Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Exemplo de como implementar um circuito com apenas uma porta lógica.

```
library ieee;
use ieee.std_logic_1164.all;

entity ckt is
    port (a,b: in  std_logic;
             x: out std_logic);
end ckt;

architecture log of ckt is

begin
    x <= a or b;
end log;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Exemplo de circuito com múltiplas portas lógicas encadeadas.

```
library ieee;
use ieee.std_logic_1164.all;

entity ckt is
    port (a,b,c: in  std_logic;
                 x: out std_logic);
end ckt;

architecture log of ckt is

signal aux1 : std_logic;
signal aux2 : std_logic;
signal aux3 : std_logic;

begin
    aux1 <= b and c;
    aux2 <= a and c;
    aux3 <= a and b;
       x <= aux1 or aux2 or aux3;
end log;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Exemplo de circuito com múltiplas portas lógicas em linha.

```
library ieee;
use ieee.std_logic_1164.all;

entity ckt is
    port (a,b,c: in  std_logic;
                 x: out std_logic);
end ckt;

architecture log of ckt is

begin
    x <= (b and c) or (a and c) or (a and b);
end log;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Vetores

```
library ieee;
use ieee.std_logic_1164.all;

entity ckt is
    port (a,b: in  std_logic_vector(0 to 3);
             x: out std_logic_vector(0 to 3));
end ckt;

architecture log of ckt is

begin
    x <= a and b;
end log;
```

# Material de suporte - Vetores

```
library ieee;
use ieee.std_logic_1164.all;

entity ckt is
    port (a,b: in  std_logic_vector(0 to 3);
             x: out std_logic_vector(0 to 3));
end ckt;

architecture log of ckt is

begin
  x(3) <= a(3) and b(3);
  x(2) <= a(2) and b(2);
  x(1) <= a(1) and b(1);
  x(0) <= a(0) and b(0);
end log;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Componentes

```
library ieee;
use ieee.std_logic_1164.all;

entity C1 is
    port (x,y: in  std_logic;
             z: out std_logic);
end C1;

architecture log of C1 is

begin
    z <= x or y;
end log;


entity C2 is
    port (x: in  std_logic;
          z: out std_logic);
end C2;

architecture log of C2 is

begin
    z <= not x;
end log;

entity ckt is
    port (a,b: in  std_logic;
             w: out std_logic);
end ckt;

architecture log of ckt is

component C1 is
    port (x,y: in  std_logic;
             z: out std_logic);
end component;

component C2 is
    port (x: in  std_logic;
          z: out std_logic);
end component;

signal s: std_logic;
begin
  U1: C1 port map(a,b,s);
  U2: C2 port map(s,w);
end log;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

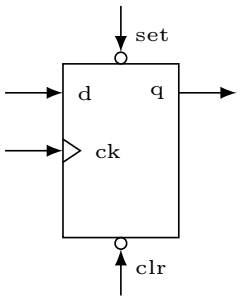
# Material de suporte - Componentes

```
library ieee;
use ieee.std_logic_1164.all;

entity C_ONES is
  port (a,b,c: in  std_logic;
              x: out std_logic);
end C_ONES;

architecture log of C_ONES is

begin
  x <= (b and c) or (a and c) or (a and b);
end log;

entity ckt is
  port (a,b: in  std_logic_vector(2 downto 0);
             w: out std_logic);
end ckt;

architecture log of ckt is

component C_ONES is
  port (a,b,c: in std_logic; x:out std_logic);
end component;

signal s1,s2: std_logic;
begin
  U1: C_ONES port map(a(2),a(1),a(0),s1);
  U2: C_ONES port map(b(2),b(1),b(0),s2);
  w <= s1 or s2;
end log;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Flip-Flop D

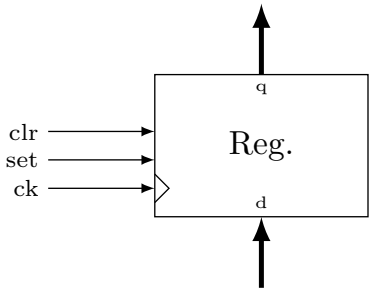



```
library ieee;
use ieee.std_logic_1164.all;

entity ffd is
   port (ck, clr, set, d : in  std_logic;
                       q : out std_logic);
end ffd;

architecture logica of ffd is

begin
   process(ck, clr, set)
   begin
      if    (set = '0')             then q <= '1';
      elsif (clr = '0')             then q <= '0';
      elsif (ck'event and ck ='1') then q <= d;
      end if;
   end process;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Registrador de 3 bits com entradas síncronas

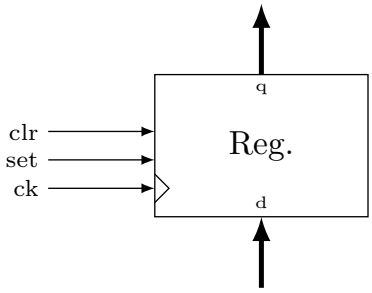



```
library ieee;
use ieee.std_logic_1164.all;

entity reg3bs is
  port (ck, clr, set : in  std_logic;
                   d : in  std_logic_vector(2 downto 0);
                   q : out std_logic_vector(2 downto 0));
end reg3bs;

architecture logica of reg3bs is

begin
  process (ck)
  begin
    if (ck'event and ck = '1') then        -- condicao do sinal relogio
      if    (clr = '1') then q <= "000";   -- teste para levar q=000
      elsif (set = '1') then q <= "111";   -- teste para levar q=111
      else                   q <= d;       -- armazena dado
      end if;
    end if;
  end process;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Registrador de 3 bits com entradas assíncronas

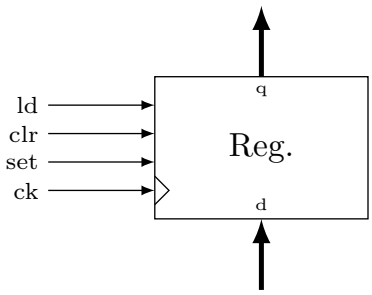



```
library ieee;
use ieee.std_logic_1164.all;

entity reg3ba is
  port (ck, clr, set : in  std_logic;
                   d : in  std_logic_vector(2 downto 0);
                   q : out std_logic_vector(2 downto 0));
end reg3ba;

architecture logica of reg3ba is

begin
  process (ck, clr, set)
  begin
    if     (clr = '1')             then q <="000"; -- q=000 independente de ck
    elsif  (set = '1')             then q <="111"; -- q=111 independente de ck
    elsif  (ck'event and ck ='1') then q <=d;     -- condicao do sinal relogio
    end if;
  end process;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - Registrador de 3 bits com entradas assíncronas e habilitador

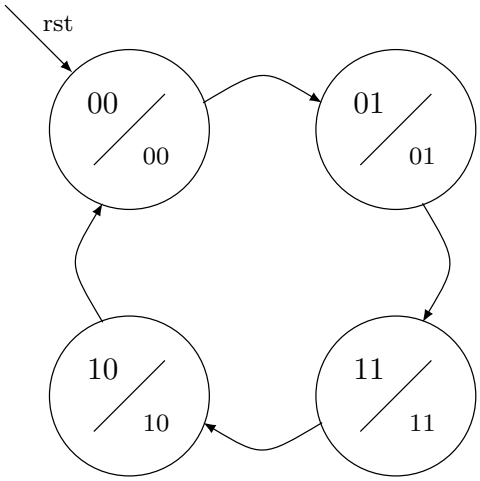



```
library ieee;
use ieee.std_logic_1164.all;

entity reg3ba_en is
  port (ck, clr, set, ld : in  std_logic;
                       d : in  std_logic_vector(2 downto 0);
                       q : out std_logic_vector(2 downto 0));
end reg3ba_en;

architecture logica of reg3ba_en is

begin
  process (ck, clr, set)
  begin
    if    (clr = '1') then q <= "000";    -- q=000 independente de ck
    elsif (set = '1') then q <= "111";    -- q=111 independente de ck
    elsif (ck'event and ck = '1') then    -- detecta borda de ck
      if  (ld = '1') then q <= d;         -- verifica habilitacao
      end if;
    end if;
  end process;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - MDE sequencial

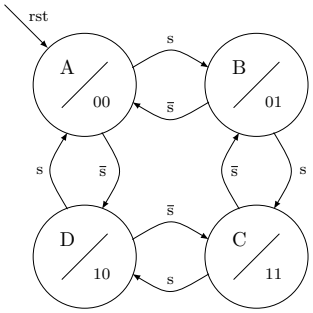



```
use ieee.std_logic_1164.all;

entity mde is
   port (ck, rst : in  std_logic;
         q       : out std_logic_vector(1 downto 0));
end mde;

architecture logica of mde is

begin
  process (ck, rst)
  begin
    if rst = '1' then                       -- estado inicial
      q <= "00";
    elsif (ck'event and ck ='1') then       -- ciclo de estados
      case q is
        when "00" => q <= "01";
        when "01" => q <= "11";
        when "11" => q <= "10";
        when "10" => q <= "00";
       end case;
    end if;
  end process;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

**Disciplina:** ELE1717 - Sistemas Digitais
**Curso:** Engenharia Mecatrônica

# Material de suporte - MDE do tipo Moore com um processo

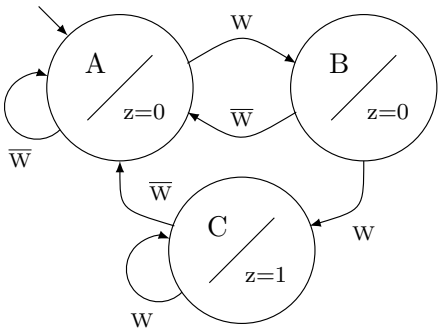



```
use ieee.std_logic_1164.all;

entity mde is
   port (ck, rst, s : in  std_logic;
                    q : out std_logic_vector(1 downto 0));
end mde;

architecture logica of mde is
  type st is (A, B, C, D);
  signal estado : st;

begin
  process (ck, rst)
  begin
    if rst = '1' then
      estado <= A;
    elsif (ck'event and ck ='1') then
      case estado is
        when A =>
          if s = '1' then estado <= B;
          else            estado <= D;
          end if;
        when B =>
          if s = '1' then estado <= C;
          else            estado <= A;
          end if;
        when C =>
          if s = '1' then estado <= D;
          else            estado <= B;
          end if;
        when D =>
          if s = '1' then estado <= A;
          else            estado <= C;
          end if;
       end case;
    end if;
  end process;

  with estado select
    q <= "00" when A,
         "01" when B,
         "11" when C,
         "10" when D;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

| | |
|---|---|
| **Disciplina:** | ELE1717 - Sistemas Digitais |
| **Curso:** | Engenharia Mecatrônica |

# Material de suporte - MDE do tipo Moore com dois processos

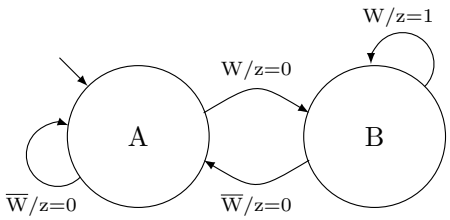



```
library ieee;
use ieee.std_logic_1164.all;

entity mde_b is
   port (ck, rst, w : in  std_logic;
                    z : out std_logic);
end mde_b;

architecture logica of mde_b is
   type state_type is (a, b, c);
   signal y_present, y_next : state_type;

begin
   process (w, y_present)
   begin
      case y_present is
         when a =>
            if w = '0' then y_next <= a;
            else            y_next <= b; end if;
         when b =>
            if w = '0' then y_next <= a;
            else            y_next <= c; end if;
         when c =>
            if w = '0' then y_next <= a;
            else            y_next <= c; end if;
      end case;
   end process;

   process (ck, rst)
   begin
      if rst = '0' then
         y_present <= a;
      elsif (ck'event and ck = '1') then
         y_present <= y_next;
      end if;
   end process;

   z <= '1' when y_present = c else '0';
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

**Disciplina:** ELE1717 - Sistemas Digitais
**Curso:** Engenharia Mecatrônica

# Material de suporte - MDE do tipo Mealy

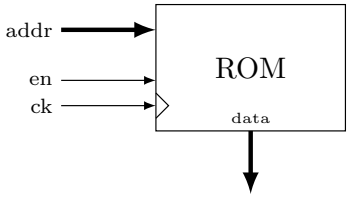



```
library ieee;
use ieee.std_logic_1164.all;

entity mde_d is
    port (ck, rst, w : in std_logic;
                  z : out std_logic);
end mde_d;

architecture logica of mde_d is
  type state_type is (a, b);
  signal y : state_type;

begin
  process (rst, ck)
  begin
   if rst = '0' then
     y <= a;
   elsif (ck'event and ck = '1') then
     case y is
       when a =>
         if w = '0' then y<=a;
         else            y<=b; end if;
       when b =>
         if w = '0' then y<=a;
         else            y<=b; end if;
     end case;
   end if;
  end process;

  process (y, w)
    begin
      case y is
        when a => z <= '0';
        when b => z <= w;
      end case;
  end process;
end logica;
```

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia - CT
Departamento de Engenharia Elétrica - DEE

**Disciplina:** ELE1717 - Sistemas Digitais
**Curso:** Engenharia Mecatrônica

# Material de suporte - Memória ROM




```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity s_rom is
   port (clk : in  std_logic;
         en  : in  std_logic;
         addr: in  std_logic_vector( 1 downto 0);
         data: out std_logic_vector(11 downto 0));
end s_rom;

architecture logica of s_rom is

type rom is array (0 to 3) of std_logic_vector(11 downto 0);
signal memoria : rom := (x"401",x"411",x"412",x"413");
signal r_addr  : std_logic_vector(0 to 1);

begin
   process(clk)
   begin
      if rising_edge(clk) then
         if en='1' then
            r_addr <= addr;
         end if;
      end if;
   end process;
   data <= memoria(to_integer(unsigned(r_addr)));
end logica;
```