

LAMPIRAN A

LIST PROGRAM MIKROKONTROLER

List Program Mikrokontroler 1

\$MOD51

Inisialisasi:

```
MOV  SCON,#50H
MOV  TMOD,#20H
MOV  TH1,#0FDH
SETB TR1
MOV  R0,#01H
MOV  R3,#00H
AJMP Start
```

Terima2: ;ask to send

```
CLR  RI
MOV  A,sbuf
CJNE A,#61H,Terima3
CJNE R0,#10H,NoData
AJMP Kirim
```

Terima3: ;access denied

```
CJNE A,#67H,Terima4
MOV  P2,#00H
MOV  A,#65H
CLR  TI
MOV  SBUF,A
JNB  TI,$
AJMP Inisialisasi
```

Terima4:

```
CJNE A,#66H,Inisialisasi ;access granted
MOV  P2,#0FFH
```

```
MOV A,#65H
CLR TI
MOV SBUF,A
JNB TI,$
AJMP Inisialisasi
```

NoData:

```
MOV A,#25H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,#65H
CLR TI
MOV SBUF,A
JNB TI,$
AJMP Inisialisasi
```

Terima:

;address checking

```
CJNE R3,#00H,Terima2
CLR RI
MOV A,sbuf
CJNE A,#52H,Start
INC R3
AJMP Start
```

Kirim:

```
MOV A,#25H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
```

```

MOV A,R4
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R5
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R6
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R7
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,#62H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
AJMP Inisialisasi

```

Delay:

```
MOV R2,#0FFH
```

Ulang:

```
MOV R1,#0FFH
DJNZ R1,$
DJNZ R2,Ulang
RET
```

Start: ;Check Input

```
JB RI,Terima
MOV P1,#07FH
JNB P1.3,Angka1
JNB P1.2,Angka2
JNB P1.1,Angka3
JNB P1.0,HurufA
MOV P1,#0BFH
JNB P1.3,Angka4
JNB P1.2,Angka5
JNB P1.1,Angka6
JNB P1.0,HurufB
AJMP Start2
```

Angka1:

```
MOV A,#31H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.3,$
AJMP Start
```

Angka2:

```
MOV A,#32H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
```

```
JNB P1.2,$
```

```
AJMP Start
```

Angka3:

```
MOV A,#33H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.1,$
```

```
AJMP Start
```

HurufA:

```
MOV A,#41H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.0,$
```

```
AJMP Start
```

Angka4:

```
MOV A,#34H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.3,$
```

```
AJMP Start
```

Angka5:

```
MOV A,#35H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.2,$
```

AJMP Start

Angka6:

```
MOV A,#36H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.1,$
AJMP Start
```

HurufB:

```
MOV A,#42H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Next1:

```
CJNE R0,#02H,Next2
MOV R5,A
INC R0
ACALL Delay
ACALL Delay
AJMP Start
```

Start2:

```
MOV P1,#0DFH
JNB P1.3,Angka7
JNB P1.2,Angka8
JNB P1.1,Angka9
JNB P1.0,HurufC
MOV P1,#0EFH
```

```
JNB P1.3,RTS
JNB P1.2,Angka0
JNB P1.1,Batal
JNB P1.0,HurufD
AJMP Start
```

Next2:

```
CJNE R0,#03H,Next3
MOV R6,A
INC R0
ACALL Delay
ACALL Delay
AJMP Start
```

RTS: ;Ready To Send

```
MOV R0,#10H
JNB P1.3,$
AJMP Start
```

Angka7:

```
MOV A,#37H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.3,$
AJMP Start
```

Angka8:

```
MOV A,#38H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.2,$
```


AJMP Start

Angka9:

```
MOV A,#39H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.1,$
AJMP Start
```

HurufC:

```
MOV A,#43H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Angka0:

```
MOV A,#30H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.2,$
AJMP Start
```

HurufD:

```
MOV A,#44H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Batal:

JNB P1.1,\$

AJMP Inisialisasi

Next3:

CJNE R0,#04H,Batal

MOV R7,A

ACALL Delay

AJMP Start

END

List Program Mikrokontroler 2

\$MOD51

Inisialisasi:

```
MOV  SCON,#50H
MOV  TMOD,#20H
MOV  TH1,#0FDH
SETB TR1
MOV  R0,#01H
MOV  R3,#00H
AJMP Start
```

Terima2: ;ask to send

```
CLR  RI
MOV  A,sbuf
CJNE A,#61H,Terima3
CJNE R0,#10H,NoData
AJMP Kirim
```

Terima3: ;access denied

```
CJNE A,#67H,Terima4
MOV  P2,#00H
MOV  A,#65H
CLR  TI
MOV  SBUF,A
JNB  TI,$
AJMP Inisialisasi
```

Terima4:

```
CJNE A,#66H,Inisialisasi ;access granted
MOV  P2,#0FFH
```

```
MOV A,#65H
CLR TI
MOV SBUF,A
JNB TI,$
AJMP Inisialisasi
```

NoData:

```
MOV A,#35H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,#65H
CLR TI
MOV SBUF,A
JNB TI,$
AJMP Inisialisasi
```

Terima:

;address checking

```
CJNE R3,#00H,Terima2
CLR RI
MOV A,sbuf
CJNE A,#53H,Start
INC R3
AJMP Start
```

Kirim:

```
MOV A,#35H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
```

```

MOV A,R4
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R5
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R6
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R7
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,#62H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
AJMP Inisialisasi

```

Delay:

```
MOV R2,#0FFH
```

Ulang:

```
MOV R1,#0FFH
DJNZ R1,$
DJNZ R2,Ulang
RET
```

Start:

```
JB RI,Terima
MOV P1,#07FH
JNB P1.3,Angka1
JNB P1.2,Angka2
JNB P1.1,Angka3
JNB P1.0,HurufA
MOV P1,#0BFH
JNB P1.3,Angka4
JNB P1.2,Angka5
JNB P1.1,Angka6
JNB P1.0,HurufB
AJMP Start2
```

Angka1:

```
MOV A,#31H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.3,$
AJMP Start
```

Angka2:

```
MOV A,#32H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
```

```
JNB P1.2,$
```

```
AJMP Start
```

Angka3:

```
MOV A,#33H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.1,$
```

```
AJMP Start
```

HurufA:

```
MOV A,#41H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.0,$
```

```
AJMP Start
```

Angka4:

```
MOV A,#34H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.3,$
```

```
AJMP Start
```

Angka5:

```
MOV A,#35H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.2,$
```

AJMP Start

Angka6:

```
MOV A,#36H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.1,$
AJMP Start
```

HurufB:

```
MOV A,#42H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Next1:

```
CJNE R0,#02H,Next2
MOV R5,A
INC R0
ACALL Delay
ACALL Delay
AJMP Start
```

Start2:

```
MOV P1,#0DFH
JNB P1.3,Angka7
JNB P1.2,Angka8
JNB P1.1,Angka9
JNB P1.0,HurufC
MOV P1,#0EFH
```



```
JNB P1.3,RTS
JNB P1.2,Angka0
JNB P1.1,Batal
JNB P1.0,HurufD
AJMP Start
```

Next2:

```
CJNE R0,#03H,Next3
MOV R6,A
INC R0
ACALL Delay
ACALL Delay
AJMP Start
```

RTS: ;Ready To Send

```
MOV R0,#10H
JNB P1.3,$
AJMP Start
```

Angka7:

```
MOV A,#37H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.3,$
AJMP Start
```

Angka8:

```
MOV A,#38H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.2,$
```

AJMP Start

Angka9:

```
MOV A,#39H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.1,$
AJMP Start
```

HurufC:

```
MOV A,#43H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Angka0:

```
MOV A,#30H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.2,$
AJMP Start
```

HurufD:

```
MOV A,#44H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Batal:

JNB P1.1,\$

AJMP Inisialisasi

Next3:

CJNE R0,#04H,Batal

MOV R7,A

ACALL Delay

AJMP Start

END

List Program Mikrokontroler 3

\$MOD51

Inisialisasi:

```
MOV  SCON,#50H
MOV  TMOD,#20H
MOV  TH1,#0FDH
SETB TR1
MOV  R0,#01H
MOV  R3,#00H
AJMP Start
```

Terima2: ;ask to send

```
CLR  RI
MOV  A,sbuf
CJNE A,#61H,Terima3
CJNE R0,#10H,NoData
AJMP Kirim
```

Terima3: ;access denied

```
CJNE A,#67H,Terima4
MOV  P2,#00H
MOV  A,#65H
CLR  TI
MOV  SBUF,A
JNB  TI,$
AJMP Inisialisasi
```

Terima4:

```
CJNE A,#66H,Inisialisasi ;access granted
MOV  P2,#0FFH
```

```
MOV A,#65H
CLR TI
MOV SBUF,A
JNB TI,$
AJMP Inisialisasi
```

NoData:

```
MOV A,#45H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,#65H
CLR TI
MOV SBUF,A
JNB TI,$
AJMP Inisialisasi
```

Terima: ;address checking

```
CJNE R3,#00H,Terima2
CLR RI
MOV A,sbuf
CJNE A,#54H,Start
INC R3
AJMP Start
```

Kirim:

```
MOV A,#45H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
```

```

MOV A,R4
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R5
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R6
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,R7
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
MOV A,#62H
CLR TI
MOV SBUF,A
JNB TI,$
ACALL Delay
AJMP Inisialisasi

```

Delay:

```
MOV R2,#0FFH
```

Ulang:

```
MOV R1,#0FFH
DJNZ R1,$
DJNZ R2,Ulang
RET
```

Start:

```
JB RI,Terima
MOV P1,#07FH
JNB P1.3,Angka1
JNB P1.2,Angka2
JNB P1.1,Angka3
JNB P1.0,HurufA
MOV P1,#0BFH
JNB P1.3,Angka4
JNB P1.2,Angka5
JNB P1.1,Angka6
JNB P1.0,HurufB
AJMP Start2
```

Angka1:

```
MOV A,#31H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.3,$
AJMP Start
```

Angka2:

```
MOV A,#32H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
```

```
JNB P1.2,$
```

```
AJMP Start
```

Angka3:

```
MOV A,#33H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.1,$
```

```
AJMP Start
```

HurufA:

```
MOV A,#41H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.0,$
```

```
AJMP Start
```

Angka4:

```
MOV A,#34H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.3,$
```

```
AJMP Start
```

Angka5:

```
MOV A,#35H
```

```
CJNE R0,#01H,Next1
```

```
MOV R4,A
```

```
INC R0
```

```
JNB P1.2,$
```


AJMP Start

Angka6:

```
MOV A,#36H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.1,$
AJMP Start
```

HurufB:

```
MOV A,#42H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Next1:

```
CJNE R0,#02H,Next2
MOV R5,A
INC R0
ACALL Delay
ACALL Delay
AJMP Start
```

Start2:

```
MOV P1,#0DFH
JNB P1.3,Angka7
JNB P1.2,Angka8
JNB P1.1,Angka9
JNB P1.0,HurufC
MOV P1,#0EFH
```

```
JNB P1.3,RTS
JNB P1.2,Angka0
JNB P1.1,Batal
JNB P1.0,HurufD
AJMP Start
```

Next2:

```
CJNE R0,#03H,Next3
MOV R6,A
INC R0
ACALL Delay
ACALL Delay
AJMP Start
```

RTS: ;Ready To Send

```
MOV R0,#10H
JNB P1.3,$
AJMP Start
```

Angka7:

```
MOV A,#37H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.3,$
AJMP Start
```

Angka8:

```
MOV A,#38H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.2,$
```

AJMP Start

Angka9:

```
MOV A,#39H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.1,$
AJMP Start
```

HurufC:

```
MOV A,#43H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Angka0:

```
MOV A,#30H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.2,$
AJMP Start
```

HurufD:

```
MOV A,#44H
CJNE R0,#01H,Next1
MOV R4,A
INC R0
JNB P1.0,$
AJMP Start
```

Batal:

JNB P1.1,\$

AJMP Inisialisasi

Next3:

CJNE R0,#04H,Batal

MOV R7,A

ACALL Delay

AJMP Start

END

LAMPIRAN B

LIST PROGRAM DELPHI

unit Hotel;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, Mask, DBCtrls, DB, ADODB, ExtCtrls, Grids, DBGrids,
ComCtrls, CPort, CPortCtl;

type

TForm1 = class(TForm)
 PageControl1: TPageControl;
 TabSheet1: TTabSheet;
 TabSheet2: TTabSheet;
 TabSheet3: TTabSheet;
 DBGrid1: TDBGrid;
 DBNavigator1: TDBNavigator;
 Label1: TLabel;
 ComboBox1: TComboBox;
 Edit1: TEdit;
 DataSource1: TDataSource;
 ADOTable1: TADOTable;
 ADOConnection1: TADOConnection;
 ADOTable1Nama: TWideStringField;
 ADOTable1Alamat: TWideStringField;
 ADOTable1WaktuCheckIn: TDateTimeField;
 ADOTable1Kamar: TIntegerField;
 ADOTable1Password: TIntegerField;
 ADOTable1UangMuka: TBCDField;

Label2: TLabel;
DBEdit1: TDBEdit;
Label3: TLabel;
DBEdit2: TDBEdit;
Label4: TLabel;
DBEdit3: TDBEdit;
Label5: TLabel;
DBEdit4: TDBEdit;
Label6: TLabel;
DBEdit5: TDBEdit;
Label7: TLabel;
DBEdit6: TDBEdit;
ADOTable1WaktuCheckOut: TDateTimeField;
ADOTable1Biaya: TBCDField;
ADOTable1Discount: TIntegerField;
ADOTable1Tagihan: TBCDField;
Label8: TLabel;
DBEdit7: TDBEdit;
Label9: TLabel;
DBEdit8: TDBEdit;
Label10: TLabel;
DBEdit9: TDBEdit;
Label11: TLabel;
DBEdit10: TDBEdit;
Label12: TLabel;
DBEdit11: TDBEdit;
Label13: TLabel;
DBEdit12: TDBEdit;
Label14: TLabel;

```
DBEdit13: TDBEdit;
Label15: TLabel;
DBEdit14: TDBEdit;
Label16: TLabel;
DBEdit15: TDBEdit;
Label17: TLabel;
DBEdit16: TDBEdit;
Button1: TButton;
ComPort1: TComPort;
Button2: TButton;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Button3: TButton;
Timer1: TTimer;
Button4: TButton;
Timer2: TTimer;
procedure DBGrid1TitleClick(Column: TColumn);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure FormShow(Sender: TObject);
procedure ComPort1RxChar(Sender: TObject; Count: Integer);
procedure Button2Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Button4Click(Sender: TObject);
```



```

    procedure Timer2Timer(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation

{$R *.dfm}

function RandomPassword(PLen: Integer): string;
var
    str: string;
begin
    Randomize;
    str := '0123456789';
    Result := '';
    repeat
        Result := Result + str[Random(Length(str)) + 1];
    until
        (Length(Result) = PLen)
end;

procedure TForm1.DBGrid1TitleClick(Column: TColumn);
begin

```

```

    ADOTable1.IndexFieldNames := Column.FieldName;
end;

procedure TForm1.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
    If Key = chr(VK_RETURN) Then
        Begin
            Key := #0;
            AdoTable1.IndexFieldNames := ComboBox1.Text;
            ADOTable1.Locate(ComboBox1.Text,Edit1.Text,[loPartialKey]);
        End;
end;

procedure TForm1.FormShow(Sender: TObject);
Var I : Integer;
begin
    ComboBox1.Clear;
    For I := 0 To ADOTable1.FieldCount-1 Do
        If ADOTable1.Fields[i].DataType = ftWideString
            Then ComboBox1.Items.Add(ADOTable1.Fields[i].FieldName);
end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
Var
    Str: String;
    w: integer;
begin
    ComPort1.ReadStr(Str,Count);
    edit3.Text := Str;

```

```

w:=120;
if Str = '%' then edit4.Text := '101'
else
if Str = '5' then edit4.text := '102'
else
if Str = 'E' then edit4.text := '103'
else
if Str = 'b' then
begin
ADOTable1.First;
while not ADOTable1.Eof do
begin
If
(Edit4.Text = ADOTable1.FieldByName('Kamar').AsString )and
(Edit5.Text = ADOTable1.FieldByName('Password').AsString )
Then
begin
Edit6.text := 'benar';
end
else
begin
end;
ADOTable1.Next;
end;
if edit6.Text ='benar' then
begin
if edit4.text = '101' then Edit2.Text := 'Rf' else
if edit4.text = '102' then Edit2.Text := 'Sf' else
if edit4.text = '103' then Edit2.Text := 'Tf';

```

```

edit3.Clear;
edit4.Clear;
edit6.Clear;
Timer2.Enabled:= true;
end
else
begin
if edit4.text = '101' then Edit2.Text := 'Rg' else
if edit4.text = '102' then Edit2.Text := 'Sg' else
if edit4.text = '103' then Edit2.Text := 'Tg';
edit3.Clear;
edit4.Clear;
edit6.clear;
Timer2.Enabled:= true;
end;
end
else
if Str = 'e' then
begin
edit5.Clear;
edit2.Clear;
timer1.Enabled := false;
timer2.Enabled := false;
sleep(w);
button3click(self)
end
else
edit5.Text := edit5.Text + Str;
end;

```

```
procedure TForm1.Button2Click(Sender: TObject);
Var
  Str : String;
begin
  Str := Edit2.Text;
  Comport1.WriteStr(Str);
end;
```

```
procedure TForm1.FormActivate(Sender: TObject);
begin
  Comport1.Open;
  Comport1.WriteStr('R');
  Comport1.Close;
  Comport1.Open;
  Comport1.WriteStr('a');
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  DBEdit5.Text := RandomPassword(4);
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);
var Str: string;
begin
  if edit4.text = '101' then Str:= 'S' else
  if edit4.text = '102' then Str:= 'T' else
  if edit4.text = '103' then Str:= 'R';
```

```
Comport1.Open;
Comport1.WriteStr(Str);
Str:='a';
Comport1.WriteStr(Str);
timer1.Enabled := true;
end;
```

```
procedure TForm1.Timer1Timer(Sender: TObject);
var Str: string;
begin
    if edit4.text = '101' then Str:= 'S' else
    if edit4.text = '102' then Str:= 'T' else
    if edit4.text = '103' then Str:= 'R';
    Comport1.Open;
    Comport1.WriteStr(Str);
    Str:='a';
    Comport1.WriteStr(Str);
end;
```

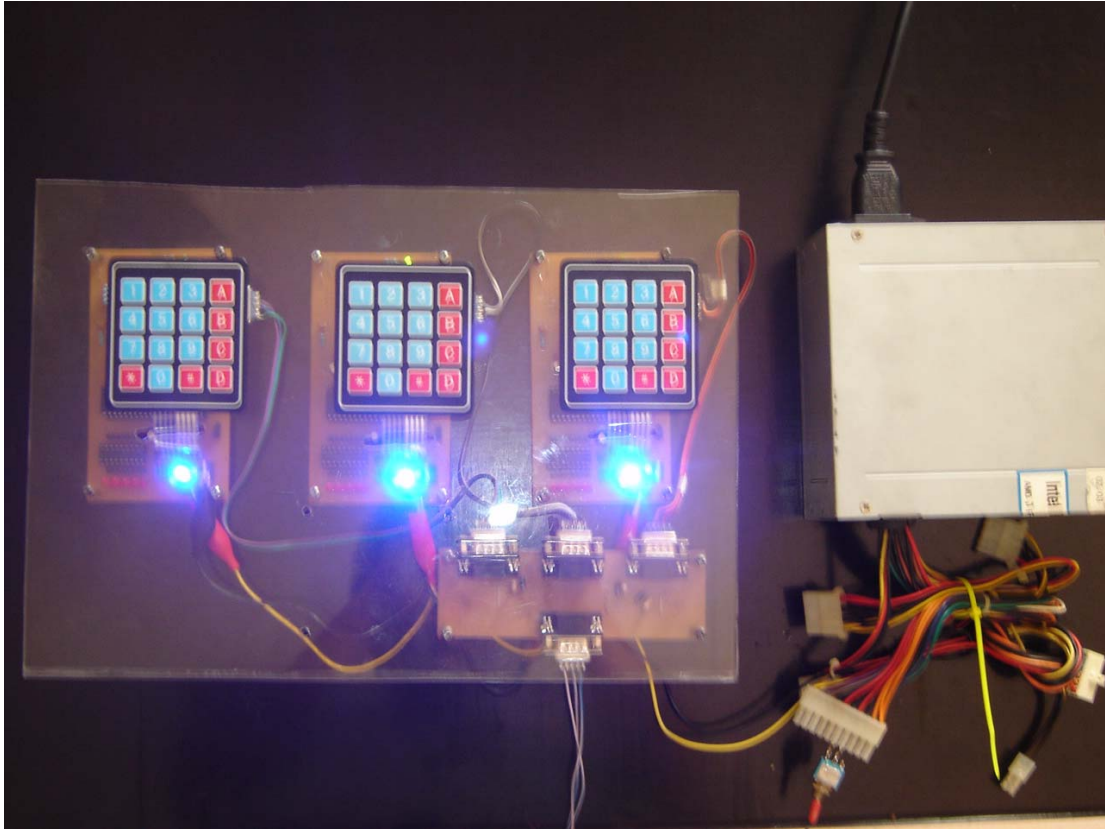
```
procedure TForm1.Button4Click(Sender: TObject);
begin
    Comport1.Close;
    edit3.Clear;
    edit4.clear;
end;
```

```
procedure TForm1.Timer2Timer(Sender: TObject);
var Str: string;
begin
```

```
Str:= edit2.text;  
Comport1.WriteStr(Str);  
edit4.text:= '101';  
end;  
  
end.
```

LAMPIRAN C

FOTO ALAT



Rangkaian Prototipe Mikrokontroler Pengendali