



Aula 07 – Comandos Concorrentes básicos



Tópicos da aula

- **Tipos de modelagem e Comandos Concorrentes**
- **Definição de circuitos combinacionais**
- **Comandos concorrentes básicos**



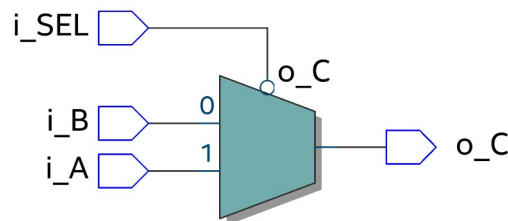
Tipos de modelagem

❑ Estrutural

❑ “O que é”

```
ENTITY mux_2x1 IS
PORT (a, b : IN BIT;
      sel : IN BIT;
      s : OUT BIT);
END mux;

ARCHITECTURE structural OF mux_2x1 IS
BEGIN
    s <= (a AND NOT sel) OR
    (b AND sel);
END structural;
```



❑ Comportamental

❑ “O que faz”

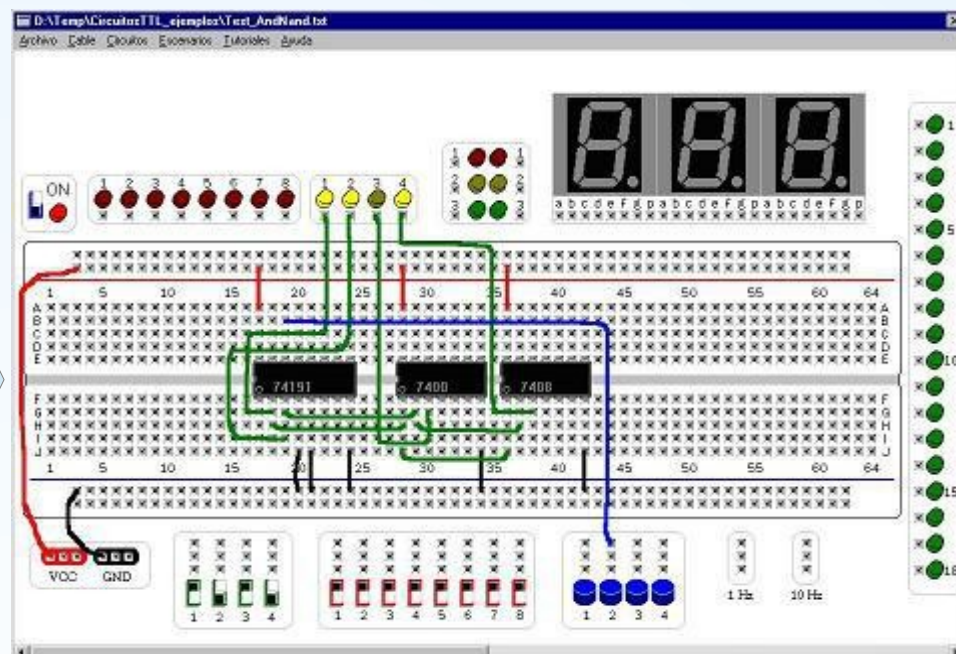
```
ENTITY mux_2x1 IS
PORT (a, b : IN BIT;
      sel : IN BIT;
      s : OUT BIT);
END mux;

ARCHITECTURE behavior OF mux_2x1 IS
BEGIN
    PROCESS(a,b,sel)
    BEGIN
        IF (sel='0') THEN
            s <= a;
        ELSE
            s <= b;
        END IF;
    END PROCESS;
END behavior;
```



Concorrência em VHDL

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3  use ieee.numeric_std.all;
4  use IEEE.STD_LOGIC_UNSIGNED.ALL;
5  use IEEE.STD_LOGIC_ARITH.ALL;
6
7  ENTITY PORTA_AND IS
8
9  PORT (
10      A      : IN STD_LOGIC;
11      B      : IN STD_LOGIC;
12      C      : OUT STD_LOGIC;
13      D      : OUT STD_LOGIC;
14      E      : OUT STD_LOGIC
15  );
16  END PORTA_AND;
17
18  ARCHITECTURE behavioral OF PORTA_AND IS
19
20  BEGIN
21
22      C <= A and B;
23      D <= (not A) and B;
24      E <= A or B;
25
26  END behavioral;
```





Circuitos Combinacionais

- Qualquer alteração em um dos sinais de **entrada**, implica em alteração **imediata** em um ou mais sinais de **saída**
- Ex.: Funções Booleanas; Portas Lógicas





Comando **WHEN ELSE**

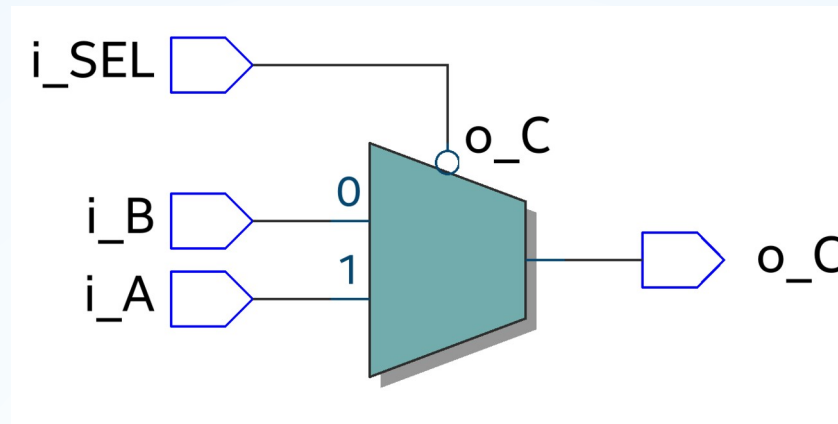
- Transferência condicional de um sinal
- **Contém:** uma lista de condições e expressões
- **Primeira condição verdadeira:** define expressão transferida
- **Formato da construção:**

```
sinal_destino <= expressao_a WHEN condicao_1 ELSE  
                    expressao_b WHEN condicao_2 ELSE  
                    expressao_c;
```



Comando **WHEN ELSE**

- ❏ Útil para a construção de sistemas seletores (MULTIPLEXADORES)
- ❏ Ex.: MUX 2x1





Comando **WHEN ELSE**

- ❏ Multiplexadores podem “comutar” barramentos de dados
- ❏ Ex.: MUX 4x1 de 4 bits

```
b <= "1000" when a = "00" else  
    "0100" when a = "01" else  
    "0010" when a = "10" else  
    "0001" when a = "11";
```




Exercícios

- **Implemente um MULTIPLEXADOR 2x1 utilizando os comandos WHEN ELSE**
- **Implemente um MULTIPLEXADOR 4x1 de 8 bits utilizando os comandos WHEN ELSE**



Construção **WITH SELECT**

- Transferência condicional de um sinal
- Contém: uma lista de opções
- Todas as condições da expressão de escolha devem ser consideradas
 - não existe uma prioridade como na construção **WHEN ELSE**
- As opções podem ser agrupadas: o caractere **|** equivale a “ou”
 - **TO** e **DOWNTO** delimitam faixas de opções
- Opções restantes: palavra reservada **OTHERS**
- Formato da construção:

```
WITH expressao_escolha SELECT      -- expressao_escolha =
  sinal_dest <= expr_a WHEN condicao_1,      -- condicao_1
  expr_b WHEN condicao_2,                  -- condicao_2
  expr_c WHEN condicao_3 | condicao_4,      -- condicao_3 ou condicao_4
  expr_d WHEN condicao_5 TO condicao_7,    -- condicao_5 ate condicao_7
  expr_e WHEN OTHERS;                    -- condicoes restantes
```



Construção **WITH SELECT**

```
with a select b <=
    "1000" when "00",
    "0100" when "01",
    "0010" when "10",
    "0001" when "11";
```



Exercícios

- ❏ Implemente um MULTIPLEXADOR 4x1 de 8 bits utilizando os comandos `WITH SELECT`
- ❏ Implemente o mesmo MULTIPLEXADOR utilizando um `PROCESS`



FIM AULA 7