

**APLIKASI BEL OTOMATIS TERPROGRAM
BERBASIS PC (PERSONAL COMPUTER)**



Oleh :

RADEN DEDY IRIANTO

NIM. M3304030

TUGAS AKHIR

ditulis dan diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Ahli Madya Ilmu Komputer

**PROGRAM DIPLOMA III ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET
SURAKARTA**

2007

TUGAS AKHIR
APLIKASI BEL OTOMATIS TERPROGRAM
BERBASIS PC (PERSONAL COMPUTER)

yang disusun oleh
RADEN DEDY IRIANTO
M3304030

dibimbing oleh

Pembimbing I,	Pembimbing II,
Fatchul Arifin, MT.	Dra. Respatiawulan, M.Si.
NIP. 132 206 815	NIP. 132 046 022

telah dipertahankan di depan Dewan Penguji
pada hari Selasa, tanggal 07 Agustus 2007
dan dinyatakan telah memenuhi syarat.

Anggota Tim Penguji	Tanda Tangan
1. Fatchul Arifin, MT.	1.
2. Ristu Saptono, S.Si, MT	2.

Surakarta, 11 Agustus 2007

Disahkan oleh

Fakultas MIPA
Dekan,

Prof. Drs. Sutarno, M.Sc, Ph.D
NIP. 131 649 948

Ketua Program Studi DIII
Ilmu Komputer,

Drs. YS. Palgunadi, M.Sc
NIP. 131 285 866

HALAMAN ABSTRAK

Tujuan penelitian ini adalah mencoba merancang sebuah aplikasi sistem kontrol melalui *port* paralel pada komputer. Dengan menerapkan sistem tersebut pada sebuah bel listrik yang akan diatur kapan jam mulai aktif menyala.

Aplikasi ini dibangun dengan menggunakan software Delphi, dengan memanfaatkan komponen-komponen yang ada antara lain *timer*, *DBGrid*, *ADOTable*, dan komponen pendukung lainnya.

Dari hasil akhir tugas ini disimpulkan bahwa komputer dapat dimanfaatkan sebagai sistem kontrol pada sebuah perangkat listrik dengan memanfaatkan port paralel yang ada pada komputer dan dipadukan dengan sebuah bahasa pemrograman dalam hal ini Delphi.

Keyword: Bel Otomatis, *Port Paralel*, Delphi.

HALAMAN MOTTO

- “Man Jadda wa Jada: *Where is the will there is the way*: Sopo Temen Bakal Tinemu”
- Ada “Maha kekuatan” di dunia ini yang menentukan segalanya, jangan kau lalaikan itu.
- Hidup adalah perjuangan tanpa henti, usah kau tangisi hari kemarin.

PERSEMBAHAN

Tugas Akhir ini dipersembahkan untuk:

*Keluarga Besar Mbah Parto Giyono atas segala dukungan yang tak terkirakan
Keluarga Besar “Jamaah Mujahadah Pengayoman Ilahi” atas bantuan moril dan spiritualnya
Teman-teman seperjuangan “Teknik Komputer’04” UINS*

MUTIARA KATA

Al Qur'an:

Alloh SWT berfirman: “Dan katakanlah: Bekerjalah kamu, maka Alloh dan Rosul-Nya serta orang-orang mukmin akan melihat pekerjaanmu itu, dan kamu akan dikembalikan kepada (Alloh) yang mengetahui akan yang ghoib dan yang nyata, lalu diberitakan-Nya kepada kamu apa yang telah kamu kerjakan”. (QS At-Taubah, 9:105)

Al-Hadist:

Rosululloh SAW bersabda: ”Duduklah kalian bersama orang-orang yang mulia, bertanyalah kepada ‘ulama dan bergabunglah dengan orang-orang yang bijaksana”. (HR. Thabrani)

Kalam ‘Ulama

Al-Imam Syafi’i: “ Aku mencintai orang-orang yang sholeh walau tak termasuk dalam kelompoknya. Kulakukan itu untuk memperoleh syafa’at mereka”.

Al-Habib Ali bin Muhammad bin Husein Al-Habsyi:”Ya Alloh, hamba-Mu yang hina ini tidaklah mampu bersyukur atas ni’mat-ni’mat yang telah Engkau berikan kepadaku, karena ungkapan syukur itu sudah termasuk dari ni’mat agung-Mu, Engkau adalah sebaik-baik tempat untuk bersyukur maka ampuni dosa-dosaku dan masukanlah hamba-Mu ini menjadi orang-orang yang mensyukuri ni’mat-Mu”.

KATA PENGANTAR

Alhamdulillah *robbil 'alamin*, puji syukur yang tiada terhingga ke hadhirot Alloh SWT yang telah melimpahkan rahmat serta hidayah-Nya atas terselesainya penulisan tugas akhir ini dengan baik dan lancar. Sholawat serta salam selalu tercurahkan kepada *nabiyuna wa habibuna wa syafi'ina sayidi* Rosul Muhammad SAW, keluarga beliau, sahabat-sahabat beliau, 'ulama, syuhada', sholihin, serta orang-orang yang mengikuti sunnahnya hingga *yaumul Qiamah*.

Maksud dari penulisan tugas akhir ini adalah dalam rangka memenuhi sebagian persyaratan memperoleh gelar Ahli Madya Ilmu Komputer di Program Diploma III (D3) Ilmu Komputer Jurusan Teknik Komputer Universitas Sebelas Maret. Dengan adanya berbagai keterbatasan dalam penulisan Laporan Tugas Akhir ini, terutama keterbatasan kemampuan dan pengetahuan serta adanya keterbatasan data yang diperoleh, sehingga hasil Laporan Tugas Akhir ini masih jauh dari sempurna.

Ucapan terimakasih yang sebesar-besarnya penulis sampaikan kepada semua pihak yang telah memberikan bantuan kepada penulis sehingga dapat tersusun dan selesainya Laporan Tugas Akhir ini. Maka dalam kesempatan ini penulis mengucapkan terimakasih kepada :

1. Bapak Irwan Susanto, DEA. Selaku Ketua Program Diploma III Ilmu Komputer Universitas Sebelas Maret.
2. Bapak Fatchul Arifin, MT. dan ibu Dra. Respatiwan, M.Si. selaku dosen pembimbing yang telah meluangkan waktunya untuk membimbing dan memberi pengarahan yang sangat berharga dalam penulisan ini.
3. Keluarga besar Mbah Parto Giyono atas segala dukungannya.
4. Teman-teman di MIPA UNS semua.

Semoga atas usaha dan hajat-hajatnya diterima dan diridhoi oleh Alloh SWT. Dan kami berharap dengan belajar ilmu-ilmu murni termasuk salah satu sarana untuk beribadah dan mendekatkan diri kepada Alloh SWT.

Akhir kata semoga tulisan ini dapat bermanfaat bagi semua pihak.

Surakarta, 25 Juli 2007

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN ABSTRAK	iii
HALAMAN MOTTO	iv
HALAMAN PERSEMBAHAN	v
MUTIARA KATA	vi
KATA PENGANTAR.....	vii
DAFTAR ISI	viii
DAFTAR GAMBAR DAN TABEL	ix

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	3

BAB II LANDASAN TEORI DAN TINJAUAN PUSTAKA

2.1. <i>Interface Port</i> Paralel Komputer	4
2.2. Port Paralel	4
2.2.1 Pengenalan <i>Port</i> Paralel.....	4
2.2.2 Diagram Pin <i>Port</i> Paralel.....	5
2.3. Pengertian Sistem	6
2.4. Pengertian <i>Database</i>	7
2.5. Pengertian Delphi	8
2.6. Resistor.....	8
2.7. Kapasitor	9
2.8. Transistor.....	9

2.9. Dioda.....	10
2.10. Optocoupler dan Triac	11
2.10.1. Optocoupler	11
2.10.2. Triac.....	12

BAB III PERANCANGAN SISTEM

3.1 Gambaran Umum Sistem	14
3.1.1. Perancangan <i>Hardware</i> (Perangkat Keras)	15
3.1.2. Perancangan Software (Perangkat Lunak)	17
3.1.3. Perancangan <i>Database</i>	20
3.2 <i>Library Research</i>	21
3.3 Alat Penelitian	21

BAB IV PENGUJIAN, IMPLEMENTASI, DAN PEMBAHASAN

4.1 Evaluasi Pengujian Sistem	23
4.1.1 Pengujian Perangkat Keras	23
a. Pengujian Driver Saklar elektronik	23
b. Pengujian <i>Port</i> Paralel	24
4.1.2. Pengujian Sistem Bel Otomatis	25
4.2 Keunggulan dan Kelemahan Sistem	27
4.2.1. Keunggulan	27
4.2.2. Kelemahan	28

BAB V PENUTUP

5.1 Kesimpulan	29
5.2 Saran	29

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL DAN GAMBAR

Tabel 2.1 Tabel Fungsi Pin Port Paralel DB-25	6
Tabel 3.1 Tabel database yang digunakan	21
Tabel 4.1 Hasil Pengujian Driver	24
Tabel 4.2 Hasil Pengujian Sistem dan	25
Gambar 2.1 Pin konektor DB-25	5
Gambar 2.2 Tampilan Program utama Delphi	9
Gambar 2.3 Simbol dan Bentuk komponen Resistor.....	10
Gambar 2.4 Simbol Kapasitor	10
Gambar 2.5 Simbol Transistor NPN.....	11
Gambar 2.6 Simbol Transistor PNP	11
Gambar 2.7 Simbol Dioda.....	12
Gambar 2.8 Simbol Optocoupler.....	12
Gambar 2.9 Simbol TRIAC dan Keterangan kaki.....	13
Gambar 2.10 Rangkaian TRIAC sederhana.....	13
Gambar 3.1 Skema sistem kendali bel listrik otomatis.....	14
Gambar 3.2 Skema catu daya	15
Gambar 3.3 Skema rangkaian internal saklar elektronik	16
Gambar 3.4 Rancangan Alur Program.....	18
Gambar 3.5 Tampilan Program	19
Gambar 3.6 Tampilan Program Aktif.....	19
Gambar 3.7 Tampilan Form Setting	20
Gambar 4.1 Driver saklar elektronik	23
Gambar 4.2 Skema Rangkaian <i>input port</i> paralel.....	24
Gambar 4.3 Tampilan program uji input port paralel.....	25
Gambar 4.4 Tampilan program yang telah aktif	26
Gambar 4.5 Form setting saat penyetingan.....	27

BAB I

PENDAHULUAN

1.1 Latar Belakang

Masih banyak orang yang berasumsi kegunaan komputer hanya sebagai alat untuk menjalankan aplikasi, seperti pengetikan, perhitungan, membuat gambar, mendengarkan musik, bermain dan sebagainya.

Padahal komputer juga bisa digunakan untuk keperluan pengontrolan peralatan listrik rumah tangga seperti bel listrik, lampu, kipas angin dan lain-lain dengan memanfaatkan *Port Paralel (Port printer)* pada komputer tersebut. Sehingga apabila sedang sibuk bekerja tidak perlu meninggalkan pekerjaannya hanya untuk mengaktifkan peralatan tersebut.

Port merupakan satu set instruksi atau perintah sinyal dimana *microprocessor* atau CPU (*Central Processing Unit*) menggunakannya untuk memindahkan data dari atau ke piranti lain. Penggunaan umum *port* adalah untuk berkomunikasi dengan *printer*, *modem*, *keyboard* dan *display*. Kebanyakan *port* komputer adalah berupa kode digital, di mana tiap-tiap sinyal atau *bit* adalah berupa kode *biner* 1 atau 0.

Tugas akhir ini memanfaatkan fungsi dari *port* paralel dengan membuat *software* dan *hardware* sistem kendali perangkat listrik yang digunakan sebagai otomatisasi perangkat listrik AC (*Alternating Current*) menggunakan *timer*. Dalam pembuatan *software*, program aplikasi tersebut dibuat dengan menggunakan bahasa pemrograman Delphi dan *hardware* berupa rangkaian saklar elektronik (*electronic switch*) relai benda padat (*Solid State Relay*) atau SSR. Selain itu, saklar elektronik juga dikenal sebagai *opto-triac* yang nantinya rangkaian ini digunakan sebagai pengatur arus daya tinggi. Adapun untuk implementasinya akan dicoba pada otomatisasi bel jam sekolah, sehingga pada tiap pergantian jam pelajaran benar-benar tepat waktu dan SDM (Sumber daya Manusia) yang ada dapat dimaksimalkan pada hal lain.

1.2 Perumusan Masalah

Berdasarkan uraian latar belakang masalah, maka rumusan masalah yang dibahas adalah bagaimana membuat suatu program aplikasi *sistem* kendali bel listrik menggunakan PC melalui *port* paralel, dan pembuatan *interface* perangkat keras yang digunakan untuk menghubungkan PC dengan perangkat listrik dalam hal ini adalah bel sebagai tanda pergantian jam pelajaran pada suatu sekolah.

1.3 Batasan Masalah

Pembatasan masalah dalam tugas akhir ini adalah memfokuskan pembuatan sistem kendali perangkat listrik dari segi *hardware* maupun *software*, sehingga mempunyai batasan masalah sebagai berikut :

1. Program aplikasi *sistem* kendali perangkat listrik dirancang menggunakan Delphi 7.0, sebagai bahasa pemrograman yang bersifat GUI (*Graphic User Interface*), dan menggunakan file *library* “inpout32.dll”, yaitu piranti untuk mempermudah pemrograman dalam mengakses *port* paralel, akan tetapi tidak dibahas struktur file “inpout32.dll”.
2. *Database Management System* menggunakan Ms. Access, digunakan untuk menyimpan *setting* waktu.
3. Bersifat *single user*, yaitu hanya bisa dikontrol melalui satu komputer.
4. *Hardware* yang digunakan untuk penghubung PC dengan perangkat listrik menggunakan *optocoupler* dan *triac* sebagai saklar, yang hanya memanfaatkan satu output yang dihasilkan dari port tersebut sebagai pemicu saklar untuk mengaktifkan bel.

1.4 TUJUAN

Tujuan dari tugas akhir ini adalah merancang *software* dan *hardware* sistem kendali bel otomatis berbasis PC (*Personal Computer*) melalui *port* paralel. Sehingga isyarat pergantian pelajaran dapat tepat sesuai dengan yang dijadwalkan

1.5 MANFAAT

Manfaat yang mungkin didapat dari tugas akhir ini adalah sebagai berikut :

- a. Menambah pengetahuan dan memacu penulis untuk mempelajari sistem kontrol suatu perangkat listrik melalui *port* paralel pada komputer.
- b. Memudahkan dalam penggunaan bel sekolah yang biasanya dilakukan dengan manual dapat digantikan secara otomatis.
- c. Bel pergantian pelajaran dapat tepat waktu sesuai dengan yang dijadwal

BAB II

LANDASAN TEORI DAN TINJAUAN PUSTAKA

2.1 Interface Port Paralel Komputer

Menurut Budhi dan Dedy (2007), *Interface* adalah media yang dapat menghubungkan komputer dengan *hardware* luar, sehingga komputer dapat menginisialisasinya. Manfaat dari proses inisialisasi ini salah satunya komputer akan memberikan informasi alamat yang belum digunakan. Jika terdapat beberapa instruksi menggunakan alamat yang sama maka komputer secara otomatis akan mengabaikannya.

Pada proses *interfacing* ini komputer digunakan sebagai pusat kendali untuk mengendalikan piranti (*device*) di luarnya, seperti: mengendalikan bel, motor, LED/lampu, dan komputer juga dapat dimanfaatkan sebagai masukan sensor-sensor, saklar dan piranti *input* yang lain.

Untuk dapat berkomunikasi dengan dunia luar dalam teknik *interfacing*, komputer dapat memanfaatkan:

- a. Saluran paralel melalui *port* paralel atau sering disebut sebagai printer port (LPT).
- b. Melalui *port* serial yaitu yang sering disebut sebagai COM port.
- c. Melalui slot ekspansi, slot ekspansi adalah fasilitas dari sebuah PC (*Personal Computer*) untuk keperluan antarmuka.

Pada tugas akhir ini teknik yang digunakan untuk interface adalah yang melalui port paralel.

2.2 Port Paralel

2.2.1 Pengenalan Port Paralel

Menurut Afgianto (2002), *port* paralel adalah *port* data komputer yang dapat menerima masukan hingga 8 bit atau keluaran hingga 12 bit pada saat bersamaan, dengan hanya membutuhkan rangkaian *eksternal* sederhana untuk melakukan tugas tertentu. *Port* paralel ini terdiri dari 4 jalur kontrol, 5 jalur status, dan 8 jalur data, yang biasa disebut dengan *port* pencetak (*printer*)

Port paralel yang baru, distandarisasi dengan IEEE.1284 yang dikeluarkan pada tahun 1984. Standar ini mendefinisikan 5 macam mode operasi sebagai berikut:

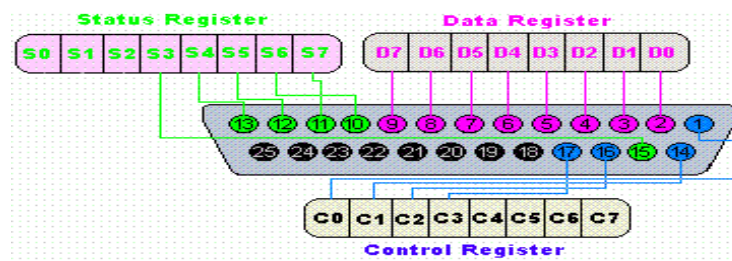
- a. Mode Kompabilitas.
- b. Mode *Nibble*.
- c. Mode *Byte*.
- d. Mode EPP (*Enhanced Parallel Port*).
- e. Mode ECP (*Extended Capability Port*).

Tujuan standar yang baru tersebut ialah untuk merancang penggerak (*driver*) dan piranti yang baru dan kompatibel antara satu dengan lainnya serta kompatibel mundur (*backwards*) dengan SPP (Standard Printer Port).

2.2.2 Diagram Pin *Port* Paralel

Menurut Sutadi (2002), ada dua macam konektor paralel *port*, yaitu 36 pin dan 25 pin. Konektor 36 pin dikenal dengan nama *Centronics* dan konektor 25 pin dikenal dengan nama DB-25. *Centronics* lebih dahulu ada dan digunakan daripada DB-25. DB-25 diperkenalkan oleh IBM (bersamaan dengan DB-9 untuk serial *port*) ketika memperkenalkan IBM PC pertama kali, yang bertujuan untuk menghemat tempat. Karena DB-25 lebih praktis, maka untuk koneksitor *port* paralel pada komputer sekarang hanya digunakan sebagai konektor pada *printer* (dan atau piranti luar lainnya).

Layaknya komponen elektronika *port* paralel dibutuhkan dengan konektor betina dan jantan. Pada komputer konektor *port* paralel yang terpasang adalah DB-25 betina, sehingga kabel penghubung kabel keluar adalah DB-25 jantan. (Gambar 2.1 : Pin konektor DB-25).



Gambar 2.1 Pin konektor DB-25

Dari 25 pin konektor DB-25 tersebut, hanya 17 pin yang digunakan untuk saluran pembawa informasi dan yang berfungsi sebagai *ground* 8 pin seperti dalam Tabel 2.1. Ketujuh belas saluran informasi itu terdiri dari tiga bagian, yakni data 8 *bit*; status 5 *bit*; dan kontrol 4 *bit*. *Bit* kontrol dan status berfungsi dalam “jabat tangan” dalam proses penulisan data ke *port* paralel, dengan maksud agar dalam pertukaran data menjadi *valid*. Jabat tangan ini hanya digunakan untuk *hardware* yang akan dihubungkan, tidak untuk *software*.

Tabel 2.1. Tabel Fungsi Pin Port Paralel DB-25

Pin No	Signal SPP	Arah	Register - bit
1	nStrobe	In/Out	Control-0
2	Data 0	Out	Data-0
3	Data 1	Out	Data-1
4	Data 2	Out	Data-2
5	Data 3	Out	Data-3
6	Data 4	Out	Data-4
7	Data 5	Out	Data-5
8	Data 6	Out	Data-6
9	Data 7	Out	Data-7
10	nAck	In	Status-6
11	Busy	In	Status-7
12	Paper-Out	In	Status-5
13	Select	In	Status-4
14	Linefeed	In/Out	Control-1
15	nError	In	Status-3
16	nInitialize	In/Out	Control-2
17	nSelect-Printer	In/Out	Control-3
18-25	Ground	-	-

2.3 Pengertian Sistem

Banyak pendapat para ahli tentang sistem yaitu sebagai berikut :

- Sistem adalah seperangkat unsur yang melakukan sesuatu kegiatan atau beberapa tujuan dengan mengolah data suatu energi dalam jangka waktu

tertentu guna menghasilkan informasi dan energi. (Mudrick dan Ross : dalam Jogiyanto, 2001).

- b. Bahwa unsur-unsur sistem terdiri dari :
 - Masukan (*Input*) yang memprosesnya dengan perhitungan, penggabungan unsur data, pemutakhiran jumlah (*up-dating account*), dll.
 - Keluaran (*Output*).

(Scot : Dalam Anwar, 2003).

- c. Bahwa sebuah sistem harus mempunyai organisasi, hubungan, integrasi dan tujuan-tujuan yang sama.(Sidharta : Dalam Anwar, 2003).

Dari beberapa definisi di atas, maka dapat diambil suatu kesimpulan, yaitu sistem adalah totalitas himpunan bagian yang satu sama lain berinteraksi dan bersama-sama untuk mencapai tujuan atau sekelompok tujuan dalam suatu lingkungan. Sedangkan bagian sistem yang biasa disebut juga dengan sub-sistem, yang merupakan suatu kumpulan dari unsur tertentu, namun dalam mencapai tujuan semua sub-sistem bekerja dalam mencapai tujuan dan keharmonisan dan keteraturan yang pasti.

2.4 Pengertian Database

Menurut Kadir (2004), *database* merupakan suatu bentuk pengelolaan data yang ditujukan agar pengaksesannya terhadap data dapat dilakukan dengan mudah. Sistem yang ditujukan untuk menangani *database* biasa disebut DBMS (*Database Management System*). Dengan menggunakan DBMS, pemakai dapat melakukan hal-hal seperti berikut dengan mudah :

- a. Menambah data
- b. Menghapus data
- c. Mengubah data
- d. Mencari data
- e. Menampilkan data dengan kriteria data
- f. Mengurutkan data

Sedangkan menurut Fatansyah (1999), *database* atau basis data terdiri dari dua kata yaitu basis dan data. Basis dapat diartikan sebagai markas atau gudang,

tempat bersarang atau berkumpul, sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya. Basis data sendiri dapat diartikan dalam sejumlah sudut pandang seperti :

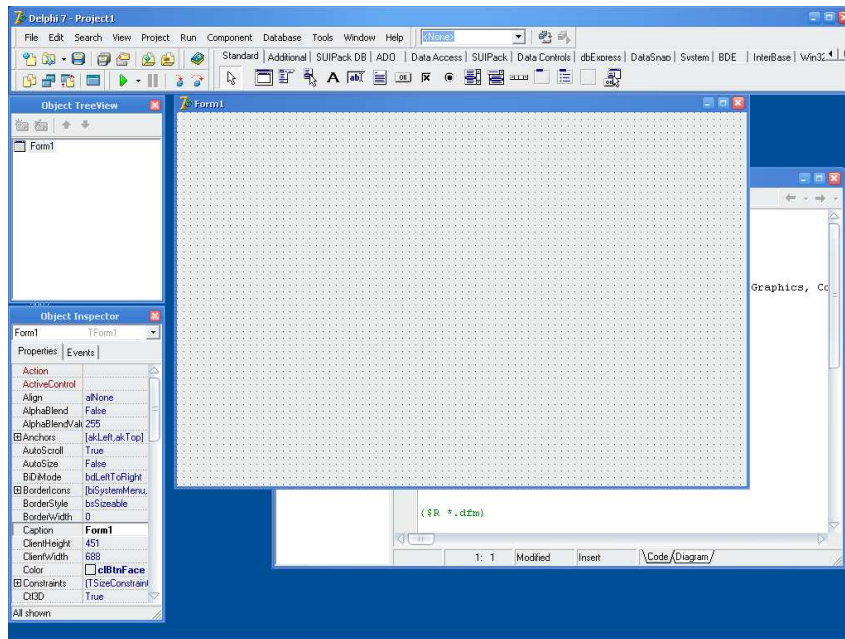
- a. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali secara cepat dan mudah.
- b. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redudansi*) yang tidak perlu untuk memenuhi berbagai kebutuhan.
- c. Kumpulan *file*, tabel atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.5 Pengertian Delphi

Menurut Kadir (2004), Delphi merupakan sebuah piranti pengembangan aplikasi berbasis windows yang dikeluarkan oleh Borland *International*. Perangkat lunak ini sangat terkenal di lingkungan pengembang aplikasi karena mudah untuk dipelajari dan dapat digunakan untuk menangani berbagai hal, dari aplikasi matematika, permainan (*games*), hingga *database*. Pada penanganan *database*, Delphi menyediakan fasilitas yang memungkinkan pemrogram dapat berinteraksi dengan *database* seperti dBase, Paradox, Oracle, MySQL, dan Access.

Pada tugas akhir ini database yang digunakan adalah Access, karena lebih *familiar* dan hanya membutuhkan sedikit data serta dalam proses koneksi dengan Delphi lebih mudah dibandingkan dengan database lain.

Beberapa bagian pokok yang dimiliki Delphi yaitu *Integrated Development Environment* (IDE) Delphi seperti gambar 2.2, pemrograman *Visual Component Library* (VCL), *Object Linking and Embedding* (OLE) dan *Dinamic Link Library* (DLL). (Martina, 2001)



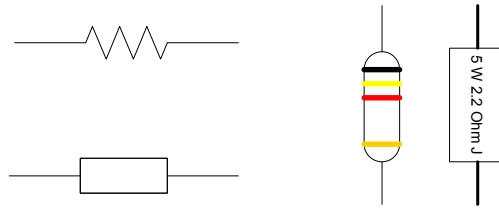
Gambar 2.2 Tampilan Program utama Delphi

Di dalam pembuatan program kontrol tidak lepas dari bagian-bagian pokok tersebut di atas. Delphi merupakan bahasa pemrograman yang ber-object Pascal yang disempurnakan seperti dalam hal pengolahan string, penambahan prosedur dan fungsi, dan sebagainya. (Martina, 2001)

Agar dapat menggunakan slot dan port pada komputer, program Delphi telah menyediakan kode-kode assembler untuk dapat berhubungan dengan peralatan luar. (Mikrodata, 2001)

2.6 Resistor

Resistor atau yang sering disebut tahanan, adalah salah satu komponen elektronika yang digunakan sebagai penghambat, baik arus(I) ataupun tegangan (V) yang akan diinputkan atau dikeluarkan ke *sirkuit* atau rangkaian lain. Tahanan resistor diberi kode berupa pita warna yang melingkari badannya. Warna pita menunjukkan tahanan dan toleransi. Warna warna tersebut melambangkan angka-angka (Aref, 2001):

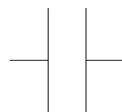


Gambar 2.3 Simbol dan bentuk komponen resistor

2.7 Kapasitor

Kapasitor adalah suatu jenis komponen elektronika yang bisa dikatakan multifungsi karena komponen ini tergantung pada jenis rangkaian atau sirkuit dimana ditempatkan. Misalnya, bila kapasitor diletakkan pada sebuah rangkaian regulator tegangan, maka fungsi kapasitor adalah sebagai filter atau penyaring tegangan AC yang tidak dikehendaki supaya tidak masuk ke rangkaian. Apabila dipasangkan pada rangkaian booster atau filter frekwensi, kapasitor ini berfungsi untuk menyaring atau membuang frekwensi-frekwensi yang tidak diinginkan (Aref, 2001).

Kapasitor memiliki satuan *farad* dan besar kapasitasnya ditulis dengan angka pada lapisan luarnya.



Gambar 2.4 Simbol kapasitor

2.8 Transistor

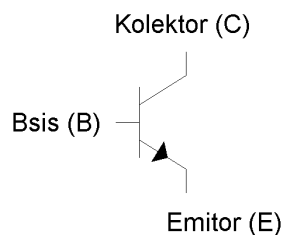
Transistor adalah suatu komponen elektronika yang bisa dikatakan sebagai komponen *vital* dan miltifungsi. Transistor biasanya digunakan sebagai penguat, baik penguat pada frekwensi rendah maupun frekwensi tinggi. Namun ada juga yang memfungsikan transistor ini sebagai *switch*/saklar otomatis pada suatu rangkaian *switching*. Berbeda dengan elektronik yang lain, transistor memiliki 3 kaki yang bisa disebut sebagai *basis*, *emitor* dan *kolektor*. Setiap transistor

memiliki susunan kaki yang berbeda. Pemasangannya pun tidak boleh sembarangan karena bila pemasangannya salah atau terbalik, transistor tersebut akan rusak secara permanen atau bahkan meledak. (Aref, 2001).

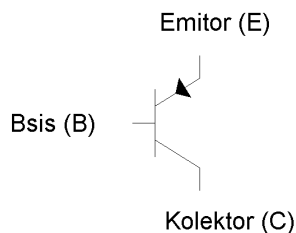
Notasi transistor ditulis dengan kode yang berlainan untuk setiap pabrik yang mengeluarkannya. Khusus untuk transistor buatan Jepang atau Taiwan memiliki notasi sebagai berikut :

- a. SA berarti transistor jenis PNP untuk frekuensi tinggi
- b. SB berarti transistor jenis PNP untuk frekuensi rendah
- c. SC berarti transistor jenis NPN untuk frekuensi tinggi
- d. SD berarti transistor jenis NPN untuk frekuensi rendah

Kualitas transistor bermacam-macam. Misalnya A,B,C,D dan seterusnya. Urutan abjad ke kanan semakin baik. Jadi B lebih baik daripada A, begitupula C lebih baik daripada B (Aref, 2001).



Gambar 2.5 Simbol transistor NPN



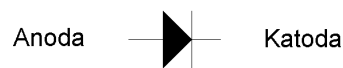
Gambar 2.6 Simbol transistor PNP

2.9 Dioda

Sebuah dioda *semikonduktor* dapat dipakai untuk menggantikan tabung hampa. Dioda semikonduktor dapat dibuat dari beberapa jenis N yang bersifat kelebihan elektron lalu disambung dengan jenis P yang kekurangan elektron.

Bahan jenis N dan P tersebut dihasilkan dengan cara penodaan (*doping*) pada bahan *semikonduktor* germanium dan silikon. Cara penyambungan bahan ini adalah dengan cara kimia, bukan dengan cara mekanik, yaitu proses pemanasan sehingga berdifusi, kemudian kedua bahan itu tersusun menjadi sebuah susunan kristal tunggal.

Apabila diteliti akan tampak bahwa pada tempat-tempat yang berdekatan dengan sambungan itu akan terjadi perembesan sedikit., baik elektron maupun hole dari wilayah masing-masing. Beberapa diantaranya akan bergabung menjadi satu yang disebut rekombinasi (Aref, 2001).

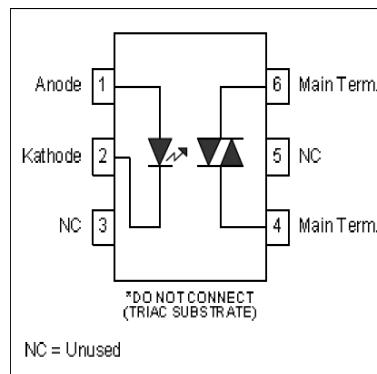


Gambar 2.7 Simbol Dioda

2.10 Optocoupler dan Triac

2.10.1 Optocoupler

Optocoupler atau *optoisolator* merupakan komponen elektronika yang banyak dipakai untuk mengatasi perpindahan informasi dari tegangan rendah ke tegangan tinggi atau sebaliknya. Sebuah *optocoupler* terdiri dari sebuah LED (GaAs) yang memancarkan cahaya infra merah dan sebuah *phototransistor* NPN yang terbuat dari silikon. Pada rangkaian optocoupler terjadi isolasi tegangan yang tinggi antara sinyal masukan dan keluaran. Dalam penerapannya, optocoupler dapat menggantikan relai atau trafo pada rangkaian digital. Gambar berikut menunjukkan simbol optocoupler dengan keluaran triac (Anonim,2002):



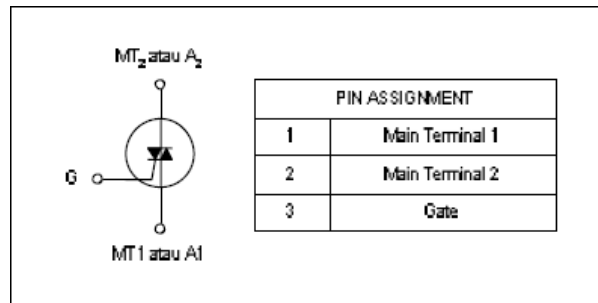
Gambar 2.8 Simbol Optocoupler

Beberapa karakteristik yang dimiliki optocoupler yaitu :

1. *Current Tranfer Ratio* (CTR) adalah perbandingan antara arus masukan pada LED dengan arus keluaran pada transistor.
2. Tegangan isolasi (*Viso*) adalah tegangan maksimum yang diperbolehkan antara masukan dan keluaran, biasanya anantara 500-5kV.
3. Vce adalah tegangan maksimum pada keluaran, biasanya antara 30-70 Volt.
4. I_F adalah arus masukan pada LED, biasanya antara 40-100 mA.
5. Waktu naik atau turun pada fototransistor keluaran biasanya adalah 2-5 detik.

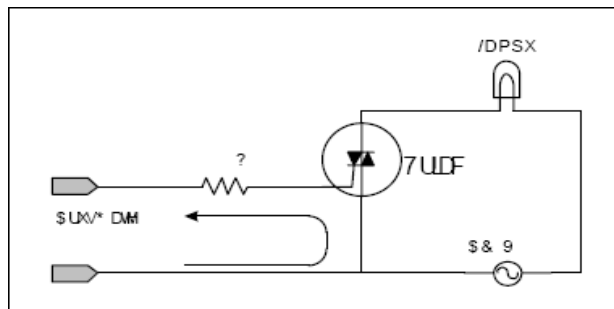
2.10.2. TRIAC

Triac adalah kependekan dari *Triode Alternating Curre Switch* atau saklar triode untuk arus bolak-balik. Simbol komponen TRIAC dan keterangan masing-masing kaki TRIAC seperti gambar berikut :



Gambar 2.9 Simbol TRIAC dan keterangan kaki

Rangkaian dasar pemakaian triac terlihat dalam gambar 2.8. Kaki MT1 dan MT2 merupakan saklar yang mengatur aliran arus beban yang berasal dari sumber tegangan bolak – balik (AC). Dalam keadaan normal kaki MT1 dan MT2 tidak terhubung, sehingga tidak ada arus yang mengalir.



Gambar 2.10 Rangkaian TRIAC sederhana

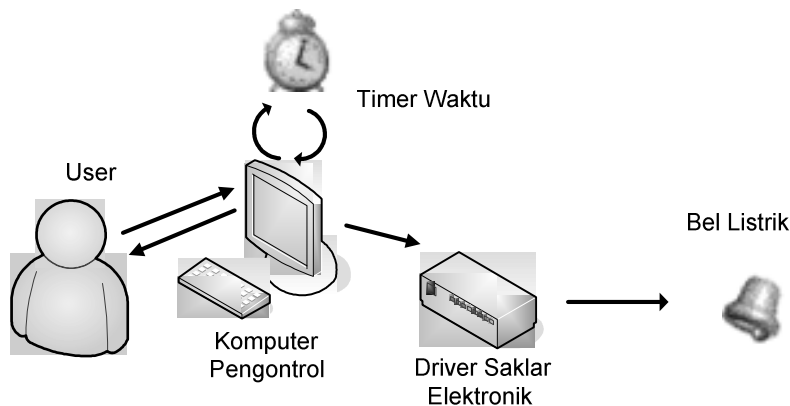
Saat arus gate mengalir, MT1 akan terhubung ke MT2 dan mengalirkan arus beban. Arus gate hanya diperlukan untuk menghubungkan MT1 dan MT2, setelah itu MT1 akan tetap terhubung ke MT2 meskipun sudah tidak ada arus gate lagi. Pemberian arus gate sesaat untuk menghubungkan MT1 dan MT2 dikatakan sebagai penyulut (trigger) triac (*Anonim*, 2002).

BAB III

PERANCANGAN SISTEM

3.1 Gambaran Umum Sistem

Perancangan sistem kendali bel listrik otomatis mencakup keseluruhan kebutuhan *software* maupun *hardware*. Gambar 3.1 Menunjukkan skema sistem kendali perangkat bel secara umum.



Gambar 3.1. Skema sistem kendali bel listrik otomatis

Berdasarkan gambar 3.1 Skema sistem kendali perangkat bel listrik terlihat bahwa sistem mempunyai beberapa blok fungsional yaitu :

- a. *User*, merupakan pengguna yang akan mengatur kerja sistem.
- b. Komputer pengontrol, digunakan sebagai jalannya program aplikasi yang mengontrol *driver* saklar elektronik yang menggunakan sistem *timer*.
- c. *Driver* saklar elektronik, berfungsi sebagai saklar yang menghubungkan antara PC dengan perangkat yang akan dikontrol dengan komputer. *Driver* saklar elektronik ini mempunyai 8 (delapan) keluaran, tetapi pada dalam tugas akhir ini hanya memanfaatkan satu keluaran saja..
- d. Peralatan listrik dapat berupa lampu pijar, televisi, kipas angin, atau peralatan yang lain, tetapi dalam tugas akhir ini perangkat listrik yang digunakan adalah bel.

Perancangan *hardware* meliputi pembuatan *driver* saklar bel yang menghubungkan PC dengan peralatan listrik yang mempunyai tegangan dan arus

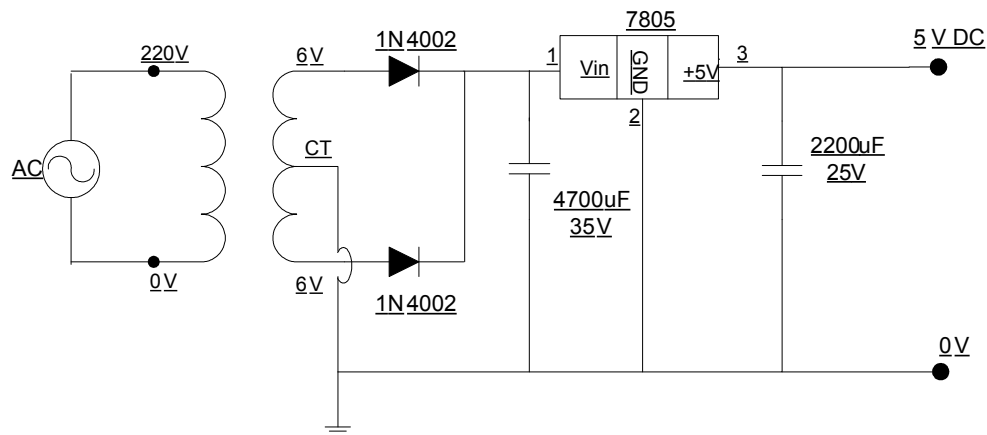
AC (*Alternating Current*). Perancangan *software* meliputi program sistem kendali perangkat listrik yang dibuat menggunakan Delphi 7.0 dan mempunyai *database* yang dibuat di Microsoft Access

A. Perancangan *Hardware* (Perangkat Keras).

Perancangan perangkat keras berupa penyusunan komponen-komponen elektronika menjadi satu kesatuan sistem rangkaian yang bisa bekerja sesuai dengan fungsinya. Dalam perancangan *hardware* terdapat dua bagian yang penting, antara lain :

1. Catu Daya.

Suatu rangkaian dapat bekerja dengan baik apabila adanya catu daya, dalam hal ini daya atau energi listrik arus searah. Gambar 3.2 Skema catu daya, menunjukkan rangkaian yang digunakan untuk mensuplai *driver* saklar elektronik.



Gambar 3.2 Skema catu daya

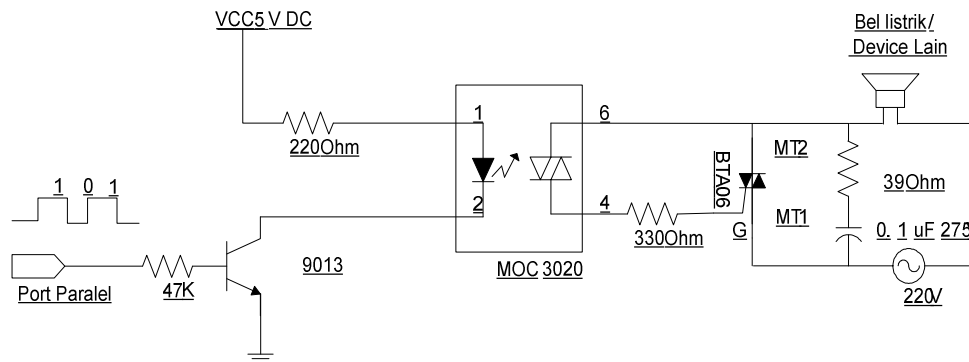
Gambar 3.2 menunjukkan rangkaian catu daya dari trafo digunakan tegangan 6 volt, tegangan ini digunakan untuk mensuplai IC 7805 pada pin 1. Pada pin ke-3 IC 7805 akan menghasilkan tegangan sebesar sebesar 5 volt. Keluaran sebesar 5 volt tersebut akan digunakan untuk mensuplai kebutuhan vcc *driver* elektronik.

2. Driver Saklar Elektronik.

Saklar elektronik (*electronic switch*) mempunyai nama lain yaitu relai benda padat (*Solid State Relay*) atau SSR. Selain itu, saklar elektronik juga dikenal sebagai *opto-triac*. Disebut *opto-triac* karena konstruksi dalamnya menggunakan gabungan antara *triac* dengan *optocoupler* sebagai komponen utama rangkaian pemicunya.

Optocoupler merupakan komponen elektronika yang digunakan untuk memindahkan informasi dari tegangan rendah ke tegangan tinggi. Sedangkan triac merupakan kependekan dari *triode alternating current switch* atau saklar triode untuk arus bolak-balik.

Biasanya SSR ini digunakan sebagai antarmuka antara rangkaian pengendali yang bertegangan DC (*direct current*) rendah dengan rangkaian jala-jala AC (*Alternating Current*) yang bertegangan jauh lebih tinggi.



Gambar 3.3 Skema rangkaian internal saklar elektronik

Berdasarkan Gambar 3.3, saklar elektronik mempunyai prinsip kerja sebagai berikut : keluaran dari *port* paralel PC akan mengaktifkan resistor 47 K Ω yang berfungsi sebagai pembatas arus untuk *basis* transistor. Transistor tersebut berfungsi sebagai saklar untuk *katode* dari resistor 220 Ω yang terhubung MOC3020. Kemudian arus Vcc akan mengalir melalui resistor 220 Ω dan akan mengaktifkan MOC3020 melalui *anode*-nya (pin1). Resistor disini berfungsi sebagai pembatas arus untuk anoda MOC3020 (pin 1).

Keluaran MOC3020 akan digunakan untuk memicu *triac* BTA06 yang dilewati arus AC (*Alternating Current*). *Triac* tersebut akan aktif jika MOC3020

mengeluarkan nilai logika 1 (maksimum 6 Volt DC). Apabila dibandingkan dengan relai elektromagnetik (*electromagnetic relay* – EMR), SSR mempunyai beberapa kelebihan antara lain :

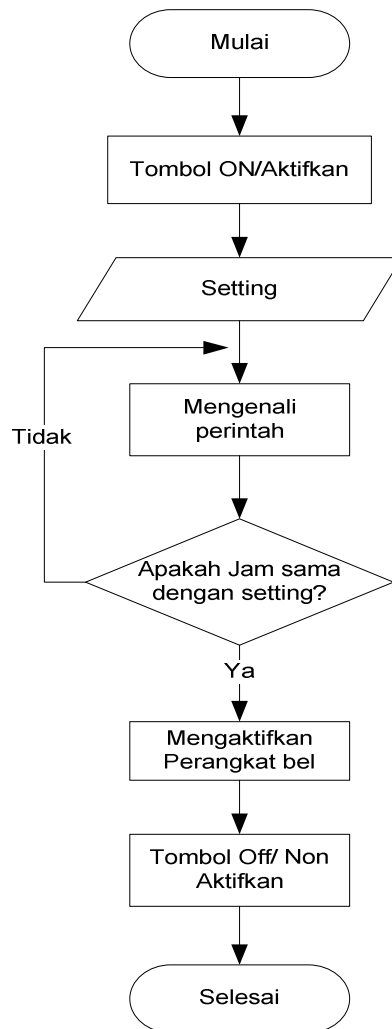
- a. Lebih dapat diandalkan dan lebih tahan lama karena tidak mempunyai bagian yang bergerak untuk pensaklaran.
- b. Sesuai dengan rangkaian yang menggunakan rangkaian terpadu (*Integrated Circuit* atau IC).
- c. Tidak menimbulkan *interferensi* elektromagnetik.
- d. Lebih tahan terhadap kejutan dan getaran.
- e. Mempunyai waktu tanggapan yang lebih cepat.

Saklar elektronik yang digunakan dalam perangkat keras ini mempunyai tegangan DC masukan maksimum 6 volt dengan keluaran berupa tegangan AC maksimum 240 volt dan dengan arus maksimum 12 ampere.

B. Perancangan Software (Perangkat Lunak)

Perancangan *software* dimaksudkan agar komputer dapat bertugas mengirimkan data ke *driver* saklar elektronik. *Software* sistem kendali bel jam sekolah ini dibuat menggunakan bahasa pemrograman Delphi 7.0 yang berbasis GUI (*Graphic User Interface*), dan menggunakan komponen *suipack* untuk desain *form*, dan Microsoft Access untuk menyimpan *database setting* hari, waktu, nama jadwal, serta jumlah bel yang diinginkan. Sedangkan untuk mempermudah pemrograman dalam mengakses *port* paralel digunakan file *library* "inpout32.dll".

Agar program dapat diimplementasikan sesuai dengan kegunaannya maka perlu merencanakan dahulu alur program yang akan dibuat. Pada gambar 3.4 berikut menggambarkan alur program aplikasi sistem kendali perangkat bel jam sekolah yang akan dirancang.



Gambar 3.4 Rancangan alur program

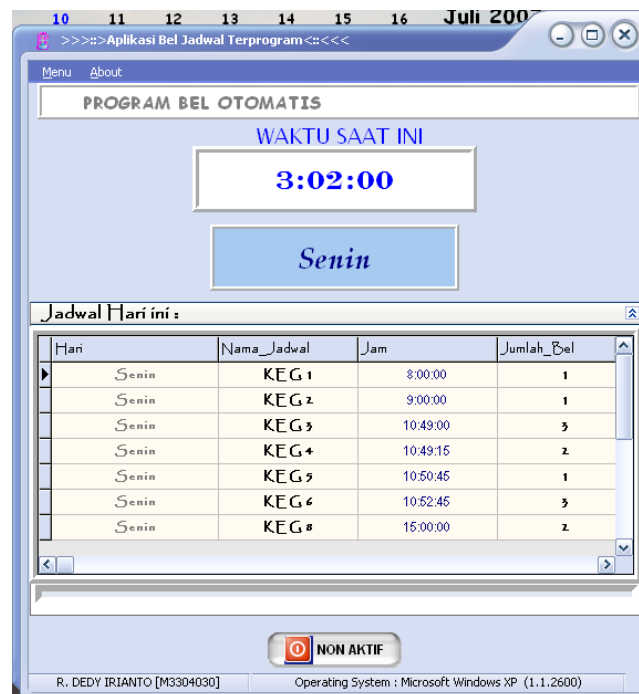
Maksud dari alur tersebut adalah, ketika *user* mulai mengaktifkan program dengan menekan tombol “Aktifkan”, program akan mengenali perintah yang berupa set waktu jam yang akan menginstruksikan kapan komputer mulai mengeluarkan sinyal ke *port* paralel sehingga akan mengaktifkan *driver* saklar elektronik. Alur tersebut akan berjalan terus sampai program dinon-aktifkan.

Perancangan aplikasi bel otomatis terprogram berbasis PC, dirancang memanfaatkan *form* yang telah disediakan dari *software* Delphi. Adapun untuk tampilan *form* utama adalah seperti berikut :



Gambar 3.5 Tampilan Program

Setelah program diaktifkan akan nampak seperti tampilan berikut :



Gambar 3.6 Tampilan Program Aktif

Tampilan pada awal program merupakan informasi mengenai nama jadwal yang terdapat pada hari saat program aktif, jam jadwal, serta jumlah isyarat bel yang akan dibunyikan.

Pada program ini yang sangat berperan utama untuk mengatur nama-nama jadwal, jam jadwal, dan jumlah isyarat bel yang diinginkan terdapat pada *form setting*. Untuk tampilan pada *form setting* ini seperti berikut :

Hari	Nama Jadwal	Jam	Jumlah Bel
Senin	KEG1	8:00:00	1
Senin	KEG2	9:00:00	2
Senin	KEG3	10:00:00	3
Senin	KEG4	11:00:00	2
Senin	KEG5	12:00:00	1

Gambar 3.7 Tampilan Form Setting

Pada form ini terdapat tombol *combobox* yang berisikan nama-nama hari dalam satu minggu, apabila tombol ini diaktifkan maka data yang muncul dari *database* dalam *component* DBGrid di bawahnya sesuai dengan hari yang diinginkan. Tetapi apabila ingin melihat keseluruhan data yang terdapat dalam *database* maka tinggal mengklik tombol “Tampil Semua”. Pada sebelah kiri *form* merupakan daftar isian yang harus diisi untuk menambah nama kegiatan, jam, dan jumlah bel pada hari yang diinginkan. Dan juga terdapat tombol *update* untuk merubah data yang diinginkan dengan sebelumnya menentukan data mana yang akan dirubah dengan cara double klik data yang akan dirubah dari *component* DBGrid.

C. Perancangan Database

Program bel otomatis ini dirancang menggunakan *database* untuk menyimpan setting hari, nama jadwal, jam, serta jumlah isyarat bel. *Database* dibuat

menggunakan Microsoft Access. Dalam sistem ini *database* hanya menggunakan satu tabel yang berisikan field seperti pada tabel 4.1 berikut :

<i>Field</i>	<i>Type</i>	Keterangan
Hari	<i>Text</i>	<i>Setting</i> Hari
Nama_Jadwal	<i>Text</i>	<i>Setting</i> Nama Jadwal
Jam	<i>Text</i>	<i>Setting</i> Jam jadwal
Jumlah_Bel	<i>Text</i>	<i>Setting</i> Jumlah isyarat bel (1-3)

Tabel 3.1 Tabel database yang digunakan

3.2 *Library Research*

Mempelajari kepustakaan yang berhubungan dengan pembuatan program sistem kendali perangkat listrik (Delphi, Ms. Access, dll) dan pembuatan perangkat keras (*port* paralel, *triac*, *optocoupler*, dll).

3.3 Alat Penelitian

Peralatan yang digunakan untuk menunjang dalam penelitian dan pengembangan sistem kendali perangkat listrik ini, antara lain sebagai berikut :

- Seperangkat komputer dengan spesifikasi :
 - Processor* : PIV 2,66 GHz
 - Memory* : DDR 256 MB
 - VGA : 32 MB
 - Hardisk* : 40 GB
 - Media masukan : Papan ketik (*keyboard*) dan *mouse*
 - Kartu suara : Realtek AC '97
 - Media *Display* : Monitor Samsung 15"
- Software* Borland Delphi 7.0 yang digunakan sebagai *visual* pemrograman berbasis GUI (*Graphic User Interface*), serta menggunakan *file library* "inout32.dll" untuk mempermudah pemrograman akses *port* paralel.
- Microsoft Access digunakan dalam pembuatan *database* untuk menyimpan *setting* waktu, tanggal dan nama ruang.

4. Komponen elektronika untuk merancang catu daya dan *driver* saklar elektronik, seperti resistor, transitor, transformator, dioda, kapasitor, *triac* BTA06, *Optocoupler* MOC3020, dll.

BAB IV

PENGUJIAN, IMPLEMENTASI, DAN PEMBAHASAN

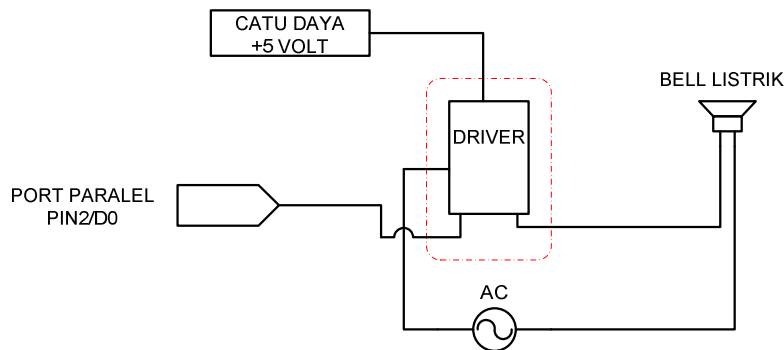
Evaluasi Pengujian Sistem

4.1.1 Pengujian Perangkat Keras

Pengujian ini dilakukan untuk mengetahui apakah perangkat dapat berkerja dengan baik. Langkah-langkah yang dilakukan dalam pengujian ini seperti berikut :

a. Pengujian *Driver* Saklar Elektronik

Bel otomatis ini hanya akan memanfaatkan satu keluaran dari dua belas keluaran yang mampu dilakukan oleh *port* paralel sehingga hanya memerlukan satu *driver* saklar elektronik. Driver tersebut akan aktif ketika diberi masukan +5 volt dari *port* paralel komputer. Gambar 4.6 menunjukkan skema *driver* saklar dengan keluaran yang akan dihubungkan ke perangkat listrik sedangkan Tabel 4.2 menunjukkan pengujian *driver* saklar dengan masukan 0 volt dan 5 volt dengan *device* bel listrik.



Gambar 4.1 *Driver* sakral elektronik

Berdasarkan gambar 4.1 masukan berasal dari *port* paralel yang nantinya akan menghasilkan nilai *output* 5 volt dan 0 volt. Apabila *port* paralel mempunyai nilai 5 volt maka akan mengaktifkan *driver* dan mengaktifkan *device* bel, begitu juga sebaliknya apabila *port* bernilai 0 volt maka *device* bel akan non aktif.

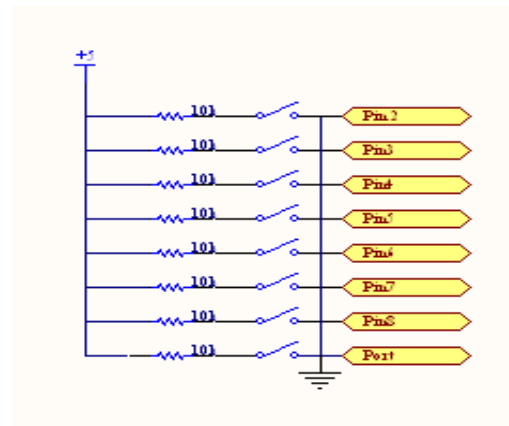
Pengujian *driver* saklar elektronik digunakan tegangan dari catu daya yang bernilai 5 volt dan 0 volt, hasil respon dari *device* bel dengan masukan 0 dan 5 volt terdapat dalam Tabel 4.2.

Device	Respon	
	Masukan 0 volt	Masukan 5 volt
Bel Listrik	Mati	Nyala

Tabel 4.1 Hasil Pengujian driver

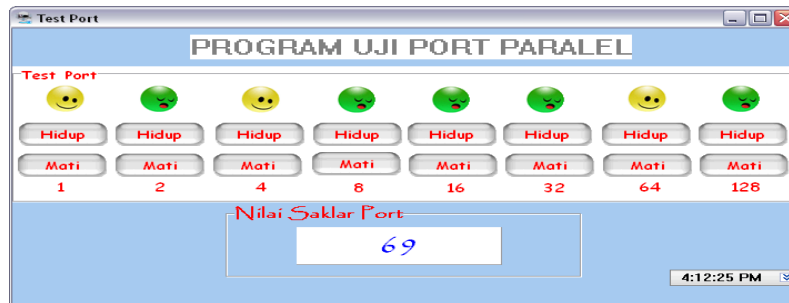
b. Pengujian port paralel

Pengujian port paralel dilakukan dengan tujuan mengecek ketepatan dalam pengalamatan untuk *input* logika yang akan diberikan ke perangkat yang akan dikontrol dalam hal ini *driver* saklar eletronik. Sehingga *driver* benar-benar dapat bekerja sesuai dengan yang kita inginkan. Pengujian *input port* paralel dilakukan seperti pada gambar 4.2 berikut :



Gambar 4.2 Skema rangkaian *input port* paralel

Pin 2-9 merupakan jalur data *port* paralel dengan lebar data 8 bit (D0-D7). Sesuai dengan gambar di atas, semua jalur data berlogika tinggi dengan kode biner 11111111(255). Jika saklar S2, S4, S5, S6, dan S7 dihubungkan, maka pin 3, 5, 6, 7,dan 9 berlogika rendah dengan kode biner 10100010 atau berharga desimal 69. Tampilan program ditunjukkan seperti pada gambar 4.3 berikut :



Gambar 4.3. Tampilan program uji input *port parallel*

Pengujian Sistem Bel Otomatis

Untuk mengetahui bahwa sistem dapat berjalan dengan baik atau tidak, maka dilakukan pengujian *sistem* secara keseluruhan dengan menggabungkan perangkat lunak dan perangkat keras. Pengujian dilakukan dengan cara memberikan perintah kemudian mengamati keberhasilan atas perintah yang diberikan.

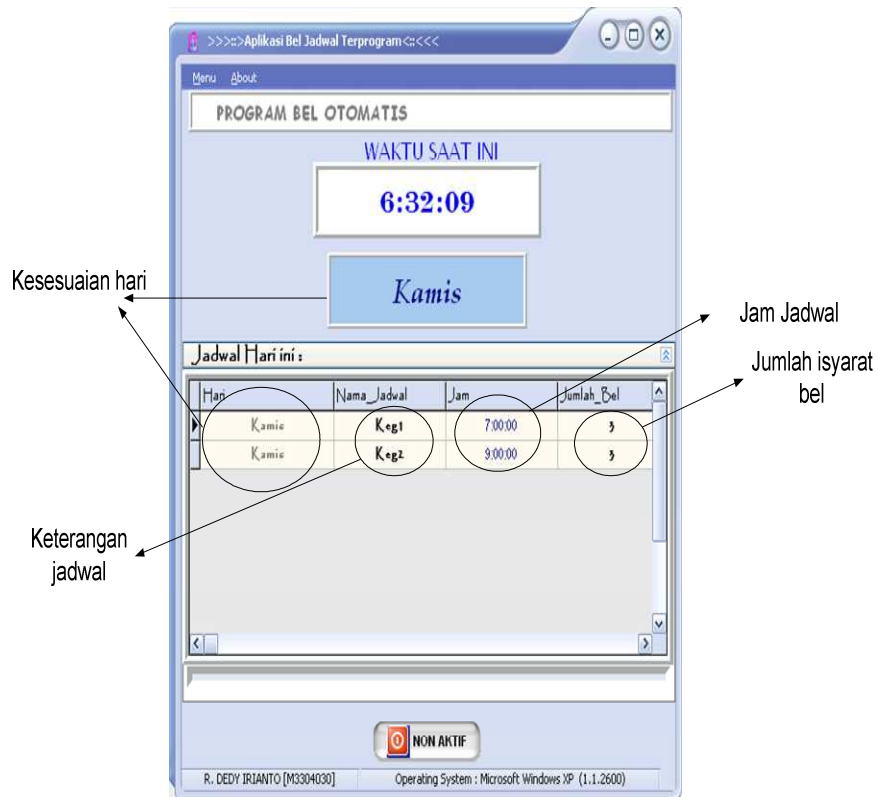
a. *Main Form*

Pada tampilan program merupakan *main form* yang menampilkan jadwal yang telah disimpan pada *database*. Data yang ditampilkan sesuai dengan hari pada komputer ketika program diaktifkan. Pada data tersebut tercantum. Setelah *device* dipasang dan program diaktifkan, maka device akan merespon perintah pengaktifan sesuai dengan hari dan jam yang disimpan pada *database*. Pada tabel 4.3 berikut merupakan hasil dari uji sistem ini setelah dilakukan manipulasi pada *setting* hari dari komputer.

Hari	Jam Jadwal	Jumlah isyarat	Respon
Senin	07:00	3	Hidup dengan 3 kali isyarat
Selasa	07:02	2	Hidup dengan 2 kali isyarat
Rabu	07:05	1	Hidup dengan 1 kali isyarat
Kamis	07:07	2	Hidup dengan 2 kali isyarat
Jumat	07:09	1	Hidup dengan 1 kali isyarat
Sabtu	07:11	3	Hidup dengan 3 kali isyarat
Minggu	07:15	2	Hidup dengan 2 kali isyarat
Senin	08:00	3	Hidup dengan 3 kali isyarat

Tabel 4.2 Hasil pengujian Sistem dan *Device*

Device akan aktif ketika dalam jadwal ada kesesuaian antara hari dan jam dari komputer yang digunakan dan dari jadwal yang telah disimpan pada database. Namun ketika program dijalankan namun tidak ada jadwal yang tersimpan dalam *database* maka pada hari saat program ini dijalankan *device* tidak akan diaktifkan.

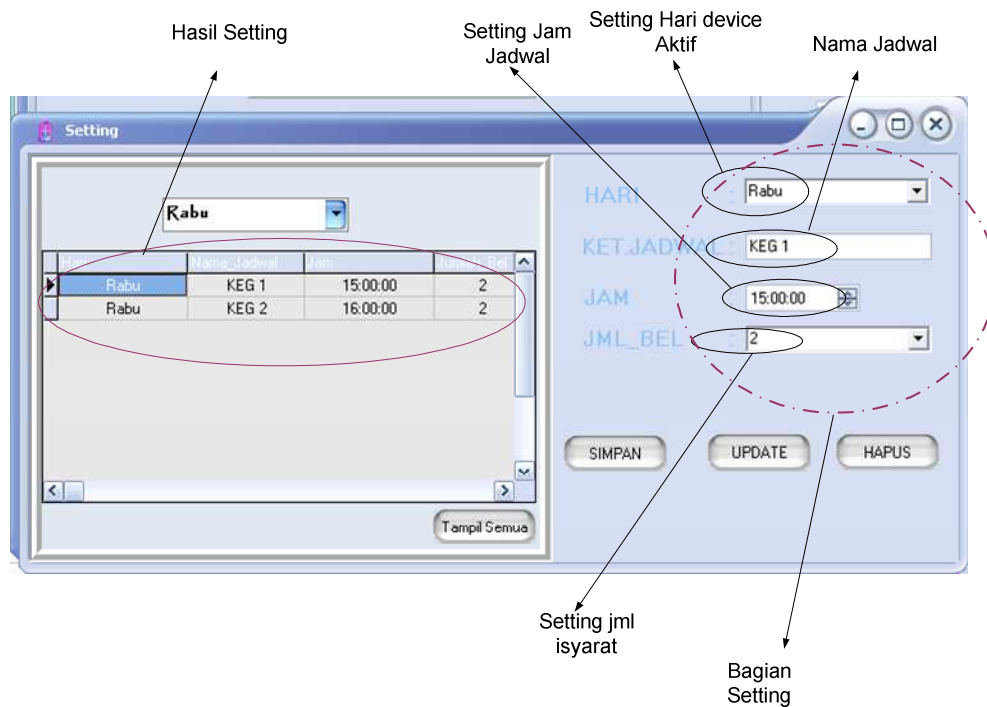


Gambar 4.4 Tampilan Program yang telah aktif

Dari gambar 4.4 akan memerintahkan *device* aktif pada jam 7.00 dengan 3 kali isyarat dan pada jam 9.00 dengan 3 kali isyarat pada tiap hari kamis. Tampilan yang diberikan pada tiap-tiap harinya akan berbeda, data yang disajikan tergantung pada data yang telah disimpan pada *database*.

b. *Form Setting*

Form inilah yang berperan memberikan masukan data ke *database*. Setelah pengisian data pada bagian setting selesai hasil akan ditampilkan pada bagian hasil setting



Gambar4.5 Form setting saat penyetingan

Keunggulan dan Kelemahan Sistem

Keunggulan

Sistem yang dirancang dapat di implementasikan dalam kehidupan sehari-hari untuk sebuah instansi misal sekolah, pabrik, rumah sakit. Juga terdapat beberapa keunggulan atau kemudahan dari rancangan sistem yang dicapai, antara lain :

- Sistem kendali *device* ini dirancang berdasarkan GUI (*Graphic User Interface*), sehingga pengoperasian sistem sangat mudah.
- Sistem ini mempunyai cara kerja otomatisasi perangkat listrik (*device*) menggunakan *timer*. *Device* akan aktif setelah ada kesesuaian antara

hari serta jam dari komputer dengan data yang tersimpan dalam *database*.

- c. Sistem kendali perangkat listrik sebenarnya dapat di implementasikan pada perangkat listrik AC (*Alternating Current*) lainnya, seperti lampu pijar, kipas angin, televisi, komputer, dan perangkat listrik lainnya.
- d. *Driver* saklar elektronik yang digunakan untuk menghubungkan PC dengan perangkat listrik dibuat menggunakan sistem *opto-triac* yang memiliki tingkat respon dan kestabilan sistem yang tinggi.

Kelemahan

Meskipun banyak kemudahan yang diperoleh dari hasil rancangan sistem ini, tentunya masih banyak kelemahan antar lain :

- a. Sistem yang dirancang masih hanya memanfaatkan satu dari dua belas output yang dapat dihasilkan dari *port* paralel. Seharusnya masih dapat dikembangkan lagi menjadi berapa *device* yang dikontrol
- b. Sistem dirancang menggunakan pemrograman yang menggunakan *interface port* paralel, sehingga apabila dalam PC tidak terdapat komponen *port* paralel sistem tidak dapat diterapkan.
- c. Karena sistem berjalan dalam PC, sehingga untuk memperoleh kegunaan dari sistem yaitu otomatisasi pada bel, *driver* saklar harus selalu terhubung dengan PC dan sistem program dijalankan (dieksekusi). Oleh sebab itu apabila PC tidak dalam keadaan ON maka sistem juga tidak dapat berjalan sebagaimana mestinya

BAB V

PENUTUP

5.1. Kesimpulan

Setelah melakukan analisa dan pembahasan yang telah diuraikan pada bab sebelumnya, maka dapat diambil beberapa kesimpulan antara lain :

1. Pada aplikasi jadwal bel omatis ini telah berhasil dilakukan perancangan *hardware* dan *software* sistem otomatisasi perangkat listrik berupa bel menggunakan PC melalui *port* paralel. *Software* menggunakan GUI (*Graphic user Interface*) yang bersifat *user friendly*.
2. Dalam sistem otomatisasi bel listrik, menggunakan *timer* untuk menyalakan *device* perangkat bel Penghubung antara PC dan perangkat listrik, menggunakan *driver* saklar elektronik (menggunakan *optocoupler* dan *triac*), yang dapat bekerja dengan baik apabila mendapat masukan +5 volt DC dari komputer.
3. Sistem otomatisasi bel listrik ini dirancang menggunakan *database*. *Database* ini digunakan untuk menyimpan setting hari, jam, serta jumlah isyarat bel yang diinginkan.

5.2. Saran

Berdasarkan permasalahan yang ada pada sistem otomatisasi bel listrik yang telah dirancang, terdapat beberapa saran yaitu :

1. Aplikasi ini dapat dikembangkan lagi untuk otomatisasi sampai dengan 12 *device* (perangkat listrik), 8 dari *port data* dan 4 dari *port control* dengan memanfaatkan kemampuan dari *port* paralel.
2. Kembangkan lagi menjadi program yang lebih menarik sehingga kegunaannya benar-benar dapat dimanfaatkan.

DAFTAR PUSTAKA

- Anonim. 2002. Lembar Data (*Datasheet*) Optocoupler MOC3020.
- Anonim. 2002. Lembar Data (*Datasheet*) Triac BT.
- Anwar, M. K. 2003. *Aplikasi Sistem Informasi Manajemen bagi Pemerintah di Era Otonomi Daerah*. Malang.
- Aref, Darmawan. 2001. *Elektronika Dasar*. Yogyakarta : Andi Offset.
- Budhi, Romy Widodo dan Joshep Dedy Irawan. 2007. *Interfacing Paralel & Serial Menggunakan Delphi*. Yogyakarta: Graha Ilmu.
- Fatansyah. 1999. *Basis Data*. Bandung : CV. Informatika
- Jogiyanto, H.M. 2001. *Analisa dan Desain Sistem Informasi, edisi kedua*. Yogyakarta : Andi Offset.
- Kadir, Abdul. 2004. *Pemrograman Database dengan Delphi 7 Menggunakan Access dan Ado*. Yogyakarta : Andi Offset.
- Malvino, Inge. 1994. *Prinsip Prinsip elektronik*. Jakarta : Erlangga.
- Martina. 2001. 36 Jam Belajar Delphi 5. Jakarta: PT.Elex Media Komputido.
- Mikrodata. 2001. *Computer and Programming*, Edisi Maret, Vol. 3 seri 6. Jakarta : PT.Elex Media Komputido.
- Putra, Afgianto. 2002. *Teknik Antarmuka Komputer: Konsep dan Aplikasi*. Yogyakarta: Graha Ilmu.
- Sutadi, D. 2002. *I / O Bus & Motherboard*. Yogyakarta : Andi Offset.
- <http://electroniclab.com/komponen-elektronik.pdf>
- <http://elektronika-elektronika.blogspot.com>.

LAMPIRAN

LAMPIRAN A

Listing Program Jadwal Bel Otomatis Terprogram

```
unit U_utama;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, SUIImagePanel, ExtCtrls, Menus, WinSkinData, ComCtrls,  
DB, ADODB, Grids, DBGrids, SUIDBCtrls, SUIButton, jpeg;
```

```
type
```

```
TMain = class(TForm)  
    MainMenu1: TMainMenu;  
    Menu1: TMenuItem;  
    Setting1: TMenuItem;  
    Manual1: TMenuItem;  
    About1: TMenuItem;  
    Keluar1: TMenuItem;  
    PMLaku: TPanel;  
    PJam: TPanel;  
    suiPanel1: TsuiPanel;  
    Label26: TLabel;  
    Tulis: TLabel;  
    Tjam_Hari: TTimer;  
    TJudul: TTimer;  
    Label3: TLabel;  
    Label5: TLabel;  
    Label6: TLabel;  
    Label7: TLabel;  
    Label8: TLabel;  
    SBar: TStatusBar;  
    Panel3: TPanel;  
    PHari: TPanel;  
    SkinData1: TSkinData;  
    AdTet: TADOQuery;  
    DataSource1: TDataSource;  
    suiDBGrid1: TsuiDBGrid;  
    TBelHi: TTimer;  
    TBelMa: TTimer;  
    TNgebel: TTimer;  
    QBunyiBel: TADOQuery;  
    Panel1: TPanel;  
    Taktif: TTimer;  
    BtNAktif: TsuiButton;  
    Image1: TImage;  
    procedure Tjam_HariTimer(Sender: TObject);  
    procedure TJudulTimer(Sender: TObject);  
    procedure Setting1Click(Sender: TObject);
```

```

    procedure FormShow(Sender: TObject);
    procedure Manual1Click(Sender: TObject);
    procedure TBelMaTimer(Sender: TObject);
    procedure TNgebelTimer(Sender: TObject);
    procedure TaktifTimer(Sender: TObject);
    procedure BtNAktifClick(Sender: TObject);
    procedure Keluar1Click(Sender: TObject);
    procedure About1Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Main: TMain;
    kiri:boolean;
    a:string;
    nyala:byte;

implementation
uses U_Setting, U_Manual, U_Help, U_About;

{$R *.dfm}

Procedure Out32(PortAddress:smallint;Value:smallint);stdcall;external'inpout32.dll';
function Inp32(PortAddress:smallint):smallint;stdcall;external 'inpout32.dll';

procedure TMain.Tjam_HariTimer(Sender: TObject);
begin
    PJam.Caption:=TimeToStr(now);
    PHari.Caption:=FormatDateTime('dddd',Now);
end;

procedure TMain.Setting1Click(Sender: TObject);
begin
    FSetting.ShowModal;
end;

procedure TMain.FormShow(Sender: TObject);
Var WinVer:TOSVersionInfo;
    SWinver:string;
    num1,num2,Bld:integer;
begin
    TNgebel.Enabled:=true;
    Out32($378,$00);
end;

procedure TMain.TNgebelTimer(Sender: TObject);
begin

```

```

with QBunyiBel do begin
  Active:=false;
  SQL.Clear;
  SQL.Text:='SELECT * FROM Tteet where hari=:hari AND jam=:jam';
  Parameters.ParamByName('hari').Value:=PHari.Caption;
  Parameters.ParamByName('jam').Value:=PJam.Caption;
  Active:=true;
end;
if QBunyiBel.RecordCount<>0 then
begin
if QBunyiBel.Fields[3].Value='1' then
begin
  Nyala:=nyala+1;
  out32($378,nyala);
  sleep(750);
  TBelMa.Enabled:=true;
end
else
if QBunyiBel.Fields[3].Value='2' then
begin
  Nyala:=nyala+1;
  out32($378,nyala);
  sleep(750);
  Nyala:=nyala-1;
  out32($378,nyala);
  sleep(750);
  Nyala:=nyala+1;
  out32($378,nyala);
  sleep(1000);
  TBelMa.Enabled:=true;
end
else
if QBunyiBel.Fields[3].Value='3' then
begin
  Nyala:=nyala+1;
  out32($378,nyala);
  sleep(750);
  Nyala:=nyala-1;
  out32($378,nyala);
  sleep(750);
  Nyala:=nyala+1;
  out32($378,nyala);
  sleep(750);
  Nyala:=nyala-1;
  out32($378,nyala);
  sleep(750);
  Nyala:=nyala+1;
  out32($378,nyala);
  sleep(1000);
  TBelMa.Enabled:=true;

```

```
end;  
end;  
end;
```

```
procedure TMain.TBelMaTimer(Sender: TObject);  
begin  
  Nyala:=nyala-1;  
  out32 ($378,nyala);  
  TBelMa.Enabled:=false;  
  TBelHi.Enabled:=true;  
end;
```

```
procedure TMain.TaktifTimer(Sender: TObject);  
begin  
  AdTet.Active:=True;  
  with AdTet do  
    begin  
      AdTet.Close;  
      SQL.Clear;  
      SQL.Add('select * from TTeet where Hari=:hari' );  
      Parameters.ParamByName('hari').Value:=PHari.Caption;  
      open;  
    end;  
end;
```

```
procedure TMain.BtNAktifClick(Sender: TObject);  
begin  
  IF BtNAktif.Caption='AKTIFKAN' then  
    begin  
      Taktif.Enabled:=true;  
      BtNAktif.Caption:='NON AKTIF';  
      BtNAktif.Glyph.LoadFromFile(extractfilepath(application.ExeName)+'\ma.bmp');  
    end  
  else  
    if btNAktif.Caption='NON AKTIF' then  
      begin  
        Taktif.Enabled:=false;  
        BtNAktif.Caption:='AKTIFKAN';  
        BtNAktif.Glyph.LoadFromFile(extractfilepath(application.ExeName)+'\hi.bmp');  
        AdTet.Active:=false;  
      end;  
    end;
```

```
procedure TMain.Keluar1Click(Sender: TObject);  
begin  
  Application.Terminate;  
end;  
end.
```

unit U_Setting;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, ComCtrls, SUIComboBox, ExtCtrls, SUIButton, DB, ADODB,
Grids, DBGrids, SUIDBCtrls, WinSkinData;

type

TFSetting = class(TForm)
 suiHari: TsuiComboBox;
 BtSimpan: TsuiButton;
 DataSource1: TDataSource;
 panel: TPanel;
 suiDBGrid1: TsuiDBGrid;
 cbkelas: TComboBox;
 Edit1: TEdit;
 JMasuk: TDateTimePicker;
 ComboBox2: TComboBox;
 ADTab: TADOQuery;
 Tampil: TsuiButton;
 Label1: TLabel;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 Label7: TLabel;
 Label8: TLabel;
 BtUpdate: TsuiButton;
 BtHapus: TsuiButton;
 Image1: TImage;
 procedure BtSimpanClick(Sender: TObject);
 procedure suiHariChange(Sender: TObject);
 procedure TampilClick(Sender: TObject);
 procedure FormShow(Sender: TObject);
 procedure BtUpdateClick(Sender: TObject);
 procedure suiDBGrid1DblClick(Sender: TObject);
 procedure BtHapusClick(Sender: TObject);

private

 { Private declarations }

public

 { Public declarations }

var

 FSetting: TFSetting;
 stredit: String;

implementation

uses U_utama, DateUtils;

{ \$R *.dfm }

procedure TFSetting.BtSimpanClick(Sender: TObject);

begin

if cbkelas.Text = " **then**

Begin

 MessageDlg('Hari belum dipilih!',mtWarning,[mbok],0);

 cbkelas.SetFocus;

 exit;

end

else

if Edit1.Text = " **then**

Begin

 MessageDlg('Keterangan Jadwal belum diisi!',mtWarning,[mbok],0);

 Edit1.SetFocus;

 exit;

end

else

if ComboBox2.Text = " **then**

Begin

 MessageDlg('Pilihan bel belum diset!',mtWarning,[mbok],0);

 ComboBox2.SetFocus;

 exit;

end

else

ShortTimeFormat:='hh:mm:ss';

with ADTab **do**

begin

 insert;

 FieldByName('Hari').AsString:=cbkelas.Text;

 FieldByName('Nama_Jadwal').AsString:=edit1.Text;

 FieldByName('Jam').AsString:=TimeToStr(JMasuk.Time);

 FieldByName('Jumlah_Bel').AsString:=ComboBox2.Text;

 post;

 ShowMessage('Data Telah Tersimpan');

 cbkelas.Text:=";

 edit1.Clear;

 combobox2.Text:=";

end;

end;

procedure TFSetting.suiHariChange(Sender: TObject);

begin

BtSimpan.Enabled:=true;

BtUpdate.Enabled:=false;

BtHapus.Enabled:=true;

if suiHari.Text='Senin' **then**

with AdTab **do**


```

begin
    AdTab.Close;
    SQL.Clear;
    SQL.Add('select * from TTeet where Hari=:hari' );
    Parameters.ParamByName('hari').Value:=suiHari.Text;
    Open;
end;
    if suiHari.Text='Selasa' then
with ADTab do
begin
    ADTab.Close;
    SQL.Clear;
    SQL.Add('select * from TTeet where Hari=:hari' );
    Parameters.ParamByName('hari').Value:=suiHari.Text;
    open;
end;
    if suiHari.Text='Rabu' then
with ADTab do
begin
    ADTab.Close;
    SQL.Clear;
    SQL.Add('select * from TTeet where Hari=:hari' );
    Parameters.ParamByName('hari').Value:=suiHari.Text;
    open;
end;
    if suiHari.Text='Kamis' then
with ADTab do
begin
    ADTab.Close;
    SQL.Clear;
    SQL.Add('select * from TTeet where Hari=:hari' );
    Parameters.ParamByName('hari').Value:=suiHari.Text;
    open;
end;
    if suiHari.Text='Jumat' then
with ADTab do
begin
    ADTab.Close;
    SQL.Clear;
    SQL.Add('select * from TTeet where Hari=:hari' );
    Parameters.ParamByName('hari').Value:=suiHari.Text;
    open;
end;
    if suiHari.Text='Sabtu' then
with ADTab do
begin
    ADTab.Close;
    SQL.Clear;
    SQL.Add('select * from TTeet where Hari=:hari' );
    Parameters.ParamByName('hari').Value:=suiHari.Text;

```

```

        open;
    end;
    if suiHari.Text='Minggu' then
        with ADTab do
            begin
                ADTab.Close;
                SQL.Clear;
                SQL.Add('select * from TTeet where Hari=:hari ');
                Parameters.ParamByName('hari').Value:=suiHari.Text;
                open;
            end;
        end;
    end;

    procedure TFSetting.TampilClick(Sender: TObject);
    begin
        with ADTab do
            begin
                Active:=False;
                Sql.Clear;
                Sql.Add('select * from TTeet');
                Active:=True;
            end;
            suiHari.Text:="";
        end;

    procedure TFSetting.FormShow(Sender: TObject);
    begin
        BtUpdate.Enabled:=false;
        ADTab.Active:=True;
        with ADTab do
            begin
                ADTab.Close;
                SQL.Clear;
                SQL.Add('select * from TTeet where Hari=:hari ');
                Parameters.ParamByName('hari').Value:=suiHari.Text;
                open;
            end;
        end;

    procedure TFSetting.BtUpdateClick(Sender: TObject);
    begin
        with ADTab do
            begin
                Locate('Hari',stredit,[]);
                edit;
                FieldByName('Hari').AsString:=cbkelas.Text;
                FieldByName('Nama_Jadwal').AsString:=edit1.Text;
                FieldByName('Jam').AsString:=TimeToStr(JMasuk.Time);
                FieldByName('Jumlah_Bel').AsString:=ComboBox2.Text;
                post;
            end;
        end;
    end;

```

end;
end;

procedure TFSetting.suiDBGrid1DbClick(Sender: TObject);
begin
 cbkelas.Text:=suiDBGrid1.Fields[0].AsString;
 Edit1.Text:=suiDBGrid1.Fields[1].AsString;
 ComboBox2.Text:=suiDBGrid1.Fields[3].AsString;
 JMasuk.Time:=suiDBGrid1.Fields[2].AsDateTime;
 stredit:=suiDBGrid1.SelectedField.KeyFields;
 BtUpdate.Enabled:=true;
 BtSimpan.Enabled:=false;
 BtHapus.Enabled:=false;
end;

procedure TFSetting.BtHapusClick(Sender: TObject);
begin
 If MessageDlg('Anda Yakin Akan menghapus data
 tsb??!!',mtconfirmation,[mbYes,MbNo],0)=MrYes **then**
 ADTab.Delete;
 end;
end.

SKEMA RANGKAIAN (Bel Otomatis Terprogram Berbasis PC)

