tentang pemisah benda berwarna merah, kuning dan biru.
BAB II

DASAR TEORI

Pemisah benda Merah, Kuning, dan Biru yang dirancang merupakan alat yang dapat membedakan warna serta memisahkan benda dalam tempat yang telah disediakan. Dalam membedakan warna tersebut melalui sensor yang menggunakan Light Dependent Resistor (LDR). Sumber cahaya menggunakan LED berwarna putih kemudian cahaya dipantulkan oleh benda berwarna yang kemudian diterima oleh LDR.

Perubahan nilai resistensi LDR yang terjadi karena perbedaan intensitas cahaya yang diterima, akan diubah menjadi perubahan arus tegangan. Perubahan arus tegangan masuk dalam pembanding yang dirancang apabila tegangan dari LDR lebih kecil dari tegangan referensi maka keluaran dari pembanding akan memberi masukan pada mikrokontroler dalam dua keadaan, yaitu arus high (5 volt) atau ‘1’ dan low ‘0’. Pemisah benda yang dideteksi menggunakan konveyor yang digerakkan oleh motor reduksi. Konveyor mempunyai tiga sensor yaitu sensor keberadaan benda yang terletak di bagian depan konveyor, sensor warna yang diletakkan setelah sensor keberadaan benda, dan sensor akhir yang terletak di bagian akhir dari konveyor. Penutup yang berfungsi mengerahkan benda agar menuju tempat masing-masing digerakkan dengan sebuah motor stepper.
2.1. Mikrokontroler AT89C51

Mikrokontroler merupakan sistem komputer dengan spesifikasi tertentu, yang mempunyai kemampuan diprogram untuk mengoperasikan data dan kemampuan menyimpan serta mengambil data. Sistem komputer tersebut terdiri dari CPU (Central Processing Unit) yang berhubungan dengan RAM (Random Access Memory) dan ROM (Read Only Memory) melalui bus data, bus alamat dan bus kendali.

AT89C51 mempunyai komponen- komponen tersebut dengan 4K byte Flash PEROM (Programmable and Erasable Read Only Memory),

AT89C51 merupakan memori dengan teknologi nonvolatile memory, isi memory tersebut dapat diisi ulang ataupun dihapus berkali-kali.

Memori ini biasa digunakan untuk menyimpan perintah berstandar MCS-51 code sehingga memungkinkan mikrokontroler ini untuk bekerja dalam mode operasi keping tunggal (single chip operation) yang tidak memerlukan memori luar untuk menyimpan source code tersebut.

2.1.1. Fitur yang dimiliki AT89C51

Fasilitas yang dimiliki mikrokontroler ini antara lain 4 Kbyte ROM, 128 Byte RAM, 4 buah i/O Port masing-masing 8 bit, 2 buah timer 16 bit, Serial Interface, 64 Kbyte Ekternal Data Memory Spaces, Boolean procesor (pada operasi bit), dan 210 lokasi yang dapat dialamati per bit. Diagram blok bagian-bagian mikrokontroler AT89C51 dan hubungan antar bagian tersebut secara lebih jelas dapat dilihat pada lampiran data sheet.
2.1.2. Memori Program

Dalam mikrokontroler terdapat memori program dan memori data yang terpisah. Memori program atau ROM digunakan untuk menyimpan program yang dibuat oleh pemrogram. Isi ROM dapat diubah oleh pemrogram dan tidak akan hilang jika mikrokontroler tidak terhubung ke catu daya. ROM menempati ruang dengan nomor heksadesimal (selanjutnya akan ditulis dengan akhiran -h). Ruang sisanya digunakan sebagai memori eksternal dengan kapasitas totalnya 64 Kbyte.

Gambar 2.1 Memori program dalam mikrokontroler AT89C51

2.1.3. Memori Data

RAM *(Random Access Memory)* merupakan memori yang dapat dibaca dan ditulis. Memori data disimpan dalam RAM tersebut yang dipakai sebagai
penyimpanan data pada saat program bekerja. Isi RAM akan hilang bila catu daya mati, karena sifat RAM yang *volatile*.

Mikrokontroler AT89C51 memiliki data internal 128 byte yang biasanya digunakan untuk menyimpan variabel atau data yang bersifat sementara.

Memori internal mempunyai alamat dari 00h sampai FFh, yang terbagi menjadi dua bagian yaitu alamat 00h sampai 7Fh seperti RAM selayaknya, sedangkan memori 80h sampai FFh dipakai sebagai SFR (*Special Function Register*). Jadi hanya setengah dari RAM yang dapat dipakai sebagai memori data.

Pembagian daerah memori ditunjukan oleh tabel berikut:

<table>
<thead>
<tr>
<th>Alamat</th>
<th>Bank Register</th>
<th>0D-FFh Alamat 7Fh (Dapat dialami secara bit 00-0Fh)</th>
<th>40-7Fh Alamat 7Fh (Dapat dialami secara bit 40-7Fh)</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h-7Fh</td>
<td>Bank Register 0</td>
<td>08 10 18 20 26 30 38 3E 20</td>
<td></td>
</tr>
<tr>
<td>80h-FFh</td>
<td>Bank Register 3</td>
<td>4C 48 50 58 60 68 70 78 2F</td>
<td></td>
</tr>
</tbody>
</table>

Register Seba Guna
(Multifunction Register)

Register Fungsi Khusus
(Special Function Register)

Gambar 2.2 Memori data internal
Memori data 00H sampai 7FH bisa dipakai sebagai memori penyimpan data biasa, dibagi menjadi 3 bagian:


2. Memori yang dapat dialamati secara bit, yaitu antara 20h sampai 2Fh. Pada daerah ini dapat dialamati secara byte (20h sampai 2Fh) atau dialamati secara bit (00h sampai 7Fh). Pengalamatan bit dapat dilakukan dengan memanggil gabungan byte dan bit (misal 20.1, 20.1 dsb) atau langsung ke alamat bit (misal 00, 01 dst).

3. Register Serba Guna (Multi Function Register). Memori 30h sampai 7Fh merupakan memori data biasa.

4. Register Fungsi Khusus (Special Function Register). Memori antara 80 sampai dengan FF digunakan untuk alamat register-register khusus, seperti ditunjukkan oleh gambar 2.3.
Gambar 2.3 Register Fungsi Khusus (Special Function Register)

2.1.4. Special Function Register (SFR)

Register Fungsi Khusus adalah bagian dari RAM yang dipakai untuk mengatur perilaku mikrokontroler yang berisi register dasar, register data input/output, dan register status. Sebagian register dasar diletakkan dalam SFR, seperti accumulator dan register B. Register-register dalam SFR antara lain:

a. Register penampung data masukan/keluaran yang berhubungan dengan port pararel P0, P1, P2, dan P3.

b. TL0/TH0 (Timer 0 Low/High), TL1/TH1 (Timer 1 Low/High) yang membentuk Timer 0 dan Timer 1 sebagai pencacah naik, dan juga bisa dipakai sebagai sumber clock pencacah. Perilaku timer ini diatur melalui register TMOD dan TCON.
c. IE (Interrupt Enable), untuk mengatur agar interupsi aktif atau tidak aktif.

d. IP (Interrupt Priority), digunakan untuk mengatur prioritas dari masing-masing sumber interupsi.

e. PCON (Power Control), untuk mengatur mode pemakaian daya oleh mikrokontroler misalnya pada saat sistem elektronik yang dikendalikannya tidak bekerja tetapi dalam keadaan stand by. Daya listrik yang digunakan sistem dapat direduksi sehingga menjadi hemat, terutama sistem yang menggunakan baterai sebagai sumber daya listrik.

f. SCON (Serial Control), sebagai register yang mengatur proses pengiriman dan penerimaan data serial ke TxD/P3.0 (pin 11) dan dari RxD/P3.1 (pin10).

g. SBUF (Serial Buffer), berhubungan dengan TxD/RxD dalam pengiriman dan penerimaan data serial.

2.1.5 Register Dasar

Yang termasuk register dasar adalah program counter (PC), accumulator (A), stack pointer (SP), program status word (PSW). Register khas dari keluarga mikrokontroler MCS51 adalah register B, dan data pointer register (DPTR).

a. PC (Program Counter)

Merupakan register 16 bit yang berisi alamat yang akan dikerjakan. Saat reset PC bernilai 0000h, nilai PC akan bertambah 1 setelah prosesor mengambil intruksi 1 byte.
b. A (Accumulator)

Sesuai dengan namanya *accumulate* yang artinya menampung, register A berfungsi untuk menampung.

c. SP (Stack Pointer)

Digunakan sebagai penyimpan sementara nilai PC sebelum prosesor menjalankan sub-rutin. Saat prosesor selesai mengerjakan sub-rutin nilai PC akan dikembalikan dengan cara mengambil dari SP.

d. PSW (Program Status Word)

Merupakan register yang menginformasikan status atau keadaan CPU, yang terdiri dari beberapa bit status seperti diunjukkan pada gambar 2.4

```
    CY  AC  F0  RS1  RS0  OV  -  P
  bit 7                   bit 0
```

_Gambar 2.4 Register Program Status Word (PSW)_

<table>
<thead>
<tr>
<th>CY</th>
<th>AC</th>
<th>F0</th>
<th>RS1</th>
<th>RS0</th>
<th>OV</th>
<th>-</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>bit 7</td>
<td>bit 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **CY**: sebagai bit *carry flag* dalam operasi aritmatika dari ALU
- **AC**: *auxiliary carry flag* dalam operasi penambahan
- **F0**: flag 0, terjadi bagi pengguna untuk keperluan umum
- **RS1, RS0**: bit untuk memilih register bank yang digunakan
- **OV**: bit tanda jika terjadi overflow dalam operasi aritmatika
- **P**: bit yang menyatakan jumlah '1' dalam *accumulator*

e. Register B
Register 8 bit yang tugasnya bersama akumulator untuk proses aritmatik selain dapat juga difungsikan sebagai register biasa.

f. DPTR (Data Pointer Register)

Register 16 bit, terbagi menjadi data pointer high byte (DPH) dan data pointer low byte (DPL) yang masing-masing 8 bit. Fungsi register ini untuk mengalami data yang lebih luas karena besarnya 16 bit.

2.2 Pemrograman Mikrokontroler AT89C51

Semua mikrokontroler yang termasuk keluarga MCS 51 menggunakan bahasa pemrograman yang sama yaitu MCS 51 assembly language, kumpulan instruksi berupa kode-kode dengan panjang satu sampai empat byte setiap instruksi. Pemrogramannya disesuaikan dengan kemampuan dan fitur yang dimiliki masing-masing jenis atau nomor seri mikrokontroler tersebut.

Intruksi dalam mikrokontroler dikelompokkan dalam beberapa bagian, yaitu:

a. Operasi aritmatika

Yang termasuk dalam kelompok ini adalah instruksi penambahan (add,addc,inc), pengurangan (subb,dec), perkalian (mul), pembagian (div), dan pengaturan desimal (da).

b. Operasi Logika

Meliputi instruksi and, or, xor, clear (clr), rotasi (rl, rlc, rr, rrc), komplemen (cpl), dan pertukaran nibble dalam akumulator (swap).
c. Transfer Data

Meliputi intruksi penyalinan data (mov, movc, movx), menyimpan dan mengambil pada stack (push, pop), dan pertukaran data (xch, xchd).

d. Operasi Boolean

Terdiri dari intruksi untuk carry dan bit, antara lain clear (clr), set bit (setb), komplement (cpl), anl, orl, mov, dan intruksi jump yang mengacu pada carry dan bit seperti je, jnc, jb, jnb, dan jbc.

e. Operasi percabangan

Terdiri dari intruksi pemanggilan sub rutin (acall, lcall), kembali dari sub rutin (ret), kembali dari interupsi (reti), lompatan relatif maupun dengan syarat (ajmp, sjmp, ljmp, jmp, jz, jnz, ejne, djnz). Ada juga intruksi agar dalam satu siklus clock tidak terjadi operasi yaitu nop.

Karena dalam pemrograman mikrokontroler data diletakkan di berbagai lokasi memori, maka dikenal jenis-jenis pengalamatan (addressing mode), antara lain:

a. Pengalamatan secara langsung

Intruksi pengalamatan secara langsung untuk menunjuk data yang berada di dalam memori dengan dengan cara menyebut nomor alamat tempat data tersebut berada. Alamat adalah lokasi pada RAM internal atau derah SFR.

Contoh: MOV A, 07h. Intruksi ini berarti memindahkan data yang berada di alamat 07h ke akumulator dan menyimpannya.

b. Pengalamatan secara tidak langsung

Intruksi yang menggunakan mode pengalamatan tidak langsung
melibatkan register yang akan di akses Contoh : MOV A,@R0, yang berarti meyalin data dari memori yang alamatnya dicatat oleh R0 ke akumulator. Jika alamat dalam R0 berubah-ubah, maka data yang akan diambil juga berubah sesuai data di alamat yang ditunjuk R0.

c. Pengalamanan Register

Instruksi dengan meyebutkan data dalam register, Contoh : MOV A,R0

Yang berarti menyalin data yang disimpan di R0 ke akumulator

d. Pengalamanan dengan segera

Instruksi yang menyebutkan data dengan segera, karena data tersebut sudah berada dalam instruksi.

Contoh : MOV A,#50h

Berarti mengisi akumulator dengan data 50h

2.3. **Pembanding**

Sebuah pembanding berfungsi untuk membandingkan tegangan input pada satu masukan dengan suatu tegangan acuan pada masukan yang lainnya. Rangkaian pembanding sederhana dapat dibuat dengan sebuah Op-Amp tanpa hambatan umpan balik, seperti pada gambar 2.5.

![Gambar 2.5 Op-Amp sebagai pembanding](image)
Bila masukan membalik *(inverting)* dihubungkan dengan tanah, maka jika ada tegangan masukan pada kaki masukan tak membalik *(non inverting)* yang kecil sudah cukup membuat *Op-Amp* menjadi jenuh. Hal ini disebabkan oleh tegangan pada simpal terbuka (gain) sangat besar, biasanya diatas 100000.

Besar tegangan masukan yang dibutuhkan untuk menghasilkan kejenuhan positif adalah :

\[ \frac{V_{in}}{V_{out}} = \frac{1}{Gain} \] (2.1)

### 2.3.1 Pembanding dengan tegangan acuan

Tegangan acuan dalam pembanding digunakan untuk menentukan keluaran dari *Op-Amp* itu sendiri. Apabila *Vin* lebih besar dari tegangan acuan maka keluarannya tinggi akan tetapi bila *Vin* lebih kecil dari tegangan acuan maka keluarannya rendah.

Besarnya nilai titik perpindahan dapat diubah, yaitu tegangan yang ditetapkan sebagai penentu keluaran dari *Op-Amp*. Dengan menerapkan tegangan acuan pada masukan membalik *(inverting)*, seperti terlihat pada gambar 2.6 berikut ini :
Gambar 2.6 Pembanding dengan titik pergeseran yang dapat diatur

Pada gambar 2.6 besarnya titik acuan yang diterapkan pada masukan pembalik adalah:

\[ V_{ref} = \frac{R_2}{R_1 + R_2} V_{cc} \]

Setelah pada masukan pembalik diberi tegangan acuan (referensi) maka titik perpindahannya sekarang sama dengan \( V_{ref} \). Bila \( V_{in} \) sedikit lebih besar daripada \( V_{ref} \) keluarannya akan mencapai kejenuhan positif. Bila \( V_{in} \) sedikit lebih kecil daripada \( V_{ref} \) keluarannya akan mencapai kejenuhan negatif.

2.4 Light Dependent Resistor (LDR)

*Light Dependent Resistor* adalah resitor yang nilai resistansinya berubah-ubah tergantung pada intensitas cahaya yang diterimanya. Komponen ini terbuat dari bahan Kadmium Sulfida yang mempunyai nilai hambatan sangat besar (lebih kurang satu mega ohm) jika tidak terkena cahaya. Begitu pula sebaliknya, nilai hambatannya akan turun hingga beberapa ratus ohm jika terkena cahaya. Karena
itu LDR dapat digunakan untuk mendeteksi perbedaan intensitas cahaya. Simbol LDR ditunjukkan pada gambar 2.7.

Gambar 2.7 Simbol Light Dependent Resistor (LDR)

2.5 Transistor Bipolar Sebagai Saklar

Transistor dua kutub (Bipolar Junction Transistor, BJT) adalah sebuah piranti dari bahan semi konduktor, yang terdiri dari dua jenis yaitu npn dan pnp, yang secara simbolik dapat dibedakan dengan memperhatikan arah panahnya. Ketiga daerah bahan semikonduktor tersebut masing-masing dihubungkan dengan terminal sebagai kolektor (c), basis (b), dan emitter (e). Transistor berfungsi sebagai penguat bila bekerja pada daerah aktif, dan sebagai saklar bila berada dalam keadaan terpuncuk (cut off) atau keadaan jenuh (saturation).

Gambar 2.8 Rangkaian dasar transistor sebagai saklar
Pada gambar 2.8, \( V_{BE} \) adalah tegangan yang mengalir antara basis dan emitter besarnya 0,7 volt untuk bahan silikon. Jika \( V_{BB} \) lebih kecil dari \( V_{BE} \) maka dapat dianggap kaki basis-emiter diberi bias balik dan arus \( I_B = 0 \), sehingga \( I_E = I_C = 0 \), \( V_C = V_{CE} = V_{CC} \), keadaan transistor yang demikian disebut terpancung. Tetapi jika \( V_{BB} \) lebih besar dari \( V_{BE} \), akan mengalir arus basis (\( I_B \)) yang besar:

\[
I_B = \frac{V_{BE} - V_{BB}}{R_B} \quad ..................................(2.3)
\]

Dan membuat transistor dalam keadaan jenuh (saturasi). keadaan ini akan menyebabkan arus kolektor (\( I_C \)) mengalir ke emitter melalui basis yang besar:

\[
I_C = \frac{V_{CC} - V_{CE}}{R_C} \quad ..................................(2.4)
\]

Dalam keadaan jenuh, tegangan antara kolektor dan emitter (\( V_{CE} \)) ≈ 0 volt, sehingga dapat dianggap \( V_C = 0 \) volt, arus emitter (\( I_E \)) yang mengalir adalah:

\[
I_B = I_C + I_B \quad ..................................(2.5)
\]
2.6 Motor Stepper


Perlakuan eksitasi pada kumparan motor stepper dapat dilakukan dengan dua mode. Mode yang pertama adalah mode eksitasi tunggal dan mode eksitasi ganda, untuk lebih jelasnya bisa dilihat dalam gambar 2.9.

![Diagram Motor Stepper](image)

a. eksitasi tunggal  
b. eksitasi ganda

Gambar 2.9 Eksitasi pada motor stepper
Pada gambar tersebut ditunjukkan bagaimana rotor dapat bergeser langkah demi langkah karena adanya gaya tarik magnet antara magnet permanen pada rotor dengan magnet yang ditimbulkan oleh kumparan yang dieksitasi. Perlakuan eksitasi tunggal akan menyebabkan kutub S rotor bergeser ke kutub U kumparan yang dieksitasi, dan pada eksitasi ganda kutub S rotor akan berada antara dua kumparan yang dieksitasi.

Urutan eksitasi pada motor stepper dapat dilakukan dengan dua cara yaitu:

1. Mode eksitasi tunggal

   Hanya satu kumparan yang dieksitasi pada satu saat, dengan urutan seperti tampak dalam tabel 2.1

   **Tabel 2.1 Urutan perlakuan eksitasi tunggal**

<table>
<thead>
<tr>
<th>Langkah</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
<th>L4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

2. Mode eksitasi ganda

   Dalam suatu saat ada dua kumparan yang dieksitasi.

   **Tabel 2.2 Urutan perlakuan eksitasi ganda**

<table>
<thead>
<tr>
<th>Langkah</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
<th>L4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
2.7 Penyangga (buffer)

Adalah piranti aktif yang di pasang antara dua tahapan yang memungkinkan pemindahan sinyal listrik tetapi juga mengisolasi antara dua tahapan tersebut. Dengan demikian perubahan impedansi pada rangkaian yang satu tidak terpengaruh terhadap yang lain. Simbol *buffer* ditunjukkan pada gambar 2.10

\[\text{in} \rightarrow \text{out}\]

Gambar 2.10 Simbol penyangga (*buffer*)

2.8 Foto Transistor

Foto transistor adalah suatu piranti elektronik yang cara kerjanya hampir sama dengan transistor biasa, perbedaannya pada foto transistor tidak menggunakan arus basis untuk mengaktifkan transistor tersebut. Pada foto transistor terdapat jendela transparan sehingga cahaya dapat mencapai pertemuan antara basis dan kolektor. Arus bocor yang disebabkan oleh cahaya yang mengenai pertemuan basis dan kolektor inilah yang bertindak sebagai arus basis. Foto transistor mempunyai sifat yang lebih peka dibanding dioda foto. Simbol dari fot transistor ditunjukkan pada gambar 2.11.

\[\text{Gambar 2.11 Simbol Foto Transistor}\]
2.9 Motor DC

Motor DC adalah suatu alat yang dapat mengubah energi listrik menjadi energi gerak atau putaran dengan prinsip elektromagnet. Prinsip kerja dari motor DC adalah sebuah lilitan kawat yang sering disebut jangkar atau lilitan jangkar diletakkan dalam suatu medan magnet. Pada saat lilitan jangkar tersebut dialiri arus listrik searah akan timbul medan magnet buatan. Karena berada dalam medan magnet permanen maka terjadilah perpotongan medan magnet sehingga lilitan jangkar menerima gaya tolak dan mendorong lilitan untuk berputar.

Gambar 2.12 Skema motor DC

Pada saat lilitan jangkar berputar akan membuat komutator ikut berputar, komutator adalah sebuah plat tembaga yang berbentuk cincin yang terbelah. Komutator ini berfungsii membalik arah arus dalam lilitan jangkar sehingga terjadi perubahan arah arus yang menyebabkan lilitan jangkar berputar terus. Arus listrik masuk melalui sikat dan dengan mengubah arah arus dari sikat dapat menentukan putaran motor kekanan atau kekiri. Simbol motor DC ditunjukkan gambar 2.13

Gambar 2.13 Simbol motor DC
BAB III
PERANCANGAN

3.1 Perancangan Perangkat Keras

3.1.1 Pemisah Benda Berwarna Merah, Kuning, Biru

Perancangan pemisah benda berwarna merah, kuning, biru secara garis besar dapat dilihat pada gambar 3.1

Gambar 3.1 Diagram blok pemisah benda berwarna merah, kuning, biru
Komponen-komponen pendukung mikrokontroler untuk membentuk sistem pemisah benda berwarna merah, kuning, biru tua adalah sebagai berikut:

a. Dua buah motor *stepper* untuk megerakkan penutup yang berfungsi mengarahkan benda yang dideteksi menuju tempat yang disediakan.

b. Sebuah motor reduksi untuk menggerakkan *conveyor*.

c. Sebuah sensor (bagian pemancar dan penerima inframerah dipisahkan) sebagai sensor keberadaan benda.

d. Sebuah sensor (bagian pemancar dan penerima inframerah dipisahkan) sebagai sensor akhir.

e. Sebuah sensor (LED putih sebagai pemancar dan LDR sebagai penerima) sebagai sensor pembeda warna.

f. Dua buah limit switch sebagai masukan agar posisi motor steper pada keadaan yang sama.

g. Resonator kristal, catu daya, dan rangkaian *reset* dihubungkan ke laki IC yang telah ditentukan.

Komponen-komponen tersebut dihubungkan dengan mikrokontroler AT89C51 untuk membentuk sistem pemisah benda berwarna merah, kuning, dan biru. Sedangkan hubungan antar komponen dengan pin keluaran dari mikrokontroler dapat dilihat dalam gambar 3.2.
Gambar 3.2 Diagram kotak lokasi komponen pendukung

Dalam pembuatan sistem pemisah benda merah, kuning, biru tua dibutuhkan alat pendukung yaitu konveyor yang berfungsi penggerak benda yang dideteksi untuk itu gambar alat secara fisik dapat dilihat dalam gambar 3.3
Gambar 3. Bentuk fisik pemisah benda berwarna

Keterangan gambar 3.3:

1. Sensor keberadaan benda.
2. Bet konveyor.
5. Motor stepper.
6. Pintu pengarah benda.
7. Penggerak bet konveyor.
10. Tempat pengarah benda.
11. Pengarah akhir.
12. Limit switch.
3.1.2 Bentuk Benda

Benda yang dideteksi adalah berbentuk tabung dengan diameter 4,5 cm dan tinggi 4 cm. Dengan bentuk tabung memudahkan benda bergeser pada saat benda tersebut diarahkan menuju tempat yang telah disediakan. Untuk lebih jelasnya dapat dilihat pada gambar 3.4.

Gambar 3.4 Bentuk benda

3.1.3 Mikrokontroler AT89C51

Dalam mikrokontroler AT89C51 mempunyai port (P) atau terminal yang berfungsi ganda yaitu dapat digunakan sebagai masukan atau keluaran. Masing-masing dilengkapi dengan latch, output driver, dan input buffer. P1, P2, P3 dilengkapi dengan internal pull up, sedangkan P0 mempunyai keluaran open drain, sehingga jika P0 diberi data ‘1’ atau high (5 volt) melalui pemrograman keadaan P0 tersebut mengambang.

Tegangan keluaran untuk logika ‘0’ (low) pada semua port adalah 0 – 0,45 volt dan untuk keadaan logika ‘1’ (high) 2 - 5 volt. Mikrokontroler memiliki
rangkaian internal clok tetapi masih memerlukan tambahan resonator kristal IC dan dipilih frekuensi 12 MHz. Sehingga dalam mengerjakan intruksi, mikrokontroler memerlukan waktu 1 mikrodetik untuk satu siklusnya.

3.1.4 Rangkaian Pembeda Warna

Dalam perancangan rangkaian pembeda warna, sensor yang digunakan untuk mendeteksi warna benda adalah dengan sistem pemantulan cahaya. Apabila di depan sensor ada sebuah benda berwarna maka sinar yang dipancarkan oleh led akan dipantulkan oleh benda tersebut dan cahaya yang dipantulkan akan diterima LDR. Setiap benda yang dideteksi akan mempunyai intensitas cahaya yang berbeda pada setiap warna benda dan akan menghasilkan tegangan yang berbeda pada setiap warna.

Rangkaian sensor tersebut dirancang dengan besarnya arus pada dioda led (I_f) sebesar 10 mA dan tegangan pada led (V_f) adalah 2 volt. Seperti pada gambar 3.5 dapat dihitung hambatan seri antara sumber tegangan dengan led adalah:

\[
R_s = \frac{V_s - V_f}{I_f}
\]

\[
R_s = \frac{5V - 2V}{10mA}
\]

\[
R_s = 300 \, \Omega
\]
Gambar 3.5 Rangkaian sensor pembeda warna

Dengan memanfaatkan rangkaian diatas untuk merancang sensor warna penulis telah melakukan penelitian sebagai data awal, pengambilan data tersebut diperoleh dengan mengukur nilai tegangan LDR pada saat benda berwarna memantulkan cahaya led, seperti pada gambar 3.5 data tersebut adalah sebagai berikut :

Tabel 3.1 Tegangan LDR

<table>
<thead>
<tr>
<th>Warna benda</th>
<th>Pengukuran (volt)</th>
<th>Rata-rata minimal</th>
<th>maximal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Merah</td>
<td>1,62</td>
<td>1,59</td>
<td>1,61</td>
</tr>
<tr>
<td>Kuning</td>
<td>1,28</td>
<td>1,29</td>
<td>1,27</td>
</tr>
<tr>
<td>Biru</td>
<td>2,27</td>
<td>2,30</td>
<td>2,28</td>
</tr>
</tbody>
</table>

Dari tabel diatas dapat diketahui bahwa nilai tegangan LDR yang paling besar adalah wana biru karena intensitas cahayanya paling kecil kemudian level dibawahnya warna merah, sedangkan warna kuning mempunyai nilai tegangan paling kecil sebab intensitas cahanya paling besar. Agar mikrokontroler dapat
mengolah perbedaan tegangan dari LDR menjadi sebuah masukan dibutuhkan rangkaian pengolah tegangan tersebut.

Dengan adanya tiga buah warna maka dibutuhkan tiga buah pembanding dan tiga tegangan referensi sebagai tegangan acuan yang didapat dari resistor variabel (VR). Pembanding sendiri mempunyai dua buah masukan yaitu membalik dan tak membalik, agar pembanding aktif maka masukan tak membalik harus lebih besar dari masukan membalik. Dengan demikian tegangan referensi dimasukan pada masukan tak membalik dan tegangan LDR dihubungkan pada masukan membalik. Dalam perancangan ini masukan yang paling besar adalah tegangan LDR warna biru kemudian level tegangan dibawahnya warna merah dan paling kecil adalah warna kuning. Karena tegangan referensi harus lebih besar dari tegangan LDR maka tegangan referensi disesuaikan dengan level tegangan pada masukan LDR. Dari uraian diatas penulis dapat merancang sensor warna dengan tiga buah masukan pada mikrokontroler dengan kondisi yang berlainan sesuai dengan warna benda masing-masing.

Pada perancangan untuk rangkaian pembanding menggunakan Op-Amp LM 324 yang pada rangkaian ini berfungsi sebagai sebuah saklar. Sebagai tegangan acuan digunakan resistor variabel sebesar 100KΩ yang dirasa cukup sebagai pembagi tegangan agar tegangan acuan memadai dan berada diatas tegangan LDR. Untuk lebih lengkapnya dapat dilihat dalam gambar 3.6
Gambar 3.6 Rangkaian pembanding LM 324

Pada rangkaian pembawa warna ini masukan dari LDR (Vldr) dihubungkan dengan masukan *inverting* op-amp LM324. Masukan *noninverting* menjadi tegangan acuan yang besarnya diatur oleh variabel resistor (VR). Pada perancangan $V_{EE}$ pada LM 324 dihubungkan ke *ground* sehingga besarnya 0 Volt. Jadi Vout dari op-amp berkisar antara 0 Volt sampai 5 Volt.

Apabila masukan noninverting lebih besar dari masukan inverting maka op-amp akan aktif atau memberi masukan ‘1’, dan pada saat masukan noninverting lebih kecil dari masukan inverting maka op-amp tidak aktif atau memberi masukan ‘0’. Untuk mendeteksi warna benda maka setiap op-amp digunakan untuk satu warna sehingga pengesetan untuk masing – masing resistor variabel disesuaikan dengan tegangan LDR pada masing – masing warna. Karena tegangan LDR warna kuning paling kecil maka VR1, VR2, VR3 > Vldr sehingga
ketiga op-amp aktif, untuk warna merah tegangan LDR berada diatas tegangan referensi warna kuning maka VR1, VR3 > Vldr sehingga hanya dua op-amp yang aktif sebab VR2 < Vldr. Sedangkan untuk warna biru mempunyai tegangan LDR diatas tegangan referensi warna merah dan warna kuning maka VR3 > Vldr dan hanya satu op-amp yang aktif sebab VR1, VR2 < Vldr. Dari uraian diatas penulis dapat membuat tabel masukan pada mikrokontroler sebagai berikut:

<table>
<thead>
<tr>
<th>Benda</th>
<th>P1.2</th>
<th>P1.3</th>
<th>P1.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kuning</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Merah</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Biru</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

3.1.5 Sensor Keberadaan Benda

Rangkaian keberadaan benda berfungsi untuk mendeteksi ada tidaknya benda yang akan dideteksi warnanya. Sensor yang digunakan berupa pasangan infra merah dan phototransistor.

![Gambar 3.7 Rangkaian keberadaan benda](image)

Pada gambar tersebut diperoleh dengan membatasi If = 30 mA, dan Vf (Vled) = 2,3 volt maka dapat dihitung:
\[ R_l = \frac{V_{cc} - V_f}{I_f} \]
\[ = \frac{5 - 2.3}{0.05} \]
\[ = 90 \, \Omega \]

Dalam perhitungan nilai hambatan yang diperoleh adalah 90 \, \Omega, namun nilai hambatan tersebut tidak ada di pasaran sehingga dalam prakteknya digunakan hambatan sebesar 100 \, \Omega.

Nilai hambatan kolektor agar phototransistor mudah memasuki daerah jenuh dengan \( I_c = 0.5 \) mA pada saat phototransistor menerima cahaya dibutuhkan hambatan kolektor sebesar:

\[ R_c = \frac{V_{cc} - V_{CE}}{I_{C_mak}} \]
\[ = \frac{5 - 0.3}{0.0005} \]
\[ = 9400 \, \Omega \]

Dalam perhitungan nilai hambatan kolektor adalah 9400 \, \Omega, namun nilai hambatan tersebut sulit dicari di pasaran sehingga dalam prakteknya digunakan hambatan sebesar 10.000 \, \Omega.

3.1.6 Penggerak Motor Stepper

Motor stepper dipakai sebagai pintu penutup dari benda yang di deteksi, kemudian diarahkan menuju papan luncur. Rangkaian penggerak motor stepper dirancang untuk bekerja sebagai saklar yaitu memutuskan atau menghubungkan salah satu kumparan motor stepper dengan ground. Untuk itu digunakan driver ULN 2803 yang memberi keuntungan sebagai penggerak motor stepper dengan
jumlah 8 masukan dan 8 keluaran cukup untuk 2 motor stepper. Rangkaian driver motor stepper dapat dilihat pada gambar 3.8.

Gambar 3.8 Rangkaian driver motor stepper

Dari rangkaian di atas keluaran dari ULN 2803 dipakai untuk dua motor sedangkan common dari motor dijadikan satu dan digabung dengan Vcc. Pada saat masukan microkontroler dan melewati buffer memberi logika 1 akan mengaktifkan driver serta menghubungkan lilitan dengan ground. Lilitan motor stepper unipolar terbagi 5 bagian yaitu L1, L2, L3, L4 serta common sehingga total masukan ada 5 yang dihubungkan dengan ULN 2803.
3.1.7 Sensor Akhir

Rangkaian sensor akhir berfungsi untuk mengetahui apakah benda yang terdeteksi telah melewati papan luncur, apabila sudah sensor akan memberi masukan pada mikrokontroler dan menghentikan konveyor. Rangkaian yang digunakan adalah pasangan infra merah dan photo transistor.

![Diagram of sensor circuit](image)

Gambar 3.9 Rangkaian infra merah dan photo transistor

Pada gambar tersebut diperoleh dengan membatasi \( I_f = 30 \) mA, dan \( V_f (V_{led}) = 2,3 \) volt maka dapat dihitung:

\[
R_1 = \frac{V_{cc} - V_f}{I_f} = \frac{5 - 2,3}{0,05} = 90 \Omega
\]

Dalam perhitungan nilai hambatan yang diperoleh adalah 90 \( \Omega \), namun nilai hambatan tersebut tidak ada di pasaran sehingga dalam prakteknya digunakan hambatan sebesar 100 \( \Omega \).
Nilai hambatan kolektor agar phototransistor mudah memasuki daerah jenuh dengan $I_c=0.5 \text{mA}$ pada saat phototransistor menerima cahaya dibutuhkan hambatan kolektor sebesar:

$$R_C = \frac{V_{CC} - V_{CB}}{I_{C\text{max}}},$$

$$= \frac{5 - 0.3}{0.0005},$$

$$= 9400 \ \Omega$$

Dalam perhitungan nilai hambatan kolektor adalah 9400 $\Omega$, namun nilai hambatan tersebut sulit dicari di pasaran sehingga dalam prakteknya digunakan hambatan sebesar 10.000 $\Omega$

Gambar 3.9 merupakan rangkaian inframerah sebagai pemancar dan fototransistor sebagai penerima. Cahaya infra merah yang masuk dalam fototransistor akan dianggap sebagai arus basis. Ini mengakibatkan keluaran dari rangkaian menjadi rendah, dan bila tidak ada cahaya yang masuk maka keluaran menjadi tinggi. Keluaran dihubungkan dengan masukan noninverting ($V_F$) op-amp 324 sedangkan masukan inverting ($V_N$) menjadi tegangan acuan yang besarnya diatur oleh variabel transistor ($V_R1$) sebesar 100Kohm yang berfungsi sebagai pembagi tegangan. Op-amp LM324 kaki $V_{EE}$ dihubungkan dengan ground sehingga keluaranya bervariasi antara 0 Volt sampai 5 Volt.
Gambar 3.10 Rangkaian Lengkap Sensor Akhir

Tegangan pada masukan op-amp dibandingkan dan hasilnya akan mempengaruhi keluaran dari pembanding. Apabila $V_P > V_N$ maka Vout akan sebesar Vcc yaitu 5 Volt. Sebaliknya apabila $V_P < V_N$ maka keluaran dari pembanding sebesar 0 Volt.

3.1.8 Penggerak motor DC

Rangkaian penggerak motor DC dirancang untuk mengaktifkan relay yang difungsikan untuk menghubungkan motor DC dengan catu daya. Sedangkan transistor difungsikan sebagai saklar dan pengendalian transistor dilakukan melalui masukan pada basis transistor tersebut yang dihubungkan dengan mikrokontroler.
Gambar 3.11 Rangkaian penggerak motor DC

Pada gambar 3.11 adalah rangkaian penggerak motor DC. Berdasarkan datasheet tegangan keluaran dari IC Buffer 74LS541 adalah 3.4 volt pada keadaan logika '1' dan 0,35 volt pada keadaan logika '0'. Pada pengukuran yang telah dilakukan besarnya hambatan dalam relay adalah 300Ω, maka besarnya arus kolektor jenuh adalah:

\[ I_{c_{sat}} = \frac{V_{CC} - V_{CE}}{R_c} \]
\[ = \frac{12 - 0.3}{300} \]
\[ = 39 mA \]

Besarnya arus minimal basis agar transistor BD677 yang mempunyai β sebesar 750 menjadi jenuh adalah:

\[ I_{b_{sat}} = \frac{I_c}{\beta} \]
\[ = \frac{39 mA}{750} \]
\[ = 0,052 mA \]
Dengan $V_{BE}$ yang dimiliki BD 677 adalah 1,4 untuk itu dipilih nilai resistansi basis sebesar 1000 $\Omega$, sehingga besarnya arus basis adalah:

$$I_B = \frac{V_{BB} - V_{BE}}{R_B}$$

$$= \frac{3,4 - 1,4}{1000}$$

$$= 2 mA$$

Sehingga arus basis lebih besar dari pada arus basis yang dibutuhkan agar transistor BD 677 menjadi jenuh, jadi hambatan 1000$\Omega$ dapat digunakan. Nilai perhitungan diatas berlaku untuk keadaan logika ‘1’ pada terminal basis. Sedangkan untuk keadaan logika ‘0’, terminal basis mendapat tegangan $V_{BB}$ sebesar 0,3 volt dan nilai ini masih dibawah nilai $V_{BE}$ sehingga transistor tidak akan jenuh.

### 3.1.9 Limit Switch

Limit Switch adalah saklar untuk memberi masukan pada saat motor stepper bergerak membuka apabila benda telah sampai pada papan luncur dan telah melewati sensor akhir.

![Gambar 3.12 Rangkaian Limit Switch](image-url)
Pada gambar 3.12 apabila saklar dalam keadaan normal (tuas tidak tertekan), terminal NO (Normaly Open) tidak terhubung ke terminal common sedangkan terminal NC (Normaly Close) terhubung. Jika saklar tertekan, keadaan berubah menjadi kebalikannya. Fungsi dari saklar tersebut adalah menghubungkan P1.5 dan P1.6 dengan ground serta memberi logika ‘0’ pada mikrokontroler (keadaan normalnya adalah logika ’1’).

3.2 Diagram Alir

Dalam perancangan perangkat lunak sistem pemisah benda berwarna merah, kuning, biru diperlukan diagram alir sistem untuk memudahkan dalam pembuatan program dari sistem tersebut. Sistem pemisah benda berwarna merah, kuning, biru juga membutuhkan diagram tersebut. Dalam diagram alir dibawah ini merupakan diagram alir dalam satu kerja sehingga program akan selalu melakukan proses yang sama bagi setiap benda berwarna yang akan dideteksi. Untuk lebih jelasnya bisa dilihat pada gambar 3.13, dari gambar dapat dijelaskan sebagai berikut:

1. Catu daya dinyalakan (on) atau reset

2. Inisialisasi variabel, penggunaan port dan masukan data awal

3. Sensor keberadaan benda mendeteksi ada benda atau tidak jika ada maka konveyor bergerak, jika tidak kembali kelangkah 3

4. Ketika sensor keberadaan benda mendeteksi adanya benda kemudian dilanjutkan ke sensor warna untuk mendeteksi apakah sensor biru belogika 1, jika ya maka konveyor berhenti jika tidak mengulang langkah 4
Gambar 3.13 Diagram alir system pemisah benda berwarna
5. Setelah konveyor berhenti sensor warna akan mendeteksi apakah warna benda tersebut, jika warnanya merah maka konveyor kembali bergerak dan kemudian motor stepper 1 akan bergerak kekiri sebanyak 25 langkah, kemudian menuju langkah 6

6. Sensor akhir akan mendeteksi apakah ada benda yang melewatinya, jika ada maka motor stepper 1 akan bergerak kekanaan sampai menekan limit switch apabila belum tertekan motor stepper 1 akan terus bergerak. Setelah limit switch tertekan maka konveyor berhenti, jika tidak mengulang langkah 6

7. Jika pada langkah 5 benda yang terdeteksi bukan berwarna merah tetapi berwarna kuning maka konveyor kembali bergerak dan kemudian motor stepper 2 akan bergerak kekiri sebanyak 32 langkah, kemudian menuju langkah 8

8. Sensor akhir akan mendeteksi apakah ada benda yang melewatinya, jika ada maka motor stepper 2 akan bergerak kekanaan sampai menekan limit switch apabila belum tertekan motor stepper 2 akan terus bergerak. Setelah limit switch tertekan maka konveyor berhenti, jika tidak mengulang langkah 8

9. Apabila pada langkah 7 benda yang terdeteksi bukan warna merah dan kuning, melainkan warna biru maka konveyor akan bergerak dan menuju langkah 10

10. Sensor akhir akan mendeteksi apakah ada benda yang melewatinya, jika ya maka konveyor akan berhenti, jika tidak akan mengulang langkah 10
11. Apabila pada langkah 9 benda yang terdeteksi bukan warna merah, kuning atau biru maka konveyor akan bergerak selama tunda waktu yang diberikan dan kemudian konveyor akan berhenti
BAB IV
PENGAMATAN DAN PEMBAHASAN

4.1 Deteksi keberadaan dan warna benda

Dalam pendeteksian ada tidaknya benda yang akan diseleksi maka program akan terus menunggu sampai ada benda yang akan diseleksi. Berikut penggalan program untuk deteksi benda:

```
Line   source
47    cek_objek:
48          jnb    sens_objek,$
49            .................
```

Pada penggalan program diatas menunjukkan bahwa pada saat sens_objek (sensor keberadaan benda) pada port P1.0 masih dalam kondisi ‘0’ maka program akan tertahan pada perintah ini. Tapi apabila P1.0 dalam kondisi ‘1’ maka program akan melanjutkan perintah selanjutnya.

Setelah ada benda yang akan dideteksi maka akan dilakukan seleksi warna dari benda tersebut. Proses penyeleksian terdapat dalam penggalan program dibawah ini:

```
Line   source
51    cek_warna:
52          jnb    sens_biru,$
53                acall    tunda
54                clr    conveyor
55                acall    mtunda
56                mov    A,pl
57                anl    A,#1ch
58    yell1 :
59                mov    r1,#5
60    yell2 :
61          cjne    A,#1ch,redl
62                acall    mtunda
63                clr    conveyor
```

45
Jika \textit{sens biru} (P1.4) berkondisi 1, maka \textit{conveyor} (P1.7) akan berhenti. \textit{Accumulator} akan menyalin isi dari \textit{port1} kemudian isi dari \textit{accumulator} akan di AND-kan dengan data \textit{lch} ini berarti yang dibaca hanya P1.2, P1.3 dan P1.4 sedangkan yang lainnya akan diabaikan. Kemudian program menjalankan perintah pada label \textit{yellow} yaitu memindahkan data 5 kedalam register \textit{r1}, selanjutnya membandingkan apakah \textit{accumulator} berisi data \textit{lch} (warna kuning) jika ya maka program akan memastikan \textit{conveyor} tetap berhenti kemudian mengurangi isi dari \textit{r1} apabila belum bernilai 0 maka akan lompat pada label \textit{yellow2} begitu seterusnya sampai kondisi \textit{r1} bernilai 0 ini untuk memastikan bahwa \textit{accumulator} bernilai \textit{lch}. Apabila nilai nya tidak \textit{lch} maka program akan melompat pada label \textit{red1} disini juga akan dibandingkan apakah \textit{accumulator} bernilai \textit{14h} (warna merah) jika ya maka program akan meneruskan perintah selanjutnya yaitu melakukan pengecekan sampai \textit{r1} bernilai 0. Apabila tidak maka program lompat pada label \textit{blue1} disini juga akan dibandingkan apakah isi \textit{accumulator} sama dengan data \textit{10h}.
(warna biru) jika ya juga akan melakukan hal yang sama jika tidak akan melompat pada label invalid.

4.2 Pemrograman gerak motor stepper

Setelah mendapatkan warna benda kemudian menggerakkan salah satu motor stepper agar benda menuju papan luncur. Untuk menggerakkan motor stepper unipolar dengan mode eksitesi ganda terdapat dalam penggalan program berikut ini:

```
<table>
<thead>
<tr>
<th>Line</th>
<th>source</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>yell_obj :</td>
</tr>
<tr>
<td>124</td>
<td>acall tunda</td>
</tr>
<tr>
<td>125</td>
<td>mov r3,#0cch</td>
</tr>
<tr>
<td>126</td>
<td>mov r0,#32</td>
</tr>
<tr>
<td>127</td>
<td>tutup_pintu_kuning :</td>
</tr>
<tr>
<td>128</td>
<td>mov pintu2,r3</td>
</tr>
<tr>
<td>129</td>
<td>acall tunda</td>
</tr>
<tr>
<td>130</td>
<td>mov A,r3</td>
</tr>
<tr>
<td>131</td>
<td>rr A</td>
</tr>
<tr>
<td>132</td>
<td>mov r3,A</td>
</tr>
<tr>
<td>133</td>
<td>djnz r0,tutup_pintu_kuning</td>
</tr>
</tbody>
</table>
```

Mode yang dipakai untuk menggerakkan motor stepper adalah mode eksitesi ganda yaitu ada dua lilitan yang dieksitasi. Pada penggalan program diatas pertama kali adalah memberi nilai awal pada register r3. Kemudian memberikan jumlah step pada register r0. Setelah memasukan nilai dalam register kemudian memindahkan data register r3 dalam pintu2 (port 3) dan diberikan tunda waktu. Kemudian data dalam r3 dipindahkan kedalam accumulator selanjutnya data dalam accumulator digeser ke kanan yang kemudian dipindah lagi kedalam r3 dengan nilai data yang sudah digeser. Selanjutnya perintah mengurangi nilai r0
dan apabila belum 0 program akan melompat pada label *tutup pintu kuning*,
begitu seterusnya sampai isi dari r0 bernilai kosong atau 0.

### 4.3 Waktu tunda

Secara umum waktu tunda merupakan kumpulan intruksi yang diulang-
ulang sehingga memerlukan waktu untuk mengerjakan intruksi—intruksi tersebut.

<table>
<thead>
<tr>
<th>Line</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>164</td>
<td><code>tunda:</code> mov R6,#60</td>
</tr>
<tr>
<td>165</td>
<td><code>tundal:</code> mov R7,#ffh</td>
</tr>
<tr>
<td>166</td>
<td>djonz R7,8</td>
</tr>
<tr>
<td>167</td>
<td>djonz R6,tundal</td>
</tr>
<tr>
<td>168</td>
<td>ret</td>
</tr>
</tbody>
</table>

Pada saat memasukkan data 60 pada R6 ini berarti program akan melakukan
putaran sebanyak 60 kali. Selanjutnya mengisi register R7 dengan bilangan
heksadesimal FF (255 desimal). Kemudian isi dari register R7 dikurang satu,
sedangkan tanda $ berarti program akan mengulang terus sampai isi register R7
bernilai 0. Sehingga waktu yang dibutuhkan untuk mengurangi isi register R7
adalah $(255 \times 2) + 3$ mikrodetik atau sama dengan 513 mikrodetik. Kemudian
mengurangi isi register R6 dan lompat pada label *tundal* apabila R6 belum sama
dengan nol. Sehingga total waktu yang diperlukan adalah $(513 \times 60)$ mikrodetik
atau sama dengan 30780 mikrodetik.
4.4 Pemrograman motor DC

Untuk menggerakkan motor DC cukup dengan memberi perintah per bit pada program mikrokontroler. Berikut adalah penggalan program untuk menggerakkan motor DC dan menghentikannya.

<table>
<thead>
<tr>
<th>Line</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td><code>setb</code> <code>conveyor</code></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td><code>clr</code> <code>conveyor</code></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Pada perintah `setb conveyor` (P1.7) berarti membuat nilai satu pada port 1 bit 7 pada microkontroller serta menggerakkan motor DC. Dan ini berlaku sebaliknya pada perintah `clr conveyor` (P1.7) ini berarti memberi nilai nol pada port 1bit 7 serta menghentikan motor DC.

4.5 Respon tegangan pada LDR

Dalam pengamatan ini benda yang diamati adalah berbentuk tabung dengan ukuran: diameter 4,5 cm, tinggi 4 cm. Benda tersebut dilapisi kertas marmer berwarna merah, kuning dan biru untuk lebih jelasnya warna dan tekstur kertas marmer yang dipakai dapat dilihat dalam lampiran 2.

Tegangan yang terukur adalah tegangan LDR. Tegangan tersebut dipengaruhi oleh kuat lampunya sinar yang diterima oleh LDR. Sinar tersebut adalah sinar pantul yang berasal dari led berwarna putih yang kemudian dipantulkan oleh benda dan diterima LDR dengan sudut 45°. Respon tegangan LDR dapat dilihat dalam tabel 4.1:
### Tabel 4.1 Respon tegangan LDR

<table>
<thead>
<tr>
<th>Warna benda</th>
<th>Pengukuran (volt)</th>
<th>Rata rata</th>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Merah</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1,64</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1,54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1,60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1,60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1,59</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1,62</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1,58</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1,64</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1,68</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1,60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1,609</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kuning</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1,29</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1,28</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1,27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1,27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1,30</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1,25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1,28</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1,27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1,29</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1,25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1,275</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biru</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2,27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2,44</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2,40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2,27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2,25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2,40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>2,35</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>2,30</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>2,29</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2,42</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2,39</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2,25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2,44</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Dari pengamatan diatas dapat diperoleh tegangan rata-rata untuk warna merah dari respon tegangan adalah:

\[
Rata - rata = \frac{\sum \text{pengukuran tegangan}}{\text{jumlah pengukuran}}
\]

\[
= \frac{16,09}{10}
\]

\[
= 1,609 \text{ volt}
\]

Dari data yang diperoleh untuk mendapatkan tegangan referensi yang lebih aman maka pengesetan resistor variabel diatas dari tegangan maksimal. Pengesetan untuk resistor variabel adalah sebagai berikut:

\[
\begin{align*}
VR1 &= 1,68 \text{ volt} \\
VR2 &= 1,30 \text{ volt} \\
VR3 &= 2,44 \text{ volt}
\end{align*}
\]

Dengan mengambil nilai maksimal dari pengukuran maka kita dapat menentukan tegangan dibawah 1,30 volt akan dideteksi warna kuning, sedangkan tegangan diatas 1,30 volt dan dibawah 1,68 volt akan dideteksi warna merah serta tegangan diatas 1,68 volt namun dibawah 2,44 akan dianggap berwana biru.
4.6 **Driver motor**


**Tabel 4.2 Data arus motor stepper**

<table>
<thead>
<tr>
<th>Motor</th>
<th>Arus maximal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motor stepper 1</td>
<td>130 mA</td>
</tr>
<tr>
<td>Motor stepper 2</td>
<td>140 mA</td>
</tr>
</tbody>
</table>

Hasil dari tabel diatas menunjukkan kondisi arus yang paling tinggi yaitu pada motor stepper 2 sebesar 140 mA. Dari data sheet ULN2803 dapat mengeluarkan arus sebesar 500 mA untuk beban kontinyu. Sehingga penggerak motor stepper berada dalam kondisi aman.

Sedangkan untuk motor DC penggeraknya menggunakan relay dan arus yang dibutuhkan motor DC adalah 630 mA. Dengan begitu penggerak tersebut juga dalam kondisi aman karena relay dapat mencapai 6 A.

4.7 **Data hasil pengamatan**

4.7.1 **Waktu pemrosesan benda**

Berikut adalah pengukuran waktu yang dibutuhkan benda dari pertama kali memotong sensor keberadaan benda sampai sensor akhir.
Tabel 4.3 Waktu pemrosesan benda

<table>
<thead>
<tr>
<th>Warna benda</th>
<th>Data pengukuran (detik)</th>
<th>Rata-rata (detik)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Merah</td>
<td>9,9</td>
<td>10,1</td>
</tr>
<tr>
<td>Kuning</td>
<td>11,8</td>
<td>12,1</td>
</tr>
<tr>
<td>Biru</td>
<td>14</td>
<td>13,6</td>
</tr>
</tbody>
</table>

Dari data diatas dapat diketahui rata-rata waktu yang dibutuhkan dalam pemrosesan benda merah adalah :

\[
Rata - rata = \frac{\sum \text{waktu pemrosesan}}{\text{jumlah pengukuran}}
\]

\[
= \frac{50}{5} = 10 \text{ detik}
\]

Perbedaan waktu pemrosesan benda dipengaruhi oleh jarak yang ditempuh setiap benda untuk mencapai papan luncur.

4.7.2 Kecepatan benda

Kecepatan adalah perpindahan yang dialami oleh suatu benda tiap satuan waktu. Dalam pengamatan ini waktu yang dibutuhkan oleh benda berwarna untuk berpindah sejauh 20 cm adalah 2,7 detik, apabila perpindahan diberi simbol \( s \) dan waktu diberi simbol \( t \) maka kecepatannya adalah:
\[ v = \frac{s}{t} \]
\[ = \frac{20 \text{ cm}}{2,7 \text{ detik}} = 7,4 \text{ cm/detik} \]

4.7.3 Waktu pintu pengarahan

Berikut adalah data waktu yang dibutuhkan motor stepper menutup dan membuka pada saat mengarahkan benda pada papan lurus pengukuran dengan menggunakan stop watch.

<table>
<thead>
<tr>
<th>Motor</th>
<th>Menutup (detik)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motor stepper1</td>
<td>0,76</td>
</tr>
<tr>
<td>Motor stepper2</td>
<td>0,89</td>
</tr>
</tbody>
</table>

Untuk membandingkan lamanya pintu pengarah menutup dengan data yang diambil dengan stop watch maka penulis melakukan pengambilan data yang lebih baik yaitu dengan menggunakan osiloscop digital. Dan berikut adalah data pengukuran waktu yang dibutuhkan oleh motor stepper untuk menutup dengan menggunakan osiloscop digital yang terdapat pada gambar 4.1.
Gambar 4.1 Hasil pengukuran waktu dengan osiloskop digital

Dari data pengamatan dapat diketahui waktu yang dibutuhkan motor stepper untuk menutup adalah 0,76 detik dengan menggunakan stop watch dan 720,5 mili detik atau 0,72 detik. Dari pengamatan terjadi perbedaan ini disebabkan oleh pengukuran dengan stop watch kurang tepat karena hanya mengandalkan pandangan mata.

Dengan demikian waktu yang dibutuhkan motor stepper untuk menutup kurang dari satu detik sedangkan benda bergerak dalam satu detik adalah 7,4 cm sedangkan panjang lintasan adalah 21 cm. Sehingga sebelum benda sampai pada papan luncur motor stepper sudah menuju dan benda dapat diarahkan dengan tepat.
BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil pengamatan, maka dapat diambil kesimpulan:

1. Kecepatan rata-rata yang dibutuhkan oleh benda dari awal sampai akhir untuk benda merah 10 detik, benda kuning 12 detik dan benda biru 14 detik.

2. Kecepatan motor stepper ditentukan oleh tunda waktu yang diberikan pada saat penrograman mikrokontroler.

3. LDR dapat digunakan sebagai sensor warna pada waktu menggunakan jenis kertas yang sama yaitu dengan memanfaakan perbedaan resistensi sehingga menghasilkan perbedaan tegangan.

4. Pada tugas akhir ini tegangan rata-rata untuk setiap warna benda adalah berbeda yaitu warna merah rata-rata 1,604 volt untuk warna kuning rata-arta 1,275 volt sedangkan untuk warna biru rata-rata adalah 2,31 volt.

5.2 Saran

Pemisah benda berwarna merah, kuning, dan biru merupakan sebuah alat yang dapat diaplikasikan dalam suatu teknologi industri. Dalam alat ini masih banyak kekurangan yaitu dalam pemrosesan benda hanya dapat memproses satu
benda saja. Alangkah baiknya alat ini dikembangkan lagi menjadi sebuah alat pendeteksi warna yang dapat memproses benda secara kontinyu dengan memanfaatkan selang waktu pada saat pengarahan benda.

Dengan adanya pemisah benda berwarna ini diharapkan menjadi salah satu inspirasi bagi pengembangan teknologi dan dapat memberi wawasan baru bagi mahasiswa Fakultas Teknik pada khususnya dan bagi seluruh mahasiswa Universitas Sanata Dharma pada umumnya.
DAFTAR PUSTAKA


Wiliam D. Stanley, *Operational Amplifier With Linier Intergrated Crcuit*
LAMPIRAN
PENGISAH BENDA BERNARMA MERAH, KUNING DAN BIRU

SY. STEFANUS HERI TRIVANTO 085114061
MCS-51 MACRO ASSEMBLER

LOC  OBJ  LINE  SOURCE

0090   1  sens_obj  bit  p1.0
0091   2  sens_akhir  bit  p1.1
0092   3  sens_merah  bit  p1.2
0093   4  sens_kuning  bit  p1.3
0094   5  sens_biru  bit  p1.4
0095   6  limit1  bit  p1.5
0096   7  limit2  bit  p1.6
0097   8  conveyor  bit  p1.7
0098   9  ledblue  bit  p0.4
0099   10  ledyieli  bit  p0.3
009A   11  ledred  bit  p0.2
009B   12  pintal  equ  p2
009C   13  pintu2  equ  p3
009D   14  org  0h
009E   15  awal:  acall  mtunda
009F   16  clr  conveyor
00A0   17  mov  p0,#0ffh
00A1   18  mov  p2,#00h
00A2   19  mov  p3,#00h
00A3   20  mov  r2,#0cch
00A4   21  mov  r3,#0cch
00A5   22  mov  r1,#100
00A6   23  jnb  sens_obj,$
00A7   24  acall  tunda
00A8   25  jb  sens_obj,$
00A9   26  acall  tunda
00AA   27  buka_pintu:
00AB   28  mov  r0,#0ffh
00AC   29  mov  pintul,r2
00AD   30  mov  pintu2,r3
00AE   31  acall  tunda
00AF   32  mov  A,r2
00B0   33  rl  A
00B1   34  mov  r2,A
00B2   35  mov  A,r3
00B3   36  rl  A
00B4   37  mov  r3,A
00B5   38  jb  limit1,kuning
00B6   39  mov  r2,#0ffh
00B7   40  kuning:  jb  limit2,buka_pintu
00B8   41  mov  r3,#0ffh
00B9   42  jb  limit1,buka_pintu
00BA   43  mov  r2,#0ffh

L3
; deteksi keberadaan & warna obyek
;----------------------------------------------------------

cek_objek:
    jnb sens_obj,$
    acall mtunda
    setb conveyor

cek_warna:
    jnb sens_biru,$
    acall tunda
    clr conveyor
    acall mtunda
    mov A,p1
    anl A,#1ch

yell1:
    mov r1,#5

yell2: cjne A,#1ch,red1

red1:
    mov r1,#5

red2: cjne A,#14h,blue1

blue1:
    mov r1,#5

blue2: cjne A,#10h,invalid

invalid:
    setb conveyor
    acall ltunda
    clr conveyor
    ajmp cek_objek

red_obj:
    acall tunda

mov r2,#0cch
mov r0,#25

; tutup pintu merah:

mov pintu1,r2
acall tunda
mov A,r2
rr A
mov r2,A
djnz r0,tutup_pintu_merah
acall tunda

;---------------------------------------------
; cek sens akhir & tutup pintu merah
;---------------------------------------------
jnb sens_akhir,$
acall tunda
jb sens_akhir,$
acall tunda
mov r2,#0cch
buka_pintu_merah:
mov pintu1,r2
acall tunda
mov A,r2
rl A
mov r2,A
jb limit1,buka_pintu_merah
acall tunda
clr conveyor
ljmp buka_pintu

;---------------------------------------------
yell_obj:
acall tunda
mov r3,#0cch
mov r0,#32

;---------------------------------------------
tutup_pintu_kuning:
mov pintu2,r3
acall tunda
mov A,r3
rr A
mov r3,A
djnz r0,tutup_pintu_kuning
acall tunda

;---------------------------------------------
; cek sens akhir & tutup pintu kuning
;---------------------------------------------

jnb sens_akhir,$
acall tunda
jb sens_akhir,$
acall tunda
mov r3,#0cch
buka_pintu_kuning:
mov pintu2,r3
acall tunda
mov A,r3
rl A
mov r3,A
jb limit2,buka_pintu_kuning
acall tunda
clr conveyor
ljmp buka_pintu

blue_obj:
acall tunda
jnb sens_akhir,$
```
00EB 11F6  156  acall tunda
00ED 2091FD 157  jb  sens_akhir,$
00F0 11F6  158  acall tunda
00F2 C297  159  clr  conveyor
00F4 011D  160  ajmp  buka_pintu

;-------------
161          ; routine waktu tunda
162          ;-------------
163
00F6 7E3C  164  tunda:  mov  R6,#60
00F6 7FFF  165  tundal: mov  R7,#0ffh
00FA DF0E  166  djnz  R7,$
00FC DEF0  167  djnz  R6,tundal
00FF 22    168  ret
0101 1F6F  169  mtunda: mov  R4,#20
0102 1F6F  170  mtundal: acall tunda
0103 5C9E  171  djnz  R4,mtunda2
0105 22    172  ret
0106 7C3C  173  ltunda: mov  R4,#60
0108 11F6  174  ltundal: acall tunda
010A 1FF6  175  djnz  R4,ltunda2
010C 22    176  ret
177  end

SYMBOL TABLE LISTING

<table>
<thead>
<tr>
<th>NAME</th>
<th>TYPE</th>
<th>VALUE</th>
<th>ATTRIBUTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWAL</td>
<td>C</td>
<td>0000H</td>
<td>A</td>
</tr>
<tr>
<td>BLUE_OBJ</td>
<td>C</td>
<td>00E6H</td>
<td>A</td>
</tr>
<tr>
<td>BLUE1</td>
<td>C</td>
<td>0071H</td>
<td>A</td>
</tr>
<tr>
<td>BLUE2</td>
<td>C</td>
<td>0073H</td>
<td>A</td>
</tr>
<tr>
<td>BUKA_PINTU_KUNING</td>
<td>C</td>
<td>00D5H</td>
<td>A</td>
</tr>
<tr>
<td>BUKA_PINTU_MERAH</td>
<td>C</td>
<td>00A7H</td>
<td>A</td>
</tr>
<tr>
<td>BUKA_PINTU</td>
<td>C</td>
<td>0010H</td>
<td>A</td>
</tr>
<tr>
<td>CEK_OBJEK</td>
<td>C</td>
<td>0038H</td>
<td>A</td>
</tr>
<tr>
<td>CEK_WARNA</td>
<td>C</td>
<td>0042H</td>
<td>A</td>
</tr>
<tr>
<td>CONVEYOR</td>
<td>B</td>
<td>0090H</td>
<td>A</td>
</tr>
<tr>
<td>INVALID</td>
<td>C</td>
<td>0082H</td>
<td>A</td>
</tr>
<tr>
<td>KUNING</td>
<td>C</td>
<td>0031H</td>
<td>A</td>
</tr>
<tr>
<td>LEDBLUE</td>
<td>B</td>
<td>000H</td>
<td>A</td>
</tr>
<tr>
<td>LEDRED</td>
<td>B</td>
<td>0080H</td>
<td>A</td>
</tr>
<tr>
<td>LEDYEL</td>
<td>B</td>
<td>0080H</td>
<td>A</td>
</tr>
<tr>
<td>LIMIT1</td>
<td>B</td>
<td>0090H</td>
<td>A</td>
</tr>
<tr>
<td>LIMIT2</td>
<td>B</td>
<td>0090H</td>
<td>A</td>
</tr>
<tr>
<td>LTUNDA</td>
<td>C</td>
<td>0106H</td>
<td>A</td>
</tr>
<tr>
<td>LTUNDA2</td>
<td>C</td>
<td>0108H</td>
<td>A</td>
</tr>
<tr>
<td>MTUNDA</td>
<td>C</td>
<td>00FFH</td>
<td>A</td>
</tr>
<tr>
<td>MTUNDA2</td>
<td>C</td>
<td>0101H</td>
<td>A</td>
</tr>
<tr>
<td>P0</td>
<td>D</td>
<td>0080H</td>
<td>A</td>
</tr>
<tr>
<td>P1</td>
<td>D</td>
<td>0090H</td>
<td>A</td>
</tr>
<tr>
<td>P2</td>
<td>D</td>
<td>00A0H</td>
<td>A</td>
</tr>
<tr>
<td>P3</td>
<td>D</td>
<td>00B0H</td>
<td>A</td>
</tr>
<tr>
<td>PINTU1</td>
<td>D</td>
<td>00A0H</td>
<td>A</td>
</tr>
<tr>
<td>PINTU2</td>
<td>D</td>
<td>00B0H</td>
<td>A</td>
</tr>
<tr>
<td>RED_OBJ</td>
<td>C</td>
<td>008AH</td>
<td>A</td>
</tr>
<tr>
<td>REDI</td>
<td>C</td>
<td>0060H</td>
<td>A</td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>Label</th>
<th>Type</th>
<th>Address</th>
<th>Byte</th>
</tr>
</thead>
<tbody>
<tr>
<td>RED2</td>
<td>C</td>
<td>0062H</td>
<td>A</td>
</tr>
<tr>
<td>SENS_AKHIR</td>
<td>B</td>
<td>0090H.1</td>
<td>A</td>
</tr>
<tr>
<td>SENS_BIRU</td>
<td>B</td>
<td>0090H.4</td>
<td>A</td>
</tr>
<tr>
<td>SENS_KUNING</td>
<td>B</td>
<td>0090H.3</td>
<td>A</td>
</tr>
<tr>
<td>SENS_MERAH</td>
<td>B</td>
<td>0090H.2</td>
<td>A</td>
</tr>
<tr>
<td>SENS_OBJ</td>
<td>B</td>
<td>0090H.0</td>
<td>A</td>
</tr>
<tr>
<td>TUNDA</td>
<td>C</td>
<td>00F6H</td>
<td>A</td>
</tr>
<tr>
<td>TUNDA1</td>
<td>C</td>
<td>00F8H</td>
<td>A</td>
</tr>
<tr>
<td>TUTUP_PINTU_KUNING</td>
<td>C</td>
<td>00BEH</td>
<td>A</td>
</tr>
<tr>
<td>TUTUP_PINTU_MERAH</td>
<td>C</td>
<td>0090H</td>
<td>A</td>
</tr>
<tr>
<td>YELL_OBJ</td>
<td>C</td>
<td>00B8H</td>
<td>A</td>
</tr>
<tr>
<td>YELL1</td>
<td>C</td>
<td>004FH</td>
<td>A</td>
</tr>
<tr>
<td>YELL2</td>
<td>C</td>
<td>0051H</td>
<td>A</td>
</tr>
</tbody>
</table>

REGISTER BANK(S) USED: 0

ASSEMBLY COMPLETE, NO ERRORS FOUND
Features
Compatible with MCS-51™ Products
4K Bytes of In-System Reprogrammable Flash Memory
  – Endurance: 1,000 Write/Erase Cycles
  – Fully Static Operation: 0 Hz to 24 MHz
Three-level Program Memory Lock
128 x 8-bit Internal RAM
32 Programmable I/O Lines
Two 16-bit Timer/Counters
Six Interrupt Sources
Programmable Serial Channel
Low-power Idle and Power-down Modes

Description
The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4K bytes of Flash programmable and erasable read-only memory (PEROM). The device manufactured by Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard MCS-51 instruction set and pinout. The on-chip flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides highly-flexible and cost-effective solution to many embedded control applications.

In Configurations

PLCC

PDIP

PQFP/TQFP

AT89C51

Not Recommended
for New Designs.
Use AT89S51.
AT89C51

Block Diagram

- Vcc
- GND
- PORT 0 DRIVERS
- PORT 2 DRIVERS
- RAM ADDR. REGISTER
- RAM
- PORT 0 LATCH
- PORT 2 LATCH
- FLASH
- B REGISTER
- ACC
- TMP2
- TMP1
- ALU
- INTERRUPT, SERIAL PORT, AND TIMER BLOCKS
- PSW
- INSTRUCTION REGISTER
- TIMING AND CONTROL
- PORT 1 LATCH
- PORT 3 LATCH
- PORT 1 DRIVERS
- PORT 3 DRIVERS
- P0.0 - P0.7
- P0.0 - P0.7
- P2.0 - P2.7
- P3.0 - P3.7
- PROGRAM ADDRESS REGISTER
- BUFFER
- PC INCREMENTER
- PROGRAM COUNTER
- DPTR
The AT89C51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port and interrupt system to continue functioning. The Power-down Mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

Pin Description

VCC
Supply voltage.

GND
Ground.

Port 0
Port 0 is an 8-bit open-drain bi-directional 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 may 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 pullups.

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

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

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

Port 2
Port 2 is an 8-bit bi-directional I/O port with internal pullups. 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 pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (IIL) because of the internal pullups.

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, it uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ R1), Port 2 emits the contents of the P2 Special Function Register.

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

Port 3
Port 3 is an 8-bit bi-directional I/O port with internal pullups. 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 pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (IIL) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51 as listed below:

<table>
<thead>
<tr>
<th>Port Pin</th>
<th>Alternate Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>P3.0</td>
<td>RXD (serial input port)</td>
</tr>
<tr>
<td>P3.1</td>
<td>TXD (serial output port)</td>
</tr>
<tr>
<td>P3.2</td>
<td>INT0 (external interrupt 0)</td>
</tr>
<tr>
<td>P3.3</td>
<td>INT1 (external interrupt 1)</td>
</tr>
<tr>
<td>P3.4</td>
<td>T0 (timer 0 external input)</td>
</tr>
<tr>
<td>P3.5</td>
<td>T1 (timer 1 external input)</td>
</tr>
<tr>
<td>P3.6</td>
<td>WR (external data memory write strobe)</td>
</tr>
<tr>
<td>P3.7</td>
<td>RD (external data memory read strobe)</td>
</tr>
</tbody>
</table>

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

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

ALE/PROG
Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (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 MOV C instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

**PSEN**

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

When the AT89C51 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

**EA/VPP**

External Access Enable. 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, EA will be internally latched on reset.

EA should be strapped to Vcc for internal program executions.

This pin also receives the 12-volt programming enable voltage (Vpp) during Flash programming, for parts that require 12-volt Vpp.

**XTAL1**

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

**XTAL2**

Output from the inverting oscillator amplifier.

**Oscillator Characteristics**

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

**Idle Mode**

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

It should be noted that when idle 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 is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

![Figure 1. Oscillator Connections](image)

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

= 40 pF ± 10 pF for Ceramic Resonators

**Status of External Pins During Idle and Power-down Modes**

<table>
<thead>
<tr>
<th>Mode</th>
<th>Program Memory</th>
<th>ALE</th>
<th>PSEN</th>
<th>PORT0</th>
<th>PORT1</th>
<th>PORT2</th>
<th>PORT3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle</td>
<td>Internal</td>
<td>1</td>
<td>1</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Idle</td>
<td>External</td>
<td>1</td>
<td>1</td>
<td>Float</td>
<td>Data</td>
<td>Address</td>
<td>Data</td>
</tr>
<tr>
<td>Power-down</td>
<td>Internal</td>
<td>0</td>
<td>0</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Power-down</td>
<td>External</td>
<td>0</td>
<td>0</td>
<td>Float</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
</tbody>
</table>
Figure 2. External Clock Drive Configuration

NC ──── XTAL2

EXTERNAL OSCILLATOR SIGNAL

XTAL1 ──── GND

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. The only exit from power-down is a hardware reset. 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.

Program Memory Lock Bits
On the chip are three lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below.

When lock bit 1 is programmed, the logic level at the 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. It is necessary that the latched value of EA be in agreement with the current logic level at that pin in order for the device to function properly.

<table>
<thead>
<tr>
<th>Lock Bit Protection Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program Lock Bits</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>LB1</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
</tbody>
</table>

Protection Type:
- No program lock features
- MOV C instructions executed from external program memory are disabled from fetching code bytes from internal memory, EA is sampled and latched on reset, and further programming of the Flash is disabled
- Same as mode 2, also verify is disabled
- Same as mode 3, also external execution is disabled
Programming the Flash

The AT89C51 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage \( V_{OE} \) program enable signal. The low-voltage programming mode provides a convenient way to program the AT89C51 inside the user's system, while the high-voltage programming mode is compatible with conventional third-party Flash or EPROM programmers.

The AT89C51 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective op-side marking and device signature codes are listed in the following table.

<table>
<thead>
<tr>
<th></th>
<th>( V_{PP} = 12V )</th>
<th>( V_{PP} = 5V )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top-side Mark</td>
<td>AT89C51</td>
<td>AT89C51</td>
</tr>
<tr>
<td></td>
<td>xxxx</td>
<td>xxxx-5</td>
</tr>
<tr>
<td></td>
<td>yyww</td>
<td>yyww</td>
</tr>
<tr>
<td>Signature</td>
<td>(030H) = 1EH</td>
<td>(030H) = 1EH</td>
</tr>
<tr>
<td></td>
<td>(031H) = 51H</td>
<td>(031H) = 51H</td>
</tr>
<tr>
<td></td>
<td>(032H) = FFH</td>
<td>(032H) = 05H</td>
</tr>
</tbody>
</table>

The AT89C51 code memory array is programmed byte-by-byte in either programming mode. To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.

Programming Algorithm: Before programming the AT89C51, the address, data, and control signals should be set up according to the Flash programming mode table and figure 3 and figure 4. To program the AT89C51, take the following steps:

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise \( EA/V_{PP} \) to 12V for the high-voltage programming mode.
5. Pulse ALE/PROG once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

Data Polling: The AT89C51 features Data Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on P0.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

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

Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase: The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/PROG low for 10 ms. The code array is written with all "1"s. The chip erase operation must be executed before the code memory can be re-programmed.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 030H, 031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

\[
\begin{align*}
(030H) = 1EH & \text{ indicates manufactured by Atmel} \\
(031H) = 51H & \text{ indicates 89C51} \\
(032H) = FFH & \text{ indicates 12V programming} \\
(032H) = 05H & \text{ indicates 5V programming}
\end{align*}
\]

Programming Interface

Every code byte in the Flash array 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.

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

<table>
<thead>
<tr>
<th>Mode</th>
<th>RST</th>
<th>PSEN</th>
<th>ALE/PROG</th>
<th>EA/Vpp</th>
<th>P2.6</th>
<th>P2.7</th>
<th>P3.6</th>
<th>P3.7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write Code Data</td>
<td>H</td>
<td>L</td>
<td></td>
<td>H/12V</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>Read Code Data</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>Write Lock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit - 1</td>
<td>H</td>
<td>L</td>
<td></td>
<td>H/12V</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>Bit - 2</td>
<td>H</td>
<td>L</td>
<td></td>
<td>H/12V</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>Bit - 3</td>
<td>H</td>
<td>L</td>
<td></td>
<td>H/12V</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>Chip Erase</td>
<td>H</td>
<td>L</td>
<td>(1)</td>
<td>H/12V</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>Read Signature Byte</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>

Note: 1. Chip Erase requires a 10 ms PROG pulse.

Figure 3. Programming the Flash

Figure 4. Verifying the Flash
Flash Programming and Verification Waveforms - High-voltage Mode ($V_{PP} = 12V$)

- P1.0 - P1.7
- P2.0 - P2.3
- PORT 0
- ALE/PROG
- $EAV_{PP}$
- P2.7 (ENABLE)
- P3.4 (RDY/BSY)

Flash Programming and Verification Waveforms - Low-voltage Mode ($V_{PP} = 5V$)

- P1.0 - P1.7
- P2.0 - P2.3
- PORT 0
- ALE/PROG
- $EAV_{PP}$
- P2.7 (ENABLE)
- P3.4 (RDY/BSY)
Flash Programming and Verification Characteristics

$T_A = 0^\circ C$ to $70^\circ C$, $V_{CC} = 5.0 \pm 10%$

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Min</th>
<th>Max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{PP}^{(1)}$</td>
<td>Programming Enable Voltage</td>
<td>11.5</td>
<td>12.5</td>
<td>V</td>
</tr>
<tr>
<td>$I_{PP}^{(1)}$</td>
<td>Programming Enable Current</td>
<td></td>
<td>1.0</td>
<td>mA</td>
</tr>
<tr>
<td>$f_{ACLCL}$</td>
<td>Oscillator Frequency</td>
<td>3</td>
<td>24</td>
<td>MHz</td>
</tr>
<tr>
<td>$t_{AVGL}$</td>
<td>Address Setup to PROG Low</td>
<td></td>
<td>48t_{CLCL}</td>
<td></td>
</tr>
<tr>
<td>$t_{GHMAX}$</td>
<td>Address Hold after PROG</td>
<td>48t_{CLCL}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{DVGL}$</td>
<td>Data Setup to PROG Low</td>
<td>48t_{CLCL}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{GHDX}$</td>
<td>Data Hold after PROG</td>
<td>48t_{CLCL}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{EHSH}$</td>
<td>P2.7 (ENABLE) High to $V_{PP}$</td>
<td>48t_{CLCL}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{GHL}$</td>
<td>$V_{PP}$ Setup to PROG Low</td>
<td>10</td>
<td></td>
<td>µs</td>
</tr>
<tr>
<td>$t_{GHS}^{(1)}$</td>
<td>$V_{PP}$ Hold after PROG</td>
<td>10</td>
<td></td>
<td>µs</td>
</tr>
<tr>
<td>$t_{GLGH}$</td>
<td>PROG Width</td>
<td>1</td>
<td>110</td>
<td>µs</td>
</tr>
<tr>
<td>$t_{AVC4V}$</td>
<td>Address to Data Valid</td>
<td>48t_{CLCL}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{ELGV}$</td>
<td>ENABLE Low to Data Valid</td>
<td>48t_{CLCL}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{EHQZ}$</td>
<td>Data Float after ENABLE</td>
<td>0</td>
<td>48t_{CLCL}</td>
<td></td>
</tr>
<tr>
<td>$t_{GHBL}$</td>
<td>PROG High to BUSY Low</td>
<td>1.0</td>
<td></td>
<td>µs</td>
</tr>
<tr>
<td>$t_{WC}$</td>
<td>Byte Write Cycle Time</td>
<td>2.0</td>
<td></td>
<td>ms</td>
</tr>
</tbody>
</table>

Note: 1. Only used in 12-volt programming mode.
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

$T_A = -40°C$ to $85°C$, $V_{CC} = 5.0V \pm 20\%$ (unless otherwise noted)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Condition</th>
<th>Min</th>
<th>Max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{IL}$</td>
<td>Input Low-voltage</td>
<td>(Except E(A))</td>
<td>-0.5</td>
<td>0.2 $V_{CC}$</td>
<td>V</td>
</tr>
<tr>
<td>$I_{L1}$</td>
<td>Input Low-voltage (E(A))</td>
<td></td>
<td>-0.5</td>
<td>0.2 $V_{CC}$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{IH}$</td>
<td>Input High-voltage</td>
<td>(Except XTAL1, RST)</td>
<td>0.2 $V_{CC}$</td>
<td>$V_{CC} + 0.5$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{IN}$</td>
<td>Input High-voltage</td>
<td>(XTAL1, RST)</td>
<td>0.7 $V_{CC}$</td>
<td>$V_{CC} + 0.5$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{OL}$</td>
<td>Output Low-voltage(^{(1)}) (Ports 1, 2, 3)</td>
<td>$I_{OL} = 1.6 mA$</td>
<td>0.45</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{OHL}$</td>
<td>Output Low-voltage (Port 0, ALE, PSE(N))</td>
<td>$I_{OL} = 3.2 mA$</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{OH}$</td>
<td>Output High-voltage (Ports 1, 2, 3, ALE, PSE(N))</td>
<td>$I_{OH} = -50 \mu A$, $V_{CC} = 5V \pm 10%$</td>
<td>2.4</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$I_{OH} = -25 \mu A$</td>
<td>0.75 $V_{CC}$</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$I_{OH} = -10 \mu A$</td>
<td>0.9 $V_{CC}$</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{OHN}$</td>
<td>Output High-voltage (Port 0 in External Bus Mode)</td>
<td>$I_{OH} = -1000 \mu A$, $V_{CC} = 5V \pm 10%$</td>
<td>2.4</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$I_{OH} = -300 \mu A$</td>
<td>0.75 $V_{CC}$</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$I_{OH} = -80 \mu A$</td>
<td>0.9 $V_{CC}$</td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$I_{L1}$</td>
<td>Logical 0 Input Current (Ports 1, 2, 3)</td>
<td>$V_{IN} = 6.45V$</td>
<td></td>
<td>$-50$</td>
<td>(\mu A)</td>
</tr>
<tr>
<td>$I_{L2}$</td>
<td>Logical 1 to 0 Transition Current (Ports 1, 2, 3)</td>
<td>$V_{VP} = 2V$, $V_{CC} = 5V \pm 10%$</td>
<td></td>
<td>$-650$</td>
<td>(\mu A)</td>
</tr>
<tr>
<td>$I_{L1}$</td>
<td>Input Leakage Current (Port 0, E(A))</td>
<td>$0.45 &lt; V_{IN} &lt; V_{CC}$</td>
<td></td>
<td></td>
<td>(\mu A)</td>
</tr>
<tr>
<td>RRST</td>
<td>Reset Pull-down Resistor</td>
<td></td>
<td>50</td>
<td>300</td>
<td>K(\Omega)</td>
</tr>
<tr>
<td>$C_{D}$</td>
<td>Pin Capacitance</td>
<td>Test Freq. = 1 MHz, $T_A = 25°C$</td>
<td></td>
<td>10</td>
<td>p(\text{F})</td>
</tr>
<tr>
<td>$I_{CC}$</td>
<td>Power Supply Current</td>
<td>Active Mode, 12 MHz</td>
<td></td>
<td>5</td>
<td>mA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Idle Mode, 12 MHz</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td></td>
<td>Power-down Mode(^{(2)})</td>
<td>$V_{CC} = 3V$</td>
<td></td>
<td>100</td>
<td>(\mu A)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$V_{CC} = 6V$</td>
<td></td>
<td>40</td>
<td>(\mu A)</td>
</tr>
</tbody>
</table>

Notes:
1. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:
   - Maximum $I_{OL}$ per Port 0: 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/PROG, and PSEN = 100 pF; load capacitance for all other outputs = 80 pF.

### External Program and Data Memory Characteristics

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>12 MHz Oscillator</th>
<th>16 to 24 MHz Oscillator</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Min</td>
<td>Max</td>
<td>Min</td>
</tr>
<tr>
<td>t1/CCL</td>
<td>Oscillator Frequency</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tHLL</td>
<td>ALE Pulse Width</td>
<td>127</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>tAVLL</td>
<td>Address Valid to ALE Low</td>
<td>43</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tLAX</td>
<td>Address Hold after ALE Low</td>
<td>48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tLLIV</td>
<td>ALE Low to Valid Instruction In</td>
<td>233</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>tLPL</td>
<td>ALE Low to PSEN Low</td>
<td>43</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tPLPH</td>
<td>PSEN Pulse Width</td>
<td>205</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>tPLV</td>
<td>PSEN Low to Valid Instruction In</td>
<td>145</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tPIXI</td>
<td>Input Instruction Hold after PSEN</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tPZX</td>
<td>Input Instruction Float after PSEN</td>
<td>59</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tPAV</td>
<td>PSEN to Address Valid</td>
<td>75</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>tAWIV</td>
<td>Address to Valid Instruction In</td>
<td>312</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tPAZ</td>
<td>PSEN Low to Address Float</td>
<td>10</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>tRLWH</td>
<td>RD Pulse Width</td>
<td>400</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>tRLDV</td>
<td>WR Pulse Width</td>
<td>400</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>tRLDV</td>
<td>RD Low to Valid Data In</td>
<td></td>
<td>252</td>
<td></td>
</tr>
<tr>
<td>tRDOX</td>
<td>Data Hold after RD</td>
<td>0</td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>tRDZ</td>
<td>Data Float after RD</td>
<td>97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tLADV</td>
<td>ALE Low to Valid Data In</td>
<td>517</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>tLADV</td>
<td>Address to Valid Data In</td>
<td>585</td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>tLWL</td>
<td>ALE Low to RD or WR Low</td>
<td>200</td>
<td>300</td>
<td>3</td>
</tr>
<tr>
<td>tAVWL</td>
<td>Address to RD or WR Low</td>
<td>203</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tQQVX</td>
<td>Data Valid to WR Transition</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tQVWH</td>
<td>Data Valid to WR High</td>
<td>433</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tWGHX</td>
<td>Data Hold after WR</td>
<td>33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tRLH</td>
<td>RD Low to Address Float</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>tWHLH</td>
<td>RD or WR High to ALE High</td>
<td>43</td>
<td>123</td>
<td>0</td>
</tr>
</tbody>
</table>
External Program Memory Read Cycle

External Data Memory Read Cycle

AT89C51
External Data Memory Write Cycle

External Clock Drive Waveforms

External Clock Drive

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Min</th>
<th>Max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>t_{CLCL}</td>
<td>Oscillator Frequency</td>
<td>0</td>
<td>24</td>
<td>MHz</td>
</tr>
<tr>
<td>t_{CLC}</td>
<td>Clock Period</td>
<td>41.6</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>t_{CHCX}</td>
<td>High Time</td>
<td>15</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>t_{CLCX}</td>
<td>Low Time</td>
<td>15</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>t_{CLCH}</td>
<td>Rise Time</td>
<td>20</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>t_{CHCL}</td>
<td>Fall Time</td>
<td>20</td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>
Serial Port Timing: Shift Register Mode Test Conditions

\( V_{CC} = 5.0 \text{ V} \pm 20\% \); Load Capacitance = 80 pF

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>12 MHz Osc</th>
<th>Variable Oscillator</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{XXL} )</td>
<td>Serial Port Clock Cycle Time</td>
<td>Min: 1.0</td>
<td>Max: 12( t_{CLL} )</td>
<td>( \mu \text{s} )</td>
</tr>
<tr>
<td>( t_{QXH} )</td>
<td>Output Data Setup to Clock Rising Edge</td>
<td>Min: 700</td>
<td>Max: 10( t_{CLL} )-133</td>
<td>( \text{ns} )</td>
</tr>
<tr>
<td>( t_{XHDX} )</td>
<td>Output Data Hold after Clock Rising Edge</td>
<td>Min: 50</td>
<td>Max: 2( t_{CLL} )-117</td>
<td>( \text{ns} )</td>
</tr>
<tr>
<td>( t_{XHDV} )</td>
<td>Input Data Hold after Clock Rising Edge</td>
<td>Min: 0</td>
<td>Max: 0</td>
<td>( \text{ns} )</td>
</tr>
<tr>
<td>( t_{KLXV} )</td>
<td>Clock Rising Edge to Input Data Valid</td>
<td>Min: 700</td>
<td>Max: 10( t_{CLL} )-133</td>
<td>( \text{ns} )</td>
</tr>
</tbody>
</table>

Shift Register Mode Timing Waveforms

AC Testing Input/Output Waveforms\(^{(1)}\)

\[ V_{CC} - 0.5 \text{ V} \]

\[ 0.2 V_{CC} + 0.8 \text{ V} \]

\[ 0.2 V_{CC} - 0.1 \text{ V} \]

TEST POINTS

0.45V

Note: 1. AC inputs during testing are driven at \( V_{CC} - 0.5 \text{ V} \) 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\(^{(1)}\)

\[ V_{LOAD} + 0.1 \text{ V} \]

\[ V_{OL} - 0.1 \text{ V} \]

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 100 mV change from the loaded \( V_{OH}/V_{OL} \) level occurs.
## Ordering Information

<table>
<thead>
<tr>
<th>Speed (MHz)</th>
<th>Power Supply</th>
<th>Ordering Code</th>
<th>Package</th>
<th>Operation Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>5V ± 20%</td>
<td>AT89C51-12AC</td>
<td>44A</td>
<td>Commercial (0°C to 70°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12JC</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12PC</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12QC</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12AI</td>
<td>44A</td>
<td>Industrial (-40°C to 85°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12JI</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12PI</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-12QI</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>5V ± 20%</td>
<td>AT89C51-16AC</td>
<td>44A</td>
<td>Commercial (0°C to 70°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16JC</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16PC</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16QC</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16AI</td>
<td>44A</td>
<td>Industrial (-40°C to 85°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16JI</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16PI</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-16QI</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>5V ± 20%</td>
<td>AT89C51-20AC</td>
<td>44A</td>
<td>Commercial (0°C to 70°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20JC</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20PC</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20QC</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20AI</td>
<td>44A</td>
<td>Industrial (-40°C to 85°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20JI</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20PI</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-20QI</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>5V ± 20%</td>
<td>AT89C51-24AC</td>
<td>44A</td>
<td>Commercial (0°C to 70°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24JC</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24PC</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24QC</td>
<td>44Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24AI</td>
<td>44A</td>
<td>Industrial (-40°C to 85°C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24JI</td>
<td>44J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24PI</td>
<td>40P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT89C51-24QI</td>
<td>44Q</td>
<td></td>
</tr>
</tbody>
</table>

## Package Type

- **44A**: 44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
- **44J**: 44-lead, Plastic J-lead Chip Carrier (PLCC)
- **40P6**: 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
- **44Q**: 44-lead, Plastic Gull Wing Quad Flatpack (PQFP)
Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flatpack (TQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-026 ACB

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

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

Controlling dimension: millimeters

40P6, 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)

AT89C51
OCTAL BUFFER/LINE DRIVER WITH 3-STATE OUTPUTS

The SN54/74LS540 and SN54/74LS541 are octal buffers and line drivers with the same functions as the LS240 and LS241, but with pinouts on the opposite side of the package.

These device types are designed to be used as memory address drivers, clock drivers and bus-oriented transmitters/receivers. These devices are especially useful as output ports for the microprocessors, allowing ease of layout and greater PC board density.

- Hysteresis at Inputs to Improve Noise Margin
- PNP Inputs Reduce Loading
- 3-State Outputs Drive Bus Lines
- Inputs and Outputs Opposite Sides of Package, Allowing Easier Interface to Microprocessors
- Input Clamp Diodes Limit High-Speed Termination Effects

LOGIC AND CONNECTION DIAGRAMS DIP (TOP VIEW)

SN54/74LS540

SN54/74LS541

GUARANTEED OPERATING RANGES

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>Supply Voltage</td>
<td>54</td>
<td>4.5</td>
<td>5.0</td>
<td>5.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>74</td>
<td>4.75</td>
<td>5.0</td>
<td>5.25</td>
</tr>
<tr>
<td>TA</td>
<td>Operating Ambient Temperature Range</td>
<td>54</td>
<td>-55</td>
<td>25</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td></td>
<td>74</td>
<td>0</td>
<td>25</td>
<td>70</td>
</tr>
<tr>
<td>IOH</td>
<td>Output Current — High</td>
<td>54</td>
<td>-12</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>74</td>
<td>-15</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>IOL</td>
<td>Output Current — Low</td>
<td>54</td>
<td></td>
<td>12</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>74</td>
<td></td>
<td>24</td>
<td></td>
</tr>
</tbody>
</table>

ORDERING INFORMATION

SN54LSXXXJ Ceramic
SN74LSXXXN Plastic
SN74LSXXXDW SOIC

FAST AND LS TTL DATA

5-1
## Block Diagram

### Inputs
- **E1**
- **E2**

### Outputs
- **LS540**
- **LS541**

### DC Characteristics Over Operating Temperature Range

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Limits</th>
<th>Test Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_{IH} )</td>
<td>Input HIGH Voltage</td>
<td>Min: 2.0 Typ:</td>
<td>Guaranteed Input HIGH Voltage for All Inputs</td>
</tr>
<tr>
<td>( V_{IL} )</td>
<td>Input LOW Voltage</td>
<td>0.4</td>
<td>Guaranteed Input LOW Voltage for All Inputs</td>
</tr>
<tr>
<td>( V_{CC} )</td>
<td>Input Clamp Diode Voltage</td>
<td>Min: -0.85 Typ: -1.5</td>
<td>( V_{CC} = \text{MIN}, I_{IN} = -18 \text{ mA} )</td>
</tr>
<tr>
<td>( V_{OH} )</td>
<td>Output HIGH Voltage</td>
<td>Min: 4 Typ: 3.4</td>
<td>( V_{CC} = \text{MIN}, I_{OH} = -3.0 \text{ mA} )</td>
</tr>
<tr>
<td>( V_{OL} )</td>
<td>Output LOW Voltage</td>
<td>Min: 0.25 Typ: 0.4</td>
<td>( I_{OL} = 12 \text{ mA} )</td>
</tr>
<tr>
<td>( V_{T\uparrow \downarrow V_{T\uparrow \downarrow}} )</td>
<td>Hysteresis</td>
<td>Min: 0.2 Typ: 0.4</td>
<td>( V_{CC} = \text{MIN} )</td>
</tr>
<tr>
<td>( I_{OZH} )</td>
<td>Output Off Current HIGH</td>
<td>Min: 20 Typ:</td>
<td>( V_{CC} = \text{MAX}, V_{OUT} = 2.7 \text{ V} )</td>
</tr>
<tr>
<td>( I_{OZL} )</td>
<td>Output Off Current LOW</td>
<td>Min: -20 Typ:</td>
<td>( V_{CC} = \text{MAX}, V_{OUT} = 0.4 \text{ V} )</td>
</tr>
<tr>
<td>( I_{IH} )</td>
<td>Input HIGH Current</td>
<td>Min: 20 Typ:</td>
<td>( V_{CC} = \text{MAX}, V_{IN} = 2.7 \text{ V} )</td>
</tr>
<tr>
<td>( I_{IL} )</td>
<td>Input LOW Current</td>
<td>Min: 0.1 Typ:</td>
<td>( V_{CC} = \text{MAX}, V_{IN} = 7.0 \text{ V} )</td>
</tr>
<tr>
<td>( I_{OS} )</td>
<td>Short Circuit Current (Note 1)</td>
<td>Min: -40 Typ: -225</td>
<td>( V_{CC} = \text{MAX} )</td>
</tr>
</tbody>
</table>

### Note 5:
Not more than one output should be shorted at a time, nor for more than 1 second.

### Fast and LS TTL Data

**5-2**
AC CHARACTERISTICS ($T_A = 25^\circ\text{C}$)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Limits</th>
<th>Unit</th>
<th>Test Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{PLH}$</td>
<td>Propagation Delay, Data to Output</td>
<td>9.0 15</td>
<td>ns</td>
<td>$V_{CC} = 5.0\text{ V}$ $C_L = 45\text{ pF}$ R$_L$ = 687 $\Omega$</td>
</tr>
<tr>
<td>$t_{PLH}$</td>
<td></td>
<td>12 15</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PHL}$</td>
<td></td>
<td>12 15</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PL}$</td>
<td></td>
<td>12 18</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PZH}$</td>
<td>Output Enable Time to HIGH Level</td>
<td>15 25</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PZH}$</td>
<td></td>
<td>15 32</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PZL}$</td>
<td>Output Enable Time to LOW Level</td>
<td>20 38</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PZL}$</td>
<td></td>
<td>20 38</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PHZ}$</td>
<td>Output Disable Time to HIGH Level</td>
<td>10 18</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PHZ}$</td>
<td></td>
<td>10 18</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>$t_{PLZ}$</td>
<td>Output Disable Time to LOW Level</td>
<td>15 28</td>
<td>ns</td>
<td>$C_L = 5.0\text{ pF}$</td>
</tr>
<tr>
<td>$t_{PLZ}$</td>
<td></td>
<td>15 28</td>
<td>ns</td>
<td></td>
</tr>
</tbody>
</table>

AC WAVEFORMS

Figure 1

Figure 2

Figure 3

Figure 4

SWITCH POSITIONS

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>SW1</th>
<th>SW2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{PZH}$</td>
<td>Open</td>
<td>Closed</td>
</tr>
<tr>
<td>$t_{PZL}$</td>
<td>Closed</td>
<td>Open</td>
</tr>
<tr>
<td>$t_{PLZ}$</td>
<td>Closed</td>
<td>Closed</td>
</tr>
<tr>
<td>$t_{PHZ}$</td>
<td>Closed</td>
<td>Closed</td>
</tr>
</tbody>
</table>

Figure 5

FAST AND LS TTL DATA

5-3
EIGHT DARLINGTON ARRAYS

- Eight Darlington transistors with common emitters and integral suppression diodes for inductive loads.
- Each Darlington features a peak load current rating of 600mA (500mA continuous) and can withstand at least 50V in the off state.
- Outputs may be paralleled for higher current capability.

Five versions are available to simplify interfacing to standard logic families: the ULN2801A is designed for general-purpose applications with a current limit resistor; the ULN2802A has a 10.5kΩ input resistor and zener for 14-25V CMOS; the ULN2803A has a 2.7kΩ input resistor for 5V TTL and CMOS; the ULN2804A has a 10.5kΩ input resistor for 6-15V CMOS and the ULN2805A is designed to sink a minimum of 350mA for standard and Schottky TTL where higher output current is required.

All types are supplied in a 18-lead plastic DIP with a copper lead frame and feature the convenient input-opposite-output pinout to simplify board layout.

September 1997
ULN2801A - ULN2802A - ULN2803A - ULN2804A - ULN2805A

SCHEMATIC DIAGRAM AND ORDER CODES

For ULN2801A (each driver for PMOS-CMOS)

For ULN2802A (each driver for 14-15 V PMOS)

For ULN2803A (each driver for 5 V, TTL/CMOS)

For ULN2804A (each driver for 6-15 V CMOS/PMOS)

For ULN2805A (each driver for high out TTL)
**ULN2801A - ULN2802A - ULN2803A - ULN2804A - ULN2805A**

### ABSOLUTE MAXIMUM RATINGS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter Description</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_D$</td>
<td>Output Voltage</td>
<td>50</td>
<td>V</td>
</tr>
<tr>
<td>$V_I$</td>
<td>Input Voltage</td>
<td>30</td>
<td>V</td>
</tr>
<tr>
<td>$I_{C}$</td>
<td>Continuous Collector Current</td>
<td>500</td>
<td>mA</td>
</tr>
<tr>
<td>$I_B$</td>
<td>Continuous Base Current</td>
<td>25</td>
<td>mA</td>
</tr>
<tr>
<td>$P_{D}$</td>
<td>Power Dissipation (one Darlington pair)</td>
<td>1.0</td>
<td>W</td>
</tr>
<tr>
<td>$T_{amb}$</td>
<td>Operating Ambient Temperature Range</td>
<td>-20 to 85</td>
<td>°C</td>
</tr>
<tr>
<td>$T_{stg}$</td>
<td>Storage Temperature Range</td>
<td>-55 to 150</td>
<td>°C</td>
</tr>
<tr>
<td>$T_J$</td>
<td>Junction Temperature Range</td>
<td>-20 to 150</td>
<td>°C</td>
</tr>
</tbody>
</table>

### THERMAL DATA

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter Description</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{th j-amb}$</td>
<td>Thermal Resistance Junction-ambient Max.</td>
<td>55</td>
<td>°C/W</td>
</tr>
</tbody>
</table>

### ELECTRICAL CHARACTERISTICS ($T_{amb} = 25°C$ unless otherwise specified)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter Description</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
<th>Fig.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{CEX}$</td>
<td>Output Leakage Current</td>
<td>$V_{CE} = 50V$, $V_{amb} = 70°C$, $V_{CEX} = 50V$</td>
<td>50</td>
<td>100</td>
<td>µA</td>
<td>1a</td>
<td>1a</td>
</tr>
<tr>
<td>$V_{CEX}$</td>
<td>Collector-emitter Saturation Voltage</td>
<td>$I_{C} = 100mA$, $I_{B} = 250µA$</td>
<td>0.9</td>
<td>1.1</td>
<td>V</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>$I_{on}$</td>
<td>Input Current</td>
<td>$V_{CE} = 2V$, $I_{C} = 500µA$</td>
<td>50</td>
<td>65</td>
<td>µA</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

---

3/8
Figure 8: Collector Current as a Function of
Saturation Voltage.

Figure 9: Collector Current as a Function of
Input Current.

Figure 10: Allowable Average Power Dissipation
as a Function of Ambient Temperature.

Figure 11: Peak Collector Current as a Function
of Duty Cycle.

Figure 12: Peak Collector Current as a Function
of Duty.

Figure 13: Input Current as a Function of Input
Voltage (for ULN2802A).
Figure 14: Input Current as a Function of Input Voltage (for ULN2804A)

Figure 15: Input Current as a Function of Input Voltage (for ULN2803A)

Figure 16: Input Current as a Function of Input Voltage (for ULN2805A)
### DIP18 PACKAGE MECHANICAL DATA

<table>
<thead>
<tr>
<th>DIM.</th>
<th>mm</th>
<th>inch</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MIN.</td>
<td>TYP.</td>
</tr>
<tr>
<td>a1</td>
<td>0.254</td>
<td>0.010</td>
</tr>
<tr>
<td>B</td>
<td>1.39</td>
<td>1.65</td>
</tr>
<tr>
<td>b</td>
<td>0.46</td>
<td></td>
</tr>
<tr>
<td>b1</td>
<td>0.25</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>23.24</td>
<td>0.915</td>
</tr>
<tr>
<td>E</td>
<td>8.5</td>
<td>0.335</td>
</tr>
<tr>
<td>e</td>
<td>2.54</td>
<td>0.100</td>
</tr>
<tr>
<td>e3</td>
<td>20.32</td>
<td>0.800</td>
</tr>
<tr>
<td>F</td>
<td>7.1</td>
<td>0.280</td>
</tr>
<tr>
<td>I</td>
<td>3.93</td>
<td>0.155</td>
</tr>
<tr>
<td>L</td>
<td>3.3</td>
<td>0.130</td>
</tr>
<tr>
<td>Z</td>
<td>1.27</td>
<td>0.050</td>
</tr>
</tbody>
</table>
COMPLEMENTARY SILICON POWER DARLINGTON TRANSISTORS

- STMicroelectronics PREFERRED SALES TYPES
- COMPLEMENTARY PNP - NPN DEVICES
- MONOLITHIC DARLINGTON CONFIGURATION
- INTEGRATED ANTI-PARALLEL COLLECTOR-EMITTER DIODE

APPLICATION
- LINEAR AND SWITCHING INDUSTRIAL EQUIPMENT

DESCRIPTION
The BD677, BD677A, BD679, BD679A and BD681 are silicon Epitaxial-Base NPN power transistors in monolithic Darlington configuration mounted in Jeteck SOT-32 plastic package.
They are intended for use in medium power linear and switching applications.
The complementary PNP types are BD678, BD678A, BD680, BD680A and BD682 respectively.

ABSOLUTE MAXIMUM RATINGS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NPN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BD677/A</td>
<td>BD679/A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BD678/A</td>
<td>BD680/A</td>
</tr>
<tr>
<td>VCEo</td>
<td>Collector-Base Voltage (Ic = 0)</td>
<td>60</td>
<td>80</td>
</tr>
<tr>
<td>VCEO</td>
<td>Collector-Emitter Voltage (Ib = 0)</td>
<td>60</td>
<td>80</td>
</tr>
<tr>
<td>VCEO</td>
<td>Emitter-Base Voltage (Ic = 0)</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>IC</td>
<td>Collector Current</td>
<td>4</td>
<td>A</td>
</tr>
<tr>
<td>ICM</td>
<td>Collector Peak Current</td>
<td>6</td>
<td>A</td>
</tr>
<tr>
<td>IB</td>
<td>Base Current</td>
<td>0.1</td>
<td>A</td>
</tr>
<tr>
<td>PD</td>
<td>Total Dissipation at Ta ≤ 25 °C</td>
<td>40</td>
<td>W</td>
</tr>
<tr>
<td>TSS</td>
<td>Storage Temperature</td>
<td>-65 to 150</td>
<td>°C</td>
</tr>
<tr>
<td>TJ</td>
<td>Max. Operating Junction Temperature</td>
<td>150</td>
<td>°C</td>
</tr>
</tbody>
</table>

For PNP types: voltage and current values are negative.

November 2003
**THERMAL DATA**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{thJC}$</td>
<td>Thermal Resistance Junction-case</td>
<td>$V_{CE} =$ rated $V_{CEO}$, $T_C = 100 , ^\circ C$</td>
<td>3.12</td>
<td></td>
<td></td>
<td>$^\circ C/W$</td>
</tr>
<tr>
<td>$R_{thJA}$</td>
<td>Thermal Resistance Junction-ambient</td>
<td>$V_{CE} =$ rated $V_{CEO}$</td>
<td>100</td>
<td></td>
<td></td>
<td>$^\circ C/W$</td>
</tr>
</tbody>
</table>

**ELECTRICAL CHARACTERISTICS** ($T_{case} = 25 \, ^\circ C$ unless otherwise specified)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{CEO}$</td>
<td>Collector Cut-off Current ($I_E = 0$)</td>
<td>$V_{CE} =$ half rated $V_{CEO}$</td>
<td>0.5</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>$I_{CEO}$</td>
<td>Collector Cut-off Current ($I_E = 0$)</td>
<td>$V_{CE} =$ rated $V_{CEO}$</td>
<td>2</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>$I_{CEO}$</td>
<td>Emitter Cut-off Current ($I_C = 0$)</td>
<td>$V_{BE} =$ 5 V</td>
<td>2</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{CEO}$</td>
<td>Collector-Emitter Sustaining Voltage ($I_E = 0$)</td>
<td>$I_E = 50 , mA$</td>
<td>60</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{CEO}$</td>
<td>Collector-Emitter Sustaining Voltage ($I_E = 0$)</td>
<td>for BD677/BD77A/BD78/BD78A</td>
<td>90</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{CEO}$</td>
<td>Collector-Emitter Sustaining Voltage ($I_E = 0$)</td>
<td>for BD679/BD77A/BD80/BD80A</td>
<td>190</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{CEO}$</td>
<td>Collector-Emitter Sustaining Voltage ($I_E = 0$)</td>
<td>for BD681/BD82</td>
<td>2.5</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$V_{BE}$</td>
<td>Base-Emitter Voltage</td>
<td>$I_C = 1.5 , A$, $V_{CE} = 3 , V$</td>
<td>2.5</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>$h_{FE}$</td>
<td>DC Current Gain</td>
<td>$I_C = 1.5 , A$, $V_{CE} = 3 , V$</td>
<td>750</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$h_{fe}$</td>
<td>Small Signal Current Gain</td>
<td>$I_C = 1.5 , A$, $V_{CEO} = 3 , V$, $f = 1 , MHz$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Pulsed: Pulse duration = 150 μs, duty cycle 1.5 %

For NPN types voltage and current values are negative.

---

**Safe Operating Areas**

**Derating Curve**
## SOT-32 (TO-126) Mechanical Data

<table>
<thead>
<tr>
<th>DIM.</th>
<th>mm</th>
<th>inch</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MIN.</td>
<td>MAX.</td>
</tr>
<tr>
<td>A</td>
<td>7.4</td>
<td>7.8</td>
</tr>
<tr>
<td>B</td>
<td>10.5</td>
<td>10.8</td>
</tr>
<tr>
<td>b</td>
<td>0.7</td>
<td>0.9</td>
</tr>
<tr>
<td>b1</td>
<td>0.40</td>
<td>0.65</td>
</tr>
<tr>
<td>C</td>
<td>2.4</td>
<td>2.7</td>
</tr>
<tr>
<td>c1</td>
<td>1.0</td>
<td>1.3</td>
</tr>
<tr>
<td>D</td>
<td>15.4</td>
<td>16.0</td>
</tr>
<tr>
<td>e</td>
<td>2.2</td>
<td></td>
</tr>
<tr>
<td>e3</td>
<td>4.4</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>3.8</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>3</td>
<td>3.2</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>2.54</td>
</tr>
<tr>
<td>H2</td>
<td>2.15</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>1.27</td>
<td></td>
</tr>
<tr>
<td>O</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>10°</td>
<td></td>
</tr>
</tbody>
</table>

### Diagram

1: Base  
2: Collector  
3: Emitter