

LAMPIRAN A
LISTING INSTRUKSI

```

unit Unit1;

interface

uses
  Math, Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Grids, DBGrids, Mask;

type
  TForm1 = class(TForm)
    Label5: TLabel;
    Label7: TLabel;
    Panel1: TPanel;
    Label23: TLabel;
    Label24: TLabel;
    Label25: TLabel;
    RegB: TEdit;
    RegA: TEdit;
    Panel2: TPanel;
    Label11: TLabel;
    Label12: TLabel;
    Label26: TLabel;
    RegY: TEdit;
    RegX: TEdit;
    Panel3: TPanel;
    Label13: TLabel;
    Label14: TLabel;
    Label16: TLabel;
    Label19: TLabel;
    Label27: TLabel;
    Label28: TLabel;
    Label29: TLabel;
    Label30: TLabel;
    Label31: TLabel;
    RegC: TEdit;
    RegS: TEdit;
    RegV: TEdit;
    RegH: TEdit;
    RegXIRQ: TEdit;
    RegIRQ: TEdit;
    RegZ: TEdit;
    RegN: TEdit;
    Eksekusi: TButton;
    Perintah: TMemo;
    Tutup: TButton;
    table1: TTable;
    DataSource1: TDataSource;
    Memory: TMemo;
    Port: TLabel;
    LabelA: TLabel;
    LabelB: TLabel;
    LabelC: TLabel;
    LabelD: TLabel;
    PortA: TEdit;
    PortB: TEdit;
    PortC: TEdit;
    PortD: TEdit;
    Mnemonic: TMemo;
    Label18: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    ProgramCounter: TEdit;
    StackPointer: TEdit;
    KondisiAwal: TButton;
    Panel4: TPanel;
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    analog: TEdit;
    digital: TEdit;
    procedure TutupClick(Sender: TObject);
    procedure EksekusiClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  end;

```

```

    procedure FormCreate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure KondisiAwalClick(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;
    buffint,buffint1,buffint2,buffint3,buffint4,buffint5:integer;
    buffstr,buffstr1,buffstr2,buffstr3,buffstr4,buffstr5:string;
    buffchar,buffchar1,buffchar2:char;

implementation

{$R *.DFM}

function huruf(input:string):string;           // dibuat menjadi huruf besar
semuanya
var
    i,pjg:integer; x:char; hrf:string;
begin
    hrf:='';
    pjg:=length(input);           // kondisi awal
    for i:=pjg downto 1 do
        begin
            x:=upcase(input[i]);   // dibesarkan dulu
            hrf:=x+hrf;
        end;
    huruf:=hrf;
end;

procedure AmbilStringPerintah(strinput:string ; var depan,tengah,belakang:string);
var
    x,spasi,koma:integer;
begin
    x:=length(strinput);           // panjang karakter
    spazi:=pos(' ',strinput);      // mencari spazi
    koma:=pos(',',strinput);        // mencari koma
    if spazi=0 then                 // hanya ada op-code
        depan:=huruf(copy(strinput,1,x))
    else if koma=0 then             // ada op-code dan 1 operand
        begin
            depan:=huruf(copy(strinput,1,spasi-1));
            tengah:=huruf(copy(strinput,spasi+1,x));
        end
    else                           // ada op-code dan 2 operand
        begin
            depan:=huruf(copy(strinput,1,spasi-1));
            tengah:=huruf(copy(strinput,spasi+1,koma-1-spasi));
            belakang:=huruf(copy(strinput,koma+1,x));
        end;
    end;
end;

procedure AmbilStringMnemonic(strinput:string ; var depan,tengah,belakang:string);
var
    x,spasi1,spasi2,spasi3:integer;
begin
    x:=length(strinput);           // panjang karakter
    spazi1:=pos(' ',strinput);     // mencari spazi pertama
    buffstr:=copy(strinput,spasi1+1,x);
    spazi2:=pos(' ',buffstr);      // mencari spazi kedua
    if spazi1=0 then               // hanya ada op-code
        depan:=huruf(copy(strinput,1,x))
    else if spazi2=0 then          // ada op-code dan 1 operand
        begin
            depan:=huruf(copy(strinput,1,spasi1-1));
            tengah:=huruf(copy(strinput,spasi1+1,x));
        end
    else if spazi3=0 then          // ada op-code dan 2 operand
        begin

```

```

        depan:=huruf(copy(strinput,1,spasi1-1));
        tengah:=huruf(copy(strinput,spasi1+1,spasi2-1-spasi1));
        belakang:=huruf(copy(strinput,spasi2+1,x));
    end;
end;

procedure bintohe(x:string; var y:string);
begin
    if x='0000' then y:='0';
    if x='0001' then y:='1';
    if x='0010' then y:='2';
    if x='0011' then y:='3';
    if x='0100' then y:='4';
    if x='0101' then y:='5';
    if x='0110' then y:='6';
    if x='0111' then y:='7';
    if x='1000' then y:='8';
    if x='1001' then y:='9';
    if x='1010' then y:='A';
    if x='1011' then y:='B';
    if x='1100' then y:='C';
    if x='1101' then y:='D';
    if x='1110' then y:='E';
    if x='1111' then y:='F';
end;

function hextobin(x:string):string;
var
    hexa:char;
begin
    hexa:=x[1];
    case upcase(hexa) of
        '0': hextobin:='0000';
        '1': hextobin:='0001';
        '2': hextobin:='0010';
        '3': hextobin:='0011';
        '4': hextobin:='0100';
        '5': hextobin:='0101';
        '6': hextobin:='0110';
        '7': hextobin:='0111';
        '8': hextobin:='1000';
        '9': hextobin:='1001';
        'A': hextobin:='1010';
        'B': hextobin:='1011';
        'C': hextobin:='1100';
        'D': hextobin:='1101';
        'E': hextobin:='1110';
        'F': hextobin:='1111';
    end;
end;

procedure bintodes(input:string ; var output:string);
var
    bit:string;
    pjg,x,i:integer;
    hasilnya,pangkat:real;
begin
    pjg:=length(input);
    x:=0; hasilnya:=0;
    for i:=pjg downto 1 do
        begin
            bit:=copy(input,i,1);
            if (bit='1') or (bit='0') then
                begin
                    pangkat:=power(2,x);
                    hasilnya:=hasilnya+(strtoint(bit)*pangkat);
                    x:=x+1
                end
            else
                begin
                    showmessage('salah memasukan data');
                    exit;
                end
            end;
        end;
    end;
end;

```

```

        end;
    end;
    output:=floattostr(hasilnya);
end;

procedure destobin(input:string ; var output:string; var e:char);
var
    y:string;
    pjg,i,a,b,c,x:integer;
begin
    // b:=ord('0'); c:=ord('9');           // 0=48           9=57
    a:=strtoint(input);
    y:='';
    repeat
        x:= a mod 2;
        y:=inttostr(x)+y;
        a:=floor((a-x)/2);
    until a<=0;
    pjg:=length(y);
    if pjg=8 then output:=y
    else if pjg<8 then
        begin
            pjg:=length(y);
            for i:=pjg to 7 do
                begin
                    y:='0'+y;
                end;
            output:=y;
        end
    else if pjg>8 then
        begin
            if pjg=16 then output:=y
            else
                begin
                    pjg:=length(y);
                    for i:=pjg to 15 do
                        begin
                            y:='0'+y;
                        end;
                    output:=y;
                end;
            end
        end
    else
        begin
            showmessage('salah memasukan data');e:='1';
        end;
end;

function dtbase(Opcode,mode:string):string;           // program untuk mengambil bahasa
mesin dari list program
begin
    with form1 do
        begin
            if table1.FindKey([Opcode]) then
                dtbase:=table1[mode];
            end;
        end;
end;

procedure bittambah(x,y:char; var hasil,carry:char);
begin           // penjumlahan bit
    if (x='0') and (y='0') then
        begin
            hasil:='0';carry:='0';
        end;
    if (x='1') and (y='0') then
        begin
            hasil:='1';carry:='0';
        end;
    if (x='0') and (y='1') then
        begin
            hasil:='1';carry:='0';
        end;
    if (x='1') and (y='1') then

```

```

        begin
            hasil:='0';carry:='1';
        end;
    end;

    procedure bitkurang(x,y:char; var hasil,carry:char);
    begin
        // pengurangan bit
        if (x='0') and (y='0') then
            begin
                hasil:='0';carry:='0';
            end;
        if (x='1') and (y='0') then
            begin
                hasil:='1';carry:='0';
            end;
        if (x='0') and (y='1') then
            begin
                hasil:='1';carry:='1';
            end;
        if (x='1') and (y='1') then
            begin
                hasil:='0';carry:='0';
            end;
        end;
    end;

    function f_and(x,y:char):char;
    begin
        if (x='1') and (y='1') then
            f_and:='1'
        else
            f_and:='0';
        end;
    end;

    function f_or(x,y:char):char;
    begin
        if (x='0') and (y='0') then
            f_or:='0'
        else
            f_or:='1';
        end;
    end;

    function angka(input:string):integer;
    var
        pjg:integer;
        a1,a2,a3,a4,x:string;
    begin
        pjg:=length(input); if pjg=2 then input:='00'+input else;
        a1:=copy(input,1,1); a2:=copy(input,2,1); a3:=copy(input,3,1);
        a4:=copy(input,4,1);
        if a1='0' then
            begin
                // awal memory di-set
                if a2='0' then
                    begin
                        // angka ratusannya 0
                        if a3='0' then begin
                            // angka puluhannya 0
                            buffstr:=hextobin(a4); bintodes(buffstr,buffstr);
                            angka:=strtoint(buffstr) end
                        else
                            begin
                                buffstr1:=hextobin(a3); buffstr2:=hextobin(a4);
                                buffstr:=buffstr1+buffstr2; bintodes(buffstr,x);
                                angka:=strtoint(x);
                            end;
                        end
                    end
                else
                    begin
                        if pjg=4 then
                            begin
                                buffstr1:=hextobin(a2);buffstr2:=hextobin(a3);buffstr3:=hextobin(a4);
                                buffstr:=buffstr1+buffstr2+buffstr3; bintodes(buffstr,x);
                                angka:=strtoint(x);
                            end
                        else
                            begin

```

```

        buffstr:=hextobin(a2);bintodes(buffstr,x);angka:=strtoint(x);
    end;
end;
else
begin
    buffstr1:=hextobin(a1); buffstr2:=hextobin(a2);
    buffstr3:=hextobin(a3); buffstr4:=hextobin(a4);
    buffstr:=buffstr1+buffstr2+buffstr3+buffstr4; bintodes(buffstr,x);
    angka:=strtoint(x);
end
end;

function jadiin4bit(inputbiner:string):string;
var pjg:integer;
begin
    pjg:=length(inputbiner);
    repeat
        inputbiner:='0'+inputbiner;
        inc(pjg);
    until pjg=4;
    jadiin4bit:=inputbiner;
end;

function jadiin8bit(inputbiner:string):string;
var pjg:integer;
begin
    pjg:=length(inputbiner);
    repeat
        inputbiner:='0'+inputbiner;
        inc(pjg);
    until pjg=8;
    jadiin8bit:=inputbiner;
end;

function jadiin16bit(inputbiner:string):string;
var pjg:integer;
begin
    pjg:=length(inputbiner);
    repeat
        inputbiner:='0'+inputbiner;
        inc(pjg);
    until pjg=16;
    jadiin16bit:=inputbiner;
end;

procedure geserkanan(opcode,RA,RB,RC:string; var hasil, bhs_msn:string; var
carry,z,e:char);
var pjg,i:integer;
begin
    buffstr:=copy(opcode,4,1);
    if buffstr='A' then
        begin
            bhs_msn:=dtbase('ASRA','INH');
            buffstr1:=copy(RA,1,1); buffstr2:=copy(RA,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end
    else if buffstr='B' then
        begin
            bhs_msn:=dtbase('ASRB','INH');
            buffstr1:=copy(RB,1,1); buffstr2:=copy(RB,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end;
    carry:=buffstr[1];
    pjg:=length(buffstr);
    for i:=pjg downto 2 do
        begin
            if buffstr[i]>'1' then
                begin
                    ShowMessage('Data yang dimasukkan salah');e:='1';
                end
            end;
        end;
    end;
end;

```

```

        else
            buffstr[i]:=buffstr[i-1];
        end;
        buffstr[pjg]:='0';
        buffstr1:=copy(buffstr,1,4); buffstr2:=copy(buffstr,5,4);
        buffstr3:=copy(buffstr,9,4); buffstr4:=copy(buffstr,13,4);
        bintohehex(buffstr1,buffstr1); bintohehex(buffstr2,buffstr2);
        bintohehex(buffstr3,buffstr3); bintohehex(buffstr4,buffstr4);
        buffstr:=buffstr1+buffstr2+buffstr3+buffstr4;
        hasil:=buffstr;
        if hasil='00' then z:='1' else z:='0';
    end;
end;

```

```

procedure geserkiri(opcode,RA,RB,RC:string; var hasil, bhs_msn:string; var
carry,e:char);
var pjg,i:integer;
begin
    buffstr:=copy(opcode,4,1);
    if buffstr='A' then
        begin
            bhs_msn:=dtbase('ASLA','INH');
            buffstr1:=copy(RA,1,1); buffstr2:=copy(RA,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end
    else if buffstr='B' then
        begin
            bhs_msn:=dtbase('ASLB','INH');
            buffstr1:=copy(RB,1,1); buffstr2:=copy(RB,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end
    else
        begin
            bhs_msn:=dtbase('ASLD','INH');
            buffstr1:=copy(RA,1,1); buffstr2:=copy(RA,2,1); buffstr3:=copy(RB,1,1);
            buffstr4:=copy(RB,2,1);
            buffstr1:=hextobin(buffstr1); buffstr2:=hextobin(buffstr2);
            buffstr3:=hextobin(buffstr3); buffstr4:=hextobin(buffstr4);
            buffstr:=buffstr1+buffstr2+buffstr3+buffstr4;
        end;
        carry:=buffstr[1];
        pjg:=length(buffstr);
        for i:=1 to pjg do
            begin
                if buffstr[i]>'1' then
                    begin
                        ShowMessage('Data yang dimasukkan salah');e:='1';
                    end
                else
                    buffstr[i]:=buffstr[i+1];
                end;
            end;
        buffstr[pjg]:='0';
        buffstr1:=copy(buffstr,1,4); buffstr2:=copy(buffstr,5,4);
        buffstr3:=copy(buffstr,9,4); buffstr4:=copy(buffstr,13,4);
        bintohehex(buffstr1,buffstr1); bintohehex(buffstr2,buffstr2);
        bintohehex(buffstr3,buffstr3); bintohehex(buffstr4,buffstr4);
        buffstr:=buffstr1+buffstr2+buffstr3+buffstr4;
        hasil:=buffstr;
    end;
end;

```

```

procedure puterkanan(opcode,RA,RB:string; var RC:char; var hasil, bhs_msn:string;
var carry,z,e:char);
var pjg,i:integer;
begin
    buffstr:=copy(opcode,4,1);
    if buffstr='A' then
        begin
            bhs_msn:=dtbase('RORA','INH');
            buffstr1:=copy(RA,1,1); buffstr2:=copy(RA,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end;
    end;
end;

```

```

        end
    else if buffstr='B' then
        begin
            bhs_msn:=dtbase('RORB','INH');
            buffstr1:=copy(RB,1,1); buffstr2:=copy(RB,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end;
        carry:=buffstr[1];
        pjg:=length(buffstr);
        for i:=pjg downto 2 do
            begin
                if buffstr[i]>'1' then
                    begin
                        ShowMessage('Data yang dimasukkan salah');e:='1';
                    end
                else
                    buffstr[i]:=buffstr[i-1];
                end;
            buffstr[pjg]:=RC;
            buffstr1:=copy(buffstr,1,4); buffstr2:=copy(buffstr,5,4);
            buffstr3:=copy(buffstr,9,4); buffstr4:=copy(buffstr,13,4);
            bintohehex(buffstr1,buffstr1); bintohehex(buffstr2,buffstr2);
            bintohehex(buffstr3,buffstr3); bintohehex(buffstr4,buffstr4);
            buffstr:=buffstr1+buffstr2+buffstr3+buffstr4;
            hasil:=buffstr;
            if hasil='00' then z:='1' else z:='0';
        end;

procedure puterkiri(opcode,RA,RB:string; var RC:char; var hasil, bhs_msn:string;
var carry,e:char;
var pjg,i:integer;
begin
    buffstr:=copy(opcode,4,1);
    if buffstr='A' then
        begin
            bhs_msn:=dtbase('ROLA','INH');
            buffstr1:=copy(RA,1,1); buffstr2:=copy(RA,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end
    else if buffstr='B' then
        begin
            bhs_msn:=dtbase('ROLB','INH');
            buffstr1:=copy(RB,1,1); buffstr2:=copy(RB,2,1);
            buffstr3:=hextobin(buffstr1); buffstr4:=hextobin(buffstr2);
            buffstr:=buffstr3+buffstr4;
        end;
        carry:=buffstr[1];
        pjg:=length(buffstr);
        for i:=1 to pjg do
            begin
                if buffstr[i]>'1' then
                    begin
                        ShowMessage('Data yang dimasukkan salah');e:='1';
                    end
                else
                    buffstr[i]:=buffstr[i+1];
                end;
            buffstr[pjg]:=RC;
            buffstr1:=copy(buffstr,1,4); buffstr2:=copy(buffstr,5,4);
            buffstr3:=copy(buffstr,9,4); buffstr4:=copy(buffstr,13,4);
            bintohehex(buffstr1,buffstr1); bintohehex(buffstr2,buffstr2);
            bintohehex(buffstr3,buffstr3); bintohehex(buffstr4,buffstr4);
            buffstr:=buffstr1+buffstr2+buffstr3+buffstr4;
            hasil:=buffstr;
        end;

procedure tambah4bit(x,y:string; var hasil:string; var z,carry:char;
var
    i,pjgx,pjgy:integer;

```

```

        q,q2,c2:char;
        x1,y1,jawab:string;
begin
    // merubah bilangan heksadesimal ke dlm btk biner
    x1:=hextobin(x);y1:=hextobin(y);
    // pengecekan panjang data
    pjgx:=length(x1);
    pjgy:=length(y1);
    if (pjgx<>4) or (pjgy<>4) then
        begin
            showmessage('salah memasukan data');
            exit;
        end;

    // looping multi bit
    jawab:='';
    carry:='0';
    c2:='0';
    for i:=pjgx downto 1 do
        begin
            bittambah(x1[i],c2,q2,c2);           // menjumlahkan bit LSB dgn carry
            bittambah(q2,y1[i],q,carry);         // menjumlahkan dgn bit berikutnya
            bittambah(carry,c2,carry,c2);        // menjumlahkan carry-nya
            c2:=carry;
            jawab:=q+jawab;                       // menyusun hasil jawabannya
        end;
        bintohej(jawab,hasil);
        if hasil='0' then z:='1'
        else z:='0';
    end;

procedure tambah8bit(x,y:string; var hasil:string; var z,carry,H:char);
var
    i,pjgx,pjgy:integer;
    q,q2,c2:char;
    hasil1,hasil2,hsl1,hsl2:string;
    a1,a2,xt,b1,b2,yt,jawab:string;
begin
    // merubah bilangan heksadesimal ke dlm btk biner
    a1:=hextobin(copy(x,1,1));a2:=hextobin(copy(x,2,1));
    xt:=a1+a2;
    b1:=hextobin(copy(y,1,1));b2:=hextobin(copy(y,2,1));
    yt:=b1+b2;
    // pengecekan panjang data
    pjgx:=length(xt);
    pjgy:=length(yt);
    if (pjgx<>8) or (pjgy<>8) then
        begin
            showmessage('salah memasukan data');
            exit;
        end;
    // looping multi bit (8 bit)
    jawab:='';
    carry:='0';
    c2:='0';
    for i:=pjgx downto 1 do
        begin
            bittambah(xt[i],c2,q2,c2);           // menjumlahkan bit LSB dgn carry
            bittambah(q2,yt[i],q,carry);         // menjumlahkan dgn bit berikutnya
            bittambah(carry,c2,carry,c2);        // menjumlahkan carry-nya
            if (i=5) then H:=carry;
            c2:=carry;
            jawab:=q+jawab;                       // menyusun hasil jawabannya
        end;
        hsl1:=copy(jawab,1,4);hsl2:=copy(jawab,5,8);
        bintohej(hsl1,hasil1);bintohej(hsl2,hasil2);
        hasil:=hasil1+hasil2;
        if hasil='00' then z:='1'
        else z:='0';
    end;

procedure kurang8bit(x,y:string; var hasil:string; var z,e:char);
var

```

```

        a,b,c,pjgx,pjgy:integer;
        hasil1,hasil2,hsl1,hsl2:string;
        a1,a2,xt,b1,b2,yt,jawab:string;
begin
    // merubah bilangan heksadesimal ke dlm btk biner
    a1:=hextobin(copy(x,1,1));a2:=hextobin(copy(x,2,1));
    xt:=a1+a2;
    b1:=hextobin(copy(y,1,1));b2:=hextobin(copy(y,2,1));
    yt:=b1+b2;
    // pengecekan panjang data
    pjgx:=length(xt);
    pjgy:=length(yt);
    if (pjgx<>8) or (pjgy<>8) then
        begin
            showmessage('salah memasukan data');E:='1';
            exit;
        end;
    // looping multi bit (8 bit)
    bintodes(xt,xt); a:=strtoint(xt); bintodes(yt,yt); b:=strtoint(yt);
    c:=a-b;
    jawab:=inttostr(c);
    destobin(jawab,jawab,e);

//COBA DULU      INGAT!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
{ buffchar1:='0'; jawab:='';
  for i:=pjgx downto 1 do
    begin
      (if (xt[i]='0') and (yt[i]='1') then
        begin
          ShowMessage('Pengurangan Overload'); e:='1';
        end;
      bitkurang(buffchar1,xt[i],buffchar2,buffchar1);
      bitkurang(buffchar2,yt[i],buffchar2,buffchar1);
      jawab:=buffchar2+jawab;
    end;
  }

  hsl1:=copy(jawab,1,4);hsl2:=copy(jawab,5,8);
  bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
  hasil:=hasil1+hasil2;
  if hasil='00' then
    begin
      z:='1'; hasil:='00';
    end
  else z:='0';
end;

procedure p_and4bit(x,y:string; var hasil:string; var z:char);
var
    i,pjgx,pjgy:integer;
    xt,yt,jawab:string;
    q:char;
begin
    // merubah bilangan heksadesimal ke dlm btk biner
    xt:=hextobin(x);yt:=hextobin(y);
    // pengecekan panjang data
    pjgx:=length(xt);
    pjgy:=length(yt);
    if (pjgx<>4) or (pjgy<>4) then
        begin
            showmessage('salah memasukan data');
            exit;
        end;
    // looping multi bit
    jawab:='';
    for i:=pjgx downto 1 do
        begin
            q:=f_and(xt[i],yt[i]);
            jawab:=q+jawab;
        end;
    bintohehex(jawab,hasil);
    if hasil='0' then z:='1' else z:='0';
end;

```

```

procedure p_and8bit(x,y:string; var hasil:string; var z:char);
var
    i,pjgx,pjgy:integer;
    a1,a2,xt,b1,b2,yt,jawab:string;
    hasil1,hasil2,hs11,hs12:string;
    q:char;
begin
    // merubah bilangan heksadesimal ke dlm btk biner
    a1:=hextobin(copy(x,1,1));a2:=hextobin(copy(x,2,1));
    b1:=hextobin(copy(y,1,1));b2:=hextobin(copy(y,2,1));
    xt:=a1+a2;yt:=b1+b2;
    // pengecekan panjang data
    pjgx:=length(xt);
    pjgy:=length(yt);
    if (pjgx<>8) or (pjgy<>8) then
        begin
            showmessage('salah memasukan data');
            exit;
        end;
    // looping multi bit
    jawab:='';
    for i:=pjgx downto 1 do
        begin
            q:=f_and(xt[i],yt[i]);
            jawab:=q+jawab;
            // menyusun hasil jawabannya
        end;
    hs11:=copy(jawab,1,4);hs12:=copy(jawab,5,8);
    bintohehex(hs11,hasil1);bintohehex(hs12,hasil2);
    hasil:=hasil1+hasil2;
    if hasil='00' then z:='1' else z:='0';
end;

procedure ABA(RA,RB,Operand:string; var hasil:string; var z,c,H:char; var
bhs_msn:string);
begin
    tambah8bit(RA,RB,hasil,z,c,H);
    bhs_msn:=dtbase('ABA','INH');
end;

procedure ADDD(RA,RB,RX,RY,operand1,operand2:string; var
highorder,loworder:string; var z,c,H,e:char; var bhs_msn:string);
var
    imm,bin,bil,w,x,y,t,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil1,hasil2,hasil3,hasil4,hs11,hs12,hs13,hs14:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1); e:='0';

            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then
                        // input desimal
                        begin
                            imm:=copy(operand1,2,pjg);
                            destobin(imm,bin,e);
                            pjg_bin:=length(bin);
                            if pjg_bin>16 then
                                begin
                                    ShowMessage('Register D Hanya 16 Bit');e:='1';
                                end
                            else
                                ;
                        end;
                    ;
                end;
            ;
            hs11:=copy(bin,1,4);hs12:=copy(bin,5,4);hs13:=copy(bin,9,4);hs14:=copy(bin,13,4);

            bintohehex(hs11,hasil1);bintohehex(hs12,hasil2);bintohehex(hs13,hasil3);bintohehex(hs14,ha
sil4);

            w:=hasil1+hasil2;x:=hasil3+hasil4;
            y:=RA;t:=RB;

```

```

        tambah8bit(t,x,loworder,z,c,H);buffstr:='0'+c;
        tambah8bit(buffstr,w,buffstr1,z,c,H);
        tambah8bit(buffstr1,y,highorder,z,c,H);
        bhs_msn:=dtbase('ADDD','IMM');
        bhs_msn:=bhs_msn+' '+w+x;
    end
else if (dolar<>0) then // input heksa
    begin
        imm:=copy(operand1,3,pjg);
        pjg_hex:=length(imm);
        if pjg_hex<4 then
            begin
                ShowMessage('Perintah ADDD Membutuhkan Operand 16
Bit');e:='1';
            end
        else if pjg_hex=4 then
            begin
                hsl1:=copy(imm,1,2);hsl2:=copy(imm,3,2);
                tambah8bit(RB,hsl2,loworder,z,c,H);buffstr:='0'+c;
                tambah8bit(hsl1,buffstr,buffstr1,z,c,H);
                tambah8bit(RA,buffstr1,highorder,z,c,H);
                bhs_msn:=dtbase('ADDD','IMM');
                bhs_msn:=bhs_msn+' '+hsl1+hsl2;;
            end
        else begin ShowMessage('Operand Pada Instruksi ADDD Tidak Boleh
Lebih Dari 16 Bit');e:='1'; end;
    end
else if (persen<>0) then // input biner
    begin
        imm:=copy(operand1,3,pjg); // mengambil biner
        pjg_bin:=length(imm); // panjang biner
        if pjg_bin<16 then
            begin
                ShowMessage('Perintah ADDD Membutuhkan Operand 16
Bit');e:='1';
            end
        else if pjg_bin=16 then
            begin

hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,4);hsl3:=copy(imm,9,4);hsl4:=copy(imm,13,4);

bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);bintohehex(hsl3,hasil3);bintohehex(hsl4,ha
sil4);

                buffstr1:=hasil1+hasil2;buffstr2:=hasil3+hasil4;
                bhs_msn:=dtbase('ADDD','IMM');
                bhs_msn:=bhs_msn+' '+buffstr1+buffstr2;
                tambah8bit(RB,buffstr2,loworder,z,c,H);buffstr:='0'+c;
                tambah8bit(buffstr1,buffstr,buffstr3,z,c,H);
                tambah8bit(RA,buffstr3,highorder,z,c,H);
            end;
        end;
    end

// PENGALAMATAN DIRECT & EXTENDED
else if (pagar=0) and (dolar<>0) and (operand2='') then
    begin
        // PENGALAMATAN DIRECT
        if pjg=3 then
            begin
                // mengambil 2 digit terakhir operand utk alamat data
                x:=copy(operand1,2,2);
                alamat:=angka(x);
                hsl1:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                hsl2:=copy(Memory.Lines[alamat+1],14,2); // mengambil data
yg ada dialamat tsb+1
                tambah8bit(RB,hsl2,loworder,z,c,H);buffstr:='0'+c;
                tambah8bit(hsl1,buffstr,buffstr1,z,c,H);
                tambah8bit(RA,buffstr1,highorder,z,c,H);
                bhs_msn:=dtbase('ADDD','DIR');
                bhs_msn:=bhs_msn+' '+hsl1+hsl2;
            end
        end
    end

```

```

// PENGALAMATAN EXTENDED
else if pjpg=5 then
begin
// mengambil 4 digit terakhir operand utk alamat data
x:=copy(operand1,2,4);
alamat:=angka(x);
hsl1:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
hsl2:=copy(Memory.Lines[alamat+1],14,2); // mengambil data
yg ada dialamat tsb+1
tambah8bit(RB,hsl2,loworder,z,c,H);buffstr:='0'+c;
tambah8bit(hsl1,buffstr,buffstr1,z,c,H);
tambah8bit(RA,buffstr1,highorder,z,c,H);
bhs_msn:=dtbase('ADDD','EXT');
bhs_msn:=bhs_msn+' '+hsl1+hsl2;
end;
end

// PENGALAMATAN INDEXED
else if (pagar=0) and (dolar<>0) and (operand2<>'') then
begin
if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
bil:=copy(operand1,2,2);
else if (persen<>0) and (dolar=0) then // input bilangan biner
begin
pjpg_bin:=length(operand1);
if pjpg_bin>16 then ShowMessage('Pengalamatan Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
bintohehex(copy(operand1,2,4),bil);
end;
op2:=huruf(copy(operand2,1,1));
if op2='X' then
begin
buffint:=strtoint(RX)+strtoint(bil); // alamatnya = data
di Reg X + bil
x:=inttostr(buffint);
alamat:=angka(x);
hsl1:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
hsl2:=copy(Memory.Lines[alamat+1],14,2); // mengambil data
yg ada dialamat tsb+1
tambah8bit(RB,hsl2,loworder,z,c,H);buffstr:='0'+c;
tambah8bit(hsl1,buffstr,buffstr1,z,c,H);
tambah8bit(RA,buffstr1,highorder,z,c,H);
bhs_msn:=dtbase('ADDD','IND X');
bhs_msn:=bhs_msn+' '+hsl1+hsl2;
end
else if op2='Y' then
begin
buffint:=strtoint(RY)+strtoint(bil); // alamatnya = data
di Reg X + bil
x:=inttostr(buffint);
alamat:=angka(x);
hsl1:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
hsl2:=copy(Memory.Lines[alamat+1],14,2); // mengambil data
yg ada dialamat tsb+1
tambah8bit(RB,hsl2,loworder,z,c,H);buffstr:='0'+c;
tambah8bit(hsl1,buffstr,buffstr1,z,c,H);
tambah8bit(RA,buffstr1,highorder,z,c,H);
bhs_msn:=dtbase('ADDD','IND Y');
bhs_msn:=bhs_msn+' '+hsl1+hsl2;
end
else begin ShowMessage('Tidak Ada Register Index Tersebut');e:='1';
end;
end;
end;
end;

procedure ADDA(RA,RX,RY,operand1,operand2:string; var hasil:string; var
z,c,H,e:char; var bhs_msn:string);
var

```

```

        imm,bin,dir,bil,x,op2:string;
        pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
        hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1); e:='0';

            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then                // input desimal
                        begin
                            imm:=copy(operand1,2,pjg);
                            destobin(imm,bin,e);
                            pjg_bin:=length(bin);
                            if pjg_bin>8 then
                                begin
                                    showmessage('Register A hanya 8 bit');e:='1';
                                end
                            else
                                begin
                                    hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
                                    bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                    hsl:=hasil1+hasil2;
                                    bhs_msn:=dtbase('ADDA','IMM');
                                    bhs_msn:=bhs_msn+' '+hsl;
                                    tambah8bit(RA,hsl,hasil,z,c,H);
                                end;
                            end
                        else if (dolar<>0) then                // input heksa
                            begin
                                imm:=copy(operand1,3,2);
                                bhs_msn:=dtbase('ADDA','IMM');
                                bhs_msn:=bhs_msn+' '+imm;
                                pjg_hex:=length(imm);
                                if pjg_hex<>2 then
                                    begin
                                        showmessage('Register A hanya 8 bit');e:='1';
                                    end
                                else
                                    begin
                                        tambah8bit(RA,imm,hasil,z,c,H);
                                    end;
                                end
                            else if (persen<>0) then                // input biner
                                begin
                                    imm:=copy(operand1,3,8);                // mengambil biner
                                    pjg_bin:=length(imm);                // panjang biner
                                    if pjg_bin<>8 then
                                        begin
                                            showmessage('Register A hanya 8 bit');e:='1';
                                        end
                                    else
                                        begin
                                            hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                                            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                            hsl:=hasil1+hasil2;
                                            bhs_msn:=dtbase('ADDA','IMM');
                                            bhs_msn:=bhs_msn+' '+hsl;
                                            tambah8bit(RA,hsl,hasil,z,c,H);
                                        end;
                                    end;
                                end;
                            end
                        else if (pagar=0) and (dolar<>0) and (operand2='') then
                            begin
                                // PENGALAMATAN DIRECT
                                if pjg=3 then
                                    begin

```

```

        // mengambil 2 digit terakhir operand utk alamat data
        x:=copy(operand1,2,2);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        tambah8bit(RA,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADDA','DIR');
        bhs_msn:=bhs_msn+' '+dir;
    end

        // PENGALAMATAN EXTENDED
    else if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand utk alamat data
        x:=copy(operand1,2,4);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        tambah8bit(RA,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADDA','EXT');
        bhs_msn:=bhs_msn+' '+dir;
    end;
end

        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then                // input bilangan
heksadesimal
            bil:=copy(operand1,2,2);
        // else if (persen<>0) and (dolar=0) then          // input bilangan biner
        // begin
        //     pjg_bin:=length(operand1);
        //     if pjg_bin>16 then ShowMessage('Pengalamatan Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
        //         bintohehex(copy(operand1,2,4),bil);
        //     end;
        op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            tambah8bit(RA,dir,hasil,z,c,H);
            bhs_msn:=dtbase('ADDA','IND X');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            tambah8bit(RA,dir,hasil,z,c,H);
            bhs_msn:=dtbase('ADDA','IND Y');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else
        begin
            ShowMessage('Tidak Ada Register Index Tersebut');e:='1';
        end;
    end;
end;
end;

procedure ADDDB(RB,RX,RY,operand1,operand2:string; var hasil:string; var
z,c,H,e:char; var bhs_msn:string);
var

```

```

        imm,bin,dir,bil,x,op2:string;
        pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
        hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1);

            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then // input desimal
                        begin
                            imm:=copy(operand1,2,pjg);
                            destobin(imm,bin,e);
                            pjg_bin:=length(bin);
                            if pjg_bin>8 then
                                begin
                                    showmessage('Register B hanya 8 bit');e:='1';end
                                end
                            else
                                begin
                                    hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
                                    bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                    hsl:=hasil1+hasil2;
                                    bhs_msn:=dtbase('ADDB','IMM');
                                    bhs_msn:=bhs_msn+' '+hsl;
                                    tambah8bit(RB,hsl,hasil,z,c,H);
                                end;
                            end
                        else if (dolar>0) then // input heksa
                            begin
                                imm:=copy(operand1,3,2);
                                bhs_msn:=dtbase('ADDB','IMM');
                                bhs_msn:=bhs_msn+' '+imm;
                                pjg_hex:=length(imm);
                                if pjg_hex>2 then
                                    begin
                                        showmessage('Register A hanya 8 bit');e:='1';end
                                    end
                                else
                                    begin
                                        tambah8bit(RB,imm,hasil,z,c,H);
                                    end;
                                end
                            end
                        else if (persen<>0) then // input biner
                            begin
                                imm:=copy(operand1,3,8); // mengambil biner
                                pjg_bin:=length(imm); // panjang biner
                                if pjg_bin<>8 then
                                    begin
                                        showmessage('Register B hanya 8 bit');e:='1';end
                                    end
                                else
                                    begin
                                        hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                                        bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                        hsl:=hasil1+hasil2;
                                        bhs_msn:=dtbase('ADDB','IMM');
                                        bhs_msn:=bhs_msn+' '+hsl;
                                        tambah8bit(RB,hsl,hasil,z,c,H);
                                    end;
                                end;
                            end;
                        end
                    end
                end
            // PENGALAMATAN DIRECT & EXTENDED
            else if (pagar=0) and (dolar<>0) and (operand2='') then
                begin
                    // PENGALAMATAN DIRECT
                    if pjg=3 then
                        begin
                            // mengambil 2 digit terakhir operand utk alamat data
                            x:=copy(operand1,2,2);
                            alamat:=angka(x);

```

```

        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        tambah8bit(RB,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADDB','DIR');
        bhs_msn:=bhs_msn+' '+dir;
    end

        // PENGALAMATAN EXTENDED
    else if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand utk alamat data
        x:=copy(operand1,2,4);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        tambah8bit(RB,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADDB','EXT');
        bhs_msn:=bhs_msn+' '+dir;
    end;
end

        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then                // input bilangan
heksadesimal
            bil:=copy(operand1,2,2);
        //      else if (persen<>0) and (dolar=0) then        // input bilangan biner
        //      begin
        //          pjg_bin:=length(operand1);
        //          if pjg_bin>16 then ShowMessage('Pengalaman Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
        //              bintohehex(copy(operand1,2,4),bil);
        //          end;
        //          op2:=huruf(copy(operand2,1,1));
        //          if op2='X' then
        //              begin
        //                  buffint:=strtoint(RX)+strtoint(bil);          // alamatnya = data
di Reg X + bil
        //                  x:=inttostr(buffint);
        //                  alamat:=angka(x);
        //                  dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        //                  tambah8bit(RB,dir,hasil,z,c,H);
        //                  bhs_msn:=dtbase('ADDB','IND X');
        //                  bhs_msn:=bhs_msn+' '+dir;
        //              end
        //          else if op2='Y' then
        //              begin
        //                  buffint:=strtoint(RY)+strtoint(bil);          // alamatnya = data
di Reg X + bil
        //                  x:=inttostr(buffint);
        //                  alamat:=angka(x);
        //                  dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        //                  tambah8bit(RB,dir,hasil,z,c,H);
        //                  bhs_msn:=dtbase('ADDB','IND Y');
        //                  bhs_msn:=bhs_msn+' '+dir;
        //              end
        //          else
        //              begin
        //                  ShowMessage('Tidak Ada Register Index Tersebut');e:='1';end;
        //              end;
        //          end;
        //      end;
    end;

    end;
end;

procedure ADCA(RA,RX,RY,RC,operand1,operand2:string; var hasil:string; var
z,c,H,e:char; var bhs_msn:string);
var
    imm,bin,dir,bil,x,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do

```

```

begin
    pagar:=pos('#',operand1); dolar:=pos('$',operand1);
persen:=pos('%',operand1);
    pjg:=length(operand1);
    buffstr:='000'+RC;
    bintohehex(buffstr,buffstr1);
    buffstr2:='0'+buffstr1;
    // PENGALAMATAN IMMEDIATE
if pagar<>0 then
    begin
        if (dolar=0) and (persen=0) then            // input desimal
            begin
                imm:=copy(operand1,2,pjg);
                destobin(imm,bin,e);
                pjg_bin:=length(bin);
                if pjg_bin>8 then
                    begin
                        showmessage('Register A hanya 8 bit');e:='1';end
                    else
                        begin
                            hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
                            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                            hsl:=hasil1+hasil2;
                            bhs_msn:=dtbase('ADCA','IMM');
                            bhs_msn:=bhs_msn+' '+hsl;
                            tambah8bit(RA,buffstr2,hasil,z,c,H);
                            tambah8bit(hasil,hsl,hasil,z,c,H);
                        end;
                    end
                else if (dolar<>0) then                // input heksa
                    begin
                        imm:=copy(operand1,3,2);
                        bhs_msn:=dtbase('ADCA','IMM');
                        bhs_msn:=bhs_msn+' '+imm;
                        pjg_hex:=length(imm);
                        if pjg_hex<>2 then
                            begin
                                showmessage('Register A hanya 8 bit');e:='1';end
                            else
                                begin
                                    tambah8bit(RA,buffstr2,hasil,z,c,H);
                                    tambah8bit(hasil,imm,hasil,z,c,H);
                                end;
                            end
                        else if (persen<>0) then                // input biner
                            begin
                                imm:=copy(operand1,3,8);                // mengambil biner
                                pjg_bin:=length(imm);                // panjang biner
                                if pjg_bin<>8 then
                                    begin
                                        showmessage('Register A hanya 8 bit');e:='1';end
                                    else
                                        begin
                                            hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                                            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                            hsl:=hasil1+hasil2;
                                            bhs_msn:=dtbase('ADCA','IMM');
                                            bhs_msn:=bhs_msn+' '+hsl;
                                            tambah8bit(RA,buffstr2,hasil,z,c,H);
                                            tambah8bit(hasil,hsl,hasil,z,c,H);
                                        end;
                                    end
                                end;
                            end
                        // PENGALAMATAN DIRECT & EXTENDED
                    else if (pagar=0) and (dolar<>0) and (operand2='') then
                        begin
                            // PENGALAMATAN DIRECT
                            if pjg=3 then
                                begin
                                    // mengambil 2 digit terakhir operand utk alamat data
                                    x:=copy(operand1,2,2);
                                    alamat:=angka(x);
                                end;
                            end
                        end
                    end
                end
            end
        end
    end

```

```

        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        tambah8bit(RA,buffstr2,hasil,z,c,H);
        tambah8bit(hasil,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADCA','DIR');
        bhs_msn:=bhs_msn+' '+dir;
    end;

        // PENGALAMATAN EXTENDED
    if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand untuk alamat data
        x:=copy(operand1,2,4);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        tambah8bit(RA,buffstr2,hasil,z,c,H);
        tambah8bit(hasil,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADCA','EXT');
        bhs_msn:=bhs_msn+' '+dir;
    end;
end

        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then                // input bilangan
heksadesimal
            bil:=copy(operand1,2,2);
        // else if (persen<>0) and (dolar=0) then        // input bilangan biner
        // begin
        //     pjg_bin:=length(operand1);
        //     if pjg_bin>16 then ShowMessage('Pengalamatan Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
        //     bintohehex(copy(operand1,2,4),bil);
        // end;
        op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            tambah8bit(RA,buffstr2,hasil,z,c,H);
            tambah8bit(hasil,dir,hasil,z,c,H);
            bhs_msn:=dtbase('ADCA','IND X');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            tambah8bit(RA,buffstr2,hasil,z,c,H);
            tambah8bit(hasil,dir,hasil,z,c,H);
            bhs_msn:=dtbase('ADCA','IND Y');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else
        begin ShowMessage('Tidak Ada Register Index Tersebut');;e:='1';end
        end;
    end;
end;

procedure ADCB(RB,RX,RY,RC,operand1,operand2:string; var hasil:string; var
z,c,H,e:char; var bhs_msn:string);
var
    imm,bin,dir,bil,x,op2:string;

```

```

        pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
        hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1);
            buffstr:='000'+RC;
            bintohe(hex(buffstr,buffstr1));
            buffstr2:='0'+buffstr1;
            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then // input desimal
                        begin
                            imm:=copy(operand1,2,pjg);
                            destobin(imm,bin,e);
                            pjg_bin:=length(bin);
                            if pjg_bin>8 then begin showmessage('Register B hanya 8
bit');e:='1';end
                        end
                    else
                        begin
                            hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
                            bintohe(hex(hsl1,hasil1));bintohe(hex(hsl2,hasil2));
                            hsl:=hasil1+hasil2;
                            bhs_msn:=dtbase('ADCB','IMM');
                            bhs_msn:=bhs_msn+' '+hsl;
                            tambah8bit(RB,buffstr2,hasil,z,c,H);
                            tambah8bit(hasil,hsl,hasil,z,c,H);
                        end;
                    end
                else if (dolar<>0) then // input heksa
                    begin
                        imm:=copy(operand1,3,2);
                        bhs_msn:=dtbase('ADCB','IMM');
                        bhs_msn:=bhs_msn+' '+imm;
                        pjg_hex:=length(imm);
                        if pjg_hex<>2 then begin showmessage('Register B hanya 8
bit');e:='1';end
                    end
                else
                    begin
                        tambah8bit(RB,buffstr2,hasil,z,c,H);
                        tambah8bit(hasil,imm,hasil,z,c,H);
                    end;
                end
            else if (persen<>0) then // input biner
                begin
                    imm:=copy(operand1,3,8); // mengambil biner
                    pjg_bin:=length(imm); // panjang biner
                    if pjg_bin<>8 then begin showmessage('Register B hanya 8
bit');e:='1';end
                end
            else
                begin
                    hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                    bintohe(hex(hsl1,hasil1));bintohe(hex(hsl2,hasil2));
                    hsl:=hasil1+hasil2;
                    bhs_msn:=dtbase('ADCB','IMM');
                    bhs_msn:=bhs_msn+' '+hsl;
                    tambah8bit(RB,buffstr2,hasil,z,c,H);
                    tambah8bit(hasil,hsl,hasil,z,c,H);
                end;
            end;
        end

        // PENGALAMATAN DIRECT & EXTENDED
        else if (pagar=0) and (dolar<>0) and (operand2='') then
            begin
                // PENGALAMATAN DIRECT
                if pjg=3 then
                    begin
                        // mengambil 2 digit terakhir operand utk alamat data
                        x:=copy(operand1,2,2);

```

```

        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);        // mengambil data yg
ada dialamat tsb
        tambah8bit(RB,buffstr2,hasil,z,c,H);
        tambah8bit(hasil,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADCB','DIR');
        bhs_msn:=bhs_msn+' '+dir;
    end;

        // PENGALAMATAN EXTENDED
    if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand untuk alamat data
        x:=copy(operand1,2,4);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);        // mengambil data yg
ada dialamat tsb
        tambah8bit(RB,buffstr2,hasil,z,c,H);
        tambah8bit(hasil,dir,hasil,z,c,H);
        bhs_msn:=dtbase('ADCB','EXT');
        bhs_msn:=bhs_msn+' '+dir;
    end;
end

        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then        // input bilangan
heksadesimal
            bil:=copy(operand1,2,2);
        // else if (persen<>0) and (dolar=0) then        // input bilangan biner
        // begin
        //     pjg_bin:=length(operand1);
        //     if pjg_bin>16 then ShowMessage('Pengalamatan Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
        //         bintohehex(copy(operand1,2,4),bil);
        //     end;
        op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil);        // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);        // mengambil data
yg ada dialamat tsb
            tambah8bit(RB,buffstr2,hasil,z,c,H);
            tambah8bit(hasil,dir,hasil,z,c,H);
            bhs_msn:=dtbase('ADCB','IND X');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil);        // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);        // mengambil data
yg ada dialamat tsb
            tambah8bit(RB,buffstr2,hasil,z,c,H);
            tambah8bit(hasil,dir,hasil,z,c,H);
            bhs_msn:=dtbase('ADCB','IND Y');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');;e:='1';end
        end;
    end;
end;

procedure SUBA(RA,RX,RY,operand1,operand2:string; var hasil:string; var z,e:char;
var bhs_msn:string);
var

```

```

        imm,bin,dir,bil,x,op2:string;
        pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
        hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1); e:='0';

            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then                // input desimal
                        begin
                            imm:=copy(operand1,2,pjg);
                            destobin(imm,bin,e);
                            pjg_bin:=length(bin);
                            if pjg_bin>8 then
                                begin
                                    showmessage('Register A hanya 8 bit');e:='1';
                                end
                            else
                                begin
                                    hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
                                    bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                    hsl:=hasil1+hasil2;
                                    bhs_msn:=dtbase('SUBA','IMM');
                                    bhs_msn:=bhs_msn+' '+hsl;
                                    kurang8bit(RA,hsl,hasil,z,e);
                                end;
                            end
                        else if (dolar<>0) then                // input heksa
                            begin
                                imm:=copy(operand1,3,2);
                                bhs_msn:=dtbase('SUBA','IMM');
                                bhs_msn:=bhs_msn+' '+imm;
                                pjg_hex:=length(imm);
                                if pjg_hex<>2 then
                                    begin
                                        showmessage('Register A hanya 8 bit');e:='1';
                                    end
                                else
                                    begin
                                        kurang8bit(RA,imm,hasil,z,e);
                                    end;
                                end
                            else if (persen<>0) then                // input biner
                                begin
                                    imm:=copy(operand1,3,8);                // mengambil biner
                                    pjg_bin:=length(imm);                // panjang biner
                                    if pjg_bin<>8 then
                                        begin
                                            showmessage('Register A hanya 8 bit');e:='1';
                                        end
                                    else
                                        begin
                                            hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                                            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                            hsl:=hasil1+hasil2;
                                            bhs_msn:=dtbase('SUBA','IMM');
                                            bhs_msn:=bhs_msn+' '+hsl;
                                            kurang8bit(RA,hsl,hasil,z,e);
                                        end;
                                    end;
                                end;
                            end;
                        end
                    // PENGALAMATAN DIRECT & EXTENDED
                    else if (pagar=0) and (dolar<>0) and (operand2='') then
                        begin
                            // PENGALAMATAN DIRECT
                            if pjg=3 then
                                begin

```

```

        // mengambil 2 digit terakhir operand utk alamat data
        x:=copy(operand1,2,2);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        kurang8bit(RA,dir,hasil,z,e);
        bhs_msn:=dtbase('SUBA','DIR');
        bhs_msn:=bhs_msn+' '+dir;
    end

        // PENGALAMATAN EXTENDED
    else if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand utk alamat data
        x:=copy(operand1,2,4);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        kurang8bit(RA,dir,hasil,z,e);
        bhs_msn:=dtbase('SUBA','EXT');
        bhs_msn:=bhs_msn+' '+dir;
    end;
end

        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then                // input bilangan
heksadesimal
            bil:=copy(operand1,2,2);
        // else if (persen<>0) and (dolar=0) then          // input bilangan biner
        // begin
        //     pjg_bin:=length(operand1);
        //     if pjg_bin>16 then ShowMessage('Pengalamatan Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
        //         bintohehex(copy(operand1,2,4),bil);
        //     end;
        op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            kurang8bit(RA,dir,hasil,z,e);
            bhs_msn:=dtbase('SUBA','IND X');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            kurang8bit(RA,dir,hasil,z,e);
            bhs_msn:=dtbase('SUBA','IND Y');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else
        begin
            ShowMessage('Tidak Ada Register Index Tersebut');e:='1';
        end;
    end;
end;
end;

procedure SUBB(RB,RX,RY,operand1,operand2:string; var hasil:string; var z,e:char;
var bhs_msn:string);
var

```

```

        imm,bin,dir,bil,x,op2:string;
        pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
        hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1); e:='0';

            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then                // input desimal
                        begin
                            imm:=copy(operand1,2,pjg);
                            destobin(imm,bin,e);
                            pjg_bin:=length(bin);
                            if pjg_bin>8 then
                                begin
                                    showmessage('Register B hanya 8 bit');e:='1';
                                end
                            else
                                begin
                                    hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
                                    bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                    hsl:=hasil1+hasil2;
                                    bhs_msn:=dtbase('SUBB','IMM');
                                    bhs_msn:=bhs_msn+' '+hsl;
                                    kurang8bit(RB,hsl,hasil,z,e);
                                end;
                            end
                        else if (dolar<>0) then                // input heksa
                            begin
                                imm:=copy(operand1,3,2);
                                bhs_msn:=dtbase('SUBB','IMM');
                                bhs_msn:=bhs_msn+' '+imm;
                                pjg_hex:=length(imm);
                                if pjg_hex<>2 then
                                    begin
                                        showmessage('Register B hanya 8 bit');e:='1';
                                    end
                                else
                                    begin
                                        kurang8bit(RB,imm,hasil,z,e);
                                    end;
                                end
                            end
                        else if (persen<>0) then                // input biner
                            begin
                                imm:=copy(operand1,3,8);                // mengambil biner
                                pjg_bin:=length(imm);                // panjang biner
                                if pjg_bin<>8 then
                                    begin
                                        showmessage('Register B hanya 8 bit');e:='1';
                                    end
                                else
                                    begin
                                        hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                                        bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                        hsl:=hasil1+hasil2;
                                        bhs_msn:=dtbase('SUBB','IMM');
                                        bhs_msn:=bhs_msn+' '+hsl;
                                        kurang8bit(RB,hsl,hasil,z,e);
                                    end;
                                end;
                            end;
                        end
                    end
                end
            // PENGALAMATAN DIRECT & EXTENDED
            else if (pagar=0) and (dolar<>0) and (operand2='') then
                begin
                    // PENGALAMATAN DIRECT
                    if pjg=3 then
                        begin

```

```

        // mengambil 2 digit terakhir operand utk alamat data
        x:=copy(operand1,2,2);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        kurang8bit(RB,dir,hasil,z,e);
        bhs_msn:=dtbase('SUBB','DIR');
        bhs_msn:=bhs_msn+' '+dir;
    end

        // PENGALAMATAN EXTENDED
    else if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand utk alamat data
        x:=copy(operand1,2,4);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
        kurang8bit(RB,dir,hasil,z,e);
        bhs_msn:=dtbase('SUBB','EXT');
        bhs_msn:=bhs_msn+' '+dir;
    end;
end

        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then          // input bilangan
heksadesimal
            bil:=copy(operand1,2,2);
        // else if (persen<>0) and (dolar=0) then          // input bilangan biner
        // begin
        //     pjg_bin:=length(operand1);
        //     if pjg_bin>16 then ShowMessage('Pengalamatan Berindeks
Membutuhkan Alamat Relatif Maksimal 16 bit')
        //         bintohehex(copy(operand1,2,4),bil);
        //     end;
        op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            kurang8bit(RB,dir,hasil,z,e);
            bhs_msn:=dtbase('SUBB','IND X');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil);          // alamatnya = data
di Reg X + bil
            x:=inttostr(buffint);
            alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg
ada dialamat tsb
            kurang8bit(RB,dir,hasil,z,e);
            bhs_msn:=dtbase('SUBB','IND Y');
            bhs_msn:=bhs_msn+' '+dir;
        end
        else
        begin
            ShowMessage('Tidak Ada Register Index Tersebut');e:='1';
        end;
    end;
end;
end;

procedure SBA(RA,RB:string; var hasil,bhs_msn:string; var z,c,e:char);
begin
    buffstr1:=hextobin(RA); buffstr2:=hextobin(RB);

```

```

bintodes(buffstr1,buffstr1); bintodes(buffstr2,buffstr2);
buffint1:=strtoint(buffstr1); buffint2:=strtoint(buffstr2);
if buffint1<buffint2 then
begin
  ShowMessage('nilai pada Register A < Register B');e:='1'
end
else
begin
  kurang8bit(RA,RB,hasil,z,e);
  bhs_msn:=dtbase('SBA','INH');
end;
end;

procedure LDA(RA,RX,RY,operand1,operand2:string; var hasil,bhs_msn:string; var
e:char);
var
  dir,bil,x,op2:string;
  pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
  hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
  with form1 do
  begin
    pagar:=pos('#',operand1); dolar:=pos('$',operand1);
    persen:=pos('%',operand1);
    pjg:=length(operand1);
    // PENGALAMATAN IMMEDIATE
    if pagar<>0 then
    begin
      if (dolar=0) and (persen=0) then // input desimal
      begin
        buffstr:=copy(operand1,2,pjg); destobin(buffstr,buffstr1,e);
        pjg_bin:=length(buffstr1);
        if pjg_bin>8 then begin showmessage('Register A hanya 8
bit');e:='1';end
      else
      begin
        hsl1:=copy(buffstr1,1,4);hsl2:=copy(buffstr1,5,8);
        bintohe(hsl1,hasil1);bintohe(hsl2,hasil2);
        hsl:=hasil1+hasil2;
        hasil:=hsl;
        bhs_msn:=dtbase('LDA','IMM'); bhs_msn:=bhs_msn+' '+hsl;
      end;
    end
    else if (dolar<>0) then // input heksa
    begin
      buffstr:=copy(operand1,3,2);
      bhs_msn:=dtbase('LDA','IMM'); bhs_msn:=bhs_msn+' '+buffstr;
      pjg_hex:=length(buffstr);
      if pjg_hex>2 then begin showmessage('Register A hanya 8
bit');e:='1';end
    else hasil:=buffstr;
    end
    else if (persen<>0) then // input biner
    begin
      buffstr:=copy(operand1,3,8); // mengambil biner
      pjg_bin:=length(buffstr); // panjang biner
      if pjg_bin>8 then begin showmessage('Register A hanya 8
bit');e:='1';end
    else
    begin
      hsl1:=copy(buffstr,1,4);hsl2:=copy(buffstr,5,8);
      bintohe(hsl1,hasil1);bintohe(hsl2,hasil2);
      hsl:=hasil1+hasil2;
      bhs_msn:=dtbase('LDA','IMM'); bhs_msn:=bhs_msn+' '+hsl;
      hasil:=hsl;
    end;
  end;
end
// PENGALAMATAN DIRECT & EXTENDED
else if (pagar=0) and (dolar<>0) and (operand2='') then
begin
  // PENGALAMATAN DIRECT
  if pjg=3 then

```

```

begin
    // mengambil 2 digit terakhir operand utk alamat data
    x:=copy(operand1,2,2); alamat:=angka(x);
    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
    hasil:=dir;
    bhs_msn:=dtbase('LDA','DIR'); bhs_msn:=bhs_msn+' '+dir;
end;
    // PENGALAMATAN EXTENDED
if pjg=5 then
begin
    // mengambil 4 digit terakhir operand untuk alamat data
    x:=copy(operand1,2,4); alamat:=angka(x);
    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
    hasil:=dir;
    bhs_msn:=dtbase('LDA','EXT'); bhs_msn:=bhs_msn+' '+dir;
end;
    // PENGALAMATAN INDEXED
else if (pagar=0) and (dolar<>0) and (operand2<>'') then
begin
    if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
            hasil:=dir;
            bhs_msn:=dtbase('LDA','IND X'); bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil); // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
            hasil:=dir;
            bhs_msn:=dtbase('LDA','IND Y'); bhs_msn:=bhs_msn+' '+dir;
        end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
        end;
    end;
end;

procedure LDB(RB,RX,RY,operand1,operand2:string; var hasil,bhs_msn:string; var
e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
    begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
        persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN IMMEDIATE
        if pagar<>0 then
        begin
            if (dolar=0) and (persen=0) then // input desimal
            begin
                buffstr:=copy(operand1,2,pjg); destobin(buffstr,buffstr1,e);
                pjg_bin:=length(buffstr1);
                if pjg_bin>8 then begin showmessage('Register B hanya 8
bit');e:='1';end
            else

```

```

begin
    hsl1:=copy(buffstr1,1,4);hsl2:=copy(buffstr1,5,8);
    bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
    hsl:=hasil1+hasil2; hasil:=hsl;
    bhs_msn:=dtbase('LDB','IMM'); bhs_msn:=bhs_msn+' '+hsl;
end;
end
else if (dolar<>0) then // input heksa
begin
    buffstr:=copy(operand1,3,2); pjg_hex:=length(buffstr);
    bhs_msn:=dtbase('LDB','IMM'); bhs_msn:=bhs_msn+' '+buffstr;
    if pjg_hex<>2 then begin showmessage('Register B hanya 8
bit');e:='1';end
    else hasil:=buffstr;
    end
else if (persen<>0) then // input biner
begin
    buffstr:=copy(operand1,3,8); // mengambil biner
    pjg_bin:=length(buffstr); // panjang biner
    if pjg_bin<>8 then begin showmessage('Register B hanya 8
bit');e:='1';end
    else
begin
    hsl1:=copy(buffstr,1,4);hsl2:=copy(buffstr,5,8);
    bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
    hsl:=hasil1+hasil2; hasil:=hsl;
    bhs_msn:=dtbase('LDB','IMM'); bhs_msn:=bhs_msn+' '+hsl;
    end;
end;
end
// PENGALAMATAN DIRECT & EXTENDED
else if (pagar=0) and (dolar<>0) and (operand2='') then
begin
    // PENGALAMATAN DIRECT
    if pjg=3 then
begin
    // mengambil 2 digit terakhir operand utk alamat data
    x:=copy(operand1,2,2); alamat:=angka(x);
    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
    hasil:=dir;
    bhs_msn:=dtbase('LDB','DIR'); bhs_msn:=bhs_msn+' '+dir;
end;
    // PENGALAMATAN EXTENDED
    if pjg=5 then
begin
    // mengambil 4 digit terakhir operand untuk alamat data
    x:=copy(operand1,2,4); alamat:=angka(x);
    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
    hasil:=dir;
    bhs_msn:=dtbase('LDB','EXT'); bhs_msn:=bhs_msn+' '+dir;
end;
end
    // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
begin
    if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
    bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
    if op2='X' then
begin
    buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
data di Reg X + bil
    x:=inttostr(buffint); alamat:=angka(x);
    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
    hasil:=dir;
    bhs_msn:=dtbase('LDB','IND X'); bhs_msn:=bhs_msn+' '+dir;
    end
    else if op2='Y' then
begin

```

```

buffint:=strtoint(RY)+strtoint(bil);           // alamatnya =
data di Reg X + bil
x:=inttostr(buffint); alamat:=angka(x);
dir:=copy(Memory.Lines[alamat],14,2);         // mengambil data
yg ada dialamat tsb
hasil:=dir;
bhs_msn:=dtbase('LDB','IND Y'); bhs_msn:=bhs_msn+' '+dir;
end
else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
end;
end;
end;

procedure LDD(RX,RY,operand1,operand2:string; var hasilRA,hasilRB,bhs_msn:string;
var e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil,hasil1,hasil2,hasil3,hasil4,hsl,hsl1,hsl2,hsl3,hsl4:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1);
            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then           // input desimal
                        begin
                            buffstr:=copy(operand1,2,pjg); destobin(buffstr,buffstr1,e);
                            pjg_bin:=length(buffstr1);
                            if pjg_bin>16 then begin showmessage('Register D hanya 16
bit');e:='1';end
                        end
                    else
                        begin
                            hsl1:=copy(buffstr1,1,4);hsl2:=copy(buffstr1,5,4);
                            hsl3:=copy(buffstr1,9,4);hsl4:=copy(buffstr1,13,4);
                            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                            bintohehex(hsl3,hasil3);bintohehex(hsl4,hasil4);
                            hsl:=hasil1+hasil2+hasil3+hasil4;
                            buffint:=length(hsl);
                            if buffint<>4 then hsl:=jadiin4bit(hsl); hasil:=hsl;
                            bhs_msn:=dtbase('LDD','IMM'); bhs_msn:=bhs_msn+' '+hsl;
                        end;
                    end
                end
            else if (dolar<>0) then           // input heksa
                begin
                    buffstr:=copy(operand1,3,4);
                    bhs_msn:=dtbase('LDD','IMM'); bhs_msn:=bhs_msn+' '+buffstr;
                    pjg_hex:=length(buffstr);
                    if pjg_hex<>4 then begin showmessage('Register D hanya 16
bit');e:='1';end
                end
            else hasil:=buffstr;
            end
            else if (persen<>0) then           // input biner
                begin
                    buffstr:=copy(operand1,3,16);           // mengambil biner
                    pjg_bin:=length(buffstr);           // panjang biner
                    if pjg_bin<>16 then begin showmessage('Register D hanya 16
bit');e:='1';end
                end
            else
                begin
                    hsl1:=copy(buffstr,1,4);hsl2:=copy(buffstr,5,4);
                    hsl3:=copy(buffstr,9,4);hsl4:=copy(buffstr,13,4);
                    bintohehex(hsl1,hsl1);bintohehex(hsl2,hsl2);
                    bintohehex(hsl3,hsl3);bintohehex(hsl4,hsl4);
                    hsl:=hsl1+hsl2+hsl3+hsl4; hasil:=hsl;
                    bhs_msn:=dtbase('LDD','IMM'); bhs_msn:=bhs_msn+' '+hsl;
                end;
            end;
        end
    end
end
end

```

```

// PENGALAMATAN DIRECT & EXTENDED
else if (pagar=0) and (dolar<>0) and (operand2='') then
begin
    // PENGALAMATAN DIRECT
    if pjg=3 then
    begin
        // mengambil 2 digit terakhir operand utk alamat data
        x:=copy(operand1,2,2); alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
        hasil:='00'+dir;
        bhs_msn:=dtbase('LDD','DIR'); bhs_msn:=bhs_msn+' '+dir;
        end;
        // PENGALAMATAN EXTENDED
    if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand untuk alamat data
        x:=copy(operand1,2,4); alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
        hasil:='00'+dir;
        bhs_msn:=dtbase('LDD','EXT'); bhs_msn:=bhs_msn+' '+dir;
        end;
    end
    // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
            hasil:='00'+dir;
            bhs_msn:=dtbase('LDD','IND X'); bhs_msn:=bhs_msn+' '+dir;
            end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil); // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
            hasil:='00'+dir;
            bhs_msn:=dtbase('LDD','IND Y'); bhs_msn:=bhs_msn+' '+dir;
            end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
        end;
        hasilRA:=copy(hasil,1,2); hasilRB:=copy(hasil,3,2);
        end;
    end;

procedure LDX(RX,RY,operand1,operand2:string; var hasil,bhs_msn:string; var
e:char);
var
    dir1,dir2,bil,x,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil1,hasil2,hasil3,hasil4,hs1,hs11,hs12,hs13,hs14:string;
begin
    with form1 do
    begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
        persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN IMMEDIATE
        if pagar<>0 then
        begin
            if (dolar=0) and (persen=0) then // input desimal

```

```

begin
    buffstr:=copy(operand1,2,pjg); destobin(buffstr,buffstr1,e);
    pjg_bin:=length(buffstr1);
    if pjg_bin>16 then begin showmessage('Register X hanya 16
bit');e:='1';end
    else
        begin
            hsl1:=copy(buffstr1,1,4);hsl2:=copy(buffstr1,5,4);
            hsl3:=copy(buffstr1,9,4);hsl4:=copy(buffstr1,13,4);
            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
            bintohehex(hsl3,hasil3);bintohehex(hsl4,hasil4);
            hsl:=hasil1+hasil2+hasil3+hasil4;
            buffint:=length(hsl);
            if buffint<>4 then hsl:=jadiin4bit(hsl); hasil:=hsl;
            bhs_msn:=dtbase('LDX','IMM'); bhs_msn:=bhs_msn+' '+hsl;
        end;
    end
    else if (dolar<>0) then // input heksa
        begin
            buffstr:=copy(operand1,3,4);
            bhs_msn:=dtbase('LDX','IMM'); bhs_msn:=bhs_msn+' '+buffstr;
            pjg_hex:=length(buffstr);
            if pjg_hex<>4 then begin showmessage('Register X hanya 16
bit');e:='1';end
            else hasil:=buffstr;
        end
    else if (persen<>0) then // input biner
        begin
            buffstr:=copy(operand1,3,16); // mengambil biner
            pjg_bin:=length(buffstr); // panjang biner
            if pjg_bin<>16 then begin showmessage('Register X hanya 16
bit');e:='1';end
            else
                begin
                    hsl1:=copy(buffstr,1,4);hsl2:=copy(buffstr,5,4);
                    hsl3:=copy(buffstr,9,4);hsl4:=copy(buffstr,13,4);
                    bintohehex(hsl1,hsl1);bintohehex(hsl2,hsl2);
                    bintohehex(hsl3,hsl3);bintohehex(hsl4,hsl4);
                    hsl:=hsl1+hsl2+hsl3+hsl4; hasil:=hsl;
                    bhs_msn:=dtbase('LDX','IMM'); bhs_msn:=bhs_msn+' '+hsl;
                end;
            end;
        end
        // PENGALAMATAN DIRECT & EXTENDED
    else if (pagar=0) and (dolar<>0) and (operand2='') then
        begin
            // PENGALAMATAN DIRECT
            if pjg=3 then
                begin
                    // mengambil 2 digit terakhir operand utk alamat data
                    x:=copy(operand1,2,2); alamat:=angka(x);
                    dir1:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                    dir2:=copy(Memory.Lines[alamat+1],14,2);
                    hasil:=dir1+dir2;
                    bhs_msn:=dtbase('LDX','DIR'); bhs_msn:=bhs_msn+' '+dir1;
                end;
            // PENGALAMATAN EXTENDED
            if pjg=5 then
                begin
                    // mengambil 4 digit terakhir operand untuk alamat data
                    x:=copy(operand1,2,4); alamat:=angka(x);
                    dir1:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                    dir2:=copy(Memory.Lines[alamat+1],14,2);
                    hasil:=dir1+dir2;
                    bhs_msn:=dtbase('LDX','EXT'); bhs_msn:=bhs_msn+' '+dir1;
                end;
            end
            // PENGALAMATAN INDEXED
        else if (pagar=0) and (dolar<>0) and (operand2<>'') then
            begin

```

```

        if (dolar<>0) and (persen=0) then          // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
            begin
                buffint:=strtoint(RX)+strtoint(bil);          // alamatnya =
data di Reg X + bil
                x:=inttostr(buffint); alamat:=angka(x);
                dir1:=copy(Memory.Lines[alamat],14,2);          // mengambil data
yg ada dialamat tsb
                dir2:=copy(Memory.Lines[alamat+1],14,2);
                hasil:=dir1+dir2;
                bhs_msn:=dtbase('LDX','IND X'); bhs_msn:=bhs_msn+' '+dir1;
            end
        else if op2='Y' then
            begin
                buffint:=strtoint(RY)+strtoint(bil);          // alamatnya =
data di Reg X + bil
                x:=inttostr(buffint); alamat:=angka(x);
                dir1:=copy(Memory.Lines[alamat],14,2);          // mengambil data
yg ada dialamat tsb
                dir2:=copy(Memory.Lines[alamat+1],14,2);
                hasil:=dir1+dir2;
                bhs_msn:=dtbase('LDX','IND Y'); bhs_msn:=bhs_msn+' '+dir1;
            end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
        end;
    end;
end;

procedure LDY(RX,RY,operand1,operand2:string; var hasil,bhs_msn:string; var
e:char);
var
    dir1,dir2,bil,x,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil1,hasil2,hasil3,hasil4,hs1,hs11,hs12,hs13,hs14:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1);
            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then          // input desimal
                        begin
                            buffstr:=copy(operand1,2,pjg); destobin(buffstr,buffstr1,e);
                            pjg_bin:=length(buffstr1);
                            if pjg_bin>16 then begin showmessage('Register Y hanya 16
bit');e:='1';end
                        end
                    else
                        begin
                            hs11:=copy(buffstr1,1,4);hs12:=copy(buffstr1,5,4);
                            hs13:=copy(buffstr1,9,4);hs14:=copy(buffstr1,13,4);
                            bintohehex(hs11,hasil1);bintohehex(hs12,hasil2);
                            bintohehex(hs13,hasil3);bintohehex(hs14,hasil4);
                            hs1:=hasil1+hasil2+hasil3+hasil4;
                            buffint:=length(hs1);
                            if buffint<>4 then hs1:=jadiin4bit(hs1); hasil:=hs1;
                            bhs_msn:=dtbase('LDY','IMM'); bhs_msn:=bhs_msn+' '+hs1;
                        end;
                    end
                else if (dolar<>0) then          // input heksa
                    begin
                        buffstr:=copy(operand1,3,4);
                        bhs_msn:=dtbase('LDY','IMM'); bhs_msn:=bhs_msn+' '+buffstr;
                        pjg_hex:=length(buffstr);
                        if pjg_hex<>4 then begin showmessage('Register Y hanya 16
bit');e:='1';end
                    else hasil:=buffstr;
                    end
                end
            end
        end
    end
end

```

```

else if (persen<>0) then                                // input biner
begin
    buffstr:=copy(operand1,3,16);                      // mengambil biner
    pjg_bin:=length(buffstr);                          // panjang biner
    if pjg_bin<>16 then begin showmessage('Register Y hanya 16
bit');e:='1';end
    else
    begin
        hsl1:=copy(buffstr,1,4);hsl2:=copy(buffstr,5,4);
        hsl3:=copy(buffstr,9,4);hsl4:=copy(buffstr,13,4);
        bintohehex(hsl1,hsl1);bintohehex(hsl2,hsl2);
        bintohehex(hsl3,hsl3);bintohehex(hsl4,hsl4);
        hsl:=hsl1+hsl2+hsl3+hsl4; hasil:=hsl;
        bhs_msn:=dtbase('LDY','IMM'); bhs_msn:=bhs_msn+' '+hsl;
    end;
end;
end
// PENGALAMATAN DIRECT & EXTENDED
else if (pagar=0) and (dolar<>0) and (operand2='') then
begin
    // PENGALAMATAN DIRECT
    if pjg=3 then
    begin
        // mengambil 2 digit terakhir operand utk alamat data
        x:=copy(operand1,2,2); alamat:=angka(x);
        dir1:=copy(Memory.Lines[alamat],14,2);        // mengambil data yg
ada dialamat tsb
        dir2:=copy(Memory.Lines[alamat+1],14,2);
        hasil:=dir1+dir2;
        bhs_msn:=dtbase('LDY','DIR'); bhs_msn:=bhs_msn+' '+dir1;
    end;
    // PENGALAMATAN EXTENDED
    if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand untuk alamat data
        x:=copy(operand1,2,4); alamat:=angka(x);
        dir1:=copy(Memory.Lines[alamat],14,2);        // mengambil data yg
ada dialamat tsb
        dir2:=copy(Memory.Lines[alamat+1],14,2);
        hasil:=dir1+dir2;
        bhs_msn:=dtbase('LDY','EXT'); bhs_msn:=bhs_msn+' '+dir1;
    end;
end
// PENGALAMATAN INDEXED
else if (pagar=0) and (dolar<>0) and (operand2<>'') then
begin
    if (dolar<>0) and (persen=0) then                // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil);      // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir1:=copy(Memory.Lines[alamat],14,2);    // mengambil data
yg ada dialamat tsb
            dir2:=copy(Memory.Lines[alamat+1],14,2);
            hasil:=dir1+dir2;
            bhs_msn:=dtbase('LDY','IND X'); bhs_msn:=bhs_msn+' '+dir1;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil);      // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir1:=copy(Memory.Lines[alamat],14,2);    // mengambil data
yg ada dialamat tsb
            dir2:=copy(Memory.Lines[alamat+1],14,2);
            hasil:=dir1+dir2;
            bhs_msn:=dtbase('LDY','IND Y'); bhs_msn:=bhs_msn+' '+dir1;
        end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;

```

```

        end;
    end;
end;

procedure LDS(RX,RY,operand1,operand2:string; var hasil,bhs_msn:string; var
e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
    hasil1,hasil2,hasil3,hasil4,hsl,hsl1,hsl2,hsl3,hsl4:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjg:=length(operand1);
            // PENGALAMATAN IMMEDIATE

            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then // input desimal
                        begin
                            buffstr:=copy(operand1,2,pjg); destobin(buffstr,buffstr1,e);
                            pjg_bin:=length(buffstr1);
                            if pjg_bin>16 then begin showmessage('Stack Pointer hanya 16
bit');e:='1';end
                        else
                            begin
                                hsl1:=copy(buffstr1,1,4);hsl2:=copy(buffstr1,5,4);
                                hsl3:=copy(buffstr1,9,4);hsl4:=copy(buffstr1,13,4);
                                bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                                bintohehex(hsl3,hasil3);bintohehex(hsl4,hasil4);
                                hsl:=hasil1+hasil2+hasil3+hasil4;
                                buffint:=length(hsl);
                                if buffint<>4 then hsl:=jadiin4bit(hsl); hasil:=hsl;
                                bhs_msn:=dtbase('LDS','IMM'); bhs_msn:=bhs_msn+' '+hsl;
                            end;
                        end
                    else if (dolar<>0) then // input heksa
                        begin
                            buffstr:=copy(operand1,3,4);
                            bhs_msn:=dtbase('LDS','IMM'); bhs_msn:=bhs_msn+' '+buffstr;
                            pjg_hex:=length(buffstr);
                            if pjg_hex<>4 then begin showmessage('Stack Pointer hanya 16
bit');e:='1';end
                        else hasil:=buffstr;
                        end
                    else if (persen<>0) then // input biner
                        begin
                            buffstr:=copy(operand1,3,16); // mengambil biner
                            pjg_bin:=length(buffstr); // panjang biner
                            if pjg_bin<>16 then begin showmessage('Stack Pointer hanya 16
bit');e:='1';end
                        else
                            begin
                                hsl1:=copy(buffstr,1,4);hsl2:=copy(buffstr,5,4);
                                hsl3:=copy(buffstr,9,4);hsl4:=copy(buffstr,13,4);
                                bintohehex(hsl1,hsl1);bintohehex(hsl2,hsl2);
                                bintohehex(hsl3,hsl3);bintohehex(hsl4,hsl4);
                                hsl:=hsl1+hsl2+hsl3+hsl4; hasil:=hsl;
                                bhs_msn:=dtbase('LDS','IMM'); bhs_msn:=bhs_msn+' '+hsl;
                            end;
                        end;
                    end;
                end
            // PENGALAMATAN DIRECT & EXTENDED
            else if (pagar=0) and (dolar<>0) and (operand2='') then
                begin
                    // PENGALAMATAN DIRECT
                    if pjg=3 then
                        begin
                            // mengambil 2 digit terakhir operand utk alamat data
                            x:=copy(operand1,2,2); alamat:=angka(x);
                            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
                            ada dialamat tsb

```

```

        hasil:='00'+dir;
        bhs_msn:=dtbase('LDS','DIR'); bhs_msn:=bhs_msn+' '+dir;
    end;
    // PENGALAMATAN EXTENDED
    if pjg=5 then
    begin
        // mengambil 4 digit terakhir operand untuk alamat data
        x:=copy(operand1,2,4); alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
        hasil:='00'+dir;
        bhs_msn:=dtbase('LDS','EXT'); bhs_msn:=bhs_msn+' '+dir;
    end;
    // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
    begin
        if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
        begin
            buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
            hasil:='00'+dir;
            bhs_msn:=dtbase('LDS','IND X'); bhs_msn:=bhs_msn+' '+dir;
        end
        else if op2='Y' then
        begin
            buffint:=strtoint(RY)+strtoint(bil); // alamatnya =
data di Reg X + bil
            x:=inttostr(buffint); alamat:=angka(x);
            dir:=copy(Memory.Lines[alamat],14,2); // mengambil data
yg ada dialamat tsb
            hasil:='00'+dir;
            bhs_msn:=dtbase('LDS','IND Y'); bhs_msn:=bhs_msn+' '+dir;
        end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
        end;
    end;
end;

procedure DCRA(input:string; var hasil,bhs_msn:string; var N,e,z:char);
var x,x1,x2,hsl:string;
begin
    if input='00' then
    begin
        N:='1';hasil:='00';z:='1'; bhs_msn:=dtbase('DECA','INH');
        ShowMessage('Register A sudah "00"');
    end
    else
    begin
        buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
        x1:=hextobin(buffstr1); x2:=hextobin(buffstr2);
        x:=x1+x2;
        bintodes(x,x);
        buffint:=strtoint(x);
        dec(buffint);
        destobin(inttostr(buffint),buffstr,e);
        x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
        bintohe(x1,x1); bintohe(x2,x2);
        hsl:=x1+x2; hasil:=hsl;N:='0';
        if hsl='00' then z:='1' else z:='0';
        bhs_msn:=dtbase('DECA','INH');
    end;
end;

procedure DCRB(input:string; var hasil,bhs_msn:string; var N,e,z:char);
var x,x1,x2,hsl:string;

```

```

begin
  if input='00' then
    begin
      N:='1';hasil:='00';z:='1'; bhs_msn:=dtbase('DECB','INH');
      ShowMessage('Register B sudah "00"');
    end
  else
    begin
      buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
      x1:=hextobin(buffstr1); x2:=hextobin(buffstr2);
      x:=x1+x2;
      bintodes(x,x);
      buffint:=strtoint(x);
      dec(buffint);
      destobin(inttostr(buffint),buffstr,e);
      x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
      bintohe(x1,x1); bintohe(x2,x2);
      if hsl='00' then z:='1' else z:='0';
      hsl:=x1+x2; hasil:=hsl;N:='0';
      bhs_msn:=dtbase('DECB','INH');
    end;
  end;

procedure DCRX(input:string; var hasil,bhs_msn:string; var N,e,z:char);
var x,x1,x2,x3,x4,hsl:string; pjg:integer;
begin
  if input='0000' then
    begin
      N:='1';hasil:='0000'; z:='1'; bhs_msn:=dtbase('DEX','INH');
      ShowMessage('Register X sudah "0000"');
    end
  else
    begin
      buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
      buffstr3:=copy(input,3,1); buffstr4:=copy(input,4,1);
      x1:=hextobin(buffstr1); x2:=hextobin(buffstr2);
      x3:=hextobin(buffstr3); x4:=hextobin(buffstr4);
      x:=x1+x2+x3+x4;
      bintodes(x,x); buffint:=strtoint(x); dec(buffint); // decrement
      destobin(inttostr(buffint),buffstr,e);
      pjg:=length(buffstr);
      if pjg<>16 then
        begin
          buffstr:=jadiin16bit(buffstr);
        end;
      x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
      x3:=copy(buffstr,9,4); x4:=copy(buffstr,13,4);
      bintohe(x1,x1); bintohe(x2,x2);
      bintohe(x3,x3); bintohe(x4,x4);
      hsl:=x1+x2+x3+x4; hasil:=hsl;N:='0';
      if hsl='0000' then z:='1' else z:='0';
      bhs_msn:=dtbase('DEX','INH');
    end;
  end;

procedure DCRY(input:string; var hasil,bhs_msn:string; var N,e,z:char);
var x,x1,x2,x3,x4,hsl:string; pjg:integer;
begin
  if input='0000' then
    begin
      N:='1';hasil:='0000'; z:='1'; bhs_msn:=dtbase('DEY','INH');
      ShowMessage('Register Y sudah "0000"');
    end
  else
    begin
      buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
      buffstr3:=copy(input,3,1); buffstr4:=copy(input,4,1);
      x1:=hextobin(buffstr1); x2:=hextobin(buffstr2);
      x3:=hextobin(buffstr3); x4:=hextobin(buffstr4);
      x:=x1+x2+x3+x4;
      bintodes(x,x); buffint:=strtoint(x); dec(buffint); // decrement
      destobin(inttostr(buffint),buffstr,e);
      pjg:=length(buffstr);

```

```

        if pjpg<>16 then
            begin
                buffstr:=jadiin16bit(buffstr);
            end;
            x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
            x3:=copy(buffstr,9,4); x4:=copy(buffstr,13,4);
            bintohe(x1,x1); bintohe(x2,x2);
            bintohe(x3,x3); bintohe(x4,x4);
            hsl:=x1+x2+x3+x4; hasil:=hsl;N:='0';
            if hsl='0000' then z:='1' else z:='0';
            bhs_msn:=dtbase('DEY','INH');
        end;
    end;

procedure DCSP(input:string; var hasil,bhs_msn:string; var N,e,z:char);
var x,x1,x2,x3,x4,hsl:string; pjpg:integer;
begin
    if input='0000' then
        begin
            N:='1';hasil:='0000'; z:='1'; bhs_msn:=dtbase('DEY','INH');
            ShowMessage('Stack Pointer sudah "0000"');
        end
    else
        begin
            buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
            buffstr3:=copy(input,3,1); buffstr4:=copy(input,4,1);
            x1:=hextoBin(buffstr1); x2:=hextoBin(buffstr2);
            x3:=hextoBin(buffstr3); x4:=hextoBin(buffstr4);
            x:=x1+x2+x3+x4;
            bintodes(x,x); buffint:=strtoInt(x); dec(buffint); // decrement
            destobin(inttostr(buffint),buffstr,e);
            pjpg:=length(buffstr);
            if pjpg<>16 then
                begin
                    buffstr:=jadiin16bit(buffstr);
                end;
                x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
                x3:=copy(buffstr,9,4); x4:=copy(buffstr,13,4);
                bintohe(x1,x1); bintohe(x2,x2);
                bintohe(x3,x3); bintohe(x4,x4);
                hsl:=x1+x2+x3+x4; hasil:=hsl;N:='0';
                if hsl='0000' then z:='1' else z:='0';
                bhs_msn:=dtbase('DES','INH');
            end;
        end;
    end;

procedure INCRA(input:string; var hasil,bhs_msn:string; var e:char);
var x,x1,x2,hsl,hsl1,hsl2:string;
begin
    buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
    x1:=hextoBin(buffstr1); x2:=hextoBin(buffstr2);
    x:=x1+x2;
    bintodes(x,buffstr);
    buffint:=strtoInt(buffstr);
    inc(buffint);
    destobin(inttostr(buffint),buffstr,e);
    x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
    bintohe(x1,hsl1); bintohe(x2,hsl2);
    hsl:=hsl1+hsl2;
    hasil:=hsl;
    bhs_msn:=dtbase('INCA','INH');
end;

procedure INCRB(input:string; var hasil,bhs_msn:string; var e:char);
var x,x1,x2,hsl,hsl1,hsl2:string;
begin
    buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
    x1:=hextoBin(buffstr1); x2:=hextoBin(buffstr2);
    x:=x1+x2;
    bintodes(x,buffstr);
    buffint:=strtoInt(buffstr);
    inc(buffint);
    destobin(inttostr(buffint),buffstr,e);

```

```

    x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
    bintohe(x1,hsl1); bintohe(x2,hsl2);
    hsl:=hsl1+hsl2;
    hasil:=hsl;
    bhs_msn:=dtbase('INCB','INH');
end;

procedure INCRX(input:string; var hasil,bhs_msn:string; var e:char);
var x,x1,x2,x3,x4,hsl:string; jpg:integer;
begin
    buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
    buffstr3:=copy(input,3,1); buffstr4:=copy(input,4,1);
    x1:=hextobin(buffstr1); x2:=hextobin(buffstr2);
    x3:=hextobin(buffstr3); x4:=hextobin(buffstr4);
    x:=x1+x2+x3+x4;
    bintodes(x,x); buffint:=strtoint(x); inc(buffint);    // increment
    destobin(inttostr(buffint),buffstr,e);
    jpg:=length(buffstr);
    if jpg<>16 then
        begin
            buffstr:=jadiin16bit(buffstr);
        end;
    x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
    x3:=copy(buffstr,9,4); x4:=copy(buffstr,13,4);
    bintohe(x1,x1); bintohe(x2,x2); bintohe(x3,x3); bintohe(x4,x4);
    hsl:=x1+x2+x3+x4;
    hasil:=hsl;
    bhs_msn:=dtbase('INX','INH');
end;

procedure INCRY(input:string; var hasil,bhs_msn:string; var e:char);
var x,x1,x2,x3,x4,hsl:string; jpg:integer;
begin
    buffstr1:=copy(input,1,1); buffstr2:=copy(input,2,1);
    buffstr3:=copy(input,3,1); buffstr4:=copy(input,4,1);
    x1:=hextobin(buffstr1); x2:=hextobin(buffstr2);
    x3:=hextobin(buffstr3); x4:=hextobin(buffstr4);
    x:=x1+x2+x3+x4;
    bintodes(x,x); buffint:=strtoint(x); inc(buffint);    // increment
    destobin(inttostr(buffint),buffstr,e);
    jpg:=length(buffstr);
    if jpg<>16 then
        begin
            buffstr:=jadiin16bit(buffstr);
        end;
    x1:=copy(buffstr,1,4); x2:=copy(buffstr,5,4);
    x3:=copy(buffstr,9,4); x4:=copy(buffstr,13,4);
    bintohe(x1,x1); bintohe(x2,x2); bintohe(x3,x3); bintohe(x4,x4);
    hsl:=x1+x2+x3+x4;
    hasil:=hsl;
    bhs_msn:=dtbase('INY','INH');
end;

procedure ANDA(RA,RX,RY,operand1,operand2:string; var hasil:string; var z,e:char;
var bhs_msn:string);
var
    imm,bin,dir,bil,x,op2:string;
    pagar,dolar,persen,pjpg,pjpg_bin,pjpg_hex,alamat:integer;
    hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
    with form1 do
        begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjpg:=length(operand1);

            // PENGALAMATAN IMMEDIATE
            if pagar<>0 then
                begin
                    if (dolar=0) and (persen=0) then    // input desimal
                        begin
                            imm:=copy(operand1,2,pjpg);
                            destobin(imm,bin,e);

```

```

        pjg_bin:=length(bin);
        if pjg_bin>8 then begin showmessage('Register A hanya 8
bit');e:='1';end
    else
        begin
            hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
            hsl:=hasil1+hasil2;
            bhs_msn:=dtbase('ANDA','IMM');
            bhs_msn:=bhs_msn+' '+hsl;
            p_and8bit(RA,hsl,hasil,z);
        end;
    end
else if (dolar<>0) then // input heksa
    begin
        imm:=copy(operand1,3,2);
        bhs_msn:=dtbase('ANDA','IMM');
        bhs_msn:=bhs_msn+' '+imm;
        pjg_hex:=length(imm);
        if pjg_hex<>2 then begin showmessage('Register A hanya 8
bit');e:='1';end
    else
        begin
            p_and8bit(RA,imm,hasil,z);
        end;
    end
else if (persen<>0) then // input biner
    begin
        imm:=copy(operand1,3,8); // mengambil biner
        pjg_bin:=length(imm); // panjang biner
        if pjg_bin<>8 then begin showmessage('Register A hanya 8
bit');e:='1';end
    else
        begin
            hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
            bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
            hsl:=hasil1+hasil2;
            bhs_msn:=dtbase('ANDA','IMM');
            bhs_msn:=bhs_msn+' '+hsl;
            p_and8bit(RA,hsl,hasil,z);
        end;
    end;
end

// PENGALAMATAN DIRECT & EXTENDED
else if (pagar=0) and (dolar<>0) and (operand2='') then
    begin
        // PENGALAMATAN DIRECT
        if pjg=3 then
            begin
                // mengambil 2 digit terakhir operand utk alamat data
                x:=copy(operand1,2,2);
                alamat:=angka(x);
                dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                p_and8bit(RA,dir,hasil,z);
                bhs_msn:=dtbase('ANDA','DIR');
                bhs_msn:=bhs_msn+' '+dir;
            end

            // PENGALAMATAN EXTENDED
        else if pjg=5 then
            begin
                // mengambil 4 digit terakhir operand utk alamat data
                x:=copy(operand1,2,4);
                alamat:=angka(x);
                dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                p_and8bit(RA,dir,hasil,z);
                bhs_msn:=dtbase('ANDA','EXT');
                bhs_msn:=bhs_msn+' '+dir;
            end;
        end;
    end
end

```

```

// PENGALAMATAN INDEXED
else if (pagar=0) and (dolar<>0) and (operand2<>'') then
begin
if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
bil:=copy(operand1,2,2);
op2:=huruf(copy(operand2,1,1));
if op2='X' then
begin
buffint:=strtoint(RX)+strtoint(bil); // alamatnya = data
di Reg X + bil
x:=inttostr(buffint);
alamat:=angka(x);
dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
p_and8bit(RA,dir,hasil,z);
bhs_msn:=dtbase('ANDA','IND X');
bhs_msn:=bhs_msn+' '+dir;
end
else if op2='Y' then
begin
buffint:=strtoint(RY)+strtoint(bil); // alamatnya = data
di Reg X + bil
x:=inttostr(buffint);
alamat:=angka(x);
dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
p_and8bit(RA,dir,hasil,z);
bhs_msn:=dtbase('ANDA','IND Y');
bhs_msn:=bhs_msn+' '+dir;
end
else begin ShowMessage('Tidak Ada Register Index Tersebut');e:='1';end
end;
end;

procedure ANDB(RB,RX,RY,operand1,operand2:string; var hasil:string; var z,e:char;
var bhs_msn:string);
var
imm,bin,dir,bil,x,op2:string;
pagar,dolar,persen,pjg,pjg_bin,pjg_hex,alamat:integer;
hasil1,hasil2,hsl,hsl1,hsl2:string;
begin
with form1 do
begin
pagar:=pos('#',operand1); dolar:=pos('$',operand1);
persen:=pos('%',operand1);
pjg:=length(operand1);

// PENGALAMATAN IMMEDIATE
if pagar<>0 then
begin
if (dolar=0) and (persen=0) then // input desimal
begin
imm:=copy(operand1,2,pjg);
destobin(imm,bin,e);
pjg_bin:=length(bin);
if pjg_bin>8 then begin showmessage('Register B hanya 8
bit');e:='1';end
else
begin
hsl1:=copy(bin,1,4);hsl2:=copy(bin,5,8);
bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
hsl:=hasil1+hasil2;
bhs_msn:=dtbase('ANDB','IMM');
bhs_msn:=bhs_msn+' '+hsl;
p_and8bit(RB,hsl,hasil,z);
end;
end
else if (dolar<>0) then // input heksa
begin
imm:=copy(operand1,3,2);

```

```

        bhs_msn:=dtbase('ANDB','IMM');
        bhs_msn:=bhs_msn+' '+imm;
        pjg_hex:=length(imm);
        if pjg_hex<>2 then begin showmessage('Register B hanya 8
bit');e:='1';end
    else
        begin
            p_and8bit(RB,imm,hasil,z);
        end;
    end
    else if (persen<>0) then // input biner
        begin
            imm:=copy(operand1,3,8); // mengambil biner
            pjg_bin:=length(imm); // panjang biner
            if pjg_bin<>8 then begin showmessage('Register B hanya 8
bit');e:='1';end
        else
            begin
                hsl1:=copy(imm,1,4);hsl2:=copy(imm,5,8);
                bintohehex(hsl1,hasil1);bintohehex(hsl2,hasil2);
                hsl:=hasil1+hasil2;
                bhs_msn:=dtbase('ANDB','IMM');
                bhs_msn:=bhs_msn+' '+hsl;
                p_and8bit(RB,hsl,hasil,z);
            end;
        end;
    end

    // PENGALAMATAN DIRECT & EXTENDED
    else if (pagar=0) and (dolar<>0) and (operand2='') then
        begin
            // PENGALAMATAN DIRECT
            if pjg=3 then
                begin
                    // mengambil 2 digit terakhir operand utk alamat data
                    x:=copy(operand1,2,2);
                    alamat:=angka(x);
                    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                    p_and8bit(RB,dir,hasil,z);
                    bhs_msn:=dtbase('ANDB','DIR');
                    bhs_msn:=bhs_msn+' '+dir;
                end

            // PENGALAMATAN EXTENDED
            else if pjg=5 then
                begin
                    // mengambil 4 digit terakhir operand utk alamat data
                    x:=copy(operand1,2,4);
                    alamat:=angka(x);
                    dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb
                    p_and8bit(RB,dir,hasil,z);
                    bhs_msn:=dtbase('ANDB','EXT');
                    bhs_msn:=bhs_msn+' '+dir;
                end;
            end

            // PENGALAMATAN INDEXED
            else if (pagar=0) and (dolar<>0) and (operand2<>'') then
                begin
                    if (dolar<>0) and (persen=0) then // input bilangan
heksadesimal
                        bil:=copy(operand1,2,2);
                        op2:=huruf(copy(operand2,1,1));
                        if op2='X' then
                            begin
                                buffint:=strtoint(RX)+strtoint(bil); // alamatnya = data
di Reg X + bil
                                x:=inttostr(buffint);
                                alamat:=angka(x);
                                dir:=copy(Memory.Lines[alamat],14,2); // mengambil data yg
ada dialamat tsb

```

```

        p_and8bit(RB,dir,hasil,z);
        bhs_msn:=dtbase('ANDB','IND X');
        bhs_msn:=bhs_msn+' '+dir;
    end
    else if op2='Y' then
    begin
        buffint:=strtoint(RY)+strtoint(bil);           // alamatnya =
data di Reg X + bil
        x:=inttostr(buffint);
        alamat:=angka(x);
        dir:=copy(Memory.Lines[alamat],14,2);         // mengambil data
yg ada dialamat tsb
        p_and8bit(RB,dir,hasil,z);
        bhs_msn:=dtbase('ANDB','IND Y');
        bhs_msn:=bhs_msn+' '+dir;
    end
    else begin ShowMessage('Tidak Ada Register Index
Tersebut');;e:='1';end;
    end;
end;

procedure kali(RA,RB:string; var output1,output2,bhs_msn:string; var c,e:char);
var
    x,y,pjg,pjg_bin:integer;
    x1,x2,y1,y2,hsl1,hsl2,hasil1,hasil2:string;
begin
    // UNTUK YANG RA
    x1:=copy(RA,1,1);buffstr1:=hextobin(x1);           // merubah heksa
ke biner
    x2:=copy(RA,2,1);buffstr2:=hextobin(x2);
    buffstr5:=buffstr1+buffstr2;
    bintodes(buffstr5,buffstr);
    x:=strtoint(buffstr);
    // UNTUK YANG RB
    y1:=copy(RB,1,1);buffstr1:=hextobin(y1);           // merubah heksa
ke biner
    y2:=copy(RB,2,1);buffstr2:=hextobin(y2);
    buffstr4:=buffstr1+buffstr2;
    bintodes(buffstr4,buffstr);
    y:=strtoint(buffstr);
    // PROSES MENGALIKANNYA
    buffint:=x*y;
    buffstr:=inttostr(buffint);
    destobin(buffstr,buffstr,e);
    pjg_bin:=length(buffstr);
    if pjg_bin<16 then
    begin
        repeat
            pjg:=length(buffstr);
            buffstr:='0'+buffstr;
        until pjg=15;
    end;
    if buffstr[1]='1' then c:='1'
    else c:='0';

    x1:=copy(buffstr,1,4);x2:=copy(buffstr,5,4);y1:=copy(buffstr,9,4);y2:=copy(buffstr
,13,4);
    bintohe(x1,hsl1);bintohe(x2,hsl2);bintohe(y1,hasil1);bintohe(y2,hasil2);
    output2:=hasil1+hasil2;
    output1:=hsl1+hsl2;
    bhs_msn:=dtbase('MUL','INH');
end;

procedure BCLR(operand1,operand2:string; var hasil,bhs_msn:string; var
posisi:integer;var e:char);
var x1,x2,y1,y2,y:string; alamat:integer;
begin
    with form1 do begin
        buffstr:=copy(operand1,1,1);
        if buffstr<>'$' then begin ShowMessage('Pengalamatannya Salah');e:='1'; end
        else
            begin

```

```

        buffstr1:=copy(operand1,2,2);
        alamat:=angka(buffstr1);
        buffstr2:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg ada
dialamat tsb
        buffint:=strtoint(operand2);
        x1:=copy(buffstr2,1,1);x2:=copy(buffstr2,2,1);
        y1:=hextobin(x1);y2:=hextobin(x2);
        y:=y1+y2;
        y[buffint]:='0';          // meng-NOL-kan bit yg
ditunjuk pd mask
        x1:=copy(y,1,4);x2:=copy(y,5,4);
        bintohe(x1,y1);bintohe(x2,y2);
        y:=y1+y2;
        hasil:=y; posisi:=alamat;
        bhs_msn:=dtbase('BCLR','DIR');
    end;
end;
end;

procedure BSET(operand1,operand2:string; var hasil,bhs_msn:string; var
posisi:integer;var e:char);
var x1,x2,y1,y2,y:string; alamat:integer;
begin
    with form1 do begin
        buffstr:=copy(operand1,1,1);
        if buffstr<>'$' then begin ShowMessage('Pengalamatannya Salah');e:='1'; end
        else
            begin
                buffstr1:=copy(operand1,2,2);
                alamat:=angka(buffstr1);
                buffstr2:=copy(Memory.Lines[alamat],14,2);          // mengambil data yg ada
dialamat tsb
                buffint:=strtoint(operand2);
                x1:=copy(buffstr2,1,1);x2:=copy(buffstr2,2,1);
                y1:=hextobin(x1);y2:=hextobin(x2);
                y:=y1+y2;
                y[buffint]:='1';          // meng-NOL-kan bit yg
ditunjuk pd mask
                x1:=copy(y,1,4);x2:=copy(y,5,4);
                bintohe(x1,y1);bintohe(x2,y2);
                y:=y1+y2;
                hasil:=y; posisi:=alamat;
                bhs_msn:=dtbase('BSET','DIR');
            end;
        end;
    end;
end;

procedure CBA(RA,RB:string; var RC,RZ:char; var bhs_msn:string);
var a,b:integer;
begin
    buffstr1:=hextobin(copy(RA,1,1)); buffstr2:=hextobin(copy(RA,2,1));
    buffstr3:=hextobin(copy(RB,1,1));buffstr4:=hextobin(copy(RB,2,1));
    bintodes(buffstr1+buffstr2,RA); bintodes(buffstr3+buffstr4,RB);
    a:=strtoint(RA); b:=strtoint(RB);
    bhs_msn:=dtbase('CBA','INH');
    if a=b then
        begin RZ:='1';RC:='0'; end
    else
        begin RZ:='0';RC:='1'; end;
end;

procedure simpanA(RA,RX,RY,operand1,operand2:string; var alamat:integer; var
bhs_msn:string; var e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg:integer;
begin
    with form1 do begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
    persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN DIRECT & EXTENDED
        if (pagar=0) and (dolar<>0) and (operand2='') then

```

```

begin
    // PENGALAMATAN DIRECT
    if pjg=3 then
        begin
            // mengambil 2 digit terakhir operand utk alamat data
            x:=copy(operand1,2,2); alamat:=angka(x);
            bhs_msn:=dtbase('STAA','DIR'); bhs_msn:=bhs_msn+' '+dir;
        end;
    // PENGALAMATAN EXTENDED
    if pjg=5 then
        begin
            // mengambil 4 digit terakhir operand untuk alamat data
            x:=copy(operand1,2,4); alamat:=angka(x);
            bhs_msn:=dtbase('STAA','EXT'); bhs_msn:=bhs_msn+' '+dir;
        end;
    end
    // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
        begin
            if (dolar<>0) and (persen=0) then // input bilangan
                heksadesimal
                bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
                if op2='X' then
                    begin
                        buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
                        data di Reg X + bil
                        x:=inttostr(buffint); alamat:=angka(x);
                        bhs_msn:=dtbase('STAA','IND X'); bhs_msn:=bhs_msn+' '+dir;
                    end
                else if op2='Y' then
                    begin
                        buffint:=strtoint(RY)+strtoint(bil); // alamatnya =
                        data di Reg X + bil
                        x:=inttostr(buffint); alamat:=angka(x);
                        bhs_msn:=dtbase('STAA','IND Y'); bhs_msn:=bhs_msn+' '+dir;
                    end
                else begin ShowMessage('Tidak Ada Register Index
                Tersebut');e:='1';end;
            end;
        end;
    end;

procedure simpanB(RB,RX,RY,operand1,operand2:string; var alamat:integer; var
bhs_msn:string; var e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg:integer;
begin
    with form1 do begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
        persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN DIRECT & EXTENDED
        if (pagar=0) and (dolar<>0) and (operand2='') then
            begin
                // PENGALAMATAN DIRECT
                if pjg=3 then
                    begin
                        // mengambil 2 digit terakhir operand utk alamat data
                        x:=copy(operand1,2,2); alamat:=angka(x);
                        bhs_msn:=dtbase('STAB','DIR'); bhs_msn:=bhs_msn+' '+dir;
                    end;
                // PENGALAMATAN EXTENDED
                if pjg=5 then
                    begin
                        // mengambil 4 digit terakhir operand untuk alamat data
                        x:=copy(operand1,2,4); alamat:=angka(x);
                        bhs_msn:=dtbase('STAB','EXT'); bhs_msn:=bhs_msn+' '+dir;
                    end;
                end
            end
        // PENGALAMATAN INDEXED
        else if (pagar=0) and (dolar<>0) and (operand2<>'') then
            begin

```

```

        if (dolar<>0) and (persen=0) then          // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
            begin
                buffint:=strtoint(RX)+strtoint(bil);          // alamatnya =
data di Reg X + bil
                x:=inttostr(buffint); alamat:=angka(x);
                bhs_msn:=dtbase('STAB','IND X'); bhs_msn:=bhs_msn+' '+dir;
            end
        else if op2='Y' then
            begin
                buffint:=strtoint(RY)+strtoint(bil);          // alamatnya =
data di Reg X + bil
                x:=inttostr(buffint); alamat:=angka(x);
                bhs_msn:=dtbase('STAB','IND Y'); bhs_msn:=bhs_msn+' '+dir;
            end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
        end;
    end;
end;

procedure simpanD(RX,RY,operand1,operand2:string; var alamat:integer; var
bhs_msn:string; var e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg:integer;
begin
    with form1 do begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
        persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN DIRECT & EXTENDED
        if (pagar=0) and (dolar<>0) and (operand2='') then
            begin
                // PENGALAMATAN DIRECT
                if pjg=3 then
                    begin
                        // mengambil 2 digit terakhir operand utk alamat data
                        x:=copy(operand1,2,2); alamat:=angka(x);
                        bhs_msn:=dtbase('STD','DIR'); bhs_msn:=bhs_msn+' '+dir;
                    end;
                // PENGALAMATAN EXTENDED
                if pjg=5 then
                    begin
                        // mengambil 4 digit terakhir operand untuk alamat data
                        x:=copy(operand1,2,4); alamat:=angka(x);
                        bhs_msn:=dtbase('STD','EXT'); bhs_msn:=bhs_msn+' '+dir;
                    end;
            end
            // PENGALAMATAN INDEXED
        else if (pagar=0) and (dolar<>0) and (operand2<>'') then
            begin
                if (dolar<>0) and (persen=0) then          // input bilangan
heksadesimal
                bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
                if op2='X' then
                    begin
                        buffint:=strtoint(RX)+strtoint(bil);          // alamatnya =
data di Reg X + bil
                        x:=inttostr(buffint); alamat:=angka(x);
                        bhs_msn:=dtbase('STD','IND X'); bhs_msn:=bhs_msn+' '+dir;
                    end
                else if op2='Y' then
                    begin
                        buffint:=strtoint(RY)+strtoint(bil);          // alamatnya =
data di Reg X + bil
                        x:=inttostr(buffint); alamat:=angka(x);
                        bhs_msn:=dtbase('STD','IND Y'); bhs_msn:=bhs_msn+' '+dir;
                    end
                else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
            end;
        end;
    end;
end;

```

```

        end;
    end;
end;

procedure simpanX(RX,RY,operand1,operand2:string; var alamat:integer; var
bhs_msn,low,high:string; var e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg:integer;
begin
    with form1 do begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
        persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN DIRECT & EXTENDED
        if (pagar=0) and (dolar<>0) and (operand2='') then
            begin
                // PENGALAMATAN DIRECT
                if pjg=3 then
                    begin
                        // mengambil 2 digit terakhir operand utk alamat data
                        x:=copy(operand1,2,2); alamat:=angka(x);
                        bhs_msn:=dtbase('STX','DIR'); bhs_msn:=bhs_msn+' '+dir;
                    end;
                // PENGALAMATAN EXTENDED
                if pjg=5 then
                    begin
                        // mengambil 4 digit terakhir operand untuk alamat data
                        x:=copy(operand1,2,4); alamat:=angka(x);
                        bhs_msn:=dtbase('STX','EXT'); bhs_msn:=bhs_msn+' '+dir;
                    end;
                end
                // PENGALAMATAN INDEXED
            else if (pagar=0) and (dolar<>0) and (operand2<>'') then
                begin
                    if (dolar<>0) and (persen=0) then // input bilangan
                        heksadesimal
                        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
                        if op2='X' then
                            begin
                                buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
                                data di Reg X + bil
                                x:=inttostr(buffint); alamat:=angka(x);
                                bhs_msn:=dtbase('STX','IND X'); bhs_msn:=bhs_msn+' '+dir;
                            end
                        else if op2='Y' then
                            begin
                                buffint:=strtoint(RY)+strtoint(bil); // alamatnya =
                                data di Reg X + bil
                                x:=inttostr(buffint); alamat:=angka(x);
                                bhs_msn:=dtbase('STX','IND Y'); bhs_msn:=bhs_msn+' '+dir;
                            end
                        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
                    end;
                    low:=copy(RX,1,2); high:=copy(RX,3,2);
                end;
            end;
        end;

procedure simpanY(RX,RY,operand1,operand2:string; var alamat:integer; var
bhs_msn,low,high:string; var e:char);
var
    dir,bil,x,op2:string;
    pagar,dolar,persen,pjg:integer;
begin
    with form1 do begin
        pagar:=pos('#',operand1); dolar:=pos('$',operand1);
        persen:=pos('%',operand1);
        pjg:=length(operand1);
        // PENGALAMATAN DIRECT & EXTENDED
        if (pagar=0) and (dolar<>0) and (operand2='') then
            begin
                // PENGALAMATAN DIRECT

```

```

        if pjpg=3 then
            begin
                // mengambil 2 digit terakhir operand utk alamat data
                x:=copy(operand1,2,2); alamat:=angka(x);
                bhs_msn:=dtbase('STY','DIR'); bhs_msn:=bhs_msn+' '+dir;
            end;
            // PENGALAMATAN EXTENDED
        if pjpg=5 then
            begin
                // mengambil 4 digit terakhir operand untuk alamat data
                x:=copy(operand1,2,4); alamat:=angka(x);
                bhs_msn:=dtbase('STY','EXT'); bhs_msn:=bhs_msn+' '+dir;
            end;
        end
        // PENGALAMATAN INDEXED
    else if (pagar=0) and (dolar<>0) and (operand2<>'') then
        begin
            if (dolar<>0) and (persen=0) then // input bilangan
                heksadesimal
                bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
                if op2='X' then
                    begin
                        buffint:=strtoint(RX)+strtoint(bil); // alamatnya =
                        data di Reg X + bil
                        x:=inttostr(buffint); alamat:=angka(x);
                        bhs_msn:=dtbase('STY','IND X'); bhs_msn:=bhs_msn+' '+dir;
                    end
                else if op2='Y' then
                    begin
                        buffint:=strtoint(RY)+strtoint(bil); // alamatnya =
                        data di Reg X + bil
                        x:=inttostr(buffint); alamat:=angka(x);
                        bhs_msn:=dtbase('STY','IND Y'); bhs_msn:=bhs_msn+' '+dir;
                    end
                else begin ShowMessage('Tidak Ada Register Index
                Tersebut');e:='1';end;
            end;
            low:=copy(RY,1,2); high:=copy(RY,3,2);
        end;
    end;

    procedure simpanSP(SP,RX,RY,operand1,operand2:string; var alamat:integer; var
    bhs_msn,low,high:string; var e:char);
    var
        dir,bil,x,op2:string;
        pagar,dolar,persen,pjpg:integer;
    begin
        with form1 do begin
            pagar:=pos('#',operand1); dolar:=pos('$',operand1);
            persen:=pos('%',operand1);
            pjpg:=length(operand1);
            // PENGALAMATAN DIRECT & EXTENDED
            if (pagar=0) and (dolar<>0) and (operand2='') then
                begin
                    // PENGALAMATAN DIRECT
                    if pjpg=3 then
                        begin
                            // mengambil 2 digit terakhir operand utk alamat data
                            x:=copy(operand1,2,2); alamat:=angka(x);
                            bhs_msn:=dtbase('STS','DIR'); bhs_msn:=bhs_msn+' '+dir;
                        end;
                    // PENGALAMATAN EXTENDED
                    if pjpg=5 then
                        begin
                            // mengambil 4 digit terakhir operand untuk alamat data
                            x:=copy(operand1,2,4); alamat:=angka(x);
                            bhs_msn:=dtbase('STS','EXT'); bhs_msn:=bhs_msn+' '+dir;
                        end;
                    end
                // PENGALAMATAN INDEXED
            else if (pagar=0) and (dolar<>0) and (operand2<>'') then
                begin

```

```

        if (dolar<>0) and (persen=0) then           // input bilangan
heksadesimal
        bil:=copy(operand1,2,2); op2:=huruf(copy(operand2,1,1));
        if op2='X' then
            begin
                buffint:=strtoint(RX)+strtoint(bil);           // alamatnya =
data di Reg X + bil
                x:=inttostr(buffint); alamat:=angka(x);
                bhs_msn:=dtbase('STS','IND X'); bhs_msn:=bhs_msn+' '+dir;
            end
        else if op2='Y' then
            begin
                buffint:=strtoint(RY)+strtoint(bil);           // alamatnya =
data di Reg X + bil
                x:=inttostr(buffint); alamat:=angka(x);
                bhs_msn:=dtbase('STS','IND Y'); bhs_msn:=bhs_msn+' '+dir;
            end
        else begin ShowMessage('Tidak Ada Register Index
Tersebut');e:='1';end;
            end;
            low:=copy(SP,1,2); high:=copy(SP,3,2);
        end;
    end;

procedure aaaaaaaaaa;
begin end;

// MAIN PROGRAM
procedure TForm1.EksekusiClick(Sender: TObject);
var
    Opcode,Operand1,Operand2:string;
    RA, RB, RX, RY, hasil, hasil1, hasil2, high, low, buff, mem:string;
    PtA, PtB, PtC, PtD:string;
    PC, SP, bhs, BARIS:string;
    RC, RZ, RS, RN, RH, RV, RIRQ, RXIRQ:char;
    j, i, urutan, posisi, alamat, loping, buf, pjg:integer;
    carry, zero, H, N, e:char;
    depan, tengah, belakang:string;
begin
    WITH FORM1 DO
    BEGIN
        Mnemonic.Clear;
        PC:=ProgramCounter.text; SP:=StackPointer.text;
        PtA:=PortA.text; PtB:=PortB.text; PtC:=PortC.text; PtD:=PortD.text;
        RA:=RegA.text; RB:=RegB.text; RX:=RegX.text; RY:=RegY.text;
        RC:=RegC.text[1]; RZ:=RegZ.text[1]; RS:=RegS.text[1]; RN:=RegN.text[1];
        RH:=RegH.text[1]; RV:=RegV.text[1]; RIRQ:=RegIRQ.text[1];
        RXIRQ:=RegXIRQ.text[1];

        buf:=0;           // buffer untuk menyimpan posisi memory pada saat
    ttt
        j:=Perintah.lines.count;           // jumlah baris perintah
        i:=0; urutan:=0; e:='0';           // nilai awal PC
        repeat
            AmbilStringPerintah(Perintah.lines[i], Opcode, Operand1, Operand2); e:='0';
            // tempat nulis perintah!!!!
            // menecetak memory
            if Opcode='ORG' then
                begin
                    Memory.Clear;
                    mem:=copy(Operand1,2,4);           // alamat awal program
                    for loping:=0 to 1000 do           // panjang alamat
                        begin
                            Memory.lines.Add( mem + ' | 00');
                            INCRX(mem, mem, bhs, e);
                        end;
                    SendMessage(Memory.Handle, WM_VSCROLL, SB_TOP, 0);
                    bhs:=''; Mnemonic.Lines.Append(bhs);
                end
            else if Opcode='ABA' then
                begin

```

```

        ABA(RA, RB, Operand1, hasil, zero, carry, H, bhs);
        Mnemonic.Lines.Append(bhs); // tampil nambahin dibawahnya
        RA:=hasil; RZ:=zero; RC:=carry; RH:=H;
    end
else if Opcode='ADDD' then
begin
    ADDD(RA, RB, RX, RY, Operand1, Operand2, hasil1, hasil2, zero, carry, H, e, bhs);
    Mnemonic.Lines.Append(bhs);
    RA:=hasil1; RB:=hasil2; RZ:=zero; RC:=carry; RH:=H;
end
else if Opcode='ADDA' then
begin
    ADDA(RA, RX, RY, Operand1, Operand2, hasil, zero, carry, H, e, bhs);
    Mnemonic.Lines.Append(bhs);
    RA:=hasil; RZ:=zero; RC:=carry; RH:=H;
end
else if Opcode='ADDB' then
begin
    ADDB(RB, RX, RY, Operand1, Operand2, hasil, zero, carry, H, e, bhs);
    Mnemonic.Lines.Append(bhs);
    RB:=hasil; RZ:=zero; RC:=carry; RH:=H;
end
else if Opcode='ADCA' then
begin
    ADCA(RA, RX, RY, RC, Operand1, Operand2, hasil, zero, carry, H, e, bhs);
    Mnemonic.Lines.Append(bhs);
    RA:=hasil; RZ:=zero; RC:=carry; RH:=H;
end
else if Opcode='ADCB' then
begin
    ADCB(RB, RX, RY, RC, Operand1, Operand2, hasil, zero, carry, H, e, bhs);
    Mnemonic.Lines.Append(bhs);
    RB:=hasil; RZ:=zero; RC:=carry; RH:=H;
end
else if Opcode='LDA' then
begin
    LDA(RA, RX, RY, Operand1, Operand2, hasil, bhs, e);
    Mnemonic.Lines.Append(bhs); RA:=hasil;
end
else if Opcode='LDB' then
begin
    LDB(RB, RX, RY, Operand1, Operand2, hasil, bhs, e);
    Mnemonic.Lines.Append(bhs); RB:=hasil;
end
else if Opcode='LDD' then
begin
    LDD(RX, RY, Operand1, Operand2, hasil1, hasil2, bhs, e);
    Mnemonic.Lines.Append(bhs); RA:=hasil1; RB:=hasil2;
end
else if Opcode='LDX' then
begin
    LDX(RX, RY, Operand1, Operand2, hasil, bhs, e);
    Mnemonic.Lines.Append(bhs); RX:=hasil;
end
else if Opcode='LDY' then
begin
    LDY(RX, RY, Operand1, Operand2, hasil, bhs, e);
    Mnemonic.Lines.Append(bhs); RY:=hasil;
end
else if Opcode='LDS' then
begin
    LDS(RX, RY, Operand1, Operand2, hasil, bhs, e);
    Mnemonic.Lines.Append(bhs); SP:=hasil;
end
else if Opcode='DECA' then
begin
    DCRA(RA, hasil, bhs, N, e, zero);
    Mnemonic.Lines.Append(bhs); RA:=hasil; RZ:=zero; RN:=N;
end
else if Opcode='DECB' then
begin
    DCRB(RB, hasil, bhs, N, e, zero);
    Mnemonic.Lines.Append(bhs); RB:=hasil; RZ:=zero; RN:=N;
end

```

```

end
else if Opcode='DEX' then
begin
DCRX(RX,hasil,bhs,N,e,zero);
Mnemonic.Lines.Append(bhs); RX:=hasil; RZ:=zero; RN:=N;
end
else if Opcode='DEY' then
begin
DCRY(RY,hasil,bhs,N,e,zero);
Mnemonic.Lines.Append(bhs); RY:=hasil; RZ:=zero; RN:=N;
end
else if Opcode='DES' then
begin
DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil; RZ:=zero; RN:=N;
Mnemonic.Lines.Append(bhs);
end
else if Opcode='ANDA' then
begin
ANDA(RA,RX,RY,Operand1,Operand2,hasil,zero,e,bhs);
Mnemonic.Lines.Append(bhs); RA:=hasil; RZ:=zero;
end
else if Opcode='ANDB' then
begin
ANDB(RB,RX,RY,Operand1,Operand2,hasil,zero,e,bhs);
Mnemonic.Lines.Append(bhs); RB:=hasil; RZ:=zero;
end
else if Opcode='CLRA' then
begin
RA:='00'; bhs:=dtbase('CLRA','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='CLRB' then
begin
RA:='00'; bhs:=dtbase('CLRB','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='INCA' then
begin
INCRA(RA,hasil,bhs,e); Mnemonic.Lines.Append(bhs); RA:=hasil;
end
else if Opcode='INCB' then
begin
INCRB(RB,hasil,bhs,e); Mnemonic.Lines.Append(bhs); RB:=hasil;
end
else if Opcode='INX' then
begin
INCRX(RX,hasil,bhs,e); Mnemonic.Lines.Append(bhs); RX:=hasil;
end
else if Opcode='INY' then
begin
INCRY(RY,hasil,bhs,e); Mnemonic.Lines.Append(bhs); RY:=hasil;
end
else if Opcode='INS' then
begin
INCRY(SP,hasil,bhs,e); Mnemonic.Lines.Append(bhs); SP:=hasil;
end
else if Opcode='MUL' then
begin
kali(RA,RB,hasil1,hasil2,bhs,carry,e);
Mnemonic.Lines.Append(bhs); RA:=hasil1;RB:=hasil2;RC:=carry;
end
else if Opcode='BCLR' then
begin
BCLR(Operand1,Operand2,hasil,bhs,posisi,e); Mnemonic.Lines.Append(bhs);
buffstr:=copy(Memory.Lines[posisi],1,13);
Memory.Lines[posisi]:=buffstr+hasil;
end
else if Opcode='BSET' then
begin
BSET(Operand1,Operand2,hasil,bhs,posisi,e); Mnemonic.Lines.Append(bhs);
buffstr:=copy(Memory.Lines[posisi],1,13);
Memory.Lines[posisi]:=buffstr+hasil;
end
else if Opcode='ASLA' then
begin

```

```

        geserkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e); RA:=hasil; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='ASLB' then
    begin
        geserkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e); RB:=hasil; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='ASLD' then
    begin
        geserkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e);
        buffstr1:=copy(hasil,1,2); buffstr2:=copy(hasil,3,2);
        RA:=buffstr1; RB:=buffstr2; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='ASRA' then
    begin
        geserkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
        RA:=hasil; RC:=carry; RZ:=zero; Mnemonic.Lines.Append(bhs);
    end
else if Opcode='ASRB' then
    begin
        geserkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
        RB:=hasil; RC:=carry; RZ:=zero; Mnemonic.Lines.Append(bhs);
    end
else if Opcode='LSLA' then
    begin
        geserkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e); RA:=hasil; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='LSLB' then
    begin
        geserkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e); RB:=hasil; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='LSLD' then
    begin
        geserkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e);
        buffstr1:=copy(hasil,1,2); buffstr2:=copy(hasil,3,2);
        RA:=buffstr1; RB:=buffstr2; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='LSRA' then
    begin
        geserkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
        RA:=hasil; RC:=carry; RZ:=zero; Mnemonic.Lines.Append(bhs);
    end
else if Opcode='LSRB' then
    begin
        geserkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
        RB:=hasil; RC:=carry; RZ:=zero; Mnemonic.Lines.Append(bhs);
    end
else if Opcode='LSRD' then
    begin
        geserkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
        buffstr1:=copy(hasil,1,2); buffstr2:=copy(hasil,3,2);
        RA:=buffstr1; RB:=buffstr2; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='ROLA' then
    begin
        puterkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e); RA:=hasil; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='ROLB' then
    begin
        puterkiri(Opcode,RA,RB,RC,hasil,bhs,carry,e); RB:=hasil; RC:=carry;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='RORA' then
    begin
        puterkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
        RA:=hasil; RC:=carry; RZ:=zero; Mnemonic.Lines.Append(bhs);
    end

```

```

end
else if Opcode='RORB' then
begin
puterkanan(Opcode,RA,RB,RC,hasil,bhs,carry,zero,e);
RB:=hasil; RC:=carry; RZ:=zero; Mnemonic.Lines.Append(bhs);
end
else if Opcode='SUBA' then
begin
SUBA(RA,RX,RY,Operand1,Operand2,hasil,zero,e,bhs);
Mnemonic.Lines.Append(bhs); RA:=hasil; RZ:=zero;
end
else if Opcode='SUBB' then
begin
SUBB(RB,RX,RY,Operand1,Operand2,hasil,zero,e,bhs);
Mnemonic.Lines.Append(bhs); RB:=hasil; RZ:=zero;
end
else if Opcode='SBA' then
begin
SBA(RA,RB,hasil,bhs,zero,carry,e); RA:=hasil; RZ:=zero;
Mnemonic.Lines.Append(bhs);
end
else if Opcode='CBA' then
begin
CBA(RA,RB,carry,zero,bhs);
Mnemonic.Lines.Append(bhs); RC:=carry; RZ:=zero;
end
else if Opcode='CLC' then
begin RC:='0';bhs:=dtbase('CLC','INH'); Mnemonic.Lines.Append(bhs); end
else if Opcode='CLV' then
begin RV:='0';bhs:=dtbase('CLV','INH'); Mnemonic.Lines.Append(bhs); end
else if Opcode='CLI' then
begin RXIRQ:='0';bhs:=dtbase('CLV','INH'); Mnemonic.Lines.Append(bhs); end
else if Opcode='SEC' then
begin RC:='1';bhs:=dtbase('SEC','INH'); Mnemonic.Lines.Append(bhs); end
else if Opcode='SEV' then
begin RV:='1';bhs:=dtbase('SEV','INH'); Mnemonic.Lines.Append(bhs); end
else if Opcode='SEI' then
begin RXIRQ:='1';bhs:=dtbase('SEI','INH'); Mnemonic.Lines.Append(bhs); end
else if Opcode='TAB' then
begin RB:=RA;bhs:=dtbase('TAB','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='TBA' then
begin RA:=RB;bhs:=dtbase('TBA','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='TSX' then
begin RX:=SP;bhs:=dtbase('TSX','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='TSY' then
begin RY:=SP;bhs:=dtbase('TSY','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='TXS' then
begin SP:=RX;bhs:=dtbase('TXS','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='TYS' then
begin SP:=RY;bhs:=dtbase('TYS','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='XGDX' then
begin
buffstr1:=copy(RX,1,2); buffstr2:=copy(RX,3,2);
RX:=RA+RB; RA:=buffstr1; RB:=buffstr2;
bhs:=dtbase('XGDX','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='XGDY' then
begin
buffstr1:=copy(RY,1,2); buffstr2:=copy(RY,3,2);
RY:=RA+RB; RA:=buffstr1; RB:=buffstr2;
bhs:=dtbase('XGDY','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='TSTA' then
begin
if RA='00' then RZ:='1' else RZ:='0';
bhs:=dtbase('TSTA','INH'); Mnemonic.Lines.Append(bhs);
end
end

```

```

else if Opcode='TSTB' then
begin
    if RB='00' then RZ:='1' else RZ:='0';
    bhs:=dtbase('TSTB','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PSHA' then
begin
    buffint:=angka(SP); buffstr:=inttostr(buffint);
    Memory.Lines[buffint]:=SP+'    '+RA;
    bhs:=dtbase('PSHA','INH'); Mnemonic.Lines.Append(bhs);
    DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil; RZ:=zero; RN:=N;
end
else if Opcode='PSHB' then
begin
    buffint:=angka(SP); buffstr:=inttostr(buffint);
    Memory.Lines[buffint]:=SP+'    '+RB;
    DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil; RZ:=zero; RN:=N;
    bhs:=dtbase('PSHB','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PSHX' then
begin
    buffint:=angka(SP); buffstr:=inttostr(buffint);
    high:=copy(RX,1,2); low:=copy(RX,3,2);
    Memory.Lines[buffint]:=SP+'    '+low;
    DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil;
    buffint:=angka(SP);
    Memory.Lines[buffint]:=SP+'    '+high;
    DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil; RZ:=zero; RN:=N;
    bhs:=dtbase('PSHX','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PSHY' then
begin
    buffint:=angka(SP); buffstr:=inttostr(buffint);
    high:=copy(RY,1,2); low:=copy(RY,3,2);
    Memory.Lines[buffint]:=SP+'    '+low;
    DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil;
    buffint:=angka(SP);
    Memory.Lines[buffint]:=SP+'    '+high;
    DCSP(SP,hasil,bhs,N,e,zero); SP:=hasil; RZ:=zero; RN:=N;
    bhs:=dtbase('PSHY','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PULA' then
begin
    INCRY(SP,hasil,bhs,e);
    buffint:=angka(hasil); RA:=copy(Memory.Lines[buffint],14,2);
    bhs:=dtbase('PULA','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PULB' then
begin
    INCRY(SP,hasil,bhs,e);
    buffint:=angka(hasil); RB:=copy(Memory.Lines[buffint],14,2);
    bhs:=dtbase('PULB','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PULX' then
begin
    INCRY(SP,hasil,bhs,e); SP:=hasil; buffint:=angka(SP);
    high:=copy(Memory.Lines[buffint],14,2);
    INCRY(SP,hasil,bhs,e); SP:=hasil; buffint:=angka(SP);
    low:=copy(Memory.Lines[buffint],14,2);
    RX:=high+low;
    bhs:=dtbase('PULX','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='PULY' then
begin
    INCRY(SP,hasil,bhs,e); SP:=hasil; buffint:=angka(SP);
    high:=copy(Memory.Lines[buffint],14,2);
    INCRY(SP,hasil,bhs,e); SP:=hasil; buffint:=angka(SP);
    low:=copy(Memory.Lines[buffint],14,2);
    RY:=high+low;
    bhs:=dtbase('PULY','INH'); Mnemonic.Lines.Append(bhs);
end
else if Opcode='STAA' then
begin

```

```

        simpanA(RA,RX,RY,Operand1,Operand2,alamat,bhs,e);
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+RA;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='STAB' then
    begin
        simpanB(RB,RX,RY,Operand1,Operand2,alamat,bhs,e);
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+RB;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='STD' then
    begin
        simpanD(RX,RY,Operand1,Operand2,alamat,bhs,e);
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+RB; alamat:=alamat+1;
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+RA;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='STX' then
    begin
        simpanX(RX,RY,Operand1,Operand2,alamat,bhs,low,high,e);
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+high; alamat:=alamat+1;
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+low;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='STY' then
    begin
        simpanY(RX,RY,Operand1,Operand2,alamat,bhs,low,high,e);
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+high; alamat:=alamat+1;
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+low;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='STS' then
    begin
        simpanSP(SP,RX,RY,Operand1,Operand2,alamat,bhs,low,high,e);
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+high; alamat:=alamat+1;
        buff:=copy(Memory.Lines[alamat],1,4);
        Memory.Lines[alamat]:=buff+'      |      '+low;
        Mnemonic.Lines.Append(bhs);
    end
else if Opcode='BCC' then
    begin
        if RC='0' then
            begin
                Mnemonic.Lines.Delete(i);
                buffstr1:=copy(Operand1,2,1);buffstr2:=copy(Operand1,3,1);
                buffstr1:=hextobin(buffstr1);buffstr2:=hextobin(buffstr2);
                bintodes(buffstr1,buffstr1);bintodes(buffstr2,buffstr2);
                buffstr:=buffstr1+buffstr2;i:=strtoint(buffstr);
                dec(i);;dec(i);
            end
        else if RC='1' then
            begin
                Mnemonic.Lines.Delete(i);
                i:=urutan;
                bhs:=dtbase('BCC','REL');Mnemonic.Lines.Append(bhs);
            end;
        end;
    end;
{procedure BCC(Operand1:string;var RC:char; var posskr,poscbg:integer
;var selisih,pengulangan:integer; var bhs_msn:string; var e:char);
begin
    if RC='0' then
        begin
            hex1:=copy(Operand1,2,1);hex2:=copy(Operand1,3,1);
            buffstr1:=hextobin(buffstr1);buffstr2:=hextobin(buffstr2);

```

```

        bintodes(buffstr1,buffstr1);bintodes(buffstr2,buffstr2);
        buffstr:=buffstr1+buffstr2;i:=strtoint(buffstr);
        dec(i);dec(i);
    end
end; }

// PROGRAM UNTUK MENYIMPAN BHS MSN KE MEMORY
AmbilStringMnemonic(Mnemonic.Lines[i],depan,tengah,belakang);
if depan='' then begin buf:=buf-1; end
else if (depan<>'') and (tengah='') and (belakang='') then
begin
    buff:=copy(Memory.Lines[buf],1,13);
    Memory.Lines[buf]:=buff+depan;
end
else if (depan<>'') and (tengah<>'') and (belakang='') then
begin
    buff:=copy(Memory.Lines[buf],1,13);
    Memory.Lines[buf]:=buff+depan; buf:=buf+1;
    pjg:=length(tengah);
    if pjg=4 then
    begin
        buffstr1:=copy(tengah,1,2); buffstr2:=copy(tengah,3,2);
        buff:=copy(Memory.Lines[buf],1,13);
        Memory.Lines[buf]:=buff+buffstr1; buf:=buf+1;
        buff:=copy(Memory.Lines[buf],1,13);
        Memory.Lines[buf]:=buff+buffstr2;
    end
    else
    begin
        buff:=copy(Memory.Lines[buf],1,13);
        Memory.Lines[buf]:=buff+tengah;
    end;
end
else if (depan<>'') and (tengah<>'') and (belakang<>'') then
begin
    buff:=copy(Memory.Lines[buf],1,13);
    Memory.Lines[buf]:=buff+depan; buf:=buf+1;
    buff:=copy(Memory.Lines[buf],1,13);
    Memory.Lines[buf]:=buff+tengah; buf:=buf+2;
    buff:=copy(Memory.Lines[buf],1,13);
    Memory.Lines[buf]:=buff+belakang;
end;
depan:='';tengah:='';belakang:='';

// PENGECEKAN ERROR
BARIS:=INTTOSTR(i+1);
IF e='1' THEN
begin
    ShowMessage('ERROR PADA BARIS KE'+ ' '+BARIS);
    PC:='0000'; SP:='0000';
    PtA:='0000'; PtB:='0000'; PtC:='0000'; PtD:='0000';
    RA:='00'; RB:='00'; RX:='0000'; RY:='0000';
    RC:='0'; RZ:='0'; RS:='0'; RN:='0';
    RH:='0'; RV:='0'; RIRQ:='0'; RXIRQ:='0';
end;

inc(urutan); inc(i); inc(buf);

// program untuk menghasilkan tampilan
buffstr1:=inttostr(urutan); destobin(buffstr1,buffstr,e);
buffint:=length(buffstr);
if buffint=16 then
begin
    buffstr1:=copy(buffstr,1,4); bintohehex(buffstr1,buffstr1);
    buffstr2:=copy(buffstr,5,4); bintohehex(buffstr2,buffstr2);
    buffstr3:=copy(buffstr,9,4); bintohehex(buffstr3,buffstr3);
    buffstr4:=copy(buffstr,13,4); bintohehex(buffstr4,buffstr4);
end
else if buffint<16 then
begin
    buffstr:=jadiin16bit(buffstr);
    buffstr1:=copy(buffstr,1,4); bintohehex(buffstr1,buffstr1);

```

```

        buffstr2:=copy(buffstr,5,4); bintohehex(buffstr2,buffstr2);
        buffstr3:=copy(buffstr,9,4); bintohehex(buffstr3,buffstr3);
        buffstr4:=copy(buffstr,13,4); bintohehex(buffstr4,buffstr4);
    end;
    hasil:=buffstr1+buffstr2+buffstr3+buffstr4;
    PC:=hasil;
    ProgramCounter.text:=PC; StackPointer.text:=SP;
    PortA.text:=PtA; PortB.text:=PtB; PortC.text:=PtC; PortD.text:=PtD;
    RegA.text:=RA; RegB.text:=RB; RegX.text:=RX; RegY.text:=RY;
    RegC.text:=RC; RegZ.text:=RZ; RegS.text:=RS; RegN.text:=RN;
    RegH.text:=RH; RegV.text:=RV;
    RegIRQ.text:=RIRQ; RegXIRQ.text:=RXIRQ;
until i>j-1;
END;
end;

// CLOSE
procedure TForm1.TutupClick(Sender: TObject);
begin
    Close;
end;

// FORM
procedure TForm1.FormActivate(Sender: TObject);
begin
    // mengosongkan semua memo
    Perintah.clear;Memory.clear;Mnemonic.clear;
    // KONDISI DEFAULT
    RegA.text:='00';RegB.text:='00';
    RegX.text:='0000';RegY.text:='0000';
    RegC.text:='0'; RegZ.text:='0'; RegS.text:='0'; RegN.text:='0';
    RegH.text:='0'; RegV.text:='0';
    RegIRQ.text:='0'; RegXIRQ.text:='0';
    ProgramCounter.text:='0000'; StackPointer.text:='0000';
    PortA.text:='00'; PortB.text:='00'; PortC.text:='00'; PortD.text:='00';

end;

procedure TForm1.FormCreate(Sender: TObject);
begin
    table1.open;
end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
    table1.close;
end;

procedure TForm1.KondisiAwalClick(Sender: TObject);
begin
    // KONDISI DEFAULT
    RegA.text:='00';RegB.text:='00';
    RegX.text:='0000';RegY.text:='0000';
    RegC.text:='0'; RegZ.text:='0'; RegS.text:='0'; RegN.text:='0';
    RegH.text:='0'; RegV.text:='0';
    RegIRQ.text:='0'; RegXIRQ.text:='0';
    ProgramCounter.text:='0000'; StackPointer.text:='0000';
    PortA.text:='00'; PortB.text:='00'; PortC.text:='00'; PortD.text:='00';

END;
end.

```

LAMPIRAN B
TABEL BAHASA MESIN

No	Opcode	INH	IMM	DIRECT	EXTENDED	INDEXED X	INDEXED Y	RELATIF
1	ABA	1B						
2	ABX	3A						
3	ABY	18 3A						
4	ADCA		89	99	B9	A9	18 A9	
5	ADCB		C9	D9	F9	E9	18 E9	
6	ADDA		8B	9B	BB	AB	18 AB	
7	ADDB		CB	DB	FB	EB	18 EB	
8	ADDD		C3	D3	F3	E3	18 E3	
9	ANDA		84	94	B4	A4	18 A4	
10	ANDB		C4	D4	F4	E4	18 E4	
11	ASLA	48						
12	ASLB	58			78	68	18 68	
13	ASLD	5						
14	ASRA	47						
15	ASRB	57			77	67	18 67	
16	BCC							24
17	BCLR			15		1D	18 1D	
18	BCS							25
19	BEQ							27
20	BGE							2C
21	BGT							2E
22	BHI							22
23	BHS							24
24	BITA		85	95	B5	A5	18 A5	
25	BITB		C5	D5	F5	E5	18 E5	
26	BLE							2F
27	BLO							25
28	BLS							23
29	BLT							2D
30	BMI							2B
31	BNE							26
32	BPL							2A
33	BRA							20
34	BRCLR			13		1F	18 1F	
35	BRN							21
36	BRSET			12		1E	18 1E	
37	BSET			14		1C	18 1C	
38	BSR							8D
39	BVC							28
40	BVS							29
41	CBA	11						
42	CLC	0C						
43	CLI							0E
44	CLR				7F	6F	18 6F	
45	CLRA	4F						
46	CLRB	5F						
47	CLV	0A						

No	Opcod e	INH	IMM	DIREC T	EXTENDE D	INDEXED X	INDEXED Y	RELATI F
48	CMP		81	91	B1	A1	18 A1	
49	COM	43			73	63	18 63	
50	CPD		1A 83	1A 93	1A B3	1A A3	CD A3	
51	CPX		8C	9C	BC	AC	CD AC	
52	CPY		18 8C	18 9C	18 BC	1A AC	18 AC	
53	DAA	19						
54	DEC				7A	6A	18 6A	
55	DECA	4A						
56	DECB	5A						
57	DES	34						
58	DEX	9						
59	DEY	18 09						
60	EOR		88	98	B8	A8	18 A8	
61	FDIV	3						
62	IDIV	2						
63	INC				7C	6C	18 6C	
64	INCA	4C						
65	INCB	5C						
66	INS	31						
67	INX	8						
68	INY	18 08						
69	JMP				7E	6E	18 6E	
70	JSR			9D	BD	AD	18 AD	
71	LDA		86	96	B6	A6	18 A6	
72	LDB		C6	D6	F6	E6	18 E6	
73	LDD		CC	DC	FC	EC	18 EC	
74	LDS		8E	9E	BE	AE	18 AE	
75	LDX		CE	DE	FE	EE	CD EE	
76	LDY		18 CE	18 DE	18 FE	1A EE	18 EE	
77	LSL	48			78	68	18 68	
78	LSLD	5						
79	LSR	44			74	64	18 64	
80	LSRD	4						
81	MUL	3D						
82	NEG	40			70	60	18 60	
83	NOP	1						
84	ORA		8A	9A	BA	AA	18 AA	
85	ORB		CA	DA	FA	EA	18 EA	
86	PSHA	36						
87	PSHB	37						
88	PSHX	3C						
89	PSHY	18 3C						
90	PULA	32						
No	Opcod e	INH	IMM	DIREC T	EXTENDE D	INDEXED X	INDEXED Y	RELATI F
91	PULB	33						

No	Opcode	INH	IMM	DIRECT	EXTENDED	INDEXED X	INDEXED Y	RELATIF
92	PULX	38						
93	PULY	18 38						
94	ROL				79	69	18 69	
95	ROLA	49						
96	ROLB	59						
97	ROR				76	66	18 66	
98	RORA	46						
99	RORB	56						
100	RTI	3B						
101	RTS	39						
102	SBA	10						
103	SBC		82	92	B2	A2	18 A2	
104	SEC	0D						
105	SEI	0F						
106	SEV	0B						
107	STAA			97	B7	A7	18 A7	
108	STAB			D7	F7	E7	18 E7	
109	STB			D7	F7	E7	18 E7	
110	STD			DD	FD	ED	18 ED	
111	STOP	CF						
112	STS			9F	BF	AF	18 AF	
113	STX			DF	FF	EF	CD EF	
114	STY			18 DF	18 FF	1A EF	18 EF	
115	SUBA		80	90	B0	A0	18 A0	
116	SUBB		C0	D0	F0	E0	18 E0	
117	SUBD		83	93	B3	A3	18 A3	
118	SWI	3F						
119	TAB	16						
120	TAP	6						
121	TBA	17						
122	TEST	0						
123	TPA	7						
124	TST				7D	6D	18 6D	
125	TSTA	4D						
126	TSTB	5D						
127	TSX	30						
128	TSY	18 30						
129	TXS	35						
130	TYS	18 35						
131	WAI	3E						
132	XGDX	8F						
133	XGDY	18 8F						