

# APLIKASI PPI 8255 DENGAN BAHASA PEMROGRAMAN BORLAND DELPHI UNTUK PENGATURAN JAM DAN PENERANGAN RUANG KULIAH

Chusni Mubarak L2F 399 377  
Jurusan Teknik Elektro Fakultas Teknik  
Universitas Diponegoro Semarang

**ABSTRAK** - Berangkat dari adanya kemudahan yang diperoleh manusia dengan adanya komputer, maka pada tugas akhir ini akan didayagunakan komputer pribadi (PC) sehingga dapat digunakan untuk mengatur jam / schedule jam kuliah dan menyalakan lampu.

Untuk keperluan pengaturan jam kuliah ini, diperlukan peralatan elektronik di luar komputer yang dapat mengoptimalkan fungsi dari PC sebagai pengatur jadwal / jam dan penerang ruang kuliah, yaitu rangkaian clock decoder yang merupakan acuan jam kuliah ini, rangkaian sensor LDR untuk mendeteksi cahaya lingkungan guna menyalakan lampu apabila kondisi ruangan gelap, rangkaian pengendali lampu, rangkaian pengendali amplifier dan amplifier sebagai penguat suara untuk memberitahukan pergantian jam kuliah.

Sebagai interface komputer dengan peralatan elektroniknya digunakan antarmuka (card) Programmable Peripheral Interface (PPI) 8225 dan bahasa pemrograman yang digunakan untuk mengendalikan pengatur jam kuliah dan menyalakan lampu ini adalah Borland Delphi.

## I. PENDAHULUAN

Kemajuan ilmu pengetahuan dan teknologi telah mendorong manusia untuk berusaha mengatasi segala permasalahan yang timbul di sekitarnya. Keterbatasannya untuk terjun langsung melakukan proses pengendalian dan pengaturan, baik karena faktor kelelahan ataupun faktor kejenuhan, lambat laun telah dapat diatasi dengan ditemukannya peralatan-peralatan kendali yang bekerja secara otomatis mekanis dan atau elektrik, salah satu contohnya adalah komputer. Alat ini dapat digunakan untuk meringankan dan memudahkan pekerjaan manusia dalam berbagai hal, diantaranya sebagai pengontrol mesin-mesin industri, sebagai media komunikasi dan lain-lain.

Pada tugas akhir ini komputer digunakan untuk mengatur jam kuliah dan menyalakan lampu yang mana untuk keperluan ini dibutuhkan peripheral sebagai penghubung antara komputer dengan perangkat elektronik bagian dari sistem pengaturan jam dan penerangan ruang kuliah. Peripheral tersebut adalah PPI 8255, sebagai pengolah data adalah bahasa pemrograman Borland Delphi.

## II. PERANGKAT KERAS

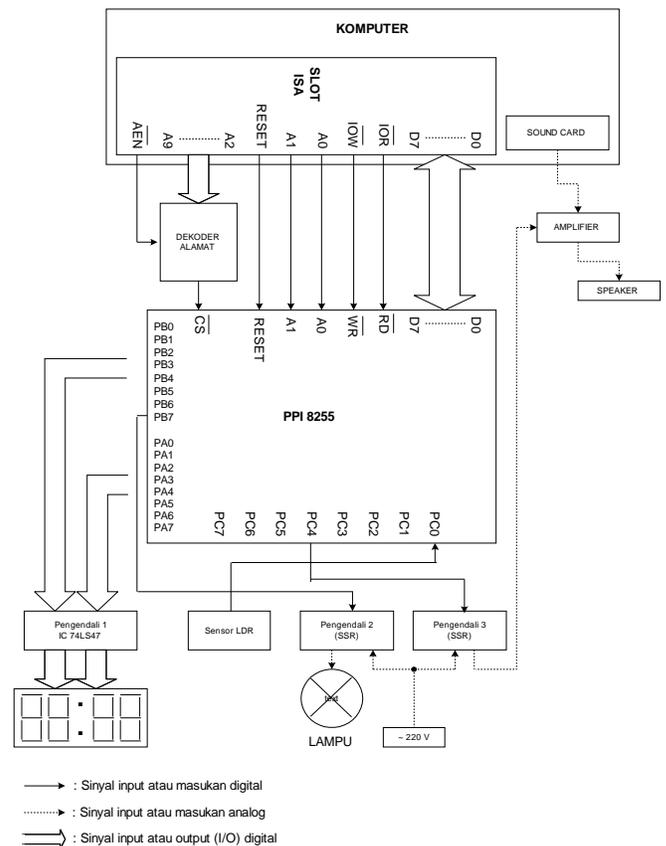
### A. Blok Diagram Keseluruhan Sistem

Blok diagram secara keseluruhan dari rancangan sistem pengaturan jam dan penerangan ruang kuliah diperlihatkan pada gambar 2.1.

Komputer berhubungan dengan PPI 8255 melalui slot ISA. Data-data yang masuk ke komputer dari perangkat elektronik di luar komputer, yaitu masukan dari sensor cahaya (LDR) akan diolah kemudian digunakan untuk menggerakkan rangkaian pengendali (driver) lampu sehingga dapat diaktifkan sesuai dengan yang diinginkan setelah sebelumnya

melalui PPI 8255, sedangkan data yang merupakan masukan program akan diolah juga oleh komputer dan kemudian digunakan untuk mengaktifkan / menggerakkan perangkat elektronik di luar komputer, seperti rangkaian clock decoder untuk menampilkan tampilan jam, rangkaian pengendali 2 (SSR / solid state relay) yang merupakan rangkaian pengendali amplifier. Pengelola data menggunakan bahasa pemrograman Borland Delphi.

Melalui bahasa pemrograman Delphi ini dibuat suatu program yang dapat mengambil file-file yang ada di hard disk (file wav), mengaktifkan clock display dengan tampilan seven-segment, mengendalikan amplifier serta dapat mengendalikan lampu.



Gambar 2.1 Diagram blok sistem pengaturan jam kuliah

### B. Rancangan Perangkat Keras Sistem

Perangkat keras pada sistem pengaturan jam dan penerangan ruang kuliah dibagi menjadi beberapa bagian sebagai berikut :

1. Rangkaian PPI 8225 (card + IC 8255)
2. Rangkaian pengendali 1 / clock decoder (IC 74LS47)
3. Rangkaian pengendali 2 dan pengendali 3 (SSR/Solid State Relay)

4. Rangkaian sensor cahaya (LDR)

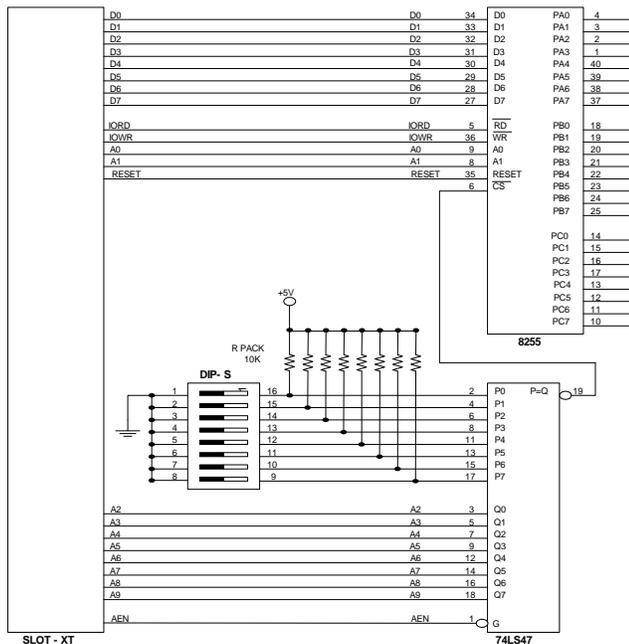
1) Rangkaian PPI 8255 (Card + IC 8255)

Rangkaian *interface* ini berfungsi sebagai penghubung antara komputer dengan rangkaian luar. PPI 8255 memiliki tiga buah *port*, yaitu *port A*, *port B* dan *port C*. Masing-masing *port* terdiri dari delapan penyemat (bit), sehingga ada 24 penyemat yang dapat difungsikan sebagai masukan atau keluaran. Pada PPI 8255 terdapat rangkaian dekoder alamat, dekoder ini digunakan untuk mengaktifkan *chip select* (CS). Komponen yang digunakan adalah IC 74LS688, resistor dan *DIP Switch*. IC 74LS688 akan aktif apabila masukan  $P_n = Q_n$ . Karena alamat yang digunakan adalah 300<sub>H</sub>, maka alamat dari A9 – A2 sama dengan 11000000, dekoder alamat akan mengaktifkan *chip select*. Sinyal kendali yang diambil dari komputer, yaitu IOR, IOW dan *reset*. Sinyal IOR menyatakan bahwa CPU sedang membaca data dari IC 8255. Sinyal IOW menyatakan bahwa CPU sedang menulis data ke IC 8255. sinyal IOR dan IOW adalah aktif rendah. Sinyal *reset* digunakan untuk mereset IC 8255, sinyal ini merupakan *power on reset* dan sinyal aktif tinggi.

Pengaturan *port-port* yang digunakan adalah sebagai berikut :

- *Port A* sebagai *output* :
  - PA<sub>0</sub> – PA<sub>3</sub> untuk *seven-segment* II (menit II)
  - PA<sub>4</sub> – PA<sub>6</sub> untuk *seven-segment* I (menit I)
- *Port B* sebagai *output* :
  - PB<sub>0</sub> – PB<sub>3</sub> untuk *seven-segment* II (jam II)
  - PB<sub>4</sub> – PB<sub>5</sub> untuk *seven-segment* I (jam I)
  - PB<sub>6</sub> untuk LED (kedip detik)
  - PB<sub>7</sub> untuk rangkaian pengendali lampu
- *Port C* atas sebagai *output* dan *port C* rendah sebagai *input* :
  - PC<sub>0</sub> menerima *input* dari rangkaian sensor LDR
  - PC<sub>4</sub> untuk rangkaian pengendali amplifier

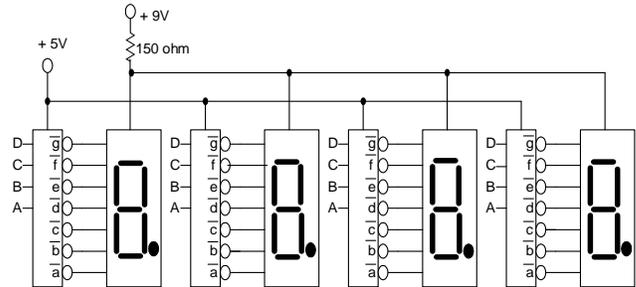
Dengan demikian *control word* yang digunakan untuk proses inisialisasi adalah 81<sub>H</sub>. Skema rangkaian PPI 8255 dapat dilihat pada gambar 2.2.



Gambar 2.2 Skema rangkaian PPI 8255

2) Rangkaian Pengendali 1 / Clock Decoder (IC 74LS47)

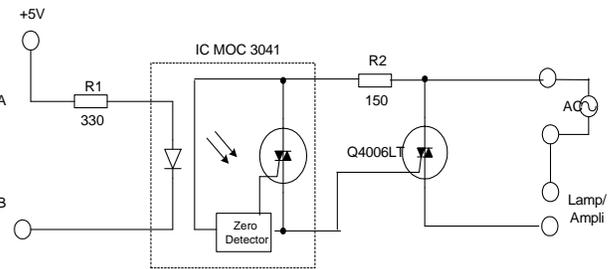
Rangkaian pengendali ini digunakan untuk mengaktifkan *display clock* dari *timer* yang ada di komputer, yaitu berupa 4 buah 7-segmen. Kerja dari rangkaian *clock decoder* ini menggunakan IC 74LS47 untuk mengaktifkan 7-segmen. Skema rangkaian IC 74LS47 dengan tampilan *clock* utamanya (4 buah 7-segmen) ditunjukkan pada gambar 2.3.



Gambar 2.3 Skema rangkaian IC 74LS47 dengan tampilan clock utama

3) Rangkaian Pengendali 2 dan Pengendali 3 (SSR/Solid State Relay)

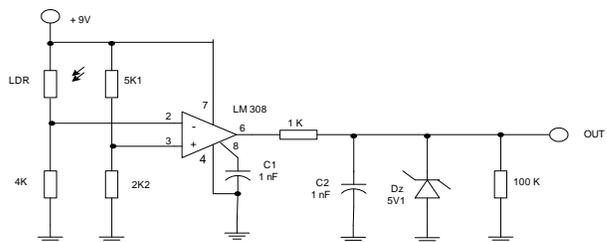
Rangkaian pengendali 2 digunakan untuk mengendalikan lampu, sedangkan rangkaian pengendali 3 untuk mengendalikan amplifier. Rangkaian pengendali ini menggunakan IC MOC 3041 dan triac. IC MOC 3041 berfungsi sebagai *driver*, prinsip kerjanya dengan memanfaatkan suatu masukan berarus kecil untuk menghidupkan LED yang berada dalam kemasan IC tersebut dan juga akan menyulut triac yang juga berada dalam kemasan IC tersebut. Sedangkan triac berfungsi seperti saklar elektronik yang dapat melewatkan arus bolak-balik dan juga dapat melindungi lampu dari lonjakan tegangan transient tinggi pada saat pertama kali lampu atau amplifier dihidupkan. Skema rangkaian pengendali 2 dan 3 ditunjukkan pada gambar 2.4.



Gambar 2.4 Skema rangkaian pengendali 2 dan pengendali 3 (SSR)

4) Rangkaian Sensor Cahaya

Sensor cahaya / *fotoreistor* digunakan untuk mengontrol *output* dari op-amp, seperti pada gambar 2.5.



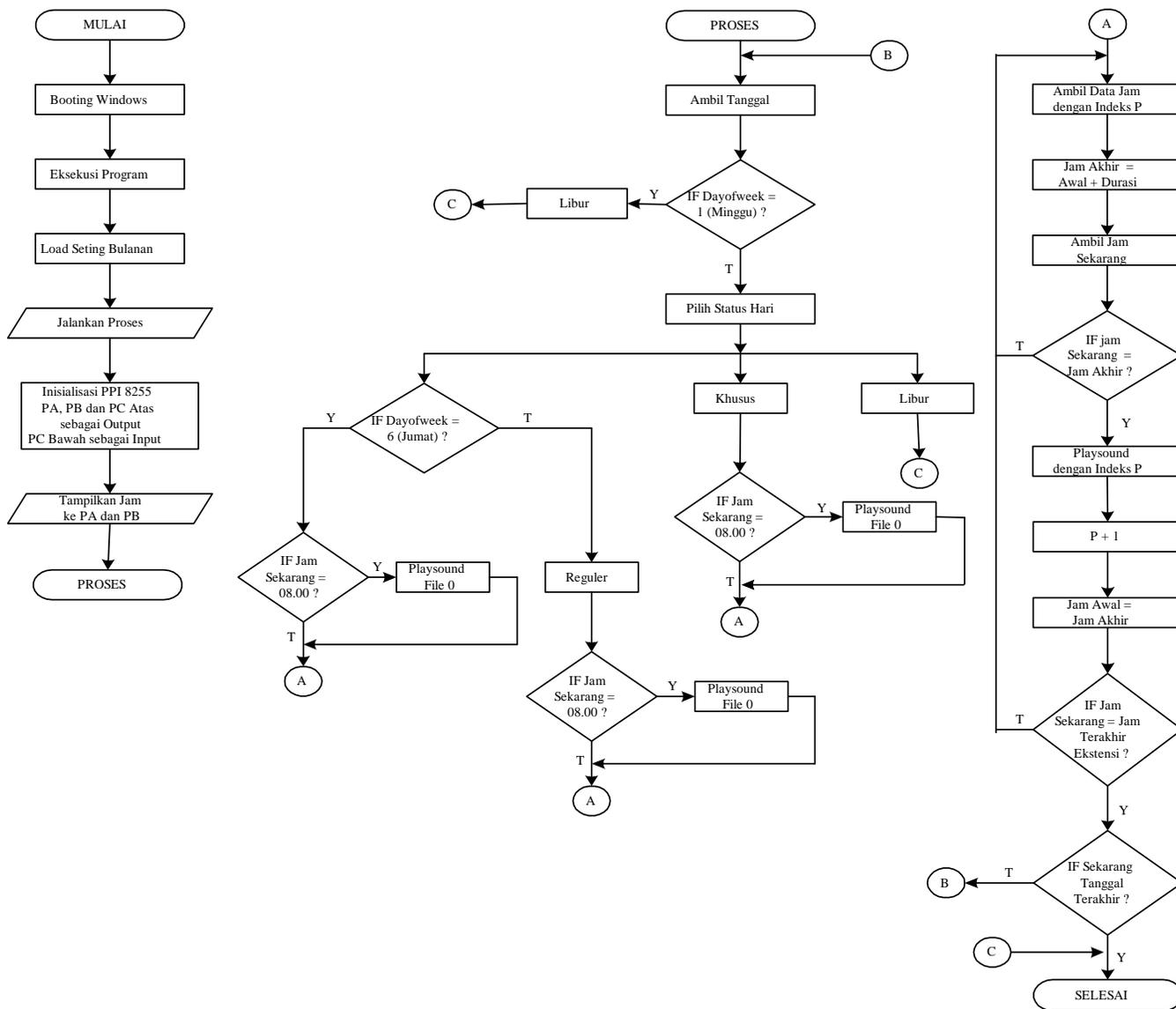
Gambar 2.5 Skema rangkaian sensor cahaya (LDR)

Pada gambar 2.5, sensor cahaya / *fotoreistor* difungsikan sebagai saklar cahaya. Fotoreistor mempunyai hambatan yang berubah terhadap penyinaran yang sampai pada permukaan *fotoreistor* tersebut, sehingga *fotoreistor* berfungsi sebagai tahanan pada rangkaian pembagi tegangan. Pada rangkaian sensor cahaya terdapat 2 pembagi tegangan, yaitu pembagi tegangan pada resistor 4K ohm dan pada resistor 2K2 ohm, yang digunakan sebagai masukan tegangan referensi pada masukan op-amp (sebagai pembanding). Pada masukan tegangan referensi (4K ohm) berfungsi sebagai acuan bagi besarnya tegangan pembanding, sehingga keluaran output op-amp akan aktif apabila terdapat selisih antara kedua masukan op-amp tersebut.

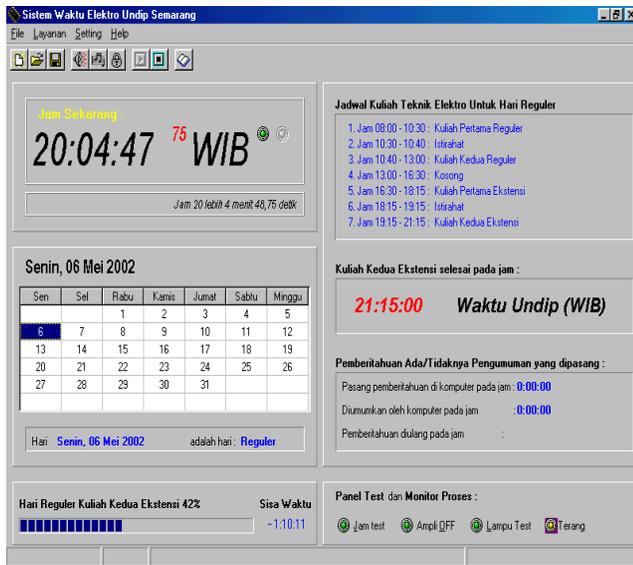
### III. PERANGKAT LUNAK

Perangkat lunak (*software*) dipakai untuk mengontrol sistem dan penampil kendali. Pembuatan perangkat lunak menggunakan bahasa pemrograman Delphi versi 3.0, penggunaan bahasa ini dimaksudkan untuk mempermudah dalam pengaturan perangkat keras (*hardware*) dan tampilan. Perancangan pembuatan perangkat lunak diawali dengan penyusunan diagram alir pada gambar 3.1. Diagram alir menunjukkan sistem yang dipakai dalam perancangan perangkat lunak, yaitu mencakup beberapa tampilan dan aplikasi.

Dalam program ini, nama form utama adalah TUtama dan tampilan menu utamanya terlihat pada gambar 3.2.



Gambar 3.1 Diagram alir program utama



Gambar 3.2 Tampilan menu utama

Tampilan menu utama terdiri 4 menu, yaitu menu *File*, menu *Layanan*, menu *Setting* dan menu *Help*.

Menu *File* terdiri dari :

- Menu Baru
- Menu Buka
- Menu Simpan
- Menu *Exit*

Menu *Layanan* terdiri dari 2 menu, yaitu:

- Menu Pengumuman
- Menu Musik

Menu *Setting* terdiri dari 1 menu, yaitu menu *Aktifitas Sehari*.

Dan untuk menu *Help* terdiri dari 3 menu, yaitu :

- Menu Spesifikasi Alat
- Menu Bantuan
- Menu *About*

Ada beberapa unit yang digunakan untuk pembuatan program ini, yaitu unit 1 sampai unit 8, yang mana dengan perintah *uses* maka unit-unit ini akan dihubungkan ke unit 1 (unit utama). Setelah pendeklarasian kontanta dan variabel-variabel tertentu maka program ini akan dioperasikan dalam beberapa prosedur / perintah pemrograman yang ada.

#### IV. PENGUJIAN PERALATAN DAN PROGRAM

Terdapat beberapa pengujian yang dilakukan terhadap sistem pengaturan jam kuliah ini, diantaranya adalah:

##### A. Pengujian *Programmable Peripheral Interface (PPI) 8255*

Pengujian terhadap alat ini dilakukan untuk mengetahui apakah alat ini dapat beroperasi dengan mantap dan benar, yaitu dapat melewati dan menyampaikan data ke perangkat luar komputer maupun ke komputer itu sendiri. Pengujian ini dilakukan dengan menggunakan rutin program dan mengukur logika keluaran dari *port A*, *port B* dan *Port C* dari PPI 8255. Rutin program uji dari PPI 8255 adalah sebagai berikut :

```
unit UjiPPI;
```

```
interface
```

```
uses
  Windows, Messages, SysUtils, Classes,
  Graphics, Controls, Forms, Dialogs, StdCtrls;
```

```
type
```

```
  TForm1 = class(TForm)
```

```
    Label1: TLabel;
```

```
    Button1: TButton;
```

```
    Button2: TButton;
```

```
    aa: TLabel;
```

```
  procedure
```

```
    kirim
```

```
(alamat:word;data:byte);
```

```
  procedure terima (alamat:word);
```

```
  procedure Button1Click(Sender:
```

```
TObject);
```

```
  procedure Button2Click(Sender:
```

```
TObject);
```

```
  private
```

```
    { Private declarations }
```

```
  public
```

```
    { Public declarations }
```

```
end;
```

```
var
```

```
  Form1: TForm1;
```

```
  a,c : integer;
```

```
implementation
```

```
{ $R *.DFM }
```

```
procedure
```

```
TForm1.kirim(alamat:word;data:byte);
```

```
begin
```

```
asm
```

```
mov dx,alamat
```

```
mov al,data
```

```
out dx,al
```

```
end;
```

```
end;
```

```
procedure TForm1.terima(alamat:word);
```

```
begin
```

```
asm
```

```
mov dx,alamat
```

```
in al,dx
```

```
mov c,al
```

```
end;
```

```
end;
```

```
procedure TForm1.Button1Click(Sender:
```

```
TObject);
```

```
begin
```

```
kirim($303,$89);
```

```
kirim($300,$00);
```

```
kirim($301,$00);
```

```
terima($302);
```

```
c := a;
```

```
aa.caption := inttostr(c);
```

```
If c = $00 then
```

```
kirim ($301,$03);
```

```
end;
```

```
procedure
```

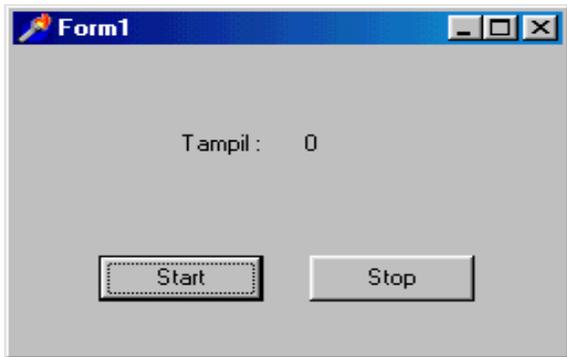
```
TForm1.Button2Click(Sender:
```

```
TObject);
```

```
begin
Close;
end;
```

end.

Sebelum program dieksekusi, *port C* harus dalam keadaan 00<sub>H</sub> (tanpa ada masukan). Setelah program dieksekusi pada tampilan (aa label) akan tertulis 0 dan data pada *port B* sama dengan 03<sub>H</sub>, tampilan dari program uji PPI adalah :



Gambar 4.1 Tampilan pengujian PPI 8255

Pengukuran logika dilakukan untuk mengetahui kebenaran nilai logika dari rangkaian yang telah dibuat. Hal ini penting karena rangkaian yang dibuat digunakan untuk mengontrol rangkaian yang lain sehingga dibutuhkan nilai logika yang mantap.

**B. Pengujian Rangkaian Pengendali 1/Clock decoder**

Pengujian untuk rangkaian pengendali 1 dilakukan dengan mengukur tegangan masukan yang berasal dari *card* PPI 8255 dan tegangan *output* dari IC 74LS47.

- Untuk tegangan *input* (A,B,C dan D) kondisi
  - Logika 0 besarnya tegangan = 0,10 volt
  - Logika 1 besarnya tegangan = 3,88 volt
- Untuk tegangan *output* ( $\bar{a}$ ,  $\bar{b}$ ,  $\bar{c}$ ,  $\bar{d}$ ,  $\bar{e}$ ,  $\bar{f}$  dan  $\bar{g}$ ) kondisi
  - Logika 0 besarnya tegangan = 0,23 volt
  - Logika 1 besarnya tegangan = 3,01 volt
- *Output* dari IC 74LS47 adalah aktif *low*

Pada pengujian rangkaian pengendali 1 di atas hasilnya sesuai dengan data yang ada di Data Sheet Book dari IC 74LS47<sup>[20]</sup> sehingga hasil penunjukan dari *seven* segmen sesuai dengan yang diharapkan. Pengujian selanjutnya adalah dengan menggunakan *software* penguji yang langsung ditampilkan pada unit 1 (utama), yaitu:

```
procedure TUtama.jamtesTimer(Sender:
TObject);
begin
hasil:=(sat div 10)*16+(sat mod 10);
kirim($300,hasil);
inc(sat);
if sat < 25 then kirim($301,hasil);
if sat > 59 then
begin
jamtes.Enabled:=false;
kediip.Enabled:= true;
sat:=0;
```

```
end;
end;
```

**C. Pengujian Rangkaian Pengendali 2 dan 3 (SSR/Solid State Relay)**

Pengujian terhadap rangkaian ini dilakukan dengan memberi catu daya sebesar +5 V atau memberi masukan logika tinggi pada kaki 1 dari IC MOC 3041, menghubungkan rangkaian ini dengan tegangan jala-jala PLN sebesar 220 V dan menghubungkan kaki 2 dari IC MOC 3041 dengan ground. Logika tinggi besarnya sekitar 3,8 – 5 V dan logika rendah besarnya sekitar 0 – 0,5 V. Tujuan pengujian ini untuk mendeteksi tegangan *output* AC (beban AC) yang dihasilkan untuk mencatu lampu atau amplifier sesuai dengan harapan, sehingga mampu menghidupkan lampu atau amplifier dengan baik. Hasil yang didapat dari pengujian ini adalah :

Tabel 4.1 Hasil pengujian rangkaian pengendali 2 dan 3 (SSR)

Kondisi Input	Kondisi Output	Kondisi Lampu/Amplifier
1	210 V AC	On
0	0 V AC	Off

Hasil pengujian di atas menunjukkan bahwa apabila rangkaian pengendali 2 dan 3 mendapat *input* logika tinggi (1) atau mendapat catu sebesar 3,8 – 5 V DC, maka *output* yang dihasilkan dari rangkaian adalah 210 V AC, sehingga dengan *output* sebesar ini dapat mengaktifkan lampu dan amplifier dengan baik. Sebaliknya apabila rangkaian 2 dan 3 mendapat *input* logika rendah (0), maka *output* yang dihasilkan dari rangkaian pengendali 2 dan 3 adalah 0 V AC, karena triac tidak terhubung antara MT1 dan MT2-nya sehingga seperti saklar yang tidak terhubung. Pada keadaan ini amplifier atau lampu akan mati.

**D. Pengujian Rangkaian Sensor Cahaya/LDR**

Pengujian ini bertujuan mendeteksi perubahan cahaya yang mengenai permukaan LDR agar didapatkan *output* sesuai yang dikehendaki sehingga dapat digunakan sebagai masukan ke komputer untuk mengaktifkan lampu. Setelah semua komponen pada rangkaian sensor cahaya terpasang semua termasuk supply daya sebesar + 9V, langkah selanjutnya untuk mendeteksi rangkaian dengan cara menutup permukaan LDR dan mengatur variabel resistor supaya tegangan pada variabel resistor yang masuk ke kaki (-) dari op-amp lebih rendah dari tegangan yang masuk ke kaki (+) dari op-amp sehingga dengan fungsi op-amp sebagai pembanding didapatkan *output* rangkaian berlogika tinggi (4,8 – 5 V). Langkah selanjutnya dengan membuka penutup LDR terus seperti langkah diatas sampai didapatkan *output* dari rangkaian berlogika rendah (0 – 0,5 V). Hasil yang di dapatkan dari pengujian ini adalah sebagai berikut :

Tabel 4.2 Hasil pengujian rangkaian sensor cahaya (LDR)

Kondisi Input LDR	Tegangan Output Op-Amp	Tegangan Output Rangkaian	Logika Output
Menerima cahaya	0,5 V DC	0,5 V DC	'0'
Tidak menerima cahaya	7,30 V DC	4,96 V DC	'1'

Setelah diputar dan didapatkan nilai *variable resistor* yang tepat untuk hasil pengujian ini, yaitu sebesar 4K ohm maka dalam aplikasinya *variable resistor* diganti dengan resistor biasa yang besarnya 4K ohm.

#### E. Pengujian Perangkat Lunak (*Software*)

Perangkat lunak yang berupa bahasa pemrograman Delphi diuji tiap prosedurnya, sehingga tiap prosedur dapat dipastikan berjalan sesuai dengan yang diharapkan. Hal ini dilakukan karena pembuat program dilakukan secara bertahap, prosedur demi prosedur dan program ini bersifat fleksibel artinya mudah diperbaharui apabila dijumpai kesalahan. Kesalahan yang ada dapat diketahui apabila prosedur atau rutin program yang ada tidak dapat dirun dan dikompilasi karena muncul pesan *error* pada monitor komputer. Pada pengujian rutin program tertentu terkadang dilakukan dengan bantuan rutin program / prosedur lainnya.

Untuk prosedur-prosedur yang tidak berhubungan dengan I/O komputer (*hardware* diluar komputer), hasilnya dapat dilihat pada tampilan layar monitor. Misalnya prosedur untuk menampilkan jam dengan satuan terbesar sampai yang terkecil (jam, menit, detik, milidetik). Dengan perintah pemrograman yang ada, yaitu :  
`DecodeTime( (Now), Hour, Min, Sec, Msec );` akan didapatkan hasil sesuai dengan yang diinginkan. Listing program untuk penampilan jam ini dapat dilihat pada lampiran A.

Untuk prosedur pada program yang berhubungan dengan piranti I/O diluar komputer, piranti I/O tersebut diuji perubahan kondisinya. Sebagai contoh proses untuk menampilkan penunjukan angka dari 0 sampai 23 yang mewakili banyaknya jam dalam sehari dan angka dari 0 sampai 59 yang mewakili banyaknya angka menit dalam satu jam pada *display seven-segment*. Pada contoh ini ditentukan alamat dari *port A* dan *port B* yang akan digunakan sebagai *output* guna memberi data masukan ke IC *dekoder 74LS47*, sehingga didapatkan *display seven-segment* yang diharapkan. Adapun cuplikan instruksinya adalah sebagai berikut:

```
hasil:=(sat div 10)*16+(sat mod 10);
kirim($300,hasil);
inc(sat);
if sat < 25 then kirim($301,hasil);
if sat > 59 then
```

Penunjukan atau perubahan angka pada *display seven-segment* dapat dilihat setelah program ini dieksekusi, yaitu untuk angka jam dari 0 sampai 23 dan untuk angka menit dari 0 sampai 59.

## V. PENUTUP

### A. Kesimpulan

Dalam pembuatan tugas akhir ini diperoleh beberapa kesimpulan sebagai berikut :

1. Pengaturan jam / waktu dan penerangan ruang kuliah dapat direalisasikan dengan menggunakan komputer.
2. Pengaturan jam kuliah yang dibuat dipergunakan untuk menampilkan waktu dari jam kuliah di teknik elektro Undip (tampilan *clock* utama) dikuti dengan suara sebagai penanda pergantian jam kuliah

3. PPI 8255 dapat digunakan sebagai *interface* yang menghubungkan komputer dengan peralatan diluar komputer.
4. Bahasa pemrograman Borland Delphi dapat digunakan sebagai media pemrograman komputer dan PPI 8255.

### B. Saran

Tugas akhir ini masih banyak kekurangannya, untuk kesempurnaan perlu ada pengembangan lebih lanjut, saran penulis untuk pengembangan adalah sebagai berikut :

1. Dengan menggunakan port-port PPI 8255 yang masih tersisa, peralatan ini dapat dikembangkan dengan menambah perangkat elektronik lain, misalnya sensor suhu (*input*) yang dapat mengontrol suhu lingkungan sehingga dapat mengaktifkan AC (*output*).
2. Pada bagian menu layanan dapat dikembangkan lagi sehingga dapat digunakan untuk menjalankan aplikasi yang lebih baik.

## DAFTAR PUSTAKA

- [1]. Albert Paul Malvino Ph.D, *Aproksimasi Rangkaian Semikonduktor*, Erlangga, Jakarta, 1996.
- [2]. Albert Paul Malvino Ph.D, *Prinsip-prinsip Elektronika*, Edisi Kedua, Erlangga, Jakarta, 1994.
- [3]. Antony Pranata, *Pemrograman Borland Delphi*, Edisi 2, Andi Yogyakarta, Yogyakarta, 1997.
- [4]. Busono, *Komputer dan Turbo Pascal Lebih Lanjut tentang Peningkatan Daya Guna Komputer*, PT. Elex Media Komputindo, Jakarta 1991.
- [5]. Charles A. Schuler, William L. McNamee, *Industrial Electronics and Robotics*, International Edition, McGraw-Hill, Inc. Singapore, 1986.
- [6]. Dani Okianto, *Panduan Belajar Borland Delphi 3.0*, PT. Elex Media Komputindo, Jakarta.
- [7]. Djoko Pramono, *Belajar Sendiri Pemrograman Delphi '95*, PT. Elex Media Komputindo, Jakarta, 1996.
- [8]. Djoko Pramono, *Mudah Menguasai Delphi 4*, Jilid I-II, PT. Elex Media Komputindo, Jakarta, 1999.
- [9]. Fredrick W. Hughes, *Panduan Op-Amp*, PT. Elex Media Komputindo, Jakarta, 1994.
- [10]. Jacob Millman Ph.D, Christor C. Halkias Ph.D, *Electronics and Devices*, McGraw-Hill.
- [11]. Jacob Millman Ph.D, Christor C. Halkias Ph.D, *Elektronika Terpadu (Integrated Electronics & System Analog & Digital)*, Erlangga, Jakarta, 1984.
- [12]. JPM Steeman. *Data Sheet Book 2*, PT. Elex Media Komputindo, Jakarta 1996.
- [13]. Link Wolfgang, *Pengukuran, Pengendalian dan Pengaturan dengan PC*, PT. Elex Media Komputindo, Jakarta, 1993.
- [14]. Gayakwad Ramakant A, *Op-Amps and Linear Integrated Circuits*, Fourth Edition, Prentice-Hall, Inc.
- [15]. P Hogenboom, *Data Sheet Book 3 : Catatan Aplikasi*, PT. Elex Media Komputindo, Jakarta, 1996.
- [16]. P Hogenboom, *Data Sheet Book 4 Peripheral Chips*, PT. Elex Media Komputindo, 1992, Jakarta, 1992.
- [17]. P Insap Santoso, *Dasar-dasar Pemrograman Pascal Teori dan Pemrograman Terapan*, Andi Yogyakarta, Yogyakarta, 1997.
- [18]. Rizal Rizkiawan, *Tutorial Perancangan Hardware III*, PT. Elex Media Komputindo, Jakarta, 1997.

- [19]. Roger L. Tokheim, *Elektronika Digital*, Edisi Kedua, Erlangga, Jakarta, 1995.
- [20]. Wasito S, *Data Sheet Book 1 Data Linier, TTL dan CMOS (Data Penting Elektronika)*, PT. Elex Media Komputindo, 1997.
- [21]. Wasito S, *Vademekum Elektronika*, Edisi Kedua, PT. Gramedia Pustaka Utama, Jakarta, 1995.
- [22]. William David Cooper, *Instrumentasi Elektronika dan Pengukuran*, Edisi Kedua, Erlangga, Jakarta, 1986.
- [23]. Wiliiam J. Mooney, *Optoelectronics Devices and Principles*, Prentice-Hall International, Inc.

#### Riwayat Hidup Penulis

Chusni Mubarak lahir tanggal 14 April 1976 di Pekalongan Jawa Tengah. Pada tahun 1998 lulus D III Teknik Elektro Universitas Diponegoro Semarang. Dan pada tahun 1999 melanjutkan studi di Jurusan Teknik Elektro (Ekstensi) Universitas Diponegoro Semarang.

Pembimbing I

Pembimbing II

Ir. Sudjadi, MT  
NIP. 131558569

Aris Triwiyatno, ST  
NIP. 132230559

