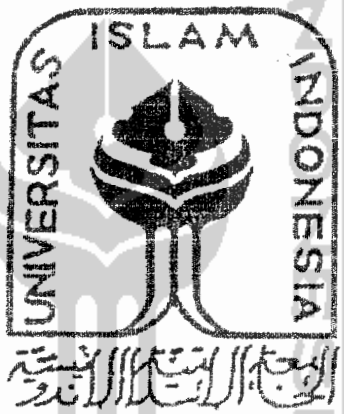
SISTEM PRESENSI DAN PENGAMAN PINTU MENGGUNAKAN *PASSWORD* BERBASIS PC

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat

Untuk Melaksanakan Tugas Akhir Pada Teknik Elektro

Fakultas Teknologi Industri Universitas Islam Yogyakarta



Disusun oleh :

Nama : Efri Sandey Hutapea

No.Mahasiswa : 98 524 025

TEKNIK ELEKTRO

FAKULTAS TEKNOLOGI INDUSTRI

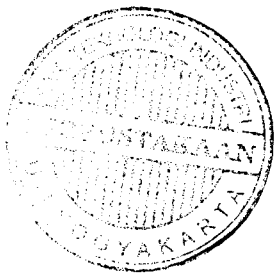
UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

2007

LEMBAR PENGESAHAN PEMBIMBING

SISTEM PRESENSI DAN PENGAMAN PINTU MENGGUNAKAN *PASSWORD* BERBASIS PC



Disusun Oleh

Nama : Efri Sandey Hutapea

No. Mahasiswa : 98 524 025

Yogyakarta, / /2007

Pembimbing I	Pembimbing II
Tito Yuwono, ST, MSc	Medilla Kusriyanto, ST

LEMBAR PENGESAHAN PENGUJI

SISTEM PRESENSI DAN PENGAMAN PINTU

MENGGUNAKAN *PASSWORD* BERBASIS PC

TUGAS AKHIR

Disusun Oleh :

Nama : Efri Sandey Hutapea

No. Mahasiswa : 98 524 025

Telah Dipertahankan Di Depan Sidang Penguji Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Pada Teknik Elektro Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, / /2007

Tim penguji,

Tito Yuwono,ST,MSc
Ketua

Medilla Kusriyanto,ST
Anggota I

Ir. H. Suyamto
Anggota II

Mengetahui,
Ketua Jurusan Teknik Elektro
Fakultas Teknologi Industri Universitas Islam Indonesia

Tito Yuwono,ST,MSc

HALAMAN PERSEMBAHAN

Kutujukan tugas akhir ini untuk :

* Allah SWT serta Nabi Muhammad Saw dan para pengikutnya
* Kupersembahkan untuk kedua orangtuaku yang telah memberikan seluruh apa yang dimilikinya demi masa depanku
* Ayahanda tercinta, atas bimbingan, keteladanan pengorbanan, kesabaran dan doa, ayahanda adalah semangat dalam hidupku.
* Ibunda tersayang, wujud kasih sayangmu, kesabaran dan keteladanan serta keselarasan hidup yang telah ibunda tunjukkan telah mendewasakanku
* Adik-adikku Rio dan Rini yang kusayangi, terima kasih atas dukungan dan motivasinya semoga kalian menjadi orang yang berguna bagi bangsa dan agama.
* Penjaga hatiku, Wiwid tercinta yang selalu meredakan emosiku dan membuat hidupku lebih berarti dan selalu membangkitkan semangatku. Terimakasih atas semua pengorbananya
* Sahabat-sahabatku yang tidak bisa Kusebutkan satu persatu terima kasih banyak atas dukungan dan bantuannya selama ini.
* Maju terus Teknik Elektro UII.

MOTTO

- Allah akan mengangkat orang-orang yang beriman dari golonganmu dan juga orang-orang yang dikarunia ilmu pengetahuan hingga beberapa derajat

(Q.S. Al Mujaadilah 58 : 11)

- Sesungguhnya sesudah kesulitan itu ada kemudahan

(Q.S. Asy Syarh 5)

- Sebaik-baik manusia ialah orang yang banyak manfaatnya (kebaikannya) kepada manusia lainnya.

(H.R Qadla'ie dari Jabir)

- Maka apabila kamu telah selesai (dari suatu urusan) kerjakanlah dengan sungguh-sungguh (urusan) yang lain dan hanya kepada Tuhanmulah hendaknya kamu berharap.

(Q.S. Al Insyirah [94] : 7 - 8)

- Allah tidak akan membebani seseorang kecuali sesuai kesanggupannya. Ia mendapat pahala (dari kebaikan) yang dikerjakannya dan ia mendapatkan siksa (dari kejahatan) yang dikerjakannya.

(Q.S. Al- Baqarah [2] : 286)

- Hai orang-orang yang beriman jadikanlah sabar dan sholat sebagai penolong kalian. Sesungguhnya Allah berserta orang yang sabar

(Q.S. Al- Baqarah [2] : 153)

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Alhamdulillah puji syukur marilah senantiasa kita panjatkan kehadirat Allah SWT karenanya hanya berkat rahmat dan karunianyalah akhirnya penulis dapat menyelesaikan laporan Tugas Akhir ini sebagai syarat untuk meraih gelar sarjana teknik di jurusan teknik elektro Universitas Islam Indonesia. Tak lupa sholawat serta salam senantiasa tercurah pada junjungan hamba nabi besar Muhammad Saw.

Selama pembuatan alat dan penyusunan laporan Tugas Akhir ini penulis tidak lepas dari hambatan dan keterbatasan, namun berkat bantuan dan dukungan dari berbagai pihak masalah tersebut dapat diatasi. Oleh karena itu sudah patutnya mengucapkan terima kasih kepada :

1. Bapak Fathul Wahid, ST, MSc., selaku dekan Fakultas Teknologi Industri Universitas Islam Indonesia
2. Bapak Tito Yuwono, ST, MSc., selaku Ketua Jurusan Teknik Elektro Fakultas Teknologi Industri Universitas Islam Indonesia
3. Bapak Tito Yuwono, ST, MSc., selaku Dosen Pembimbing Tugas Akhir I
4. Bapak Medilla Kusriyanto, ST., selaku Dosen Pembimbing Tugas Akhir II
5. Kedua Orangtuaku, Papa, Mama dan Adik-adik Ku tercinta yang selalu mendoakan, memberikan dukungan dan semangat selama ini.

6. Wiwid yang tercinta terima kasih telah sabar menunggu dan selalu mendoakan dan memberikan semangat selama ini.
7. Segenap Dosen Fakultas Teknologi Universitas Islam Indonesia umumnya dan khususnya Dosen Jurusan Teknik Elektro yang telah membimbing dan memberikan ilmunya kepada Saya selama ini.
8. Teman-temanku Teknik Elektro, dan anak-anak kost terima kasih atas dukungan doa dan bantuannya
9. Semua teman-teman yang belum saya sebutkan yang telah membantu dan memberikan sumbangsihnya dalam penyelesaian laporan Tugas Akhir ini.

Semoga amal baik yang telah diberikan mendapatkan balasan dari Allah SWT amin. Tak lupa penulis menyadari dalam tugas akhir ini masih banyak kekurangan dan kesalahan oleh karena itu penulis mohon maaf yang sebesar-sebesarnya atas kekhilafan karena kesempurnaan hanya milik Allah SWT.

Tak lupa kritik dan saran yang membangun sangat penulis harapkan semoga laporan ini dapat bermanfaat. Amin.

Yogyakarta / / 2007

Efri Sandey Hutapea

DAFTAR ISI

HALAMAN JUDUL i

LEMBAR PENGESAHAN PEMBIMBING ii

LEMBAR PENGESAHAN PENGUJI iii

LEMBAR PERSEMBAHAN iv

HALAMAN MOTTO v

KATA PENGANTAR vi

DAFTAR ISI viii

DAFTAR GAMBAR xi

DAFTAR TABEL xii

DAFTAR LAMPIRAN xiii

ABSTRAK xiv

BAB I PENDAHULUAN 1

1.1 Latar Belakang Masalah 1

1.2 Rumusan Masalah 2

1.3 Batasan Masalah 2

1.4 Tujuan Penelitian 3

1.5 Manfaat Penelitian 3

1.6 Sistematika Penulisan 3

BAB II LANDASAN TEORI 6

2.1 Dasar Komputer 6

2.1.1 *CPU* 6

2.1.2 Memori (Pengingat) 6

2.1.3 *Addres Bus* (Bus Alamat) 7

2.1.4 *Data Bus* (Bus Data) 7

2.1.5 *Control Bus* (Bus Kendali) 8

2.2 Sistem Perangkat Lunak (*Software*) 8

2.3 Piranti Masukan/ Keluaran 8

2.3.1 Port Printer 8

2.3.2 Jenis-jenis Port 10

2.3.3 Pengalamatan 11

2.4 *Driver* 14

2.4.1 Resistor 15

2.4.2 Dioda 16

2.4.3 Transistor 17

2.4.4 Relai 18

2.5 *Keypad* 20

BAB III PERANCANGAN SISTEM 21

3.1 Gambaran Umum Sistem 21

3.2 Komputer 22

3.3 Inisialisasi Port Parallel 23
3.4 Pintu Simulasi 25
3.5 *Driver* 26
3.6 Perancangan *Software* 29
3.7 *Flow Chart* 30
3.8 Perancangan Database 32

BAB IV **PENGUJIAN DAN ANALISIS** 36
4.1 Pengujian *Driver* 36
4.2 Pengujian *Keypad* 37
4.2 Pengujian Perangkat Lunak 38

BAB V **PENUTUP** 42
5.1 Kesimpulan 42
5.2 Saran 42

DAFTAR PUSTAKA 44

LAMPIRAN 45

DAFTAR GAMBAR

Gambar 2.1. Port Printer 11

Gambar 2.2. Skema *Driver* 15

Gambar 2.3. Resistor 16

Gambar 2.4. Dioda 17

Gambar 2.5. Transistor 17

Gambar 2.6. Simbol Transistor PNP dan NPN 18

Gambar2.7. Sistem Mekanik Relai dan Simbol Relai 19

Gambar 2.8. *Keypad* 20

Gambar 3.1. Blok Diagram Sistem Presensi dan Pengamanan Pintu 21

Gambar 3.2. Kotak Simulasi dan CD-Rom Drive 24

Gambar 3.3. Skema *Driver* 25

Gambar 3.4. Skema *Keypad* 28

Gambar 3.5. Perancangan Tampilan Form Utama 29

Gambar 3.6. Perancangan Tampilan Form Input Data 29

Gambar 3.7. Diagram Alir Program 31

Gambar 3.8. Tabel Restruktur Paradox 32

Gambar 4.1. Program Ketika Dijalankan 38

Gambar 4.2. Tampilan Form Ketika Memasukkan Data 39

Gambar 4.3. Tampilan Ketika Keypad Ditekan 39

DAFTAR TABEL

Tabel 2.1. Alamat Port Printer 11

Tabel 2.2. Register Port Printer 11

Tabel 2.3. Definisi Bit Port Data 13

Tabel 2.4. Definisi Bit Port Status 13

Tabel 2.5. Definisi Bit Port Kontrol 14

Tabel 3.1. Definisi Penggunaan Port Data 22

Tabel 3.2. Definisi Penggunaan Port Status 23

Tabel 3.3. Definisi Penggunaan Port Kontrol 23

Tabel 3.4. Definisi Pengaturan Port Kontrol 24

Tabel 3.5. Kebutuhan Komponen Program dan Perubahan Property 33

Tabel 4.1. Posisi Pintu Terbuka dan Tertutup 36

DAFTAR LAMPIRAN

Lampiran A Skematik Rangkaian

Lampiran B *Listing Program*

ABSTRAK

Komputer atau sering disebut PC (*Personal Computer*) hampir semua orang memilikinya. Komputer bisa digunakan untuk keperluan pengontrolan melalui *parallel port*. Misalnya ingin mengatur lampu mana yang akan hidup dengan menekan tombol dari keyboard computer. Ataupun mengatur kapan lampu kamar akan menyala atau mati. Dalam tugas akhir ini, akan dibuat sebuah alat pengontrolan dengan menggunakan *parallel port* ini. Adapun tugas akhir ini berjudul *Sistem Presensi dan Pengaman Pintu Menggunakan Password Berbasis PC*. Alat ini dapat digunakan untuk menghitung kehadiran setiap orang yang masuk melewati pintu dengan cara memasukkan *password* terlebih dahulu. Untuk memasukkan *password*, sistem ini menggunakan tombol *keypad* tersendiri yang dihubungkan melalui *port parallel*. Komputer bertindak sebagai penerima instruksi dari *keypad* dan penyimpan data presensi. Hasil penelitian menunjukkan perancangan sistem presensi dan pengaman pintu menggunakan password berbasis PC ini dapat mendaya gunakan *port parallel* yang sering tidak terpakai. Karakter-karakter *password* yang dimasukkan hanya berupa karakter angka, sedangkan untuk hak akses melewati pintu harus dilakukan pengisian data terlebih dahulu menggunakan komputer. Sistem ini tidak mempunyai tampilan yang dapat dipantau oleh pengguna ketika memasukkan *password*. Untuk pengembangan lebih lanjut untuk tampilan ke pengguna dapat menggunakan tampilan berupa LCD ataupun *Seven Segmen* dengan menambah *interface* PPI 8255.

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Komputer merupakan suatu alat yang dapat digunakan sebagai pusat pengendali. Komputer mempunyai banyak jenis antarmuka (*interface*) untuk berhubungan dengan peralatan luar tambahan (*external device*). Antarmuka itu diantaranya parallel port (LPT), serial port (COM), universal serial bus(USB). Apabila antarmuka ini dapat didayagunakan untuk mengendalikan suatu peralatan, maka akan memperoleh keuntungan yang lebih baik. Teknologi komputer itu sendiri dapat diberdayakan melalui peningkatan unjuk kerja perangkat keras (*hardware*) dan tidak terlepas pula dari perangkat lunak (*software*) karena satu sama lain saling terkait dan saling melengkapi.

Dengan mempertimbangkan beberapa hal tersebut, maka penulis bermaksud untuk memanfaatkan port parallel (LPT) untuk "SISTEM PRESENSI DAN PENGAMAN PINTU MENGGUNAKAN *PASSWORD* BERBASIS PC". Dengan adanya alat ini akan lebih menjamin keamanan sebuah ruangan dengan menggunakan sebuah *password* sekaligus bisa digunakan sebagai sistem presensi.

1.2 Perumusan Masalah

Dengan semakin cepatnya perkembangan teknologi, banyak peralatan luar (*external device*) beralih menggunakan antarmuka universal serial bus (USB) untuk berhubungan dengan komputer. Namun muncul permasalahan karena

penggunaan port parallel untuk berhubungan dengan peralatan luar semakin jarang digunakan ataupun tidak pernah terpakai. Bagaimana cara mendayagunakan port parallel yang tidak terpakai ini? Bagaimana cara merancang alat untuk dapat berhubungan dengan port parallel? Bagaimana cara merancang program untuk mengendalikan alat?

1.3 Batasan Masalah

Pada perancangan sistem presensi dan pengaman pintu menggunakan *password* berbasis PC, penulis bermaksud membuat sebuah alat sederhana, yang dapat diterapkan pada pintu ruangan yang digunakan untuk membuka serta menutup pintu secara otomatis menggunakan *password*.

Pada batasan masalah perancangan sistem presensi dan pengaman pintu menggunakan *password* berbasis PC, dibatasi pada bahasan yang mencakup :

1. Perangkat keras (*hardware*) yang digunakan :
   a. Sistem antar muka menggunakan parallel *port* sebagai terminal masukan dan keluaran data.
   b. Menggunakan *keypad* yang berfungsi untuk memasukkan karakter *password*.
   c. Menggunakan motor DC yang berfungsi untuk menggerakkan pintu secara otomatis.
   d. Menggunakan relai yang berfungsi sebagai pengatur catu daya ke motor DC.

2. Perangkat lunak (*software*) yang digunakan :
   a. Pembuatan program atau (software) yang akan digunakan untuk memantau dan menjalankan alat, yaitu menggunakan bahasa pemrograman Delphi.
   b. Aplikasi pengendali dijalankan pada system operasi *Windows XP*.

1.4 Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini adalah merancang *keypad* yang digunakan sebagai tombol kunci (*password*) dan *driver* sebagai pengendali pintu secara otomatis yang akan bergerak berdasarkan masukan dari *keypad*, agar dalam penggunaannya nanti menjadi lebih efisien dan tidak tergantung pada papan *keyboard*.

1.5 Manfaat Penelitian

Manfaat penelitian dari pembuatan tugas akhir adalah sebagai berikut :

a. Dengan adanya sistem presensi dan pengaman pintu berbasis komputer akan membuat akses ke ruangan menjadi lebih aman dari pihak-pihak yang tidak berkepentingan.
b. Dapat diterapkan dalam institusi tertentu yang membutuhkan sistem presensi dan pengaman pintu menggunakan komputer.

1.6 Sistematika Penulisan

Sistematika penulisan dari laporan ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisikan penjelasan latar belakang masalah, menjelaskan permasalahan yang dihadapi dan penyelesaian yang diambil, maksud dan tujuan, pembatasan masalah akan menjelaskan tentang rancangan sistem yang akan dibuat, serta menjelaskan tentang sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisikan penjelasan-penjelasan dasar mengenai sistem kerja dari komponen pokok yang digunakan, yang berguna untuk mendukung pembuatan sistem presensi dan pengaman pintu berbasis PC.

BAB III PERANCANGAN SISTEM

Bagian ini menjelaskan metode-metode perancangan yang digunakan, cara mengimplementasikan rancangan dan batasan dan hambatan yang ditemui selama proses perancangan dan implementasi sistem.

BAB IV ANALISIS DAN PEMBAHASAN

Membahas hasil-hasil pengujian dan pengamatan yang diperoleh dari uji coba perangkat keras (*hardware*) dan perangkat lunak (*software*).

BAB V PENUTUP

Memuat kesimpulan yang didapat dari hasil pengujian dan analisa serta implementasi sistem dan juga kelemahan dan keterbatasan yang ditemui, serta saran-saran yang perlu dilakukan untuk mengoptimalkan sistem.

BAB II

LANDASAN TEORI

2.1 Dasar Komputer

Secara garis besar komputer terdiri dari tiga bagian yaitu *Central Processing Unit* (CPU) bagian pemroses utama, memori (pengingat) serta piranti masukan dan keluaran. Untuk menghubungkan masing-masing bagian dipergunakan penghubung parallel atau yang dikenal istilah *Bus*.

2.1.1 CPU

CPU (*Central Processing Unit*) merupakan pusat dari pemroses utama data. CPU membaca perintah-perintah yang terdapat pada pengingat kemudian memberikan reaksi-reaksi instruksi yang dibacanya. CPU tersusun dari *Arithmetic and Logical Unit* (ALU), bagian kendali serta *register-register*.

Dengan adanya ALU maka instruksi-instruksi seperti penjumlahan, pengurangan, perkalian serta instruksi-instruksi logika seperti *OR, XOR, AND* dan *INVERTER* dapat dilaksanakan. Sedangkan bagian kendali digunakan untuk mengatur urutan penyerempakan kerja bagian-bagian lain.

2.1.2 Memori (Pengingat)

Memori mempunyai dua kegunaan yaitu sebagai penyimpan kode biner urutan instruksi pada saat menuliskan program dan untuk menyimpan data hasil kerja CPU atau data yang dimasukkan dari luar. Ada dua macam memori, yang pertama disebut ROM

(*Read Only Memory*) yaitu memori yang hanya dapat dibaca. Memori ini bisaanya sudah terisi data atau kode program. CPU hanya bisa membaca isi dari memori jenis ROM. Disamping itu sifat memori ROM meskipun catu daya dimatikan, data yang tersimpan didalamnya tidak akan hilang.

Memori yang kedua adalah memori baca tulis atau yang disebut RAM (*Random Access Memory*), yaitu memori yang isinya dapat diubah-ubah dan dibaca oleh CPU. Sifat lain dari memori jenis RAM adalah jika catu daya dimatikan, data yang tersimpan didalamnya akan hilang.

2.1.3 *Address Bus* (Bus Alamat)

Untuk menentukan lokasi yang menjadi tujuan digunakan *address bus* yang berupa saluran pararel. *Address Bus* hanya menyambungkan isyarat dalam satu arah, yaitu dari CPU ke bagian-bagian lain, seperti ke bagian memori dan peralatan masukan atau keluaran.

2.1.4 *Data Bus* (Bus *Data*)

Data bus digunakan oleh komputer untuk mengadakan pertukaran data antara CPU kebagian lainnya. Aliran data bus terjadi dua arah (*bi-directional*). Bus tersebut membawa sinyal khusus yang memberitahu peralatan untuk terhubung ke jalur yang sedang dilewati data.

2.1.5 *Control Bus* (Bus Kendali)

Control Bus digunakan untuk mengkoordinasikan kerja dari masing-masing bagian atau dengan kata lain *Control Bus* ini digunakan oleh komputer untuk Tanya jawab antar bagian dengan tujuan untuk pengendalian kerja. Bus ini mempunyai aliran isyarat selalu ke arah CPU, yaitu dipergunakan untuk menayakan atau mendeteksi adanya penyela.

2.2 Sistem Perangkat Lunak (*Software*)

Pembuatan *Software* "SISTEM PRESENSI DAN PENGAMAN PINTU MENGGUNAKAN *PASSWORD* BERBASIS PC" ini menggunakan program Borland Delphi 5. Untuk penyimpanan data presensi dan data *password* digunakan *Database* Paradox.

2.3 Piranti Masukan dan Keluaran

Untuk menghubungkan dengan *user* atau pemakai, komputer dilengkapi dengan piranti masukan/ keluaran. Piranti ini berupa *printer, keyboard, disk drive, monitor* dan sebagainya. Piranti-piranti tersebut dapat dihubungkan dengan komputer melalui unti antarmuka masukan/ keluaran (*I/O Interface Unit*) atau juga disebut *I/O Port*.

2.3.1 LPT (Line Printer)

LPT (*Line PrinTer*) adalah istilah untuk *port* yang biasanya dihubungkan ke printer. Komunikasi yang digunakan oleh LPT adalah komunikasi paralel. Komunikasi paralel berjalan lebih cepat dari komunikasi serial karena jumlah bit yang dikirimkan

bersamaan dalam suatu waktu oleh komunikasi paralel lebih banyak daripada komunikasi serial.

Jumlah kawat yang digunakan oleh komunikasi paralel tentunya juga lebih banyak daripada komunikasi serial. Meskipun jumlah pin port COM2 pada komputer umumnya berjumlah sama dengan jumlah pin port LPT, tidak semua pin dalam port COM2 digunakan.

Dalam dunia komputer, port adalah jalur-jalur dimana jalur ini akan dilewati intruksi atau perintah dimana *microprocessor* atau CPU menggunakanya untuk memindahkan data dari atau ke piranti lain. Penggunaan umum port adalah untuk berkomunikasi dengan *printer*, *modem*, keyboard, dan *display*. Kebanyakan port-port komputer adalah berupa kode digital, dimana tiap-tiap sinyal atau bit adalah berupa kode biner 0 atau 1. Port printer atau lebih dikenal dengan *parallel port* mentransfer beberapa bit secara bersamaan, sementara *serial port* menstranfer satu bit setiap saatnya. Port printer bisa ditemukan dalam setiap PC. Bersama dengan serial RS-232, port printer sangat berperan dalam komunikasi PC. Pada PC yang lebih baru, kita bisa dapati port-port lain seperti SCSI, USB, dan IrDA, tapi port printer tetap popular karena kemampuan dan fleksibilitasnya.

Port printer PC original memiliki 8 bit output, 5 bit input, dan 4 bit dua – arah ini cukup untuk berkomunikasi dengan berbagai jenis *peripheral*. Pada PC yang baru, 8 bit output juga bisa bertindak sebagai input atau dengan kata lain bersifat dua-arah. hal ini penting untuk berkomunikasi yang lebih cepat dengan *scanner*, *drive*, dan piranti lain yang mengirim data ke PC. Port printer pada awalnya memang didesain sebagai port untuk printer, dan banyak nama–nama pinnya seperti *PaperEnd*, *Error*, dan lain–lain.

mencerminkan penggunaan untuk printer.

2.3.2 Jenis-jenis Port

Seiring dengan perkembangan desain PC, beberapa perusahaan memperkenalkan port printer versi yang lebih baik. Di bawah ini adalah ringkasan jenis-jenis yang tersedia antara lain sebagai berikut :

a) Original (SPP/*Standar Parallel Port*).

Port printer ini terdapat dalam IBM PC original. Nama lain yang digunakan adalah AT-type atau ISA-compatible. SPP menggunkan mode *Nibble* yang mentransfer 4 bit setiap saatnya.

b) Tipe PS/2 (Dua Arah Sederhana).

Perbaikan awal port printer adalah port data dua arah yang diperkenalkan IBM lewat mode PS/2. Port dua arah meng-*enable peripheral* mentransfer 8 bit setiap saat ke PC. Istilah PS/2 merujuk pada setiap port yang mempunyai port data dua arah tapi tidak mendukung mode EPP dan ECP.

c) EPP (*Enhanced* Parallel *Port*).

Pertama kali dikembangkan oleh chip intel, pabrik PC Zenith, dan pembuat pemuat produk-produk port printer jaringan Xircom. Sebagai mana tipe PS/2, port data bersifat dua arah. EPP bisa memutar arah secara langsung sehingga bisa sangat efisien ketika digunakan dengan piranti yang mentransfer data dalam dua arah.

d) ECP (*Extended* Capabilities Port).

Pertama kali diperkenalkan oleh Hewlett Packard. Bersifat dua arah, dan bisa mentransfer data pada kecepatan bus ISA. ECP mempunyai *buffer* dan support untuk transfer DMA (*Direct Memory Access*) dan kompresi data.

2.3.3 Pengalamatan

Ada beberapa nama bagi port printer. Port printer yang bukan di video (*monochrome*) adapter diberi nama LPT1 dan LPT2, masing-masing mempunyai alamat sendiri-sendiri seperti pada tabel 2.1 dibawah ini.

Tabel 2.1 Alamat Port Printer

Alamat Port Printer	
Parallel port	Alamat Dasar
LPT0	$3BC
LPT1	$378
LPT2	$278

LPT 0 tidak support untuk alamat ECP. Tanda dollar ($) didepan alamat menandakan tanda bilangan heksadesimal dalam bahasa pemrograman.

Tabel 2.2 Register Port Printer

Register Port Printer	
Nama Register	Alamat
Register Data	Dasar + 0
Register Status	Dasar + 1
Register Kontrol	Dasar + 2

Pada tabel 2.2 alamat register ditentukan berdasarkan alamat dasarnya. Sebagai contoh jika port printer yang digunakan ialah LPT1 yang alamat dasarnya adalah $378, maka port data, port status, dan port control berturut-turut adalah $378+0, $378+1, dan $378+2 atau $378, $379, dan $37A. Register data sering disebut juga port data. Begtu

juga port register status dan register control sering sering disebut port status dan port control.

1. Register data/data port

   Port data (alamat dasar +0), port ini terletak pada pin (2-9) mempunyai kemampuan membaca data dan menulis data 8 bit.

2. Register status/ status port

   Sering disebut port status (alamat dasar +1) hanya merupakan read only port. Setiap data yang ditulis pada port ini akan diabaikan. Status port terdiri dari 5 input jalur-jalur data (pin 10,11,12,13,dan 15), sebuah IRQ register status dan dua bit yang tidak dipakai.

3. Register control/ port control

   Port control (alamat dasar +2) pada kemampuan normal atau terbalik dapat membaca, mengeluarkan data, dan menuliskanya. Port control ini terdiri dari 4 dari 5 bit yang mempunyai output dikonektor. Bit ke 5 hanya untuk dirinya sendiri, tidak keluar dikonektor.

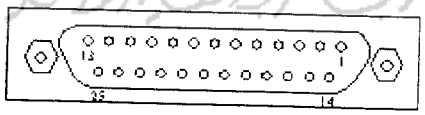
Tabel 2.3 Definisi Bit Port Data

Bit	No pin	i	SPP	EPP/ECP	Sifat
7 (MSB)	9	D7	S	Dua Arah	Normal
6	8	D6	S	Dua Arah	Normal
5	7	D5	S	Dua Arah	Normal
4	6	D4	S	Daa Arah	Normal
3	5	D3	S	Dua Arah	Normal
2	4	D2	S	Dua Arah	Normal
1	3	D1	S	Dua Arah	Normal
0 (LSB)	2	D0	S	Dua Arah	Normal

Tabel 2.4 Definisi Bit Port Status

Bit	No pin	Fungsi	SPP	EPP/ECP	Sifat
7	11	Busy	Baca	Baca	Terbalik
6	10	Acknowledge	Baca	Baca	Normal
5	12	Paper status	Baca	Baca	Normal
4	13	Selection status	Baca	Baca	Normal
3	15	Error status	Baca	Baca	Normal
2		Tidak dipakai	-	-	
1		Tidak dipakai	-	-	
0 (LSB)		Tidak dipakai	-	-	

Pada kolom sifat terbalik dan normal maksudnya adalah, bersifat terbalik, jika port sedang berlogika *true*, maka pada kaki konektor DB 25 yang bersangkutan berlogika *false*, demikian pula sebaliknya. Pada port control, hanya 4 dari 5 bit yang mempunyai output dikonektor. Bit ke 5 hanya untuk dirinya sendiri, tidak keluar dikonektor. Lalu sisanya pin 18 sampai 25 berfungsi sebagai *ground*. Wujud fisik dari port printer berupa konektor DB-25 tipe (*female*) atau betina seperti yang terlihat pada gambar 2.1 konektor ini bisaanya terdapat pada belakang PC.



Gambar 2.1 Port Printer

Tabel 2.5 Definisi Bit Port Control

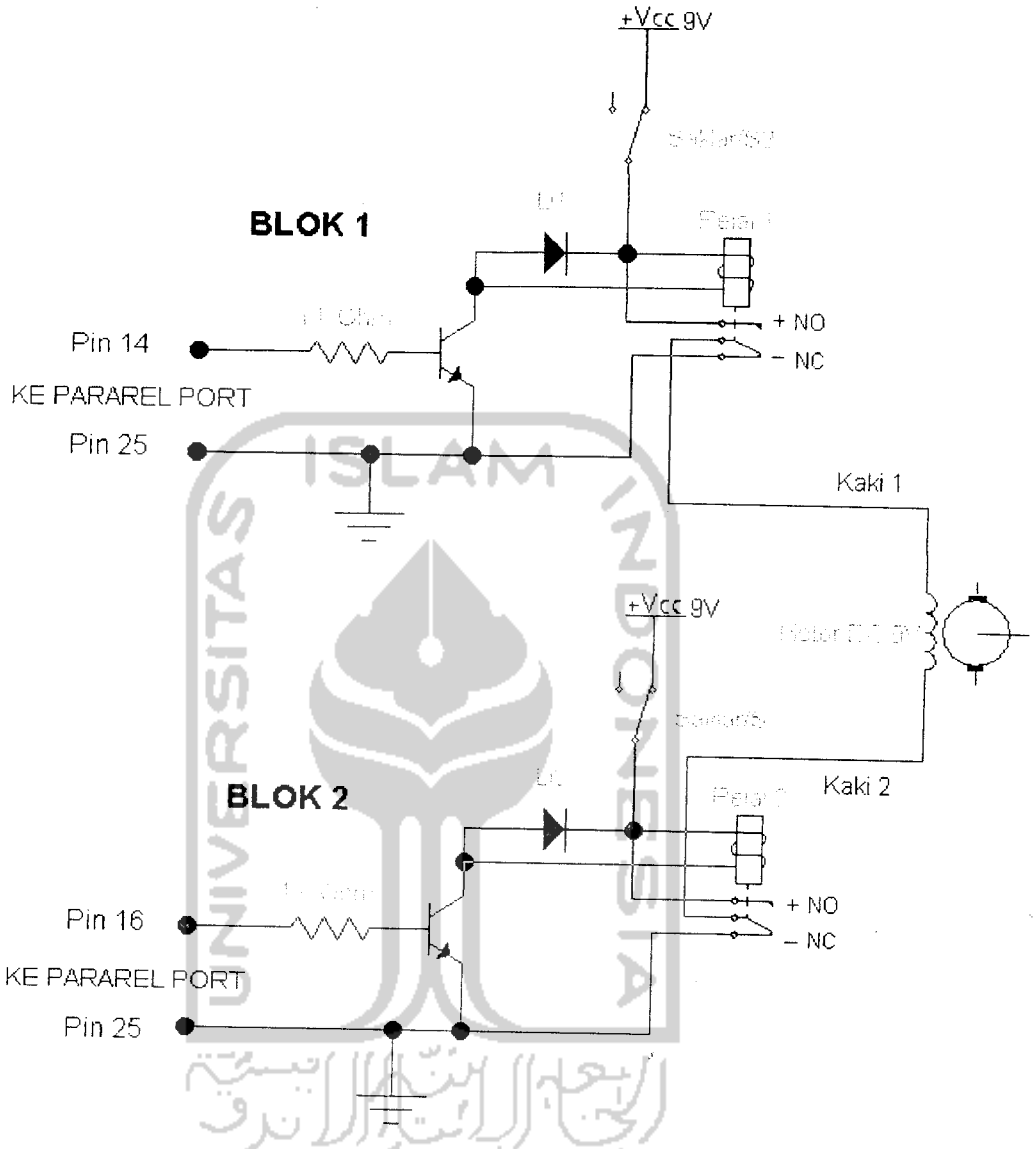
Bit	No pin	Fungsi	SPP	EPP/ECP	Sifat
7 (MSB)		Tidak dipakai	-	-	
6		Tidak dipakai	-	-	
5		Tidak dipakai	-	-	
4		Interrupt Control	Dua Arah	Tulis	Terbalik
3	17	Select	Dua Arah	Tulis	Terbalik
2	16	Inisialize	Dua Arah	Tulis	Nomal
1	14	Auto feed	Dua Arah	Tulis	Terbalik
0 (LSB)	1	Strobe	Dua Arah	Tulis	Terbalik

2.4 *Driver*

Driver adalah sebagai pengendali pergerakan pintu ke arah membuka atau menutup. Gerakkan pintu membuka atau menutup akan digerakkan oleh motor DC. Sedangkan pergerakkan motor DC ini ditentukan oleh logika-logika yang diterima masukan *driver* yang didapat dari keluaran port pararel komputer. Komponen-komponen *driver* ini terdiri dari transistor, dioda relai dan saklar.

Cara kerja *driver* untuk Blok1 dan Blok2 adalah sama, jika blok-blok ini diberi masukan tegangan maka tegangan ini akan dilewatkan melalui resistor, resistor ini berfungsi sebagai penurun tegangan, kemudian tegangan ini akan diumpankan ke kaki basis transistor. Karena pada kaki basis transistor diberi tegangan positif (5 volt) dan emitter mendapat tegangan negatif dengan kondisi seperti ini maka transistor akan bersifat maju (*forward*). Tegangan yang dilewatkan melalui kaki emitter akan diteruskan

menuju relai. Relai akan bergerak dan akan menghubungkan kaki motor DC menuju sumber tegangan.



Gambar 2.2 Skema *Driver*

2.4.1 Resistor

Resistor adalah komponen dasar elektronika yang digunakan untuk membatasi jumlah arus yang mengalir dalam satu rangkaian. Sesuai dengan namanya resistor bersifat

resistif dan umumnya terbuat dari bahan karbon. Dari hukum Ohm diketahui, resistansi berbanding terbalik dengan jumlah arus yang mengalir melaluinya. Satuan resistansi dari suatu resistor disebut Ohm atau dilambangkan dengan simbol Ω (*Omega*).

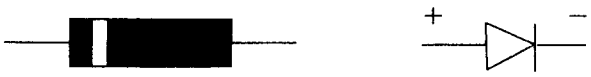
Tipe resistor yang umum adalah berbentuk tabung dengan dua kaki tembaga di kiri dan kanan. Pada badannya terdapat lingkaran membentuk gelang kode warna untuk memudahkan pemakai mengenali besar resistansi tanpa mengukur besarnya dengan Ohmmeter. Kode warna baku tersebut adalah standar manufaktur yang dikeluarkan oleh EIA (Electronic Industries Association).



Gambar 2.3 a)Resistor b)Simbol Resistor

2.4.2 Dioda

Dioda adalah komponen aktif yang terbuat dari bahan semikonduktor yang saling dipertemukan (junction) titik elektrodanya, bisa disebut dengan titik PN (Positif fan Negatif). Titik elektroda dari dioda ada dua yang satu disebut sebagai anoda (P) dan satunya lagi disebut katoda (N). Fungsi dari dioda antara lain sebagai penyearah, penyearah merupakan proses dimana arus atau tegangan bolak-balik diubah menjadi arus atau tegangan searah. Setiap peralatan listrik yang memberikan resistansi rendah ke arus menurut satu arah dan resistansi tinggi pada arah yang berlawananan dinamakan penyearahan. Pada dioda arus listrik hanya dapat mengalir dari kutub *Anoda* ke kutub *Katoda*, tetapi bila sebaliknya arus listrik tidak dapat mengalir.

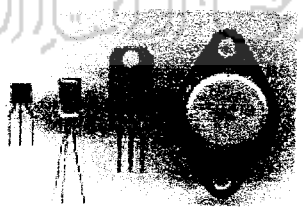


Gambar 2.4 a)Dioda b)Simbol Dioda

2.4.3 Transistor

Transistor merupakan salah satu komponen listrik semikonduktor yang dapat digunakan untuk perataan arus, menahan sebagian arus, menguatkan arus, membangkitkan frekunesi rendah maupun frekuensi tinggi. Transistor merupakan dioda dengan dua sambungan (*Junction*).

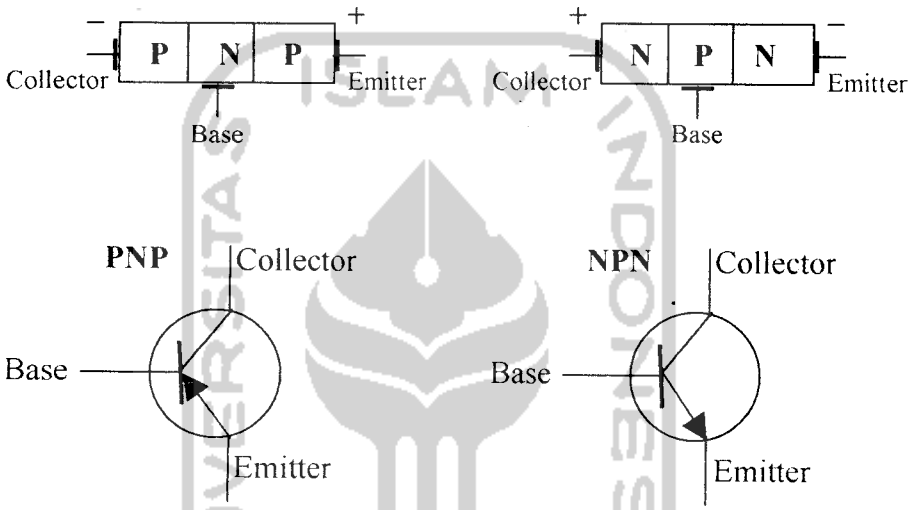
Sambungan ini membentuk transistor PNP maupun NPN. Transistor biasanya terdiri dari 3 buah kaki yang masing-masing kaki diberi nama kolektor (C atau K), basis (B) dan emitter (E). jika ada kaki keempat dinamakan kaki sasis. Basis selalu di tengah, diantara emitter dan kolektor.



Gambar 2.5 Transistor

Dalam bidang elektronika komponen transistor banyak sekali macamnya, diantaranya jenis transistor bipolar dan transistor jenis transistor efek medan (FET).

Bipolar adalah jenis transistor yang paling umum dan banyak dipergunakan dalam rangkaian. Disebut transistor bipolar karena struktur dan prinsip kerjanya tergantung dari perpindahan elektron di kutub mengisi kekurangan elektron (*hole*) di kutub positif. Umumnya diberi tanda tertentu untuk memudahkan penentuan salah satu kaki-kaki basis, kolektor dan emitter.

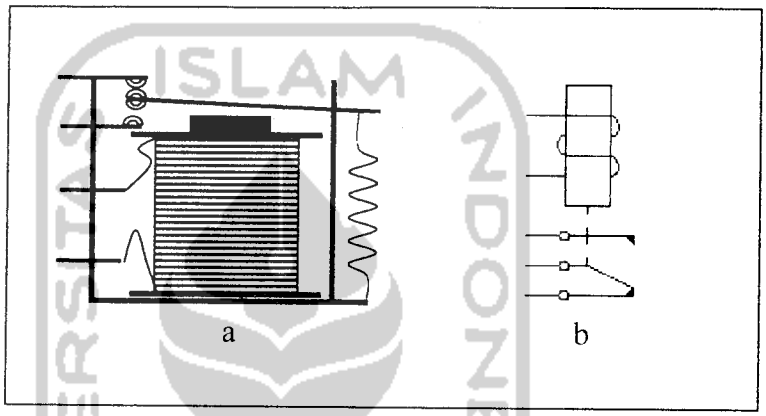


Gambar 2.6 Simbol Transistor PNP dan NPN

2.4.4 Relai

Relai merupakan komponen listrik yang dioperasikan dengan listrik yang secara mekanis mengendalikan penghubung rangkaian listrik. Relai adalah bagian yang penting dari banyak sistem pengendali. Relai dapat digunkan untuk kendali jarak jauh dan untuk pengendali jarak yang bersumber pada tegangan maupun arus tinggi dengan sinyal kendali tegangan dan arus rendah.

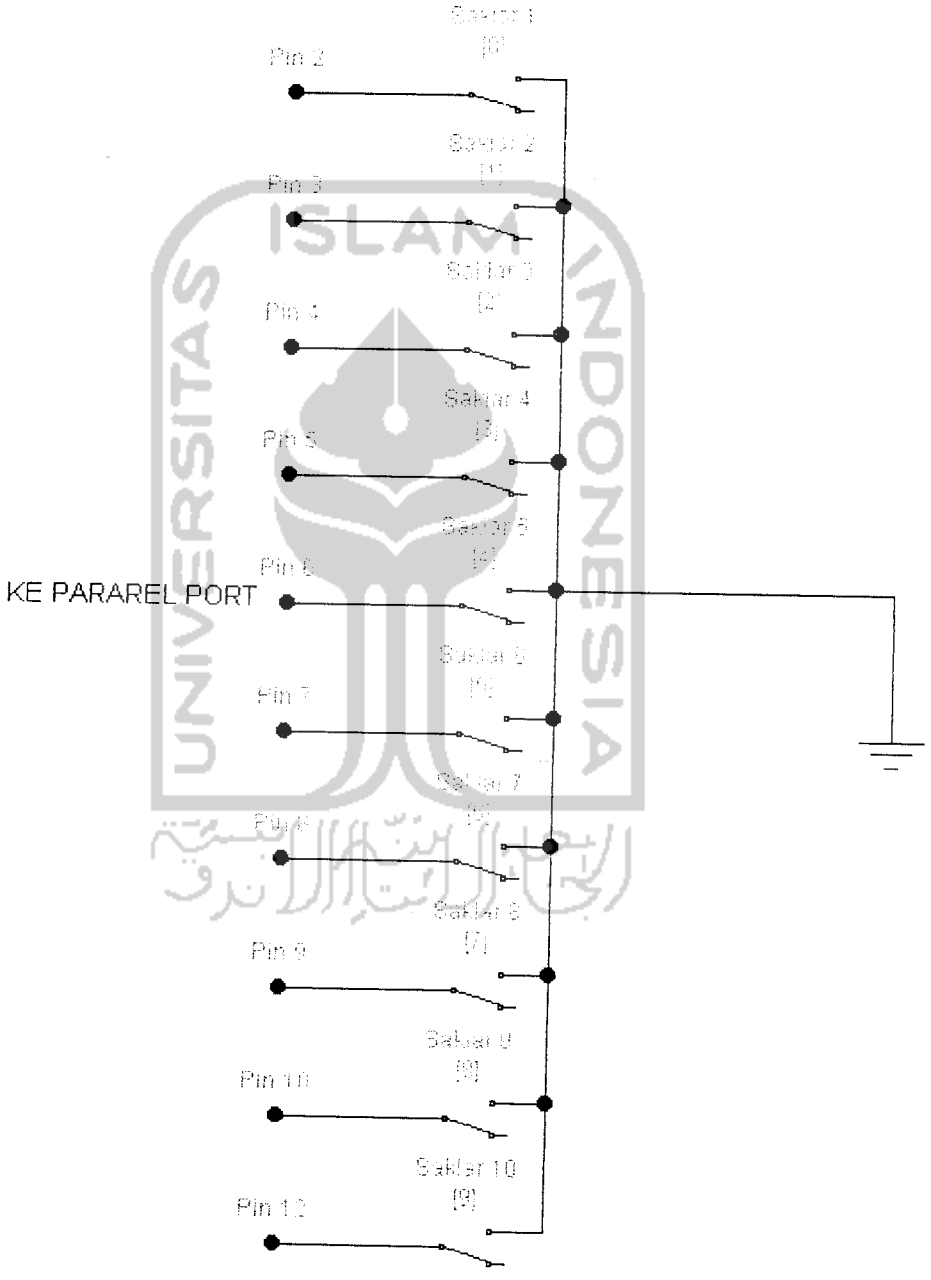
Cara kerja relai ini adalah jika arus dialirkan melalui lilitan relai, maka lilitan akan terinduksi dan menyebabkan inti besi akan bermagnet. Medan magnet akan menarik lengan armatur ke besi inti sehingga kontak akan terhubung. Berdasarkan penggunaanya hubungan kontak relai dibuat dalam 2 jenis hubungan yaitu : *Normally Open* (NO) atau normal buka dan *Normally Close* (NC) atau normal tutup.



Gambar 2.7 a)Sistem Mekanik Relai b)Simbol Relai

2.5 *Keypad*

Tombol-tombol *keypad* adalah tombol-tombol yang digunakan sebagai masukan *password*. Tombol-tombol ini selain digunakan sebagai masukan *password* juga berguna sebagai kunci pembuka pintu yang akan dicocokkan dengan isi *Database*.



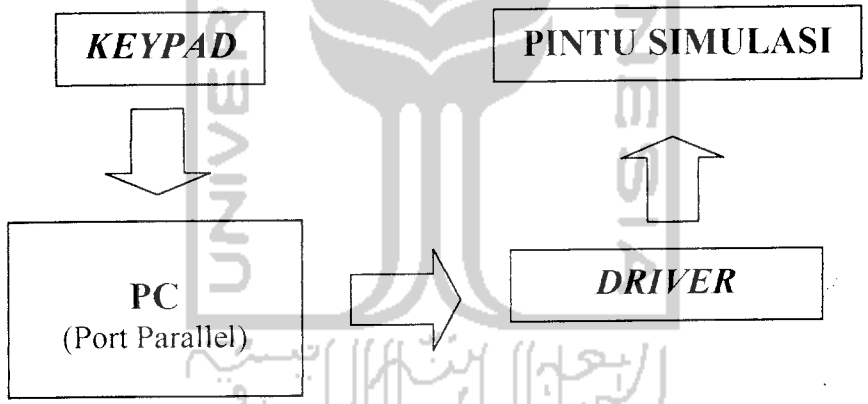
Gambar 2.8 Skema *Keypad*

BAB III

PERANCANGAN SISTEM

3.1 Gambaran Umum Sistem

Dalam pembuatan dan perancangan sistem untuk sebuah sistem presensi dan pengaman pintu menggunakan *password* dibutuhkan urutan proses yang akan dijalankan. Proses yang dibangun membutuhkan perangkat keras, perangkat keras yang diperlukan terdiri dari beberapa bagian dasar, yaitu bagian komputer, *keypad*, *driver*, catu daya dan perangkat yang dikontrol dalam hal ini adalah pintu. Adapun prinsip kerja alat ini dapat digambarkan sebagai berikut :



Gambar 3.1 Blok Diagram Sistem Presensi dan Pengaman Pintu

Di dalam Gambar 3.1 merupakan blok keseluruhan dari prinsip kerja alat dimana bagian – bagian ini memiliki fungsi yang berbeda satu dengan yang lainnya. *Keypad* merupakan tempat memasukkan input yang berupa karakter-karakter angka yang

digunakan sebagai *password*. Pada bagian komputer, setelah output data *keypad* diterima oleh port parallel, kemudian data akan diolah, dicocokkan dan dibandingkan dengan isi *database* komputer. Jika *password* sesuai maka akan diteruskan dengan memberi perintah *driver* untuk membuka atau menutup pintu simulasi.

3.2 Komputer

Disini komputer bertindak sebagai pengirim instruksi ke *driver* yang akan menggerakkan pintu simulasi ke posisi membuka dengan cara mengirim logika *high* ke port parallel pin 14 dan logika logika *low* ke pin 16. Untuk menutup pintu simulasi komputer akan mengirim logika *low* ke port parallel pin 14 dan logika *high* ke pin 16.

Selain mengirim instruksi komputer komputer juga menerima data dari *keypad* melalui port parallel. Data yang diterima dari *keypad* berupa logika *low* (0 volt) akan diartikan oleh program di dalam komputer sebagai karakter-karakter angka dan untuk logika *high* tidak akan diartikan. Selain itu komputer berfungsi sebagai penyimpan data presensi dan *password*. Definisi penggunaan port parallel sebagai masukkan data dari *keypad* dapat dilihat pada Tabel 3.1 berikut ini :

Tabel. 3.1 Definisi Penggunaan Port Data

	Port Data ($378)							
	Bit 7 (MSB)	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0 (LSB)**
Logika	0	0	0	0	0	0	0	0
Karakter	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Pin 9	Pin 8	Pin 7	Pin 6	Pin 5	Pin 4	Pin 3	Pin 2

Tabel. 3.2 Definisi Penggunaan Port Status

Port Status ($379)								
	Bit 7 (MSB	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0 (LSB)
Logika	-	0	0	-	-	-	-	-
Karakter	-	**8**	**9**	-	-	-	-	-
	-	Pin 10	Pin 12	-	-	-	-	-

Tabel. 3.3 Definisi Penggunaan Port Kontrol

Port Kontrol $37A				
	Bit 3	Bit 2	Bit 1	Bit 0
BUKA	-	**0**	**1**	-
TUTUP	-	**1**	**0**	-
		Pin 16	Pin 14	

3.3 Inisialisasi Port Pararel

Untuk memfungsikan port data $378 dan port status $379 agar dapat digunakan sebagai port baca (input) port control harus diinisialisasi dahulu. Agar semua pin port data dan port control berlogika *high* (5V) dapat dilakukan dengan cara memberi logika secara berurutan kepada port control. Urutan pemberian logika kepada port control dapat dilihat pada Tabel 4.1.

Tabel 3.4. Definisi Pengaturan Port Kontrol

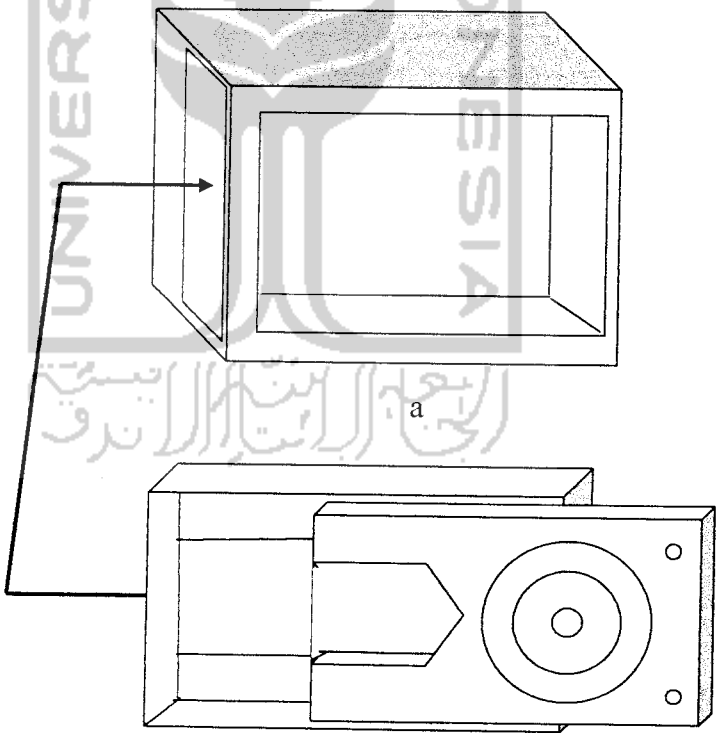
Port Kontrol $37A					
	Bit 3	Bit 2	Bit 1	Bit 0	HEX
Langkah 1	X	1	0	0	$4
Langkah 2	1	0	0	0	$8
Langkah 3	1	1	1	0	$FE
Langkah 4	1	0	1	1	$FB
No Kaki	Pin 17	Pin 16	Pin 14	Pin 1	

Pada perancangan pengalamatan pararel port untuk membaca tombol-tombol *password*, dibutuhkan 10 pin pada port pararel yang akan dihubungkan dengan tombol *password*. Untuk karakter tombol 0 sampai 7 akan dialamatkan pada port data dan dihubungkan ke port pararael dari pin 2 sampai 9, karena port data sudah penuh, maka sisa tombol untuk karakter 8 dan 9 akan menggunakan port status yang dihubungkan port pararel pada pin 10 dan 12.

Setiap penekanan tombol-tombol *password* akan memberikan logika *low* kepada port pararel dan jika logika *low* tersebut terpenuhi maka akan ditambahkan dengan karakter string yang akan dicocokkan dengan *database* oleh program, apabila cocok port kontrol $37A pada port pararel pin 14 akan dituliskan logika *high* kemudian diteruskan menuju transistor1 guna menggerakkan relai 1, sehingga membuat motor DC berputar ke arah kiri dan menggeser pintu pada posisi terbuka. Untuk alamat port kontrol $37A pin 16 digunakan memberi logika *high* pada transistor 2 guna menggerakkan relai 2, sehingga mengggeser pintu menjadi menutup kembali. Untuk Pin 18 sampai 25 dijadikan ground (Pentanahan).

3.4 Pintu Simulasi

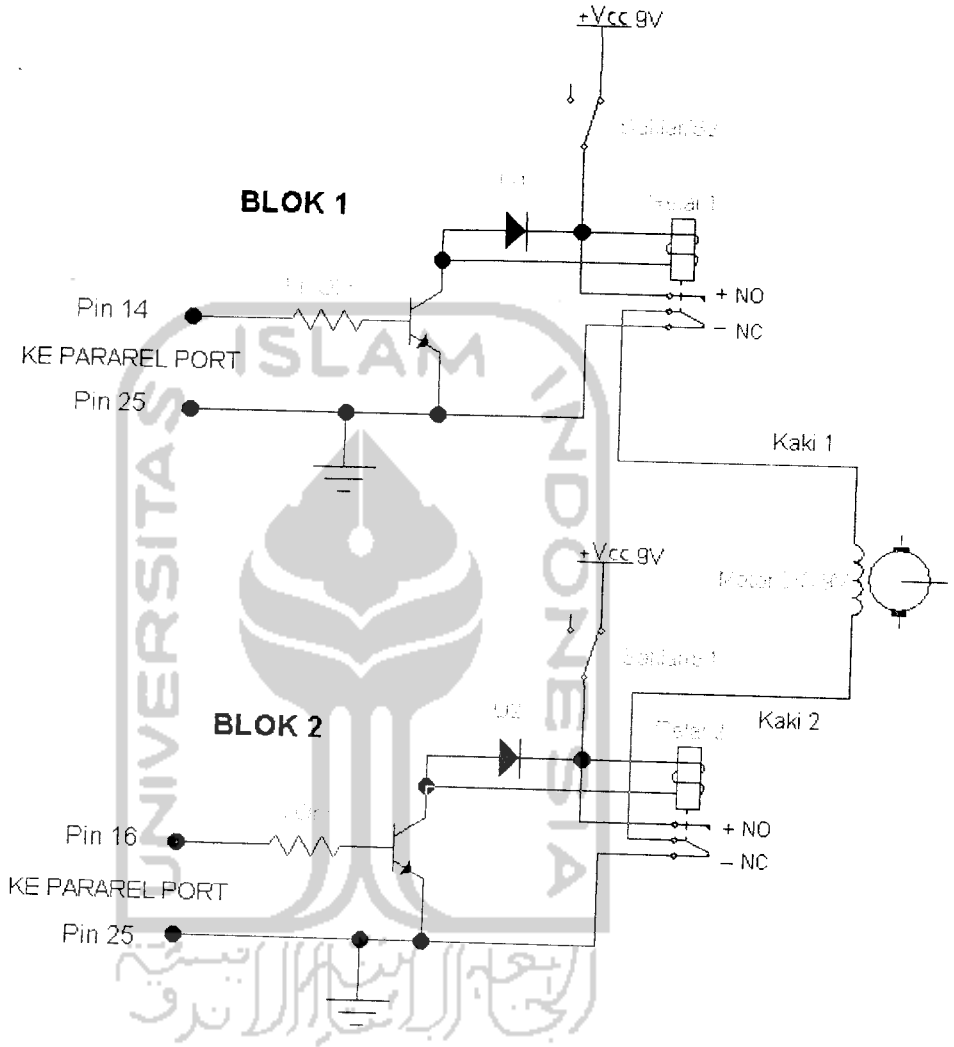
Untuk perancangan pintu simulasi dibuat dari kayu berbentuk kotak dengan dimensi ukuran panjang 18 cm x lebar 8 cm x tinggi 18 cm. Di dalam kotak kayu ini diisi sebuah *CD-ROM Drive* yang sudah tidak terpakai. *CD-ROM Drive* tersebut hanya diambil bagian mekanik dan motornya saja untuk digunakan sebagai pintu geser mekanis dan dipasangkan di dalam kotak kayu, di dalam *CD-ROM Drive* tersebut dipasang dua buah saklar, satu dibagian depan dan satunya lagi di bagian belakang *CD-ROM Drive*, untuk saklar 1 dibagian depan berguna untuk menghentikan laju motor ketika pada waktu pembukaan pintu sudah mencapai penuh, sedangkan untuk saklar 2 dibagian belakang digunakan untuk menghentikan laju motor ketika penutupan pintu sudah mencapai penuh. Untuk lebih jelasnya dapat dilihat seperti gambar 3.1 di berikut ini :



Gambar 3.2 a)Kotak Simulasi b)CD-ROM Drive

3.5 Driver

Driver berfungsi sebagai penyaluran dan pengaturan catu daya ke motor DC setelah menerima instruksi dari parallel port.



Gambar 3.3 Skema *driver*

Berikut adalah cara kerja dari *driver* untuk setiap bagian blok :

a. Blok Pertama

Komputer akan mengirim logika *high* (5 volt) melalui port control pada pin 14 yang diterima oleh *driver* menuju Blok 1 dan logika *low* (0 volt) melalui port control

pada pin 16 menuju blok 2, kemudian logika *high* ini akan diumpankan ke kaki basis transistor. Karena pada kaki basis transistor diberi tegangan positif (5 volt) dan emitter mendapat tegangan negatif dengan kondisi seperti ini maka transistor akan bersifat maju (*forward*). Tegangan yang dilewatkan melalui kaki emitter akan diteruskan menuju relai 1. Kumparan relai yang teraliri listrik ini akan membuat besi inti di dalam kumparan menjadi bermagnet dan akan menarik lengan armatur menuju positif (NO) *normally open*, sehingga kaki 1 akan terhubung dengan (NO) setelah terhubung motor DC bergerak kearah kanan. Pergerakan motor ke arah kanan ini akan mengakibatkan pintu membuka dan akan berhenti ketika pintu mengenai saklar1.

b. Blok kedua

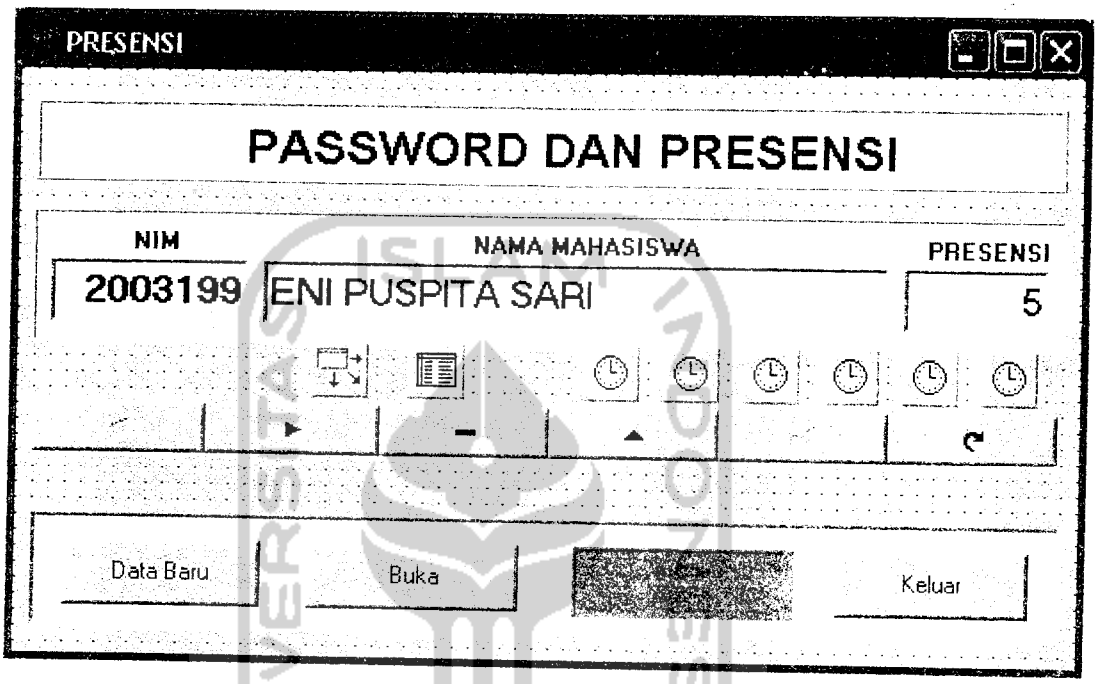
Komputer akan mengirim logika *high* (5 volt) melalui port control pada pin 16 yang diterima oleh *driver* menuju Blok 2 dan logika *low* (0 volt) melalui port control pada pin 14 menuju blok 1, kemudian logika *high* ini akan diumpankan ke kaki basis transistor. Karena pada kaki basis transistor diberi tegangan positif (5 volt) dan emitter mendapat tegangan negatif dengan kondisi seperti ini maka transistor akan bersifat maju (*forward*). Tegangan yang dilewatkan melalui kaki emitter akan diteruskan menuju relai 2. Kumparan relai yang teraliri listrik ini akan membuat besi inti di dalam kumparan menjadi bermagnet dan akan menarik lengan armatur menuju positif (NO) *normally open*, sehingga kaki 1 akan terhubung dengan (NO) setelah terhubung membuat motor DC bergerak ke arah kiri. Pergerakkan motor ke arah kiri ini akan mengakibatkan pintu menutup dan akan berhenti ketika pintu mengenai saklar 2.

c. Pemberian logika terhadap port control pada pin 14 dan 16 akan selalu terbalik, maksudnya jika pada blok 1 diberi logika *high* maka kaki 1 pada motor DC akan

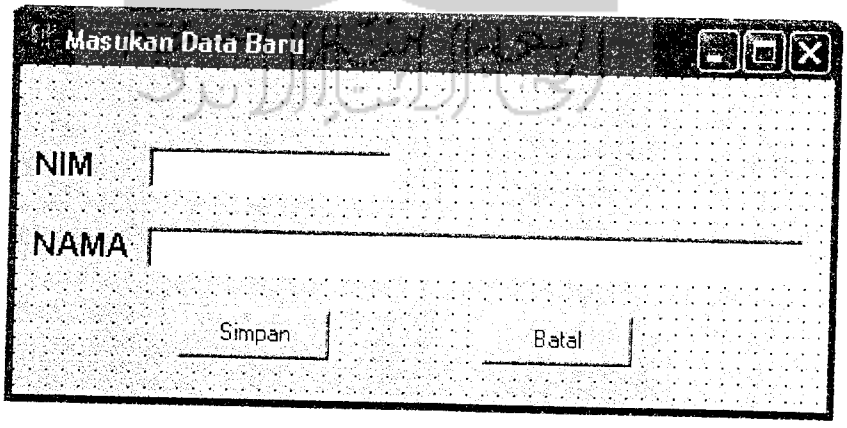
terhubung dengan tegangan positif dan pemberian logika *low* pada blok 2 akan membuat kaki 2 motor DC terhubung dengan tegangan negatif, sehingga mengakibatkan motor DC bergerak ke arah kanan dan mengakibatkan pintu pada posisi membuka. Begitu juga dengan pemberian logika *low* pada blok 1 dan pemberian logika *high* pada blok 2 akan mengakibatkan motor bergerak ke arah kiri dan mengakibatkan pintu pada posisi menutup.

3.6 Perancangan *Software*

Perancangan *software* yang baik adalah mudah digunakan dan dimengerti oleh pemakainya, selain itu juga mudah diintegrasikan ke berbagai sistem operasi. Dalam perancangan ini penggunaan komponen-komponen visualnya adalah sebagai berikut :



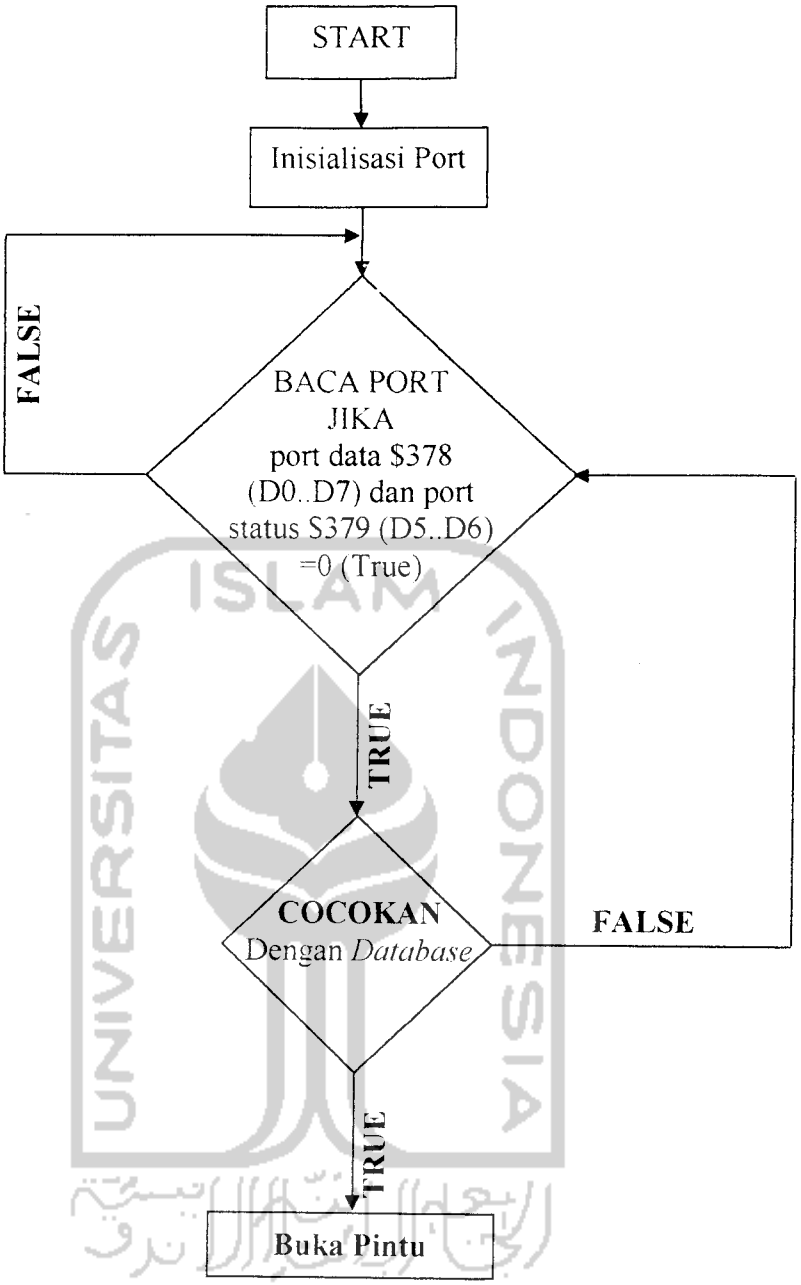
Gambar 3.5 Perancangan Tampilan Form Utama



Gambar 3.6 Perancangan Tampilan Form Input Data

3.7 *Flow* Chart

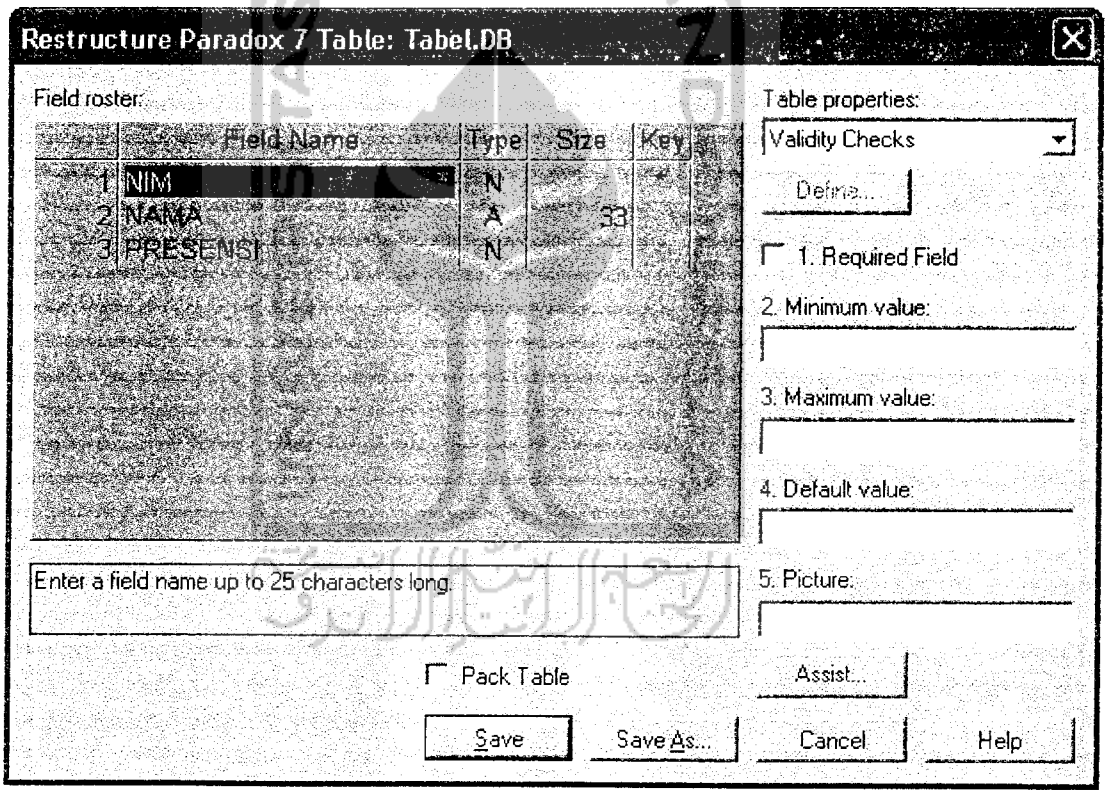
Urutan *flow chart* dimulai dari start kemudian akan diteruskan dengan inisialisasi port. Dalam proses inisialisasi ini, semua port data $378 dan port status $379 akan dibuat berlogika *high* (5 volt). Setelah dilakukan inisialisasi kemudian akan dilakukan pembacaan port data $378 pada bit ke 0 sampai bit ke 7 dan pembacaan port status $379 pada bit ke 5 dan bit ke 6. Jika pada port $378 atau $379 diberi masukkan logika *low*, maka bit-bit rendah ini akan dirubah sebagai karakter angka. Kemudian karakter angka ini akan dicocokkan dengan isi *database*, apabila karakter angka sesuai maka akan dilakukan perintah untuk membuka pintu.



Gambar 3.7 Diagram Alir Program

3.8 Perancangan *Database*

Dalam perancangan *database* digunakan *database* Paradox, untuk penentuan field-nya ada 3, untuk field yang pertama diberi nama NIM dengan Type N (Number) dengan karakter hanya untuk angka, dan disini NIM digunakan sebagai index pengurutan dan sebagai *password*, untuk field yang kedua digunakan sebagai field penyimpanan nama mahasiswa dan menggunakan Type A (Alpha) dengan ukuran karakter huruf maksimal sebanyak 33 huruf atau karakter dan field yang ketiga menggunakan Type N (Number).



Gambar 3.8 Tabel Restruktur Paradox

Tabel 3.5 Kebutuhan Komponen Program dan Perubahan Properti

No.	Komponen	Property	Perubahan
1	Panel1	Caption	-
		BevelInner	Bv*Low*ered
		BevelOuter	BvSpace
		Color	ClInfoBk
		height	41
		Left	8
		Taborder	3
		Top	16
		Width	513
2	Label 1	Caption	*Password*
		Font	(Tfont)
		Height	29
		Left	102
		Parenfont	False
		Top	8
		Width	311
3	Panel 2	BevelOuter	bv*low*ered
		Color	ClInfoBk
		Height	65
		Left	8
		TabOrder	7
		Top	72
		Width	513
4	Label 2	Caption	NIM
		Height	13
		Left	32
		Parenfont	false
		Top	8
		Width	24
5	Label 3	Caption	Nama Mahasiswa
		Height	13
		Left	212
		Parenfont	False
		Top	8
		Width	115
6	Label 4	Caption	PRESENSI
		Height	13
		Left	435
		Parenfont	False

		Top	8
		Width	65
7	DBEdit 1	Datafield	NIM
		Datasource	DataSource 1
		Height	32
		Left	16
		Parentfont	False
		TabOrder	4
		Top	96
		Width	97
8	DBEdit 2	Datasource	DataSource 1
		Height	32
		Left	120
		Parentfont	False
		TabOrder	5
		Top	96
		Width	313
9	DBEdit 3	Datafield	PRESENSI
		Datasource	Datasource 1
		Height	32
		Left	440
		Parentfont	False
		TabOrder	6
		Top	96
		Width	73
10	Data Source 1	AutoEdit	True
		Data Set	Tabel
		Enabled	True
		Tag	0
11	Tabel Db	Tabble type	ttDefault
		Tag	0
		Aktive	True
12	Timer 1	Enabled	True
		Interval	1000
		Tag	0
13	Timer 2	Enabled	True
		Interval	1000
		Tag	0
14	DBNavigator	DataSource	DataSourse 1
		Height	25
		Left	8
		TabOrder	9
		Visible Bottons	(nbprior, nbtext, nbdelete, nbedit, nbpost, nbrefresh)
		Top	168
		Width	510

15	Panel 3	Bevellnoer	Bv*low*ered
		BevelOuter	V*low*ered
		Left	8
		TabOrder	8
		Top	224
		Width	213
16	Botton 1	Caption	Data baru
		Height	33
		Left	24
		TabOrder	1
		Top	237
		Width	97
17	Botton 2	Caption	Keluar
		Height	33
		Left	200
		TabOrder	2
		Top	237
		Width	97
18	Edit 1	BevelOuter	Bv*low*ered
		Color	Clwhite
		Font	(TFont)
		Height	32
		Left	384
		ReadOnly	True
		TabOrder	0
		Top	237
		Width	121

BAB IV

PENGUJIAN DAN ANALISIS

Dalam analisis dan pembahasan alat sistem presensi dan pengaman pintu menggunakan *password* ini, perangkat keras yang diuji terdiri dari beberapa bagian yaitu, pengujian *driver* motor DC, pengujian *keypad*, dimana bagian-bagian ini memiliki fungsi yang berbeda satu dengan yang lainnya.

4.1 Pengujian *Driver*

Dalam pengujian *driver* untuk penggerak mekanis motor DC ini untuk posisi membuka dan menutup atau menggerakkan motor berputar ke arah kiri dan ke arah kanan, urutan kerja dapat dilihat pada Tabel 4.1.

Tabel. 4.1 Posisi Pintu Terbuka dan Tertutup

Posisi Membuka				
Blok 1	**Blok 2**	**Motor**	**Saklar 1**	**Saklar 2**
High (5 volt)	*Low* (0 volt)	Ke Kanan	Tersambung	Terputus
Posisi Menutup				
Blok 1	**Blok 2**	**Motor**	**Saklar 1**	**Saklar 2**
Low (0 volt)	*High* (5 volt)	Ke Kiri	Terputus	Tersambung

a. Saklar 2 berfungsi sebagai pemutus tegangan pada motor sewaktu pintu terbuka penuh dan posisi pada saklar 1 akan tetap tersambung.

b. Saklar 1 berfungsi sebagai pemutus tegangan pada motor sewaktu pintu tertutup penuh dan posisi pada saklar 2 akan tetap tersambung. Selain itu saklar 1 dan saklar 2 berfungsi sebagai pengaman motor terhadap beban lebih jika terjadi logika yang berjalan tidak sesuai *(error program)* misalnya, kondisi logika yang berada di Pin 14 dan 16 yang menuju transistor 1 dan transistor 2 semuanya *high* (5V) sehingga pintu tidak bisa membuka ataupun menutup.

4.2 Pengujian *Keypad*

Pengujian terhadap *keypad* dapat dilakukan menggunakan multimeter atau multitester pada posisi saklar menunjuk ke arah symbol Ω (Ohm), untuk faktor perkaliannya pada multimeter X10, X100, X1K atau X100k dapat diabaikan dalam arti bebas pemilihannya, karena yang diukur hanya saklar, jadi tidak diperlukan keakuratan dalam hal pengukuran hambatan (resistansi). Urutan pengertian pin dan cara pengujian adalah sebagai berikut :

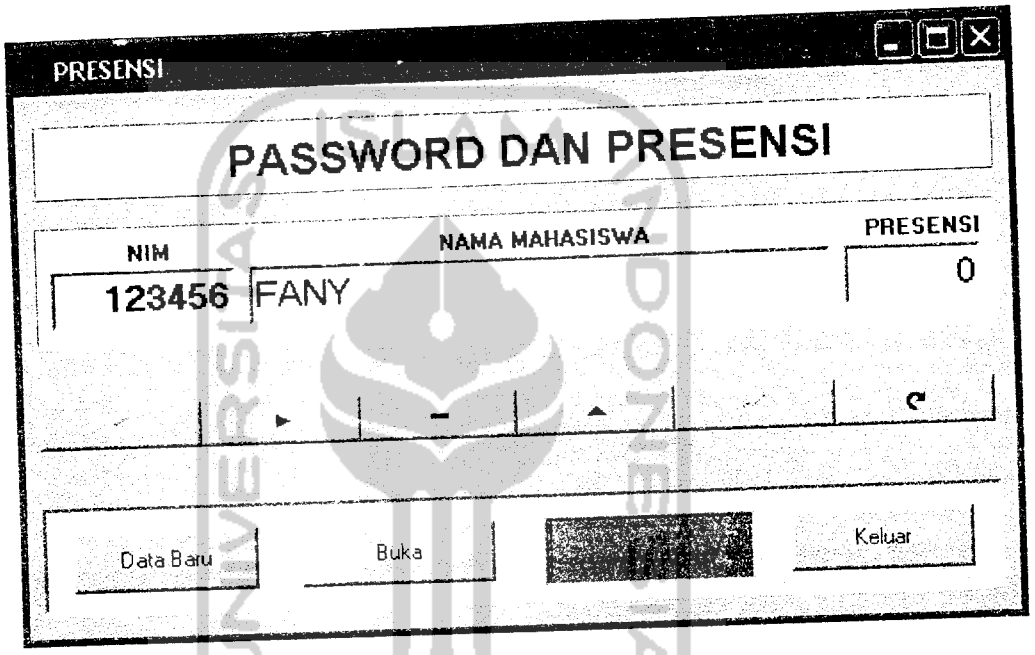
a. Kabel warna strip hitam yang paling tepi digunakan untuk karakter angka 0 (pin 2). Untuk sisa warna sesudah strip hitam digunakan untuk karakter 1, 2, 3, 4, 5, 6, 7, 8, 9.

b. Untuk pengujian setiap karakter apakah berjalan normal atau tidak dapat dilakukan dengan mengukur pin ground dan pin data. Jika dilakukan penekanan pada *keypad* dan kondisi *keypad* baik maka arah jarum multimeter menyimpang kearah kanan.

4.3 Pengujian Perangkat Lunak (*Software*)

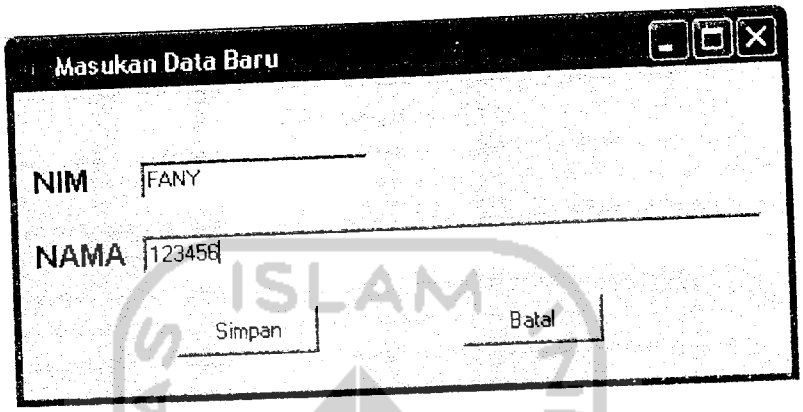
Pengujian perangkat lunak dapat dilakukan dengan menghubungkan rangkaian *driver* dan *keypad* menuju port paralel kemudian langkah-langkah selanjutnya sebagai berikut:

a. Jalankan perangkat lunak sistem presensi dan pengaman pintu.



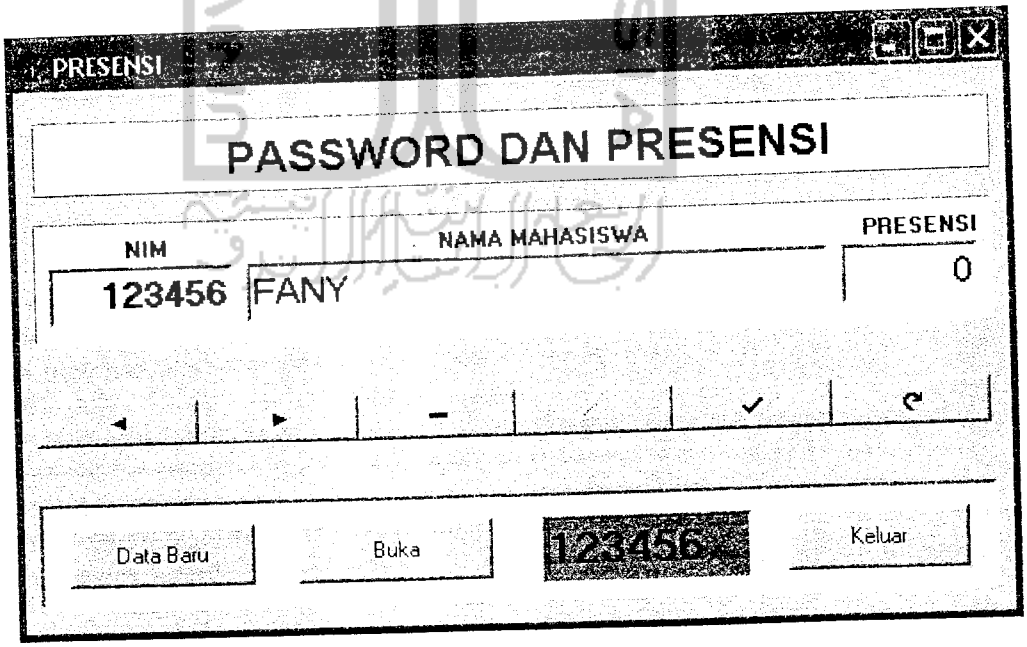
Gambar 4.1. Program Ketika Dijalankan

b. Isi data terlebih dahulu jika data masih kosong dengan cara menekan tombol Data Baru, kemudian akan tampil Form Input Data, isi kolom NIM diisi dengan *password* dan kolom Nama dengan nama pengguna.



Gambar 4.2. Tampilan Form Ketika Memasukkan Data

c. Masukkan *password* dengan menekan *keypad* sesuai dengan data NIM yang dimasukkan.



Gambar 4.3. Tampilan Ketika *Keypad* Ditekan

Setelah langkah-langkah tersebut dilakukan kemudian dilakukan uji dengan cara menekan tombol *keypad* untuk memasukkan *password* jika *password* cocok dengan data *password* yang terletak di field NIM, maka akan dilakukan pengisian data angka di field PRESENSI dan selanjutnya akan dilakukan perintah untuk membuka pintu.

Pengujian perubahan interval pada timer-timer akan memberi sifat dan pengaturan yang berbeda serta mempunyai kegunaan tersendiri, pengujian tersebut didapatkan diuraikan sebagai berikut :

a. *Timer 1* dan 2 berfungsi untuk mengatur respon penekanan *keypad* dengan merubah interval kedua *timer* tersebut. Selain itu pada *even timer-timer* ini berfungsi untuk membaca logika yang terjadi pada port data dan port status.
b. *Timer 3* berfungsi memberi perintah penulisan logika *high* (I) pada port control di pin 14. Pada event timer ini diberi perintah untuk menggerakan motor DC ke arah kanan atau membuka. Karena sifat dari pin 14 adalah terbalik maka diberi logika AND agar sifatnya menjadi normal.
c. *Timer 4* berfungsi memberi perintah penulisan logika *low* (0) pada port control di pin 14, agar pada waktu diberi perintah untuk menutup tidak akan terjadi kesamaan logika dengan pin 16. Pengaturan lamanya *(delay)* penutupan pintu ditentukan oleh interval timer ini dengan mengatur intervalnya.
d. *Timer 5* berfungsi memberi perintah penulisan logika *high* (I) pada port control di pin 16. Lalu pada event timer ini diberi perintah untuk menggerakan motor DC ke arah kiri atau menutup.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pengujian dan analisa yang telah dilakukan pada bab-bab sebelumnya, penulis dapat mengambil beberapa kesimpulan sebagai berikut :

a. *Input driver* akan selalu mendapat logika yang berbeda, karena jika masukan logika sama maka relai akan menghubungkan kedua kaki motor DC menuju polaritas yang sama dan ini akan mengakibatkan motor DC tidak bekerja.
b. Dikarenakan keterbatasan catu daya yang dimiliki oleh komputer, maka *driver* berfungsi juga sebagai penyalur catu daya yang lebih besar ke motor.
c. *Driver* menjembatani hubungan antara komputer dengan motor.
d. Perangkat lunak akan mengisi data presensi dan membuka pintu apabila *password* sesuai dengan isi *database*.

5.2 Saran

Beberapa saran yang dapat memberi manfaat terhadap sistem presensi dan pengamanan pintu menggunakan *password* ini adalah dengan menambahkan atau mengganti komponen-komponen perangkat keras tertentu, sehingga menunjang dalam kinerjanya, seperti dengan mengganti komponen motor DC yang dapat memutar beban di atas 10 kg, serta menambahkan antarmuka PPI 8255 dan *seven segmen* atau LCD sebagai tampilan layar yang dipasang di depan pintu, agar pada saat penekanan *password* dapat dilihat langsung dari luar.

DAFTAR PUSTAKA

Amos, S.W., Maret 1997. Kamus Elektronika, Elex Media Komputindo.

Chattopadhyay, dkk, 1989. Dasar Elektronika. Penerbit Universitas Indonesia, UI-PRESS.

Sudono, Agus. 2004. Memanfaatkan Port Printer komputer menggunakan Delphi, SMART BOOK.

Wahana Komputer, 2003. Panduan Praktis Pemrograman Delphi 7.0, Andi Offset, Yogyakarta.

Wijaya Citra, Paulus. Agustus 1999. Buku Data IC Catu Daya, PT. Elex Media Komputindo.

Zamidra Zam, Elvy, Oktober 2002. Mudah Menguasai Elektronika. Penerbit Indah. Surabaya.

LAMPIRAN

LISTING PROGRAM DARI DELPHI

```
unit Unit1;

interface

uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, ExtCtrls, Mask, DBCtrls, Db, DBTables;

type
TForm1 = class(TForm)
Edit1: TEdit;
Timer1: TTimer;
DataSource1: TDataSource;
Tabel: TTable;
Button1: TButton;
Button2: TButton;
Panel1: TPanel;
Label1: TLabel;
DBEdit1: TDBEdit;
DBEdit2: TDBEdit;
DBEdit3: TDBEdit;
Panel2: TPanel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Panel3: TPanel;
DBNavigator1: TDBNavigator;
Timer2: TTimer;
Timer3: TTimer;
Timer4: TTimer;
Button3: TButton;
TabelNIM: TFloatField;
TabelNAMA: TStringField;
TabelPRESENSI: TFloatField;
Timer5: TTimer;
Timer6: TTimer;
procedure FormActivate(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Edit1Change(Sender: TObject);
```

```
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure Timer3Timer(Sender: TObject);
procedure Timer4Timer(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Timer5Timer(Sender: TObject);
procedure Timer6Timer(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
Form1: TForm1;

implementation

uses Unit2;

{$R *.DFM}

function PortIn(Port:Word):Byte;stdcall;external'io.dll';

procedure PortOut(Port:Word;Data:Byte);stdcall;external'io.dll';

procedure TForm1.FormActivate(Sender: TObject);
var tmp:byte;
begin
 Form1:=TForm1.Create (Self);
 tmp:=PortIn($37A);
 PortOut($37A,tmp or $4);
 PortOut($37A,tmp or $8);
 PortOut($37A,tmp or $FE);
 PortOut($37A,tmp or $FB);
 Timer3.Enabled:=false;
 Timer4.Enabled:=false;
 Timer5.Enabled:=false;
 Timer6.Enabled:=false;
 Tabel.Active:=true;
end;
```

```
procedure TForm1.Timer1Timer(Sender: TObject);
var x:byte;
begin
x:=PortIn($378);
if x mod 2= 0 then
Edit1.Text:=Edit1.Text+'0';
if (x shr 1) mod 2=0 then
Edit1.Text:=Edit1.Text+'1';
if (x shr 2) mod 2=0 then
Edit1.Text:=Edit1.Text+'2';
if (x shr 3) mod 2=0 then
Edit1.Text:=Edit1.Text+'3';
if (x shr 4) mod 2=0 then
Edit1.Text:=Edit1.Text+'4';
if (x shr 5) mod 2=0 then
Edit1.Text:=Edit1.Text+'5';
if (x shr 6) mod 2=0 then
Edit1.Text:=Edit1.Text+'6';
if (x shr 7) mod 2=0  then
Edit1.Text:=Edit1.Text+'7';
end;

procedure TForm1.Edit1Change(Sender: TObject);
var PresensiBaru : Extended;
PresensiLama : Extended;
STotal     : String;
Total      : Extended;
Ada        : Boolean;
begin
Edit1.ReadOnly:=True;
Ada:=Tabel.FindKey([Edit1.Text]);
if ada then
begin
Tabel.Edit;
PresensiLama:= Tabel.Fields[2].AsFloat;
PresensiBaru := 1;
Total:=PresensiLama+PresensiBaru;
Str (Total, STotal);
Tabel.Fields[2].AsString:= STotal;
Tabel.Post;
Timer3.Enabled:=true;
if length(Edit1.Text)= 7 then
Edit1.SelectAll;
```

```
end else
begin
if length(Edit1.Text)= 8 then
Edit1.Clear;
end;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
Form2 := TForm2.Create (Self);
Form2.ShowModal;
if (Form2.ModalResult = mrOk) then
with Tabel, Form2 do
begin
Insert;
FieldByName ('NIM').AsString := Edit1.Text;
FieldByName ('NAMA').AsString := Edit2.Text;
Post;
end;
  Form2.Free;
end;

procedure TForm1.Button2Click(Sender: TObject);
var tmp:byte;
begin
tmp:=PortIn($37A);
PortOut($37A, tmp and $fb);
Application.Terminate;
Form1.Free;
end;

procedure TForm1.Timer2Timer(Sender: TObject);
var x:byte;
begin
x:=PortIn($379);
if (x shr 5) mod 2=0 then
Edit1.Text:=Edit1.Text+'9';
if (x shr 6) mod 2=0 then
Edit1.Text:=Edit1.Text+'8';
if (x shr 3) mod 2= 1 then
end;

procedure TForm1.Timer3Timer(Sender: TObject);
```

```
var tmp:byte;
begin
tmp:=PortIn($37A);
PortOut($37A, tmp and $2);
Timer4.Enabled:=true;
end;

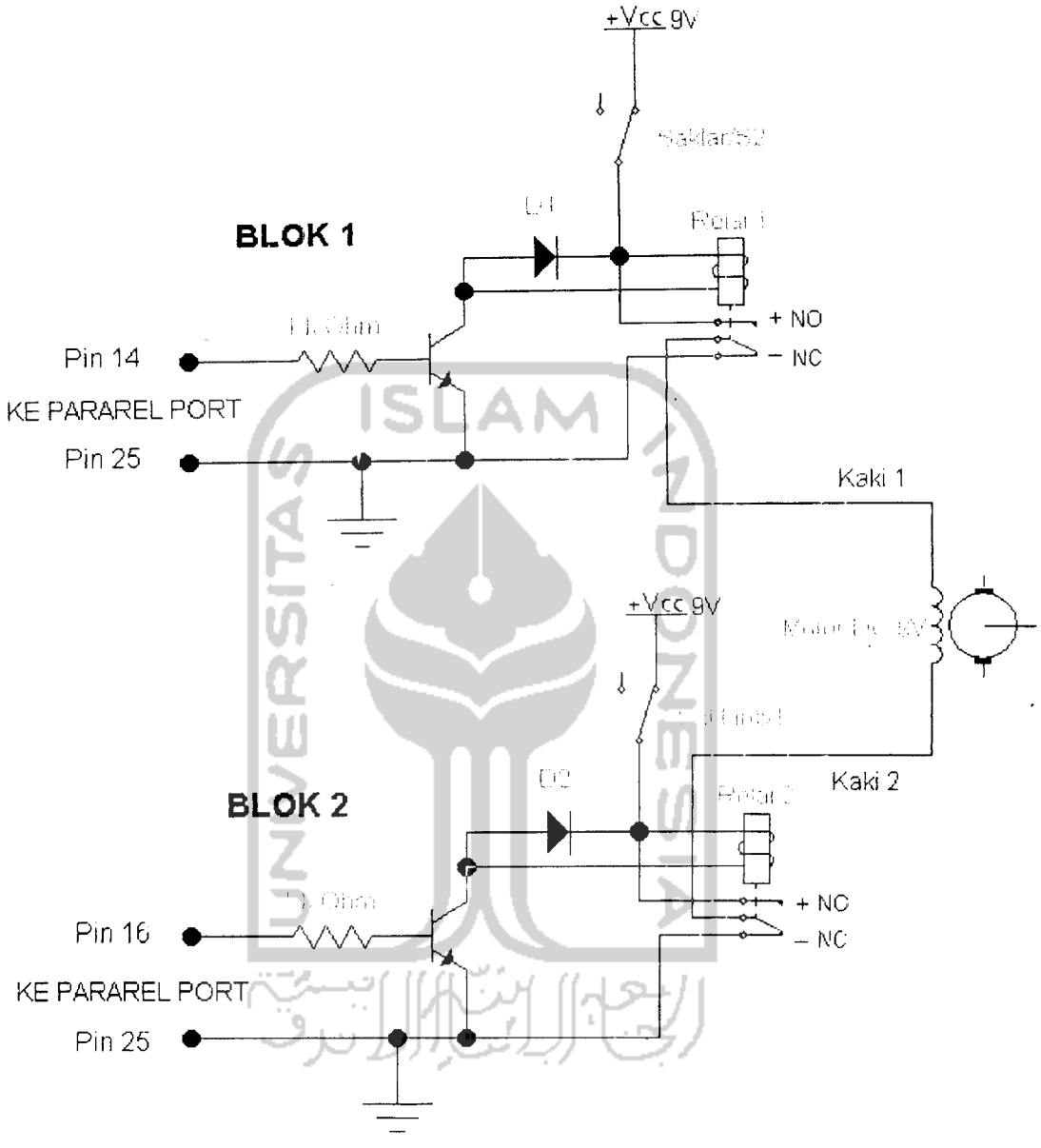
procedure TForm1.Timer4Timer(Sender: TObject);
var tmp:byte;
begin
tmp:=PortIn($37A);
PortOut($37A, tmp and $fd);
Timer3.Enabled:=false;
Timer5.Enabled:=true;
end;

procedure TForm1.Button3Click(Sender: TObject);
begin
Timer6.Enabled:=true;
end;

procedure TForm1.Timer5Timer(Sender: TObject);
var tmp:byte;
begin
tmp:=PortIn($37A);
PortOut($37A, tmp or $4);
Timer4.Enabled:=false;
Timer6.Enabled:=true;
end;

procedure TForm1.Timer6Timer(Sender: TObject);
var tmp:byte;
begin
tmp:=PortIn($37A);
PortOut($37A, tmp and $fb);
Timer5.Enabled:=false;
PortOut($37A,tmp or $4);
PortOut($37A,tmp or $8);
PortOut($37A,tmp or $FE);
PortOut($37A,tmp or $FB);
Timer6.Enabled:=false;
end;
end.
```

SKEMATIK DRIVER



E PARAREL P

SKEMATIK KEYPAD

