

# VHDL

## Sinais e Tipos de Dados

Rafael Corsi Ferrão - IMT



`rafael.corsi@maua.br`  
`http://www.maua.br`

8 de outubro de 2014

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

1 Concorrente

2 Atribuição

3 WHEN ELSE

4 WITH SELECT

5 BLOCK

## Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

### 1 Concorrente

### 2 Atribuição

### 3 WHEN ELSE

### 4 WITH SELECT

### 5 BLOCK

## Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

## Definição

Comandos concorrentes são executados simultaneamente, sendo suas sequências irrelevante

Numa região de código concorrente, a ordem do código não influencia no seu funcionamento. Podemos definir os seguintes comandos concorrentes:

- ▶ Atribuição de um valor
- ▶ WHEN ELSE
- ▶ WITH SELECT
- ▶ BLOCK

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

1 Concorrente

2 Atribuição

3 WHEN ELSE

4 WITH SELECT

5 BLOCK

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

## Definição

É a forma de atribuir um valor a um sinal ou a uma porta.

Exemplo:

```
out1 <= (i0 AND i1) OR (i3 AND (NOT i4));  
out2 <= i5;
```

Que é igual á:

```
out2 <= i5;  
out1 <= (i3 AND (NOT i4)) OR (i0 AND i1);
```

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

## Definição

É a forma de atribuir um valor a um sinal ou a uma porta.

Exemplo:

```
out1 <= (i0 AND i1) OR (i3 AND (NOT i4));  
out2 <= i5;
```

Que é igual á:

```
out2 <= i5;  
out1 <= (i3 AND (NOT i4)) OR (i0 AND i1);
```

Mas que é diferente de :

```
out2 <= i5;  
out1 <= i3 AND ((NOT i4) OR i0) AND i1;
```

Concorrente

Atribuição

**WHEN ELSE**

WITH SELECT

BLOCK

1 Concorrente

2 Atribuição

**3 WHEN ELSE**

4 WITH SELECT

5 BLOCK



Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

## Definição

Permite a transmissão condicional de um sinal

Construção da expressão WHEN ELSE:

---

```
sinal/porta <= expressao_a WHEN condicao1 ELSE  
                expressao_b WHEN condicao2 ELSE  
                expressao_c;
```

---

Exemplo :

---

```
Dout <= in0 WHEN mux = 0 OR rst = 1 ELSE  
        in1 WHEN mux = 1 AND rst = 0 ELSE  
        in2;
```

---

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

1 Concorrente

2 Atribuição

3 WHEN ELSE

4 WITH SELECT

5 BLOCK

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

## Definição

Transfere um sinal de destino seguindo uma relação de opção

Construção da expressão WITH SELECT:

---

```
WITH expressao SELECT
    sinal_destino <= expressao_a WHEN condicao_1,
                    expressao_b WHEN condicao_2 TO condicao_5,
                    expressao_c WHEN condicao_6 | condicao_7,
                    expressao_n WHEN OTHERS;
```

---

Exemplo :

---

```
WITH mux SELECT
    Dout <= in0 WHEN 0,
           in1 WHEN 1,
           in2 WHEN OTHERS;
```

---

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

1 Concorrente

2 Atribuição

3 WHEN ELSE

4 WITH SELECT

5 BLOCK

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

## Definição

O comando BLOCK tem a finalidade de segmentar o código

Construção da expressão BLOCK:

---

```
nome_block : BLOCK
-- sinais internos do bloco
BEGIN
-- comandos

END BLOCK nome_block;
```

---

Concorrente

Atribuição

WHEN ELSE

WITH SELECT

BLOCK

Exemplo :

---

```
teste : BLOCK
-- sinais internos do bloco
  SIGNAL tmp : STD_LOGIC_VECTOR(1 downto 0);
BEGIN
-- comandos
  WITH sel SELECT
    tmp <= "00" WHEN 0,
          "01" WHEN 1,
          "10" WHEN 2,
          "11" WHEN OTHERS;

  out <= tmp;
END BLOCK nome_block;
```

---