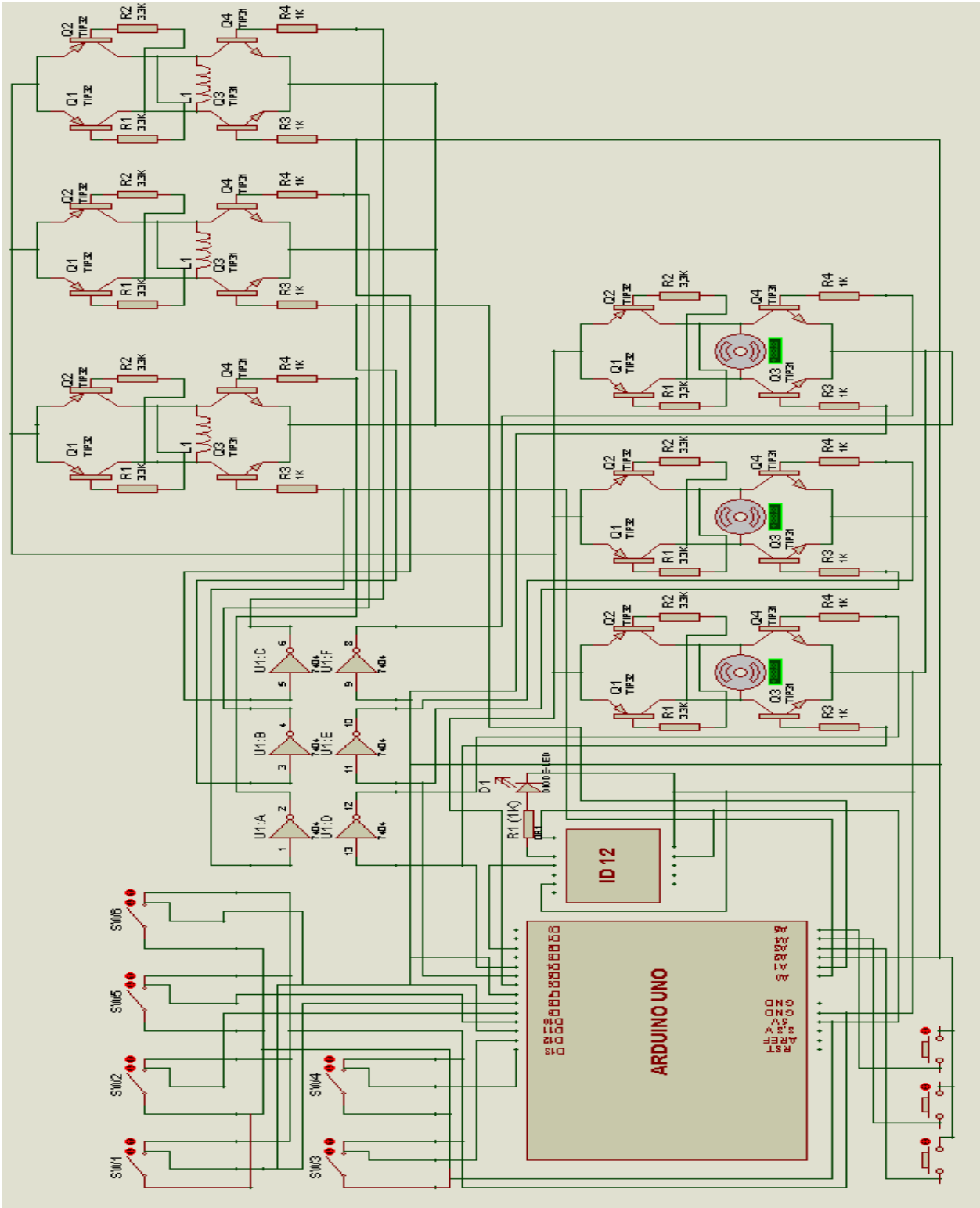


LAMPIRAN



Listing Program pada Arduino kit

```
#include <SoftwareSerial.h>
#define rxPin 2
#define txPin 3

SoftwareSerial rfid = SoftwareSerial(rxPin,txPin);

char tag = 0;
int val = 0;
int motor1 = 4;
int motor2 = 5;
int enable = 6;
int motor3 = 7;
int saklar1= 8;
int saklar2= 9;
int saklar3= 10;
int saklar4= 11;
int saklar5= 12;
int saklar6= 13;
int solenoid1 = 14;
int solenoid2 = 15;
int solenoid3 = 16;
int push1 = 17;
int push2 = 18;
int push3 = 19;

int DC1 = 0;
int sole1 = 0;

int DC2 = 0;
int sole2 = 0;

int DC3 = 0;
int sole3 = 0;

void setup() {
  Serial.begin(9600);
  rfid.begin(9600);
  pinMode (push1, INPUT);
  pinMode (push2, INPUT);
  pinMode (push3, INPUT);
  pinMode (saklar1, INPUT);
  pinMode (saklar2, INPUT);
  pinMode (saklar3, INPUT);
```

```

pinMode (saklar4, INPUT);
pinMode (saklar5, INPUT);
pinMode (saklar6, INPUT);
pinMode (enable,OUTPUT);
pinMode (motor1,OUTPUT);
pinMode (motor2,OUTPUT);
pinMode (motor3, OUTPUT);
pinMode (solenoid1, OUTPUT);
pinMode (solenoid2, OUTPUT);
pinMode (solenoid3, OUTPUT);
}

void loop () {
tag =rfid.read();
if(tag!=-1){
Serial.print(tag);

val =Serial.read();
if (val!=-1){

if (val=='a'){
digitalWrite(solenoid1,HIGH);
analogWrite(enable,255);
delay(50);
analogWrite(enable,0);
sole1=1;
}

if (val=='b'){
digitalWrite(solenoid2,HIGH);
analogWrite(enable,255);
delay(50);
analogWrite(enable,0);
sole2=1;
}

if (val=='c'){
digitalWrite(solenoid3,HIGH);
analogWrite(enable,255);
delay(50);
analogWrite(enable,0);
sole3=1;
}
}
}
TutupSlot1();

```

```

TutupSlot2();
TutupSlot3();
}

void TutupSlot1(){
  if(sole1==1){
    digitalWrite(motor1,HIGH);
    analogWrite(enable,255);
    DC1=1;
  }
  if ((digitalRead(saklar1)==1)&&(DC1==1)){
    analogWrite(enable,0);
    DC1=0;
  }
  if ((digitalRead(push1) == 0) &&(DC1 == 0)){
    digitalWrite(motor1,LOW);
    analogWrite(enable,255);
  }
  if ((digitalRead(saklar2)==1)&& ((digitalRead(saklar1)) == 0)){
    analogWrite(enable,0);
    DC1=1;
  }
  if ((digitalRead(saklar2) == 1)&& (DC1 == 1)){
    digitalWrite (solenoid1,LOW);
    analogWrite(enable,255);
    delay(50);
    analogWrite(enable,0);
    sole1=0;
    DC1=0;
  }
}

void TutupSlot2(){
  if(sole2==1){
    digitalWrite(motor2,HIGH);
    analogWrite(enable,255);
    DC2=1;
  }
  if ((digitalRead(saklar3)==1)&&(DC2==1)){
    analogWrite(enable,0);
    DC2=0;
  }
  if ((digitalRead(push2) == 0) &&(DC2 == 0)){
    digitalWrite(motor2,LOW);
    analogWrite(enable,255);
  }
}

```

```

}
if ((digitalRead(saklar4) == 1) && ((digitalRead(saklar3)) == 0)) {
  analogWrite(enable, 0);
  DC2 = 1;
}
if ((digitalRead(saklar4) == 1) && (DC2 == 1)) {
  digitalWrite(solenoid2, LOW);
  analogWrite(enable, 255);
  delay(50);
  analogWrite(enable, 0);
  sole2 = 0;
  DC2 = 0;
}
}
}
void TutupSlot3() {
  if(sole3 == 1) {
    digitalWrite(motor3, HIGH);
    analogWrite(enable, 255);
    DC3 = 1;
  }
  if ((digitalRead(saklar5) == 1) && (DC3 == 1)) {
    analogWrite(enable, 0);
    DC3 = 0;
  }
  if ((digitalRead(push3) == 0) && (DC3 == 0)) {
    digitalWrite(motor3, LOW);
    analogWrite(enable, 255);
  }
  if ((digitalRead(saklar6) == 1) && ((digitalRead(saklar3)) == 0)) {
    analogWrite(enable, 0);
    DC3 = 1;
  }
  if ((digitalRead(saklar6) == 1) && (DC3 == 1)) {
    digitalWrite(solenoid3, LOW);
    analogWrite(enable, 255);
    delay(50);
    analogWrite(enable, 0);
    sole3 = 0;
    DC3 = 0;
  }
}
}
}

```

Listing Program Pada Delphi

Program Menu Utama

```
unit U_MenuUtama;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, XPMAn, ExtCtrls, ToolWin, ComCtrls, Buttons, Menus, jpeg;
type
  TfrmMenu = class(TForm)
    SpeedButton1: TSpeedButton;
    Bevel1: TBevel;
    StatusBar1: TStatusBar;
    CoolBar1: TCoolBar;
    spbtnPelanggan: TSpeedButton;
    spbtnPembelian: TSpeedButton;
    SpeedButton2: TSpeedButton;
    Timer1: TTimer;
    SpeedButton3: TSpeedButton;
    SpeedButton4: TSpeedButton;
    Image1: TImage;
    procedure Timer1Timer(Sender: TObject);
    procedure spbtnPelangganClick(Sender: TObject);
    procedure spbtnPembelianClick(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure SpeedButton3Click(Sender: TObject);
    procedure SpeedButton4Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  frmMenu: TfrmMenu;
implementation
uses U_DaftarPelanggan, U_InputPelanggan, U_Module, U_HapusPelanggan, U_SlotParkir,
  U_DataLog;
{$R *.dfm}
procedure TfrmMenu.Timer1Timer(Sender: TObject);
begin
  StatusBar1.Panels[1].Text := FormatDateTime('dd mmmm yyyy', now);
  StatusBar1.Panels[2].Text := FormatDateTime('hh:mm:ss', now);
end;
procedure TfrmMenu.spbtnPelangganClick(Sender: TObject);
begin
  frmPelanggan.show;
end;
procedure TfrmMenu.spbtnPembelianClick(Sender: TObject);
```

```

begin
frmDPelanggan.show;
end;
procedure TfrmMenu.SpeedButton2Click(Sender: TObject);
begin
application.Terminate;
end;
procedure TfrmMenu.SpeedButton3Click(Sender: TObject);
begin
frmDaftarParkir.Show;
end;
procedure TfrmMenu.SpeedButton4Click(Sender: TObject);
begin
frmDataLog.Show;
end;
end.

```

Program Menu Registrasi

```

unit U_InputPelanggan;

```

```

interface

```

```

uses

```

```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, Grids, DBGrids, StdCtrls, Buttons, ComCtrls, ExtCtrls;

```

```

type

```

```

TfrmPelanggan = class(TForm)
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label6: TLabel;
edKodePlg: TEdit;
edNamaPlg: TEdit;
edAlamatPlg: TEdit;
edRfid: TEdit;
edHP: TEdit;
BitBtn1: TBitBtn;
BitBtn4: TBitBtn;
StatusBar1: TStatusBar;
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);

```

```

private

```

```

{ Private declarations }

```

```

public

```

```

{ Public declarations }

```

```

end;

```



```

var
    frmPelanggan: TfrmPelanggan;

implementation

uses U_MenuUtama, U_Module, U_DaftarPelanggan,U_SlotParkir;

{$R *.dfm}

procedure TfrmPelanggan.BitBtn4Click(Sender: TObject);
begin
    frmMenu.show;
    frmPelanggan.Hide;
end;

procedure TfrmPelanggan.BitBtn1Click(Sender: TObject);
begin
    With dm.QData Do
        Begin
            Close;
            Sql.Clear;
            Sql.Add('Insert Into member(NoRfid, KodePlg,NamaPlg,AlamatPlg,HP) '+
                'Values(:nr, :kp, :np, :al,:hp)');
            Prepare;
            ParamByName('nr').AsString := EdRfid.Text;
            ParamByName('kp').AsString := EdKodePlg.Text;
            ParamByname('np').AsString := EdNamaPlg.Text;
            ParamByname('al').AsString := EdAlamatPlg.Text;
            ParamByname('hp').AsString := EdHP.Text;
            ExecSql;
        End;
        MessageDlg('Apakah data telah diisi dengan lengkap?',mtWarning,[mbOK],0);
        dm.trSlot.Commit;

        EdRfid.Clear;
        EdKodePlg.Clear;
        EdNamaPlg.Clear;
        EdAlamatPlg.Clear;
        EdHP.Clear;
        EdKodePlg.SetFocus;

        With dm.QProses Do
            Begin
                Close;
                Sql.Clear;
                Sql.Add('Select * From member');
                Open;
                frmDPelanggan.show;
                frmPelanggan.Hide;
            End;
        end;
end;

```

end.

Program Menu Daftar Slot Parkir

```
unit U_SlotParkir;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, CPort, CPortCtl, ExtCtrls, Buttons, ComCtrls, Grids,  
DBGrids;
```

```
type
```

```
TfrmDaftarParkir = class(TForm)
```

```
ComPort1: TComPort;
```

```
Edit1: TEdit;
```

```
Label6: TLabel;
```

```
edSlot: TEdit;
```

```
btnKeluar: TBitBtn;
```

```
btnTambah: TBitBtn;
```

```
Label15: TLabel;
```

```
StatusBar1: TStatusBar;
```

```
Timer2: TTimer;
```

```
Edit2: TEdit;
```

```
Edit3: TEdit;
```

```
BitBtn1: TBitBtn;
```

```
Edit4: TEdit;
```

```
Timer3: TTimer;
```

```
Label7: TLabel;
```

```
Label8: TLabel;
```

```
Label16: TLabel;
```

```
Label1: TLabel;
```

```
BitBtn2: TBitBtn;
```

```
Label10: TLabel;
```

```
Label11: TLabel;
```

```
Button2: TButton;
```

```
edCom: TEdit;
```

```
ComLed1: TComLed;
```

```
edRfid: TEdit;
```

```
Edit11: TEdit;
```

```
Edit12: TEdit;
```

```
Edit13: TEdit;
```

```
Edit14: TEdit;
```

```
Edit10: TEdit;
```

```
Edit15: TEdit;
```

```
Label12: TLabel;
```

```
edNama: TEdit;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```

BitBtn3: TBitBtn;
DBGrid1: TDBGrid;

procedure ComPort1RxChar(Sender: TObject; Count: Integer);
procedure btnTambahClick(Sender: TObject);
procedure btnKeluarClick(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure Timer3Timer(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmDaftarParkir: TfrmDaftarParkir;
  awal,akhir : tDateTime;
  state: Integer;
  Str:String;
  add:boolean;
  login:boolean;
  logout:boolean;
implementation
  uses U_Module, U_MenuUtama;
  {$R *.dfm}

procedure TfrmDaftarParkir.ComPort1RxChar(Sender: TObject; Count: Integer);

begin
  comport1.ReadStr(Str,count);
  edRfid.Text:=edRfid.Text+Str;
  if (copy(edRfid.text,1,1)=' ') and (copy(edRfid.text,16,1)=' ') then
  begin
    str:=Copy(edRfid.Text,2,length(edRfid.text)-4);
    edRfid.Text:=Str;
    edit1.Text := edRfid.Text;
    edRfid.Clear;

    with dm.qData do
    begin
      close;
      sql.clear;
      sql.add('select namaplg from member');

```

```

sql.Add('where norfid =:id');
prepare;
parambyname('id').AsString :=edit1.Text;
open;
edNama.Text := fields[0].AsString;

if(edNama.Text="") then
begin
showmessage('KARTU TIDAK TERDAFTAR');
end
else
begin
if (str = edit12.Text) then
comport1.WriteStr('a');
if (str = edit13.Text) then
comport1.WriteStr('b');
if (str = edit10.Text) then
comport1.WriteStr('c');
end;
End;
end;
end;
procedure TfrmDaftarParkir.btnKeluarClick(Sender: TObject);
begin
close;
end;

procedure TfrmDaftarParkir.Timer2Timer(Sender: TObject);
begin
akhir := time;
edit3.Text := timetostr(akhir);
edit4.Text := datetostr(date);
end;

procedure TfrmDaftarParkir.FormCreate(Sender: TObject);
begin
timer2.Interval := 1000;
with dm.qProses Do
begin
close;
sql.clear;
sql.add('Select * From slotparkir');
open;
end;

end;

procedure TfrmDaftarParkir.BitBtn1Click(Sender: TObject);
begin
timer2.Enabled := false;

```

```

timer2.Enabled := true;
edit3.Text := timetostr(time);
timer3.Enabled := false;
timer3.Enabled := true;
edit2.Text := timetostr(time);
edit1.Clear;
edslot.Clear;
ednama.Clear;
edrfid.Clear;
end;

```

```

procedure TfrmDaftarParkir.Timer3Timer(Sender: TObject);
begin
akhir := time;
edit2.Text := timetostr(akhir);
end;

```

```

procedure TfrmDaftarParkir.btnTambahClick(Sender: TObject);
begin
With dm.QData Do
Begin
Close;
Sql.Clear;
Sql.Add('Insert Into parkirlog(norfid, slot, tanggal, Login) '+
'Values(:no, :sl, :tg, :li)');
Prepare;
ParamByName('no').AsString := trim(edit1.Text);
ParamByName('tg').AsString := trim(Edit4.Text);
ParamByName('li').AsString := trim(Edit3.Text);
ParamByName('sl').AsString := trim(Edslot.Text);
ExecSql;

timer2.Enabled := false;
awal := time;
edit3.Text := timetostr(time);
End;
dm.trSlot.Commit;

With dm.QProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select * From slotparkir');
Open;
end;
end;

```

```

procedure TfrmDaftarParkir.BitBtn2Click(Sender: TObject);
begin

```

```

with dm.qData do
begin
close;
sql.Clear;
sql.Add('select slot,tanggal,login from parkirlog');
sql.add('where norfid =:id');
prepare;
parambyname('id').AsString := edit1.Text;
open;
edSlot.Text :=fields[0].AsString;
edit4.Text :=fields[1].AsString;
edit3.Text := fields[2].AsString;
end;

```

```

with dm.qData do
begin
close;
sql.Clear;
sql.add('update slotparkir set norfid =:dn where slot =:st');
prepare;
parambyname('st').AsString := trim(edSlot.Text);
parambyname('dn').AsString := trim("");
execsql;
end;
dm.trSlot.Commit;

```

```

with dm.qData do
begin
sql.Clear;
sql.Add('delete from parkirlog');
sql.add('where norfid =:id');
prepare;
parambyname('id').AsString := edit1.Text;
execsql;
end;
dm.trSlot.Commit;

```

```

with dm.qData do
begin
Close;
Sql.Clear;
Sql.Add('Insert Into log(norfid, slot, tanggal, Login,logout) '+
'Values(:no, :sl, :tg, :li,:lo)');
Prepare;
ParamByName('no').AsString := trim(edit1.Text);
ParamByName('sl').AsString := trim(edSlot.Text);
ParamByName('tg').AsString := trim(Edit4.Text);
ParamByName('li').AsString := trim(Edit3.Text);
ParamByName('lo').AsString := trim(Edit2.Text);
ExecSql;
timer3.Enabled := false;

```

```

    awal := time;
    edit2.Text := timetostr(time);
    End;
    dm.trSlot.Commit;

    With dm.qProses Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select * From slotparkir');
    Open;
    end;
    end;

procedure TfrmDaftarParkir.FormActivate(Sender: TObject);

begin
showmessage('Harap Lakukan Setting Comport');
statusbar1.Panels[1].Text:='Disconnected';
edCom.SetFocus;
end;

procedure TfrmDaftarParkir.Button2Click(Sender: TObject);
begin
comport1.Port:=edCom.Text;
comport1.Open;
statusbar1.Panels[1].Text:='Connected';
statusbar1.Panels[0].Text:='Port in use ' + comport1.Port;
end;

procedure TfrmDaftarParkir.BitBtn3Click(Sender: TObject);
begin
with dm.qData do
begin
close;
sql.Clear;
sql.add('update slotparkir set Norfid =:dn where slot =:st');
prepare;
parambyname('st').AsString := trim(edSlot.Text);
parambyname('dn').AsString := trim(edit1.Text);
execsql;
end;
dm.trSlot.Commit;

    With Dm.QData Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select norfid From slotparkir');

```

```
    sql.add('where slot =:st');
  Prepare;
  parambyname('st').AsString := trim(edit11.Text);
  Open;
  edit12.Text := trim(Fields[0].AsString);
```

```
  With Dm.QData Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select norfid From slotparkir');
  sql.add('where slot =:sl');
  Prepare;
  parambyname('sl').AsString := trim(edit15.Text);
  Open;
  edit13.Text := trim(Fields[0].AsString);
```

```
  With Dm.QData Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select norfid From slotparkir');
  sql.add('where slot =:sl');
  Prepare;
  parambyname('sl').AsString := trim(edit14.Text);
  Open;
  edit10.Text := trim(Fields[0].AsString);
```

```
  With dm.qProses Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select * From slotparkir');
  Open;
  end;
  end;
  end;
  end;
end;
```

end.

Program Daftar Member

```
unit U_DaftarPelanggan;
```

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, Grids, DBGrids, Buttons, ToolWin, ComCtrls,
  DB, IBCustomDataSet, IBQuery, RpCon, RpConDS, RpDefine, RpRave;
```



```

type
  TfrmDPelanggan = class(TForm)
    btnExit: TBitBtn;
    dbgMobil: TDBGrid;
    StatusBar1: TStatusBar;
    btnSimpan: TBitBtn;
    BitBtn2: TBitBtn;
    procedure btnSimpanClick(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure btnExitClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmDPelanggan: TfrmDPelanggan;

implementation

uses U_InputPelanggan, U_MenuUtama, U_Module, U_HapusPelanggan,U_SlotParkir;

{$R *.dfm}

procedure TfrmDPelanggan.btnSimpanClick(Sender: TObject);
begin
  frmPelanggan.Show;
  frmDpelanggan.Hide;
end;

procedure TfrmDPelanggan.BitBtn2Click(Sender: TObject);
begin
  frmHapusPelanggan.show;
end;

procedure TfrmDPelanggan.btnExitClick(Sender: TObject);
begin
  close;
end;

procedure TfrmDPelanggan.FormActivate(Sender: TObject);
begin
  with dm.qProses Do
    begin
      close;
      sql.clear;
      sql.add('Select * From member order by kodeplg');
    end;
end;

```

```
    open;  
    end;
```

```
end;
```

```
end.
```

Program Daftar Log

```
unit U_DataLog;
```

```
interface
```

```
uses
```

```
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
    Dialogs, StdCtrls, Buttons, Grids, DBGrids;
```

```
type
```

```
    TfrmDataLog = class(TForm)  
        DBGrid1: TDBGrid;  
        btnExit: TBitBtn;  
        procedure FormActivate(Sender: TObject);  
        procedure btnExitClick(Sender: TObject);  
    private  
        { Private declarations }  
    public  
        { Public declarations }  
    end;
```

```
var
```

```
    frmDataLog: TfrmDataLog;
```

```
implementation
```

```
    uses U_Module, U_MenuUtama;  
    {$R *.dfm}
```

```
    procedure TfrmDataLog.FormActivate(Sender: TObject);  
    begin  
        with dm.qProses Do  
            begin  
                close;  
                sql.clear;  
                sql.add('Select * From log order by norfid');  
                open;  
                end;  
            end;  
    end;
```

```
    procedure TfrmDataLog.btnExitClick(Sender: TObject);  
    begin  
        close;  
        end;
```

```

end.

Program Hapus Member
unit U_HapusPelanggan;

interface

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

type
  TfrmHapusPelanggan = class(TForm)
    Label1: TLabel;
    edKodePlg: TEdit;
    btnOk: TBitBtn;
    btnCancel: TBitBtn;
    procedure btnOkClick(Sender: TObject);
    procedure btnCancelClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmHapusPelanggan: TfrmHapusPelanggan;

implementation

uses U_Module, U_InputPelanggan, U_DaftarPelanggan;

{$R *.dfm}

procedure TfrmHapusPelanggan.btnOkClick(Sender: TObject);
begin
  With dm.QProses Do
  Begin
    Close;
    Sql.Clear;
    Sql.Add('Delete From slotparkir');
    Sql.Add('Where NoRfid =:nr');
    Prepare;
    ParamByName('nr').AsString := EdKodePlg.Text;
    ExecSql;
  End;

  dm.trSlot.Commit;
  MessageDlg('Hapus data yang telah dipilih?',mtWarning,[mbOK],0);
  frmDPelanggan.show;
  frmHapuspelanggan.Hide;

```

```
With dm.QProses Do
Begin
  Close;
  Sql.Clear;
  Sql.Add('Select * From slotparkir');
  Open;
end;
end;
procedure TfrmHapusPelanggan.btnCancelClick(Sender: TObject);
begin
  frmDPelanggan.show;
  frmHapuspelanggan.Hide;
end;

end
```

