

Introdução a VHDL

Comandos de Repetição:

[Label]: For **parâmetro** in **faixa**
Loop
 <**comandos seqüenciais**>
end Loop [Label];

Next: **pula a próx, iteração**

Exit: **sai do comando de
repetição**

Opção

Generate

<**comandos concorrentes**>

Ilustrações:

Ex: for I in 1 to 10 Loop
 A(i):=i*i;
end Loop Ex;

Ex1: For i in 0 to max_limit Loop
 If (done(i)=true) then Next;
 else done (i):=true;
 end if;
 q(i)<=a(i) and (b(i);
end Loop Ex1;

Introdução a VHDL

Comando de **repetição**:

```
[Label]: While expressão  
    Loop  
    <comandos sequenciais>  
End Loop [Label];
```

Exemplo:

```
Ex2: while index < 8 Loop  
    saída (index)<=entrada(index);  
    Index:=index + 1;  
end Loop Ex2;
```

Wait on (signal)

Wait until expressão booleana

Wait For expressão de tempo;

Process

-- exemplo de um FF-D

Begin

```
wait until clock='1' and clock 'event;  
q<=d;  
end Process;
```

Introdução a VHDL

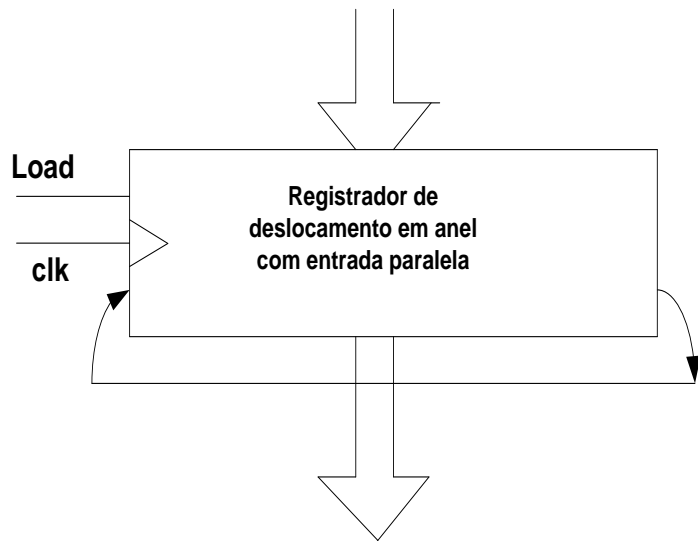
Exemplo FF-D com reset síncrono *Exemplo FF-D com reset assíncrono*

```
Process
Begin
  wait until clock ='1' and clock
    'event;
  If (reset='1' ) then q<='0';
    else q<=d;
  endif;
end Process;
```

```
Process
Begin
  If (reset='1' then q<='0';
    else if clock 'event and
      clock='1'
      then q<='d';
    end if;
  Wait on reset, clock;
end Process.
```

Introdução a VHDL

Exemplo:



Entity **rotate** is

Port (clk,rst,load: in bit; data: in bit_vector (0 to 7);
Q: out bit_vector (0 to 7));

End **rotate**.

Architecture **rotate1** of **rotate** is

Signal Qreg: bit_vector (0 to 7);

Begin

Process (rst, clk)

Begin

If rst='1' then Qreg<="00000000";

else if (clk='1' and clk 'event)

then if (load='1') then Qreg<=data;

else Qreg <=Qreg (1 to 7) & Qreg (0)

end if;

end if;

End Process;

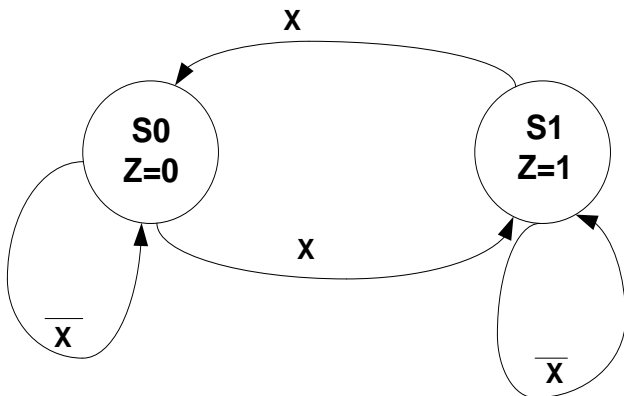
Q<=Qreg;

End **rotate1**;

Introdução a VHDL

Máquina de estado em
VHDL

Ex: Moore



Tipos:

a) Estrutural

equações Booleanas

Processo de sintetização

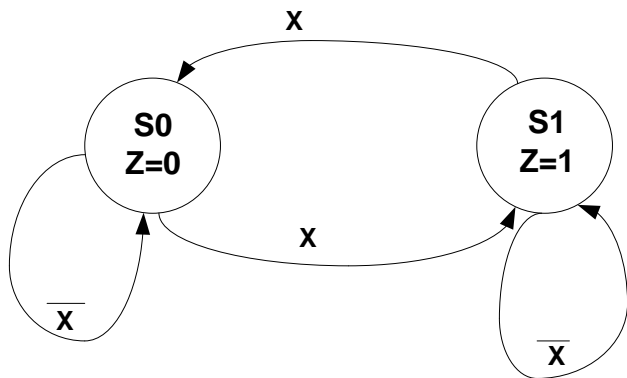
b) Comportamental

Descrição do digrama de
estado

Processo de sincronização

Introdução a VHDL

Exemplo: Moore

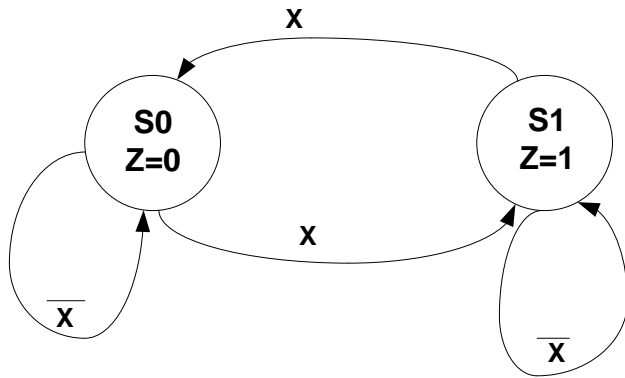


Entity **state_machine** is
Port (clk in bit; x: in bit; z: out bit);
End **state_machine**;

Architecture **Ex1** of **state_machine** is
Type **estados** is (S0,S1);
Signal est: **estados**:=S0;
Signal next_est: **estados**:=S0;
Begin - - **processo de sincronização**
Clkd: Process (clk)
Begin
If (clk 'event and clk='1')
then est<=next_est;
end if;
End Process **clkd**;

Introdução a VHDL

Exemplo: Moore



D_est: process (est,x)

Begin

case est is

when S0 => Z<=0;

if (x='1') then next_est<=S1;

else next_est<=S0;

end if;

when S1 => Z<=1;

if (x='1') then next_est<=S0;

else next_est<=S1;

end if;

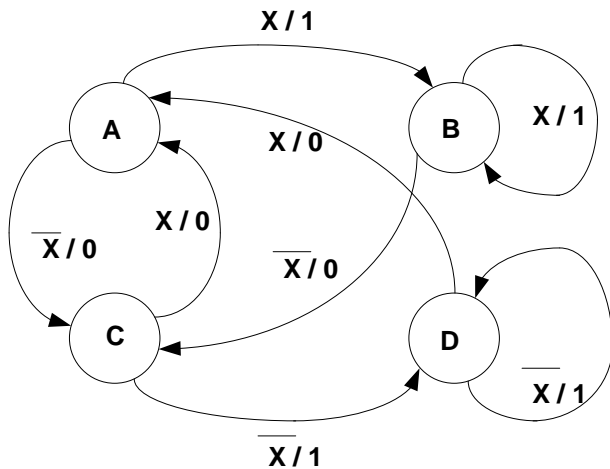
end case;

end process D_est;

end ex1;

Introdução a VHDL

Exemplo: Mealy

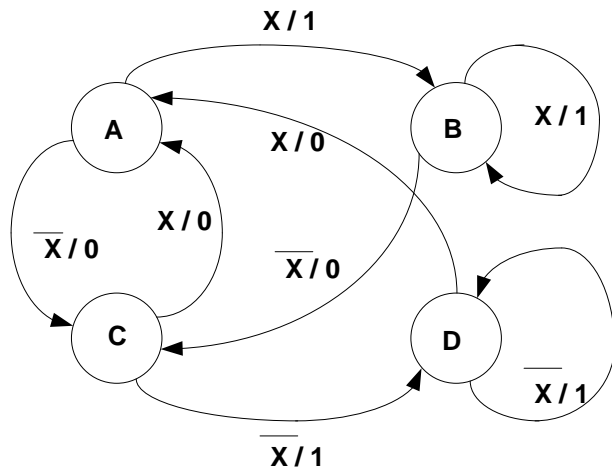


```
entity Mealy_machine is  
port (clk: in bit; x: in bit; z: out  
      bit);  
end Mealy_machine;
```

```
architecture Ex2 of  
  Mealy_machine is  
    type estados is (A,B,C,D);  
    signal est: estados := A;  
  Begin
```


Introdução a VHDL

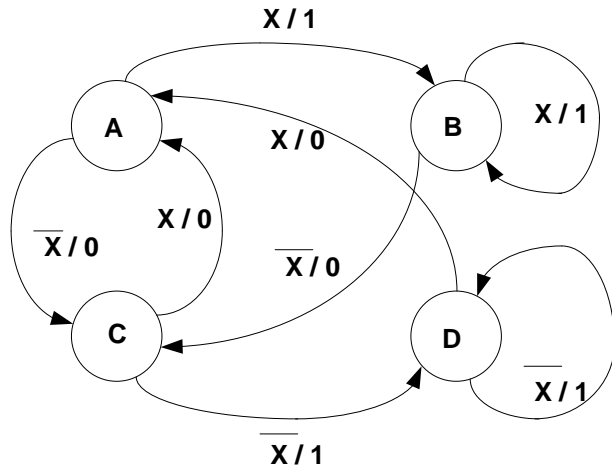
Exemplo: Mealy



```
D_est: process (clk) - - próximo estado e FF
begin
  if (clk 'event and clk='1') then
    case est is
      when A => if (x='1') then est <=B;
                  else est <=C;
            end if;
      when B => if (x='1') then est <=B;
                  else est <=C;
            end if;
      when C => if (x='1') then est <=A;
                  else est <=D;
            end if;
      when D => if (x='1') then est <=A;
                  else est <=D;
            end if;
    end case;
  end if;
end process D_est;
```

Introdução a VHDL

Exemplo: Mealy



```
Saida: process (est,x) - - saidas
begin
  case est is
    when A => if (x='1') then z <='1';
               else z <='0';
            end if;
    when B => if (x='1') then z <= '1';
               else z <= '0';
            end if;
    when C => if (x='1') then z <= '0';
               else z <= '1';
            end if;
    when D => if (x='1') then z <= '0';
               else z <= '1';
            end if;
  end case;
end process Saida;
end Ex2;
```

Introdução a VHDL

Conceito de função

Function <nome>
 (parâmetros) **return**
 type_signal **is**
 <declaração>

Begin

<comandos sequenciais>

Return <variavel>

end <nome>

Exemplo:

```
Function Ex (S: bit_vector) return  
    integer is  
    variable result: integer :=0;  
Begin  
    For I in 0 to 7 loop  
        result:=result*2;  
        If S(I)='1' then result:=result + 1;  
        endif;  
    end loop;  
    return result;  
end Ex;
```

Introdução a VHDL

Outro exemplo:

Function rising-edge (signal S: bit)
return Boolean is

Begin

if (S 'event and (s='1'))

then return true;

else return false;

endif;

End rising-edge;

entity **Dff** is

port (D,clk: in bit; q: out bit);

end **Dff**;

Architecture **comportamento** of **Dff** is

begin

process (clk)

begin

if (rising-edge(clk))

then q<=D;

end if;

end process;

end **comportamento**;

Introdução a VHDL

**Conceito de
sub-rotina**

**Equivalente a outras ling.
Comandos sequenciais**

Opção: INOUT

**Exemplo: Flip-Flop JK com
clear**

```
Procedure JKFF (signal rst,clk, J,K: in bit;  
                signal Q,QBAR: out bit);
```

```
Begin
```

```
  if rst='1'
```

```
    then Q<='0';
```

```
    else if (clk='1'and clk 'event)
```

```
      then if s='1'and k='1'
```

```
        then Q<=QBAR;
```

```
        else if j='1'and k='0' then Q='1';
```

```
        else if j='0'and k='1'then Q='0';
```

```
        end if;
```

```
      end if;
```

```
      QBAR<= NOT Q;
```

```
    end JKFF;
```