

VHDL

Sequencial

Rafael Corsi Ferrão - IMT



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

6 de outubro de 2014

Sequencial

Process

Comandos
sequenciais

IF ELSE
CASE WHEN

1 Sequencial

2 Process

3 Comandos sequenciais

- IF ELSE
- CASE WHEN

Sequencial

Process

Comandos sequenciais

IF ELSE
CASE WHEN

1 Sequencial

2 Process

3 Comandos sequenciais

- IF ELSE
- CASE WHEN

Sequencial

Process

Comandos
sequenciais

IF ELSE
CASE WHEN

Definição

Comandos sequencias são avaliados de maneira sequencial, na ordem que aparecem no código.

A descrição de uma lógica concorrente é alocado em uma região específica de código, chamada **PROCESS**. Os comandos sequenciais são:

- ▶ Atribuição de valor (sinal e **variável**)
- ▶ IF
- ▶ CASE
- ▶ WAIT

Sequencial

Process

Comandos
sequenciais

IF ELSE
CASE WHEN

1 Sequencial

2 Process

3 Comandos sequenciais

- IF ELSE
- CASE WHEN

Definição

A região definida como sendo um processo (process) permite a execução de comandos sequencias.

A região de processo é definida por duas partes:

- ▶ lista de sensibilidade
- ▶ comandos sequenciais

A lista de sensibilidade é usada para indicar que o processo deve ser executado somente quando algum dos sinais sofreu alteração no seu valor.

Sequencial

Process

Comandos
sequenciais

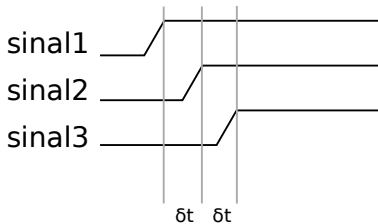
IF ELSE
CASE WHEN

```
rotulo : PROCESS (lista de sensibilidade)
    -- Declaracao de sinais e variaveis
BEGIN
    -- Parte referente aos comandos sequencias
    -- comando
    -- comando
END PROCESS rotulo;
```

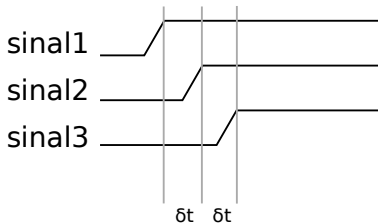
Exemplo:

```
ex1 : PROCESS (sinal1, sinal2, sinal3)
BEGIN
    sinal2 <= sinal1;
    sinal3 <= sinal2;
END PROCESS ex1;
```

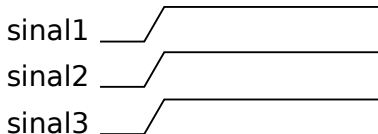
Com o exemplo, obtemos a seguinte carta de tempo:



Com o exemplo, obtemos a seguinte carta de tempo:



mas $\delta t = 0$:



Sequencial

Process

Comandos
sequenciaisIF ELSE
CASE WHEN

Importante a notar os sinais que fazem parte da lista de sensibilidade, imagine a situação em que o sinal2 não é listado:

```
ex2 : PROCESS (sinal1, sinal3)
BEGIN
    sinal2 <= sinal1;
    sinal3 <= sinal2;
END PROCESS ex2;
```

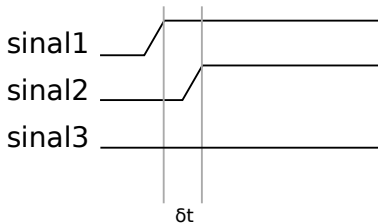
Sequencial

Process

Comandos
sequenciaisIF ELSE
CASE WHEN

Importante a notar os sinais que fazem parte da lista de sensibilidade, imagine a situação em que o sinal2 não é listado:

```
ex2 : PROCESS (sinal1, sinal3)
BEGIN
    sinal2 <= sinal1;
    sinal3 <= sinal2;
END PROCESS ex2;
```



Sequencial

Process

**Comandos
sequenciais**

IF ELSE
CASE WHEN

1 Sequencial

2 Process

3 Comandos sequenciais

- IF ELSE
- CASE WHEN

Definição

Permite a execução condicional de um ou mais comandos sequências

A ordem de execução respeita a ordem de aparição no código.
A utilização é ilustrada a seguir:

```
IF condicao_1 THEN
    -- comandos 1
ELSIF condicao_2 THEN
    -- comandos 2
ELSIF condicao_3 THEN
    -- comandos 3
ELSE
    -- comandos restantes;
END IF;
```

Sequencial

Process

Comandos
sequenciaisIF ELSE
CASE WHEN

```

ENTITY mux IS
    PORT(
        i0, i1, i2, i3 : IN STD_LOGIC; -- entradas
        s : IN STD_LOGIC_VECTOR(1 downto 0); -- selecao
        o : OUT STD_LOGIC -- saida
    );
END mux;

ARCHITECTURE bhv OF mux IS
BEGIN
    PROCESS(i0,i1,i2,i3,s)
    BEGIN
        IF s = "00" THEN
            o <= i0;
        ELSIF s="01" THEN
            o <= i1;
        ELSIF s="10" THEN
            o <= i2;
        ELSE
            o <= i3;
        END IF;
    END PROCESS;
END bhv;

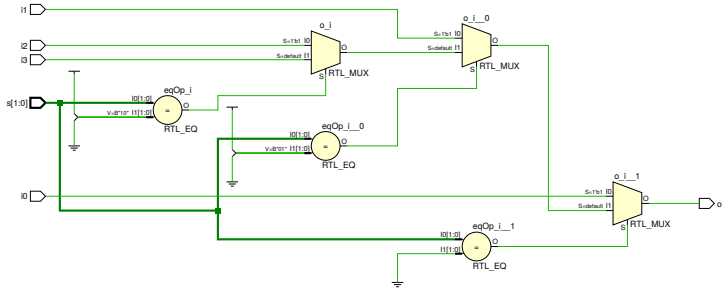
```

Sequential

Process

Comandos
sequenciais

IF ELSE
CASE WHEN

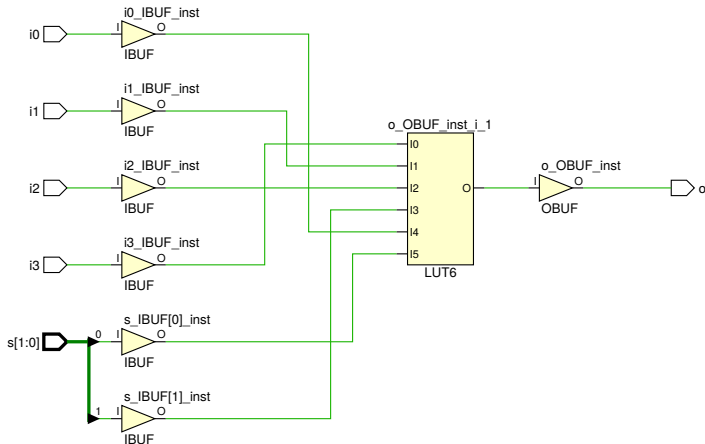


Sequential

Process

Comandos
sequenciais

IF ELSE
CASE WHEN



Definição

Permite a execução condicional de um ou mais comandos sequências. Cada condição define um ou mais comandos sequências a serem executados.

As condições devem ser mutualmente exclusivas, e cada condição pode executar um ou mais comandos.

```
CASE expressao IS
    WHEN condicao_1                => comandos_a;
    WHEN condicao_2                => comandos_b;
    WHEN condicao_3 | condicao_4    => comandos_c;
    WHEN condicao_5 OR condicao_6  => comandos_d;
    WHEN OTHERS                   => comandos_e;
END CASE;
```

Sequencial

Process

Comandos
sequenciais

IF ELSE
CASE WHEN

```
ENTITY mux_2 IS
    PORT(
        i0, i1, i2, i3 : IN STD_LOGIC; -- entradas
        s : IN STD_LOGIC_VECTOR(1 downto 0); -- selecao
        o : OUT STD_LOGIC -- saida
    );
END mux_2;

ARCHITECTURE bhv OF mux_2 IS
BEGIN

    PROCESS(i0,i1,i2,i3,s)
    BEGIN
        CASE s IS
            WHEN "00"    => o <= i0;
            WHEN "01"    => o <= i1;
            WHEN "10"    => o <= i2;
            WHEN OTHERS => o <= i3;
        END CASE;
    END PROCESS;

END bhv;
```

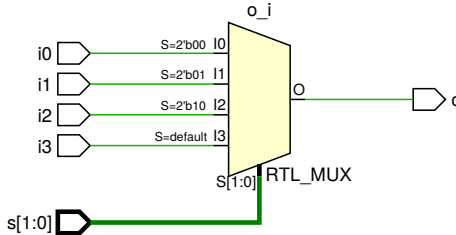
Sequential

Process

Comandos
sequenciais

IF ELSE

CASE WHEN



Sequential

Process

Comandos
sequenciais

IF ELSE
CASE WHEN

