

PERANCANGAN PINTU OTOMATIS BERBASIS MIKROKONTROLER AT89C51

**Diajukan Untuk Memenuhi Syarat
Guna Mengikuti Ujian Sarjana
Untuk Mencapai Gelar Ke-Sarjanaan**

Disusun Oleh :

**Nama : YOGA SUGAMA
NIM : 01498-075**



**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS MERCU BUANA
JAKARTA
2006**

ABSTRAK

Teknologi berkembang dimulai dari yang bersifat manual menuju kearah yang bersifat otomatis. Teknologi yang bersifat manual akan membutuhkan tenaga kerja dan waktu yang jauh lebih besar dibanding dengan teknologi yang bersifat otomatis selain itu dapat mengurangi kesalahan manusia. Salah satu implementasi penggunaan teknologi untuk mempermudah pekerjaan manusia adalah pintu otomatis. Pintu otomatis adalah suatu alat yang berfungsi untuk membuka dan menutup pintu secara otomatis dan pengaturannya dikontrol dengan menggunakan suatu piranti mikrokontroler.

Pintu otomatis yang dirancang terdiri dari mikrokontroler AT89C51, jam digital, driver, motor stepper sensor infra merah, tombol set jam dan menit dan komponen miniatur pintu.

Pintu otomatis ini juga dilengkapi dengan jam digital sebagai penunjuk waktu dan sebagai acuan (referensi) kapan sensor mulai berfungsi dan berhenti berfungsi. Bila waktu yang ditunjukkan jam digital belum menunjukkan waktu yang telah diprogram sebelumnya, maka sensor belum aktif bekerja dan pintu tetap dalam keadaan menutup. Pada saat jam digital menunjukkan waktu referensi, maka sensor akan mulai aktif bekerja sambil menunggu adanya input dari luar (objek). Apabila ada objek yang menghalangi sensor, maka sensor akan memberikan sinyal masukkan ke mikrokontroler untuk memerintahkan motor stepper untuk berputar dan menggerakkan pintu dapat terbuka kemudian menutup kembali setelah selang sekian waktu (*delay time*).

Tombol darurat yang dipasang berfungsi sebagai alat pengaman yang apabila ditekan, maka pintu akan terbuka sekaligus menyalakan alarm (buzzer) sebagai tanda bahaya

ABSTRACT

Technology expand started from having the character of manual to toward having the character of automatically. Technology having the character of manual will require the much more compared to big time and labor technologically having the character of automatically besides can lessen the human being mistake. One of technological usage implementation to water down the work of human being is automatic door. Automatic door is functioning appliance to open and close the door automatically and its arrangement is controlled by using apparatus mikrokontroller.

Automatic door designed consisted of by the mikrokontroller AT89C51, digital watch, driver, motor stepper sensor infra squeeze the, knob set the hour and minute and component of door miniatur. This Automatic door is also provided with by digital watch as time indicator and as reference (referention) when sensor to start to function and desist to function. When time posed at digital watch not yet shown the program time which have previous, hence censor not yet active work and door remain to in a state of closing. At the (time) of digital watch show the reference time, hence censor will come into play to work at the same time await the existence of input from outside (object). If there is object hindering censor, hence censor will give the sinyal enter to mikrokontroller to command the motor stepper to rotate and move the door can be opened later then close to return after pipe of so much time (delay time).

Emergency knob attached function as a means of peacemaker which is if depressed, hence door will be opened at the same time fire the alarm (buzzer) as writing on the wall

KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah SWT, penulis panjatkan atas segala nikmat yang telah di berikan, terutama nikmat Islam, Iman dan Kesehatan. Shalawat serta salam semoga selalu tercurah untuk Nabi Muhammad SAW dan semoga keselamatan bagi para pengikutnya yang tetap setia dalam memegang panji Islam sampai hari akhir.

Penulis bersyukur, bahwa setelah berupaya keras, berdo'a dan bertawakal kepada Allah SWT serta atas bantuan dan dukungan dari semua pihak, akhirnya dapat menyelesaikan pembuatan dan penulisan tugas akhir ini dengan baik dan sesuai dengan waktunya. Pada kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Bapak **Jaja Kustija, Msc.**, selaku Dosen Pembimbing dan Kepala Laboratorium jurusan Teknik Elektro yang telah banyak memberikan masukan, bantuan dan dorongan yang besar dalam pembuatan dan penulisan tugas akhir ini.
2. Bapak **Ir. Budi Yanto Husoda, Msc.**, selaku Koordinator Tugas Akhir jurusan Teknik Elektro yang turut membantu sehingga selesainya pembuatan tugas akhir ini.
3. Bapak **Ir. Yudhi Gunardi MT**, selaku koordinator kerja praktek.
4. Semua dosen-dosen teknik elektro Universitas Mercu Buana
5. **Kedua Orang Tua** yang aku cintai dan selalu memberikan bantuan, dorongan dan do'a demi selesainya pembuatan dan penulisan tugas akhir ini.

6. **Achmad Paiz, ST**, yang telah banyak membantu dalam pembuatan tugas akhir ini serta terima kasih banyak masukan dan dukungannya.
7. **Nadia Apyanti**, terima kasih atas support dan sprit yang telah di berikan kepada penulis serta terima kasih atas jalan-jalannya.
8. **Teman-teman sejurusan** ; Banani khamsa '98, Ariyanto '98, Wahyu Wijayanto '98, Dony'99, Adi purnomo '98, Indra '98,Dudi '98, Fina '98, Robi '98, Toto'97, Suryadi'98, Adi chandra'98, Edwin'98, Sigit '98, Murgianto'98, setyo'98 dan seluruh angkatan 98 teknik elektro terima kasih atas dukungan dan bantuannya.
9. Muji dan Imam terima kasih atas waktunya dalam membantu ketikan tugas akhir saya.
10. Semua pihak yang telah membantu selesainya pembuatan dan penulisan tugas akhir ini yang tidak dapat saya sebutkan satu persatu.

Penulis menyadari bahwa dalam laporan tugas akhir ini mungkin masih terdapat kekurangan, baik itu berupa penyusunan maupun penulisannya. Untuk itu saran yang bersifat membangun diharapkan, sehingga penulisan laporan tugas akhir ini bisa menjadi lebih baik dan bermanfaat.

Akhir kata, penulis berharap semoga laporan tugas akhir ini dapat bermanfaat bagi semua pihak yang membutuhkan.

Jakarta, Februari 2005

Penulis

DAFTAR ISI

	Halaman
DAFTAR PERNYATAAN	
LEMBAR PENGESAHAN	
ABSTRAK	
KATA PENGANTAR	i
DAFTAR ISI.....	iii
DAFTAR TABEL.....	vii
DAFTAR GAMBAR	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Permasalahan	2
1.3 Batasan Masalah.....	3
1.4 Tujuan	3
1.5 Metode Penulisan	4
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI.....	6
2.1 Pengertian Mikrokontroler.....	6
2.1.1 Arsitektur Mikrokontroler AT89C51	7
2.1.2 Fungsi-fungsi Pin Mikrokontroler AT89C51	9
2.1.3 Organisasi Memori.....	10
2.1.3.1 Memori Program	10
2.1.3.2 Memori Data	10

2.1.4	Register	11
2.1.5	Struktur Port	14
2.1.5.1	Port 0	14
2.1.5.2	Port 1	14
2.1.5.3	Port 2	14
2.1.5.4	Port 3	15
2.1.6	Metode Pengalamatan	16
2.1.7	Software Mikrokontroler AT98C51.....	16
2.2	Motor Stepper	19
2.2.1	Prinsip Kerja Motor Stepper	19
2.2.2	Motor Stepper Unipolar	29
2.2.3	Karakteristik Motor Stopper	22
2.3	Display Seven Segment.....	23
2.4	Photodioda	25
2.5	Phototransistor	26
2.6	IC 74LS47 Decoder.....	27
2.7	IC 74LS14 Schmitt Triggers	29
2.8	IC ULN 2003	32
BAB III PERANCANGAN ALAT.....		34
3.1	Identifikasi Perancangan	34
3.2	Analisis Perancangan	34
3.3	Perancangan Hardware.....	35
3.3.1	Mikrokontroler AT89C51	39

3.3.2	Rangkaian Pembangkit Detak	39
3.3.3	Rangkaian untuk Membangkitkan Reset	39
3.3.4	Penampil Seven Segment	40
3.3.5	Perancangan Rangkaian Keyboard	41
3.3.6	Rangkaian Transduser	42
3.3.7	Rangkaian Penggerak/Driver Motor Stepper	43
3.3.8	Rangkaian Penggerak Relay Pada Buzzer	44
3.4	Perancangan Software	46
3.4.1	Program Inisialisasi	46
3.4.2	Program Utama	47
3.4.3	Subrutin Tombol	49
3.4.4	Subrutin Buka Pintu	50
3.4.5	Subrutin Buku2	51
3.4.6	Subrutin Scan	52
3.4.7	Subrutin Jam	54
3.4.8	Subrutin Menit	56
3.4.9	Subrutin Darurat.....	57
3.4.10	Subrutin Putar1	58
3.4.11	Subrutin Putar2	59
3.4.12	Subrutin Balikin	60
3.4.13	Subrutin SER.....	62
3.4.14	Subrutin Tunda.....	62
3.4.15	Subrutin Periode.....	63

3.4.16 Subrutin Periode0.....	63
3.4.17 Subrutin Inter0	63
3.4.18 Subrutin Timer1	65
3.4.19 Subrutin Interex0.....	66
BAB IV PENGUJIAN DAN PEMBAHASAN.....	68
4.1 Rangkaian Sensor.....	68
4.2 Rangkaian Pembangkit Detak	69
4.3 Rangkaian Reset.....	69
4.4 Pengujian Display	70
4.5 Pengujian Driver Motor Stepper	72
4.6 Pengujian Terhadap Relay Driver dan Buzzer.....	73
4.7 Pengujian Secara Keseluruhan dengan Perangkat Lunak ..	73
BAB V KESIMPULAN DAN SARAN.....	76
5.1 Kesimpulan	76
5.2 Saran.....	76

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

	Halaman
Tabel 2.1 Fungsi Pengganti Port 3	15
Tabel 2.2 Pola Kemudi Motor Stopper Unipolar	22
Tabel 2.3 Konversi Kode Biner pada Seven Segment	24
Tabel 2.4 Tabel Kebenaran Pada IC 74LS47	29
Tabel 2.5 Tabel Kebenaran dari Gerbang Logika NOT (74LS14.pdf)	31
Tabel 4.1 Tabel Hasil Pengujian Sensor	68
Tabel 4.2 Hasil Pengujian Display	71
Tabel 4.3 Hasil Pengujian Rangkaian Driver Motor Stepper.....	72
Tabel 4.4 Pengujian Keseluruhan.....	74

DAFTAR GAMBAR

	Halaman
Gambar 2.1	Arsitektur Internal Mikrokontroler AT89C51 8
Gambar 2.2	Konfigurasi Pin Mikrokontroler AT89C51..... 9
Gambar 2.3	Skema Motor Stepper..... 20
Gambar 2.4	Rangkaian Penggerak Motor Stopper 21
Gambar 2.5	Susunan Lilitan Motor Stopper Unipolar..... 21
Gambar 2.6	Display Seven Segment..... 23
Gambar 2.7	Led pada Seven Segment Common Anoda..... 24
Gambar 28	(a) Simbol Photodiode Pemancar, (b) Kurva Photodiode Pemancar 26
Gambar 2.9	Simbol Phototransistor 27
Gambar 2.10	IC 74LS47 Dekoder 28
Gambar 2.11	Pembentukan Gelombang Kotak Schmitt Trigger Inverter 30
Gambar 2.12	Simbol Gerbang Logika NOT dan Konfigurasi IC TTL 74LS14 30
Gambar 2.13	(a) Masukan Periodic pada Pemicu Schmitt Menghasilkan Keluaran Persegi, (b) Karakteristik Transfer Pembalik 31
Gambar 2.14	Simbol Darlington dan Konfigurasi IC ULN 2003..... 33
Gambar 3.1	Diagram Blok Pintu Otomatis..... 35
Gambar 3.2	Rangkaian Keseluruhan 38
Gambar 3.3	Rangkaian Pembangkit Detak/Clock 39

Gambar 3.4	Rangkaian Reset.....	40
Gambar 3.5	Rangkaian Penampil Seven Segment.....	41
Gambar 3.6	Rangkaian Pendeteksi Objek/Orang	42
Gambar 3.7	Rangkaian Driver Motor Stepper.....	44
Gambar 3.8	Rangkaian penggerak relay pada buzzer.....	45
Gambar 3.9	Flowchart Program Utama	48
Gambar 3.10	Flowchart Subrutin Tombol.....	50
Gambar 3.11	Flowchart Subrutin Buka Pintu.....	51
Gambar 3.12	Flowchart Subrutin Buka2	52
Gambar 3.13	Flowchart Subrutin Scan	53
Gambar 3.14	Flowchart Subrutin Jam	55
Gambar 3.15	Flowchart Subrutin Menit	56
Gambar 3.16	Flowchart Subrutin Darurat.....	57
Gambar 3.17	Flowchart Subrutin Putar1	58
Gambar 3.18	Flowchart Subrutin Putar2	59
Gambar 3.19	Flowchart Subrutin Balikin.....	60
Gambar 3.20	Flowchart Subrutin Ser	62
Gambar 3.21	Flowchart Subrutin Inter0	64
Gambar 3.22	Flowchart Subrutin Inter1	66
Gambar 3.23	Flowchart Subrutin Interex0	67
Gambar 4.1	Pengujian Display	71
Gambar 4.2	Rangkaian Pengujian Driver Motor Stepper.....	72
Gambar 4.3	Rangkaian Pengujian Relay Driver dan Buzzer.....	73

BAB I

PENDAHULAN

1.1 Latar Belakang

Kebutuhan manusia terhadap teknologi semakin meningkat sesuai kebutuhan serta kegiatan manusia yang semakin kompleks, yang menyebabkan manusia harus berfikir untuk membuat atau merancang suatu peralatan elektronik guna memudahkan manusia untuk membantu dalam aktifitas kesehariannya.

Teknologi berkembang dimulai dari yang bersifat manual menuju kearah yang bersifat otomatis. Teknologi yang bersifat manual akan membutuhkan tenaga kerja dan waktu yang jauh lebih besar dibanding dengan teknologi yang bersifat otomatis. Selain itu teknologi yang bersifat otomatis dapat menghindarkan/mengurangi human error. Sistem otomatis di industri memberikan banyak manfaat yang diperoleh yaitu: dapat meningkatkan produksi, kualitas produk, mengurangi biaya pekerja dan dapat melakukan pekerjaan yang tidak dapat dilakukan oleh manusia.

Salah satu implementasi penggunaan teknologi untuk mempermudah pekerjaan manusia pada tugas akhir ini yaitu pintu otomatis. Pintu otomatis yang akan dijelaskan menggunakan sensor infrared dan diprogram dengan menggunakan suatu perangkat mikrokontroller ATMEL 89C51.

Pintu otomatis ini merupakan pintu yang akan membuka dan menutup pintu secara otomatis. Sensor infrared akan bekerja bila terhalang suatu objek.

Bila sensor infrared terhalang maka akan menggerakkan motor stepper yang akan menyebabkan pintu terbuka secara otomatis. Pintu ini diatur waktu kuncinya dengan menggunakan jam digital yang akan ditampilkan dengan menggunakan seven segment. Waktu kuncinya berdasarkan waktu referensi yang telah di program sebelumnya pada mikrokontroller. Fungsi dari waktu referensi untuk meningkatkan keamanan didalam gedung karena pintu akan terbuka secara otomatis pada waktu/jam tertentu saja.

Pada pintu otomatis ini dipasang tombol buka pintu sehingga jika waktu buka pintu otomatis tidak aktif (waktu yang telah ditetapkan telah habis) maka pihak keamanan dapat menekan tombol buka pintu agar pintu terbuka dan pintu akan menutup kembali setelah beberapa detik. Pintu otomatis ini juga dipasang tombol darurat yang dapat berfungsi untuk membuka pintu dan sekaligus membunyikan buzzer (alarm) bila di gedung yang bersangkutan terjadi kebakaran atau bahaya lainnya.

1.2 . Identifikasi Permasalahan

Untuk mewujudkan tercapainya pembuatan alat yang dapat memecahkan masalah pintu otomatis yang mempunyai kunci otomatis yang berdasarkan waktu referensi yang akan ditampilkan oleh jam digital maka dapat diidentifikasi sebagai berikut:

1. Bagaimana cara agar pintu dapat membuka dan menutup pintu secara otomatis.
2. Bagaimana cara mengunci pintu secara otomatis berdasarkan waktu yang dilengkapi dengan jam digital.

3. Bagaimana cara membunyikan alarm sekaligus membuka pintu secara terus-menerus bila terjadi kebakaran ataupun bahaya lainnya.
4. Jenis Mikrokontroller dan komponen pendukung apa yang akan digunakan agar alat tersebut dapat bekerja sesuai dengan yang diharapkan.

1.3 Batasan Masalah

Perancangan pintu otomatis akan dibangun di sini dibatasi kepada penghitungan waktu agar pintu dapat membuka secara otomatis dan menutup setelah delay waktu yang di berikan telah habis. Pintu dapat membuka secara otomatis disini ditentukan oleh waktu yang diprogram. Waktu disini adalah sebagai kunci yang telah disimpan didalam memori mikrokontroller.

Untuk membuka pintu secara manual cukup dengan menekan tombol buka pintu. Jika terjadi kebakaran ataupun bahaya lainnya maka tombol darurat dapat ditekan sehingga pintu akan terbuka dan diikuti dengan bunyinya buzzer (alarm) sebagai tanda adanya bahaya. Tombol set jam dan menit digunakan untuk menyesuaikan jam dan menit dengan waktu yang lainnya.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah merancang dan membuat pintu otomatis dengan menggunakan suatu media sensor infrared, penampil dan pemrograman alat tersebut dengan menggunakan mikrokontroller AT89C51.

1.5 Metode Penulisan

Metode penulisan tugas akhir ini dilakukan dengan beberapa cara, yaitu:

a. Metode Kajian Pustaka

Yaitu dengan cara melakukan penulisan melalui referensi-referensi penunjang tema yang didapat dari perkuliahan dan dari buku-buku penunjang lainnya.

b. Metode Eksperimen

Yaitu dengan cara pengambilan data melalui pengujian serta pengukuran pada rancangan alat.

1.6 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini akan dibagi dalam lima bab, dengan masing-masing bab diuraikan sebagai berikut:

BAB I PENDAHULUAN

Dalam bab ini diuraikan latar belakang pemilihan judul, pembatasan masalah, metode penelitian, perancangan dan pengujian serta sistematika pembahasan dalam penulisan tugas akhir ini.

BAB II TEORI PENUNJANG

Bab ini menguraikan tentang teori penunjang dari alat yang akan dibahas dan juga menjelaskan tentang pengertian mikrokontroller, penampil seven segment, phototransistor dan infra merah, IC 74LS14 Schmitt Trigger, IC 74LS47, motor stepper serta IC ULN 2003.

BAB III ANALISA DAN PERANCANGAN

Bab ini berisi blok diagram, analisa blok diagram rangkaian, pembahasan cara kerja dan realisasi rangkaian.

BAB IV PENGUJIAN ALAT

Bab ini menguraikan langkah-langkah pengujian, pengamatan dari alat yang akan dibuat.

BAB V KESIMPULAN

Dalam bab ini penulis memaparkan tentang kesimpulan dan saran-saran dari perancangan.

BAB II

LANDASAN TEORI

2.1 Pengertian Mikrokontroler

Mikrokontroler adalah kemampuan digital yang menirukan fungsi otak manusia. Fungsi kemampuan otak meliputi : aritmatika (berhitung), logika, (mempertimbangkan suatu kondisi), dan memori.

Berbeda halnya dengan suatu mikroprosesor. Mikroprosesor adalah suatu pemrosesan yang terdiri dari *Central Processing Unit* (CPU) dan register-register, tanpa memori, tanpa I/O (input dan output), dan *peripheral* yang dibutuhkan oleh suatu sistem supaya dapat bekerja. Bila sebuah mikroprosesor dikombinasikan dengan I/O dan memori (RAM dan ROM) akan dihasilkan sebuah Mikrokontroler, dimana kombinasi dari komponen-komponen tersebut sudah terdapat di suatu chip *Integrated Circuit* (IC). Jadi apabila ingin merancang suatu alat dengan menggunakan Mikrokontroler maka tidak diperlukan menambah *clock generator, address latcher, chip selector, memory dan PPI (Programmable Peripheral Interface)*.

Variasi register di dalam mikroprosesor beragam bergantung pada tipe, fungsi khusus yang diinginkan dan pabrik pembuatannya. Karena Mikrokontroler telah dilengkapi dengan berbagai peripheral yang sudah terdapat pada satu chip IC maka Mikrokontroler memiliki keunggulan dalam hal :

- a. Harga yang lebih ekonomis .
- b. Tingkat keamanan yang lebih terjamin.

- c. Ukuran sistem yang jauh lebih ringkas.
- d. Kemudahan dalam merancang sistem yang berbasis Mikrokontroler.

Berdasarkan perbedaan dalam aplikasi, Mikrokontroler mempunyai beberapa set instruksi yang berbeda dengan mikroprosesor. Set instruksi mikroprosesor bersifat *processing intensive* untuk operasi data volume besar, yang beroperasi secara nibble, byte dan word. Mode pengalamatan memberikan akses ke data array yang besar, memakai pointer alamat dan offset.

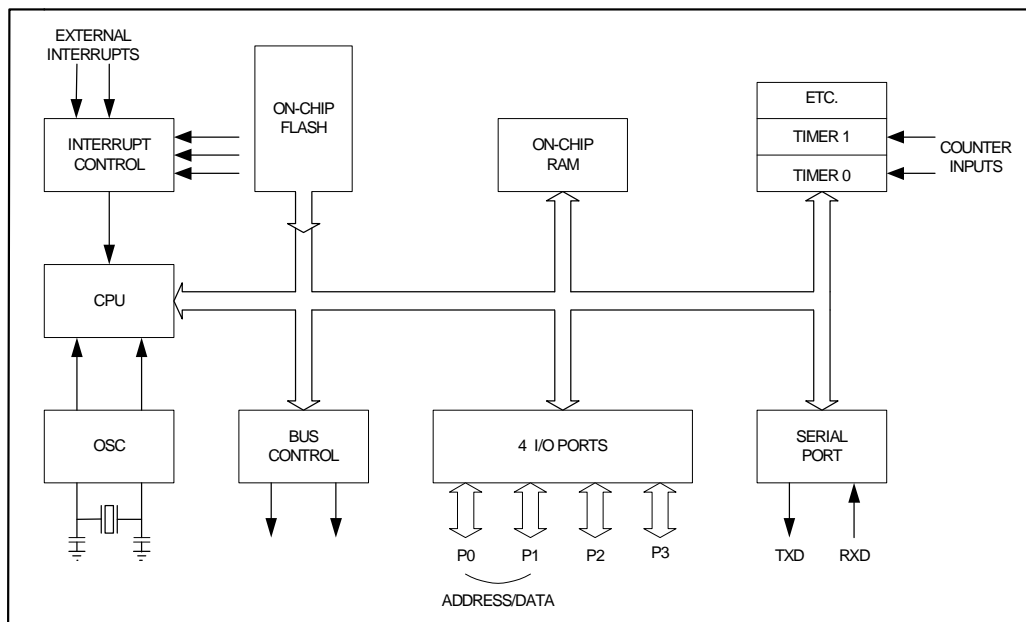
Di sisi lain Mikrokontroler mempunyai instruksi yang berkaitan dengan kontrol dari masukan dan keluaran. Antarmuka ke berbagai masukan dan keluaran memakai bit tunggal.

2.1.1 Arsitektur Mikrokontroler AT89C51

Mikrokontroler AT89C51 adalah sebuah mikrokomputer 8 bit dengan *low power supply* dan performansi tinggi yang terdiri dari CMOS dengan *Flash Programmable dan Erasable Read Only Memory (PEROM)* sebesar 4 Kbyte didalamnya. Alat tersebut dibuat dengan menggunakan teknologi tinggi *nonvolatile* berdensitas tinggi dari ATMEL yang kompatibel dengan keluarga MCS-51 yang merupakan standar industri. Dengan menggunakan *flash* memori, program dapat diisi dan dihapus secara elektrik, yaitu dengan memberikan kondisi-kondisi tertentu (*high / low*) pada pin-pinnya sesuai dengan konfigurasi untuk memprogram atau menghapus. Cara ini lebih praktis dibandingkan dengan menggunakan EPROM yang penghapusan program atau datanya menggunakan sinar ultraviolet.

Fasilitas yang tersedia pada AT89C51 adalah :

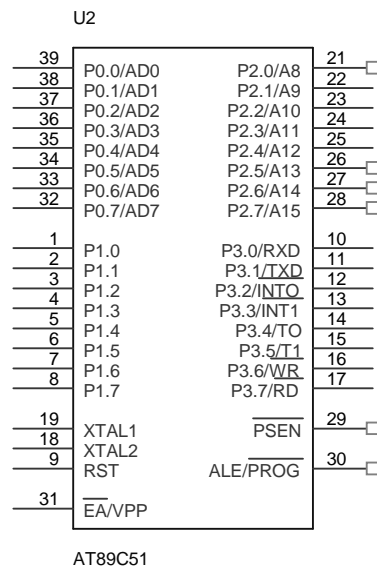
- a. 4 Kbytes ROM.
- b. 128 x 8 bit internal RAM.
- c. 32 jalur I / O.
- d. 2 (dua) 16 bit Timer / counter.
- e. 6 (enam) sumber Interupsi.
- f. Serial interface.
- g. 4 Kbyte *In-system Reprogrammable flash Memory* dengan kemampuan sampai 1000 kali pemrograman dan penghapusan.
- h. Kompatibel dengan MCS-51.
- i. Range operasi 0-24 MHz.



Gambar 2.1. Arsitektur Internal Mikrokontroler AT89C51

2.1.2 Fungsi – fungsi Pin Mikrokontroller AT89C51

Susunan kaki–kaki Mikrokontroller AT89C51 dapat dilihat pada gambar dibawah ini :



Gambar 2.2. Konfigurasi Pin Mikrokontroller AT89C51

RST, Reset input. Semua pin I/O akan high ketika input reset high. Memberikan sinyal high pada reset input ini selama 2 siklus akan me-reset Mikrokontroller.

XTAL1, adalah kaki masukan ke rangkaian oscilator internal. Sebuah oscilator kristal atau sumber oscilator luar dapat digunakan.

XTAL2, adalah kaki keluaran dari rangkaian oscilator internal. Pin ini dipakai bila menggunakan oscilator kristal.

ALE / $\overline{\text{PROG}}$, **ALE** (*Address Latch Enable*) digunakan untuk menahan alamat memori eksternal selama pelaksanaan instruksi. Pada operasi normal ALE dikeluarkan secara konstan pada 1/6 frekuensi oscilator dan dapat dipakai untuk timing atau clocking eksternal. Jika diinginkan, ALE dapat di-*disabled* dengan men-set operasi MOVX atau MOV. C.

\overline{PSEN} , *Program Strobe Enable* merupakan sinyal pengontrol yang memperbolehkan program memori eksternal masuk ke dalam bus selama proses pemberian / pengambilan instruksi (*fetching*)

$\overline{EA} / \overline{VPP}$, \overline{EA} harus dihubungkan ke ground untuk mem-*fetch* kode dari memori program eksternal mulai dari 0000H – FFFFH. Untuk eksekusi program internal, \overline{EA} harus dihubungkan ke Vcc.

Ruang memori program dan memori data pada Mikrokontroler AT89C51 memiliki ROM internal, jadi untuk menyimpan program selama tidak melebihi kapasitas ROM internal, tidak diperlukan lagi ROM eksternal. Dari 32 buah saluran I/O yang tersedia, 24 buah digunakan untuk I/O paralel dan untuk sinyal kontrol yaitu pin P3.0 sampai P3.7. Sinyal control tersebut adalah WR, RD, T0, T1, INT0, INT1 dan dua saluran masukan / keluaran serial yaitu RXD dan TXD.

2.1.3 Organisasi Memori

2.1.3.1 Memori Program

Mikrokontroler AT89C51 memiliki ruang operasi yang terpisah dengan memori data, ruang memori program dapat mencapai alamat hingga 4 kByte flash ROM (address 00h sampai 0FFFh).

2.1.3.2 Memori Data

Memori data pada Mikrokontroler AT89C51 memiliki ruang hingga 64 kByte, pengaksesan data ke memori data eksternal dilakukan dengan menggunakan data pointer melalui instruksi "*MOVX*".

Disamping memiliki RAM eksternal, Mikrokontroler AT89C51 memiliki RAM internal yang pengaksesannya terpisah dengan RAM eksternal. RAM internal berkapasitas 128 byte ditambah sejumlah SFR (*Special Function Register*), alamat terendah dari 128 byte internal dapat diakses dengan mode pengalamatan langsung, atau pengalamatan tidak langsung. Sedangkan SFR hanya dapat diakses secara pengalamatan langsung. 128 byte internal RAM AT89C51 terbagi atas 3 (tiga) bagian yaitu :

- a. 32 byte dari alamat 00h sampai 1Fh yang membentuk 32 register yang bekerja diatur sebagai 4 bank dari masing-masing 8 register. 4 register diberi nomor 0 sampai 3 dan membentuk 8 register R0 sampai R7. Masing-masing register dapat dialamatkan dengan nama ataupun dengan alamat ROM-nya. Byte RS0 dan RS1 pada PSW menentukan bank mana yang akan digunakan.
- b. Daerah pengalamatan bit dari 16 byte pada RAM dengan alamat 20h sampai 2Fh membentuk 128 pengalamatan bit. Pengalamatan bit dapat menggunakan alamat bit dari 00h sampai 7Fh atau 8bit dapat membentuk alamat byte dari 20h sampai 7Fh.
- c. Daerah penggunaan umum RAM terletak diatas daerah bit dari 30h sampai 7Fh pengalamatan sebagai byte.

2.1.4 Register

Di dalam setiap operasinya Mikrokontroler akan selalu menyertakan register dalam proses pengolahan datanya. Register adalah memori kecil berukuran 1 byte dimana 1 byte-nya bisa terdiri dari 8 bit, namun ada pula register

yang mampu menampung data 16 bit sekaligus sehingga 1 byte dari suatu register tertentu terdiri dari 16 bit data.

Register akan menampung data sebelum diolah, register juga akan menampung data hasil olahan sebelum kembali atau dikirim keluar jalur data *input/ output*. Register-register yang ada di Mikrokontroler adalah sebagai berikut:

- a. Register A, disebut juga sebagai akumulator (SFR = E0h) yaitu gerbang keluar / masuk data pada Mikrokontroler.
- b. Register B, disebut juga Base Register (SFE = F0h) register ini jarang dipakai karena hanya dipakai untuk operasi perkalian dan pembagian saja.
- c. Register R0 s/d R7. Register serbaguna boleh dipakai untuk apa saja. Khusus untuk register R0 dan R1 dapat digunakan untuk menyimpan data bagi pengalamatan tak langsung.
- d. Register DPTR, merupakan satu-satunya register 16 bit yang ada pada Mikrokontroler keluarga 8031. Register ini mempunyai fungsi serbaguna seperti halnya register R0 s/d R7, dan dapat juga untuk menyimpan data bagi pengalamatan tak langsung. Register ini dapat dibagi menjadi dua bagian yaitu register DPL dan register DPH. Register DPL adalah *nibble* bawah dari DPTR yaitu bit ke-0 s/d bit ke-7, sedangkan register DPH adalah *nibble* atas dari DPTR yaitu bit ke-8 s/d bit ke-15.
- e. PSW (*Program Status Word*), SFR = D0h, merupakan register yang berisi data-data kondisi Mikrokontroler setelah suatu operasi selesai dijalankan.

Register ini memiliki bit ke-3 dan bit ke-4 digunakan untuk pemilihan register bank.

- f. Register Port, terdiri dari 4 (empat) yaitu P0, P1, P2, dan P3. Register port berisi data-data yang akan dikirim keluar dan data-data yang telah dibaca dari luar.
- g. SP (*Stack Pointer*), SFR =81h, berisi alamat *stack* tertinggi dari alamat *stack* didalam internal RAM. Register ini berguna untuk menyimpan alamat-alamat data yang ditandai pada saat terjadi pemanggilan subprogram dan interupsi program.
- h. Register pewaktu, digunakan untuk menyimpan data offset yang akan digunakan untuk membatasi perhitungan waktu. Pewaktu yang ada di AT89C51 adalah pewaktu 16 bit sebanyak 2 (dua) buah, maka terdapat 2 buah register pewaktu yang dibentuk dari pasangan-pasangan register 8 bit yaitu TH0 dan TL0 dengan TH1 dan TL1. Timer 0 dan timer 1 keduanya dapat beroperasi sebagai timer atau counter. Pada fungsi “timer”, isi register ditambah satu setiap siklus mesin. Jadi, seperti menghitung siklus mesin. Karena satu siklus mesin terdiri dari 12 periode oscilator, maka kecepatannya = $1/12$ frekuensi oscilator. Pada fungsi “counter”, isi register ditambah satu setiap transisi 1 ke 0 pada input eksternal yang bersesuaian dengan T0 atau T1. Untuk mengenali transisi 1 ke 0 dibutuhkan 2 siklus mesin (24 periode), maka input maksimum ialah $1/24$ frekuensi oscillator.

- i. Register-register kontrol, terdiri atas register-register IP (*Interrupt Priority*), IE (*Interrupt Enable*), TMOD (*Timer Mode*), TCON (*Timer Control*), dan PCON (*Paralel Control*).

2.1.5 Struktur Port

2.1.5.1 Port 0

Port 0 (nol) terdapat pada pin 32 s/d pin 39. Port 0 adalah port paralel 8 bit dua arah yang belum dilengkapi dengan rangkaian pull-up. Output dari port 0 dapat men-sink 8 buah input TTL. Pada saat pin-pin port 0 berada pada kondisi high, maka port 0 dapat digunakan sebagai input. Port 0 dapat digunakan sebagai output jika keluaran pada port 0 diberi resistor pull-up, karena jika tidak diberi resistor pull-up maka keluarannya tidak akan menjadi logika 1.

2.1.5.2 Port 1

Port 1 terdapat pada pin 1 s/d pin 8. Port 1 adalah port paralel 8 bit dua arah yang telah dilengkapi dengan rangkaian *pull-up*. Output dari port 1 dapat memberikan input high/low kepada 4 buah input TTL. Pada saat port 1 berada pada kondisi *high* karena internal *pull-up*, maka port 1 dapat digunakan sebagai input.

2.1.5.3 Port 2

Port 2 terdapat pada pin 21 s/d pin 28. Port 2 adalah paralel dua arah yang dilengkapi dengan rangkaian *pull-up*. Output dari port 2 dapat memberikan input *high / low* kepada 4 buah input TTL. Pada saat pin-pin port 2 berada pada kondisi *high*, karena internal pull-up maka port 2 dapat digunakan sebagai *input*.

Port 2 memberikan *high order address byte* selama mengakses program diluar dan selama mengakses ke data memori diluar yang menggunakan address 16 bit. Untuk keadaan ini port 2 menggunakan *internal pull-up* yang kuat.

2.1.5.4 Port 3

Port 3 terdapat pada pin 10 s/d 17. Port 3 adalah port paralel dua arah yang dilengkapi dengan rangkaian *pull-up*. Output dari port 3 dapat memberikan input *high / low* kepada 4 buah TTL. Pada saat pin-pin 3 berada pada kondisi *high* karena internal pull – up, maka port 3 dapat digunakan sebagai *input*. Selain itu port 3 juga memiliki fungsi pengganti yang dapat dilihat pada table dibawah ini :

Tabel 2.1. Fungsi Pengganti Port 3

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

2.1.6 Metode Pengalamatan

Metode pengalamatan pada Mikrokontroller AT89C51 adalah sebagai berikut :

- a. Pengalamatan Langsung (*direct addressing*)

Pada pengalamatan langsung, didalam instruksinya operand ditunjukkan oleh suatu alamat yang lebarnya 8 bit, dan hanya bisa digunakan untuk data internal RAM dan SFR.

b. Pengalamatan tidak langsung (*indirect addressing*)

Pada pengalamatan tak langsung, instruksi menunjukkan suatu register yang isinya adalah alamat dari operand .

c. Pengalamatan berindeks (*indexed addressing*)

Yang dapat diakses dengan pengalamatan berindeks hanya memori program. Metode ini dimaksudkan untuk membaca *look-up table* di memori program.

d. Pengalamatan segera (*immediate addressing*)

Dalam memori program suatu opcode dapat langsung diikuti oleh suatu konstanta.

2.1.7 Software Mikrokontroller AT89C51

Instruction set Mikrokontroller AT89C51 dikelompokkan kedalam 5 kelompok yaitu instruksi aritmatika, instruksi logika, instruksi transfer data, instruksi manipulasi bit, dan instruksi percabangan.

a. Instruksi Aritmatika

Instruksi aritmatika digunakan untuk keperluan operasi aritmatika yaitu penjumlahan, pengurangan, pembagian, dan perkalian. berikut ini adalah instruksi-instruksi aritmatika:

- | | |
|----------------|-----------------|
| • ADD A,Rn | • ADDC A,Rn |
| • ADD A,direct | • ADDC A,direct |
| • ADD A,@Rn | • ADDC A,@Rn |
| • ADD A,#data | • ADDC A,#data |

- SUBB A,Rn
- SUBB A,direct
- SUBB A,@Rn
- SUBB A,#data
- ADD A,Rn
- ADD A,direct
- ADD A,@Rn
- ADD A,#data
- ADDC A,Rn
- ADDC A,direct
- ADDC A,@Rn
- ADDC A,#data
- SUBB A,Rn
- SUBB A,direct
- SUBB A,@Rn
- SUBB A,#data

b. Instruksi logika

Instruksi logika digunakan untuk operasi logika AND, OR, X-OR, penghapusan, komplemen, dan rotasi atau pergeseran. Instruksi yang termasuk dalam kelompok ini adalah:

- ANL A,Rn
- ANL A,direct
- ANL A,@Ri
- ANL A,#data
- ANL direct,A
- ANL direct,#data
- ORL A,Rn
- ORL A,direct
- ORL A,@Ri
- ORL A,#data
- ORL direct,A
- ORL direct,#data
- XRL A,Rn
- XRL A,direct
- XRL A,@Ri
- XRL A,#data
- XRL direct,A
- XRL direct,#data
- CLR A
- CPL A
- RL A
- RLC A
- RR A
- RRC A
- SWAP A

c. Instruksi Transfer Data

Instruksi transfer data digunakan untuk pemindahan data. Instruksi yang termasuk dalam kelompok ini adalah:

- MOV Rn,A
- MOV Rn,direct
- MOV Rn,#data
- MOV direct,A
- MOV direct,Rn
- MOV direct,direct
- MOV direct,@Rn
- MOV direct,#data
- MOV @Ri,A
- MOV @Ri,direct
- MOVX A,@Ri
- MOVX A,@DPTR
- MOVX @Ri,A
- MOVX @DPTR,A
- PUSH direct
- POP direct

- XCH A,Rn
- XCH A,direct
- XCH A,@Ri
- XCHD A,@Ri

d. Instruksi Manipulasi Bit

Instruksi yang termasuk dalam kelompok manipulasi bit ini adalah:

- CLR C
- CLR bit
- SETB C
- SETB bit
- CPL C
- CPL bit
- ANL C,bit
- ANL C,/bit
- ORL C,bit
- ORL C,/bit
- MOV C,bit
- MOV bit,C
- JC REL
- JNC rel
- JB bit,rel
- JNB bit,rel
- JBC bit,rel

e. Instruksi Percabangan

Instruksi percabangan digunakan untuk melakukan percabangan. Instruksi

yang termasuk dalam kelompok ini adalah:

- ACALL addr11
- LCALL addr16
- RET
- RETI
- AJMP addr11
- LJMP ret
- SJMP rel
- JMP @A+DPTR
- JZ rel
- JNZ rel
- CJNE a,direct,rel
- CJNE A, #data,rel
- CJNE Rn,#data,rel
- CJNE @RI,#data,rel
- DJNZ Rn,REL
- DJNZ direct,REL
- NOP

2.2 Motor Stepper

Motor stepper adalah peralatan elektronika yang berputar untuk merubah masukan pulsa digital menjadi gerakan analog secara bertahap (step). Dengan kata lain prinsip kerja motor stepper, yaitu adanya gaya medan elektro magnet yang menggerakkan rotor karena adanya arus yang mengalir melalui lilitan stator, dimana setiap gelombang berbentuk pulsa akan menggerakkan motor. Pemakaian motor stepper digunakan khususnya pada peralatan digital seperti printer, plotter x-y, robotika, disk drive dan lain sebagainya. Keuntungan dari pemakaian motor stepper adalah,

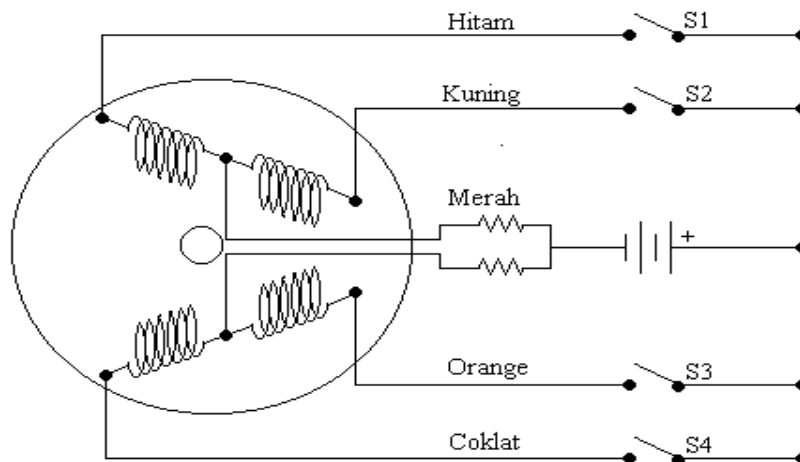
1. Mudah dihubungkan dengan peralatan digital, karena input dan outputnya diskrit.
2. Resolusi putarannya beragam.

2.2.1 Prinsip Kerja Motor Stepper

Motor stepper bekerja berdasarkan pulsa-pulsa arus yang diberikan dalam urutan yang tepat kepada stator. Selain itu, pulsa-pulsa arus tersebut harus cukup kuat untuk menggerakkan kumparan-kumparan stator motor stepper. Karena arus yang diperlukan motor cukup besar, maka diperlukan rangkaian penggerak untuk menggerakkan motor stepper.

Motor tersebut terdiri dari 5 buah kabel untuk dihubungkan dengan sumber tegangan dan pulsa penggerak motor. Bentuk skematik motor dan hubungan kabel secara umum dapat dilihat pada gambar 2.5. Tiap saklar yang tertutup akan memberikan arus ke kumparan dan mengakibatkan stator

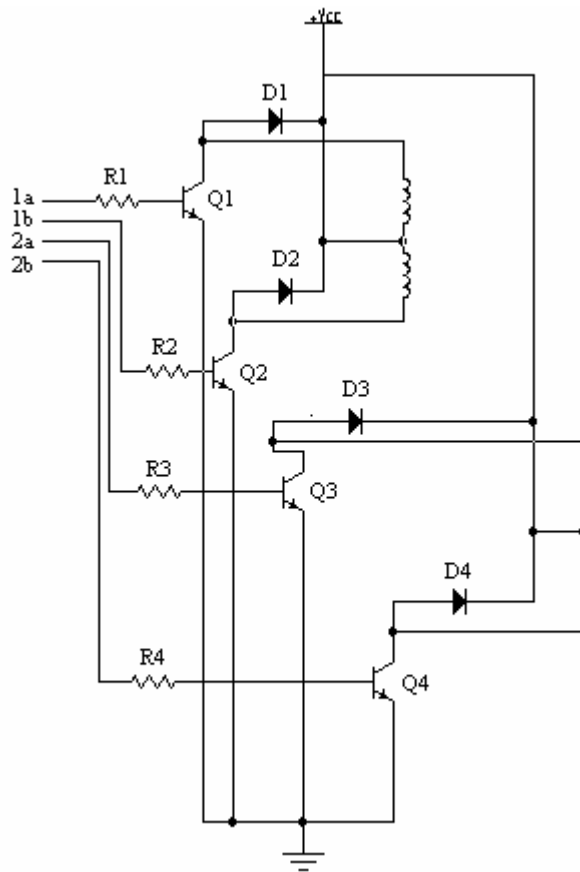
termagnetasi menjadi kutub utara atau selatan (tergantung dari arus yang dilewatkan). Sedangkan pada saat saklar terbuka stator menjadi netral. Pembukaan dan penutupan saklar-saklar akan mengakibatkan motor berputar searah atau berlawanan arah jarum jam.



Gambar 2.3 Skema Motor Stepper

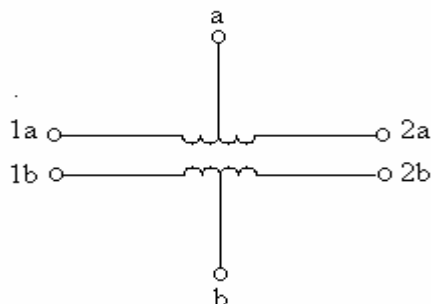
2.2.2 Motor Stepper Unipolar

Motor unipolar relatif mudah untuk dioperasikan. Rangkaian penghitung 1-dari-n yang sederhana dapat menghasilkan urutan langkah yang tepat dan penggerak yang paling sederhana 1 transistor untuk tiap lilitan adalah memungkinkan untuk motor unipolar. Motor ini memiliki karakteristik khusus dengan lilitan tengah-tengah. Skema kawat dasarnya adalah menyatukan titik tengah dan memberikan +MV (motor voltage). Rangkaian penggerak akan membumikan tiap lilitan untuk memberikan energi.



Gambar 2.4 Rangkaian Penggerak Motor Stepper

Motor stepper unipolar dikenal dengan lilitan titik tengahnya. Jumlah fasa adalah dua kali jumlah lilitan, karena tiap koil dibagi dua. Gambar dibawah ini memiliki dua lilitan titik tengah, menunjukkan hubungan dari motor stepper unipolar 4-fasa.



Gambar 2.5 Susunan Lilitan Motor Stepper Unipolar

Tabel 2.2. Pola Kemudi Motor Stepper Unipolar

Step	1a	1b	2a	2b
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

2.2.3 Karakteristik Motor Stepper

Karakteristik motor stepper ditentukan oleh beberapa unsur, yaitu :

a. Tegangan

Motor stepper biasanya memiliki jangkauan tegangan. Hal ini tercetak langsung pada motor stepper dan ditentukan pada lembar data dari motor tersebut. Melampaui dari jangkauan tegangan bisa diperlukan untuk mendapatkan torsi yang diinginkan dari motor, tetapi dengan melakukan hal ini akan mengakibatkan panas berlebihan atau memperpendek umur pemakaian motor.

b. Resistansi

Resistansi per-lilitan adalah karakteristik dari motor stepper. Resistansi ini akan menentukan tarikan arus dari motor, yang berhubungan langsung pada kurva torsi motor dan kecepatan maksimum pada putaran motor.

c. Derajat Per-Langkah

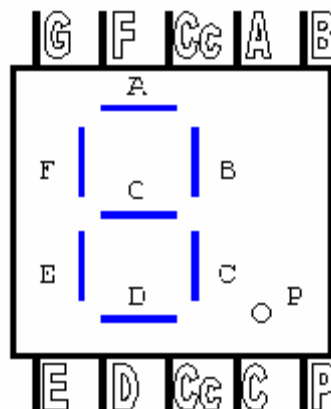
Derajat perlangkah merupakan faktor yang paling penting dalam menentukan motor stepper yang akan digunakan pada suatu aplikasi. Faktor ini menentukan besar derajat dari putaran motor untuk setiap langkah. Untuk

operasi setengah langkah akan menggandakan jumlah langkah/putaran dan membagi derajat per-langkah meliputi : 0.72, 1.8, 3.6, 7.5, 1.5 dan 90°. Derajat per-langkah sering juga disebut resolusi dari motor. Seperti pada kasus motor tak bertanda, jika motor hanya memiliki satuan angka langkah/putaran yang tercetak, membagi 360 dengan angka tersebut akan menghasilkan nilai derajat perlangkah.

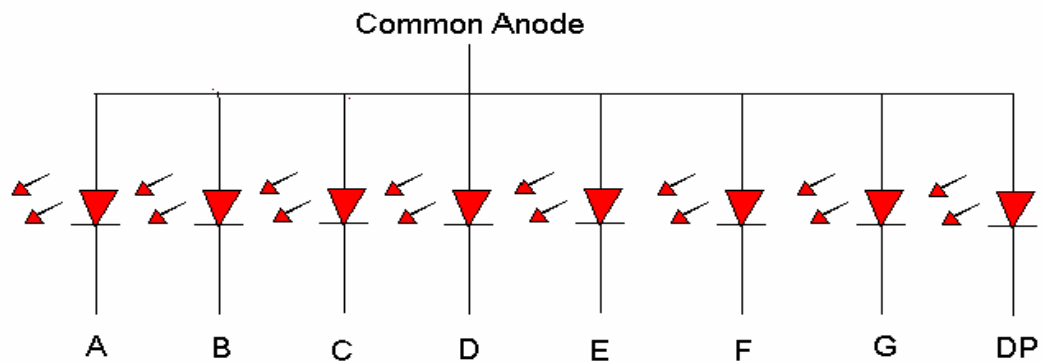
2.3 Display Seven Segment

Seven segment merupakan suatu rangkaian yang terdiri dari beberapa led yang disusun sedemikian rupa sehingga dapat membentuk angka atau karakter yang diinginkan dengan mengaktifkan led-led tertentu.

Led-led itu dihubungkan ke satu titik yang disebut common. Pada display seven segment terdiri dari dua common yaitu common anoda dan common katoda. Pada common katoda, kaki common dihubungkan ke 0 volt dan untuk mengaktifkan suatu led harus diberi logika tinggi (high), sedangkan common anoda, kaki common dihubungkan ke 5 volt dan untuk mengaktifkannya harus diberi logika nol (low).



Gambar 2.6. Display Seven Segment



Gambar 2.7. Led pada Seven Segment common anoda

Seven segment terdiri dari 8 buah dioda LED yang disusun sedemikian rupa sehingga dapat menampilkan yang dikehendaki. 7 buah led digunakan untuk menampilkan bilangan dan 1 buah untuk menampilkan tanda dot atau titik. Tegangan maju LED biasanya sebesar 1,6 - 2 Volt dan arus nya sebesar 10-20 mA. Dalam aplikasinya diperlukan transistor sebagai saklar on dan off.

Agar kita dapat menampilkan bilangan yang kita inginkan maka kita harus mengetahui table dari display seven segment itu sendiri, karena antara seven segment common anoda dan katoda mempunyai tabel kebenaran yang berbeda. Untuk common anoda table kebenarannya dapat dilihat pada table dibawah ini:

Tabel 2.3. Konversi Kode Biner pada Seven Segment

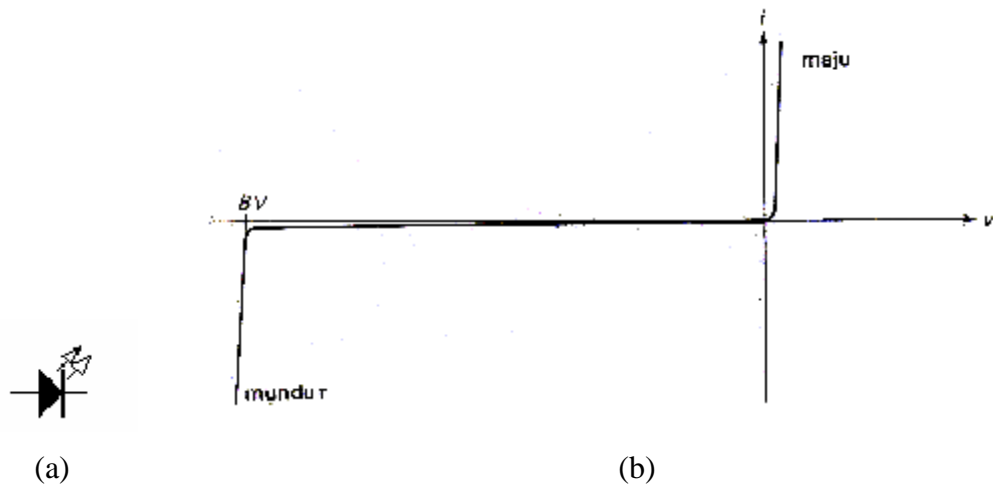
Desimal	Data Led yang Aktif						
	a	b	c	d	e	f	G
0	0	0	0	0	0	0	1
1	1	0	0	1	1	1	1
2	0	0	1	0	0	1	0
3	0	0	0	0	1	1	0
4	1	0	0	1	1	0	0
5	0	1	0	0	1	0	0
6	1	1	0	0	0	0	0
7	0	0	0	1	1	1	1
8	0	0	0	0	0	0	0
9	0	0	0	1	1	0	0

2.4 Photodioda

Photodioda merupakan peralatan semikonduktor yang dapat mengubah energi cahaya menjadi gerakan elektron suatu arus listrik. P-N junction di disain sehingga cahaya dengan mudah jatuh di daerah sensitive atau daerah radiant. Ketika cahaya mengenai pn junction yang di reverse bias, sepasang muatan carrier (hole dan elektron) dihasilkan. Arus yang dihasilkan disebut photocurrent. Photocurrent sebanding dengan intensitas cahaya yang mengenai permukaan sensitif cahaya. photodioda silicon dapat digunakan sebagai detector cahaya ketika di reverse bias, juga dapat digunakan sebagai sel photovoltaic. Sebagai photovoltaic, photodioda memiliki tegangan open circuit yang tinggi. Karena karakteristiknya itu photodioda sering digunakan pada pengukur cahaya kamera.

Junction ini sering dikemas dalam bungkus plastik dengan lensa tipis bundar yang terfokus ke daerah sensitif cahaya. Photodioda di disain untuk menerima cahaya yang tampak ataupun tidak tampak.

Beberapa karakteristik dari photodioda silicon antara lain: daya bocor max (250 mW), tegangan reverse (7 Vdc), sensitivitas cahaya hingga panjang gelombang 550 nm (dalam jangkauan cahaya tampak), dan tegangan open sirkuit hingga 390 mV. Symbol dan kurva photodioda pemancar cahaya yang menunjukkan bias maju pada tegangan dan arus positif dan bias mundur pada tegangan dan arus negatif dapat di lihat pada gambar 2.8.



Gambar 2.8. (a) Simbol photodiode pemancar. (b) Kurva photodiode pemancar

2.5 Phototransistor

Phototransistor adalah transistor dengan arus collector (I_c) berubah bila hubungan kolektor basis disinari. Phototransistor akan merubah energi cahaya, dalam hal ini energi cahaya infra merah, menjadi pulsa-pulsa sinyal listrik. Phototransistor harus mampu mengumpulkan sinyal infra merah sebanyak mungkin sehingga pulsa-pulsa sinyal listrik yang dihasilkan kualitasnya cukup baik. Semakin besar intensitas infra merah yang diterima maka sinyal pulsa listrik yang dihasilkan akan baik, jika sinyal infra merah yang diterima intensitasnya lemah maka infra merah tersebut harus mempunyai pengumpul cahaya (*light collector*) yang cukup baik dan sinyal pulsa yang dihasilkan oleh sensor infra merah ini harus dikuatkan.

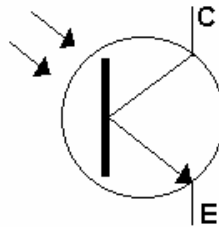
Jika phototransistor di gunakan saklar ketika basis terkena cahaya infer merah maka V_{ce} akan mendekati 0 volt.

$$V_{CE} \cong 0$$

Dan jika tidak terkena cahaya maka tegangan pada V_{ce} akan mendekati tegangan V_{cc} .

$$V_{CE} \cong V_{CC}$$

Gambar 2.8 dibawah merupakan simbol dari Phototransistor. Phototransistor mempunyai collector, emitter dan basis yang disinari oleh cahaya



Gambar 2.9. Simbol phototransistor

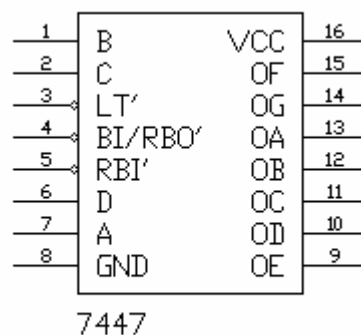
Infra merah mempunyai panjang gelombang elektromagnetik yang dapat diterima oleh *Photodetektor* (photodiode dan phototransistor) sebesar 10^{-3} - 10^{-6} m dan frekuensi 10^{11} - 10^{14} Hz. Infra merah dibuat dari LED (*Light Emitting Diode*) yang bahan semikonduktornya adalah Ga dan As (*Gallium arsenate*).

2.6 IC 74LS47 Decoder

Sering timbul kebutuhan untuk mengalihkan suatu sistem bilangan tertentu ke sistem bilangan dari tipe yang lain. Misalnya pada penggunaan kalkulator, bilangan-bilangan pada tombol tekan harus dialihkan ke BCD (Binary Coded Decimal) atau biner. Jika bilangan-bilangan hendak diperagakan perlu dilakukan pengalihan dari BCD ke kode tujuh segmen. Pengalih yang mengubah kode seperti misalnya BCD atau biner ke suatu kode yang mudah terbaca seperti

desimal disebut dekoder, sedangkan rangkaian yang mengalihkan kekode mesin umumnya disebut Enkoder. Dekoder ini mempunyai empat saluran masukan dan tujuh keluaran. Sinyal keluaran 1 dari dekoder tidak mengaktifkan LED sedang sinyal keluaran 0 mengaktifkan LED.

Dekoder mengubah kode biner menjadi kode khusus yang menyalakan ruas yang tepat pada peraga 7 segmen yang merupakan peralatan yang biasa digunakan untuk memperagakan bilangan desimal. Dekoder menerjemahkan dari bahasa mesin ke bahasa manusia. Untuk mengaktifkan segmen-segmen LED yang sesuai pada peraga ini diperlukan suatu tegangan GND (rendah). Masukan pendekode berupa bilangan 4 bit (masukan A,B,C dan D). Bilangan BCD di-kodekan sehingga membentuk kode 7 segmen yang akan menyalakan ruas-ruas yang sesuai pada peraga LED. Masukan BCD yang tak berlaku [desimal 10,11,12,13,14,dan 15], meskipun masukan ini bukan merupakan bilangan BCD namun masukan-masukan tersebut menghasilkan suatu keluaran yang unik.



Gambar 2.10. IC 74LS47 Dekoder

- Pin 1,2,6,7 [B,C,D,A] : Masukan BCD
- Pin 5 : RBI (Pengosongan)
- Pin 4 : BI/RBO (Pengosongan)

- Pin 3 : LT (Lamp Test / Lampu uji)
- Pin 8 : Ground
- Pin 9,10,11,12,13,14,15 : [e,d,c,b,a,g,f] Output kode 7 segmen
- Pin 16 : Vcc

Tabel 2.4. Tabel kebenaran pada IC 74LS47

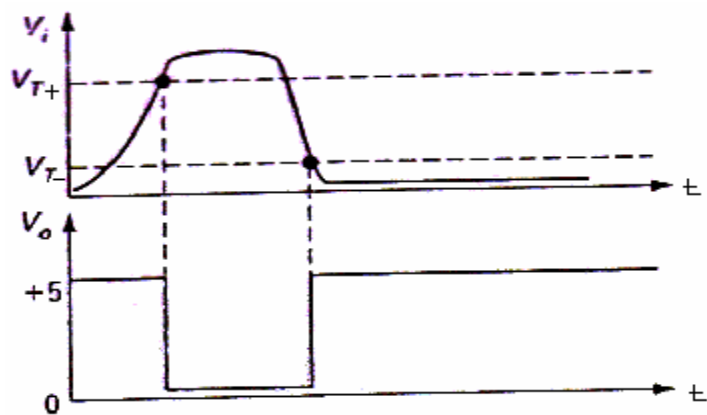
TRUTH TABLE

DECIMAL OR FUNCTION	INPUTS						OUTPUTS							NOTE	
	$\overline{\text{LT}}$	$\overline{\text{RBI}}$	D	C	B	A	$\overline{\text{BURBO}}$	a	b	c	d	e	f		g
0	H	H	L	L	L	L	H	L	L	L	L	L	L	H	A
1	H	X	L	L	L	H	H	H	L	L	H	H	H	H	A
2	H	X	L	L	H	L	H	L	L	H	L	L	H	L	
3	H	X	L	L	H	H	H	L	L	L	L	H	H	L	
4	H	X	L	H	L	L	H	H	L	L	H	H	L	L	
5	H	X	L	H	L	H	H	L	H	L	L	H	L	L	
6	H	X	L	H	H	L	H	H	H	L	L	L	L	L	
7	H	X	L	H	H	H	H	L	L	L	H	H	H	H	
8	H	X	H	L	L	L	H	L	L	L	L	L	L	L	
9	H	X	H	L	L	H	H	L	L	L	H	H	L	L	
10	H	X	H	L	H	L	H	H	H	H	L	L	H	L	
11	H	X	H	L	H	H	H	H	H	L	L	H	H	L	
12	H	X	H	H	L	L	H	H	L	H	H	H	L	L	
13	H	X	H	H	L	H	H	L	H	H	L	H	L	L	
14	H	X	H	H	H	L	H	H	H	H	L	L	L	L	
15	H	X	H	H	H	H	H	H	H	H	H	H	H	H	
BI	X	X	X	X	X	X	L	H	H	H	H	H	H	H	B
RBI	H	L	L	L	L	L	L	H	H	H	H	H	H	H	C
LT	L	X	X	X	X	X	H	L	L	L	L	L	L	L	D

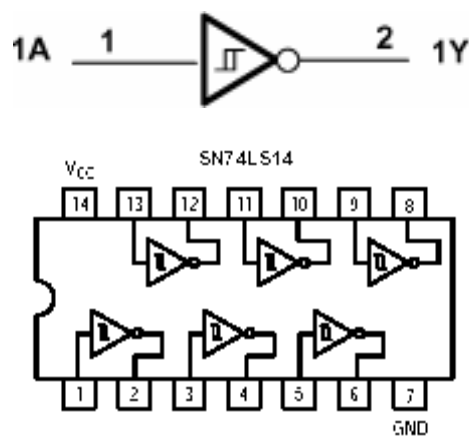
2.7 IC 74LS14 Schmitt triggers

IC 74LS14 merupakan IC TTL yang terdiri dari 6 buah gerbang not schmitt triggers. Fungsi dari IC 74LS14 adalah sebagai pembalik dan pemantap untuk mendeteksi taraf dan membentuk kembali pulsa-pulsa yang buruk pada tepi-tepinya agar dapat direspon dengan baik. Cara kerja schmitt trigger adalah

tergantung pada inputnya walaupun keluarannya hanya mempunyai dua keadaan yaitu low atau high. Schmitt triggers mempunyai tegangan atas (V_+) dan tegangan bawah (V_-). Ketika tegangan inputnya diatas (V_+) maka keluarannya low dan jika tegangan inputnya dibawah (V_-) maka keluarannya high. Signal input yang berganti-ganti melintasi antara V_+ dan V_- dengan ketentuan keluarannya membentuk gelombang kotak. Sifat ini disebut *hysteresis*. Pada 7414 mempunyai ambang batas 0,9 V untuk tegangan input V_- dan 1,7 V untuk keluaran tegangan input V_+ . Sinyal keluaran Schmitt triggers dengan keluaran gelombang kotak dapat dilihat pada gambar 2.11.



Gambar 2.11. Pembentukan gelombang kotak pada Schmitt trigger inverter



Gambar 2.12. Simbol gerbang logika NOT dan Konfigurasi

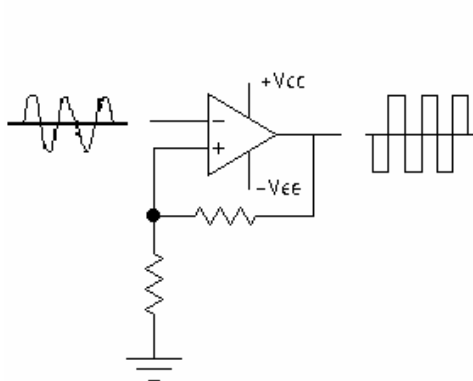
IC TTL 74LS14

Tabel 2.5. Tabel kebenaran dari gerbang logika NOT (74LS14.pdf)

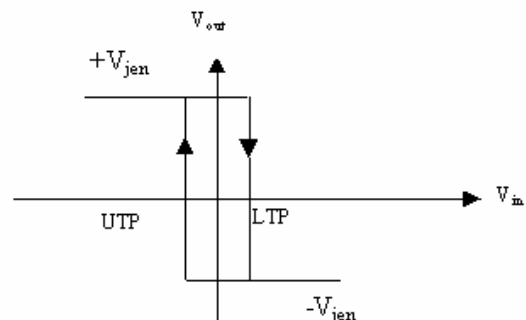
x	\overline{x}
0	1
1	0

Dari table kebenaran diatas jika masukannya low “0” maka keluarannya akan menjadi high “1”, begitu pula jika masukannya high “1” maka keluarannya akan menjadi low “0”.

Pada rangkaian Schmitt trigger inverter dapat juga menggunakan sebuah op-amp, seperti gambar 2.13 dibawah ini.



2.13a Masukan periodic pada pemicu Schmitt menghasilkan keluaran persegi



2.13b. Karakteristik transfer pembalik

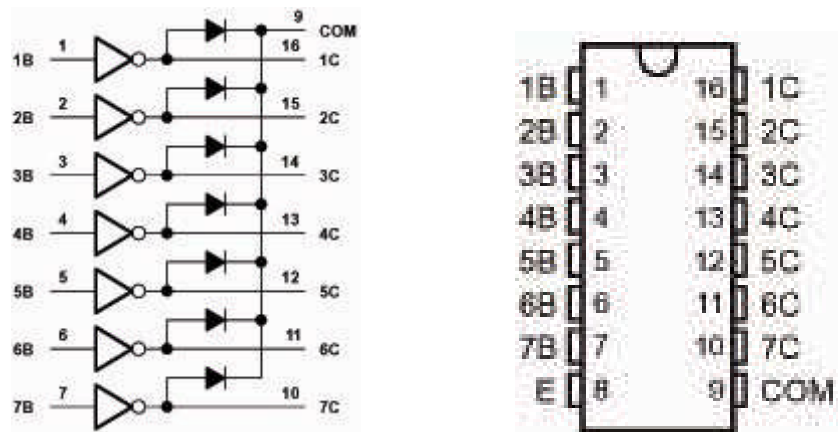
Gambar 2.13a memperlihatkan sebuah pemicu Schmitt dan gambar 2.13b adalah karakteristik transfernya. Bila sinyal masuknya periodic (siklus berulang-ulang), pemicu Schmitt menghasilkan keluaran gelombang persegi, dengan anggapan bahwa sinyal masuknya cukup besar untuk melewati kedua titik perpindahan. Bila pada saat sedang berayun keatas pada setengah siklus positif

tegangan masuknya melebihi UTP, tegangan keluaranya beralih ke $-V_{jen}$. Pada setengah berikutnya, tegangan masuk menjadi lebih negatif dari pada LTP, dan keluarannya beralih kembali ke $+V_{jen}$.

Pemicu Schmitt selalu menghasilkan gelombang persegi terlepas dari bentuk gelombang sinyal masuknya. Dengan kata lain, tegangan masuknya tidak harus sinoida seperti gambar 2.13a. Selama bentuk gelombangnya periodic dan mempunyai amplitudo cukup besar untuk melewati titik-titik perpindahan, kita mendapatkan keluaran gelombang persegi dari pemicu Schmitt.

2.8 IC ULN 2003

IC ULN 2003 merupakan IC yang terdiri dari 6 buah rangkaian Darlington dan dioda proteksi. Fungsi dari IC ULN 2003 adalah sebagai penggerak relay dan juga sebagai driver dari motor stepper berdasarkan penguatan arus, karena motor stepper dan relay membutuhkan arus yang cukup besar. Arus yang keluar pada IC ULN2003 sebesar 500mA (datasheet. pdf). Gambar 2.11 dibawah merupakan gambar konfigurasi ULN 2003 yang mempunyai 7 masukan (1B-7B) dan 7 keluaran (1C-7C). IC ULN 2003 juga mempunyai 7 buah dioda proteksi yang berfungsi melindungi rangkaian Darlington dari kerusakan bila terjadi tegangan balik yang dihasilkan oleh ggl induksi.



Gambar 2.14. Simbol Darlington dan Konfigurasi IC ULN 2003

BAB III

PERANCANGAN ALAT

3.1 Identifikasi Perancangan

Dalam pengembangan rancangan harus mempertimbangkan beberapa kebutuhan yang sangat berpengaruh terhadap tercapainya tujuan, diantaranya adalah masalah komponen penyusunannya yaitu:

1. Adanya sensor yang dapat mendeteksi objek.
2. Adanya keyboard yang digunakan untuk menjalankan alat secara manual dan mengeset waktu.
3. Adanya Mikrokontroler sebagai suatu kesatuan prosesor, memori, dan *input output* (I/O).
4. Adanya display untuk menampilkan waktu.
5. Adanya software untuk menjalankan fungsi diatas.

3.2 Analisis Perancangan

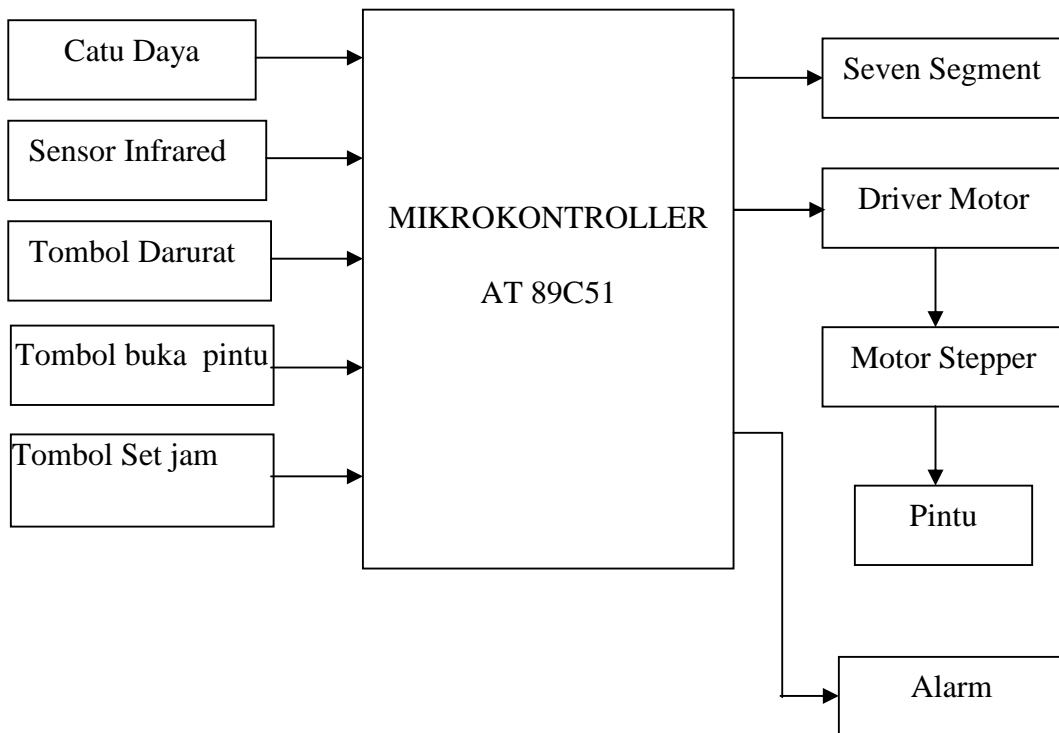
Dari identifikasi kebutuhan diatas, sehingga dapat diperoleh suatu alat yang memiliki spesifikasi sebagai berikut:

1. Menggunakan phototransistor sebagai pendeteksi objek yang menghalangi infra merah sebagai sumber cahayanya.
2. Menggunakan keyboard untuk mengeset waktu dan menggerakkan motor stepper serta untuk membunyikan buzzer.

3. Menggunakan Mikrokontroller AT89C51 sebagai suatu kesatuan prosesor, memori dan input output.
4. Menggunakan seven segment untuk menampilkan waktu.

3.3 Perancangan Hardware

Secara garis besar blok diagram rangkaian model pintu otomatis berbasis mikrokontroller AT89C51 ditunjukkan pada gambar 3.1.



Gambar 3.1. Diagram blok pintu otomatis

Dari blok diagram diatas dapat dilihat bahwa tiap-tiap blok yang ada mempunyai fungsinya masing-masing seperti dapat dijelaskan dibawah ini:

- a. Catu daya (power supply), sebagai sumber tegangan untuk menghidupkan seluruh rangkaian

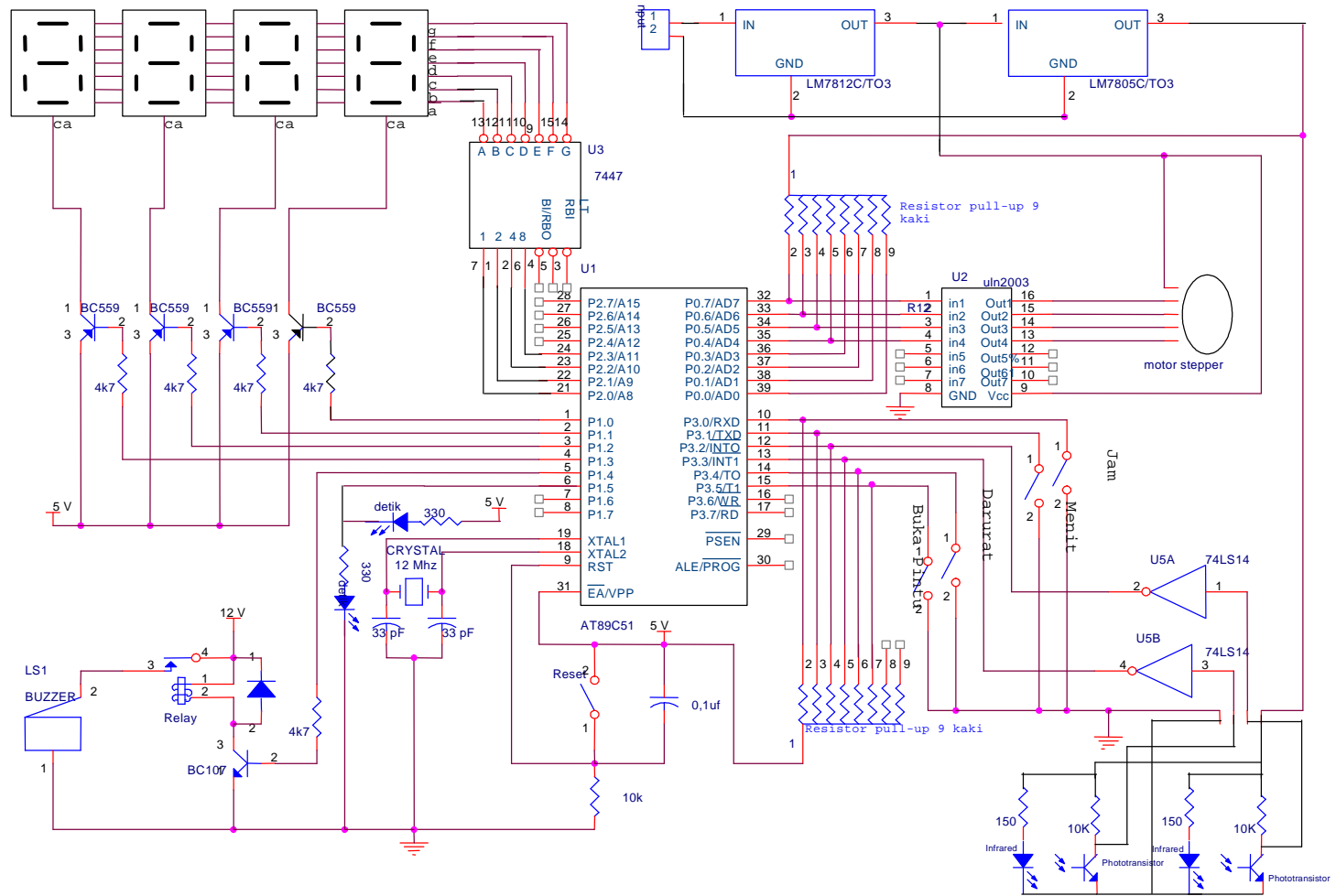
- b. Mikrokontroler AT89C51, merupakan bagian utama dari sistem yang berfungsi untuk mengaktifkan dan mengontrol setiap blok rangkaian sesuai dengan masukan atau instruksi.
- c. Sensor infra merah yang berfungsi untuk mendeteksi adanya objek yang menghalangi sensor infra merah sehingga menghasilkan masukan dan kemudian akan menggerakkan motor stepper secara otomatis.
- d. Seven segment, merupakan bagian dari alat ini yang memberikan tampilan jam digital sebagai acuan (referensi) waktu untuk mengatur aktif tidaknya sensor.
- e. Tombol darurat, berfungsi sebagai tombol yang akan membuat pintu terbuka secara terus-menerus dan membuat sensor tidak aktif bekerja serta membunyikan alarm. Hal ini berguna apabila didalam gedung terjadi kebakaran atau bahaya yang lainnya.
- f. Motor stepper, merupakan suatu piranti yang mengkonversi pulsa elektrik menjadi gerakan mekanik yang proposional.
- g. Driver motor stepper, sebagai rangkaian tambahan dari motor stepper agar motor stepper dapat bergerak dengan semestinya.
- h. Alarm (buzzer), sebagai piranti tambahan yang dipakai untuk menghasilkan bunyi yang bekerja bersamaan pada saat tombol darurat ditekan sebagai tanda bahaya.
- i. Miniatur pintu, sebagai model (miniatur) pintu berukuran mini yang menyerupai pintu yang sesungguhnya.

Jadi cara kerja alat, pada saat catu daya di aktifkan, jam digital langsung running ditandai dengan led detik yang menyala secara bergantian menghitung detik dan tampilan seven segment menunjukkan waktu pukul 00.00. Apabila ada

objek yang menghalangi sensor maka pintu akan tetap diam dan dalam keadaan menutup karena sensor belum aktif bekerja. Bila waktu pada jam di-set menjadi pukul 08.00, maka sensor aktif bekerja dan pintu akan terbuka pada saat ada objek yang menghalangi sensor tersebut, dan kemudian akan kembali menutup setelah selang waktu sekitar 8 detik. Pada saat menutup jika ada objek yang menghalangi sensor infra merah maka pintu akan kembali terbuka dan segera menutup kembali setelah putaran pada motor stepper telah habis tanpa ada delay waktu. Setelah waktu menunjukkan pukul 21.00 maka sensor akan kembali pasif dan tidak akan merespon masukan dari infra merah.

Jika tombol set jam dan menit ditekan maka baik tampilan pada jam maupun menit akan bertambah satu, kemudian jika tombol buka pintu ditekan maka pintu akan terbuka dan akan menutup kembali setelah 8 detik. Pada tombol buka pintu ini dapat bekerja tidak tergantung pada waktu sama dengan tombol pintu darurat jika ditekan maka pintu akan terbuka terus-menerus sekaligus mengaktifkan buzzer. Buzzer akan terus berbunyi sampai tombol darurat atau reset ditekan untuk mengembalikan keadaan seperti semula.

Gambar 3.2. Rangkaian Keseluruhan

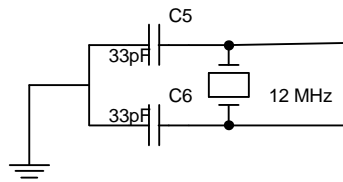


3.3.1 Mikrokontroller AT89C51

Perancangan alat model pintu otomatis menggunakan Mikrokontroller AT89C51 karena *feature* yang dimilikinya, selain itu AT89C51 mudah didapatkan dipasaran dan dengan harga yang murah. Rangkaian standart yang harus dimiliki agar Mikrokontroller dapat digunakan adalah rangkaian pembangkit detak (*clock*), dan rangkaian reset.

3.3.2 Rangkaian Pembangkit Detak

Untuk rangkaian pembangkit detak dapat dilihat pada gambar 3.3 rangkaian ini dapat menggunakan xtal dengan frekuensi 3 Hz - 24 MHz untuk jenis IC AT89C51-24PC dan 2 kapasitor. Untuk perancangan rangkaian alat pintu otomatis menggunakan XTAL 12 MHz dan kapasitor 33 pF, output dari rangkaian pembangkit detak ini masuk Mikrokontroller AT89C51 pada pin 18 dan 19.

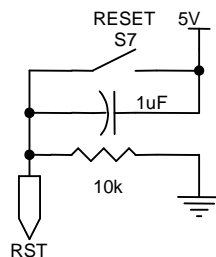


Gambar 3.3. Rangkaian pembangkit detak/clock

3.3.3 Rangkaian untuk Membangkitkan Reset

Rangkaian reset digunakan untuk menghentikan kerja Mikrokontroller dengan kembali ke alamat 0000/reset. Rangkaian reset dapat dilihat pada gambar 13. Untuk mereset Mikrokontroller AT89C51 yaitu dengan memberikan logik tinggi pada pin reset (pin 9) Mikrokontroller AT89C51, logik tinggi ini dibuat minimal 2 *machine cycles* (24 *Oscilator periode*), jika menggunakan xtal 12 Mhz maka logik tinggi minimal selama 1,35us. Besarnya kapasitor adalah

minimal 1 uF sedangkan besarnya resistor dapat dihilangkan karena pin RST sudah mempunyai *internal pulldown*.



Gambar 3.4.Rangkaian reset

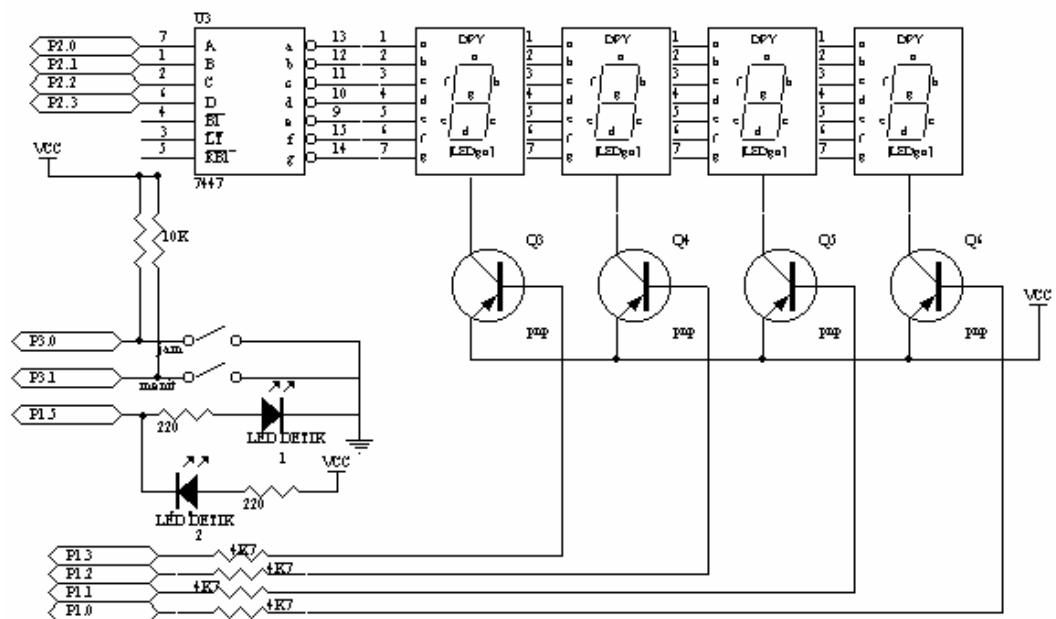
3.3.4 Penampil Seven Segment

Untuk penampil jam digital digunakan seven segment *common anoda*. Dipilih seven segment karena seven segment salah satu jenis penampil yang harganya murah jika dibandingkan dengan penampil lain (LCD dan LED dot matrix), selain itu dalam pembuatan alat ini seven segment hanya digunakan untuk menampilkan data angka (0-9) numerik bukan huruf. Dalam pembuatan alat ini digunakan 4 buah seven segment.

Seven segment mempunyai 7 (tujuh) segment ditambah dengan nyala titik (.), yang masing-masing segmentnya dapat dikontrol secara sendiri-sendiri. Untuk pengontrolan, kontrol segment pada seven segment digunakan IC 74LS47, kemudian (D, C, B, A, F, G, E) pada seven segment dihubungkan dengan pin keluaran IC74LS47 (0A, 0B, 0C,0D, 0E, 0F, 0G). Masukan untuk IC 74LS47 dihubungkan dengan Port 2.0 sampai Port 2.3. IC 74LS47 berfungsi untuk mengubah masukan yang berupa bilangan biner menjadi bilangan desimal. Sedangkan untuk mengatur nyala tiap seven segment/scanning display, masing-

masing pin common anoda seven segment dihubungkan dengan kaki kolektor transistor. Kaki basis pada transistor dihubungkan ke Port 1.0 sampai Port 1.3.

Sistem scanning agar dapat berjalan dengan baik maka pada Port yang terhubung dengan kaki basis pada transistor harus diberi logika 0, sehingga arus pada kaki kolektor akan mengalir ke seven segment dan seven segment menyala. Komponen yang perlu ditambahkan pada penampil jam digital seven segment ini adalah led detik. Led detik menggunakan dua led dan menyala secara bergantian dan dihubungkan ke P1.5, untuk lebih jelasnya dapat dilihat gambar 3.5 rangkaian dibawah ini:



Gambar 3.5. Rangkaian penampil seven segment

3.3.5 Perancangan Rangkaian Keyboard

Keyboard dalam alat pintu otomatis ini digunakan untuk menentukan pilihan, yaitu mengeset jam (P3.0) dan menit (P3.1), menggerakkan motor stepper

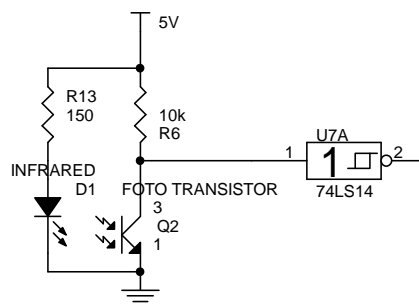
(membuka pintu secara manual) pada P3.5 dan membunyikan alarm/buzzer yang sekaligus menggerakkan motor stepper (pintu terbuka) pada P3.4.

Keyboard yang digunakan agar dapat teridentifikasi oleh Mikrokontroller harus diberi logika low (aktif rendah). Agar dapat menghasilkan aktif rendah maka masing-masing keyboard pada salah satu kakinya harus dihubungkan ke ground dan kaki keyboard yang lainnya dihubungkan kekaki Mikrokontroller (P3.0, P3.1, P3.4, P3.5).

3.3.6 Rangkaian Transduser

Transduser yang digunakan dalam alat ini berupa phototransistor dan infrared. Dalam aplikasinya Transduser ini dipasang didepan pintu dan belakang pintu.

Dalam perancangannya, keluaran dari rangkaian ini masuk ke Mikrokontroller AT89C51 pada Port 3.3 dan Port 3.4. Masukan ke Mikrokontroller berupa aktif rendah (low). Rancangan rangkaian sensor dapat dilihat pada gambar 3.6. *Phototransistor* digunakan sebagai saklar yaitu mati (cut-off) atau hidup (saturasi), jika digunakan sebagai saklar $V_{CC} > R_L \times I_{CC}$. Untuk mengoperasikan phototransistor sebagai saklar besarnya resistor biasanya dipakai nilai 5 k Ω atau lebih.



Gambar 3.6. Rangkaian pendeteksi objek/orang

Ketika ada objek/orang yang melewati antara *phototransistor* dan *infrared*, maka phototransistor tidak terkena cahaya dari *infrared*/terhalang objek/orang akan mengakibatkan phototransistor tidak aktif, sehingga masukan pada kaki IC 74LS14 tinggi dan IC74LS14 akan menyempurnakan pulsa-pulsa yang pada tepi-tepinya tidak sempurna serta mengubah keluaran dari sensor yang merupakan aktif *high* menjadi aktif *low* (rendah) agar dapat teridentifikasi oleh Mikrokontroller, karena Mikrokontroller merupakan piranti dengan aktif *low* (rendah).

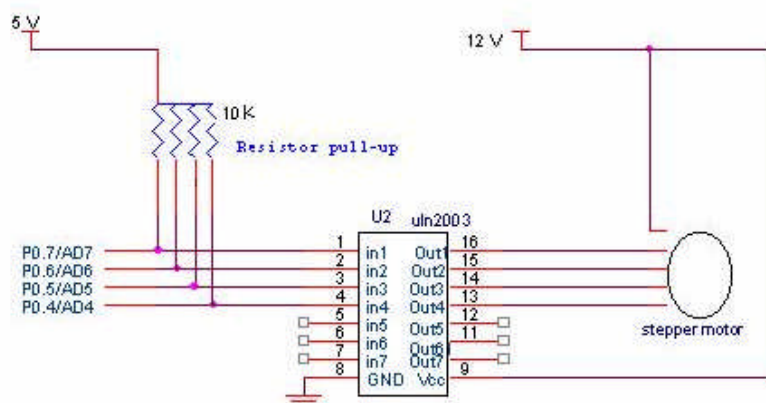
3.3.7 Rangkaian Penggerak / Driver Motor Stepper

Arus yang diperlukan untuk membunyikan menggerakkan motor stepper jauh melebihi kuat arus yang dapat dihasilkan keluaran pada tingkat penyangga sistem digital. Dengan demikian diperlukan suatu untai antarmuka untuk menyediakan arus penggerak untuk mengoperasikan motor stepper.. Penggerak arus disini menggunakan sebuah IC ULN 2003.

IC ULN2003 merupakan penguat arus yang dapat menggerakkan motor stepper. Input untuk IC ULN 2003 dihubungkan kekaki Mikrokontroller (P0.4 – P0.7) dan pada Port 0 dipasang resistor *pull-up* agar keluarannya dapat menghasilkan logik 1. keluaran pada IC ULN 2003 (1C-4C) dihubungkan dengan motor stepper. Pada IC ULN2003 juga mempunyai dioda yang dipasang secara reverse bias. Hal ini berfungsi untuk menjaga agar transistor yang ada tidak rusak bila terjadi tegangan balik yang dihasilkan oleh motor stepper.

Tegangan balik dapat terjadi karena lilitan motor mengalami perubahan arus persatuan waktu, sehingga timbul ggl induksi dan sedangkan untuk menggerakkan motor mengalami perubahan arus yang cepat. Bila ada perubahan dari 0 ke 1 maka ggl yang terjadi akan berlawanan dengan arus yang melewati transistor.

Transistor, dalam keadaan on ggl induksi yang terjadi tidak mempengaruhi transistor. Bila ada perubahan dari 1 ke 0 maka transistor dalam keadaan off dan ggl induksi dapat merusak transistor. Pada saat transistor dalam keadaan off tersebut dioda berfungsi membypass ggl yang terjadi sehingga transistor aman. Rangkaian driver motor stepper dapat dilihat pada gambar 3.7



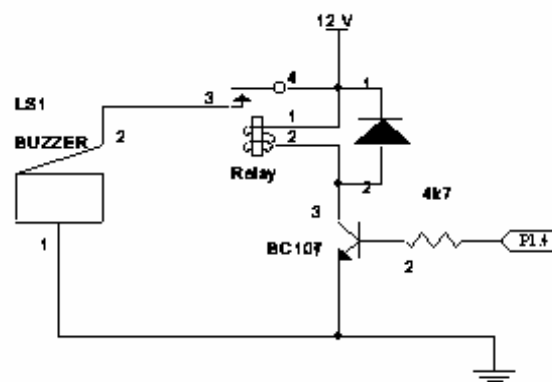
Gambar 3.7. Rangkaian Driver Motor Stepper

3.3.8 Rangkaian penggerak relay pada buzzer

Arus yang diperlukan untuk membunyikan buzzer jauh melebihi kuat arus yang dapat dihasilkan keluaran pada tingkat penyangga sistem digital. Dengan demikian diperlukan suatu untai antarmuka untuk menyediakan arus penggerak

untuk mengoperasikan buzzer. Penggerak arus disini menggunakan sebuah transistor, relay dan dioda yang dipasang secara reverse bias.

Agar transistor dapat berfungsi maka Kaki basis pada transistor dihubungkan ke Port 1.4 dan pada kolektornya dihubungkan ke relay. Pada saat logika 1 diumpankan ke untai antarmuka, transistor akan berprasiap maju, menghasilkan arus yang melewati kumparan relay dan akan menggerakkan relay sehingga saklar pada relay akan tertutup yang akan membunyikan buzzer. Pada saat logika 0 diumpankan ke untai transistor akan *cut-off* dan arusnya menjadi nol dan saklar pada relay terbuka. Dioda pada rangkaian penggerak arus ini berfungsi sebagai pelindung yang disisipkan untuk mencegah timbulnya tegangan lebih pada transistor yang mungkin muncul pada kolektornya disebabkan oleh gaya gerak listrik balik yang dibangkitkan oleh kumparan pada saat transistor off. Tegangan lebih ini sering kali melebihi tegangan maksimum rata-rata kolektor, sehingga bisa merusak transistor. Dioda yang dipasang prasikap balik akan menghantarkan jika tegangan melebihi tegangan rata-rata. Untuk lebih jelasnya dapat dilihat pada gambar 3.8 dibawah ini:



Gambar 3.8. Rangkaian penggerak relay pada buzzer

3.4 Perancangan Software

Dalam perancangan sistem mikroprosesor termasuk juga Mikrokontroler diperlukan software/program untuk menjalankannya. Software atau sering disebut dengan perangkat lunak digunakan memberikan langkah-langkah yang harus dilakukan oleh CPU. Bahasa yang digunakan untuk memprogram Mikrokontroler adalah bahasa assembly.

Program-sumber assembly (assembly source program) merupakan kumpulan dari baris-baris perintah yang ditulis dengan program penyunting-teks (text editor) sederhana. Contoh program assembly adalah notepad. Dalam perancangan alat model pintu otomatis menggunakan bahasa assembly Mikrokontroler AT89C51.

3.4.1 Program inisialisasi

Berisi program inisialisasi stack pointer yang diletakkan pada alamat 50H, inisialisasi interupsi timer 0 dan timer 1, interupsi eksternal 0 dan 1. selain itu juga menyiapkan pengesetan waktu interupsi timer. Timer 0 dan timer 1 menggunakan mode yaitu mode 16 bit. Untuk menunda selama satu detik(1.000.000us) =50000 us x 20 pengulangan. Jadi yang harus diisikan ke TL0/TH0 yaitu 50000us atau dalam bentuk heksa C350h. untuk interupsi eksternal 0 dan 1 terjadi ketika posisi transisi dari 1 ke 0. selain itu juga berisi program pengosongan data memory alamat 01h sampai 60h. Berikut listing program Inisialisasi.

```
ORG    00H           ;alamat awal reset 000H
AJMP   START        ;melompat ke start

ORG    0003H        ;alamat routine interupsi external 0

                0003h
ACALL  INTEREXO     ;panggil subrutine interex0
RETI                          ;kembali ke program utama
```

```

ORG 000BH ;alamat routine interupsi timer 0 000bh
ACALL INTER0 ;memanggil program timer0/subrutine
inter0
RETI ;kembali ke program utama

ORG 0013H ;alamat routine interupsi external 1
0013h
ACALL INTEREXO ;panggil subroutine interexo0
RETI ;kembali ke program utama

ORG 001BH ;alamat routine interupsi timer1 001bh
ACALL INTER1 ;memanggil program timer1/ subrutine
inter1
RETI ;kembali ke program utama

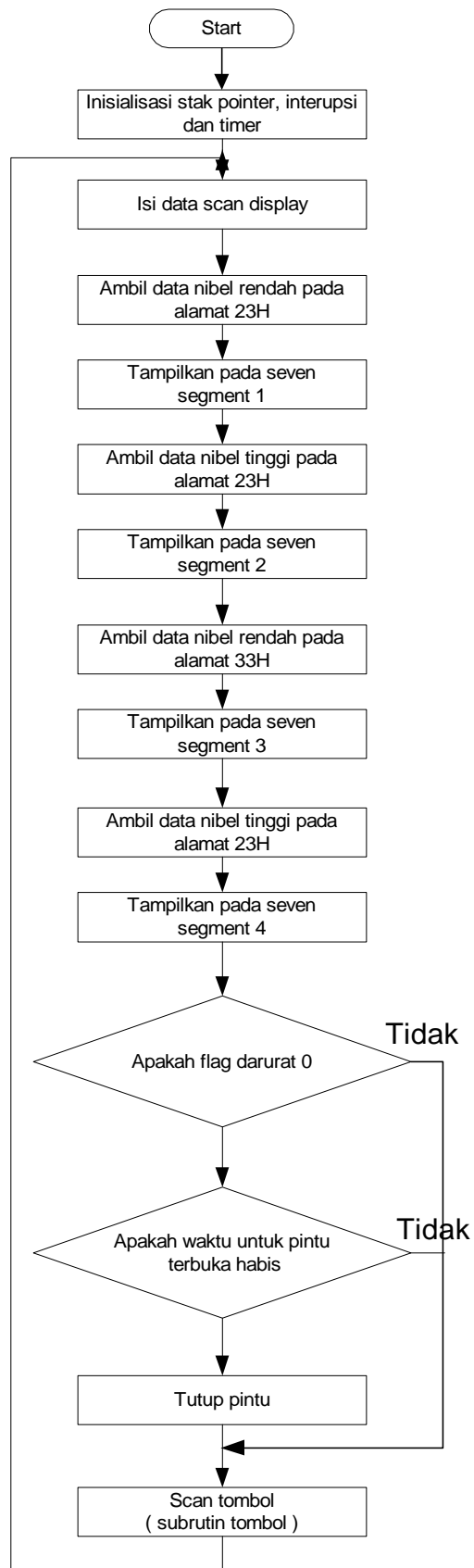
START: MOV SP,#50H ;alamat stack pointer 50h
MOV TMOD,#011H ;timer mode 1
SETB EA ;aktifkan sistem interupsi
SETB ET0 ;interupsi timer 0
SETB ET1 ;interupsi timer 1
SETB EX0 ;interupsi external 0
SETB EX1 ;interupsi external 1
SETB IT0 ;intrpsex0 terjd pd posisi transisi dari
1 ke 0
SETB IT1 ;intrpsex1 terjd pd posisi transisi dari
1 ke 0
MOV TL0,#-50H ;mengisi cacahan pada timer 0 dan timer1
MOV TH0,#-0C3H ;memberikan interupsi dalam 1 detik
MOV TH1,#-0C3H ;sebanyak 20 kali interupsi
MOV TL1,#-50H ;

MOV R0,#01H ;pengosongan alamat 01h sampai alamat
60h
KOS: MOV @R0,#00H ;kosongkan alamat yang ditunjuk oleh R0
INC R0 ;tambah satu data pada R0
CJNE R0,#60H,KOS ;apakah sudah sampai alamat 60H jika
belum
Kembali ke subrutin kos
MOV 20H,#00H ;data melihat time over
MOV 31H,#08H ;31 waktu yang dikehendaki untuk timer1
(berapa second)
MOV 28H,#00H ;kosongkan alamat 28h
SETB 58H ;flag clock 08.00-21.00
CLR 44H ;flag keadaan darurat
SETB TR0 ;aktifkan timer 0

```

3.4.2 Program Utama

Berisi program menampilkan data jam dan menit. Data jam diletakkan pada alamat 33H sedangkan data menit diletakkan pada alamat 23H. Berikut ini flowchart dan listing program utama.



Gambar 3.9. Flowchart program utama

```

AWAL:      MOV    3FH,#0FEH    ;isi alamat 3F dengan data 0FE
           MOV    A,23H      ;isi A dengan data alamat 23H
           ANL   A,#0FH     ;ambil data nibble rendah A
           MOV   P2,A        ;hasilnya keluarkan ke Port 2
           ACALL SCAN       ;panggil subrutine scan
           MOV   A,23H      ;ambil data dari alamat 23 ke A
           ANL   A,#0F0H    ;ambil data nibble tinggi A
           SWAP  A          ;pindah nibble tinggi A ke nibble rendah
                           & sebaliknya
           MOV   P2,A        ;hasilnya keluarkan ke p2
           ACALL SCAN       ;panggil subrutin scan
           MOV   A,33H      ;pindah data pada 33h ke A
           ANL   A,#0FH     ;ambil nibble rendah
           MOV   P2,A        ;keluarkan ke P2
           ACALL SCAN       ;panggil subrutin scan
           MOV   A,33H      ;pindah data pada 33h ke A
           ANL   A,#0F0H    ;ambil nibble tinggi
           SWAP  A          ;hasilnya pindah ke nibble rendah
           MOV   P2,A        ;keluarkan ke P2
           ACALL SCAN       ;panggil subrutin scan
           JB    44H,TRUS    ;PERIKSA TOMBOL DARURAT ON/MATI
           JNB   08H,TRUS    ;jika bit 08h=0 lompat ke trus
           CLR   TR1        ;clearkan TR1
           MOV   21H,#00H   ;kosongkan alamat 21h
           SETB  70H        ;set bit 70h (tanda pintu sedang
                           menutup)
           ACALL PUTAR2     ;panggil subrutin putar2. (tutup pintu)
           CLR   70H        ;clearkan bit 70h(tanda pintu sudah
                           Selesai menutup)
           MOV   P0,#00H    ;reset keadaan motor stepper
           MOV   27H,#00H   ;tanda pintu telah di tutup
TRUS:      ACALL TOMBOL     ;panggil subrutin tombol
           SJMP  AWAL       ;kembali ke awal

```

3.4.3 Subrutin Tombol

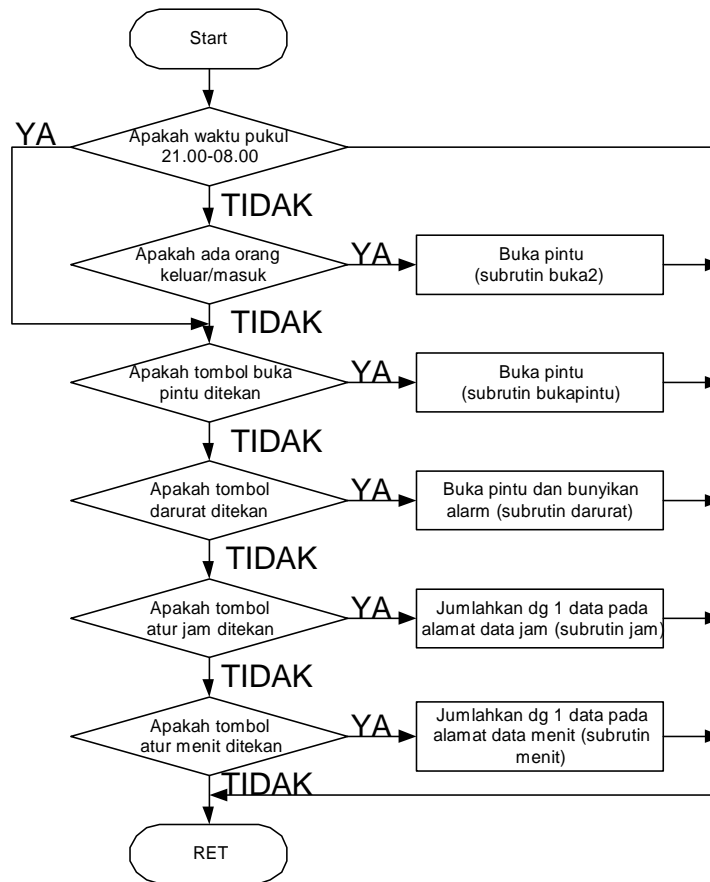
Berisi program scanning tombol dan sensor inframerah. Berikut ini flow

chart dan listing program subrutin tombol

```

TOMBOL:   JB    58H,TOM2    ;jika flag 58h=0 pintu tidak off, jika 1
                           lompat ke tom2
           JNB   71H,TOM2    ;jika flag IR 1 buka pintu jika 0 lompat ke
                           tom2
           ACALL BUKA2      ;buka pintu
TOM2:     JB    P3.0,TOM3    ;apa tombol jam ditekan, jika tidak lompat ke
                           tom3
           ACALL JAM        ;jika ya tambah satu data jam
TOM3:     JB    P3.1,TOM4    ;apa tombol menit ditekan, jika tidak Lompat
                           ke tom4
           ACALL MENIT      ;jika ya tambah satu data menit
TOM4:     JB    P3.4,TOM5    ;apa tombol darurat ditekan, jika tidak lompat
                           ke tom5
           ACALL DARURAT    ;jika ya buka pintu & bunyikan alarm
TOM5:     JB    P3.5,ENDTOMBOL ;apa tombol buka pintu ditekan, jika
                           tidak RET
           ACALL BUKAPINTU  ;jika ya buka pintu
ENDTOMBOL: RET             ;selesai menjalankan subrutin tombol

```



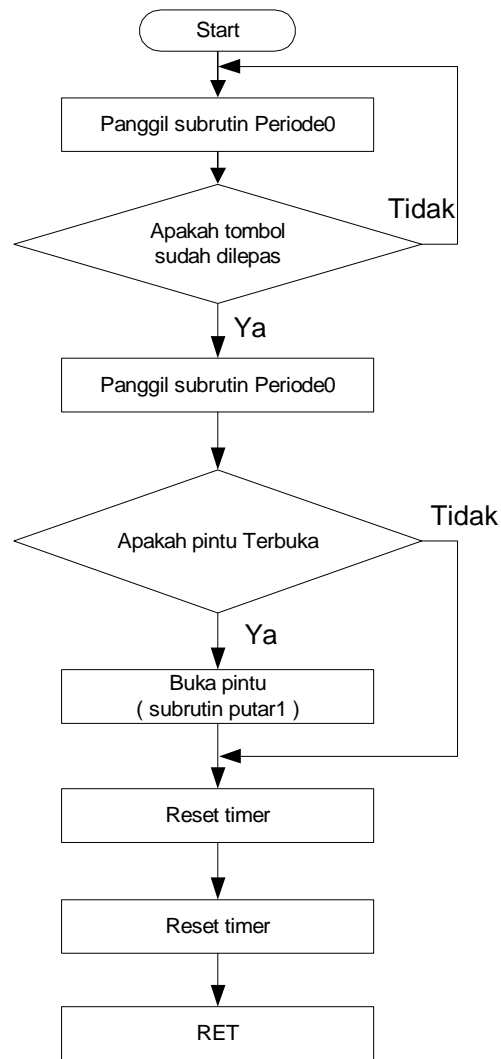
Gambar 3.10. Flowchart subrutin tombol

3.4.4 Subrutin Buka Pintu

Berisi program pemanggilan waktu tunda untuk mengatasi bouncing dan pemanggilan subrutin putar1 untuk membuka pintu dan mengaktifkan timer 1 selama 8 detik. subrutin buka pintu yang dipantau adalah tombol buka pintu. Berikut ini flow chart dan listing program Buka pintu.

```

BUKAPINTU:  ACALL PERIODEO           ;MENGATASI BOUNCING
             JNB P3.5,BUKAPINTU   ; jika p3.5=1 maka
             ACALL PERIODEO       ;mengatasi bouncing
             JB 38H,RESET_WAKTU3  ;ALAMAT 27.1, KEADAAN PINTU
             ACALL PUTAR1 ;panggil subrutin putar1
             MOV P0,#00H          ;reset keadaan motor stepper
RESET_WAKTU3: MOV R7,#00H        ;RESET WAKTU
             MOV 27H,#01H         ;keadaan pintu sedang terbuka
             SETB TR1             ;aktifkan timer1
             RET
  
```



Gambar 3.11. Flowchart subrutin buka pintu

3.4.5 Subrutin Buka2

Berisi subrutin putar1 untuk membuka pintu dan mengaktifkan timer 1 selama 8 detik. subrutin buka2 yang dipantau adalah flag bit 71h. Bit 71h adalah bit tanda pintu harus dibuka karena sensor mendeteksi orang. Berikut ini flowchart dan listing program buka2

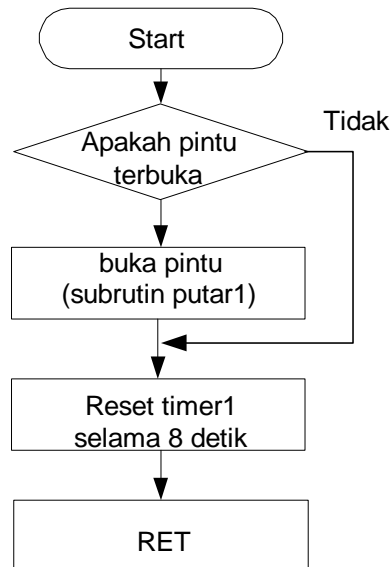
```

BUKA2:      CLR    71H           ;kosongkan bit 71h (bit 71 tanda
                                pintu harus dibuka
            JB     38H,RESET_WAKT2 ;ALAMAT 27.1, KEADAAN PINTU
            ACALL  PUTAR1       ;panggil subrutin putar1
            MOV    P0,#00H      ;reset keadaan motor stepper
RESET_WAKT2: MOV    R7,#00H     ;RESET WAKTU
  
```

```

MOV    27H,#01H    ; keadaan pintu sedang terbuka
SETB   TR1         ; aktifkan timer 1 timer 8 detik
RET     ; keluar ke subrutin buka2

```



Gambar 3.12. Flowchart subrutin buka2

3.4.6 Subrutin Scan

Berisi program untuk scanning display seven segment. Data scanning diletakkan pada alamat 3FH. Scanning dimulai dari seven segment 1 kemudian digeser kekiri. Agar tidak mempengaruhi led detik dan sirine maka sebelum data dikeluarkan ke p1 maka data scan di and kan dengan 0fh (00001111)/ diambil nibble rendah hasilnya kemudian ditambahkan dengan data pada alamat 28h. Alamat 28h berisi bit led detik dan bit sirine. Untuk bit led yaitu bit 28.5 (bit 45h) dan bit sirine yaitu bit 28.4 (bit 44h). Kemudian hasilnya dikeluarkan ke p1. Berikut ini Flow Chart dan listing program subrutin scan.

```

SCAN:  PUSH   ACC           ;simpan data a ke stack pointer
        MOV    A,3FH        ;pindah data pada 3fh ke A
        ANL   A,#0FH       ;ambil nibble rendah (AND A dengan data
                             00001111B
        ORL   A,28H        ;hasilnya di jumlahkan dengan data pada alamat
                             28h

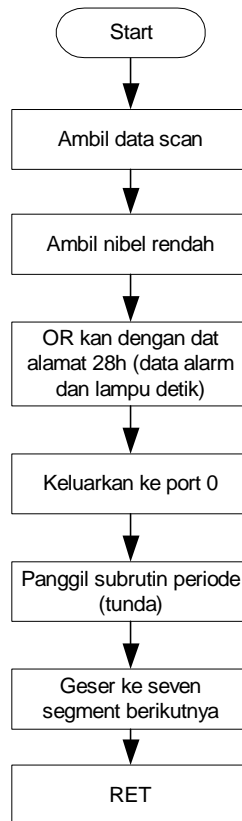
```



```

MOV     P1,A           ;hasilnya keluarkan ke p1
ACALL  PERIODE        ;panggil subrutin periode
MOV     A,3FH         ;data di alamat 3fh pindah ke A
RL      A             ;data di A digeser ke kiri
MOV     3FH,A         ;hasilnya simpan kembali di alamat 3fh
POP     ACC           ;ambil data A dari stack pointer
RET                               ; kembali ke program semula

```



Gambar 3.13. Flowchart subrutin scan

3.4.7 Subrutin Jam

Berisi program pengaturan jam. Ketika tombol jam ditekan data jam pada alamat 33h dijumlahkan dengan 1 selain itu juga dilakukan pendeteksian waktu, yaitu pukul 8 dan pukul 21.00. Berikut ini Flowchart dan listing subrutin jam.

```

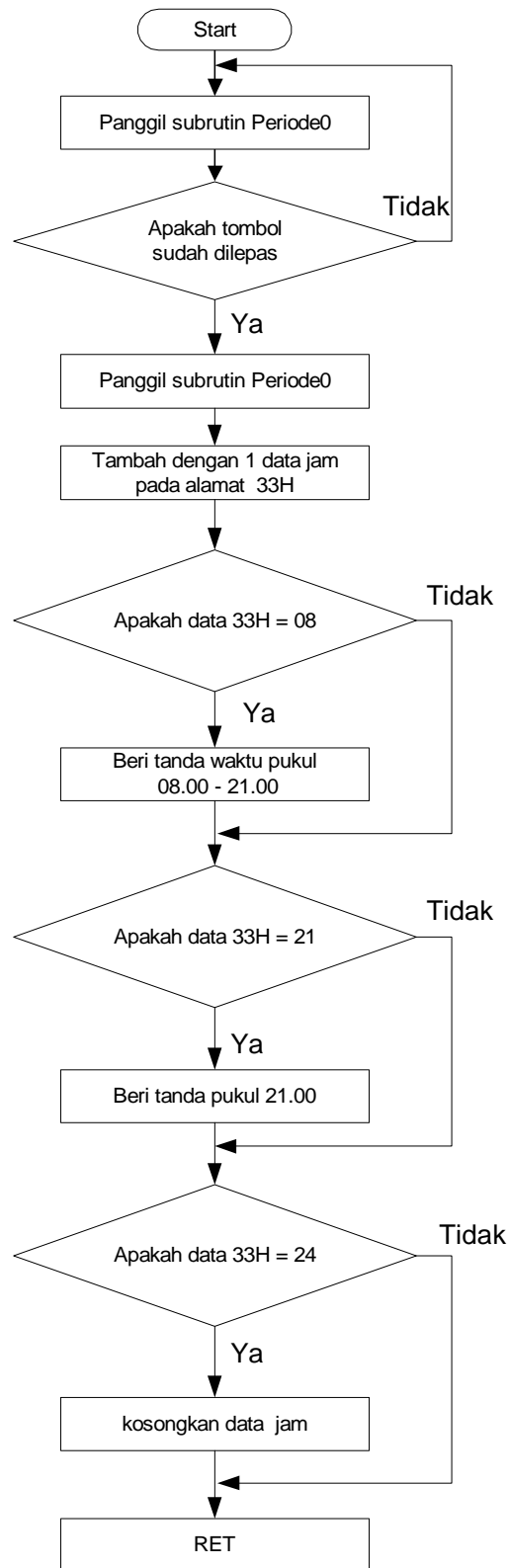
JAM:      PUSH  ACC           ;simpan data A ke stack pointer
          ACALL PERIODEO      ;panggil subrutin periodeo
          JNB   P3.0,JAM      ;jika p3.0=0 tunggu sampai 1
          ACALL PERIODEO      ;jika sudah 1 panggil periodeo
                                   (mengatasi bouncing)
          MOV   A,33H         ;data jam pada alamat 33h pindah ke A

```

```

ADD    A,#01H      ;data A tambah dengan 1
DA     A           ;ubah ke bilangan BCD
MOV    33H,A       ;hasil simpan lagi ke alamat 33h
CJNE   A,#08H,RESIS ;jika A=08 (clock 08.00) maka
CLR    58H         ;beri tanda pada bit 58h=0 (sensor pintu
                    berfungsi)
RESIS: CJNE   A,#21H,LAN2 ;jika A=21 (clock 21.00) maka
SETB   58H        ;beri tanda pada bit 58h=1 (sensor off)
LAN2:  CJNE   A,#24H,TERUS5 ;jika data jam = 24 (clock 24.00) maka
MOV    33H,#00H   ;kosongkan 33h/clock 00.00
TERUS5: POP    ACC ;kembalikan data A dari stack pointer
RET                                ;kembali

```

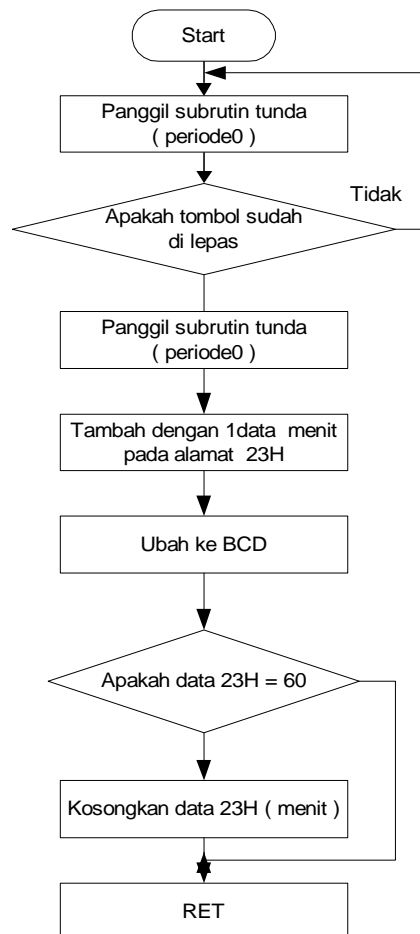


Gambar 3.14. Flowchart subrutin jam

3.4.8 Subrutin Menit

Berisi program pengaturan menit. Ketika tombol menit ditekan dilakukan penjumlahan data pada alamat 23H. Penjumlahan maksimal sampai 60. Berikut ini listing dan flowchart subrutin menit

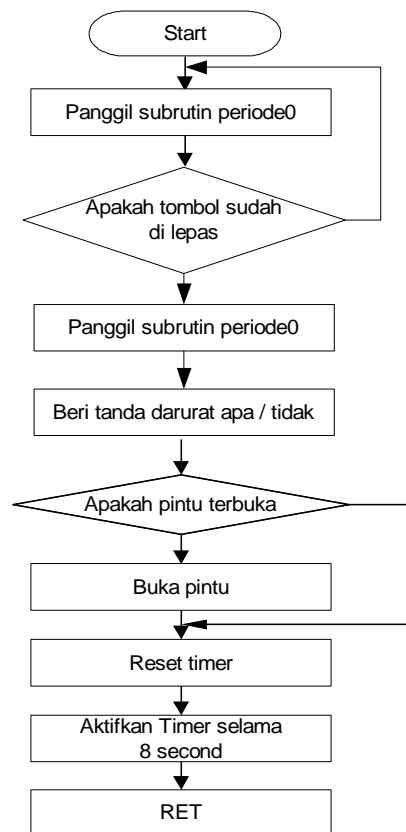
```
MENIT:      PUSH  ACC           ;simpan data A ke stack pointer
            ACALL PERIODEO ;panggil subrutin periodeo
            JNB  P3.1,MENIT ;jika p3.1 =1 maka
            ACALL TUNDA    ;panggil subrutin tunda
            ACALL PERIODEO ;panggil subrutin periodeo (mengatasi
                          bouncing)
            MOV  A,23H     ;pindah data 23h ke A
            ADD  A,#01H    ;data di A tambah dengan 1 data
            DA   A         ;ubah ke BCD
            MOV  23H,A     ;hasil simpan kembali ke alamat 23h
            CJNE A,#60H,TERUS6 ;jika A = 60 maka
            MOV  23H,#00H ;kosongkan data menit
TERUS6:     POP  ACC       ;kembalikan data A dari stack pointer
            RET           ;kembali
```



Gambar 3.15. Flowchart subrutin menit

3.4.9 Subrutin Darurat

Subrutin darurat berisi program untuk membuka pintu dan membunyikan alarm tanda bahaya. Untuk mematikan dengan cara menekan lagi tombol darurat atau dengan menekan tombol reset tetapi akan kembali ke awal program. Berikut ini listing dan Flow Chart subrutin darurat



Gambar 3.16. Flowchart subrutin darurat

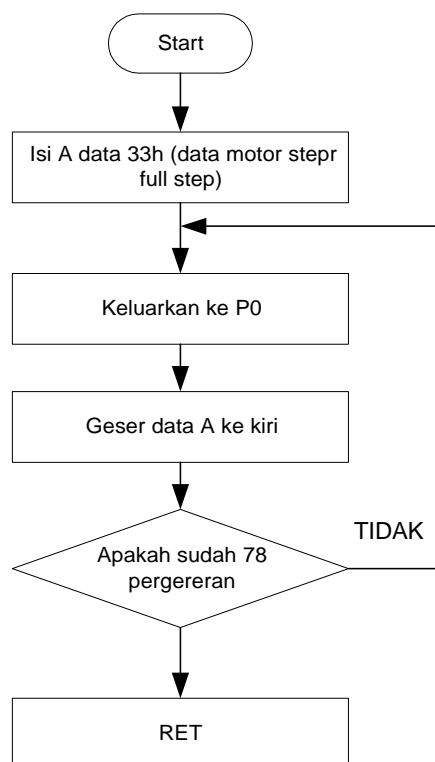
```

DARURAT:      ACALL PERIODEO          ;MENGATASI BOUNCING
              JNB     P3.4,DARURAT  ;jika p3.4 = 1 maka
              ACALL PERIODEO          ;panggil periodeo (mengatasi
                                      bouncing)
              CPL     44H             ;balik flag darurat
              CPL     P1.4           ;bunyikan sirine
              JB      38H,RESET_WAKTU2 ;ALAMAT 27.1, KEADAAN PINTU
              ACALL  PUTAR1          ;panggil subrutin putar1
              MOV     P0,#00H        ;reset motor stepper
RESET_WAKTU2: MOV     R7,#00H        ;reset waktu, kosongkan data
                                      r7
              MOV     27H,#01H       ;
              SETB   TR1             ;aktifkan timer1
              RET                    ;kembali
  
```

3.4.10 Subrutin Putar1

Berisi program menjalankan motor stepper untuk membuka pintu. Data yang dikeluarkan ke P0 step 1: 00110011, step ke 2: 01100110 step ke 3 :11001100 step ke4: 10011001 dan kembali ke 00110011 sampai 27h atau bilangan desimalnya 39 kali dan diulang sebanyak dua kali jadi 78 kali putaran.

berikut ini listing program dan flowchart subrutin putar1



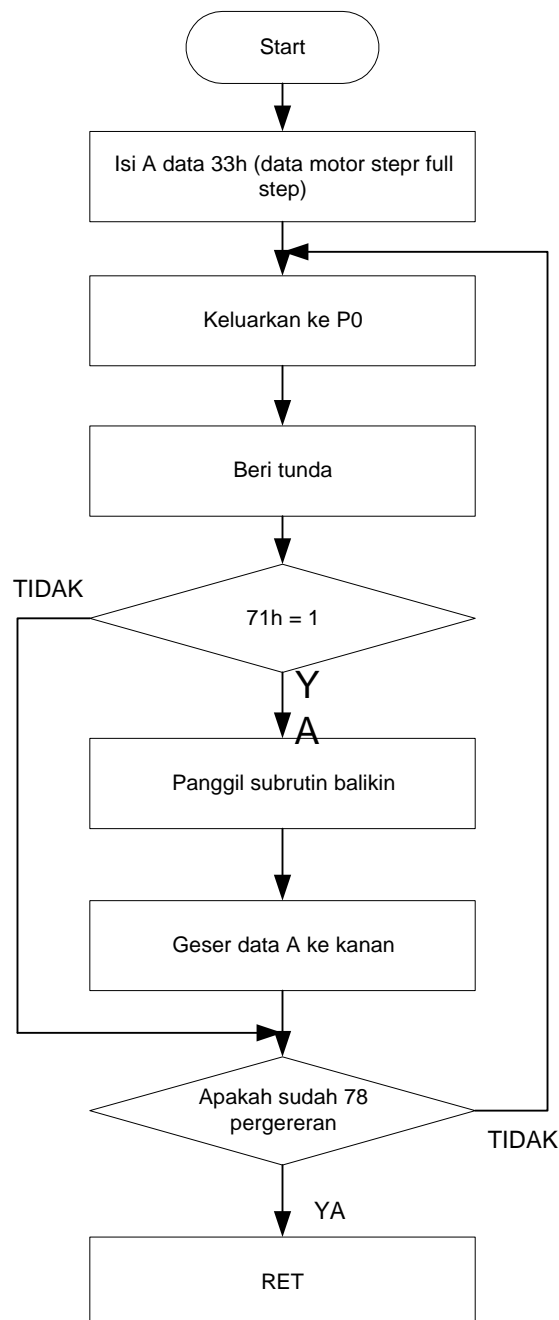
Gambar 3.17. Flowchart subrutin putar1

```

PUTAR1:  PUSH  ACC           ;simpan data a ke stack pointer
        MOV   R5,#33H  ;isi r5 data 33h
        MOV   3CH,#02H ;isi 3ch data 02
        MOV   A,R5     ;pindah data r5 ke a
AL:      MOV   24H,#27H ;Isi 24h data 27h
UL:      MOV   P0,A     ;keluarkan data di A ke P0
        ACALL TUNDA   ;panggil tunda
        RL    A        ;data A digeser ke kiri
        MOV   R5,A     ;hasil simpan kembali ke r5
        DJNZ 24H,UL    ;24h-1 jika belum 0 lompat ke ul jika
                    ;24h=0 maka
        DJNZ 3CH,AL    ;3ch-1 jika belum 0 lompat ke al jika
                    ;3ch=0 maka
        POP  ACC       ;ambil data a dari stack pointer
        RET           ;kembali
  
```

3.4.11 Subrutin Putar2

Berisi subrutin balikin, selain itu program ini untuk menjalankan motor stepper/ menutup pintu. Data yang dikeluarkan ke P0 step 1: 00110011, step ke 2: 01100110 step ke 3 :11001100 step ke4: 10011001 dan kembali ke 00110011. Berikut ini listing program dan flowchart subrutin putar2



Gambar 3.18. Flowchart subrutin putar2

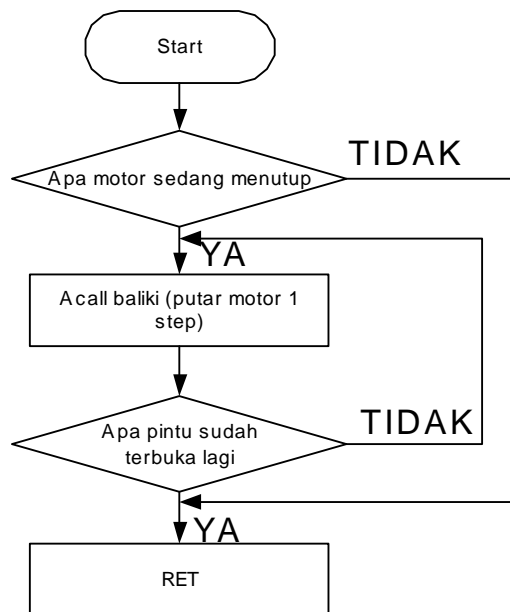
```

PUTAR2:    PUSH  ACC           ;simpan data A ke SP
           MOV   R5,#33H    ;isi r5 data 33h (data M stepper
           MOV   3CH,#02H   ;isi 3ch data 02h (jumlah pengali
                               geseran stepper)
           MOV   A,R5       ;pindahkan data r5 ke A
AL2:       MOV   24H,#27H   ;isi 24h data 27h (data jumlah
                               geseran pintu)
UL2:       MOV   P0,A       ;keluarkan data M stepper ke P0
           ACALL TUNDA     ;beri tunda (panggil subrutin tunda)
           RR    A          ;geser kekanan
           MOV   R5,A       ;hasil geseran simpan di R5
           JNB  72H,TDKNORMAL ;apa ketika menutup sensor on
                               jika ya maka
           ACALL BALIKIN   ;buka kembali (panggil subrutin
                               balikin)
           CLR  72H        ;kosongkan bit 72h (bit tanda pintu
                               harus dibuka)
TDKNORMAL: DJNZ  24H,UL2    ;24h - 1 jika hasilnya 0 maka
           DJNZ  3CH,AL2    ;3ch-1 jika hasilnya 0 maka
           POP  ACC         ;ambil data

```

3.4.12 Subrutin Balikin

Berisi program untuk menjalankan motor stepper. Subrutin ini dipanggil ketika pintu sedang menutup sensor mendeteksi ada orang, maka pintu harus dibuka lagi. Pintu akan dibuka sebanyak pergeseran yang telah dilakukan pada proses penutupan pintu. Motor diputar sampai mencapai kedudukan awal (pintu terbuka). Alamat 3Ch dan 24h digunakan untuk menyimpan data pergeseran. Alamat 3Ch sebagai pengali. Alamat 24h terisi 27h atau 39 dalam desimalnya dan alamat 3Ch berisi 02h maka motor stepper akan diputar sebanyak $39 \times 2 = 78$ step. Berikut ini listing program dan flowchart subrutin balikin.



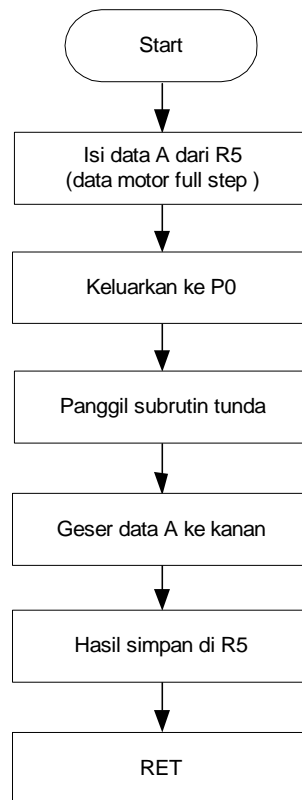
Gambar 3.19. Flowchart subrutin balikin

```

BALIKIN:  PUSH  00H           ;simpan data r0 ke SP
          MOV   R0,3CH       ;pindah data 3ch ke R0
          CJNE  R0,#02H,AW2  ;apa r0 = 02 jika benar maka
          SJMP  SERYES       ;melompat ke seryes
AW2:      INC   3CH          ;data di 3ch tambah 1
          PUSH  24H          ;simpan data 24h ke SP
          MOV   24H,#00H     ;isi data 24h data 00h
ABC:      INC   24H          ;data 24h tambah 1
          ACALL SER          ;geser 1 step (panggil subrutin ser)
          MOV   R0,24H       ;pindahkan data 24h ke r0
          CJNE  R0,#27H,ABC  ;apa R0 =27H jika benar maka
          POP   24H          ;kembalikan data 24h dari stack
                               pointer
SERYES:   MOV   R0,24H       ;pindahkan data 24h ke r0
          CJNE  R0,#27H,AW1  ;apa data di r0=27h jika benar maka
          POP   00H          ;kembalikan data 00 dari stack
                               pointer
          RET                ;kembali
AW1:      INC   24H          ;data 24 tambah 1
          ACALL SER          ;panggil subrutin ser
          SJMP  SERYES       ;lompat ke subrutin seryes
  
```

3.4.13 Subrutin SER

Berisi program untuk menjalankan motor stepper 1 step



Gambar 3.20. Flowchart subrutin ser

```
SER:      PUSH  ACC           ;simpan data A ke SP
          MOV   A,R5        ;pindahkan data r5 ke A
          MOV   P0,A        ;keluarkan data a ke P)
          ACALL TUNDA       ;panggil subrutin tunda
          RL    A           ;geser kekiri
          MOV   R5,A        ;hasilnya simpan ke r5
          POP  ACC          ;ambil data A dari SP
          RET                ;kembali
```

3.4.14 Subrutin Tunda

Subrutin tunda digunakan untuk memberikan tunda pada motor stepper.

Waktu tunda yang di berikan untuk menjalankan motor stepper sebesar 30723 μ S

atau 30,723 ms. Berikut ini listing program subrutin tunda:

```

TUNDA:      MOV    25H,#0F0H
TUN2:      MOV    26H,#05H
TUN:       NOP
           DJNZ   26H,TUN
           DJNZ   25H,TUN2
           RET

```

3.4.15 Subrutin Periode

Subrutin periode digunakan untuk memberikan tunda pada scanning penyalan seven segment. Waktu tunda yang di berikan untuk scanning penyalan seven segment adalah sebesar 2460 μ S atau 2,46 ms. . Berikut ini listing program subrutin periode:

```

PERIODE:    MOV    1CH,#09H
PERIODE1:   MOV    1DH,#09H
PERIODE2:   MOV    1EH,#09H
PERIODE3:   NOP
           DJNZ   1EH,PERIODE3
           DJNZ   1DH,PERIODE2
           DJNZ   1CH,PERIODE1
           RET

```

3.4.16 Subrutin Periode0

Subrutin periodeo tunda untuk mengatasi bouncing pada tombol. Waktu tunda yang di berikan untuk mengatasi bouncing adalah sebesar 176448 μ S atau 176,448 ms. Berikut ini listing program subrutin periode0:

```

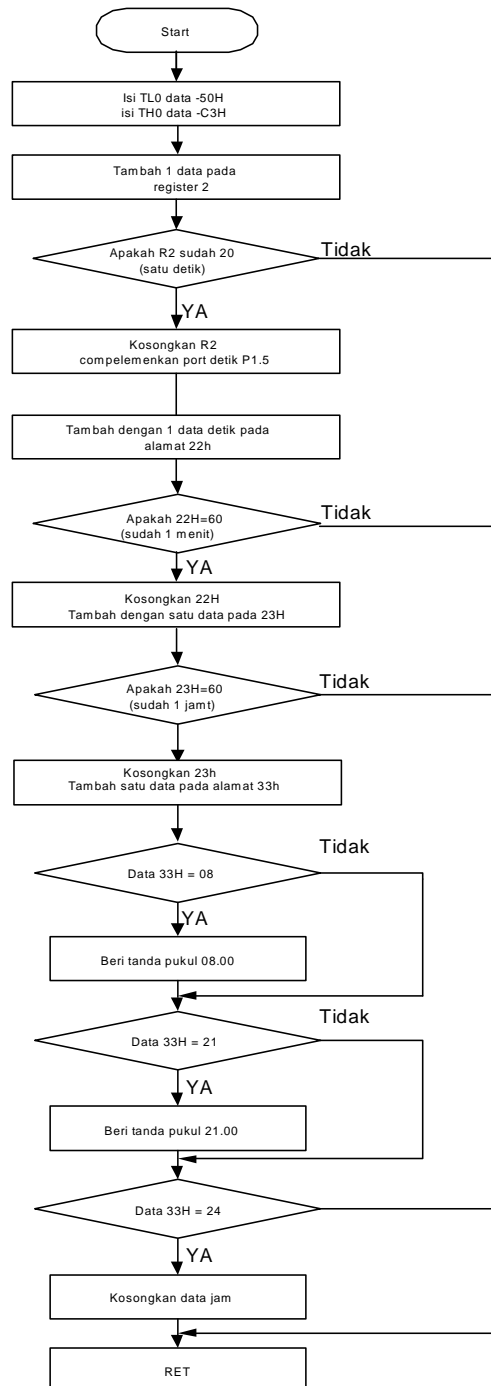
PERIODEO:   mov r1,#00000001b
PERIODEO1:  mov r4,#11111111b
PERIODEO2:  mov r3,#01111111b
           djnz r3,$
           djnz r2,PERIODEO2
           djnz r3,PERIODEO1
           ret

```

3.4.17 Subrutin Inter0

Berisi program menjumlahkan data detik menit jam dan mendeteksi jam pintu bisa beroperasi yaitu pukul 08.00 sampai 21.00. Data limpahan interupsi setiap 50000us disimpan di R2 (setiap 1 detik menjalankan subrutin timer0 sebanyak 20 kali). data detik disimpan di alamat 22H data menit disimpan di

alamat 23H dan alamat jam disimpan di alamat 33H. Berikut ini Flowchart dan listing subrutin timer0



Gambar 3.21. Flowchart subrutin inter0

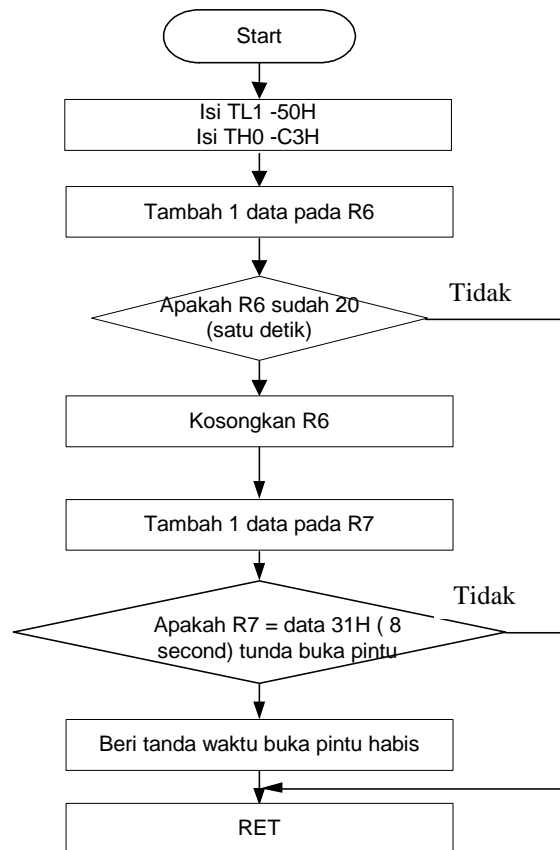
```

INTER0:  PUSH  ACC           ;simpan data A ke stack pointer
         MOV   TL0,#-50H ;isi cacahan timer (50000 / c350H)
         MOV   TH0,#-0C3H
         INC   R2         ;tambah 1 data pada r2
         CJNE  R2,#20,TERUS1 ;jika r2=20 maka
         MOV   R2,#00H    ;kosongkan r2
         CPL   P1.5       ;ubah nyala led(matikan /hidupkan)secara
                           gantian
         CPL   45H        ;bit alamat on/off led
         INC   22h        ;data alamat 22h tambah 1(data detik)
         MOV   A,22h      ;data pada alamat 22h kirim ke A
         DA    A          ;ubah ke BCD
         MOV   22H,A      ;hasilnya simpan ke alamat 22h
         CJNE  A,#60H,TERUS1 ;jika belum a=60 maka
         MOV   22H,#00H   ;sudah 1 menit, kosongkan
                           alamat 22h
         MOV   A,23H      ;pindah data menit ke A
         ADD   A,#01H     ;tambah 1 data menit
         DA    A          ;ubah ke BCD
         MOV   23H,A      ;hasil simpan kembali ke alamat 23h
         CJNE  A,#60H,TERUS1 ;jika A=60 maka
         MOV   23H,#00H   ;sudah 1jam,kosongkan data menit
         MOV   A,33H      ;pindah data 33h ke A
         ADD   A,#01H     ;data A tambah dengan 1
         DA    A          ;ubah ke BCD
         MOV   33H,A      ;hasilnya simpan lagi ke 33H
         CJNE  A,#08H,RES1 ;jika a 08 (clock 08.00) maka
         CLR   58H        ;beri tanda sensor boleh aktif
RES1:    CJNE  A,#21H,LAN  ;jika A = 21 maka
         SETB  58H        ;beri tanda sensor tidak boleh ON
LAN:     CJNE  A,#24H,TERUS1 ;jika A=24h maka
         MOV   33H,#00H   ;kosongkan data jam (clock 00.00)
TERUS1:  POP   ACC        ;ambil data A dari stack pointer
         RET              ;kembali

```

3.4.18 Subrutin Inter1

Subrutin timer1digunakan untuk memberikan tunda setelah pintu terbuka. Tunda yang dipakai selama 8s. Untuk menyimpan data limpahan setiap 50000us menggunakan R6, untuk menyimpan data detik R7. setiap 8 detik selesai otomatis akan mengeset alamat 21h logik 01, logik inilah yang digunakan sebagai tanda pintu masih dalam keadaan terbuka atau tertutup. Berikut ini flowchart dan listing subrutin inter1



Gambar 3.22. Flowchart subrutin inter1

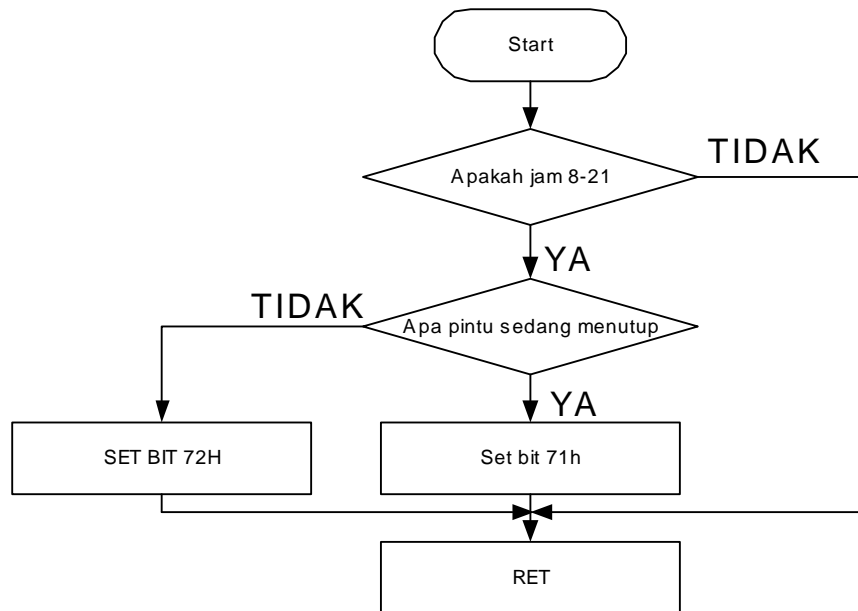
```

INTER1:    PUSH    ACC                ;simpan data a ke stack pointer
           MOV     TL1,#-50H        ;isi cacahan timer
           MOV     TH1,#-0C3H       ;idem
           INC     R6                ; data r6 tambah dengan 1
           CJNE   R6,#20,TERUS2     ;jika r6=20 maka
           MOV     R6,#00H          ;sudah 1 detik
           INC     R7                ;data pada r7 tambah dengan 1
           MOV     A,R7              ;data pada r7 pindah ke A
           CJNE   A,31H,TERUS2     ;jika a=data pada alamat 31h maka
           MOV     R7,#00H          ;kosongkan r7
           MOV     21h,#01H         ;beri flag bit 08h
TERUS2:    POP     ACC                ;kembalikan data a (ambil dari
                                           stack pointer)
           RET                       ;kembali
  
```

3.4.19 Subrutin Interex0

Subrutin interex0 digunakan untuk memberikan flag jika sensor aktif. Bit 70h untuk mendeteksi apakah pintu sedang menutup atau tidak, Bit 71h akan diaktifkan jika pintu harus dibuka secara normal (keadaan pintu sedang tertutup),

untuk bit 72h akan diaktifkan jika pintu harus dibuka secara tidak normal (karena pintu sedang proses menutup).



Gambar 3.23. Flowchart subrutin interex0

```

INTEREX0:  JB    58H,REBUBAR ;jika 58h=0 maka
           JB    70H,ONPINTU ;deteksi apa pintu sedang menutup
           ;jika tidak (70=0)
           SETB  71H ;beri tanda pintu dibuka secara normal
           SJMP  REBUBAR ;melompat ke rebubar
ONPINTU:   SETB  72H ;beri tanda pintu harus dibuka balik
REBUBAR:   RET    ;kembali
           END    ;selesai
  
```

BAB IV

PENGUJIAN DAN PEMBAHASAN

Pengujian ini dilakukan tiap blok rangkaian untuk mengetahui apakah alat yang dirancang sudah sesuai dengan rencana yang ingin dicapai dan dapat dilakukan perbaikan-perbaikan yang dianggap perlu. Setelah itu dilakukan pengujian alat secara keseluruhan dan pengamatan apakah fungsi-fungsi yang ingin dicapai dapat terlaksana. Alat yang digunakan dalam pengujian alat disini menggunakan multimeter digital, power supply, project board dan kabel.

4.1 Rangkaian Sensor

Sensor yang digunakan adalah sensor inframerah yaitu menggunakan photo transistor. Berikut ini data tegangan VCE ketika photo transistor terkena cahaya inframerah atau tidak terkena cahaya.

Tabel 4.1 Tabel hasil pengujian sensor

No	Keadaan photo transistor	VCE
1.	Tersinari Inframerah	0,15 Volt
2.	Tidak tersinari Inframerah	3,5 Volt

Foto transistor dalam rangkaian ini digunakan sebagai saklar (*switch mode*). Ketika foto transistor terkena cahaya inframerah maka phototransistor akan *saturasi* (saklar tertutup). Begitu sebaliknya jika foto transistor tidak terkena

cahaya (terhalang orang yang lewat) maka foto transistor akan *Cut-off* (saklar terbuka). Dari hasil pengujian rangkaian sensor yaitu sensor inframerah, tegangan keluaran / VCE sebesar 3,5 volt (ketika ada orang yang lewat) sudah memenuhi untuk input IC TTL 74LS14 untuk logika 1 dan begitu juga VCE 0,15 V sudah memenuhi untuk input IC TTL 74LS14 untuk logika 0. Fungsi dari 74LS14 pada rangkaian ini untuk mendeteksi taraf dan membentuk kembali pulsa-pulsa yang buruk pada tepi-tepinya. Dari hasil yang diperoleh diatas dapat disimpulkan bahwa sensor berfungsi dengan baik.

4.2 Rangkaian reset

Rangkaian reset berfungsi menghentikan kerja CPU dan kemudian mengulang dari awal (program counter ke alamat 0000). Saat catu daya dihidupkan rangkaian reset menunda kerja dari CPU hingga tegangan stabil (*power on reset*). Reset pada mikrokontroler AT89C51 adalah aktif tinggi. Dengan nilai $R=10\text{ k}\Omega$ dan $C = 1\text{ }\mu\text{F}$ dan *input high* mikrokontroler AT89C51 adalah (0,7 Vcc) atau sebesar 3,5 V maka

$$\begin{aligned}
 3,5 &= 5 \left(1 - e^{-\frac{1}{RC}}\right) \\
 3,5 &= 5 \left(1 - e^{-\frac{1}{0,01}}\right) \\
 3,5 &= 5 \left(1 - e^{-100t}\right) \\
 3,5 &= 5 - 5 e^{-100t} \\
 5 - 3,5 &= 5 e^{-100t} \\
 \frac{1,5}{5} &= e^{-100t} \\
 0,3 &= e^{-100t} \\
 \ln 0,3 &= -100t \\
 t &= \frac{\ln 0,3}{-100}
 \end{aligned}$$

$$t = \frac{-1,2}{-100}$$
$$t = 0,012s$$
$$t = 12ms$$

Dari perhitungan diatas waktu yang diperlukan untuk reset menggunakan rangkaian reset dalam sistem ini adalah 12 ms, sehingga ini telah memenuhi persyaratan karena persyaratan minimalnya adalah 1,35 μ s (2 machine cycle).

4.3 Pengujian Display

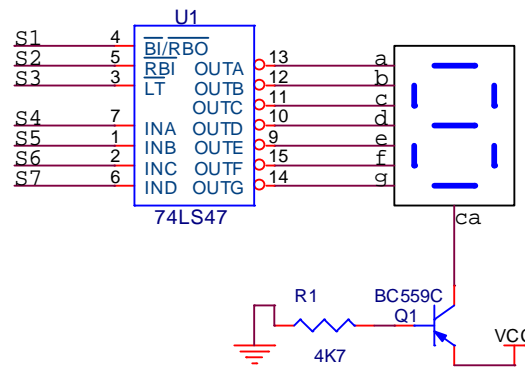
Display merupakan rangkaian yang penting untuk mendukung proses penyelesaian alat ini, karena penampil hasil dari perhitungan program di ATMEL maka perlu menguji apakah display tersebut sudah terangkai dengan benar.

Peralatan pengukuran:

1. catu daya 5 volt
2. multi meter
3. papan percobaan (project board)

langkah-langkah pengukuran:

1. Cari dari kaki seven segment yang berfungsi sebagai a, b, c, d, e, f dan g.
2. Buat rangkaian seperti gambar
3. Berikan catu daya 0 volt pada basis transistor dan 5 V pada emitter
4. S1 sampai S7 merupakan input, logika 1 jika diberi tegangan 5 V dan logika 0 jika di beri tegangan 0V. hasil dari pengujian dapat dilihat pada tabel 4.2.



Gambar 4.1 Pengujian display

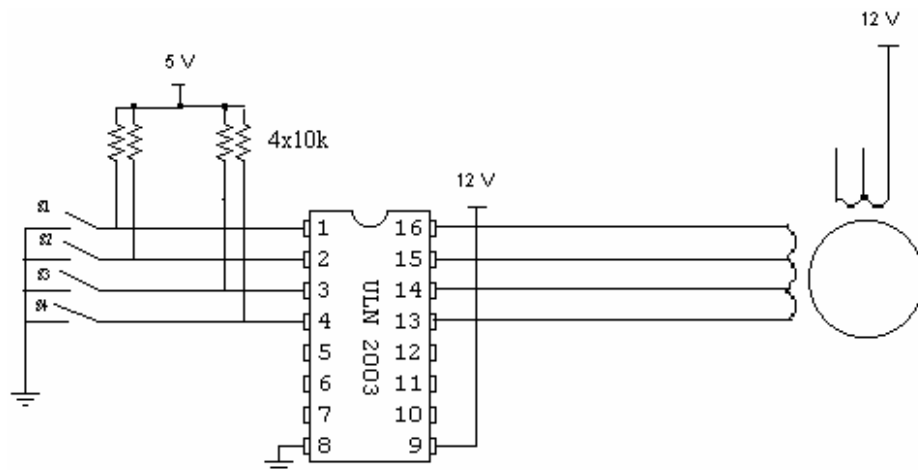
Tabel 4.2 Hasil Pengujian Display

DESIMAL	MASUKAN				KELUARAN						
	BI/RB0	LT	RBI	D C B A	a	b	c	d	e	f	g
0	1	1	1	0 0 0 0	0	0	0	0	0	0	1
1	1	1	0	0 0 0 1	1	0	0	1	1	1	1
2	1	1	0	0 0 1 0	0	0	1	0	0	1	0
3	1	1	0	0 0 1 1	0	0	0	0	1	1	0
4	1	1	0	0 1 0 0	1	0	0	1	1	0	0
5	1	1	0	0 1 0 1	0	1	0	0	1	0	0
6	1	1	0	0 1 1 0	1	1	0	0	0	0	0
7	1	1	0	0 1 1 1	0	0	0	1	1	1	1
8	1	1	0	1 0 0 0	0	0	0	0	0	0	0
9	1	1	0	1 0 0 1	0	0	0	1	1	0	0

4.4 Pengujian Driver Motor Stepper

Pengujian motor stepper dan *driver* motor stepper ditunjukkan pada gambar 4.3. Sebagai masukan ke *driver* motor stepper, dipasang empat buah saklar yang

dihubungkan ke rangkaian *pull-up* untuk menguji apakah rangkaian yang dibuat dapat menggerakkan motor stepper dengan baik.



Gambar 4.2 Rangkaian Pengujian *Driver* Motor Stepper

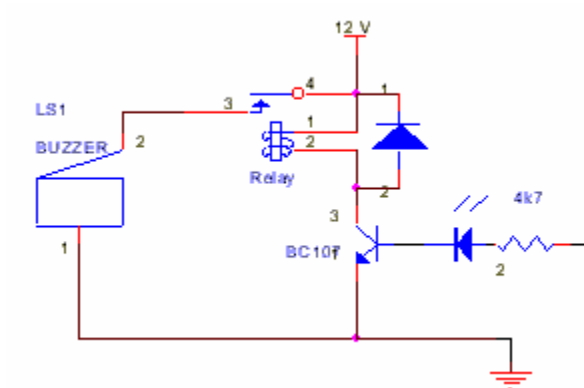
Tabel 4.3 Hasil Pengujian Rangkaian *Driver* Motor Stepper

S1	S2	S3	S4	Reaksi Motor Stepper
On	Off	Off	Off	Berputar
Off	On	Off	Off	Berputar
Off	Off	On	Off	Berputar
Off	Off	Off	On	Berputar

Dari hasil pengujian yang dilakukan, maka dapat diketahui bahwa rangkaian driver motor stepper dapat bekerja dengan baik, begitu pula dengan piranti motor stepper itu sendiri.

4.5 Pengujian Terhadap Relay *Driver* dan buzzer

Pengujian Terhadap Relay *Driver* dan buzzer ditunjukkan pada gambar 4.3 dan sebuah led sebagai indicator.



Gambar 4.3 Rangkaian Pengujian Relay *Driver* dan buzzer

Pada saat basis transistor di berikan tegangan sebesar 5V maka led akan menyala dan buzzer akan berbunyi serta tegangan pada relay sebesar 11,8V, jika diberikan tegangan sebesar 0V maka led akan mati dan buzzer tidak berbunyi serta tegangan pada relay sebesar 0,12V

4.6 Pengujian secara keseluruhan dengan perangkat Lunak

Data pengujian secara keseluruhan dapat dilihat dalam tabel 9. berikut ini. Pengujian ini dilakukan dengan cara memberikan masukan pada sensor infra merah (dengan cara menghalangi sinar infra merah) pada jam tertentu kemudian melihat respon mikrokontroller terhadap motor stepper. Pengujian juga dilakukan terhadap tombol S1 = Set jam,S2 = Set menit, S3 = Buka pintu dan S4 = Darurat.

Table 4.4 Pengujian keseluruhan

Jam	tombol				sensor	Reaksi Pada Motor Stepper	Seven segment		Buzzer
	S1	S2	S3	S4			Jam	Menit	
00.00	-	-	-	-	Terhalangi	Motor tidak berputar	-	-	-
08.00	-	-	-	-	Terhalangi	Motor berputar	-	-	-
20.59	-	-	-	-	Terhalangi	Motor berputar	-	-	-
21.00	-	-	-	-	Terhalangi	Motor tidak berputar	-	-	-
07.59	-	-	-	-	Terhalangi	Motor tidak berputar	-	-	-
08.00	-	-	S3	-	-	Motor berputar	-	-	-
20.59	-	-	S3	-	-	Motor berputar	-	-	-
21.00	-	-	S3	-	-	Motor berputar	-	-	-
05.59	-	-	S3	-	-	Motor berputar	-	-	-
06.00	-	-	-	S4	-	Motor berputar	-	-	Berbunyi
20.59	-	-	-	S4	-	Motor berputar	-	-	Berbunyi
21.00	-	-	-	S4	-	Motor berputar	-	-	Berbunyi
05.59	-	-	-	S4	-	Motor berputar	-	-	Berbunyi
-	S1	-	-	-	-	-	Bertambah 1	-	-
-	-	S2	-	-	-	-	-	Bertambah 1	-

Berdasarkan table diatas, saat catu daya di aktifkan, jam digital langsung running ditandai dengan led detik yang menyala secara bergantian menghitung detik dan tampilan seven segment menunjukkan waktu pukul 00.00. Apabila ada objek yang menghalangi sensor maka pintu akan tetap diam dan dalam keadaan menutup karena sensor belum aktif bekerja. Bila waktu pada jam di-set menjadi pukul 08.00, maka sensor aktif bekerja dan pintu akan terbuka pada saat ada objek yang menghalangi sensor tersebut, dan kemudian akan kembali menutup setelah selang waktu sekitar 8 detik. Pada saat menutup jika ada objek yang menghalangi sensor infra merah maka pintu akan kembali terbuka dan segera menutup kembali

setelah putaran pada motor stepper telah habis tanpa ada delay waktu. Setelah waktu menunjukkan pukul 21.00 maka sensor akan kembali pasif dan tidak akan merespon masukan dari infra merah.

Jika tombol set jam dan menit ditekan maka baik tampilan pada jam maupun menit akan bertambah Satu, kemudian jika tombol buka pintu ditekan maka pintu akan terbuka dan akan menutup kembali setelah 8 detik. Pada tombol buka pintu ini dapat bekerja tidak tergantung pada waktu sama dengan tombol pintu darurat jika ditekan maka pintu akan terbuka terus-menerus sekaligus mengaktifkan buzzer. Buzzer akan terus berbunyi sampai tombol darurat atau reset ditekan untuk mengembalikan keadaan seperti semula.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah selesainya tahap-tahap analisa dan pengujian sistem, maka dapat diambil kesimpulan:

1. Berdasarkan pengujian, peralatan ini dapat bekerja dengan baik sesuai dengan yang di inginkan.
2. Motor stepper dapat bekerja dengan baik Menarik beban pintu dan akan berhenti sejenak dengan waktu 8s
3. Setelah tombol darurat ditekan maka pintu akan terbuka dengan diikuti suara buzzer dan akan menutup kembali jika tombol darurat ditekan kembali.
4. Waktu kerja sensor dimulai pukul 08.00 sampai dengan pukul 21.00.

5.2 Saran

1. Alat pintu otomatis ini dapat dikembangkan dengan mengeset jam dan menit sesuai dengan yang diinginkan agar tidak terbatas dengan jam dan menit yang telah diprogram pada mikrokontroller. Selain itu tampilan dapat diganti dengan tampilan LCD dengan menambah IC RTC agar dapat menampilkan kalender.
2. Untuk pengembangan alat rangkaian sensor dapat diganti dengan sensor pendeteksi panas tubuh manusia atau dapat juga di ganti dengan sensor pendeteksi gerak.

DAFTAR PUSTAKA

1. Albert Paul Malvino, Ph. D, Prinsip – prinsip Elektronika, Edisi ketiga, Jilid 1, Penerbit Erlangga, Jakarta 1992.
2. Andi Nalwan, Paulus, Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrokontroller AT89C51, penerbit Elek Media Komputindo, Jakarta 2003.
3. Eko Putra, Agfianto, Belajar Mikrokontroller AT89C51/52/55 Teori dan Aplikasi, Edisi Pertama, Penerbit Gava Media, Yogyakarta 2002.
4. KF Ibrahim, Ir. P. Insap santoso, Msc, Teknik Digital, Edisi Pertama, Penerbit Andi offset, Yogyakarta 1996.
5. [Http://www.atmel.com](http://www.atmel.com)