

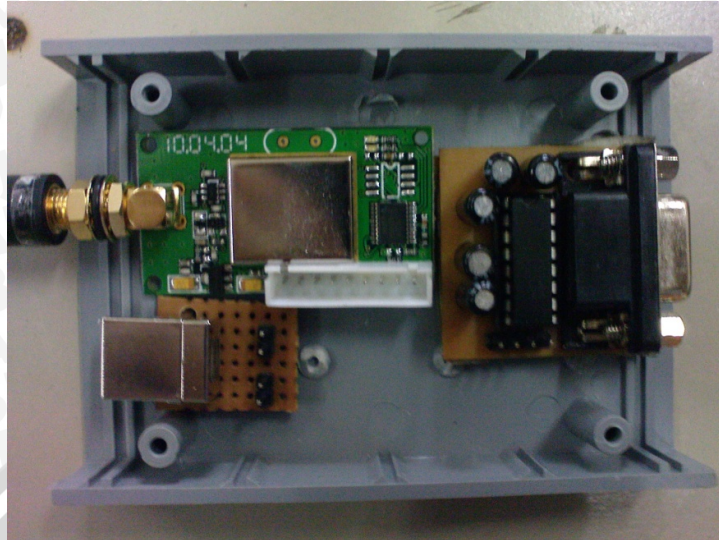
LAMPIRAN



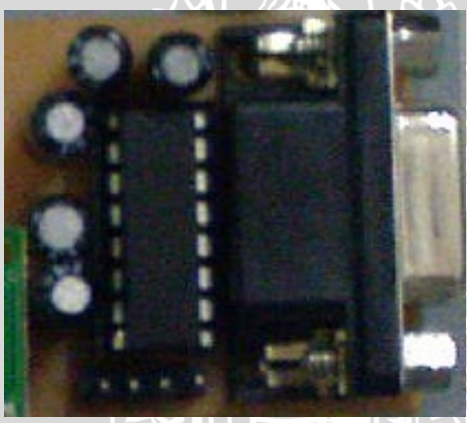
LAMPIRAN I

FOTO ALAT





Gambar 1 Alat secara keseluruhan



Gambar 2 Rangkaian MAX 232



Gambar 3 Modul RF YS-1020



UNIVERSITAS BRAWIJAYA



LAMPIRAN II

LISTING PROGRAM
Perangkat Lunak PC



PROGRAM UTAMA

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, IniFiles, ComCtrls, XComDrv, ExtCtrls, intel_hex, DateUtils,  
Menus;
```

```
type
```

```
buffer= packed array of byte;
```

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

```
  BtTulisFlash: TButton;
```

```
  MemoDataHex: TMemo;
```

```
  OpenDialogHex: TOpenDialog;
```

```
  LabelPanjangData: TLabel;
```

```
  MemoDataASCII: TMemo;
```

```
  XComm1: TXComm;
```

```
  Timer1: TTimer;
```

```
  BtSetting: TButton;
```

```
  BtConnect: TButton;
```

```
  Label2: TLabel;
```

```
  Label3: TLabel;
```

```
  LabelDeviceSignature: TLabel;
```

```
  CmbJenisMK: TComboBox;
```

```
  Panel1: TPanel;
```

```
  BtReloadFlash: TButton;
```

```
  Label5: TLabel;
```

```
  Label6: TLabel;
```

```
  DirektoriFile: TEdit;
```

```
  Panel2: TPanel;
```

```
  CmbPilihMK: TComboBox;
```

```
  Label7: TLabel;
```

```
  Label8: TLabel;
```

```
  BtPilihMK: TButton;
```

```
  BtHapusMemori: TButton;
```

```
  MainMenu1: TMainMenu;
```

```
  MnFile: TMenuItem;
```

```
  MnLoadFlash: TMenuItem;
```

```
  MnLoadEEPROM: TMenuItem;
```

```
  N1: TMenuItem;
```

```
  MnSaveFlash: TMenuItem;
```

```
  MnSaveEEPROM: TMenuItem;
```

```
  N2: TMenuItem;
```

```
  MnExit: TMenuItem;
```

```
  MnProgram: TMenuItem;
```

```
  MnHapusMemori: TMenuItem;
```

```
  N3: TMenuItem;
```

```
  MnTulisFlash: TMenuItem;
```

```
  MnTulisEEPROM: TMenuItem;
```

```
  N4: TMenuItem;
```

```
  MnFuseBits: TMenuItem;
```

```
  MnHelp: TMenuItem;
```

```
  MnPetunjuk: TMenuItem;
```

```
  MnAbout: TMenuItem;
```

```
  BtTulisEEPROM: TButton;
```

```
  BtBacaFlash: TButton;
```

```
  BtBacaEEPROM: TButton;
```

```
  BtReloadEEPROM: TButton;
```

```

GroupBox1: TGroupBox;
GroupBox2: TGroupBox;
Panel3: TPanel;
OpenDialogEEPROM: TOpenDialog;
SaveDialogFlash: TSaveDialog;
SaveDialogEEPROM: TSaveDialog;
Panel4: TPanel;
procedure BtTulisFlashClick(Sender: TObject);
//procedure ComPort1RxChar(Sender: TObject; Count: Integer);
procedure BtSettingClick(Sender: TObject);
procedure BtConnectClick(Sender: TObject);
procedure CmbJenisMKChange(Sender: TObject);
procedure BtReloadFlashClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure BtPilihMKClick(Sender: TObject);
procedure BtHapusMemoriClick(Sender: TObject);
procedure MnLoadFlashClick(Sender: TObject);
procedure MnFuseBitsClick(Sender: TObject);
procedure MnExitClick(Sender: TObject);
procedure MnLoadEEPROMClick(Sender: TObject);
procedure BtReloadEEPROMClick(Sender: TObject);
procedure MnSaveFlashClick(Sender: TObject);
procedure MnSaveEEPROMClick(Sender: TObject);
procedure BtBacaFlashClick(Sender: TObject);
procedure XComm1Data(Sender: TObject; const Received: Cardinal);
procedure BtTulisEEPROMClick(Sender: TObject);
procedure BtBacaEEPROMClick(Sender: TObject);

private
  { Private declarations }
public

```

```

  { Public declarations }
  procedure matikanFungsional;
  procedure aktifkanFungsional;
  procedure aktifkanFungsiMK;
  procedure matikanFungsiMK;
end;

var
  Form1: TForm1;
  panjang : string;
  Connect: boolean;
  Buffer_data: widestring;
  Buffer_data_2, sdata: string;
  result: string;
  i: integer;
  ReadyState, ReadyState_2: boolean;

// File .ini
Inifile:TiniFile;
tipeAVR: string;
deviceSignature: string;

const
  OPERASI_TULIS_FLASH= 10;// ASCII 'F'
  OPERASI_BACA_FLASH= 20;// ASCII 'f'
  OPERASI_TULIS_EEPROM= 30;
  OPERASI_BACA_EEPROM= 40;
  OPERASI_TULIS_FUSEBITS= 50;// ASCII 'B'
  OPERASI_BACA_FUSEBITS= 60;// ASCII 'b'
  OPERASI_HAPUS_CHIP= 70;// ASCII 'H'

```

implementation

```
uses Unit2, Unit3;
```

```
type
```

```
  TByteArr = array of byte;
```

```
{ $R *.dfm }
```

```
//ubah label panjang data
```

```
function hex(s:string):string;
```

```
var
```

```
  i:integer;
```

```
begin
```

```
  s:='';
```

```
  i:=1;
```

```
  while i<=length(IntToStr(panjangData)) do
```

```
  begin
```

```
    s:=s+chr(strtoint('$'+copy(IntToStr(panjangData),i,2)));
```

```
    i:=i+2;
```

```
  end;
```

```
panjang:=s;
```

```
end;
```

```
//ubah string to byte
```

```
function StringToBytes(aString: String): TByteArr;
```

```
var
```

```
  i: integer;
```

```
begin
```

```
  SetLength( Result, Length(aString) );
```

```
  for i := 0 to Length(aString) - 1 do
```

```
    Result[i] := ord(aString[i + 1]) { - 48 } ;
```

```
end;
```

```
//ubah byte to string
```

```
function BytesToString(aBytes : TByteArr): String;
```

```
begin
```

```
  Result := PChar(aBytes);
```

```
end;
```

```
//hidupkan fungsi pilih MK
```

```
procedure TForm1.aktifkanFungsiMK;
```

```
begin
```

```
  Panel2.Enabled:= true;
```

```
  Panel2.Visible:= true;
```

```
end;
```

```
//matikan fungsi pilih MK
```

```
procedure TForm1.matikanFungsiMK;
```

```
begin
```

```
  Panel1.Enabled:= false;
```

```
  Panel1.Visible:= false;
```

```
end;
```

```
//matikan semua fungsi
```

```
procedure TForm1.matikanFungsional;
```

```
begin
```

```
  Panel1.Enabled:= false;
```

```
  Panel1.Visible:= false;
```

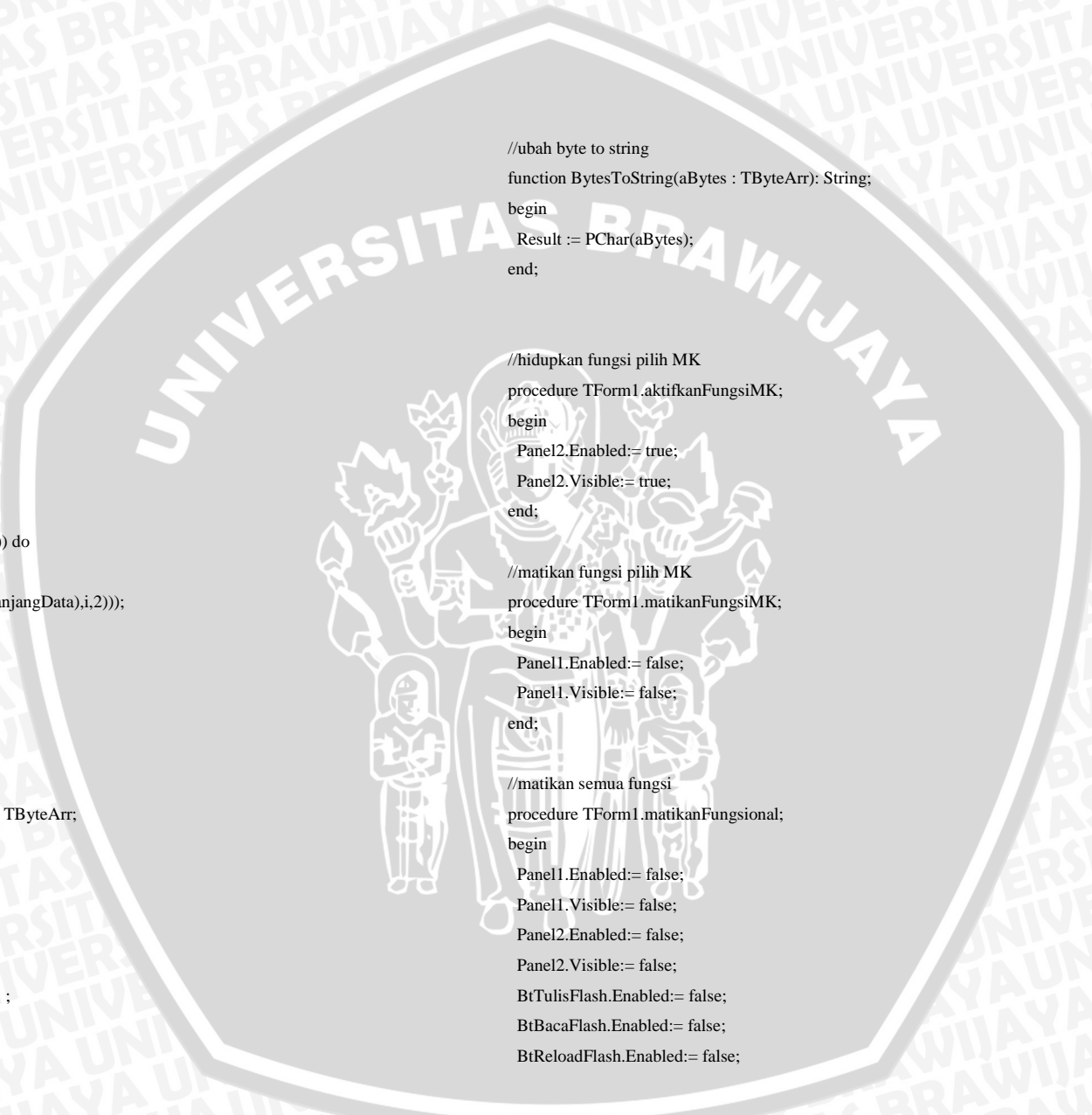
```
  Panel2.Enabled:= false;
```

```
  Panel2.Visible:= false;
```

```
  BtTulisFlash.Enabled:= false;
```

```
  BtBacaFlash.Enabled:= false;
```

```
  BtReloadFlash.Enabled:= false;
```



UNIVERSITAS BRAWIJAYA


```

BtTulisEEPROM.Enabled:= false;
BtBacaEEPROM.Enabled:= false;
BtReloadEEPROM.Enabled:= false;
BtHapusMemori.Enabled:= false;
MnProgram.Enabled:= false;
end;

//hidupkan semua fungsi
procedure TForm1.aktifkanFungsional;
var
  indeksChip: integer;
begin
  Panel1.Enabled:= true;
  Panel1.Visible:= true;
  Panel2.Enabled:= true;
  Panel2.Visible:= true;
  BtTulisFlash.Enabled:= true;
  BtBacaFlash.Enabled:= true;
  BtTulisEEPROM.Enabled:= true;
  BtBacaEEPROM.Enabled:= true;
  BtHapusMemori.Enabled:= true;
  MnProgram.Enabled:= true;

  Inifile:= TIniFile.Create(ChangeFileExt(Application.ExeName, '.ini'));
  tipeAVR:= Inifile.ReadString('jenis_avr_terpilih','chip,');
  indeksChip:= CmbJenisMK.Items.IndexOf(tipeAVR);
  CmbJenisMK.ItemIndex:= indeksChip;

  CmbJenisMK.Change(nil);
end;

```

```

//kirim flash
procedure TForm1.BtTulisFlashClick(Sender: TObject);
var
  buffer : TByteArray;
  j,i,k,r : integer;

  sukses,Str,awal,header1,id_device,header2,id_idperintah,device_signature,devsignature,pembatas,panjang_data,panj_data,devicesig,akhir,head,hasil,sum,pd,u,x, checksum : string;
begin
  //untuk panjang data;
  pd:= IntToHex(panjangData,4);
  panj_data:= "";
  r:=1;
  while r<=length(pd) do
  begin
    panj_data:=panj_data+chr(strtoint('$'+copy(pd,r,2)));
    r:=r+2;
    If ( r > 4) then
      break;
    end;
    panjang_data:=panj_data;
    //untuk jenis mk
    buffer := StringToBytes(deviceSignature);
    devicesig := BytesToString(buffer);
    devsignature:= "";
    i:=1;
    while i<=length(devicesig) do
    begin
      devsignature:=devsignature+chr(strtoint('$'+copy(devicesig,i,2)));
      i:=i+3;
      If ( i > 8) then

```

```

break;
end;
device_signature:=devsignature;
//header
awal := chr(strtoint('$'+('00')));
header1 := '<';
id_device := chr(strtoint('$'+('81')));
header2 := '>';
//id perintah
id:= introstr(OPERASI_TULIS_FLASH);
idperintah:= chr(strtoint('$'+(id)));
pembatas := '|';
hasil :=
awal+awal+awal+awal+awal+header1+id_device+header2+idperintah+pembatas+panjang_data+pembatas+device_signature+pembatas+akhir;
akhir := 'OK';
for j := 0 to (-1 + MemoDataHex.Lines.Count) do
  Str := Str + MemoDataHex.Lines[j];
  head:= "";
  for k := 1 to length(Str) div 2 do
    head:= head + Char(StrToInt('$'+Copy(Str,(k-1)*2+1,2)));
    checksum:= checksum xor head;
    sum := header1+id_device+header2+idperintah+pembatas+head+checksum+akhir;
    XComm1.SendString(hasil);
    sukses:=chr(strtoint('$'+('AA')));
    u:=pembatas+sukses+pembatas+sukses;
    x:=u+u;
    if (XComm1.WaitForString([u], 2000) <> -1) then
      begin
        XComm1.SendString(sum);
        if (XComm1.WaitForString([x], 200000) <> -1) then

```

```

begin
  ShowMessage('Pengiriman Selesai');
end
else ShowMessage('pengiriman gagal');
end
else ShowMessage('pengiriman perintah gagal');
end;

//load setting COM
procedure TForm1.BtSettingClick(Sender: TObject);
begin
  Form2.Show;
end;

//connect COM
procedure TForm1.BtConnectClick(Sender: TObject);
begin
  if (connect=false) then begin
    XComm1.OpenDevice;
    connect:=true;
    aktifkanFungsiMK;
    BtConnect.Caption:= 'Disconnect';
  end
  else begin
    XComm1.CloseDevice;
    matikanFungsional;
    connect:=false;
    BtConnect.Caption:= 'Connect';
  end;
end;
end;

```

```
//jenis mikrokontroller
procedure TForm1.CmbJenisMKChange(Sender: TObject);
begin
  Inifile:=TIniFile.Create(ChangeFileExt(Application.ExeName,'.ini') );
  try
    tipeAVR:= CmbJenisMK.Text;
    deviceSignature:= Inifile.ReadString('jenis_avr',CmbJenisMK.Text,"");
    LabelDeviceSignature.Caption:= deviceSignature;
  finally
    Inifile.Free;
  end;
end;
```

```
//Reload File *hex
procedure TForm1.BtReloadFlashClick(Sender: TObject);
var
  baris: string;
  fileFlash: textfile;
  kodeError: integer;
begin
  DirektoriFile.Text:=OpenDialogHex.FileName;
  panjangData:= 0;
  MemoDataHex.Lines.Clear;
  MemoDataASCII.Lines.Clear;
  AssignFile(fileFlash, OpenDialogHex.FileName);
  Reset(fileFlash);
  isiHex:= "";
  while not eof(fileFlash) do
  begin
    Readln(fileFlash, baris);
    kodeError:= bacaBaris(baris, bufferHex);
```

```
    if kodeError = 0 then
    begin
      MemoDataHex.Lines.Add(DataHex);
      isiHex:= isiHex+DataHex;
      MemoDataASCII.Lines.Add(DataASCII)
    end
    else
    begin
      if kodeError = -1 then
        ShowMessage('Bukan intel hex!')
      else if kodeError = -2 then
        ShowMessage('Record type invalid!')
      else if kodeError = -3 then
        ShowMessage('Checksum error!');
      break;
    end;
    end;
    LabelPanjangData.Caption:= IntToStr(panjangData)+' bytes';
    CloseFile(fileFlash);
  end;

  //create program
  procedure TForm1.FormCreate(Sender: TObject);
  begin
    matikanFungsional;
  end;

  //pilih mikrokontroller
  procedure TForm1.BtPilihMKClick(Sender: TObject);
  var
```

```

pilih,a,b,c,d,m,o,hasil,u,t : string;
begin
pilih := IntToStr(CmbPilihMK.ItemIndex+1);
t := chr(strtoint('$'+(pilih)));
a := chr(strtoint('$'+('00')));
b := '<';
c := chr(strtoint('$'+('81')));
d := '>';
m := '!';
hasil := a+a+a+a+a+b+c+d+t+m;
XComm1.SendString(hasil);
o:=chr(strtoint('$'+('AA')));
u:=m+o+m+o;
if (XComm1.WaitForString([u], 2000) <> -1) then
begin
  ShowMessage('Mikrokontroler telah terpilih');
  aktifkanFungsional;
end
else ShowMessage('Cek Mikrokontroler');

end;

//perintah hapus memori MK
procedure TForm1.BtHapusMemoriClick(Sender: TObject);
var
a,b,c,d,e,f,g,m,o,hasil,u : string;
begin
a := chr(strtoint('$'+('00')));
b := '<';
c := chr(strtoint('$'+('81')));
d := '>';

```

```

e:= inttostr(OPERASI_HAPUS_CHIP);
f:= chr(strtoint('$'+(e)));
g:=chr(strtoint('$'+('FF')));
m := '!';
hasil := a+a+a+a+a+b+c+d+f+m+g+g+m+g+g+m;
XComm1.SendString(hasil);
o:=chr(strtoint('$'+('AA')));
u:=m+o+m+o;
if (XComm1.WaitForString([u], 8000) <> -1) then
begin
  ShowMessage('Hapus Mikrokontroler sukses');
end
else ShowMessage('Cek Mikrokontroler');

end;

//Load File *hex
procedure TForm1.MnLoadFlashClick(Sender: TObject);
var
baris: string;
fileFlash: textfile;
kodeError: integer;
begin
if OpenFileDialogHex.Execute then
  DirektoriFile.Text:=OpenDialogHex.FileName;
panjangData:= 0;
MemoDataHex.Lines.Clear;
MemoDataASCII.Lines.Clear;
AssignFile(fileFlash, OpenFileDialogHex.FileName);
Reset(fileFlash);

```

```

isiHex:= "";
while not eof(fileFlash) do
begin
  Readln(fileFlash, baris);
  kodeError:= bacaBaris(baris, bufferHex);
  if kodeError = 0 then
  begin
    MemoDataHex.Lines.Add(DataHex);
    MemoDataASCII.Lines.Add(DataASCII);
    isiHex:= isiHex+DataHex;
  end
  else
  begin
    if kodeError = -1 then
      ShowMessage('Bukan intel hex!')
    else if kodeError = -2 then
      ShowMessage('Record type invalid!')
    else if kodeError = -3 then
      ShowMessage('Checksum error!');

    break;
  end;
end;
LabelPanjangData.Caption:= IntToStr(panjangData)+' bytes';
CloseFile(fileFlash);

BtReloadFlash.Enabled:= true;
BtReloadEEPROM.Enabled:= false;
end;

end;

```

```

//Load Fuse Bits
procedure TForm1.MnFuseBitsClick(Sender: TObject);
begin
  Form3.Show;
end;

//Exit Program
procedure TForm1.MnExitClick(Sender: TObject);
begin
  Close;
end;

//Load File *EEPROM
procedure TForm1.MnLoadEEPROMClick(Sender: TObject);
var
  baris: string;
  fileEEPROM: textfile;
  kodeError: integer;
begin
  if OpenFileDialogEEPROM.Execute then
  begin
    DirektoriFile.Text:=OpenDialogEEPROM.FileName;
    panjangData:= 0;
    MemoDataHex.Lines.Clear;
    MemoDataASCIILines.Clear;
    AssignFile(fileEEPROM, OpenFileDialogEEPROM.FileName);
    Reset(fileEEPROM);
    isiHex:= "";
    while not eof(fileEEPROM) do
    begin

```

```
Readln(fileEEPROM, baris);
kodeError:= bacaBaris(baris, bufferHex);
if kodeError = 0 then
begin
MemoDataHex.Lines.Add(DataHex);
isiHex:= isiHex+DataHex;
MemoDataASCIILines.Add(DataASCII)
end
else
begin
if kodeError = -1 then
ShowMessage('Bukan intel hex!')
else if kodeError = -2 then
ShowMessage('Record type invalid!')
else if kodeError = -3 then
ShowMessage('Checksum error!');

break;
end;
end;

LabelPanjangData.Caption:= IntToStr(panjangData)+' bytes';
CloseFile(fileEEPROM);

BtReloadEEPROM.Enabled:= true;
BtReloadFlash.Enabled:= false;
end;
end;

//Reload File *EEPROM
procedure TForm1.BtReloadEEPROMClick(Sender: TObject);
```

```
var
baris: string;
fileEEPROM: textfile;
kodeError: integer;
begin
DirektoriFile.Text:=OpenDialogEEPROM.FileName;
panjangData:= 0;
MemoDataHex.Lines.Clear;
MemoDataASCIILines.Clear;
AssignFile(fileEEPROM, OpenFileDialogEEPROM.FileName);
Reset(fileEEPROM);
isiHex:= "";
while not eof(fileEEPROM) do
begin
Readln(fileEEPROM, baris);
kodeError:= bacaBaris(baris, bufferHex);
if kodeError = 0 then
begin
MemoDataHex.Lines.Add(DataHex);
isiHex:= isiHex+DataHex;
MemoDataASCIILines.Add(DataASCII)
end
else
begin
if kodeError = -1 then
ShowMessage('Bukan intel hex!')
else if kodeError = -2 then
ShowMessage('Record type invalid!')
else if kodeError = -3 then
ShowMessage('Checksum error!');
```

```

break;
end;
end;

LabelPanjangData.Caption:= IntToStr(panjangData)+' bytes';
CloseFile(fileEEPROM);
end;

//Save file *hex
procedure TForm1.MnSaveFlashClick(Sender: TObject);
var
fileFlash: textfile;
barisData: string;
dataPerBaris: string;
alamatData: integer;
jmlBaris, jmlDataSisa, i, j: integer;
begin
if SaveDialogFlash.Execute then
begin
AssignFile(fileFlash, SaveDialogFlash.FileName);
Rewrite(fileFlash);
panjangData:= Length(isiHex) div 2;

jmlBaris:= panjangData div 16;
jmlDataSisa:= panjangData mod 16;

i:= 1;
j:= 0;
alamatData:= 0;
while j < jmlBaris do
begin

```

```

barisData:= Copy(isiHex, i, 32);
dataPerBaris:= tulisBaris(barisData, IntToHex(alamatData, 4));
WriteLn(fileFlash, dataPerBaris);
i:= i+32;
Inc(j);
alamatData:= alamatData+16;
end;

if jmlDataSisa <> 0 then
begin
barisData:= Copy(isiHex, i, jmlDataSisa*2);
dataPerBaris:= tulisBaris(barisData, IntToHex(alamatData, 4));
WriteLn(fileFlash, dataPerBaris);
end;
WriteLn(fileFlash, ':00000001FF');
CloseFile(fileFlash);
end;
end;

//Save file *EEPROM
procedure TForm1.MnSaveEEPROMClick(Sender: TObject);
var
fileEEPROM: textfile;
barisData: string;
dataPerBaris: string;
alamatData: integer;
jmlBaris, jmlDataSisa, i, j: integer;
begin
if SaveDialogEEPROM.Execute then

```

```

begin
AssignFile(fileEEPROM, SaveDialogEEPROM.FileName);
Rewrite(fileEEPROM);
panjangData:= Length(isiHex) div 2;

jmlBaris:= panjangData div 32;
jmlDataSisa:= panjangData mod 32;

i:= 1;
j:= 0;
alamatData:= 0;
while j < jmlBaris do
begin
barisData:= Copy(isiHex, i, 64);
dataPerBaris:= tulisBaris(barisData, IntToHex(alamatData, 4));
WriteLn(fileEEPROM, dataPerBaris);
i:= i+64;
Inc(j);
alamatData:= alamatData+32;
end;

if jmlDataSisa <> 0 then
begin
barisData:= Copy(isiHex, i, jmlDataSisa*2);
dataPerBaris:= tulisBaris(barisData, IntToHex(alamatData, 4));
WriteLn(fileEEPROM, dataPerBaris);
end;

WriteLn(fileEEPROM, '00000001FF');

CloseFile(fileEEPROM);

end;
end;
//bacaFlash
procedure TForm1.BtBacaFlashClick(Sender: TObject);
Var
buffer : TByteArr;
waktu: TDateTime;
count: integer;
d,data:string;
i2,i,j,k,a,l : integer;
baris,s: string;
fileFlash: textfile;
kodeError: integer;
sPaketData, sPaketPerintahTarget: String;
awal.header1_id_device,header2.id,idperintah,pembatas.perintah,nn :string;
device_signature,devicesig,devsignature,checksum,x,u,sukses:string;
b:byte;
t: string;
begin
//-----kirim paket perintah-----//
ReadyState := False;
Buffer_data := '';
waktu := now;
//header
awal := chr(strtoint('$'+('00')));
header1 := '<';
id_device := chr(strtoint('$'+('81')));
header2 := '>';
//id perintah

```



```

id:= inttostr(OPERASI_BACA_FLASH);
idperintah:= chr(strtoint('$'+(id)));
pembatas := '|';
nn := chr(strtoint('$'+('FF')));
//untuk jenis mk
buffer := StringToBytes(deviceSignature);
devicesig := BytesToString(buffer);
devsignature:="";
l:=1;
while l<=length(devicesig) do
begin
  devsignature:=devsignature+chr(strtoint('$'+copy(devicesig,1,2)));
  l:=l+3;
  If (l > 8) then
    break;
end;
device_signature:=devsignature;
perintah
:=
awal+header1+id_device+header2+idperintah+pembatas+nn+pembatas+device_signature+pembatas
+checksum;
XComm1.SendString(perintah);//paket perintah
while (Not ReadyState) and (SecondsBetween(waktu, Now) < 10)
do Application.ProcessMessages;
if (Pos('OK', Buffer_data) > 0) then
begin
  MemoDataHex.Lines.Append('OK');
  ReadyState := False;
  Buffer_data := "";
  waktu := now;
  while (Not ReadyState) and (SecondsBetween(waktu, Now) < 10)
do Application.ProcessMessages;

```

```

if (Pos('OK', Buffer_data) > 0) then
begin
  sPaketPerintahTarget := Buffer_data;
  MemoDataHex.Lines.Append(Buffer_data);
end
else begin
  ShowMessage('Baca Flash Gagal 0');
  exit;
end;
end
else
begin
  ShowMessage('Baca Flash Gagal 1');
  exit;
end;
end;
//-----Kirim konfirmasi-----//
ReadyState := False;
Buffer_data := "";
waktu := now;
sukses:=chr(strtoint('$'+('AA')));
u:=pembatas+sukses+pembatas+sukses;
x:=u+u;
XComm1.SendString(x);
while (Not ReadyState) and (SecondsBetween(waktu, Now) < 10)
do Application.ProcessMessages;
if (Pos('OK', Buffer_data) > 0) then
begin
  sPaketData := Buffer_data;
  MemoDataHex.Lines.Append(Buffer_data);
end

```

```

else
begin
  ShowMessage('Baca Flash Gagal 2');
  exit;
end;
XComm1.SendString(x); // kirim konfirmasi terakhir
ShowMessage('Baca Flash Berhasil');

end;

procedure TForm1.XComm1Data(Sender: TObject; const Received: Cardinal);
var
  bData: byte;
  Data: string;
  idata : integer;
begin
  XComm1.ReadString(data);
  Buffer_data := Buffer_data + data;
  if (not ReadyState) and (Pos('OK', Buffer_data) > 0) then
  begin
    ReadyState := True;
    MemoDataASCII.Lines.Append(Buffer_data);
  end;

end;

procedure TForm1.BitulisEEPROMClick(Sender: TObject);
var
  buffer : TByteArr;
  j,i,k,r : integer;
  sukses,Str,awal,header1,id_device,header2,id,idperintah,device_signature,devsignature,pembatas,panjang_data,panj_data,devicesig,akhir,head,hasil,sum,pd,u,x,checksum : string;
begin
  // untuk panjang data;
  pd:= IntToHex(panjangData,4);
  panj_data:= "";
  r:=1;
  while r<=length(pd) do
  begin
    panj_data:=panj_data+chr(strtoint('$'+copy(pd,r,2)));
    r:=r+2;
    If ( r > 4) then
      break;
    end;
    panjang_data:=panj_data;
    // untuk jenis mk
    buffer := StringToBytes(deviceSignature);
    devicesig := BytesToString(buffer);
    devsignature:= "";
    i:=1;
    while i<=length(devicesig) do
    begin
      devsignature:=devsignature+chr(strtoint('$'+copy(devicesig,i,2)));
      i:=i+3;
      If ( i > 8) then
        break;
      end;
      device_signature:=devsignature;
      // header
      awal := chr(strtoint('$'+('00')));

```

```

header1 := '<';
id_device := chr(strtoint('$'+('81')));
header2 := '>';
//id perintah
id:= inttostr(OPERASI_TULIS_EEPROM);
idperintah:= chr(strtoint('$'+(id)));
pembatas := '|';
hasil
:=
awal+awal+awal+awal+awal+header1+id_device+header2+idperintah+pembatas+panjang_data+pe
mbatas+device_signature+pembatas+akhir;
akhir := 'OK';
for j := 0 to (-1 + MemoDataHex.Lines.Count) do
  Str := Str + MemoDataHex.Lines[j];
  head:= '';
for k := 1 to length(Str) div 2 do
  head:= head + Char(StrToInt('$'+Copy(Str,(k-1)*2+1,2)));
  checksum:=checksum xor head;
  sum := header1+id_device+header2+idperintah+pembatas+head+checksum+akhir;
XComm1.SendString(hasil);
sukses:=chr(strtoint('$'+('AA')));
u:=pembatas+sukses+pembatas+sukses;
x:=u+u;
if (XComm1.WaitForString([u], 2000) <> -1) then
begin
  XComm1.SendString(sum);
  if (XComm1.WaitForString([x], 200000) <> -1) then
  begin
    ShowMessage('Pengiriman Selesai');
  end
  else ShowMessage('pengiriman gagal');
end
end

```

```

else ShowMessage('pengiriman perintah gagal');
end;
procedure TForm1.BtBacaEEPROMClick(Sender: TObject);
Var
  buffer : TByteArr;
  waktu : TDateTime;
  count: integer;
  d,data:string;
  i2,i,j,k,a,l : integer;
  baris,s: string;
  fileFlash: textfile;
  kodeError: integer;
  sPaketData, sPaketPerintahTarget: String;
  awal,header1,id_device,header2,id,idperintah,pembatas,perintah,nn :string;
  device_signature,devicesig,devsignature,checksum,x,u,sukses:string;
  b:byte;
  t: string;
begin
  //-----kirim paket perintah-----//
  ReadyState := False;
  Buffer_data := '';
  waktu := now;
  //header
  awal := chr(strtoint('$'+('00')));
  header1 := '<';
  id_device := chr(strtoint('$'+('81')));
  header2 := '>';
  //id perintah
  id:= inttostr(OPERASI_BACA_EEPROM);

```

```

idperintah:= chr(strtoint('$'+(id)));
pembatas := '|';
nn := chr(strtoint('$'+('FF')));
//untuk jenis mk
buffer := StringToBytes(deviceSignature);
devicesig := BytesToString(buffer);
devsignature:="";
l:=1;
while l<=length(devicesig) do
begin
  devsignature:=devsignature+chr(strtoint('$'+copy(devicesig,l,2)));
  l:=l+3;
  If (l > 8) then
    break;
end;
device_signature:=devsignature;
perintah
:=
awal+header1+id_device+header2+idperintah+pembatas+nn+pembatas+device_signature+pembatas
+checksum;
XComm1.SendString(perintah);//paket perintah
while (Not ReadyState) and (SecondsBetween(waktu, Now) < 10)
do Application.ProcessMessages;
if (Pos('OK', Buffer_data) > 0) then
begin
  MemoDataHex.Lines.Append('OK');
  ReadyState := False;
  Buffer_data := "";
  waktu := now;
  while (Not ReadyState) and (SecondsBetween(waktu, Now) < 10)
do Application.ProcessMessages;
if (Pos('OK', Buffer_data) > 0) then

```

```

begin
  sPaketPerintahTarget := Buffer_data;
  MemoDataHex.Lines.Append(Buffer_data);
end
else begin
  ShowMessage('Baca EEPROM Gagal');
  exit;
end;
end
else
begin
  ShowMessage('Baca EEPROM Gagal');
  exit;
end;
end;
//-----Kirim konfirmasi-----//
ReadyState := False;
Buffer_data := "";
waktu := now;
sukses:=chr(strtoint('$'+('AA')));
u:=pembatas+sukses+pembatas+sukses;
x:=u+u;
XComm1.SendString(x);
while (Not ReadyState) and (SecondsBetween(waktu, Now) < 10)
do Application.ProcessMessages;
if (Pos('OK', Buffer_data) > 0) then
begin
  sPaketData := Buffer_data;
  MemoDataHex.Lines.Append(Buffer_data);
end
else

```

```

begin
    ShowMessage('Baca EEPROM Gagal');
    exit;
end;
XComm1.SendString(x);//kirim konfirmasi terakhir
ShowMessage('Baca EEPROM Berhasil');

panjangData: integer;

function bacaBaris(HexLine: string; var Buf: buffer): integer;
function tulisBaris(HexLine: string; alamatData: string): string;

implementation

// Parsing file Intel Hex
function bacaBaris(HexLine: string; var Buf: buffer): integer;
var
    alamat, count: integer;
    ChkSum, JmlBaris, RecLen, RecType: byte;
    t: string;
begin
    // Atur panjang array, dapat disesuaikan sesuai keinginan
    SetLength(bufferHex, 100000);

    if HexLine[1] = ':' then
    begin
        t:= '$'+copy(HexLine, 2, 2);

        // Baca panjang
        RecLen:= StrToInt(t);
        ChkSum:= 0;
        ChkSum:= ChkSum+RecLen;
        t:= '$'+copy(HexLine, 4, 4);

        // Baca alamat
        alamat:= StrToInt(t);
        ChkSum:= ChkSum+lo(alamat)+hi(alamat);
        t:= '$'+copy(HexLine, 8, 2);
    end;
end.

```

PROGRAM INTEL HEX

```

unit intel_hex;

interface

uses
    Classes, SysUtils;

type
    buffer= packed array of byte;

var
    bufferHex: buffer;
    isiHex: string;
    Data: byte;
    DataHex: string;
    DataASCII: string;

```

```

RecType:= StrToInt(t);
ChkSum:= ChkSum+RecType;
DataHex:= "";
DataASCII:= "";

case RecType of
0:
begin
// Blok data
count:= 0;
while(count < RecLen) do
begin
t:= '$'+copy(HexLine, 10+2*count, 2);
DataASCII:= DataASCII+Chr(StrToInt(t));
DataHex:= DataHex+copy(HexLine, 10+2*count, 2);
Data:= StrToInt(t);
ChkSum:= ChkSum+Data;
Buf[alamat+count]:= Data;
Inc(count);
Inc(panjangData);
end;
t:= '$'+copy(HexLine, 10+2*count, 2);
JmlBaris:= StrToInt(t);
result:= 0;
end;
1:
begin
// End of file
t:= '$'+copy(HexLine, 10, 2);
JmlBaris:= StrToInt(t);
result:= 1;

end;
else
begin
// Record type salah
result:= -2;
exit;
end;

Data:= JmlBaris+ChkSum;
if Data <> 0 then
result:= -3
else
result:= -1;
end;
end;
end;

// Bangun file Intel Hex
function tulisBaris(HexLine: string; alamatData: string): string;
var
jmlDataPerBaris, i: integer;
isiPerBaris: string;
nilaiChecksum: byte;
begin
jmlDataPerBaris:= Length(HexLine) div 2;

isiPerBaris:= IntToHex(jmlDataPerBaris, 2)+alamatData+'00'+HexLine;
i:= 1;
nilaiChecksum:= 0;
while i <= Length(isiPerBaris) do
begin

```

```

nilaiChecksum:= nilaiChecksum+StrToInt('$'+Copy(isiPerBaris, i, 2));
i:= i+2;
end;
nilaiChecksum:= 256-nilaiChecksum;
result:= ':'+isiPerBaris+IntToHex(nilaiChecksum, 2);
end;

end.

```

PROGRAM KONEKSI

```

unit Unit2;

interface

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

type
  TForm2 = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    ComboBox1: TComboBox;
    ComboBox2: TComboBox;
    BitBtn1: TBitBtn;
  procedure BitBtn1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

```

```

var
  Form2: TForm2;

implementation

uses Unit1;
{$R *.dfm}

procedure TForm2.BitBtn1Click(Sender: TObject);
begin
  Form1.XComm1.BaudValue:= StrToInt(ComboBox1.Text);
  Form1.XComm1.DeviceName:= ComboBox2.Text;
  Close;
end;

end.

```

PROGRAM FUSEBITS

```

unit Unit3;

interface

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

type
  TForm3 = class(TForm)
    Panel1: TPanel;
    CheckBox1: TCheckBox;
    CheckBox2: TCheckBox;
    Label1: TLabel;
    Label2: TLabel;
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    CheckBox3: TCheckBox;
    CheckBox4: TCheckBox;
    CheckBox5: TCheckBox;
    CheckBox6: TCheckBox;

```

```

CheckBox7: TCheckBox;
CheckBox8: TCheckBox;
CheckBox9: TCheckBox;
CheckBox10: TCheckBox;
CheckBox11: TCheckBox;
CheckBox12: TCheckBox;
CheckBox13: TCheckBox;
CheckBox14: TCheckBox;
CheckBox15: TCheckBox;
CheckBox16: TCheckBox;
CheckBox17: TCheckBox;
CheckBox18: TCheckBox;
CheckBox19: TCheckBox;
CheckBox20: TCheckBox;
CheckBox21: TCheckBox;
CheckBox22: TCheckBox;
CheckBox23: TCheckBox;
CheckBox24: TCheckBox;
CheckBox25: TCheckBox;
CheckBox26: TCheckBox;
procedure FormShow(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form3: TForm3;
  jenisMK: string;
implementation

uses Unit1,Unit2;
{$R *.dfm}

function Pow(i, k: Integer): Integer;
var
  j, Count: Integer;
begin
  if k>0 then j:=2
  else j:=1;
  for Count:=1 to k-1 do

```

```

j:=j*2;
  Result:=j;
end;

function BinToDec(Str: string): Integer;
var
  Len, Res, i: Integer;
  Error: Boolean;
begin
  Error:=False;
  Len:=Length(Str);
  Res:=0;
  for i:=1 to Len do
    if (Str[i]='0')or(Str[i]='1') then
      Res:=Res+Pow(2, Len-i)*StrToInt(Str[i])
    else
      begin
        MessageDlg('It is not a binary number', mtInformation, [mbOK], 0);
        Error:=True;
        Break;
      end;
    if Error=True then Result:=0
    else Result:=Res;
  end;

function HexToDec(Str: string): Integer;
var
  i, M: Integer;
begin
  Result:=0;
  M:=1;
  Str:=AnsiUpperCase(Str);
  for i:=Length(Str) downto 1 do
    begin
      case Str[i] of
        '1'..'9': Result:=Result+(Ord(Str[i])-Ord('0'))*M;
        'A'..'F': Result:=Result+(Ord(Str[i])-Ord('A')+10)*M;
      end;
      M:=M shl 4;
    end;
  end;

function DecToBin(N: Integer): string;
var

```



```

S: string;
i: Integer;
Negative: Boolean;
begin
if N<0 then Negative:=True;
N:=Abs(N);
for i:=1 to SizeOf(N)*8 do
begin
if N<0 then S:=S+'1'
else S:=S+'0';
N:=N shl 1;
end;
Delete(S,1,Pos('1',S)-1);
if Negative then S:=S;
Result:=S;
end;

function HexToBin(Hexadecimal: string): string;
const
BCD: array [0..15] of string =
('0000', '0001', '0010', '0011', '0100', '0101', '0110', '0111',
'1000', '1001', '1010', '1011', '1100', '1101', '1110', '1111');
var
i: integer;
begin
for i:= Length(Hexadecimal) downto 1 do
Result := BCD[StrToInt('$' + Hexadecimal[i])] + Result;
end;

procedure TForm3.FormShow(Sender: TObject);
begin
Label2.Caption:= 'Jenis mikrokontroler: '+tipeAVR;
jenisMK:= tipeAVR;
//label2.Caption:=jenisMK;
if (jenisMK = 'AT90S2333') or (jenisMK = 'AT90S4433') then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;

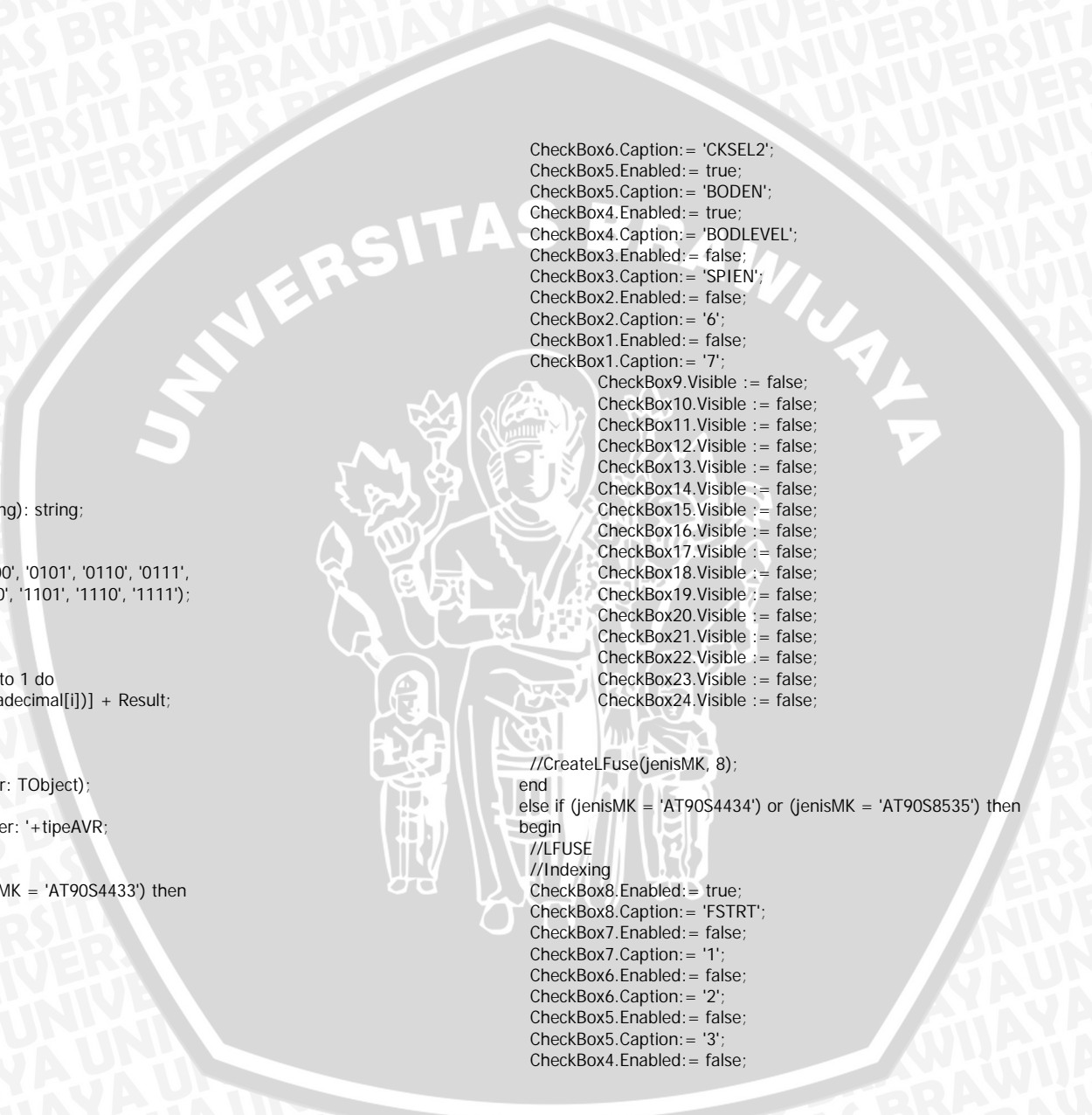
```

```

CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'BODEN';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'BODLEVEL';
CheckBox3.Enabled:= false;
CheckBox3.Caption:= 'SPIEN';
CheckBox2.Enabled:= false;
CheckBox2.Caption:= '6';
CheckBox1.Enabled:= false;
CheckBox1.Caption:= '7';
CheckBox9.Visible := false;
CheckBox10.Visible := false;
CheckBox11.Visible := false;
CheckBox12.Visible := false;
CheckBox13.Visible := false;
CheckBox14.Visible := false;
CheckBox15.Visible := false;
CheckBox16.Visible := false;
CheckBox17.Visible := false;
CheckBox18.Visible := false;
CheckBox19.Visible := false;
CheckBox20.Visible := false;
CheckBox21.Visible := false;
CheckBox22.Visible := false;
CheckBox23.Visible := false;
CheckBox24.Visible := false;

//CreateLFuse(jenisMK, 8);
end
else if (jenisMK = 'AT90S4434') or (jenisMK = 'AT90S8535') then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'FSTRT';
CheckBox7.Enabled:= false;
CheckBox7.Caption:= '1';
CheckBox6.Enabled:= false;
CheckBox6.Caption:= '2';
CheckBox5.Enabled:= false;
CheckBox5.Caption:= '3';
CheckBox4.Enabled:= false;

```



```

CheckBox4.Caption:= '4';
CheckBox3.Enabled:= false;
CheckBox3.Caption:= '5';
CheckBox2.Enabled:= false;
CheckBox2.Caption:= '6';
CheckBox1.Enabled:= false;
CheckBox1.Caption:= '7';
    CheckBox9.Visible := false;
    CheckBox10.Visible := false;
    CheckBox11.Visible := false;
    CheckBox12.Visible := false;
    CheckBox13.Visible := false;
    CheckBox14.Visible := false;
    CheckBox15.Visible := false;
    CheckBox16.Visible := false;
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

```

```

//CreateLFuse(jenisMK, 8);
end
else if jenisMK = 'AT90S2343' then
begin
    //LFUSE
    //Indexing
    CheckBox8.Enabled:= true;
    CheckBox8.Caption:= 'RCEN';
    CheckBox7.Enabled:= false;
    CheckBox7.Caption:= '1';
    CheckBox6.Enabled:= false;
    CheckBox6.Caption:= '2';
    CheckBox5.Enabled:= false;
    CheckBox5.Caption:= '3';
    CheckBox4.Enabled:= false;
    CheckBox4.Caption:= '4';
    CheckBox3.Enabled:= false;
    CheckBox3.Caption:= '5';
    CheckBox2.Enabled:= false;
    CheckBox2.Caption:= '6';

```

```

CheckBox1.Enabled:= false;
CheckBox1.Caption:= '7';
    CheckBox9.Visible := false;
    CheckBox10.Visible := false;
    CheckBox11.Visible := false;
    CheckBox12.Visible := false;
    CheckBox13.Visible := false;
    CheckBox14.Visible := false;
    CheckBox15.Visible := false;
    CheckBox16.Visible := false;
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

```

```

//CreateLFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega8' then
begin
    //LFUSE
    //Indexing
    CheckBox8.Enabled:= true;
    CheckBox8.Caption:= 'CKSEL0';
    CheckBox7.Enabled:= true;
    CheckBox7.Caption:= 'CKSEL1';
    CheckBox6.Enabled:= true;
    CheckBox6.Caption:= 'CKSEL2';
    CheckBox5.Enabled:= true;
    CheckBox5.Caption:= 'CKSEL3';
    CheckBox4.Enabled:= true;
    CheckBox4.Caption:= 'SUT0';
    CheckBox3.Enabled:= true;
    CheckBox3.Caption:= 'SUT1';
    CheckBox2.Enabled:= true;
    CheckBox2.Caption:= 'BODEN';
    CheckBox1.Enabled:= true;
    CheckBox1.Caption:= 'BODLEVEL';

```

```

//CreateLFuse(jenisMK, 8);

```



```

//HFUSE
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZO';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'CKOPT';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'WDTON';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= 'RSTDISBL';
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

// CreateHFuse(jenisMK, 8);
end
else if (jenisMK = 'ATmega16') or (jenisMK = 'ATmega32') then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSELO';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUTO';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';

```

```

CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'BODEN';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'BODLEVEL';

```

```
//CreateLFuse(jenisMK, 8);
```

```

//HFUSE
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZO';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'CKOPT';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'JTAGEN';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= 'OCDEN';

```

```
//CreateHFuse(jenisMK, 8);
```

```

CheckBox17.Visible := false;
CheckBox18.Visible := false;
CheckBox19.Visible := false;
CheckBox20.Visible := false;
CheckBox21.Visible := false;
CheckBox22.Visible := false;
CheckBox23.Visible := false;
CheckBox24.Visible := false;

```

```

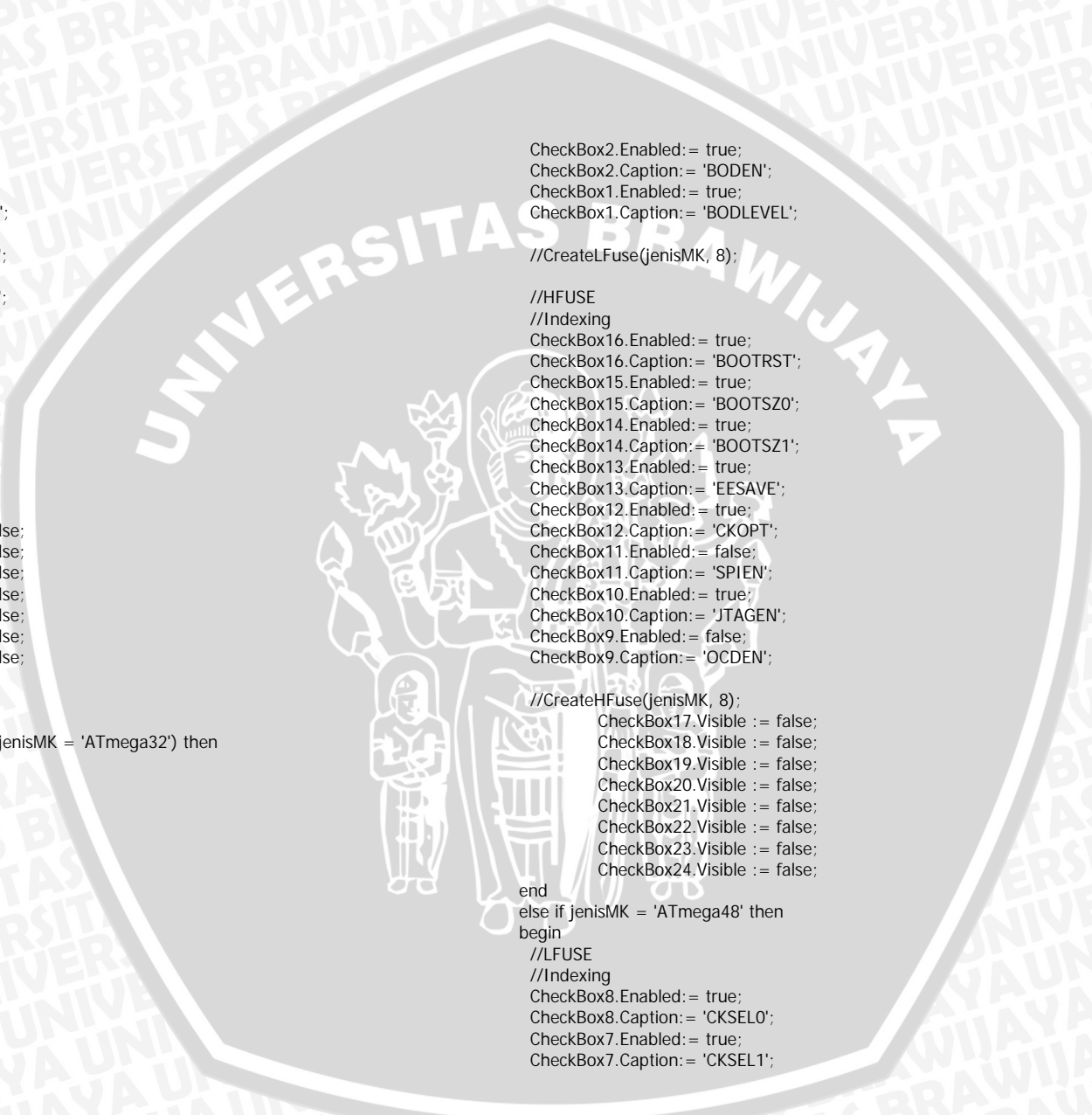
end
else if jenisMK = 'ATmega48' then
begin

```

```

//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSELO';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';

```



```

CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUTO';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'CKOUT';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'CKDIV8';

```

```
//CreateLFuse(jenisMK, 8);
```

```
//HFUSE
```

```
//Indexing
```

```

CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BODLEVEL0';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BODLEVEL1';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BODLEVEL2';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'WDTON';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'DWEN';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= 'RSTDISBL';

```

```
//CreateHFuse(jenisMK, 8);
```

```
//EFUSE
```

```
//Indexing
```

```

CheckBox24.Enabled:= true;
CheckBox24.Caption:= 'SELFPRGEN';
CheckBox23.Enabled:= false;
CheckBox23.Caption:= '1';
CheckBox22.Enabled:= false;
CheckBox22.Caption:= '2';

```

```

CheckBox21.Enabled:= false;
CheckBox21.Caption:= '3';
CheckBox20.Enabled:= false;
CheckBox20.Caption:= '4';
CheckBox19.Enabled:= false;
CheckBox19.Caption:= '5';
CheckBox18.Enabled:= false;
CheckBox18.Caption:= '6';
CheckBox17.Enabled:= false;
CheckBox17.Caption:= '7';

```

```
//CreateEFuse(jenisMK, 8);
```

```
end
```

```
else if (jenisMK = 'ATmega64') or (jenisMK = 'ATmega128') then
```

```
begin
```

```
//LFUSE
```

```
//Indexing
```

```

CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUTO';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'BODEN';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'BODLEVEL';

```

```
//CreateLFuse(jenisMK, 8);
```

```
//HFUSE
```

```
//Indexing
```

```

CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZ0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';

```

```

CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'CKOPT';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'JTAGEN';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= 'OCDEN';

```

```
//CreateHFuse(jenisMK, 8);
```

```
//EFUSE
```

```
//Indexing
```

```

CheckBox24.Enabled:= true;
CheckBox24.Caption:= 'WDTON';
CheckBox23.Enabled:= true;
CheckBox23.Caption:= 'M103C';
CheckBox22.Enabled:= false;
CheckBox22.Caption:= '2';
CheckBox21.Enabled:= false;
CheckBox21.Caption:= '3';
CheckBox20.Enabled:= false;
CheckBox20.Caption:= '4';
CheckBox19.Enabled:= false;
CheckBox19.Caption:= '5';
CheckBox18.Enabled:= false;
CheckBox18.Caption:= '6';
CheckBox17.Enabled:= false;
CheckBox17.Caption:= '7';

```

```
//CreateEFuse(jenisMK, 8);
```

```
end
```

```
else if jenisMK = 'ATmega88' then
```

```
begin
```

```
//LFUSE
```

```
//Indexing
```

```

CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSELO';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';

```

```

CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUT0';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'CKOUT';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'CKDIV8';

```

```
//CreateLFuse(jenisMK, 8);
```

```
//HFUSE
```

```
//Indexing
```

```

CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BODLEVEL0';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BODLEVEL1';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BODLEVEL2';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'WDTON';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'DWEN';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= 'RSTDISBL';

```

```
//CreateHFuse(jenisMK, 8);
```

```
//EFUSE
```

```
//Indexing
```

```

CheckBox24.Enabled:= true;
CheckBox24.Caption:= 'BOOTRST';
CheckBox23.Enabled:= true;
CheckBox23.Caption:= 'BOOTSZ0';
CheckBox22.Enabled:= true;
CheckBox22.Caption:= 'BOOTSZ1';
CheckBox21.Enabled:= false;
CheckBox21.Caption:= '3';

```

```

CheckBox20.Enabled := false;
CheckBox20.Caption := '4';
CheckBox19.Enabled := false;
CheckBox19.Caption := '5';
CheckBox18.Enabled := false;
CheckBox18.Caption := '6';
CheckBox17.Enabled := false;
CheckBox17.Caption := '7';

//CreateEFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega103' then
begin
//LFUSE
//Indexing
CheckBox8.Enabled := true;
CheckBox8.Caption := 'SUT0';
CheckBox7.Enabled := true;
CheckBox7.Caption := 'SUT1';
CheckBox6.Enabled := false;
CheckBox6.Caption := '2';
CheckBox5.Enabled := true;
CheckBox5.Caption := 'EESAVE';
CheckBox4.Enabled := false;
CheckBox4.Caption := '4';
CheckBox3.Enabled := false;
CheckBox3.Caption := '5';
CheckBox2.Enabled := false;
CheckBox2.Caption := '6';
CheckBox1.Enabled := false;
CheckBox1.Caption := '7';
CheckBox9.Visible := false;
CheckBox10.Visible := false;
CheckBox11.Visible := false;
CheckBox12.Visible := false;
CheckBox13.Visible := false;
CheckBox14.Visible := false;
CheckBox15.Visible := false;
CheckBox16.Visible := false;
CheckBox17.Visible := false;
CheckBox18.Visible := false;
CheckBox19.Visible := false;
CheckBox20.Visible := false;
CheckBox21.Visible := false;

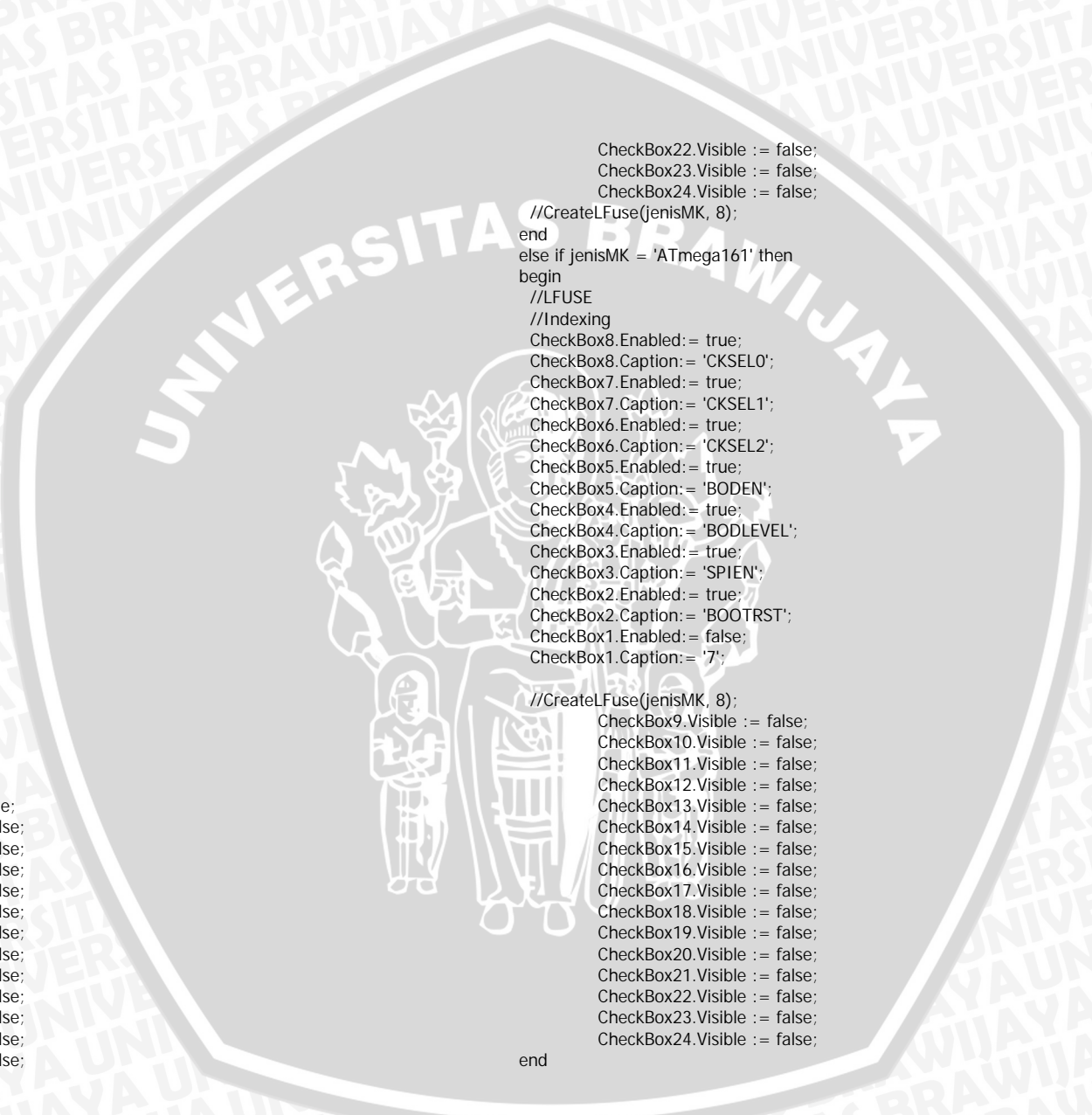
CheckBox22.Visible := false;
CheckBox23.Visible := false;
CheckBox24.Visible := false;

//CreateLFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega161' then
begin
//LFUSE
//Indexing
CheckBox8.Enabled := true;
CheckBox8.Caption := 'CKSELO';
CheckBox7.Enabled := true;
CheckBox7.Caption := 'CKSEL1';
CheckBox6.Enabled := true;
CheckBox6.Caption := 'CKSEL2';
CheckBox5.Enabled := true;
CheckBox5.Caption := 'BODEN';
CheckBox4.Enabled := true;
CheckBox4.Caption := 'BODLEVEL';
CheckBox3.Enabled := true;
CheckBox3.Caption := 'SPIEN';
CheckBox2.Enabled := true;
CheckBox2.Caption := 'BOOTRST';
CheckBox1.Enabled := false;
CheckBox1.Caption := '7';

//CreateLFuse(jenisMK, 8);
CheckBox9.Visible := false;
CheckBox10.Visible := false;
CheckBox11.Visible := false;
CheckBox12.Visible := false;
CheckBox13.Visible := false;
CheckBox14.Visible := false;
CheckBox15.Visible := false;
CheckBox16.Visible := false;
CheckBox17.Visible := false;
CheckBox18.Visible := false;
CheckBox19.Visible := false;
CheckBox20.Visible := false;
CheckBox21.Visible := false;
CheckBox22.Visible := false;
CheckBox23.Visible := false;
CheckBox24.Visible := false;

end

```



```
else if jenisMK = 'ATmega162' then
begin
```

```
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUT0';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'CKOUT';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'CKDIV8';
```

```
//CreateLFuse(jenisMK, 8);
```

```
//HFUSE
//Indexing
```

```
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZ0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'WDTON';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'JTAGEN';
CheckBox9.Enabled:= true;
CheckBox9.Caption:= 'OCDEN';
```

```
//CreateHFuse(jenisMK, 8);
```

```
//EFUSE
```

```
//Indexing
CheckBox24.Enabled:= false;
CheckBox24.Caption:= '0';
CheckBox23.Enabled:= true;
CheckBox23.Caption:= 'BODLEVEL';
CheckBox22.Enabled:= true;
CheckBox22.Caption:= 'BOD1LEVEL';
CheckBox21.Enabled:= true;
CheckBox21.Caption:= 'BOD2LEVEL';
CheckBox20.Enabled:= true;
CheckBox20.Caption:= 'M161C';
CheckBox19.Enabled:= false;
CheckBox19.Caption:= '5';
CheckBox18.Enabled:= false;
CheckBox18.Caption:= '6';
CheckBox17.Enabled:= false;
CheckBox17.Caption:= '7';
```

```
//CreateEFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega163' then
begin
```

```
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= false;
CheckBox4.Caption:= '4';
CheckBox3.Enabled:= false;
CheckBox3.Caption:= '5';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'BODEN';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'BODLEVEL';
```

```
//CreateLFuse(jenisMK, 8);
```



```
//HFUSE
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZ0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= false;
CheckBox13.Caption:= '3';
CheckBox12.Enabled:= false;
CheckBox12.Caption:= '4';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= '5';
CheckBox10.Enabled:= false;
CheckBox10.Caption:= '6';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= '7';
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;
```

```
//CreateHFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega169' then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUTO';
CheckBox3.Enabled:= true;
```

```
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'CKOUT';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'CKDIV8';
```

```
//CreateLFuse(jenisMK, 8);
```

```
//HFUSE
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZ0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'WDTON';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'JTAGEN';
CheckBox9.Enabled:= true;
CheckBox9.Caption:= 'OCDEN';
```

```
//CreateHFuse(jenisMK, 8);
```

```
//EFUSE
//Indexing
CheckBox24.Enabled:= false;
CheckBox24.Caption:= 'RSTDISBL';
CheckBox23.Enabled:= true;
CheckBox23.Caption:= 'BODLEVEL';
CheckBox22.Enabled:= true;
CheckBox22.Caption:= 'BOD1LEVEL';
CheckBox21.Enabled:= true;
CheckBox21.Caption:= 'BOD2LEVEL';
CheckBox20.Enabled:= false;
CheckBox20.Caption:= '4';
CheckBox19.Enabled:= false;
CheckBox19.Caption:= '5';
CheckBox18.Enabled:= false;
```



```

CheckBox18.Caption:= '6';
CheckBox17.Enabled:= false;
CheckBox17.Caption:= '7';

// CreateEFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega8515' then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUT0';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'BODEN';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'BODLEVEL';

//CreateLFuse(jenisMK, 8);

//HFUSE
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZ0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'CKOPT';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;

```

```

CheckBox10.Caption:= 'WDTON';
CheckBox9.Enabled:= true;
CheckBox9.Caption:= 'S8515C';
CheckBox17.Visible := false;
CheckBox18.Visible := false;
CheckBox19.Visible := false;
CheckBox20.Visible := false;
CheckBox21.Visible := false;
CheckBox22.Visible := false;
CheckBox23.Visible := false;
CheckBox24.Visible := false;

// CreateHFuse(jenisMK, 8);
end
else if jenisMK = 'ATmega8535' then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUT0';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'BODEN';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'BODLEVEL';

//CreateLFuse(jenisMK, 8);

//HFUSE
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BOOTRST';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BOOTSZ0';
CheckBox14.Enabled:= true;

```



```

CheckBox14.Caption:= 'BOOTSZ1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'EESAVE';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'CKOPT';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'WDTON';
CheckBox9.Enabled:= true;
CheckBox9.Caption:= 'S8535C';
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

// CreateHFuse(jenisMK, 8);
end
else if jenisMK = 'ATtiny12' then
begin
    //LFUSE
    //Indexing
    CheckBox8.Enabled:= true;
    CheckBox8.Caption:= 'CKSELO';
    CheckBox7.Enabled:= true;
    CheckBox7.Caption:= 'CKSEL1';
    CheckBox6.Enabled:= true;
    CheckBox6.Caption:= 'CKSEL2';
    CheckBox5.Enabled:= true;
    CheckBox5.Caption:= 'CKSEL3';
    CheckBox4.Enabled:= false;
    CheckBox4.Caption:= 'RSTDISBL';
    CheckBox3.Enabled:= false;
    CheckBox3.Caption:= 'SPIEN';
    CheckBox2.Enabled:= true;
    CheckBox2.Caption:= 'BODEN';
    CheckBox1.Enabled:= true;
    CheckBox1.Caption:= 'BODLEVEL';
        CheckBox9.Visible := false;
        CheckBox10.Visible := false;

    CheckBox11.Visible := false;
    CheckBox12.Visible := false;
    CheckBox13.Visible := false;
    CheckBox14.Visible := false;
    CheckBox15.Visible := false;
    CheckBox16.Visible := false;
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

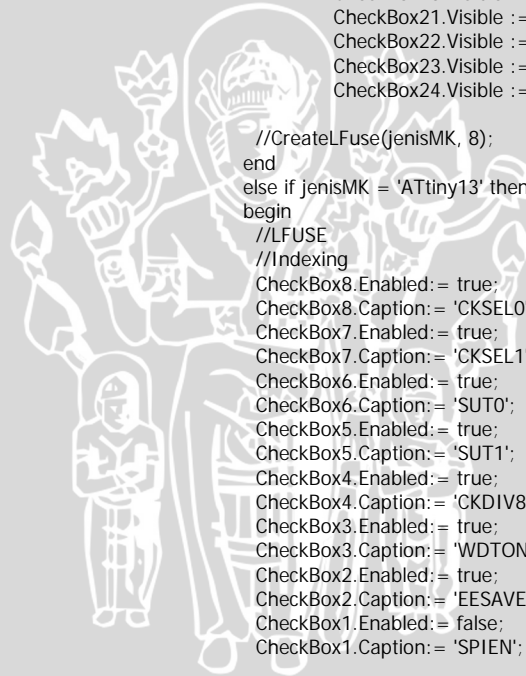
    //CreateLFuse(jenisMK, 8);
end
else if jenisMK = 'ATtiny13' then
begin
    //LFUSE
    //Indexing
    CheckBox8.Enabled:= true;
    CheckBox8.Caption:= 'CKSELO';
    CheckBox7.Enabled:= true;
    CheckBox7.Caption:= 'CKSEL1';
    CheckBox6.Enabled:= true;
    CheckBox6.Caption:= 'SUT0';
    CheckBox5.Enabled:= true;
    CheckBox5.Caption:= 'SUT1';
    CheckBox4.Enabled:= true;
    CheckBox4.Caption:= 'CKDIV8';
    CheckBox3.Enabled:= true;
    CheckBox3.Caption:= 'WDTON';
    CheckBox2.Enabled:= true;
    CheckBox2.Caption:= 'EESAVE';
    CheckBox1.Enabled:= false;
    CheckBox1.Caption:= 'SPIEN';

    //CreateLFuse(jenisMK, 8);

    //HFUSE
    //Indexing
    CheckBox16.Enabled:= false;
    CheckBox16.Caption:= 'RSTDISBL';

```

UNIVERSITAS



JAWA

```

CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BODLEVEL0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BODLEVEL1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'DWEN';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'SELFPRGEN';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= '5';
CheckBox10.Enabled:= false;
CheckBox10.Caption:= '6';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= '7';
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

```

```

//CreateHFuse(jenisMK, 8);
end
else if jenisMK = 'ATTiny15' then
begin
    //LFUSE
    //Indexing
    CheckBox8.Enabled:= true;
    CheckBox8.Caption:= 'CKSELO';
    CheckBox7.Enabled:= true;
    CheckBox7.Caption:= 'CKSEL1';
    CheckBox6.Enabled:= false;
    CheckBox6.Caption:= '2';
    CheckBox5.Enabled:= false;
    CheckBox5.Caption:= '3';
    CheckBox4.Enabled:= false;
    CheckBox4.Caption:= 'RSTDISBL';
    CheckBox3.Enabled:= false;
    CheckBox3.Caption:= 'SPIEN';
    CheckBox2.Enabled:= true;
    CheckBox2.Caption:= 'BODEN';
    CheckBox1.Enabled:= true;

```

```

CheckBox1.Caption:= 'BODLEVEL';
    CheckBox9.Visible := false;
    CheckBox10.Visible := false;
    CheckBox11.Visible := false;
    CheckBox12.Visible := false;
    CheckBox13.Visible := false;
    CheckBox14.Visible := false;
    CheckBox15.Visible := false;
    CheckBox16.Visible := false;
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;

```

```

    // CreateLFuse(jenisMK, 8);
    end
    else if jenisMK = 'ATTiny26' then
    begin
        //LFUSE
        //Indexing
        CheckBox8.Enabled:= true;
        CheckBox8.Caption:= 'CKSELO';
        CheckBox7.Enabled:= true;
        CheckBox7.Caption:= 'CKSEL1';
        CheckBox6.Enabled:= true;
        CheckBox6.Caption:= 'CKSEL2';
        CheckBox5.Enabled:= true;
        CheckBox5.Caption:= 'CKSEL3';
        CheckBox4.Enabled:= true;
        CheckBox4.Caption:= 'SUT0';
        CheckBox3.Enabled:= true;
        CheckBox3.Caption:= 'SUT1';
        CheckBox2.Enabled:= true;
        CheckBox2.Caption:= 'CKOPT';
        CheckBox1.Enabled:= true;
        CheckBox1.Caption:= 'PLLCK';

```

```

// CreateLFuse(jenisMK, 8);

```

```

//HFUSE

```



UNIVERSITAS BRAWIJAYA

```
//Indexing
CheckBox16.Enabled:= true;
CheckBox16.Caption:= 'BODEN';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BODLEVEL';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'EESAVE';
CheckBox13.Enabled:= false;
CheckBox13.Caption:= 'SPIEN';
CheckBox12.Enabled:= false;
CheckBox12.Caption:= 'RSTDISBL';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= '5';
CheckBox10.Enabled:= false;
CheckBox10.Caption:= '6';
CheckBox9.Enabled:= false;
CheckBox9.Caption:= '7';
    CheckBox17.Visible := false;
    CheckBox18.Visible := false;
    CheckBox19.Visible := false;
    CheckBox20.Visible := false;
    CheckBox21.Visible := false;
    CheckBox22.Visible := false;
    CheckBox23.Visible := false;
    CheckBox24.Visible := false;
```

```
// CreateHFuse(jenisMK, 8);
end
else if jenisMK = 'ATtiny2313' then
begin
//LFUSE
//Indexing
CheckBox8.Enabled:= true;
CheckBox8.Caption:= 'CKSEL0';
CheckBox7.Enabled:= true;
CheckBox7.Caption:= 'CKSEL1';
CheckBox6.Enabled:= true;
CheckBox6.Caption:= 'CKSEL2';
CheckBox5.Enabled:= true;
CheckBox5.Caption:= 'CKSEL3';
CheckBox4.Enabled:= true;
CheckBox4.Caption:= 'SUT0';
CheckBox3.Enabled:= true;
CheckBox3.Caption:= 'SUT1';
```

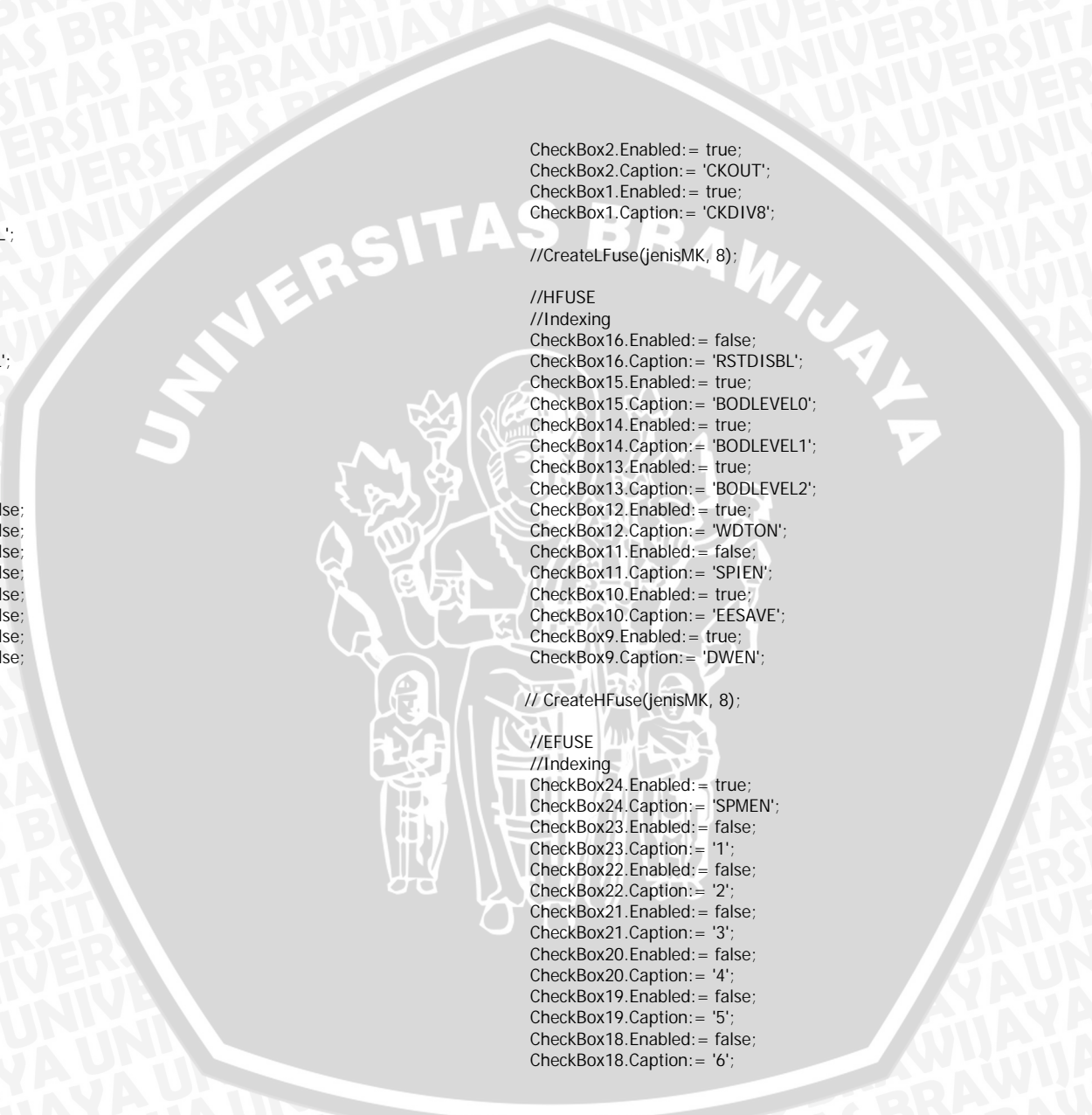
```
CheckBox2.Enabled:= true;
CheckBox2.Caption:= 'CKOUT';
CheckBox1.Enabled:= true;
CheckBox1.Caption:= 'CKDIV8';
```

```
//CreateLFuse(jenisMK, 8);
```

```
//HFUSE
//Indexing
CheckBox16.Enabled:= false;
CheckBox16.Caption:= 'RSTDISBL';
CheckBox15.Enabled:= true;
CheckBox15.Caption:= 'BODLEVEL0';
CheckBox14.Enabled:= true;
CheckBox14.Caption:= 'BODLEVEL1';
CheckBox13.Enabled:= true;
CheckBox13.Caption:= 'BODLEVEL2';
CheckBox12.Enabled:= true;
CheckBox12.Caption:= 'WDTON';
CheckBox11.Enabled:= false;
CheckBox11.Caption:= 'SPIEN';
CheckBox10.Enabled:= true;
CheckBox10.Caption:= 'EESAVE';
CheckBox9.Enabled:= true;
CheckBox9.Caption:= 'DWEN';
```

```
// CreateHFuse(jenisMK, 8);
```

```
//EFUSE
//Indexing
CheckBox24.Enabled:= true;
CheckBox24.Caption:= 'SPMEN';
CheckBox23.Enabled:= false;
CheckBox23.Caption:= '1';
CheckBox22.Enabled:= false;
CheckBox22.Caption:= '2';
CheckBox21.Enabled:= false;
CheckBox21.Caption:= '3';
CheckBox20.Enabled:= false;
CheckBox20.Caption:= '4';
CheckBox19.Enabled:= false;
CheckBox19.Caption:= '5';
CheckBox18.Enabled:= false;
CheckBox18.Caption:= '6';
```



```

CheckBox17.Enabled := false;
CheckBox17.Caption := '7';

```

```

//CreateEFuse(jenisMK, 8);
end;
end;

```

```

procedure TForm3.Button1Click(Sender: TObject);

```

```

var
i:integer;
kode : array [1..24] of string;
checkbox : array [1..24] of TCheckBox;
b,d,f: integer;
lowfuse,a,c,e,highfuse,extendfuse,sukses,pembatas: string;
lfuse,hfuse,efuse,k,l,m,fuse,p,q,r,fusebits,x,u :string;
begin

```

```

i:=1;
for i:=1 to high(checkbox) do
begin
if Tcheckbox(FindComponent('Checkbox'+IntToStr(i))).Checked = true then kode[i]:='0'
else kode[i]:='1';
fuse := fuse+kode[i];
lfuse := copy(fuse,1,8);
hfuse := copy(fuse,9,8);
efuse := copy(fuse,17,8);
if lfuse<>" then
k:=IntToStr(BinToDec(lfuse));
a:=IntToStr(BinToDec(hfuse));
b:=strtoint(a);
lowfuse:= IntToHex(b,2);
p:=lowfuse;
if hfuse<>" then
l:=IntToStr(BinToDec(hfuse));
c:=IntToStr(BinToDec(hfuse));
d:=strtoint(c);
highfuse:= IntToHex(d,2);
q:=highfuse;
if efuse<>" then
m:=IntToStr(BinToDec(efuse));
e:=IntToStr(BinToDec(efuse));
f:=strtoint(e);
extendfuse:= IntToHex(f,2);
r:=extendfuse;

```

```

fusebits := p+q+r;
xcomm1.sendstring(fusebits);
sukses:=chr(strtoint('$'+('AA')));
pembatas := '|';
u:=pembatas+sukses+pembatas+sukses;
x:=u+u;
if (XComm1.WaitForString([u], 2000) <> -1) then
begin
ShowMessage('Pengiriman Selesai')
end;
else ShowMessage('pengiriman gagal');
end;
end;

```

```

procedure TForm3.Button2Click(Sender: TObject);

```

```

var
i,j,k:integer;
a,c,e : integer;
lfuse,hfuse,efuse,lowfuse,highfuse,extendfuse,fuse,fusebits,data,data1,data2 : string;
kode : array [1..24] of string;
begin
xcomm1.readstring(data);
fusebits:=data;
data1:= copy(fusebits,1,2);
if data1 = data2 then
lowfuse:= copy(fusebits,4,2);
highfuse:= copy(fusebits,6,2);
extendfuse:= copy(fusebits,8,2);
lfuse:= HexToBin(lowfuse);
hfuse:= HexToBin(highfuse);
efuse:= HexToBin(extendfuse);
fuse:= lfuse+hfuse+efuse;
i:=1;
j:=1;
for i:=1 to length(fuse) do
begin
kode[i] := copy(fuse,i,j);
if kode[i]='0' then
Tcheckbox(FindComponent('Checkbox'+IntToStr(i))).Checked := true
else
Tcheckbox(FindComponent('Checkbox'+IntToStr(i))).Checked := false;
end;
ShowMessage('Pengiriman Selesai')
if data <> data1 then

```

```
begin  
  ShowMessage('pengiriman gagal');  
end;  
end.
```



LAMPIRAN III

DATASHEET

