

**PEMBUATAN *PROTOTYPE* PENGATUR KETINGGIAN  
MISTAR PADA OLAHRAGA LOMPAT GALAH  
DENGAN PAPAN *DISPLAY***

**SKRIPSI**

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Teknik



Disusun Oleh:

**EKA PUSPA NINGRUM**

**NIM. 0410632011 - 63**

**DEPARTEMEN PENDIDIKAN NASIONAL  
UNIVERSITAS BRAWIJAYA  
FAKULTAS TEKNIK  
MALANG  
2006**

**PEMBUATAN *PROTOTYPE* PENGATUR KETINGGIAN  
MISTAR PADA OLAHRAGA LOMPAT GALAH  
DENGAN PAPAN *DISPLAY***

**SKRIPSI**

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Teknik



Disusun Oleh:

**EKA PUSPA NINGRUM**

**NIM. 0410632011 - 63**

Telah Diperiksa dan Disetujui oleh

Dosen Pembimbing

Ir. Nurussa'adah  
NIP. 131 994 339

Moch. Rif'an, ST,MT  
NIP. 132 283 659

**PEMBUATAN *PROTOTYPE* PENGATUR KETINGGIAN  
MISTAR PADA OLAHRAGA LOMPAT GALAH  
DENGAN PAPAN *DISPLAY***

Disusun Oleh :

**EKA PUSPA NINGRUM**

**NIM. 0410632011 – 63**

Skripsi ini telah diuji dan dinyatakan lulus pada  
Tanggal 15 Agustus 2006



Dosen Penguji

M. Julius St., Ir. MT  
NIP. 131 124 655

Bambang Siswojo, Ir  
NIP. 131 759 588

Ir. Ponco Siwindarto, MS  
NIP. 131 837 966

Panca Mudjirahardjo, ST, MT  
NIP. 132 288 163

Mengetahui  
Ketua Jurusan

Ir. Purwanto, MT  
NIP. 131 574 847

## ABSTRAK

Eka Puspa Ningrum. 0410632011-63. 2006. *Pembuatan Prototype Pengatur Ketinggian Mistar Lompat Galah Dengan Papan Display*. Skripsi Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya.

Pembimbing: Ir. Nurussa'adah dan Moch. Rif'an, ST, MT.

Sejalan dengan perkembangan ilmu pengetahuan dan teknologi, keadaan ini bagi kita manusia sering memerlukan gerak badan yang secukupnya dan komunikasi yang tidak dipaksakan. Dengan demikian maka bagi kita diperlukan olahraga, olahraga sebagai program kesegaran yang terarah, olahraga sebagai kegiatan mengisi waktu, ataupun sebagai permainan. Salah satu cabang olahraga adalah atletik dimana terdiri dari bermacam-macam olahraga, misalnya saja lompat galah. Sistem dari olahraga ini dimana setiap pergantian pelompat harus merubah posisi mistar naik atau turun. Alat yang ada saat ini dengan memutar sisi kanan dan kiri secara bersamaan untuk menghindari adanya ketinggian yang tidak sama dan kemudian dicek kembali apakah sudah pada posisi yang diinginkan

Oleh karena itu perlu adanya suatu alat yang dapat mengatur ketinggian mistar pada olah raga lompat galah tersebut, sehingga dengan alat ini diharapkan memberikan kemudahan bagi masyarakat dalam beraktifitas. Maka dibuat suatu prototype yang dapat mengatur ketinggian mistar pada olahraga tersebut.

Prototype ini terdiri dari sensor optocoupler sebagai counter yang dapat mendeteksi ketinggian yang diinginkan. Keypad sebagai masukkan data, motor yang digunakan untuk menaikkan dan menurunkan penompang mistar dan alat ini dikontrol oleh mikrokontroller dengan *display* LCD 16x2 sebagai penunjuk informasi untuk memasukkan data yang diinginkan dan *seven segmen* sebagai tampilan bagi penonton.

Prototype ini didesain mampu mengukur ketinggian 50 cm dengan kenaikan dan penurunan minimal 1 cm. Secara keseluruhan prototype yang dibuat ini memiliki tingkat kesalahan pembacaan rata-rata sebesar 0,248%.

**Kata kunci:** *Alat ukur, Ketinggian Mistar, Lompat Galah*

## PENGANTAR

Syukur Alhamdulillah, penulis panjatkan atas kehadiran Allah SWT karena berkat rahmat dan karunia-Nya jualah penulis dapat menyelesaikan skripsi dengan judul "Pembuatan *Prototype* Pengatur Ketinggian Mistar Lompat Galah dengan Papan *Display*". Pengajuan skripsi ini merupakan persyaratan yang wajib ditempuh untuk meraih gelar Sarjana Teknik. Penulis menyadari bahwa tugas akhir ini masih jauh dari sempurna, oleh karena itu penulis berharap semoga ada pengembangan untuk dapat menyempurnakan tugas akhir ini, dan semoga tugas akhir ini dapat bermanfaat bagi siapapun yang memerlukannya.

Dalam kesempatan ini penulis mengucapkan rasa terimakasih yang sebesar-besarnya kepada :

1. Ibu Ir. Nurussa'adah., selaku Pembimbing I
2. Bapak Moch. Rif'an, ST, MT., selaku Pembimbing II

Yang telah memberikan bimbingan, pengarahan, dan nasehat kepada penulis dalam menyelesaikan Laporan skripsi ini.

Penulis juga mengucapkan terima kasih atas bantuan dan kesempatan yang telah diberikan sehingga penulis dapat menyelesaikan studi di Universitas Brawijaya, kepada :

1. Bapak Ir. Purwanto, MT., selaku Ketua Jurusan Teknik Elektro
2. Bapak Ir. Hery Purnomo, selaku Sekretaris Jurusan Teknik Elektro
3. Bapak Ir. Ponco Siwindarto, MS., selaku Ketua Kelompok Dosen Keahlian Bidang Elektronika
4. Ir. M. Julius St. MT selaku dosen penguji yang telah memberi masukan, kritik dan saran membangun bagi penulis.
5. Bambang Siswojo, Ir selaku dosen penguji yang telah memberi masukan, kritik dan saran membangun bagi penulis.
6. Ir. Ponco Siwindarto, MS selaku dosen penguji yang telah memberi masukan, kritik dan saran membangun bagi penulis.
7. Panca Mudjirahardjo, ST, MT selaku dosen penguji yang telah memberi masukan, kritik dan saran membangun bagi penulis.
8. Seluruh Dosen, Laboran, dan Staf pada Jurusan Teknik Elektro Universitas Brawijaya
9. Seluruh mahasiswa Teknik Elektro terutama rekan-rekan Alih Program 2004, yang telah membantu hingga Skripsi ini dapat selesai tepat pada waktunya.
10. Terimakasih yang tak terhingga untuk keluargaku yang senantiasa memberikan pengertian, perhatian, kasih sayang, dan doa.

Akhir kata, penulis berharap laporan tugas akhir ini dapat memberikan manfaat bagi siapa saja yang membacanya. Amin

Malang, Agustus 2006  
Penulis,

DAFTAR ISI

	Halaman
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PERSETUJUAN DOSEN PEMBIMBING</b> .....	ii
<b>LEMBAR PENGESAHAN</b> .....	iii
<b>ABSTRAK</b> .....	iv
<b>PENGANTAR</b> .....	v
<b>DAFTAR ISI</b> .....	vi
<b>DAFTAR GAMBAR</b> .....	ix
<b>DAFTAR TABEL</b> .....	x
<b>BAB I PENDAHULUAN</b>	
1.1. Latar Belakang Masalah .....	1
1.2. Rumusan Masalah .....	2
1.3. Batasan Masalah .....	3
1.4. Tujuan .....	3
1.5. Sistematika Penulisan .....	4
<b>BAB II TINJAUAN PUSTAKA</b>	
2.1. Lompat Galah .....	5
2.2. Sensor Optocoupler.....	5
2.3. Display LCD ( <i>Liquid Crystal Display</i> ) .....	6
2.4. <i>Keypad</i> .....	7
2.5. Encoder <i>keypad</i> .....	8
2.6. Register Geser 8 bit.....	9
2.7. Display <i>Seven Segment</i> .....	10
2.8. Motor Dc .....	11
2.9. Mikrokontroler .....	11
2.9.1. Umum .....	11
2.9.2. Konfigurasi Pin-pin AT89C51.....	13
2.9.3. Osilator.. .....	15



2.9.4. Reset.....	15
2.10. Transistor sebagai Saklar .....	17

### **BAB III METODE PENELITIAN**

3.1. Studi Literatur .....	19
3.2. Penentuan Spesifikasi Alat .....	19
3.3. Perencanaan dan Pembuatan Alat .....	19
3.4. Pengujian dan Analisa Rangkaian .....	20

### **BAB IV PERENCANAAN DAN PEMBUATAN ALAT**

4.1. Spesifikasi Alat .....	22
4.2. Blok Diagram Sistem .....	22
4.3. Prinsip Kerja Alat.....	24
4.4. Perencanaan dan Pembuatan Alat .....	25
4.4.1. Perencanaan Bagian Sensor .....	25
4.4.2. Perencanaan Bagian Input ( <i>Keypad</i> ) .....	26
4.4.3. Perencanaan Bagian rangkaian Driver Motor .....	27
4.4.4. Perencanaan Bagian Tampilan .....	28
4.4.5. Perancangan Buzzer.....	32
4.4.5. Perencanaan Bagian Kontrol dan pengolah Data .....	33
4.4.6. Perencanaan Bagian Perangkat Lunak .....	37

### **BAB V PENGUJIAN DAN ANALISIS ALAT**

5.1. Pengujian Sensor .....	39
5.2. Pengujian Rangkaian <i>keypad</i> .....	41
5.3. Pengujian Rangkaian <i>driver</i> Motor.....	42
5.4. Pengujian <i>display</i> LCD .....	44
5.5. Pengujian <i>display 7-segment</i> .....	45
5.6. Pengujian Rangkaian <i>Buzzer</i> .....	48
5.5. Pengujian Keseluruhan .....	49

### **BAB VI PENUTUP**

6.1. Kesimpulan .....	52
-----------------------	----

6.2. Saran ..... 52

**DAFTAR PUSTAKA** ..... 53

**LAMPIRAN**



## DAFTAR GAMBAR

Gambar 2.1.	Rangkaian Tipikal Optocoupler .....	5
Gambar 2.2.	LCD 16x2.....	6
Gambar 2.3.	Blok Diagram Modul LCD .....	6
Gambar 2.4.	Penampang Dasar <i>Keypad</i> 3x4.....	8
Gambar 2.5..	Diagram Koneksi IC 4094 .....	12
Gambar 2.7.	Tampilan <i>7-segment Common Anoda</i> .....	11
Gambar 2.8.	Unit-unit Mikrokontroler.....	12
Gambar 2.9.	Konfigurasi pin AT89C51 .....	13
Gambar 2.10.	Rangkaian Reset.....	16
Gambar 2.11.	Grafik Karakteristik Transistor .....	18
Gambar 4.1.	Blok Diagram Sistem .....	23
Gambar 4.2.	Rangkaian Sensor .....	25
Gambar 4.3.	Rangkaian <i>keypad</i> .....	27
Gambar 4.4.	Rangkaian <i>Driver</i> Motor .....	28
Gambar 4.5.	Rangkaian LCD .....	29
Gambar 4.6.	IC 4094 sebagai <i>Driver 7-segment</i> .....	31
Gambar 4.7.	Rangkaian <i>Display 7-segment</i> .....	31
Gambar 4.8.	Rangkaian <i>Buzzer</i> .....	32
Gambar 4.9.	Rangkaian Osilator.....	34
Gambar 4.10.	Rangkaian Reset.....	35
Gambar 4.11.	Rangkaian Mikrokontroler AT89C51.....	36
Gambar 4.12.	Diagram Alir Program Utama.....	38
Gambar 5.1.	Blok Pengujian Sensor Tinggi.....	40
Gambar 5.2	Diagram Blok Pengujian Rangkaian <i>Keypad</i> .....	41
Gambar 5.3	Blok Diagram Pengujian <i>Driver</i> Motor.....	43
Gambar 5.4	Blok Pengujian <i>Display</i> LCD.....	44
Gambar 5.5	Hasil Pengujian <i>Display</i> LCD.....	45
Gambar 5.6	Blok Pengujian <i>Display 7-segment</i> .....	46
Gambar 5.7	Pengujian <i>Buzzer</i> .....	49
Gambar 5.8	Blok Pengujian keseluruhan.....	50

**DAFTAR TABEL**

Tabel 2.1.	<i>Register Selection</i> Pada LCD .....	7
Tabel 2.2.	Tabel kebenaran encoder <i>Keypad</i> Mm74C922 .....	9
Tabel 2.3.	Tabel kebenaran IC 4094 .....	10
Tabel 2.4.	Fungsi Alternatif Port 3 .....	14
Tabel 4.1.	Tabel Kebenaran keluaran dari IC 74C922.....	27
Tabel 4.2	Fungsi-fungsi terminal pada LCD.....	30
Tabel 5.1.	Hasil Pengujian Rangkaian Sensor.....	40
Tabel 5.2.	Hasil Pengujian Rangkaian <i>Keypad</i> .....	42
Tabel 5.3.	Hasil Pengujian Rangkaian <i>Driver</i> Motor.....	43
Tabel 5.4.	Tampilan <i>7-segment</i> .....	47
Tabel 5.6	Hasil Pengujian Rangkaian <i>Buzzer</i> .....	49
Tabel 5.7	Hasil Pengujian Keseluruhan.....	51



## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang Masalah

Dunia kerja kita beserta lingkungannya kini semakin teknis dan semakin otomatis. Keadaan itu bagi kita manusia pada umumnya sering memerlukan gerak badan yang secukupnya dan komunikasi yang tidak dipaksakan, hubungan yang santai dan menyegarkan. Dengan demikian maka bagi kehidupan kita olahragalah yang makin memegang peranan penting. Olahraga sebagai program kesegaran yang terarah, olahraga sebagai kegiatan mengisi waktu senggang, ataupun sebagai permainan yang memerlukan kerja sama antar pemain.

Banyak argumen tentang olahraga baik yang menentang maupun yang setuju. Sering dikatakan bahwa olahraga itu hanya merupakan kegiatan untuk bersenang-senang saja, bahkan dapat disalahgunakan bagi kepentingan politik, sering dikomersialkan dan digunakan sebagai iklan. Walaupun demikian, argumen yang setuju dengan olahraga jauh lebih banyak. “Makin lemah tubuh, makin banyak ia memerintah; tetapi makin kuat tubuh itu, makin banyak ia mau mendengarkan.” Kata-kata yang berasal dari perancis pada Zaman Rasoinalisme, Jean-Jacques Rousseau, itu hingga kini masih tetap besar artinya. Jadi siapa yang menemukan kesenangan dalam melatih dan mengikutsertakan jasmani, akan juga segera dengan keputusan dan kemauannya sendiri secara bebas, dapat mengakui dan menghargai usaha yang sehat untuk mencapai prestasi sukses. Pencapaian dan pengalaman sukses itu sendiri merupakan soal psikologis yang penting. Dan tempat untuk memperolehnya, dalam hal olahraga, ialah dalam perlombaan.

Atletik telah dikenal sejak lama di berbagai Negara, kemudian berkembang dengan kondisi serta peningkatan dalam bidang sarana, teknis dan gaya. Kita semua mengetahui bahwa atletik merupakan olahraga yang mendasari semua cabang olahraga. Bagi kita yang ingin mengkhususkan diri dalam suatu cabang olahraga, sebaiknya segala sesuatu dalam latihan dimulai dengan latihan atletik yaitu: lari, lompat, dan lempar.

Lompat terdiri dari empat macam, yaitu: lompat jauh, lompat jangkit, lompat tinggi dan lompat tinggi galah. Semua teknik lompat mempunyai tujuan untuk memenangkan ketinggian sebesar mungkin. Dalam setiap permainan ataupun pertandingan khususnya untuk lompat tinggi dan lompat tinggi galah setiap peserta diharuskan mampu melompati ketinggian mistar yang ditentukan. Setelah peserta mampu melewati ketinggiannya maka peserta wajib melewati ketinggian yang berikutnya. Pergantian ketinggian mistar ini dilakukan dengan cara memutar posisi penopang di kanan dan kiri tiang secara bersama-sama dan periode kecepatan yang sama pula hal ini untuk menghindari kenaikan pada sisi kanan dan kiri yang berbeda. Pada sisi tengah diukur lagi dengan mistar untuk memastikan kebenaran ketinggian itu. Kendala yang dirasakan yaitu bila kita memutarnya tidak dengan kecepatan yang sama maka penopang akan tinggi sebelah, dan memakan waktu yang lumayan lama karena perlu seseorang dari kejauhan untuk memastikan mistar lurus kemudian untuk memastikan ketinggian harus diukur kembali. Dalam hal ini dirasakan terlalu lama dan kurang praktis.

Berdasarkan ilustrasi di atas dibuatlah suatu alat pengatur ketinggian mistar untuk olahraga lompat tinggi galah. Alat ini dirancang untuk mengatur ketinggian mistar dari jauh dengan hanya memasukkan angka yang diinginkan dilengkapi dengan papan *display* sehingga penontonpun dapat mengetahui ketinggian yang harus dilewati peserta dan apabila mistar jatuh ada indikator yang berbunyi menandakan peserta tersebut diskualifikasi (gagal).

## 1.2 Pembatasan Masalah

Untuk menjaga agar tidak melebar nya masalah yang dibahas dalam penelitian ini maka akan dibatasi permasalahan yang ada, adapun permasalahan yang dibahas adalah :

- 1) Masalah perangkat keras yang meliputi mikrokontroler sebagai pengolah data, sensor, *keypad* dan tampilan berupa LCD dan seven segmen yang akan memberikan informasi posisi ketinggian diminta.
- 2) Masalah perangkat lunak yang meliputi diagram alur (*flow chart*) dan program.

- 3) Tidak membahas sistem mekanik.
- 4) Tidak membahas catu daya yang dipakai.

### 1.3 Rumusan Masalah

Mengacu pada latar belakang diatas yang berkaitan dengan proses penelitian alat ini, maka dapat disusun rumusan masalah sebagai berikut :

1. Bagaimana membuat alat yang berfungsi sebagai alat ukur yang dapat digunakan sebagai pengatur ketinggian mistar lompat tinggi.
2. Bagaimana merancang dan membuat sensor sebagai pembaca posisi yang diinginkan.
3. Bagaimana proses pengolahan data pada mikrokontroller yang digunakan.
4. Bagaimana membuat perangkat lunak penampil posisi ketinggian mistar yang diinginkan berupa angka-angka decimal pada *display* LCD dan *seven segment*.
5. Bagaimana merancang *keypad* sehingga berfungsi sebagai masukkan data yang diinginkan.
6. Bagaimana merancang driver untuk alat tersebut.
7. Bagaimana merancang motor sehingga dapat bergerak naik turun sesuai posisi yang diinginkan.
8. Bagaimana merancang *buzzer* sehingga dapat berbunyi ketika mistar jatuh.

### 1.4 Tujuan

Dari ilustrasi permasalahan tujuan pembuatan prototype pengatur ketinggian mistar ini adalah:

- 1) Dapat membuat alat yang berfungsi sebagai alat ukur yang dapat digunakan untuk mengatur ketinggian mistar pada olahraga lompat galah.
- 2) Dapat memproses data yang masuk pada mikrokontroller yang digunakan.
- 3) Dapat membuat perangkat-lunak untuk menampilkan ketinggian mistar pada *display* LCD dan *seven segment*.
- 4) Dapat memberikan kemudahan bagi masyarakat dalam berolahraga, pertandingan atau sekedar menikmati kegemaran mereka.

## 1.5 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan laporan tugas akhir ini adalah sebagai berikut:

- Bab I : Menjelaskan tentang Latar Belakang Permasalahan, Rumusan dan Pembatasan Masalah, serta Tujuan dan Kegunaan Kajian.
- Bab II : Menjelaskan tentang teori dasar yang berisi tentang penjelasan prinsip dasar sensor, prinsip dasar motor, mikrokontroler, *keypad* dan *display* LCD dan Seven Segment
- Bab III : Menjelaskan tentang metodologi penelitian, perencanaan dan pembuatan alat dan cara pengujian alat.
- Bab IV : Menjelaskan tentang blok diagram perancangan dan pembuatan alat yang meliputi prinsip kerja, spesifikasi alat, perancangan *hardware*, dan perancangan *software*.
- Bab V : Menjelaskan tentang pengujian dan analisa yang meliputi pengujian tiap-tiap bagian dan pengujian secara keseluruhan
- Bab VI : Memberikan kesimpulan dan saran

## BAB II

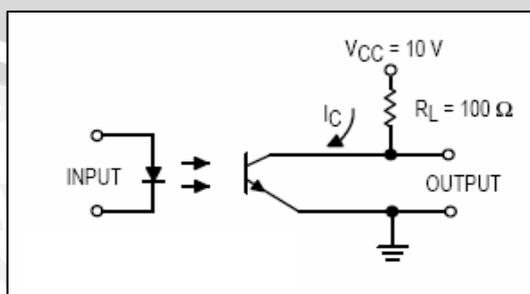
### TINJAUAN PUSTAKA

#### 2.1 Lompat Tinggi Galah

Lompat galah merupakan salah satu jenis keterampilan untuk melewati mistar yang berada diantara kedua tiang dengan menggunakan galah. Seorang peserta dikatakan gagal lompatannya bila setelah melompat, mistar lompat tidak tetap berada diatas penompang disebabkan oleh gerakan si pelompat pada waktu melompat. Mistar lompat ini dibuat dari *fibre-glass*, metal atau materi lain yang cocok, dengan penampang bulat. Tiang lompat yang digunakan terbuat dari plat. Tiang lompat ini haruslah cukup tinggi untuk melebihi tinggi sebenarnya di mana dinaikkan dengan minimum 10 cm. Jarak antara tiang lompat harus tidak kurang dari 4.50 m juga tidak lebih dari 4.54 m. Mistar lompat tinggi galah ini harus diletakkan pada pasak-pasak sehingga bila tersentuh oleh si pelompat atau oleh galah, ini akan mudah jatuh ke tanah ke arah tempat pendaratan.

#### 2.2 Sensor Optocoupler

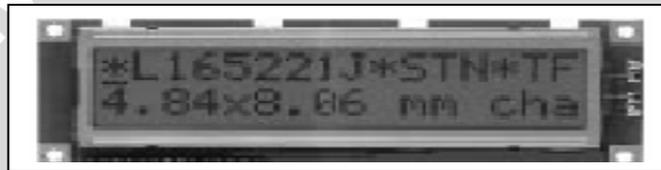
Optocoupler adalah komponen yang tersusun dari sebuah LED (*Light Emitting Diode*) dan sebuah phototransistor yang terintegrasi dalam satu wadah. Prinsip kerja dari optocoupler adalah saat kaki basis terkena cahaya maka akan tercipta pasangan elektron-hole didaerah persambungan. Pasangan elektron-hole ini mengakibatkan terciptanya arus basis. Dengan munculnya arus basis maka transistor akan aktif. Rangkaian tipikal dari optocoupler dan bentuk keluarannya dapat dilihat dalam Gambar 2.1.



Gambar 2.1. Rangkaian tipikal optocoupler  
Sumber : Motorola, 2005

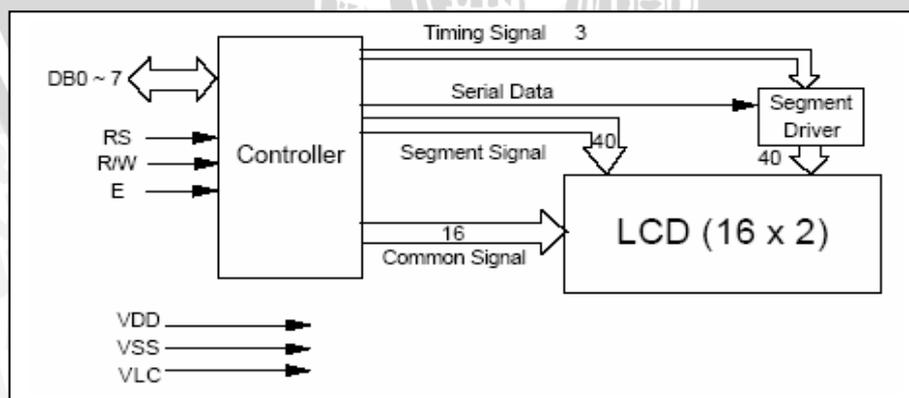
### 2.3 Display LCD (Liquid Crystal Display)

LCD (*Liquid Crystal Display*) adalah salah satu jenis penampil yang digunakan untuk menampilkan angka, karakter atau bahkan angka dan karakter. LCD terdiri atas tumpukan tipis atau sel dari dua lembar kaca dengan pinggiran yang tertutup rapat. Di antara dua lembar kaca tersebut diberi bahan kristal cair (*liquid crystal*) yang tembus cahaya di mana akan beremulsi apabila diberi tegangan. Tampilannya berupa dot matrik 5 x 7 dengan jenis huruf yang ditampilkan akan lebih banyak dan lebih baik resolusinya jika dibandingkan dengan *seven-segment*. Adapun contoh modul LCD 16x2 ditunjukkan dalam Gambar 2.2.



**Gambar 2.2** LCD 16x2  
Sumber: *Anonymous*, 1998:4

Selain sebagai suatu modul tampilan yang mempunyai konsumsi daya yang relatif rendah, LCD juga memiliki keistimewaan dibandingkan tampilan yang lain seperti *sevent-segment* yaitu kemampuan untuk menampilkan karakter dan berbagai macam simbol. Adapun blok diagram LCD ditunjukkan dalam Gambar 2.3 berikut:



**Gambar 2.3** Blok Diagram Modul LCD  
Sumber: *Anonymous*, 1998:13

Untuk dapat menggunakan LCD maka hal-hal yang perlu diperhatikan adalah sinyal kontrol dari LCD yaitu : RS, R/W dan EN. Instruksi operasi meliputi operasi dasar *register*, *busy flag*, *address counter*, *display data RAM*, *character generator ROM*.

Fungsi yang sebagaimana ditunjukkan dalam Tabel 2.1

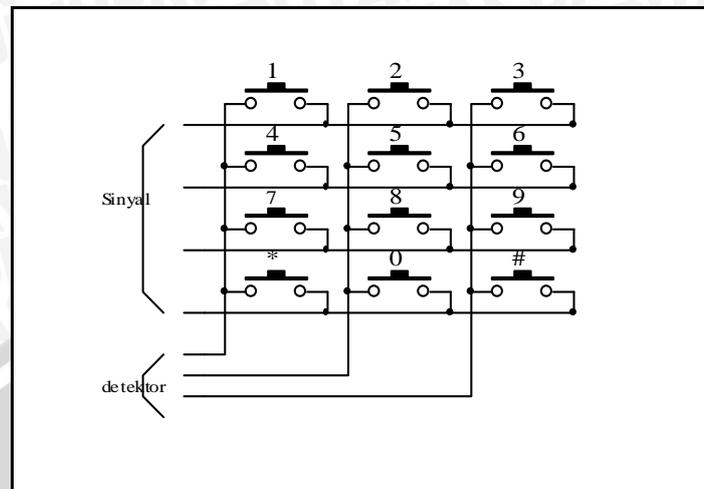
**Tabel 2.1** Fungsi pin-pin pada LCD M1632

No Pin	Nama Pin	Fungsi
16	V – BL	Sebagai <i>ground</i> dari <i>backlight</i>
15	V + BL	Sebagai kutub positif dari <i>backlight</i>
7 - 14	DB0– DB7	Merupakan saluran data, berisi perintah dan data yang akan ditampilkan.
6	E	Sinyal operasi awal, sinyal ini mengaktifkan data tulis atau baca
5	R/W	Sinyal seleksi tulis atau baca 0 = tulis 1 = baca
4	RS	Sinyal pemilih register 0 = register instruksi (tulis) 1 = register data (tulis dan baca)
3	Vlc	Untuk mengendalikan kecerahan LCD dengan mengubah Vlc
2	Vcc	Tegangan catu + 5 volt
1	Vss	Terminal <i>Ground</i>

Sumber: Anonymous, 1998:12

## 2.4 Keypad

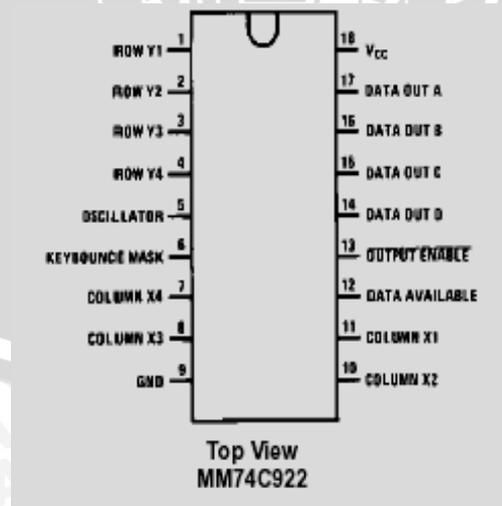
*Keypad* digunakan untuk memasukkan nilai-nilai ketinggian yang diinginkan. Proses kerja *keypad* 3x4 yang digunakan pada system ini dengan cara metode matrik di mana keluaran yang dihasilkan merupakan kombinasi dari baris dan kolom yang ditekan. Terdapat tujuh buah pin yang berfungsi sebagai keluaran dari *keypad*. Tiga buah pin yang menunjukkan kolom dari tombol yang ditekan dan empat buah pin yang menunjukkan baris. Penampang dasar dari *keypad* 3x4 dapat dilihat dalam Gambar 2.4.



Gambar 2.4 Penampang Dasar Keypad 3x4

### 2.5 Encoder keypad

IC decoder 74C922 merupakan perangkat yang berfungsi sebagai pengkode data dari papan *keypad* ke BCD untuk di olah menjadi data digital. Banyaknya kode digital yang dapat dikonversikan tergantung dari jumlah baris dan kolom masukan dari papan *keypad* atau biasa dikatakan jumlah data yang dikonversikan dalam perkalian baris dan kolom masukan. IC 74C922 akan mendekodekan baris dan kolom mana yang menyambung menjadi data biner 4 bit pada keluaran bus data.



Gambar 2.5 Konfigurasi MM74C922  
 Sumber: Fairchild Semiconductor, October 1987:1

Dalam Gambar 2.5 pin 1 – 4 untuk baris keypad 1 – 4, pin 11,10,8,7 untuk kolom keypad 1 – 4, pin 14 – 17 untuk keluaran data. Pin osilator dan *keybounce mask* dihubungkan dengan kapasitor. Nilai dari kapasitor tersebut ditentukan sebagai berikut:

elektrik seri, nilai resistansi R dapat dicari dengan menggunakan persamaan :

$$C_{keybounce\ mask} = 10 \times C_{osilator}$$

Persamaan tersebut diambil dari datasheet *encoder keypad MM74C922*.

Untuk mengetahui fungsi logika dari data keluaran keypad. Tabel 2.2 merupakan tabel kebenaran dari MM74C922.

**Tabel 2. 2** Tabel kebenaran *encoder keypad MM74C922*

(Pins 0 through 11)												
Switch Position	0	1	2	3	4	5	6	7	8	9	10	11
	Y1, X1	Y1, X2	Y1, X3	Y1, X4	Y2, X1	Y2, X2	Y2, X3	Y2, X4	Y3, X1	Y3, X2	Y3, X3	Y3, X4
D												
A A	0	1	0	1	0	1	0	1	0	1	0	1
T B	0	0	1	1	0	0	1	1	0	0	1	1
A C	0	0	0	0	1	1	1	1	0	0	0	0
O D	0	0	0	0	0	0	0	0	1	1	1	1
U E (Note 1)	0	0	0	0	0	0	0	0	0	0	0	0
T												

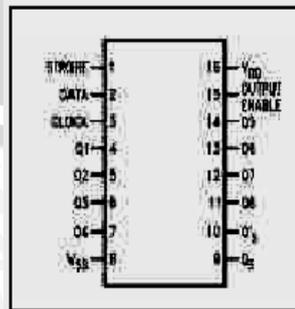
(Pins 12 through 19)									
Switch Position	12	13	14	15	16	17	18	19	
	Y4, X1	Y4, X2	Y4, X3	Y4, X4	Y5 (Note 1), X1	Y5 (Note 1), X2	Y5 (Note 1), X3	Y5 (Note 1), X4	
D									
A A	0	1	0	1	0	1	0	1	
T B	0	0	1	1	0	0	1	1	
A C	1	1	1	1	0	0	0	0	
O D	1	1	1	1	0	0	0	0	
U E (Note 1)	0	0	0	0	1	1	1	1	
T									

Note 1: Omit for MM74C922

Sumber: Fairchild Semiconductor, October 1987:2

## 2.6 Register Geser 8 bit

IC 4094 merupakan register geser 8-bit. Diagram koneksi dari IC 4094 ditunjukkan dalam Gambar 2.6.



**Gambar 2.6.** Diagram Koneksi IC 4094

Sumber: Anonim, 1990:1

Pada IC 4094 terdapat input Strobe dan OE (output enable), sedangkan pada outputnya selain mempunyai output QS dan Q'S. Output QS merupakan output data terakhir (*MSB*) yang dapat digunakan untuk keperluan kaskade dengan shift register berikutnya. Prinsip kerja dari IC dapat dilihat dalam Tabel 2.3

**Tabel 2.3.** Tabel Kebenaran IC 4094

CL <sup>A</sup>	Outputs Enable	Strobe	Data	Parallel Outputs		Serial Outputs	
				Q1	QN	QS*	Q'S
$\overline{1}$	0	X	X	OC	OC	Q7	NC
$\overline{1}$	0	X	X	OC	OC	NC	Q7
$\overline{1}$	1	0	X	NC	NC	Q7	NC
$\overline{1}$	1	1	0	0	Q <sub>N-1</sub>	Q7	NC
$\overline{1}$	1	1	1	1	Q <sub>N-1</sub>	Q7	NC
$\overline{1}$	1	1	1	NC	NC	NC	Q7

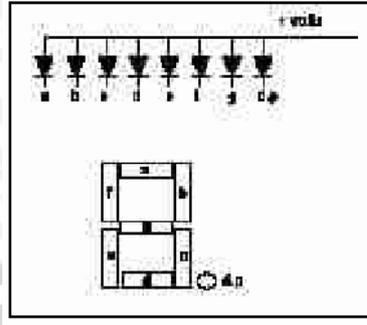
Sumber: 1995,4

Ada empat cara di mana register geser dapat digunakan untuk menyimpan dan memindah data dari suatu bagian ke bagian yang lain, yaitu:

1. Masukkan seri keluaran paralel (SIPO)
2. Masukkan seri keluaran seri (SISO)
3. Masukkan paralel keluaran seri (PISO)
4. Masukkan paralel keluaran paralel (PIPO)

## 2.7. Display 7-segment

*Display 7-segment* adalah gabungan dari beberapa LED (*Light Emitting Diode*) yang dapat dicatu secara individu atau masing-masing. Warna yang ditampilkan pada *display* ini umumnya warna merah. LED-LED tersebut tersusun menyerupai angka delapan dan jika dicatu semuanya maka akan menampilkan angka delapan. Seperti namanya, ada tujuh *segment* LED yang dapat diatur kombinasinya untuk menampilkan angka dan huruf. Untuk lebih lengkapnya tampilan *7-segment* terdapat dalam Gambar 2.7.



**Gambar 2.7.** Tampilan 7-segment Common Anoda

Sumber : Anonim, 2004:1

Ada dua macam hubungan pada *display 7-segment*, yaitu:

1. *Common Anode*

Pada hubungan ini semua *anoda* dihubungkan menjadi satu ke positif catu daya dan katodanya difungsikan sebagai masukan.

2. *Common Katode*

Pada hubungan ini semua *katoda* dihubungkan menjadi satu ke *ground* dan anodanya sebagai masukan.

## 2.8. Motor Dc

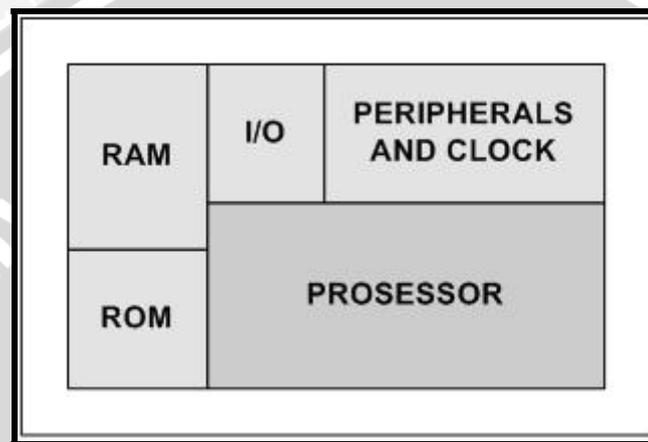
Motor Dc dalam system ini digunakan untuk menggerakkan penompang pada sisi kanan dan kiri tiang. Motor arus searah mengubah tegangan listrik arus searah menjadi tenaga gerak mekanik yang berupa putaran dari rotor. Pada mesin arus searah, kumparan medan merupakan stator (bagian yang tidak berputar) dan kumparan jangkar merupakan rotor (bagian yang berputar). Untuk dapat menghentikan motor dalam relative singkat dilakukan pengereman. Pengereman mendadak yaitu dengan cara membalik polaritas motor. (Zuhul, 1990 : 155).

## 2.9. Mikrokontroler

### 2.9.1. Secara Umum

Secara umum, mikrokontroler berfungsi sama dengan *personal computer* (PC). Bedanya adalah mikrokontroler memiliki desain dalam sebuah *single chip* (*Integrated Circuit*). Mikrokontroler terdapat di hampir semua peralatan

elektronik, di dalam tape, TV, radio, telepon genggam (*handphone*) dan lain-lain. Mikrokontroler memiliki kemampuan yang diperlukan untuk membuat keputusan berdasarkan masukan yang diberikan dari luar dengan kata lain mikrokontroler merupakan otak dari sebuah perangkat elektronik. Mikrokontroler terdiri atas empat unsur yaitu prosesor (*Central Processing Unit*), memori, perangkat I/O dan perangkat lain. Seperti ditunjukkan dalam Gambar 2.8.



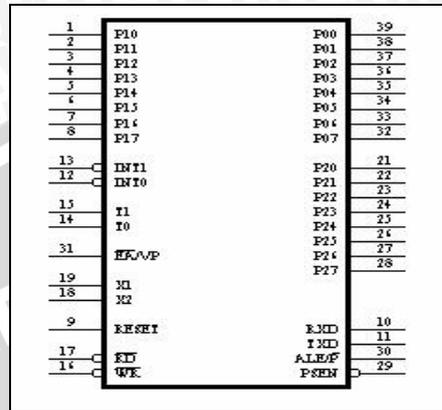
**Gambar 2.8** Unit-unit Mikrokontroler  
Sumber: Budiman, 2005:2

Blok CPU, memori dan I/O merupakan blok utama sebuah mikrokontroler. Setiap mikrokontroler pasti memiliki blok tersebut. Selain tiga blok utama tersebut terdapat perangkat (*peripheral*) lain. Ketersediaan peripheral-peripheral dalam mikrokontroler tersebut dapat mengurangi adanya perangkat eksternal sehingga memperkecil ukuran alat elektronik secara keseluruhan.

Mikrokontroler didesain dengan instruksi-instruksi lebih luas dan 8 bit instruksi yang digunakan membaca data instruksi dari internal memori ke *Arithmetic Logic Unit* (ALU). Banyak instruksi yang digabung dengan pin-pin *chip*nya. Pin tersebut yaitu pin yang dapat diprogram (*programmable*) yang mempunyai beberapa fungsi yang berbeda tergantung pada kehendak *programmer*. Untuk memprogram sebuah *chip* mikrokontroler dibutuhkan sebuah alat pengisi kode mesin (*emulator*). *Mikroprosesor* didesain sangat fleksibel dan mempunyai banyak *byte* instruksi. Semua instruksi bekerja dalam sebuah konfigurasi perangkat keras yang membutuhkan banyak ruang memori dan perangkat I/O dihubungkan ke alamat dan pin-pin bus data pada *chip*.

### 2.9.2. Konfigurasi Pin-pin AT89C51

Mikrokontroler AT89C51 memiliki pin sebanyak 40 seperti ditunjukkan dalam Gambar 2.9.



Gambar 2.9. Konfigurasi Pin AT89C51

Fungsi dari pin-pin MCU AT89C51

1. VCC  
Pin positif sumber tegangan 5 Volt DC
2. Vss  
Pin *grounding* sumber tegangan
3. Port 0  
Port 0 merupakan port 8 bit open drain dua arah. Jika port 0 sebagai *output* maka kondisi rendah mampu menyerap 8 unit *load input* TTL dengan label LS. Jika diinginkan port 0 sebagai input maka logika 1 harus dituliskan terlebih dahulu untuk membuat *low addres* dan data bus selama mengakses program memori (ROM) dan data memori (RAM). Dalam pemakaiannya pada saat port 0 mengeluarkan logika 1 dibutuhkan eksternal *pull-up* yang besar, dan pada logika 0 dapat menyerap arus sebanyak 8 unit *load input* TTL dengan label LS.
4. Port 1  
Port 1 adalah terminal *input/output* dua arah dengan *pull up* internal. *Buffer output* port 1 dapat menyerap arus sebanyak 4 unit *load input* TTL dengan label LS. Jika port 1 dipergunakan sebagai *input* logika 1 harus dituliskan

terlebih dahulu pada *latch* untuk membuat FET dalam kondisi OFF. Sehingga rangkaian eksternal dapat mensupply logika 0 atau logika 1.

5. Port 2

Port 2 dapat dipergunakan sebagai *input* atau *output* seperti port 1. Alternatif lain dari port 2 dapat dipergunakan sebagai *address bus high byte* pada saat mengakses memori eksternal

6. Port 3

Port 3 adalah terminal *input* atau *output* dengan internal *pull-up*. *Buffer output* port 3 dapat menyerap arus sebanyak 4 bit unit *load input* TTL dengan label LS. Jika port 3 dipergunakan sebagai *input*, logika 1 harus dituliskan pada *latch* untuk membuat FET dalam kondisi OFF, sehingga rangkaian eksternal dapat mensupply logika 0 atau logika 1. Port 3 juga mempunyai fungsi lain yang ditunjukkan dalam Tabel 2.4.

Tabel 2.4. Fungsi Alternatif Port 3

Port Pin	Fungsi Alternatif
P3.0	RXD ( <i>serial input port</i> )
P3.1	TXD ( <i>serial output port</i> )
P3.2	$\overline{INT0}$ (eksternal interrupt 0)
P3.3	$\overline{INT1}$ (eksternal interrupt 1)
P3.4	T0 (timer 0 eksternal input)
P3.5	T1 (timer 1 eksternal input)
P3.6	WR (eksternal <i>data memory write strobe</i> )
P3.7	RD (eksternal <i>data memory data read strobe</i> )

Sumber: Atmel, 1997:4

7. RST

Pin ini merupakan input *reset*. Kondisi tinggi dari pin ini selama dua siklus clock akan menyebabkan *reset device* ini.

8. ALE/PROG

*Output Address Latch Enable* berfungsi untuk me-*latch* byte rendah dari *address* semala melakukan akses memori eksternal. ALE dapat mengemudikan 8 unit *load input* TTL dengan label LS. Pin ini juga merupakan pulsa input program selama terjadi pemrograman EPROM internal. Di dalam aplikasi normal ALE mengeluarkan deretan pulsa yang

konstan dengan kecepatan  $1/6$  frekuensi osilator, dan mungkin pulsa ini dapat dipergunakan untuk keperluan clock eksternal.

9. PSEN

*Program Store Enable* merupakan *strobe* output yang dipergunakan untuk membaca eksternal program memori. PSEN aktif setiap dua siklus mesin.

10. EA/Vpp

*Ekternal Acces Enable* secara eksternal harus disambungkan ke logika 0 jika diinginkan MCS-51 menjadi *enable* untuk mengakses kode mesin dari program memori eksternal. Jika EA dihubungkan ke logika 1 maka *device* akan mengambil kode mesin dari internal program memori kecuali kalau *program counter* berisi lebih besar dari 0FFFh.

11. XTAL1

Pin ini merupakan *input inverting* amplifier osilator.

12. XTAL2

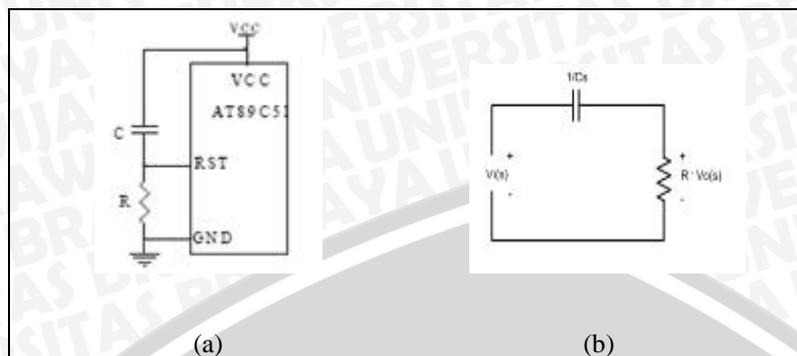
Pin ini merupakan *output inverting* amplifier osilator

### 2.9.3. Osilator

Mikrokontroler AT89C51 memiliki osilator internal (*on chip oscillator*) yang dapat digunakan sebagai sumber pewaktuan (*clock*) bagi CPU. Untuk menggunakan internal diperlukan sebuah kristal atau resonator keramik antara pin XTAL1 dan pin XTAL2 dan sebuah kapasitor ke *ground*. Untuk kristalnya dapat digunakan dengan frekuensi dari 6 sampai 12 MHz, sedangkan kapasitor dapat bernilai 27 pF sampai 33 pF.

### 2.9.4. Reset

Rangkaian *power on reset* diperlukan untuk *mereset* mikrokontroler secara otomatis setiap catu daya *on*. Gambar 2.10 menunjukkan rangkaian *power on reset*. Ketika catu daya diaktifkan, rangkaian *reset* menahan logika tinggi pin RST dengan jangka waktu yang ditentukan oleh besarnya pengisian muatan C, yang mana nilai jangka waktunya dihitung dengan Persamaan 2.1.



**Gambar 2.10.** (a) Rangkaian *power on reset* (b) Rangkaian ekivalen *power on reset*  
 Sumber: Atmel, 1997: 2-63

Dalam rangkaian ekivalen seperti ditunjukkan dalam Gambar 2.10 (b) maka didapat

$$V_o(s) = \frac{R}{R + \frac{1}{Cs}} \cdot V_i(s) = \frac{RCs}{RCs + 1} \cdot V_i(s) \dots\dots\dots (2.1)$$

dengan tegangan \$V\_i\$ adalah tegangan \$V\_{cc}\$ yaitu 5V, dalam fungsi Laplace adalah \$5/s\$, sehingga;

$$V_o = \frac{RCs}{RCs + 1} \cdot \frac{5}{s} = \frac{RC}{RCs + 1} \cdot 5 = 5 \cdot \left( \frac{1}{s + \frac{1}{RC}} \right) \dots\dots\dots (2.2)$$

$$V_o = 5 \cdot e^{-\left(\frac{t}{RC}\right)} \dots\dots\dots (2.2)$$

$$\frac{5}{V_o(t)} = e^{\frac{t}{RC}}$$

$$\ln\left(\frac{5}{V_o}\right) = \frac{t}{RC}$$

Maka

$$t = R \cdot C \cdot \left( \ln \frac{5}{V_o} \right), \quad \text{atau;} \quad t = R \cdot C \cdot \frac{\left( \log \frac{5}{V_o} \right)}{\log e} \dots\dots\dots (2.3)$$

dengan nilai \$V\_o\$ adalah tegangan logika nominal yang diijinkan oleh pin RST (Atmel, 1997: 4-37), di mana

$$V_o = 0,7 \times V_{cc} = 0,7 \times 5 \text{ volt} = 3,5 \text{ volt.}$$



maka

$$t = R \cdot C \cdot \frac{\left( \log \frac{5}{3,5} \right)}{\log e}, \text{ sehingga}$$

$$t = 0,357 \cdot R \cdot C \quad (2.4)$$

Untuk memastikan keabsahan *reset*, logika tinggi harus ditahan lebih dari 2 siklus mesin (24 periode osilator).

### 2.10 Transistor

Menggunakan transistor sebagai sebuah switch berarti meletakkan titik kerja pada titik saturasi atau titik *cut off*. Titik sumbat (*cut off*) adalah titik dimana garis beban memotong kurva  $I_B=0$ . Pada titik ini arus basis adalah nol dan arus kolektor kecil mendekati nol atau dapat dikatakan tidak ada arus melewati transistor. Sehingga bekerja seperti saklar terbuka. Gambar 2.11 menunjukkan perpotongan dari garis beban dan kurva basis yang lebih tinggi masih menghasilkan titik kejenuhan yang sama. Untuk perkiraan aproksimasi tegangan kolektor-emiter adalah:

$$V_{CE}(\text{cutoff}) = V_{CC} \quad (2.5)$$

Perpotongan dari garis beban dan kurva  $I_B = I_B(\text{sat})$  disebut penjenuhan (saturasi). Pada titik ini arus basis sama dengan  $I_B(\text{sat})$  dan arus kolektor adalah maksimum. Sehingga transistor bekerja seperti saklar tertutup. Untuk perkiraan aproksimasi, arus kolektor pada penjenuhan adalah:

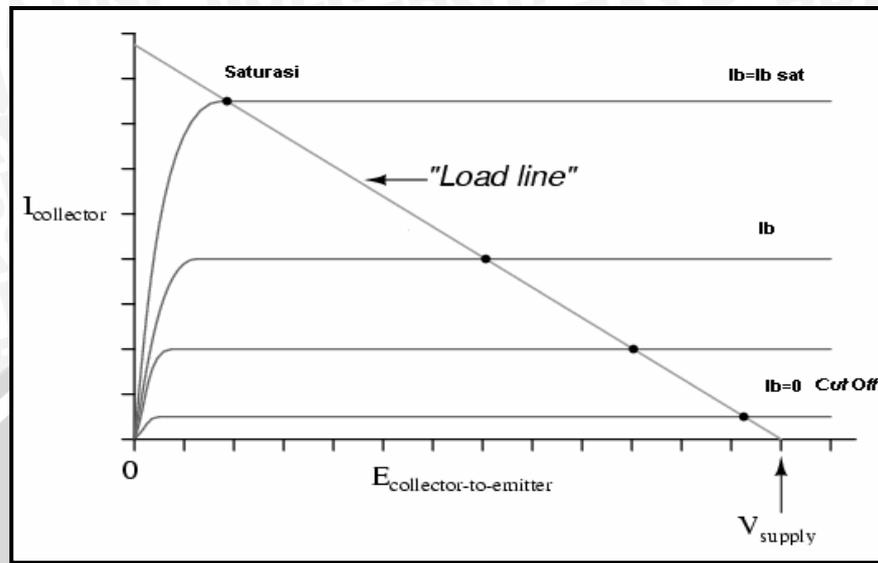
$$I_{C(\text{sat})} \cong \frac{V_{CC}}{R_C} \quad (2.6)$$

Dan arus basis yang tepat menimbulkan penjenuhan adalah sebagai berikut:

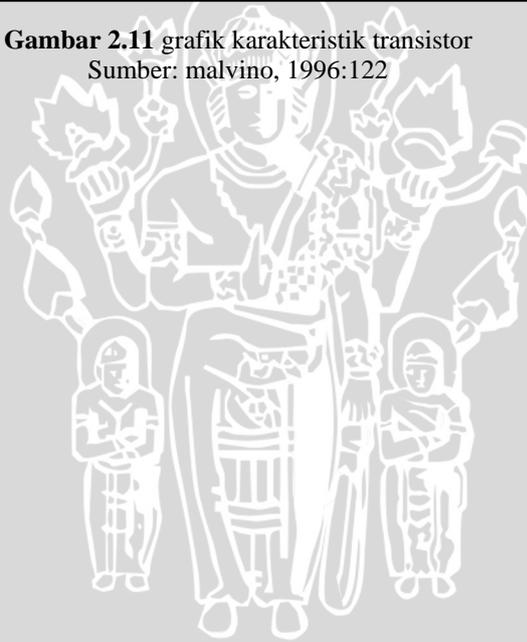
$$I_{B(\text{Sat})} = \frac{I_{C(\text{sat})}}{\beta_{dc}} \quad (2.7)$$

Tegangan kolektor –emiter pada penjenuhan adalah

$$V_{CE} = V_{CE(\text{sat})} \quad (2.8)$$



Gambar 2.11 grafik karakteristik transistor  
 Sumber: malvino, 1996:122



## BAB III

### METODOLOGI

Pada bab ini diuraikan metode penelitian yang dilakukan dalam perancangan *Prototype* Pengatur Ketinggian Mistar Pada Olahraga lompat Galah dilengkapi Dengan Papan *Display*. Perancangan ini mengacu pada rumusan masalah yang telah dibuat sebelumnya. Metodologi yang digunakan secara umum meliputi pengumpulan data berupa studi literature, perancangan sistem, pembuatan perangkat keras dan perangkat lunak, pengujian alat dan analisis hasil pengujian, serta pengambilan kesimpulan dan saran

#### 3.1 Studi Literatur

Bentuk kajian awal metode penelitian dilakukan dengan studi literatur agar dapat memperoleh hasil yang optimal. Studi literatur dilakukan untuk mendapatkan pengetahuan dasar tentang segala sesuatu yang mendukung perancangan serta pembuatan *prototype* pengatur ketinggian mistar pada olahraga lompat galah dilengkapi dengan papan *display* ini. Dalam pembuatan alat ini menggunakan referensi dari buku-buku maupun artikel dari internet untuk mengetahui karakteristik komponen, prinsip kerja serta teori yang menunjang.

#### 3.2 Penentuan Spesifikasi Alat

Studi literature dipelajari kemudian dari permasalahan yang ada dibuat spesifikasi alat yang akan dibuat, penentuan spesifikasi alat dilakukan selain untuk memudahkan melakukan perancangan dan pembuatan alat juga dapat memberikan informasi tentang kemampuan alat.

#### 3.3 Perencanaan dan Pembuatan alat

Dari penentuan spesifikasi alat, tahap selanjutnya adalah perencanaan alat. Dalam perencanaan ini langkah-langkah yang dilakukan meliputi penentuan spesifikasi sistem yang akan direncanakan, penyusunan blok diagram sistem untuk mempermudah pemahaman tentang alur kerja dari alat yang akan dibuat, pembuatan skema rangkaian

dan perhitungan besaran-besaran listriknya. Selanjutnya dilakukan perhitungan dan perencanaan tiap-tiap blok rangkaian dilakukan analisis dan perhitungan untuk mencapai hasil yang optimal dari komponen yang digunakan, yang akan disesuaikan dengan komponen yang ada di pasaran.

Dalam perencanaan dan pembuatan alat ini dibagi menjadi dua bagian utama sebagai berikut:

a) Perangkat Keras

- Pembuatan blok diagram lengkap dari alat yang dirancang.

Blok diagram atas sensor beserta rangkaiannya, mikrokontroler, rangkaian keypad, rangkaian *driver* motor, rangkaian LCD dan rangkaian *display 7-segment*, serta rangkaian *buzzer*..

- Perencanaan rangkaian tiap-tiap blok rangkaian.

Melakukan perhitungan dalam penyusunan rangkaian dalam masing-masing blok diagram sehingga terbentuk skema rangkaian yang saling terintegrasi antara input, unit mikrokontroler dan unit output.

b) Perangkat Lunak

- Pembuatan *flowchart* dan perangkat lunak

Membuat *flowchart* dan perangkat lunak agar mikrokontroler dapat berfungsi sesuai dengan yang direncanakan sebelumnya. Penulisan perangkat lunak menggunakan program assembler.

### 3.4 Pengujian dan Analisis Rangkaian

Untuk mengetahui tingkat keberhasilan sistem dan kesesuaian dengan perencanaan maka dilakukan pengujian rangkaian. Pengujian dilakukan pada masing-masing blok secara keseluruhan. Adapun macam pengujian yang dilakukan adalah:

- Pengujian Rangkaian sensor
- Pengujian Rangkaian *keypad*.
- Pengujian Rangkaian *driver* motor.
- Pengujian Rangkaian LDC.
- Pengujian Rangkaian *display 7-segment*.
- Pengujian Rangkaian *buzzer*.

Pengujian peralatan secara keseluruhan adalah dengan memcobakan seluruh fasilitas yang ada antara lain ;

- Kemampuan sensor dalam membaca ketinggian
- Kemampuan motor bergerak naik dan turun.
- Kemampuan *keypad* sebagai masukkan data
- Kemampuan LCD dan *7-segment* sebagai tampilan.
- Kemampuan buzzer sebagai indikator mistar jatuh.

Pengujian yang dilakukan menghasilkan data-data yang nantinya di analisis untuk mengetahui tingkat keberhasilan perencanaan. Analisis dilakukan terhadap masing-masing pengujian blok, perangkat lunak serta keseluruhan sistem. Dari hasil analisis tersebut selanjutnya disusun suatu kesimpulan.



## BAB IV

### PERANCANGAN DAN PEMBUATAN ALAT

Pada dasarnya perencanaan dan pembuatan *prototype* pengatur posisi ketinggian pada lompat galah ini ada dua bagian utama yaitu bagian perencanaan perangkat keras dan bagian perencanaan perangkat lunak. Bagian perencanaan perangkat keras terbagi lagi menjadi lima bagian (blok) yang terpisah yaitu bagian sensor, bagian input, bagian keluaran, bagian tampilan, bagian kontrol dan pengolah data. Pada masing-masing bagian tersebut disusun dengan pemilihan beberapa jenis komponen dengan fungsi sesuai dengan perencanaan, sehingga nantinya akan dihasilkan satu bentuk bagian (blok) dengan fungsi sesuai dengan perencanaan yang dilakukan di awal. Pada bagian perencanaan perangkat lunak dilakukan perancangan pembuatan diagram alir program. Dengan terbentuknya bagian-bagian tersebut akan dihasilkan satu sistem pengatur ketinggian lompat tinggi galah.

#### 4.1 Spesifikasi Alat

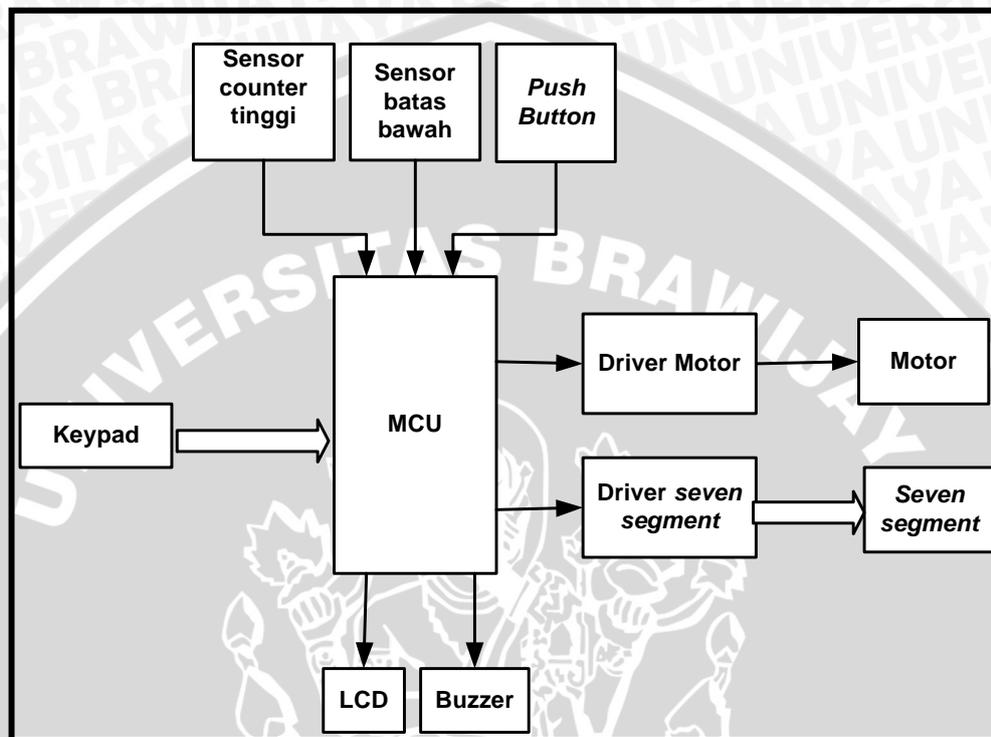
Dalam perencanaan dan pembuatan *prototype* pengatur posisi ketinggian pada lompat tinggi galah ini memiliki spesifikasi berdasarkan studi literature dari rumusan masalah yang ada sebagai berikut :

1. Prototype ini dirancang dengan range maksimum 50 cm.
2. Resolusi pengukuran dari alat ini adalah ketinggian dalam satuan centi meter.
3. Kenaikan dan penurunan minimal 1 cm.
4. Tampilan dalam LCD dan *Seven segment*

#### 4.2 Blok Diagram Sistem

Diagram blok sistem merupakan salah satu bagian terpenting dalam perancangan *Prototype* pengatur ketinggian Mistar Lompat Galah ini, karena dari diagram blok dapat diketahui cara kerja (prinsip kerja) keseluruhan rangkaian. Selain itu juga memudahkan proses perancangan dan pembuatan pada masing-

masing bagian, sehingga akan terbentuk suatu sistem yang sesuai dengan perancangan sebelumnya. Adapun blok diagram sistem ditunjukkan dalam Gambar 4.1.



Gambar 4.1 Blok Diagram Sistem

Dari blok diagram di atas dapat dijelaskan secara universal mengenai bagian-bagian yang menyusun keseluruhan sistem dari alat ini di antaranya:

### 1. Bagian Sensor

Sensor yang digunakan yaitu *optocoupler* dan *push button*, dimana digunakan dua buah *optocoupler* yang pertama sebagai sensor counter tinggi dengan membaca titik yang ada pada samping tiang untuk memperoleh ketinggian yang diinginkan. *Optocoupler* yang kedua sebagai sensor batas bawah untuk kembali pada posisi awal apabila posisi mistar berada ditengah. *Push Button* digunakan sebagai indikasi apabila mistar jatuh.

### 2. Keypad

*Keypad* digunakan sebagai masukkan data, yang kemudian mengirim sinyal pada MCU untuk diolah.

### 3. MCU

MCU yang digunakan disini memiliki beberapa fungsi diantaranya : memberikan instruksi pada motor, memberikan instruksi pada LCD dan 7-segment sebagai *display*..

### 4. LCD (*Liquid Crystal Display*)

LCD merupakan *display* yang digunakan untuk menampilkan perintah-perintah untuk menaikkan dan menurunkan posisi ketinggian yang diinginkan.

### 5. 7 - Segment

Merupakan *display* yang menampilkan posisi ketinggian yang baru bagi penonton.

### 6. Motor

Sebagai penggerak mekanik yang berfungsi untuk menggerakkan posisi penopang mistar naik dan turun.

### 7. Buzzer

Berfungsi sebagai indikator bila mistar jatuh.

## 4.3 Prinsip Kerja Alat

Metode yang digunakan dalam sistem ini adalah mengatur ketinggian mistar dengan cara memasukkan data ketinggian yang diinginkan melalui *keypad* dan dengan bantuan perintah perintah instruksi yang tampil dalam LCD. *Keypad* mengirim sinyal pada MCU. Mikrokontroler menerjemahkan masukkan sesuai dengan program yang disimpan dalam memori pada mikrokontroler. Setelah proses inisialisasi mikrokontroler memerintahkan motor bekerja sesuai dengan sinyal masukkan tadi yaitu menaikkan atau menurunkan penopang mistar.

Pada sisi kiri dan kanan tiang penopang terdapat sensor optocoupler, sensor ini yang digunakan untuk membaca ketinggian yang diinginkan dan sebagai counter batas bawah. Apabila pada tengah –tengah mistar jatuh maka push button ditekan dan *buzzer* akan berbunyi sebagai indikator dan penopang akan bergerak kembali kebawah. Dan ketinggian yang diinginkan akan ditampilkan pada *display* 7 – Segment agar penonton mengetahui ketinggian yang sekarang.

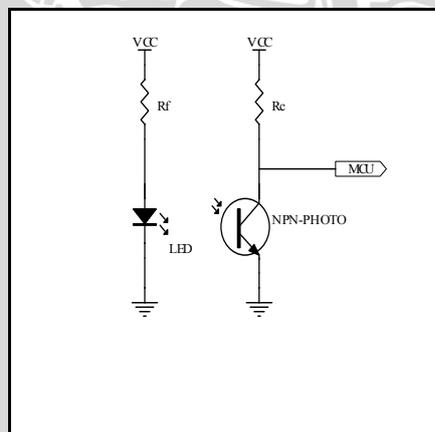
#### 4.4 Perencanaan dan Pembuatan Alat

Dari Gambar 4.1 blok diagram sistem dan uraian di atas maka perencanaan dari masing-masing bagian (blok) dapat diuraikan sebagai berikut:

- 1) Perencanaan bagian perangkat keras (*hardware*), yaitu perencanaan;
  - a) Bagian sensor
  - b) Bagian Input (*Keypad*)
  - c) Bagian keluaran (Motor)
  - d) Bagian tampilan (LCD 16x2) dan *7-segment*.
  - e) Bagian kontrol dan pengolah data (mikrokontroler AT89C51)
- 2) Perencanaan bagian perangkat lunak.

##### 4.4.1 Perencanaan bagian sensor

Pada bagian ini digunakan optocoupler internal berbentuk U sebanyak 2 buah. Rangkaian ini bertujuan untuk mendeteksi tiap centi meter kenaikan mistar dan sebagai counter batas bawah. Sensor ini dirangkai dengan 2 resistor 220  $\Omega$  dan 10 k $\Omega$ . rangkaian sensor ditunjukkan dalam Gambar 4.2



Gambar 4.2 Rangkaian Sensor Optocoupler

Perhitungan nilai yang digunakan adalah sebagai berikut:

$$R_F = \frac{V_{cc} - V_F}{I_F}$$

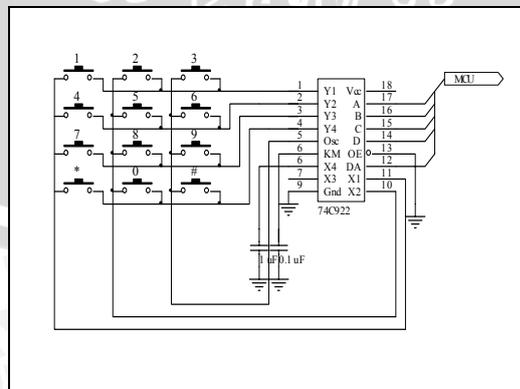
$$\begin{aligned}
 &= \frac{5V - 1V}{20mA} \\
 &= \frac{4V}{20mA} \\
 &= 200 \Omega \approx 220 \Omega
 \end{aligned}$$

Vcc ditentukan sebesar 5 volt dan besar Vce didapat dari *datasheet* sebesar 0,3 volt. Dan  $I_{c(sat)}$  pada perancangan sebesar 0,15 mA. Maka nilai Rc didapat :

$$\begin{aligned}
 R_c &= \frac{V_{cc} - V_{ce(sat)}}{I_{c(sat)}} \\
 &= \frac{5v - 0,3v}{0,15mA} \\
 &= \frac{4,7Volt}{0,15mA} \\
 &= 9200 \Omega \approx 10 \text{ k}\Omega
 \end{aligned}$$

#### 4.4.2 Perencanaan Input ( Keypad )

Rangkaian *keypad* berfungsi untuk memasukkan data atau nilai-nilai ketinggian yang diinginkan. Digunakan *keypad* matrik 3x4 akan dikodekan oleh IC 74C922 sehingga dihasilkan kode-kode yang akan diterjemahkan oleh mikrokontroller. Konfigurasi *keypad* dengan IC 74C922 ditunjukkan dalam Gambar 4.3.



Gambar 4.3. Rangkaian Keypad

Tabel kebenaran data keluaran dari IC 74C922 ditunjukkan dalam Tabel 4.1.

**Tabel 4.1** kebenaran data keluaran dari IC 74C922

Switch Position	1 Y1.X1	2 Y1.X2	3 Y1.X3	4 Y2.X1	5 Y2.X2	6 Y2.X3	7 Y3.X1	8 Y3.X2	9 Y3.X3	* Y4.X1	0 Y4.X2	# Y4.X3
A	0	1	0	0	1	0	0	1	0	1	1	1
B	0	0	1	0	0	1	0	0	1	1	1	1
C	0	0	0	1	1	1	0	0	0	0	1	0
D	0	0	0	0	0	0	1	1	0	0	0	1

Nilai  $C_1$  dan  $C_2$  dapat ditentukan sesuai dengan persamaan dari *datasheet*.

Dimana  $C_1$  sebagai kapasitor untuk *keybouncemask*, sedangkan  $C_2$  sebagai kapasitor osilator untuk proses *scanning keypad*.

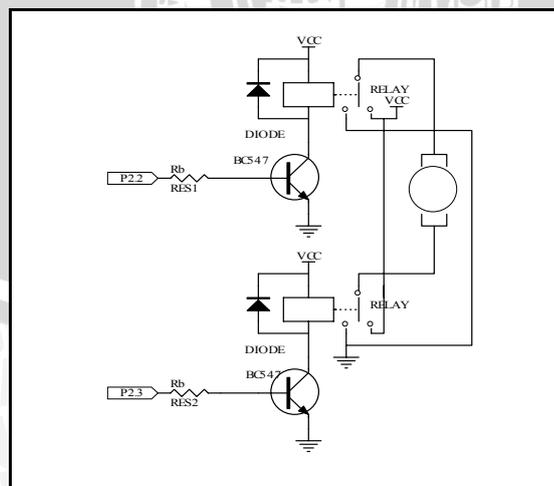
$$C_1 = 10 \times 0.1 \mu F$$

$$C_1 = 1 \mu F$$

Dalam perancangan  $C_2$  diambil nilai  $0.1 \mu F$ . Hal ini dikarenakan sesuai ketentuan dari *datasheet* MM74C922 nilai yang diijinkan antara  $0.01 \mu F - 10 \mu F$ .

#### 4.4.3 Perencanaan Rangkaian *Driver* Motor

Pada bagian ini berfungsi untuk mengaktifkan motor setelah mendapatkan data masukan dari mikrokontroller. Rangkaian ini terdiri dari transistor sebagai switching, relay dan sebuah dioda. Adapun gambar rangkaiannya ditunjukkan dalam Gambar 4.4.



**Gambar 4.4** Rangkaian Driver

Untuk menggerakkan motor melalui *driver relay*. Digunakan port 2 dalam perancangan ini, relay yang digunakan dapat beroperasi dengan tegangan 12 Dc. Arus minimal yang diperlukan untuk dapat mengaktifkan *relay* adalah 20 mA dari pengukuran. Berdasarkan datasheet data  $I_{OH} = 60 \mu\text{A}$  (port 2) dengan  $V_{OH} = 2,4 \text{ V}$ . Arus tersebut terlalu kecil untuk dapat menggerakkan relay. Transistor yang digunakan minimal mempunyai penguatan sebesar :

$$hfe = \frac{I_c}{I_b} = \frac{20 \cdot 10^{-3}}{60 \cdot 10^{-6}} = 333,33\Omega$$

Maka transistor yang dipakai dalam perancangan ini adalah tipe BC547. Transistor NPN ini mampu mendorong arus kolektor maksimum sebesar 100mA dan memiliki jangkauan nilai penguatan sampai dengan 800.

Nilai  $R_b$  didapat dari persamaan :

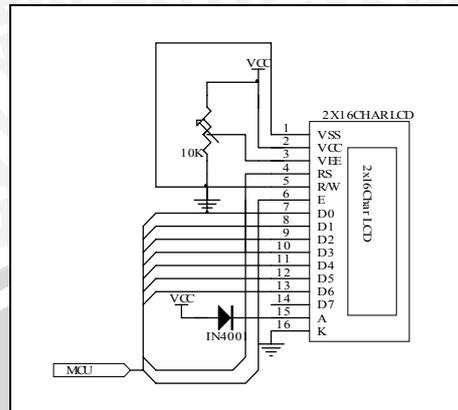
$$R_b = \frac{V_{OH} - V_{BE}}{I_B}$$

$$R_b = \frac{2,4 - 0,7}{60 \cdot 10^{-6}} = 28,33k\Omega$$

#### 4.4.4 Perencanaan Bagian Tampilan

##### 4.4.4.1 Bagian LCD

Sebagai penampil sistem ini, port 0 mikrokontroler digunakan sebagai jalur data bagi modul penampil LCD. Port 0 ini harus di *pull-up* 10 k $\Omega$  ke VCC agar mempunyai kondisi logika yang benar, karena port 0 tidak mempunyai resistor internal seperti pada port-port lain. Sedangkan sebagai sinyal-sinyal control untuk modul LCD, 2 buah pin dari port 2 yaitu port 2.6 dan port 2.7 masing-masing dimanfaatkan sebagai jalur control RS dan EN. Dengan mengubah nilai VEE menggunakan resistor variabel akan diperoleh tingkat kecerahan yang berbeda pada tampilan LCD. Dioda D1 dipasang antara VCC dan terminal BL+ untuk mencatu lampu datar tampilan LCD pada tegangan 4,3 V. Rangkaian antarmuka modul LCD ditunjukkan dalam Gambar 4.5



Gambar 4.5 Rangkaian Penampil

LCD tipe M1632 ini memiliki spesifikasi sebagai berikut:

- LCD ini terdiri dari 32 karakter dengan 2 baris masing-masing 16 karakter dengan *display* dot matrik 5x7
- Karakter generator ROM dengan 192 tipe karakter
- Karakter generator RAM dengan 8 bit karakter
- 80x8 bit display data RAM
- Dapat diinterfacekan ke MCU 8 atau 4
- Dilengkapi fungsi tambahan antara lain *display clear*, *cursor home*, *display on/off*, *cursor on/off*, *display character blink*, *cursor shift*, *display shift*
- *Internal data*
- *Internal* otomatis, reset pada saat *power on*
- Tegangan +5 volt PSU tunggal

Tabel 4.2. Fungsi-fungsi terminal pada LCD

Nama Sinyal	No.Term	I/O	Tujuan	Fungsi
DB <sub>0</sub> -DB <sub>3</sub>	4	I/O	MPU	Sebagai lalu lintas data dan instruksi ke dan dari MPU, <i>lower byte</i>
DB <sub>4</sub> -DB <sub>7</sub>	4	I/O	MPU	Sebagai lalu lintas data dan instruksi ke dan dari MPU, <i>lower byte</i>
DIMENSI	1	I	MPU	Sinyal <i>Start (read/write)</i>
R/ $\bar{W}$	1	I	MPU	Sinyal Seleksi Register 0: <i>write</i> 1: <i>Read</i>
RS	1	I	MPU	Sinyal Seleksi Register 0: Instruksi Register <i>Busy Flag &amp; @ (read)</i>
VEE	1	-	PSU	<i>Driver LCD</i>
VDD	1	-	PSU	5 volt
VSS	1	-	PSU	<i>Ground terminal</i> : 0 volt

Sumber: *EL-TECH Electronic*, 1987:50

#### 4.4.4.2 Bagian 7-Segment

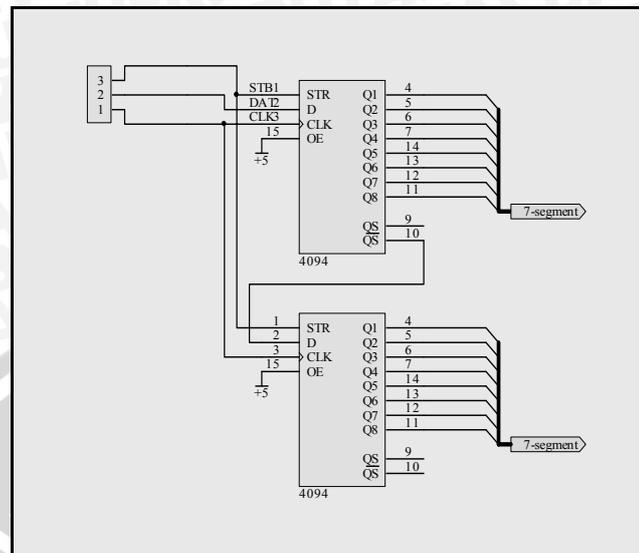
*7-segment* dalam sistem ini berfungsi untuk menampilkan informasi ketinggian yang dilalui pemain bagi penonton.

##### a) Shift Register

Pada sistem ini untuk mengirim data dan mengendalikan *7-segment* menggunakan IC 4094 yaitu IC shift register yang digunakan untuk menggeser data masukkan secara serial yang berasal dari mikrokontroler yang kemudian menghasilkan data keluaran secara paralel untuk kemudian digunakan sebagai data dari *7-segment*.

Penggunaan IC 4094 yaitu sebuah IC digunakan untuk men-*driver* sebuah *7-segment*. Salah satu keunggulan IC tipe ini adalah IC tersebut dapat dikaskade sehingga untuk sinyal *strobe* dan *clock* dapat menjadi satu jalur. Sedangkan untuk mengkaskade IC tersebut dengan cara keluaran pada pin QS pada IC pertama berfungsi sebagai data masukkan pada IC berikutnya.

Masukkan shift register 4094 (*strobe*, data dan *clock*) berasal dari mikrokontroler port 3.0, port 3.1 port 3.4. Data yang keluar dari *output* paralel *shift* register akan dimasukkan ke katoda dari *7-segment*. Gambar 4.6 menunjukkan penggunaan IC 4094 sebagai *driver* bagi *7-segment*.

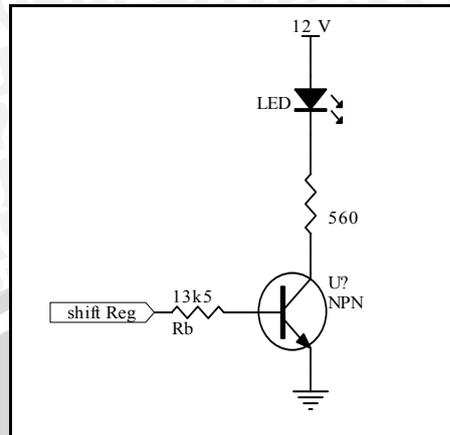


Gambar 4.6. IC 4094 Sebagai Driver 7-segment

b) 7-segment

*Display 7-Segment* ini merupakan gabungan dari beberapa LED yang dapat dicatu secara individu atau masing-masing. Pada perancangan ini digunakan *7-segment common anoda* sebanyak 2 pasang modul. *Common anoda* berarti semua anoda LED dihubungkan menjadi satu ke positif catu daya sedangkan katodanya difungsikan sebagai masukan. LED akan menyala apabila katodanya diberi masukan logika 0.

Penggunaan *7-segment* ini harus dipasang resistor seri dan transistor terhadap setiap LED untuk membatasi arus yang mengalir. Berdasarkan *datasheet* Tegangan maju (*forward voltage*) untuk *7-segment* yang digunakan sebesar 2 Volt. Dan arus maju (*forward current*) sebesar 20 mA. Adapun gambar rangkaian *display 7-segment* ditunjukkan dalam Gambar 4.7.



Gambar 4.7 Rangkaian Display 7-segment

Pada perencanaan ini dibuat arus yang melewati sebuah LED adalah 20 mA, dan  $V_{LED}$  adalah 2 Volt sehingga resistor yang dibutuhkan adalah :

$$R_C = \frac{V_{CC} - V_{LED} - V_{Ce(sat)}}{I_{LED}} = \frac{12 - 2 - 0,1}{20 \cdot 10^{-3}}$$

$$= 495 \Omega \approx 560 \Omega$$

Untuk nilai  $R_B$  dapat diperoleh dengan :

$$I_C = I_{LED} = I_B \cdot \beta$$

$$I_B = \frac{I_C}{\beta} = \frac{20 \cdot 10^{-3}}{64} = 0,3125 mA$$

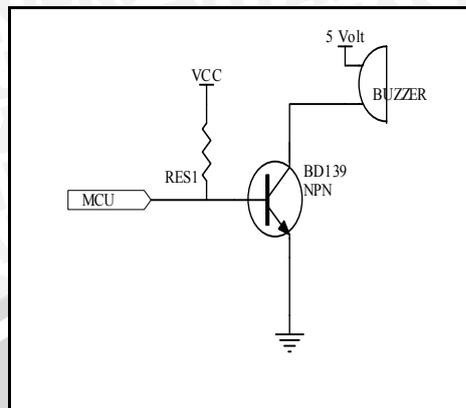
Maka nilai R sebesar:

$$R_b = \frac{V_{oh} - V_{BE}}{I_B} = \frac{5 - 0,8}{0,3125 \cdot 10^{-3}} = 13440 \Omega$$

$$\approx 13k5 \Omega$$

#### 4.4.5 Perancangan buzzer sebagai Indikator

Buzzer disini digunakan sebagai indikator bila mistar jatuh maka buzzer akan berbunyi. Gambar rangkaiannya ditunjukkan dalam Gambar 4.8.



Gambar 4.8 Rangkaian Buzzer

Ditetapkan Vcc yang digunakan pada perancangan sebesar 5 volt dan dirancang menggunakan transistor BD 139 dimana nilai  $I_c = I_{\text{Buzzer}} = 22 \text{ mA}$  maka diperoleh nilai tahanan sebesar:

$$I_B = \frac{I_c(\text{sat})}{\beta}$$

$$I_B = \frac{22 \cdot 10^{-3}}{250}$$

$$I_B = 0,088 \text{ mA}$$

Karena Ioh dari MCU sebesar 60 uA sehingga diperlukan arus tambahan sebesar 28 uA untuk memenuhi kebutuhan arus. Maka dibutuhkan R pullup sebesar :

$$R = \frac{V_{cc} - V_{be}}{I_{\text{Pullup}}}$$

$$R = \frac{5 - 0,7 \text{ V}}{28 \cdot 10^{-6}}$$

$$R = 153571 \Omega$$

$$R \approx 160 \text{ k}\Omega$$

#### 4.4.6. perancangan Bagian Kontrol dan Pengolah Data

Mikrokontroler didalam perancangan ini merupakan komponen utama, karena komponen mengontrol semua *switch* dan merupakan otak dari sistem agar dapat bekerja dengan baik dan optimal.

Mikrokontroler AT89C51 digunakan sebagai kendali sistem alat ini. Pemilihan mikrokontroler AT89C51 mengingat MCS ini mempunyai internal memori sehingga tidak perlu menambah memori luar, selain itu MCS ini sangat mudah didapatkan dipasaran dengan harga yang relatif murah.

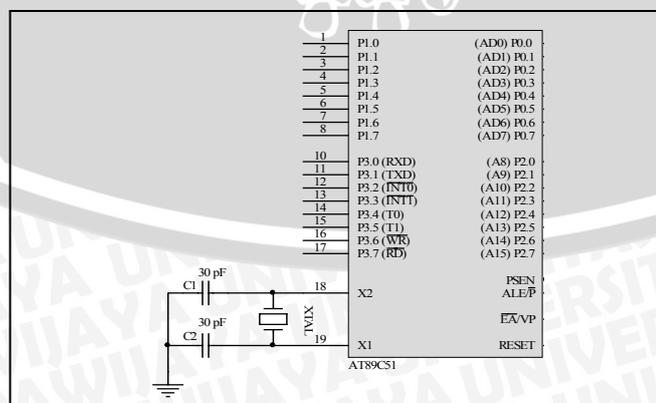
Mikrokontroler berfungsi sebagai pembaca masukkan dari *keypad*, mikrokontroler kemudian memerintahkan motor untuk bergerak naik atau turun dan sensor akan bekerja dengan membaca lubang sesuai posisi ketinggian yang diinginkan. Apabila posisi ketinggian yang baru sudah didapat maka mikrokontroler akan menunjukkan data tersebut dengan memerintahkan LCD dan *7-segment* sebagai *display*

Perancangan MCS 89C51 dapat dijelaskan sebagai berikut:

#### 4.4.6.1 Rangkaian Osilator

Osilator internal MCS AT89C51 dapat dibangkitkan dengan menggunakan kristal sebagai pembangkit pulsa dimana besarnya kristal yang diijinkan sebesar 0 – 24 MHz. Dalam perancangan ditentukan sebesar 12 MHz, sehingga kecepatan pelaksanaan intruksi persiklus sebesar 1  $\mu$ s ( $((1/12\text{MHz}) \times 12 \text{ siklus perioda})$ ).

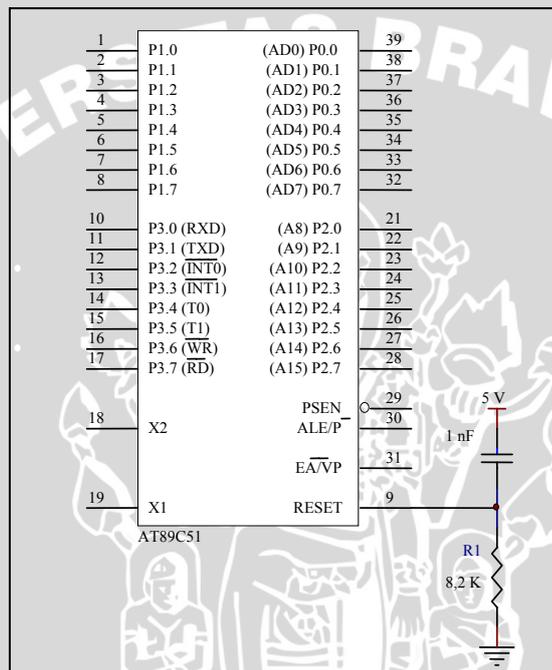
Gambar 4.8 menunjukkan rangkaian osilator yang digunakan. Kristal dihubungkan ke kaki 18 dan 19 pada MCS AT89C51 dengan menambahkan C<sub>1</sub> dan C<sub>2</sub> sebesar 30 pF. Penentuan nilai kapasitor sebesar 30 pF berdasarkan *datasheet*. (*Atmel datasheet* : 4)



Gambar 4.9. Rangkaian Osilator

#### 4.4.6.2. Rangkaian Reset

Untuk mereset mikrokontroller AT89C51, pin RST harus diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal reset, kapasitor dihubungkan dengan Vcc dan sebuah resistor dihubungkan ke ground. Selain itu diantara kapasitor dipasang sebuah saklar untuk memberikan sinyal reset secara manual. Rangkaian reset ditunjukkan dalam Gambar 4.10.



Gambar4.10 Rangkaian reset

Karena kristal yang digunakan mempunyai frekuensi sebesar 12 MHz, maka satu periode membutuhkan waktu sebesar :

$$T = \frac{1}{f_{XTALL}} = \frac{1}{12MHz} s = 0,833 \cdot 10^{-8} s$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk mereset mikrokontroler adalah :

$$\begin{aligned} t_{reset(min)} &= T \times \text{periode yang dibutuhkan} \\ &= 0,833 \cdot 10^{-8} \times 24 \\ &= 2 \mu S \end{aligned}$$

Jadi mikrokontroler membutuhkan waktu minimal  $2 \mu\text{s}$  untuk mereset. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dari persamaan 2.4 dengan menentukan nilai  $R = 8,2 \text{ k}\Omega$

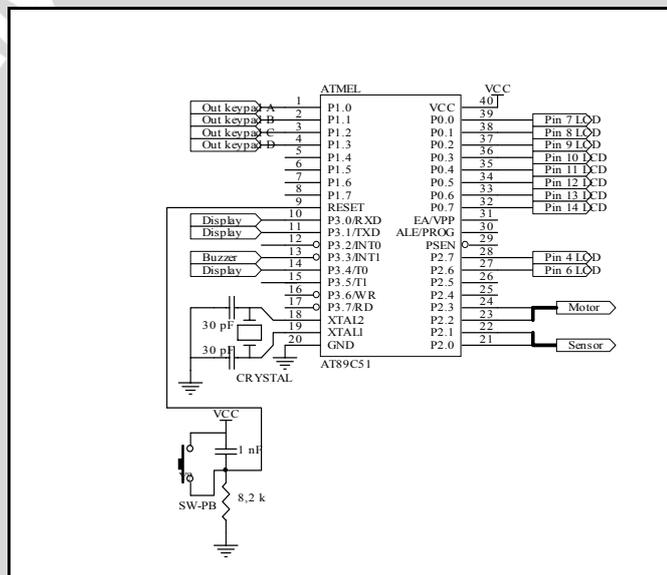
$$t = 0,357 \cdot R \cdot C$$

$$2 \cdot 10^{-6} = 0,357 \cdot 8,2 \cdot 10^3 C$$

$$C = 0,683 \cdot 10^{-9} F$$

Jadi dengan nilai komponen  $R = 8,2 \text{ k}\Omega$  nilai kapasitor yang dapat memenuhi syarat untuk mereset mikrokontroler harus diatas  $0,683 \text{ nF}$ . Untuk kemudahan perancangan dipilih  $C = 1 \text{ nF}$ .

Rangkaian AT89C51 sebagai kontrol ditunjukkan dalam Gambar 4.11



**Gambar 4.11** Rangkaian Mikrokontroler AT89C51

Pin-pin yang digunakan pada IC AT89C51 adalah sebagai berikut:

- 1) Port 1 (P1.0 – P1.4)

Digunakan sebagai keluaran dari IC encoder 74C922 sebagai masukan data.

- 2) Port 0 (P0.0 – P0.7)

Digunakan sebagai keluaran data ke LCD atau sebagai masukan ke LCD data

- 3) Port 2 (P2.6 – P2.7)

Digunakan sebagai keluaran dan masukan data. Berikut deskripsi I/O-nya;

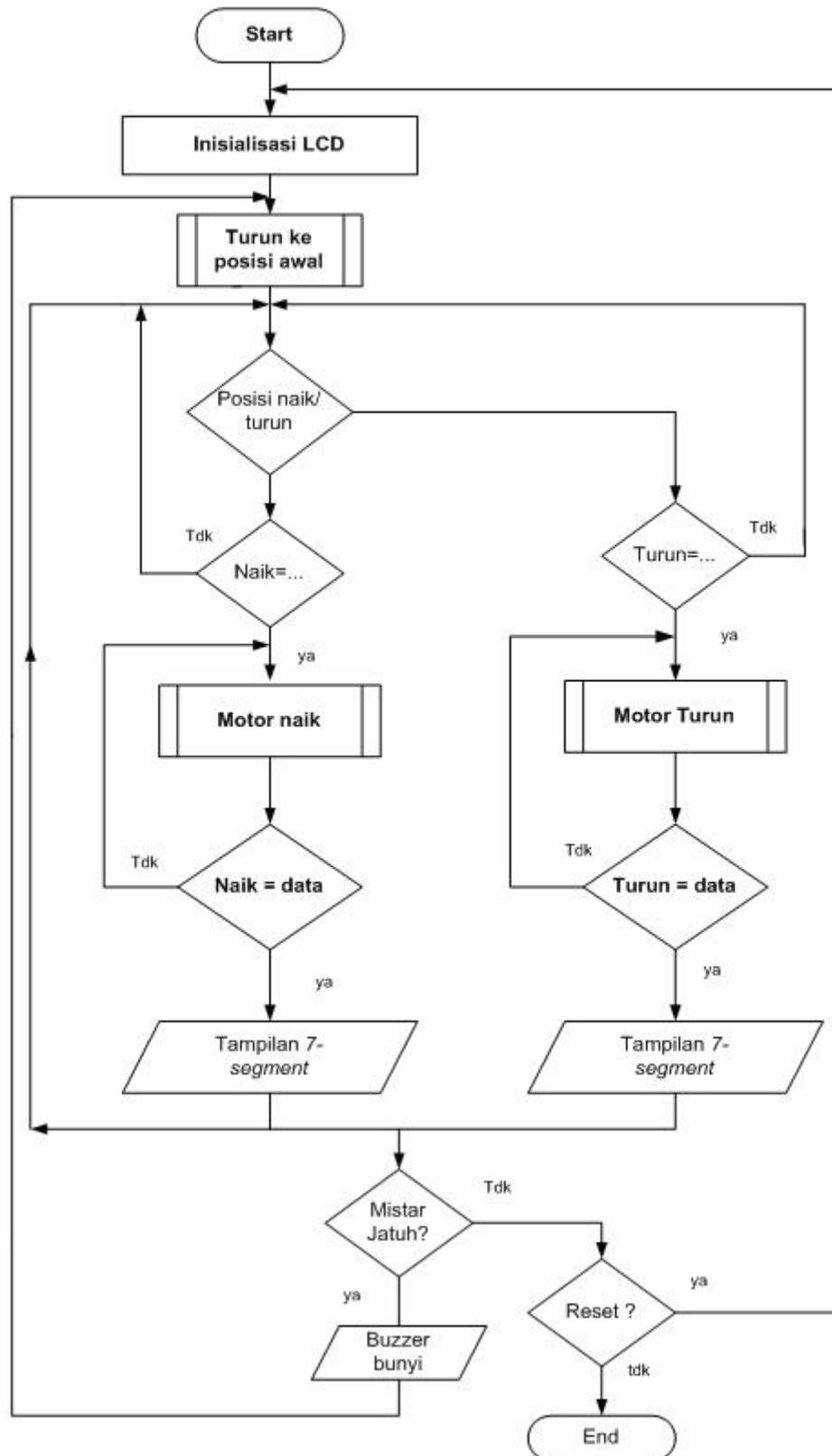
- P2.6 – P2.7 : digunakan untuk mengirimkan data kontrol LCD.

- 4) Port 2 (P2.0 - P2.4)

- P2.0 – P2.1 : digunakan sebagai masukan data dari sensor
  - P2.2 – P2.3 : digunakan sebagai keluaran dari motor
- 5) Port 3 (P3.0 – P3.4)
- P3.0, P3.1, P3.4 : Digunakan sebagai keluaran data *7-segment*.
  - P3.3 : Digunakan sebagai keluaran dari *Buzzer*.
- 6) X1 dan X2
- Digunakan sebagai masukan bagi rangkaian osilator kristal yang akan membangkitkan pulsa *clock* yang akan menggerakkan seluruh operasi internal mikrokontroler.
- 7) Reset
- Digunakan sebagai *input reset*. Kondisi tinggi dari pin ini selama dua siklus *clock* akan menyebabkan *reset device* ini yaitu selama 2 $\mu$ s.

#### 4.4.7 Perangkat Lunak

Perangkat lunak yang dirancang dengan menggunakan bahasa assembler mikrokontroler MCS-51. Untuk memberikan gambaran umum jalannya program dan memudahkan pembuatan perangkat lunak, maka dibuat *flowchart* yang menunjukkan jalannya program. Gambar 4.12 menunjukkan diagram alir utama.



Gambar 4.12 Flowchat Program Utama

## BAB V

### PENGUJIAN DAN ANALISIS ALAT

Tujuan pengujian alat ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perencanaan. Pengujian dilakukan dengan cara menguji rangkaian setiap blok secara terpisah. Pengujian setiap blok ini dilakukan untuk mempermudah analisis apabila alat ini tidak bekerja sesuai dengan perencanaan.

Dalam pelaksanaan pengujian dilakukan dengan dua cara yaitu secara perangkat keras dan perangkat lunak. Secara perangkat keras dilakukan melalui pemeriksaan sambungan pengawatan dan pengukuran dengan alat-alat ukur. Sedangkan pengujian perangkat lunak, pengujian dilakukan melalui pembuatan *software* dan hasilnya diamati dengan bantuan alat peraga atau melalui alat ukur.

Pengujian ini dibagi menjadi:

- a) Pengujian sensor tinggi.
- b) Pengujian rangkaian  *keypad*.
- c) Pengujian rangkaian motor.
- d) Pengujian  *display* LCD.
- e) Pengujian  *display 7-segment*.
- f) Pengujian  *buzzer*.

#### 5.1 Pengujian Rangkaian Sensor.

##### 5.1.1 Tujuan

Pengujian ini bertujuan untuk mengetahui apakah sensor dapat bekerja dengan baik atau tidak.

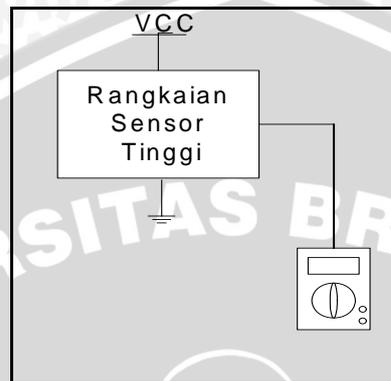
##### 5.1.2 Peralatan yang Digunakan

1. Catu daya +5V DC
2. Rangkaian sensor
3. Multimeter
4.  *Projectboard*

### 5.1.3 Prosedur Pengujian

Langkah-langkah pengujian adalah sebagai berikut :

1. Merangkai rangkaian blok pengujian sensor seperti ditunjukkan dalam Gambar 5.1 berikut :



Gambar 5.1 Blok pengujian sensor Tinggi.

2. Menghubungkan catu daya 5 volt pada rangkaian
3. Menghubungkan multimeter digital pada bagian keluaran rangkaian sensor.
4. Hasil pengujian terdapat pada multimeter digital.

### 5.1.4 Hasil Pengujian dan Analisis

Dari pengujian didapatkan hasil seperti ditunjukkan dalam Tabel 5.1 Saat kondisi opto tidak terhalangi (On) dan kondisi opto saat terhalangi (Off), hal ini mengindikasikan bahwa opto dapat bekerja dengan baik.

Tabel 5.1 Hasil pengujian Rangkaian sensor Optocoupler

Parameter	Pengukuran
V saat On	0,25 v
V saat Off	4,96 v
I saat On	0,47 mA
I saat off	0 mA

Dari hasil pengujian diatas bahwa saat sensor terhalang (*Off*) maka memberikan logika tinggi sehingga apabila dihubungkan dengan mikrokontroler

tidak dapat sebagai data. Karena sensor ini bergerak naik dan turun maka dapat terjadi kondisi saat sensor mengenai lubang sehingga tidak terhalangi (*On*) dengan demikian keluaran berlogika rendah. Keluaran ini diambil pada kolektor fototransistor. Prinsip kerja dari sensor ini jika fototransistor mendapat pancaran dari infra red maka fototransistor dalam kondisi *on* sehingga kaki kolektor seperti terhubung dengan kaki emitor yang terhubung ground.

## 5.2 Pengujian Rangkaian Keypad.

### 5.2.1 Tujuan

1. Mengetahui keluaran encoder 74C922.
2. Mengetahui sinyal DA (Data Available) IC 74C922 yang digunakan sebagai sinyal indicator adanya penekanan *keypad*.

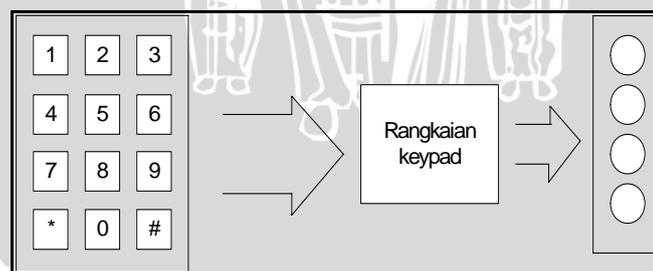
### 5.2.2 Peralatan yang Digunakan

1. Unit papan tombol
2. Catu daya +5volt
3. *Projectboard*

### 5.2.3 Prosedur Pengujian

Langkah langkah pengujian adalah sebagai berikut :

1. Merangkai rangkaian pengujian *keypad* seperti ditunjukkan dalam Gambar 5.2



Gambar 5.2 Diagram blok pengujian rangkaian keypad

2. Menghubungkan catu daya 5 V
3. Menghubungkan keluaran *encoder keypad* dengan LED
4. Menekan tombol secara bergantian
5. Mengamati hasil pengujian dengan memperhatikan nyala masing-masing LED

### 5.2.4 Hasil Pengujian dan Analisis

Dari pengujian didapatkan hasil seperti ditunjukkan dalam Tabel 5.2 Saat penekanan tombol *keypad* DA menyala, hal ini mengindikasikan *keypad* bisa dipakai.

**Tabel 5.2** Tabel Hasil pengujian rangkaian *Keypad*.

Tombol yang Ditekan	DATA				
	A	B	C	D	DA
1	0	0	0	0	1
2	1	0	0	0	1
3	0	1	0	0	1
4	0	0	1	0	1
5	1	0	1	0	1
6	0	1	1	0	1
7	0	0	0	1	1
8	1	0	0	1	1
9	0	1	0	1	1
0	1	0	1	1	1
*	0	0	1	1	1
#	0	1	1	1	1

## 5.3 Pengujian Rangkaian Driver Motor

### 5.3.1 Tujuan

Untuk mengetahui apakah rangkaian *driver* motor telah berfungsi dengan baik. Perangkat lunak yang digunakan adalah dengan memberikan data keluaran pada modul system minimum AT89C51 yang langsung dihubungkan rangkaian *driver*.

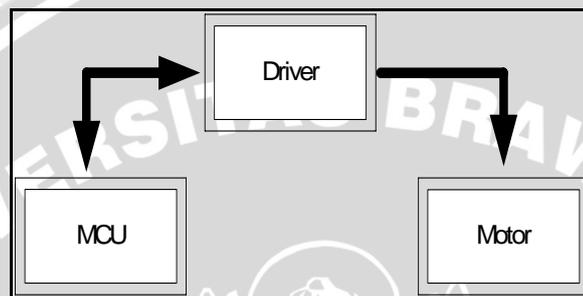
### 5.3.2 Peralatan yang digunakan

1. Seperangkat computer
2. Modul programmer AT89C51
3. Modul system minimum AT89C51

4. Rangkaian *driver* motor.
5. Motor sebagai peraga.
6. Catu daya 12 volt.

### 5.3.3 Prosedur Pengujian

1. Merangkai peralatan seperti dalam Gambar 5.3



Gambar 5.3 Blok diagram Pengujian *Driver* Motor

2. Cek semua rangkaian sudah benar apa belum, kemudian hubungkan dengan MCU dan motor.
3. Mengamati hasil pengujian pada alat peraga yang berupa motor.

### 5.3.4 Hasil pengujian dan Analisis

Dari hasil pengujian yang ditampilkan melalui alat peraga berupa motor, didapat bahwa data yang diberikan sesuai dengan nyalanya motor dan arah motor. Untuk lebih jelasnya dapat dilihat dalam Tabel 5.3.

Tabel 5.3. Hasil Pengujian Rangkaian *Driver* Motor

Arah Putaran Motor	Input 1 (MCU P2.2)	Input 2 (MCU P2.3)
Naik	0	1
Turun	1	0
Diam	0	0

## 5.4 Pengujian *display* LCD.

### 5.4.1 Tujuan

Untuk mengetahui apakah *display* LCD dapat bekerja dengan baik atau tidak

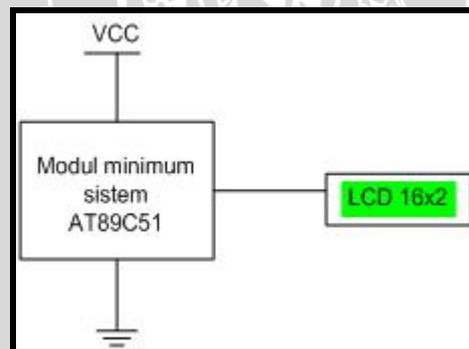
### 5.4.2 Peralatan yang Digunakan

1. Rangkaian *display* LCD.
2. Modul minimum sistem mikrokontroler AT89C51
3. *Projectboard*.
4. Catu daya +5volt.

### 5.4.3 Prosedur Pengujian

Langkah-langkah pengujian adalah sebagai berikut:

- 1) Merangkai blok rangkaian pengujian *display* LCD seperti dalam Gambar 5.4.



Gambar 5.4 Blok Pengujian *display* LCD

- 2) Menghubungkan modul minimum sistem mikrokontroler AT89C51 dengan blok rangkaian pengujian *display* LCD.
- 3) Membuat program pengujian LCD.
- 4) Memberikan catu daya +5volt pada rangkaian *display* LCD dan modul minimum sistem mikrokontroler AT89C51.
- 5) Hasil pengujian terdapat pada *display* LCD

#### 5.4.4 Hasil Pengujian dan Analisis

Dari hasil pengujian diperoleh hasil bahwa modul *display* LCD mampu menampilkan karakter “MISTAR LOMPAT GALAH” dengan baik sesuai dengan instruksi yang diberikan. Dengan demikian *display* LCD dapat berfungsi dan dapat diakses oleh perangkat lunak. Gambar 5.5 menunjukkan hasil pengujian *display* LCD.



Gambar 5.5 Hasil Pengujian *display* LCD

#### 5.5 Pengujian *display* 7-segment.

##### 5.5.1 Tujuan

Untuk mengetahui apakah Display 7-segment dapat bekerja dengan baik atau tidak.

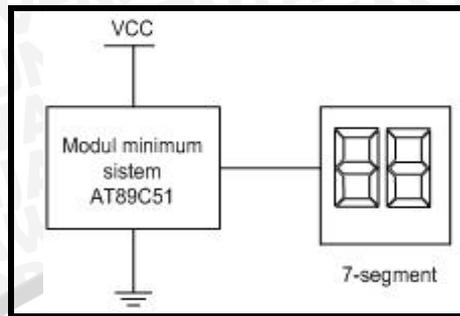
##### 5.5.2 Peralatan yang Digunakan

1. Rangkaian *display* 7-segment
2. Modul minimum sistem mikrokontroler AT89C205
3. *Projectboard*
4. Catu daya +12V DC

##### 5.5.3 Prosedur Pengujian

Langkah-langkah pengujian adalah sebagai berikut:

- 1) Merangkai blok rangkaian pengujian *display* 7-segment seperti dalam Gambar 5.6.



Gambar 5.6 Blok Pengujian *display 7-segment*.

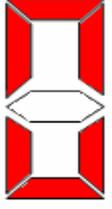
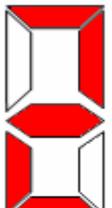
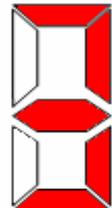
- 2) Menghubungkan modul minimum sistem mikrokontroler AT89C51 dengan blok rangkaian pengujian *display 7-segment*.
- 3) Membuat program pengujian *7-segment*.
- 4) Memberikan catu daya +12V pada rangkaian *display 7-segment* dan +5V pada modul minimum sistem mikrokontroler AT89C51.
- 5) Hasil pengujian terdapat pada *display 7-segment*.

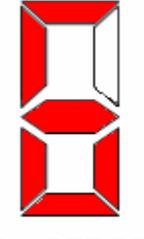
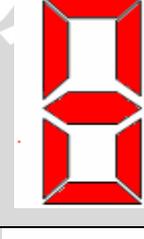
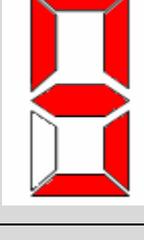
#### 5.5.4 Hasil Pengujian dan Analisis

Dari hasil percobaan didapat bahwa mikrokontroler akan langsung mengirimkan data ke *driver 7-segment* sesuai dengan data hasil pengolahan pada mikrokontroler. Tabel 5.5 menunjukkan data yang dikirim oleh mikrokontroler berdasarkan tampilannya pada *7-segment*. Berikut adalah contoh penulisan *database* pada mikrokontroler sebagai data yang akan dikirim ke *7-segment*.

DB	0FCH	;0 DESIMAL
DB	60H	;1 DESIMAL
DB	0DAH	;2 DESIMAL
DB	0F2H	;3 DESIMAL
DB	66H	;4 DESIMAL
DB	0B6H	;5 DESIMAL
DB	0BEH	;6 DESIMAL
DB	0E0H	;7 DESIMAL
DB	0FEH	;8 DESIMAL
DB	0F6H	;9 DESIMAL

Tabel 5.5 Data mikrokontroler berdasarkan Tampilannya pada 7-segment

Desimal	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0	Heksadesimal	Tampilan
	A	B	C	D	E	F	G	X		
0	1	1	1	1	1	1	0	0	FC	
1	0	1	1	0	0	0	0	0	60	
2	1	1	0	1	1	0	1	0	DA	
3	1	1	1	1	0	0	1	0	F2	
4	0	1	1	0	0	1	1	0	66	
5	1	0	1	1	0	1	1	0	B6	

6	1	0	1	1	1	1	1	0	BE	
7	1	1	1	0	0	0	0	0	E0	
8	1	1	1	1	1	1	1	0	FE	
9	1	1	1	1	0	1	1	0	F6	

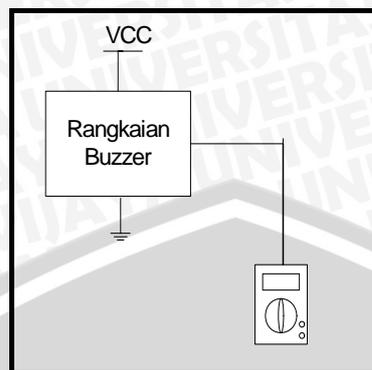
## 5.6 Pengujian Rangkaian *Buzzer*.

### 5.6.1 Tujuan

Untuk mengetahui apakah *buzzer* berfungsi dengan baik atau tidak.

### 5.6.2 Peralatan yang Digunakan

1. Merangkai blok rangkaian pengujian *Buzzer* seperti dalam Gambar 5.7.



Gambar 5.7 Pengujian Buzzer

2. Menghubungkan catu daya +5V pada rangkaian.
3. Hasil pengujian terdapat dalam Tabel 5.6

### 5.6.3 Hasil Pengujian dan Analisis

Dari hasil pengujian didapat nilai tegangan dan arus saat *buzzer* On dan Off seperti dalam Tabel 5.6.

Tabel 5.6 Hasil Pengujian Rangkaian Buzzer

Parameter	Pengukuran
V On	5 v
V Off	1,4 v
Ic_sat	6,25 mA
Ib_sat	0,2 mA

## 5.7 Pengujian Keseluruhan

### 5.7.1 Tujuan

Pengujian keseluruhan bertujuan untuk mengetahui kinerja sistem secara keseluruhan, apakah sesuai dengan spesifikasi yang direncanakan di awal dan mengetahui kesalahan dari alat tersebut.

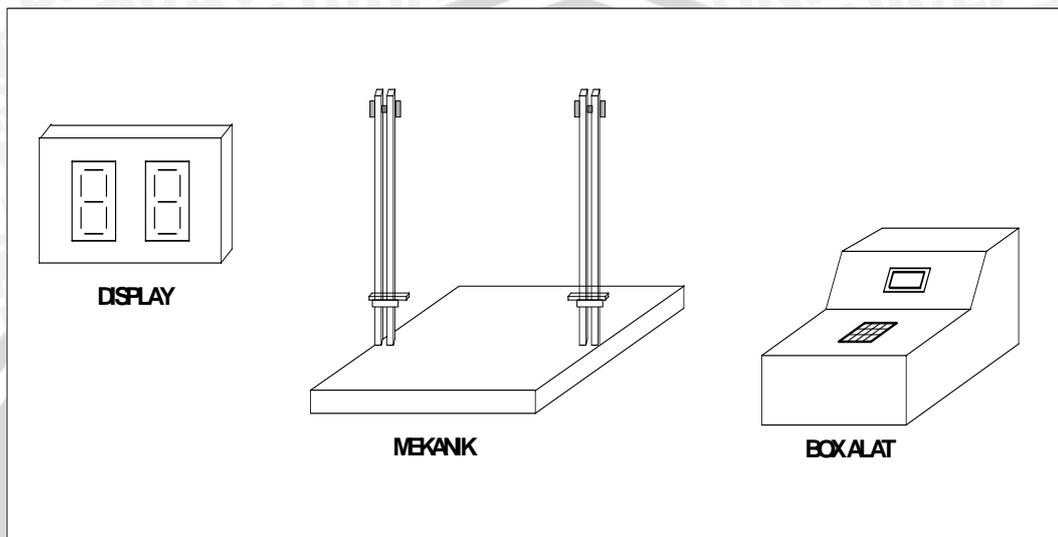
### 5.7.2 Peralatan Yang digunakan

1. Catu daya +5V dan +12V
2. Rangkaian system secara keseluruhan
3. Miniatur tiang lompat Tinggi Galah

### 5.7.3 Prosedur pengujian

Langkah-langkah pengujian adalah sebagai berikut :

1. Merangkai blok pengujian alat seperti ditunjukkan dalam Gambar 5.8



Gambar 5.8. Blok Pegujian alat

2. Pastikan semua rangkaian sudah benar sebelum menghubungkan ke sumber tegangan.
3. Hasil pengujian terdapat pada *display* LCD dan *7-segment*.

### 5.7.4 Hasil pengujian dan Analisis

Tabel 5.7 Hasil Pengujian Alat secara Keseluruhan

No	Ketinggian yang diminta (Cm)	Posisi mistar		Tampilkan LCD	Display 7-segment	Buzzer	Posisi mistar pengukuran (Cm)	% Eror
		Naik(Cm)	Turun(Cm)					
1	25	25	-	Naik	25	●	25,1	0,4
2	35	10	-	Naik	35	●	35,1	0,28
3	45	10	-	Naik	45	●	45,1	0,4
4	49	4	-	Naik	49	●	49,1	0,20
5	50	1	-	Naik	50	●	50	0
6	47	-	3	Turun	47	●	47,1	0,21
7	37	-	10	Turun	37	●	37,1	0,27
8	29	-	8	Turun	29	●	29,1	0,34
9	26	-	3	Turun	26	●	26,1	0,38
10	20(mistar Jatuh)	-	-	Ke posisi awal	00	●	-	-
Persentase rata-rata kesalahan (error)								0,284

Ket: Saat ON Saat OFF



Dari hasil pengujian diatas prototype ini mampu menaikkan dan menurunkan mistar sesuai dengan data ketinggian yang diminta. Dari pengujian secara keseluruhan didapatkan hasil bahwa alat ini telah bekerja sesuai dengan perencanaan dengan kesalahan rata-rata 0,284%. Alat ini dapat naik dan turun sesuai ketinggian yang diinginkan melalui keypad. Data yang diinginkan dapat ditampilkan dengan baik pada 7-segment yaitu berupa ketinggian yang baru. Alat ini mampu menaikkan mistar pada ketinggian maksimum 50 cm, dan bila ketinggian yang diinginkan lebih dari 50 cm maka tidak dapat di enter sehingga mistar tidak naik.



## BAB VI PENUTUP

### 6.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis alat maka dapat ditarik kesimpulan sebagai berikut:

- 1) Range ketinggian yang digunakan maksimum 50 cm.
- 2) Prototype pengatur ketinggian mistar pada olahraga lompat galah ini mampu mengatur ketinggian mistar sesuai data yang diinginkan dengan kesalahan pembacaan sekitar 0,284%.
- 3) Karena keterbatasan mekanik, maka setiap mistar jatuh push button ditekan sebagai indikator dan untuk kembali ke posisi awal.

### 6.2 Saran

Alat pengatur ketinggian mistar ini dapat dikembangkan lagi agar lebih sempurna dalam penggunaannya, diantaranya dapat dilakukan dengan:

- 1) Bentuk fisik dari alat dapat dibuat lebih besar dengan range ketinggian yang lebih tinggi lagi sehingga mempermudah pemakai dalam menggunakan alat ini.
- 2) Tampilan dari alat ini dapat dibuat lebih bervariasi tidak hanya menampilkan data ketinggian saja.
- 3) Dalam pembuatan mekanik hendaknya lebih dioptimalkan lagi sehingga keadaan mistar benar-benar presisi.

### Daftar Pustaka

- Agus Pracoyo, 2001, *MCS-51 Instruction Set, Architecture dan Hardware Design*, Politeknik Negeri Malang, Malang,
- Anonymous, 1998. *Liquid Crystal Displays, Standart Character Modules application Notes*. Seiko Instruments, germany.  
[http://www.seiko.com/seiko\\_manual.pdf](http://www.seiko.com/seiko_manual.pdf)
- Amonymous. *74C922 Datasheet, Fairchild Semikonduktor*-web document.  
<http://pdf1.alldatasheet.com/datasheet-pdf/view/53736/FAIRCHILD/74C922/datasheet.pdf>
- Malvino, Albert Paul. 1987. *Prinsip-prinsip Elektronika, Jilid 2, Cetakan ketiga*, terjemahan Prof. M. Barmawi. Jakarta : Erlangga.
- Nalwan, Paulus Andi. *Teknik antarmuka dan Pemrograman mikrokontroller AT89C51*. Jakarta: Elex Media Komputindo, 2003
- Prof. Dr. gunter Bernhard, 1986, *Prinsip Dasar Latihan loncat tinggi, jauh, jangkrit dan loncat galah*, Semarang : Effar offset.
- Suyono Ds, 2001, *Peraturan/Ketentuan Perlombaan atletik 2002-2003PASI*, Jakarta : IAAF-RDC
- U.Jonath, E.Haag, R.krempel, 1987, *Atletik*, terjemahan soeparmo, Jakarta: PT

# LAMPIRAN

- 
- I. Listing Program
  - II. Gambar Rangkaian lengkap
  - III. Foto Alat
  - IV. Data sheet
    1. Optocoupler
    2. KT C9013
    3. BD 135; BD 137; BD 139
    4. BC 546/ 547/ 548/549/550
    5. HCC/HCF 4094 B
    6. 7-Segmen Kingbright
    7. MM 74C922
    8. Seiko Manual LCD 16x2
    9. Atmel 89C51

```

=====
; PROGRAM LOMPAT TINGGI
=====

key          bit    p1.4
sen1         bit    p2.0
sen2         bit    p2.1
mot1         bit    p2.2
mot2         bit    p2.3
LCD_RS       bit    p2.6
LCD_EN       bit    p2.7
strobe       bit    p3.0
dat          bit    p3.1
buzer        bit    p3.3
clock        bit    p3.4
limit        bit    p3.5
puluhan      equ    30h
satuan       equ    31h
pulsa        equ    32h
counter      equ    35h
pul          equ    36h
sat          equ    37h
count        equ    38h
up           equ    39h
down         equ    3ah
nul          equ    3bh
bufer        equ    40h
bufer1       equ    41h
bu_dat       equ    42h
cursor       equ    43h

org          00h
ljmp        mulai

mulai:      org    100h
            mov    p3,#0ffh
            clr    buzer
            clr    strobe
            clr    dat
            clr    clock
            clr    mot1
            clr    mot2
            call   init_lcd
            call   cler
            call   konv
            call   display
            mov    dptr,#hep
            call   baris_atas
            call   baris_bawah
            mov    dptr,#blank
            call   baris_atas
            call   baris_bawah
            mov    dptr,#balik
            call   baris_atas
            call   baris_bawah
            call   blik
            mov    dptr,#pilih
            call   baris_atas
            call   baris_bawah
            mov    bufer,#0
            mov    bufer1,#0
            mov    cursor,#8ah

            scan_again:
            call   scan_key
            jb     limit,runn
            call   lim
            runn:  sjmp  scan_again

=====Clear data RAM
cler:       mov    r7,#0fh
            mov    r0,#30h
itali:      mov    a,#0
            mov    @r0,a
            inc   r0
            djnz  r7,itali
            ret

=====Limit
lim:        jnb    limit,$
            call   cler
            call   bunyi
            mov    dptr,#balik
            call   baris_atas
            call   baris_bawah
            call   blik
            call   konv
            call   display
            mov    dptr,#pilih
            call   baris_atas
            call   baris_bawah
            ret

=====
;SCANNING KEYPAD 3X4 PAKE
ENCODER
=====
scan_key:
            jnb    key,trus1
            mov    a,p1
            anl    a,#00001111b
            cjne   a,#00h,trus1
            mov    bu_dat,#1    ;data key = 1
            jb     key,$
            call   baris_a
            ret

trus1:
            jnb    key,trus2
            mov    a,p1
            anl    a,#00001111b
            cjne   a,#01h,trus2
            mov    bu_dat,#2    ;data key = 2
            jb     key,$
            call   baris_a
            ret

trus2:
            jnb    key,trus3
            mov    a,p1
            anl    a,#00001111b
            cjne   a,#02h,trus3
            mov    bu_dat,#3    ;data key = 3
            jb     key,$
            call   baris_a
            ret

```

```

trus3:
jnb    key,trus4
mov    a,p1
anl    a,#00001111b
cjne   a,#04h,trus4
mov    bu_dat,#4    ;data key = 4
jb     key,$
call   baris_a
ret

trus4:
jnb    key,trus5
mov    a,p1
anl    a,#00001111b
cjne   a,#05h,trus5
mov    bu_dat,#5    ;data key = 5
jb     key,$
call   baris_a
ret

trus5:
jnb    key,trus6
mov    a,p1
anl    a,#00001111b
cjne   a,#06h,trus6
mov    bu_dat,#6    ;data key = 6
jb     key,$
call   baris_a
ret

trus6:
jnb    key,trus7
mov    a,p1
anl    a,#00001111b
cjne   a,#08h,trus7
mov    bu_dat,#7    ;data key = 7
jb     key,$
call   baris_a
ret

trus7:
jnb    key,trus8
mov    a,p1
anl    a,#00001111b
cjne   a,#09h,trus8
mov    bu_dat,#8    ;data key = 8
jb     key,$
call   baris_a
ret

trus8:
jnb    key,trus9
mov    a,p1
anl    a,#00001111b
cjne   a,#0Ah,trus9
mov    bu_dat,#9    ;data key = 9
jb     key,$
call   baris_a
ret

trus9:
jnb    key,trus10
mov    a,p1
anl    a,#00001111b
cjne   a,#0Dh,trus10
mov    bu_dat,#0    ;data key = 0
jb     key,$
call   baris_a
ret

trus10:
jnb    key,trus11
mov    a,p1
anl    a,#00001111b
cjne   a,#0Ch,trus11 ;data key = *
mov    bu_dat,#10
jb     key,$
call   baris_a
ret

trus11:
jnb    key,akhir
mov    a,p1
anl    a,#00001111b
cjne   a,#0Eh,akhir
mov    bu_dat,#11
jb     key,$
call   baris_a    ;data key = #
ret

akhir:
ret

;=====inisialisasi LCD
init_lcd:
mov    a,#03fh
acall  write_inst
acall  write_inst
mov    a,#0dh
acall  write_inst
mov    a,#06h
acall  write_inst
mov    a,#01h
acall  write_inst
mov    a,#0ch
acall  write_inst
ret

;=====menulis instruksi ke LCD
write_inst:
clr    LCD_RS
mov    p0,A
clr    LCD_EN
setb   LCD_EN
acall  delay
ret

;=====menulis data ke LCD
write_data:
setb   LCD_RS
mov    p0,A
clr    LCD_EN
setb   LCD_EN
acall  delay
ret

;=====menulis data LCD baris atas
baris_atas:
mov    r3,#16
mov    a,#80h
acall  write_inst

tulisl:
clr    a
movc   a,@a+dptr
inc    dptr
acall  write_data
djnz  r3,tulisl
ret

```

```

;=====menulis baris bawah
baris_bawah:
    mov     r3,#16
    mov     a,#0C0h
    call    write_inst
tulis2:
    clr     a
    movc   a,@a+dptr
    inc    dptr
    call   write_data
    djnz   r3,tulis2
    ret

;=====routin delay
delay:
    mov     r0,#0
delay1:
    mov     r5,#50h
    djnz   r5,$
    djnz   r0,delay1
    ret

;=====routin Ldelay
Ldelay:
    mov     r2,#30h
Ld1:
    call    delay
    djnz   r2,Ld1
    ret

;-----
;KHUSUS UNTUK DATA KEYPAD
;-----
baris_a:
    mov     a,bufer
    cjne   a,#0,lop
    mov     a,bu_dat
    cjne   a,#10,nai    ;naik
    inc    up
    mov     bu_dat,#0
    mov     bufer1,#1
    mov     bufer,#1
    mov     dptr,#naik
    call   baris_atas
    call   baris_bawah
    ljmp   ouw
nai:     mov     a,bu_dat
    cjne   a,#11,ouw    ;turun
    mov     a,pul
    mov     b,#10
    mul    ab
    add    a,sat
    mov     pulsa,a
    cjne   a,#0,ulag    ;jika data = 0 keluar
    mov     bufer,#0
    mov     cursor,#8ah
    sjmp   ouw
ulag:   mov     bu_dat,#0
    inc    down
    mov     bufer1,#2
    mov     bufer,#1
    mov     dptr,#turun
    call   baris_atas

call    baris_bawah
ljmp   ouw
lop:   mov     a,bufer
    cjne   a,#1,ouw
    mov     a,bu_dat
    cjne   a,#10,nai    ;cancel
    mov     bu_dat,#0
    mov     bufer,#0
    mov     cursor,#8ah
    mov     dptr,#pilih
    call   baris_atas
    call   baris_bawah
    ljmp   ouw
naii:  mov     a,bu_dat
    cjne   a,#11,turr    ;enter
    mov     bu_dat,#0
    mov     bufer,#0
    call   eksekusi
gaki:  ljmp   ouw
turr:  call    pos_cur
    mov     dptr,#data_key ;0123456789
    mov     a,bu_dat
    movc   a,@a+dptr
    call   write_data
ouw:   ret
pos_cur:
    mov     a,cursor
    call   write_inst
    call   simpan
    inc    cursor
    mov     a,cursor
    cjne   a,#8ch,back_pos
    mov     cursor,#8ah
back_pos:
    ret
simpan:
    mov     a,cursor
    cjne   a,#8bh,gak
    mov     satuan,bu_dat
    sjmp   keluar
gak:
    mov     puluhan,bu_dat
    keluar:
    ret
eksekusi:
    call   save
    mov     a,pul
    mov     b,#10
    mul    ab
    add    a,sat
    mov     pulsa,a
    subb   a,#51        ;jika > 51 keluar
    jnc    milan
    sjmp   juve
milan:  mov     bufer,#1
    mov     cursor,#8ah
    sjmp   metu
juve:   mov     a,bufer1
    cjne   a,#1,mudn
    mov     dptr,#hi_gh

```

```

call baris_atas
call baris_bawah
call count_naik
call mungah
call jalan
mov a,#1
cjne a,nul,lau
mov nul,#0
sjmp metu
lau: mov dptr,#pilih
call baris_atas
call baris_bawah
mov bufer,#0
mov cursor,#8ah
sjmp get
mudn: mov a,bufer1
cjne a,#2,metu
mov dptr,#lo_w
call baris_atas
call baris_bawah
call count_turun
call mudun
call jalan
mov a,#1
cjne a,nul,laju
mov nul,#0
sjmp metu
laju: mov dptr,#pilih
call baris_atas
call baris_bawah
mov bufer,#0
mov cursor,#8ah
get: call konv
call display
metu: ret

save:
mov pul,puluhan ;simpan data masukan
mov sat,satuan
ret

count_naik:
mov a,pulsa
subb a,counter
mov count,a ;jumlah counter
mov a,up
cjne a,#1,up_1
mov down,#0
mov a,count
add a,#1 ;jumlah saat ditempat

asal
mov count,a ;jumlah counter
up_1: mov counter,pulsa ;data terakhir
ret

count_turun:
mov a,pulsa
subb a,counter
cpl a
add a,#1 ;jumlah kebalikan
mov count,a ;jumlah counter
mov a,down
cjne a,#1,down_1
mov up,#0

mov a,count
add a,#1 ;jumlah saat ditempat asal
mov count,a ;jumlah counter
down_1:
mov counter,pulsa ;data terakhir
ret

mungah:
clr mot1
setb mot2
ret

mudun:
setb mot1
clr mot2
ret

jalan: mov r7,count ;counter
naik/turun
jal: jb sen2,$
mov r6,#0
djnz r6,$
jb sen2,jal
jb limit,jai
call lim
sjmp fio
jai: jnb sen2,$
djnz r6,$
jnb sen2,jai
jb limit,jail
call lim
sjmp fio
jail: djnz r7,jal ;ulangi jika belum 0
clr mot1
clr mot2
sjmp rent
fio: mov nul,#1
rent: ret

;=====MOTOR BALIK
blik: setb mot1
clr mot2
jnb sen1,$
clr mot1
clr mot2

;=====BUZER
bunyi: setb buzer
call delay_lama
clr buzer
ret

delay_lama:
mov r0,#10
tunda3:
mov r1,#100
tunda2:
mov r2,#250
tunda1:
djnz r2,tunda1
djnz r1,tunda2
djnz r0,tunda3
ret

```

```

;=====DISPLAY
konv:  mov r0,#36h
       mov r1,#33h
       mov dptr,#DATA_SEGMENT

```

```

tidak: mov a,@r0
       movc a,@a+dptr
       mov @r1,a
       inc r0
       inc r1
       cjne r1,#35h,tidak
       ret

```

;SUB ROUTIN DISPLAY

```

display: clr strobe ;strobe
         clr clock ;clock
         mov r1,#33h
         mov r6,#2;tampilan 2 digit desml

```

```

ulang:  mov r5,#8 ;ada 8 bit
         mov a,@r1

```

```

lagi:   rrc a
         jc  tinggi
         clr dat
         sjmp pah

```

```

tinggi: setb dat

```

```

pah:    setb clock
         call delay2
         clr clock
         djnz r5,lagi
         inc r1
         djnz r6,ulang
         setb strobe
         ret

```

```

delay2: mov r2,#10
ab1:    mov r3,#255
         djnz r3,$
         djnz r2,ab1
         ret

```

```

hep:    DB 'PENGATUR MISTAR'
         DB 'LOMPAT GALAH '

```

```

balik:  DB ' KE POSISI AWAL '
         DB ' '

```

```

pilih:  DB ' POSISI '
         DB ' NAIK TURUN '

```

```

naik:   db 'NAIK = '
         db 'CANCEL ENTER '

```

```

turun:  db 'TURUN = '
         db 'CANCEL ENTER '

```

```

hi_gh:  db ' NAIK..... '
         db ' '
         lo_w: db 'TURUN..... '
         db ' '

```

```

data_key: DB '0123456789*#'

```

```

blank:  DB ' '
         DB ' '

```

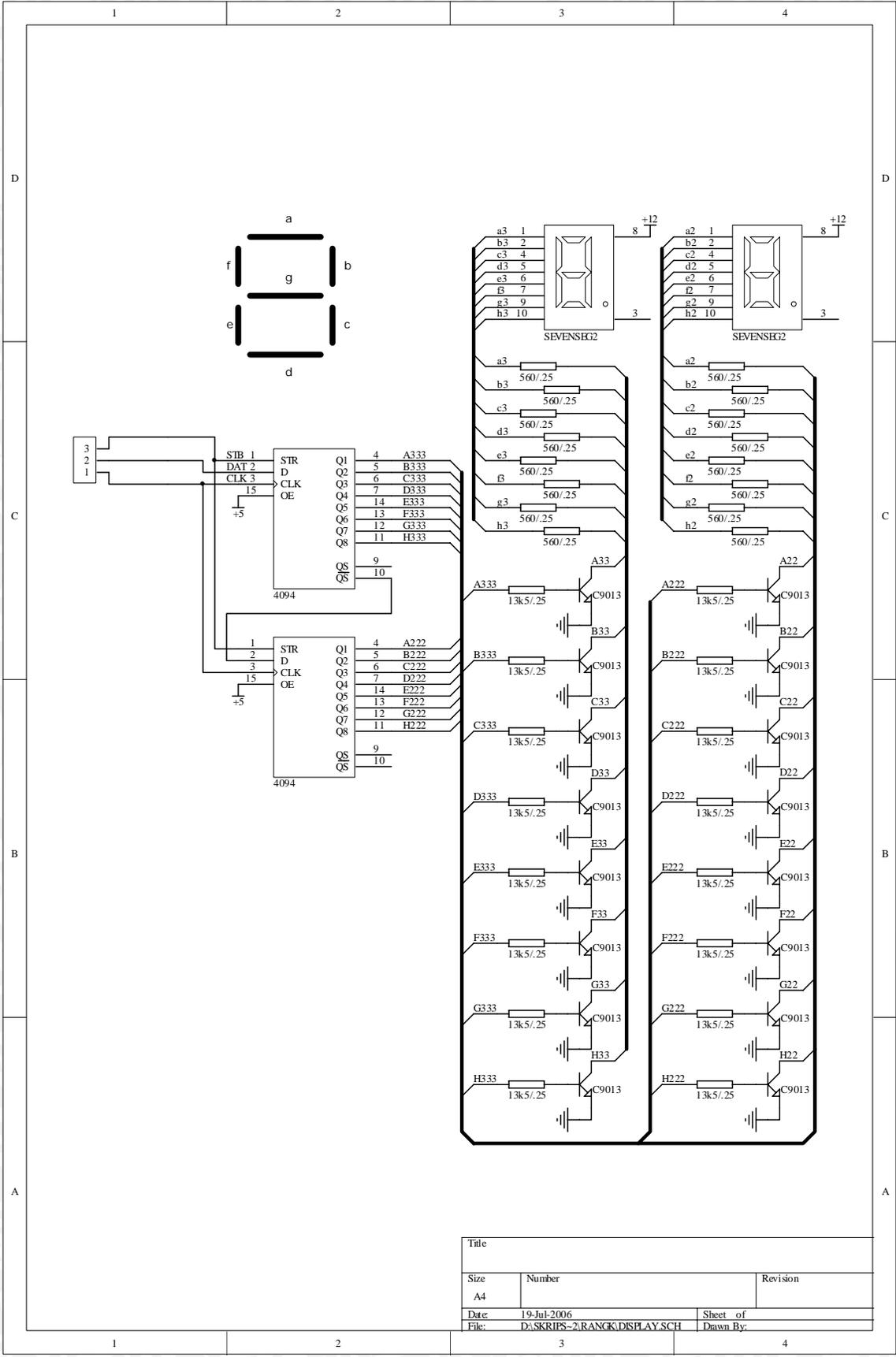
DATA\_SEGMENT:

```

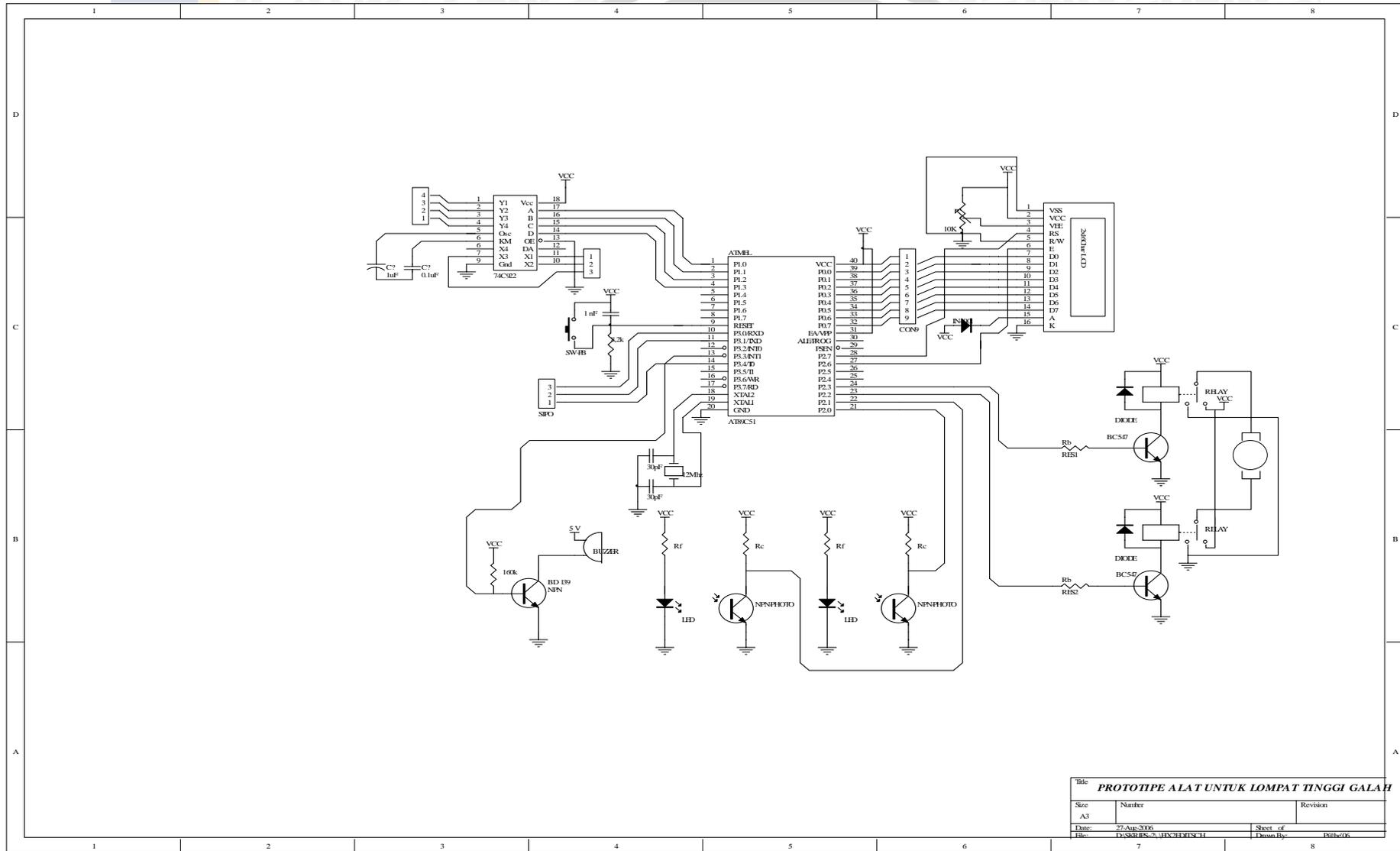
;a b c d e f g dp
db 0fch ;nilai desimal 0
db 60h ;nilai desimal 1
db 0dah ;nilai desimal 2
db 0f2h ;nilai desimal 3
db 66h ;nilai desimal 4
db 0b6h ;nilai desimal 5
db 0beh ;nilai desimal 6
db 0e0h ;nilai desimal 7
db 0feh ;nilai desimal 8
db 0f6h ;nilai desimal 9
db 00h ;clear
db 0ffh ;full
end

```





Title		
Size	Number	Revision
A4		
Date:	19-Jul-2006	Sheet of
File:	D:\SKRIPS-2\RANGK\DISPLAY.SCH	Drawn By:



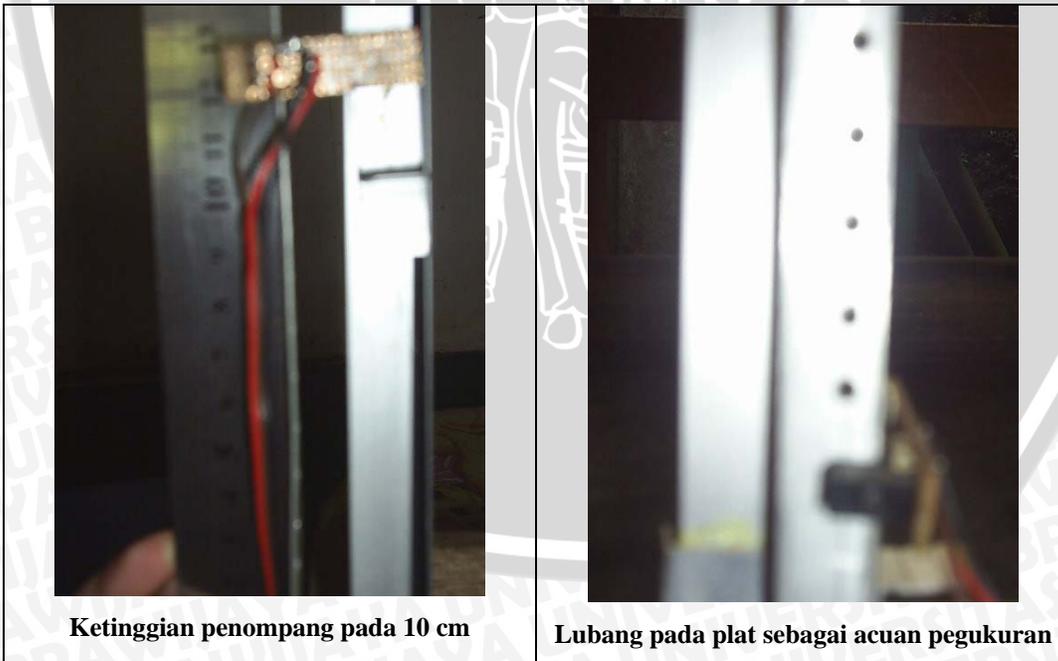
Title: <b>PROTOTYPE ALAT UNTUK LOMPAT TINGGI GALAH</b>			
Size: A3	Name:	Revision:	
Date: 27-08-2006	Sheet of:	Drawn By:	
File: D:\SKRIPSI-2\TFC\211025311	Sheet of:	Drawn By:	



# FOTO ALAT



Miniatur Lompat Galah



Ketinggian penompang pada 10 cm

Lubang pada plat sebagai acuan pengukuran



**Introduction:**

Prototype of jump pole ruler high for adjustment high ruler with display so we can manage ruler more efisien.

**Operating Voltage:** 220-240 volt AC.

**Color:** Black.

**Dimension:** see Fig. 4

**All Dimension in centimeter**

**Specification:**

**Mechanic:**

- Made from wood.
- Ruler made from plat.
- Motor for
- Dimension: see Fig. 1
  - High : 8.5 Cm
  - Width : 27 Cm
  - Length : 47 Cm

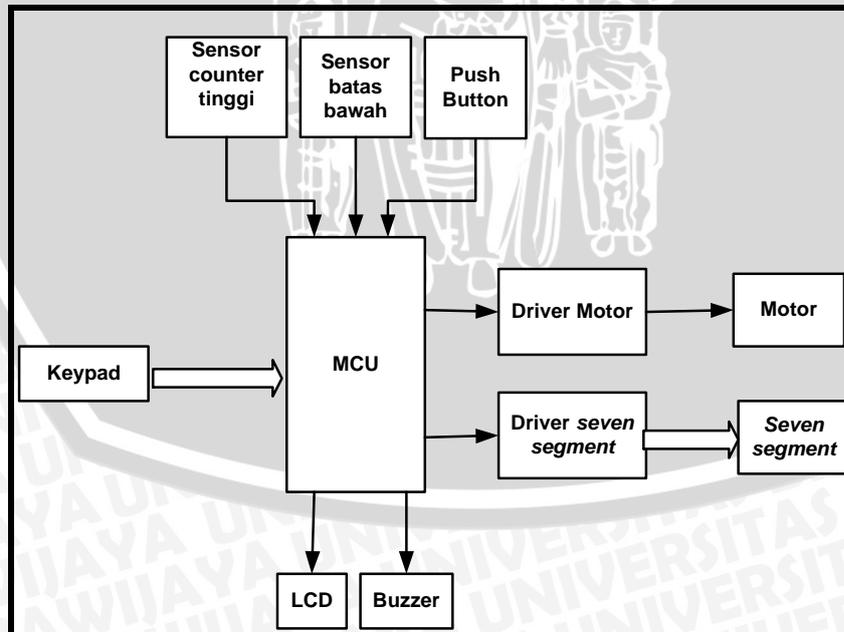
**Fig. 1: Mechanic Dimension**



**Electronic Component:**

- LCD 16x2 Character as menu program.
- 7-segment jumbo as display.
- Keypad 3X4 as input.
- 2 x DB9 Female Connector.

**Fig.2 : Internal Block Diagram**





### Operation Manual

1. Plug in DB 9 Connector Male to Female DB 9 Connector.
2. Plug in the power plug into AC 220V source.
3. Switch on the power.
4. At the first time you will see opening display on LCD.
5. After that you will see:

POSISI  
NAIK TURUN

6. Then you should press button (\*) for NAIK and press button (#) for TURUN.
7. If you press button (\*) you can see :

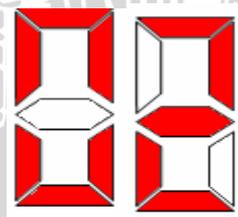
Naik.....Cm  
tekan \*

This menu allow to enter the data which you want from keypad.  
(Data contain 2 digit number).

And ruler automatic moves and you can see :

Naik.....

8. And display 7-segment will presenting data ruler. For example in fig 5



9. After that back to number 5
10. If you want to cancel pres button (#)

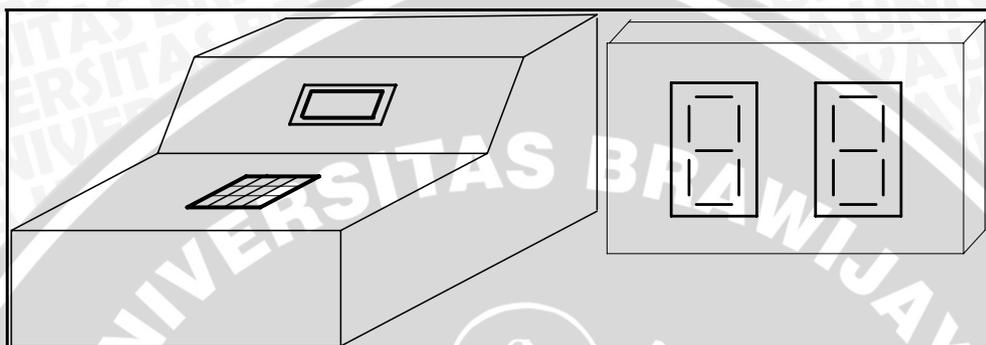




**Fig. 3: Physical Dimension (General View)**

**Physical Dimensions**

	Unit	Elektronic	Display
Weight	kg		
Length	cm	24	19
Width	cm	17	10
High	cm	7	17

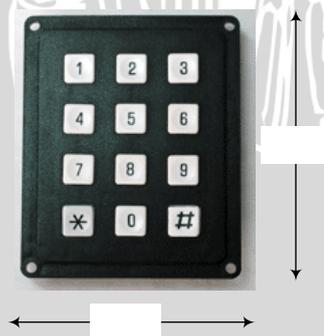


**Note:**

1. Physical Dimension of LCD.



2. Physical Dimension of Keypad



**Note:**

0-9 : Numeric Button.  
This picture not fit to scale

Design by: Eka Puspa  
Copyrights: pah\_eee.co.Inc.

[http://www.pah\\_eee@yahoo.com](http://www.pah_eee@yahoo.com)

