

## Akses Port Paralel (Menggunakan Delphi 7.0) (Drs. Agfianto Eko Putra, M.Si – © 2005)

---

### TENTANG INPOUT32.DLL

Delphi yang merupakan pengembangan lebih lanjut OOP Pascal dengan sistem operasi Windows ternyata telah menghilangkan fungsi khas Pascal dalam akses perangkat keras, yaitu perintah

```
port[alamat] := data atau variabel := port[alamat];
```

Dengan demikian dibutuhkan suatu cara agar Delphi bisa melakukan akses perangkat keras. Salah satunya dengan pustaka `inpout32.dll`, yang bisa didownload dari [www.logix4u.net](http://www.logix4u.net) secara gratis, yang harus disimpan atau ditempatkan pada folder `<drive>:/windows/system32` pada komputer Anda.

Cara penggunaannya sangat mudah, pertama definisikan terlebih dahulu fungsi dan prosedur untuk akses masukan maupun keluaran perangkat keras pada bagian deklarasi Delphi sebagai berikut:

```
Function Inp32(alamat:word):byte; stdcall; external 'inpout32.dll';  
Procedure Out32(alamat:word;data:byte); stdcall; external 'inpout32.dll';
```

Kedua tinggal penggunaan fungsi dan prosedur tersebut secara biasa, fungsi `Inp32` membutuhkan dua parameter yaitu alamat perangkat keras dan variabel hasil pembacaan data dari perangkat keras dengan tipe data byte. Sedangkan prosedur `Out32` membutuhkan dua parameter juga, yaitu alamat perangkat keras dan nilai atau variabel yang menyimpan nilai yang akan dikirimkan ke perangkat keras yang bersangkutan.

### PENDAHULUAN PORT PARALEL

Port Paralel adalah sarana port yang sederhana dan murah untuk membuat projek-projek atau alat-alat yang dikendalikan menggunakan komputer. Sederhana karena, sebagaimana akan dilihat nanti, hanya melibatkan 3 macam register saja, murah karena hampir semua komputer, desktop maupun portable, selalu dilengkapi dengan Port Paralel atau yang dikenal juga dengan Port Printer.

Kesederhanaan dan kemudahan pemrograman menggunakan Port Paralel membuat Port ini semakin populer di kalangan para hobis. Port Paralel ini sering digunakan untuk, misalnya, robot yang dikendalikan komputer, pemrogram mikrokontroler Atmel atau PIC, otomasi rumah, akuisisi data praktis dan lain sebagainya.

### EKSPERIMEN AKSES PORT PARALEL

Standar IEEE 1284 yang dipublikasikan pada tahun 1994 mendefinisikan 5 (lima) macam mode transfer data Port Paralel:

1. Mode Kompatbel;
2. Mode Nibel;
3. Mode Byte;
4. EPP, dan
5. ECP

Port Paralel ini terhubung dengan dunia luar melalui konektor DB25, yang terbagi atas tiga kelompok register, yaitu:

1. Register Data;
2. Register Control, dan
3. Register Status

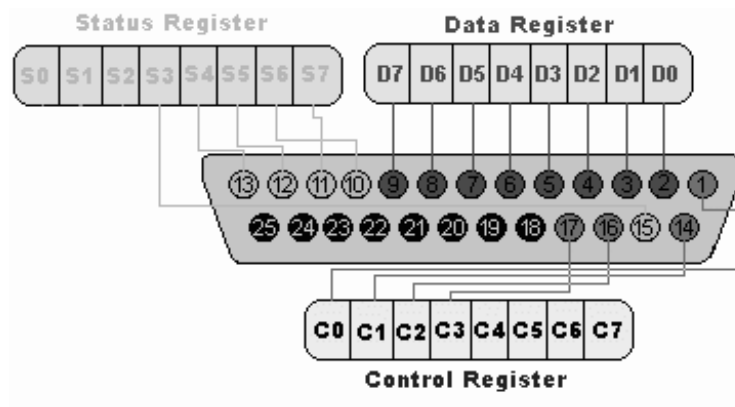
Seperti namanya, masing-masing register tersebut digunakan untuk mentransfer data, melakukan pengontrolan periferal serta memeriksa atau mendapatkan status periferal eksternal. Masing-masing register terhubung ke konektor DB25 tetapi tidak semua bit yang terhubung ke masing-masing pin.

Ada beberapa pin (selain pin-pin pada register data) yang bersifat sungsang atau *inverse*, jika pin tersebut sebagai masukan, maka jika terbaca 0 artinya mendapatkan logika 1 (demikian juga sebaliknya). Jika pin tersebut adalah keluaran, jika dikirimkan logika 1, maka yang terkirim sesungguhnya adalah logika 0 (demikian juga sebaliknya). Selain itu sifatnya normal, terbaca 1 ya 1, terkirim 1 ya 1, demikian seterusnya. Lebih lengkapnya ditunjukkan pada Tabel 1.

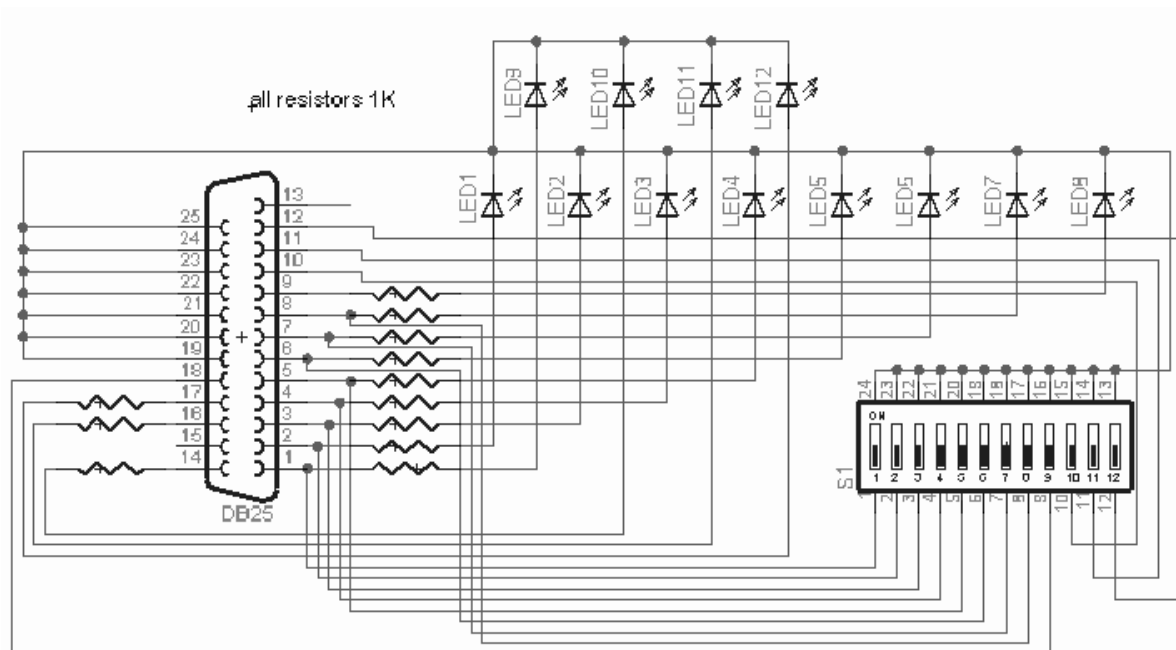
Tabel 1

No.Pin (DB25)	Nama Sinyal	Arah	Register - bit	Sungsang?
1	nStrobe	Out	Control-0	Ya
2	Data0	In/Out	Data-0	Tidak
3	Data1	In/Out	Data-1	Tidak
4	Data2	In/Out	Data-2	Tidak
5	Data3	In/Out	Data-3	Tidak
6	Data4	In/Out	Data-4	Tidak
7	Data5	In/Out	Data-5	Tidak
8	Data6	In/Out	Data-6	Tidak
9	Data7	In/Out	Data-7	Tidak
10	nAck	In	Status-6	Tidak
11	Busy	In	Status-7	Ya
12	Paper-Out	In	Status-5	Tidak
13	Select	In	Status-4	Tidak
14	Linefeed	Out	Control-1	Ya
15	nError	In	Status-3	Tidak
16	nInitialize	Out	Control-2	Tidak
17	nSelect-Printer	Out	Control-3	Ya
18-25	Ground	-	-	-

Daftar pin pada DB25 Port Paralel ditunjukkan pada Gambar 1, sedangkan rangkaian yang digunakan untuk eksperimen ditunjukkan pada Gambar 2. Perlu mendapat perhatian bahwa penggunaan DIP Switch hanya sebagian saja, menyesuaikan dengan bit-bit pada Register Status yang terhubung dengan DB25 (ada 4). Pada Tabel 2 ditunjukkan koneksi antara LED, DIP Switch dan konektor DB25 Port Paralel. Mohon bisa diperhatikan sepenuhnya.



Gambar 1



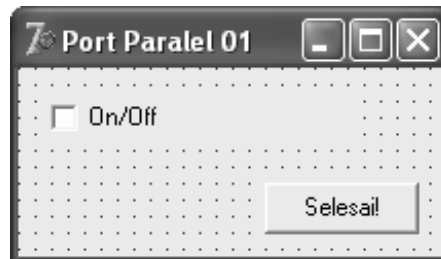
Gambar 2

Tabel 2

Rangkaian	DB25	Rangkaian	DB25
LED1	2 - D0	LED9	1 - /PC0
LED2	3 - D1	LED10	14 - /PC1
LED3	4 - D2	LED11	16 - PC2
LED4	5 - D3	LED12	17 - /PC3
LED5	6 - D4	DIP1	15 - PS3
LED6	7 - D5	DIP2	13 - PS4
LED7	8 - D6	DIP3	12 - PS5
LED8	9 - D7	DIP4	10 - PS6
		DIP5	11 - /PS7
		Gnd	18 s/d 25

## Contoh program 01

Contoh program pertama ini digunakan untuk mematikan dan menghidupkan LED-LED yang terpasang pada pin-pin data. Menggunakan bantuan komponen **TCheckBox**, Anda bisa mengganti dengan komponen lainnya selama bisa digunakan untuk proses **toggle** (ON-OFF secara bergantian). Status dari LED langsung ditunjukkan pada komponen tersebut.



Gambar 3

Komponen	Properti	Prosedur/Fungsi
TCheckBox	Name: CheckBox1 Caption: "On/Off"	CheckBox1Click(Sender: TObject)
TButton	Name: Button1 Caption: "Selesai!"	Button1Click(Sender: TObject)
TForm	Name: TForm1 Caption: "Port Paralel 01"	FormCreate(Sender: TObject)

```

procedure TForm1.CheckBox1Click(Sender: TObject);
begin
  if checkbox1.Checked then
  begin
    out32($378,$ff);
    CheckBox1.Caption := 'Lampu ON';
  end else
  begin
    out32($378,0);
    CheckBox1.Caption := 'Lampu OFF';
  end;
end;

```

```

procedure TForm1.FormCreate(Sender: TObject);
begin
  CheckBox1.Caption := 'Status Lampu LED'
end;

```

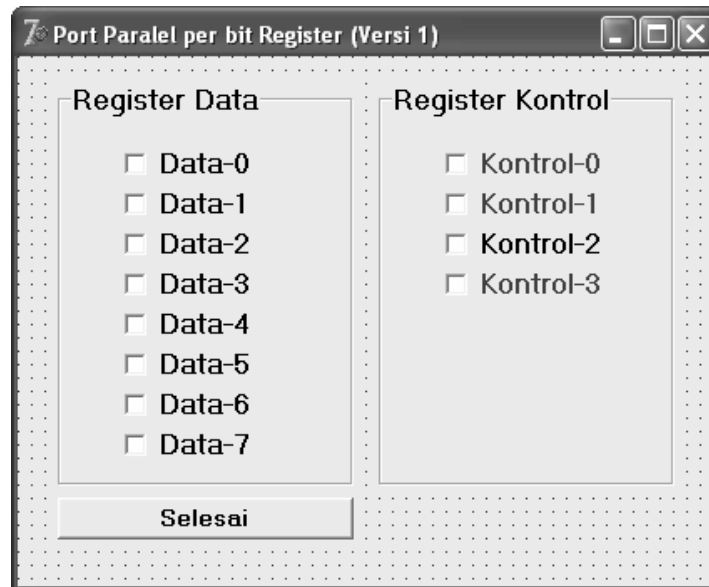
```

procedure TForm1.Button1Click(Sender: TObject);
begin
  Close;
end;

```

## Contoh program 02

Contoh program yang kedua ini lebih spesifik, karena menghidupkan LED baik di pin-pin data maupun kontrol dilakukan satu persatu. Dengan cara memeriksa status komponen **TCheckBox** saat dilakukan *event* pencetangan. Perhatikan adanya keanehan hasil pada saat melakukan pencetangan pada **TCheckBox** untuk Kontrol-0 s/d Kontrol-3. Ingat adanya pin-pin yang sungsang sebagaimana dijelaskan sebelumnya.



Gambar 4

Komponen	Properti	Prosedur/Fungsi
TGroupBox	Name: GroupBox1	-
TCheckBox	Name: data0 Caption: "Data-0"	<b>data0Click (Sender: TObject)</b>
TCheckBox	Name: data1 Caption: "Data-1"	<b>Data1Click (Sender: TObject)</b>
TCheckBox	Name: data2 Caption: "Data-2"	<b>Data2Click (Sender: TObject)</b>
TCheckBox	Name: data3 Caption: "Data-3"	<b>Data3Click (Sender: TObject)</b>
TCheckBox	Name: data4 Caption: "Data-4"	<b>Data4Click (Sender: TObject)</b>
TCheckBox	Name: data5 Caption: "Data-5"	<b>Data5Click (Sender: TObject)</b>
TCheckBox	Name: data6 Caption: "Data-6"	<b>Data6Click (Sender: TObject)</b>
TCheckBox	Name: data7 Caption: "Data-7"	<b>Data7Click (Sender: TObject)</b>
TGroupBox	Name: GroupBox2	-
TCheckBox	Name: kontrol0 Caption: "Kontrol-0"	<b>kontrol0Click (Sender: TObject)</b>
TCheckBox	Name: kontrol1 Caption: "Kontrol-1"	<b>Kontrol1Click (Sender: TObject)</b>

Komponen	Properti	Prosedur/Fungsi
TCheckBox	Name: kontrol2 Caption: "Kontrol-2"	Kontrol2Click(Sender: TObject)
TCheckBox	Name: kontrol3 Caption: "Kontrol-3"	Kontrol3Click(Sender: TObject)
Form	Name: TForm1 Caption: "Port Paralel per bit register (versi 1)"	FormCreate(Sender: TObject)
TButton	Name: Button1 Caption: "Selesai!"	Button1Click(Sender: TObject)

```

procedure TForm1.data0Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data0.Checked then Out32($378, tmp or $01)
  else Out32($378, tmp and $FE);
end;

procedure TForm1.data1Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data1.Checked then Out32($378, tmp or $02)
  else Out32($378, tmp and $FD);
end;

procedure TForm1.data2Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data2.Checked then Out32($378, tmp or $04)
  else Out32($378, tmp and $FB);
end;

procedure TForm1.data3Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data3.Checked then Out32($378, tmp or $08)
  else Out32($378, tmp and $F7);
end;

procedure TForm1.data4Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data4.Checked then Out32($378, tmp or $10)
  else Out32($378, tmp and $EF);
end;

procedure TForm1.data5Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data5.Checked then Out32($378, tmp or $20)
  else Out32($378, tmp and $DF);
end;

```

```
procedure TForm1.data6Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data6.Checked then Out32($378, tmp or $40)
  else Out32($378, tmp and $BF);
end;

procedure TForm1.data7Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data7.Checked then Out32($378, tmp or $80)
  else Out32($378, tmp and $7F);
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  Close;
end;
```

```
procedure TForm1.kontrol0Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if kontrol0.Checked then Out32($37A, tmp or $01)
  else Out32($37A, tmp and $FE);
end;

procedure TForm1.kontrol1Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if kontrol1.Checked then Out32($37A, tmp or $02)
  else Out32($37A, tmp and $FD);
end;

procedure TForm1.kontrol2Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if kontrol2.Checked then Out32($37A, tmp or $04)
  else Out32($37A, tmp and $FB);
end;

procedure TForm1.kontrol3Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if kontrol3.Checked then Out32($37A, tmp or $08)
  else Out32($37A, tmp and $F7);
end;
```

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  Out32($378,$00);
  Out32($37A,$0B);
end;
```

## Contoh program 03

---

Contoh program ketiga ini digunakan untuk memperbaiki adanya pin-pin yang sungsang, perhatikan perubahannya...

```
procedure TForm1.kontrol0Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if not kontrol0.Checked then
  begin
    Out32($37A, tmp or $01);
    kontrol0.Font.Color := ClBlack;
  end else
  begin
    Out32($37A, tmp and $FE);
    kontrol0.Font.Color := ClRed;
  end;
end;

procedure TForm1.kontrol1Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if not kontrol1.Checked then
  begin
    Out32($37A, tmp or $02);
    kontrol1.Font.Color := ClBlack;
  end else
  begin
    Out32($37A, tmp and $FD);
    kontrol1.Font.Color := ClRed;
  end;
end;

procedure TForm1.kontrol2Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if kontrol2.Checked then
  begin
    Out32($37A, tmp or $04);
    kontrol2.Font.Color := ClRed;
  end else
  begin
    Out32($37A, tmp and $FB);
    kontrol2.Font.Color := ClBlack;
  end;
end;

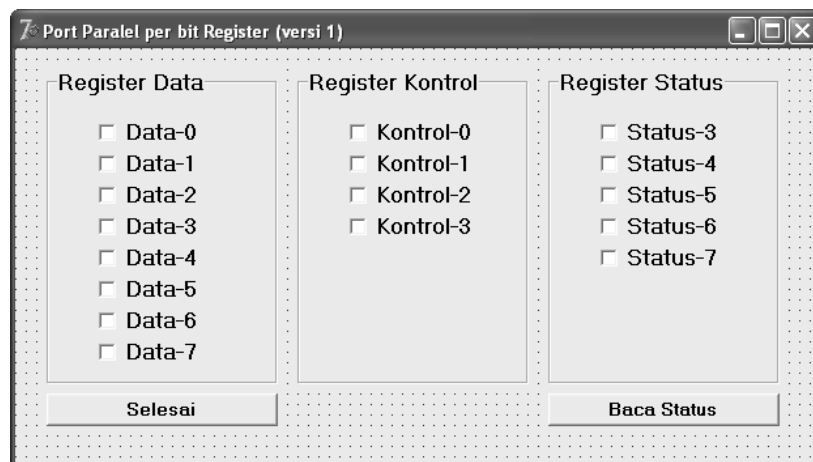
procedure TForm1.kontrol3Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if not kontrol3.Checked then
  begin
    Out32($37A, tmp or $08);
    kontrol3.Font.Color := ClBlack;
  end else
```



```
begin
  Out32($37A, tmp and $F7);
  kontrol3.Font.Color := ClRed;
end;
end;
```

## Contoh program 04

Contoh program keempat jauh lebih lengkap karena selain dapat digunakan untuk menghidupkan dan mematikan LED juga untuk memeriksa status DIP Switch melalui pin-pin status. Ubah DIP Switch kemudian klik tombol "Baca Status", maka data status akan langsung terbaca. Perhatikan juga prosedur yang menangani PS7 karena sifatnya yang sungsang!



Gambar 5

```
procedure TForm1.data0Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data0.Checked then Out32($378, tmp or $01)
  else Out32($378, tmp and $FE);
end;

procedure TForm1.data1Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data1.Checked then Out32($378, tmp or $02)
  else Out32($378, tmp and $FD);
end;

procedure TForm1.data2Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data2.Checked then Out32($378, tmp or $04)
  else Out32($378, tmp and $FB);
end;

procedure TForm1.data3Click(Sender: TObject);
begin
```

```
tmp := Inp32($378);
if data3.Checked then Out32($378, tmp or $08)
else Out32($378, tmp and $F7);
end;

procedure TForm1.data4Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data4.Checked then Out32($378, tmp or $10)
  else Out32($378, tmp and $EF);
end;

procedure TForm1.data5Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data5.Checked then Out32($378, tmp or $20)
  else Out32($378, tmp and $DF);
end;

procedure TForm1.data6Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data6.Checked then Out32($378, tmp or $40)
  else Out32($378, tmp and $BF);
end;

procedure TForm1.data7Click(Sender: TObject);
begin
  tmp := Inp32($378);
  if data7.Checked then Out32($378, tmp or $80)
  else Out32($378, tmp and $7F);
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  Close;
end;

procedure TForm1.kontrol10Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if not kontrol10.Checked then Out32($37A, tmp or $01)
  else Out32($37A, tmp and $FE);
end;

procedure TForm1.kontrol11Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if not kontrol11.Checked then Out32($37A, tmp or $02)
  else Out32($37A, tmp and $FD);
end;

procedure TForm1.kontrol12Click(Sender: TObject);
begin
  tmp := Inp32($37A);
```

```
if kontrol2.Checked then Out32($37A, tmp or $04)
else Out32($37A, tmp and $FB);
end;

procedure TForm1.kontrol3Click(Sender: TObject);
begin
  tmp := Inp32($37A);
  if not kontrol3.Checked then Out32($37A, tmp or $08)
  else Out32($37A, tmp and $F7);
end;

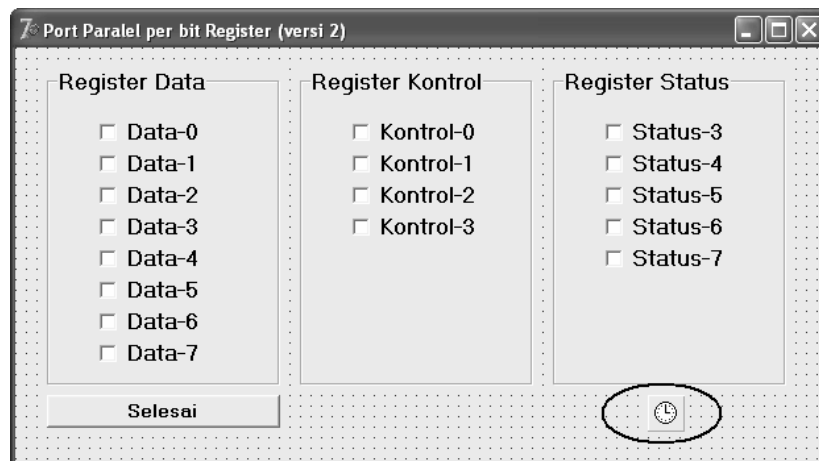
procedure TForm1.Button2Click(Sender: TObject);
begin
  tmp := Inp32($379);
  if (tmp and $08) = $08 then stat3.Checked := true
  else stat3.Checked := false;
  if (tmp and $10) = $10 then stat4.Checked := true
  else stat4.Checked := false;
  if (tmp and $20) = $20 then stat5.Checked := true
  else stat5.Checked := false;
  if (tmp and $40) = $40 then stat6.Checked := true
  else stat6.Checked := false;
  if (tmp and $80) = $00 then stat7.Checked := true
  else stat7.Checked := false;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  Out32($378, $00);
  Out32($37A, $0B);
end;

end.
```

## Contoh program 05

Sama seperti contoh keempat, hanya saja pemeriksaan status DIP Switch dilakukan secara periodik!



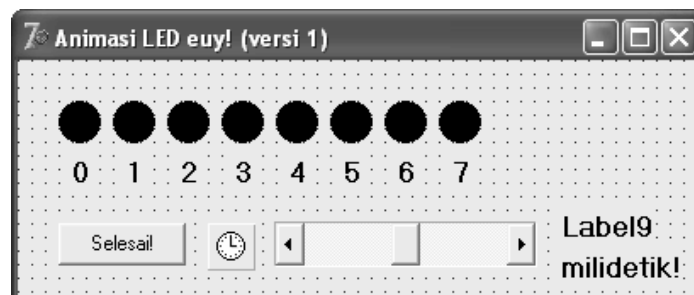
Gambar 6

Prosedur pada objek Timer (tanda lingkaran pada Gambar 6)

```
procedure TForm1.Timer1(Sender: TObject);
begin
  tmp := Inp32($379);
  if (tmp and $08) = $08 then stat3.Checked := true
  else stat3.Checked := false;
  if (tmp and $10) = $10 then stat4.Checked := true
  else stat4.Checked := false;
  if (tmp and $20) = $20 then stat5.Checked := true
  else stat5.Checked := false;
  if (tmp and $40) = $40 then stat6.Checked := true
  else stat6.Checked := false;
  if (tmp and $80) = $00 then stat7.Checked := true
  else stat7.Checked := false;
end;
```

## Contoh program 06

Contoh keenam ini digunakan untuk menganimasikan LED yang terhubung ke pin data dari LED0 ke LED7 dan berulang lagi, kecepatannya ditentukan oleh posisi **TScrollBar**.



Gambar 7

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  data := $01;
  out32($378,data);
  scrollbar1.Position := 250;
  label9.Caption := inttostr(scrollbar1.Position);
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
  out32($378,data);
  led0.Brush.Color := clRed * (data and $01);
  led1.Brush.Color := clRed * ((data shr 1) and $01);
  led2.Brush.Color := clRed * ((data shr 2) and $01);
  led3.Brush.Color := clRed * ((data shr 3) and $01);
  led4.Brush.Color := clRed * ((data shr 4) and $01);
  led5.Brush.Color := clRed * ((data shr 5) and $01);
  led6.Brush.Color := clRed * ((data shr 6) and $01);
  led7.Brush.Color := clRed * ((data shr 7) and $01);
end;
```

```
data := data shl 1;
if data > $80 then data := $01;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
    close;
end;

procedure TForm1.ScrollBar1Change(Sender: TObject);
begin
    label9.Caption := inttostr(scrollbar1.Position);
    timer1.Interval := scrollbar1.Position;
end;

end.
```

## Contoh program 07

Sama seperti contoh keenam hanya saja gerakannya dari LED7 s/d ke LED0, perhatikan perubahannya...

```
procedure TForm1.Timer1Timer(Sender: TObject);
begin
    out32($378,data);
    led0.Brush.Color := clRed * (data and $01);
    led1.Brush.Color := clRed * ((data shr 1) and $01);
    led2.Brush.Color := clRed * ((data shr 2) and $01);
    led3.Brush.Color := clRed * ((data shr 3) and $01);
    led4.Brush.Color := clRed * ((data shr 4) and $01);
    led5.Brush.Color := clRed * ((data shr 5) and $01);
    led6.Brush.Color := clRed * ((data shr 6) and $01);
    led7.Brush.Color := clRed * ((data shr 7) and $01);
    data := data shr 1;
    if data < $01 then data := $80;
end;
```

## Contoh program 08

Pada contoh ke delapan digunakan 2x7segment



Gambar 8

```
...
type
  TForm1 = class(TForm)
    Button1: TButton;
    digitpul: TEdit;
    digitsat: TEdit;
    set_7: TButton;
    procedure Button1Click(Sender: TObject);
    procedure set_7Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

Function Inp32(alamat:word):byte; stdcall; external 'inout32.dll';
Procedure      Out32(alamat:word;data:byte);      stdcall;      external
'inout32.dll';

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
  Close;
end;

procedure TForm1.set_7Click(Sender: TObject);
var kirim,pul,sat : byte;
begin
  pul := strtoint(digitpul.Text);
  sat := strtoint(digitsat.Text);
  kirim := (pul shl 4) or sat;
  out32($378,kirim);
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  digitpul.Text := '1';
  digitsat.Text := '2';
end;

end.
```

---

## Contoh program 09

---

Pada contoh ke sembilan digunakan 2x7segment untuk membuat pencacah Naik (UP)



Gambar 9

```
...
Function Inp32(alamat:word):byte; stdcall; external 'inpout32.dll';
Procedure      Out32(alamat:word;data:byte);      stdcall;      external
'inpout32.dll';

var
  Form1: TForm1;
  data,disp : byte;

implementation

{$R *.dfm}

procedure TForm1.btStopClick(Sender: TObject);
begin
  jam.Enabled := False;
end;

procedure TForm1.btMulaiClick(Sender: TObject);
begin
  jam.Enabled := True;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  data := 0;
  disp := 0;
  jam.Enabled := False;
end;

procedure TForm1.JamTimer(Sender: TObject);
begin
  label1.Caption := inttostr(disp);
  Out32($378,data);
  inc(data);
  inc(disp);
  if (data and $0A) = $0A then data := data + 6;
  if data > $99 then begin
    data := 0;
    disp := 0;
  end;
end;
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  close
end;
```

## Contoh program 09

Pada contoh ke sepuluh digunakan 2x7segment untuk membuat pencacah Turun (DOWN)



Gambar 10

```
Function Inp32(alamat:word):byte; stdcall; external 'inport32.dll';
Procedure      Out32(alamat:word;data:byte);      stdcall;      external
'inport32.dll';

var
  Form1: TForm1;
  data,disp : integer;

implementation

{$R *.dfm}

procedure TForm1.btStopClick(Sender: TObject);
begin
  jam.Enabled := False;
end;

procedure TForm1.btMulaiClick(Sender: TObject);
begin
  jam.Enabled := True;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  data := $99;
  disp := 99;
  jam.Enabled := False;
end;
```



```
procedure TForm1.JamTimer(Sender: TObject);
begin
  label1.Caption := inttostr(data);
  Out32($378,data);
  dec(data);
  dec(data);
  if (data and $0F) = $0F then data := data - 6;
  if data < $00 then begin
    data := $99;
    disp := 99;
  end;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  close
end;
```

-- selesai --