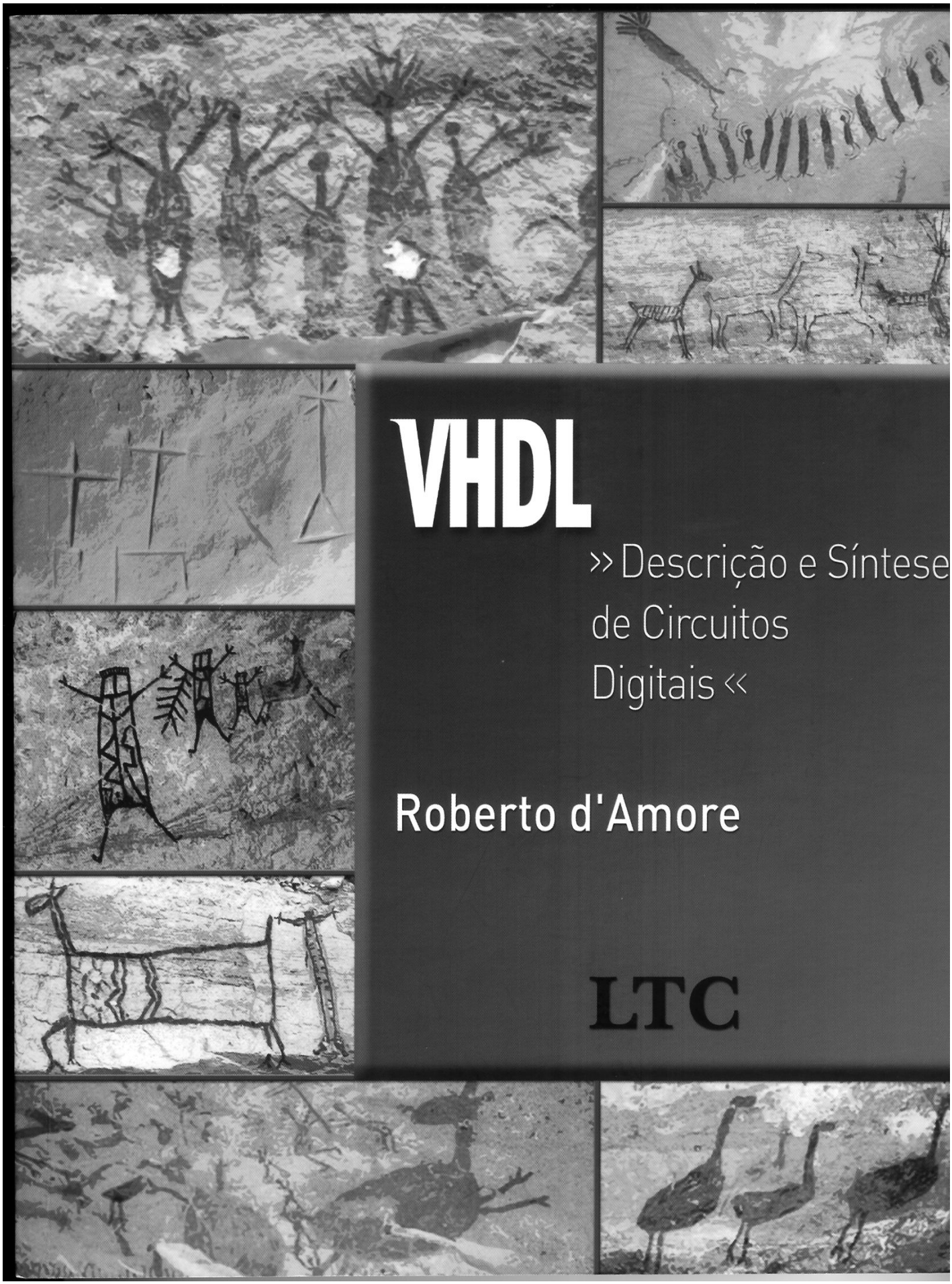
Lista de Exercícios – SEL0632

Capítulo 04



```
 1  ENTITY cas_sel3 IS
 2    PORT (i0, i1, b0, b1       : IN  BIT;
 3          c0, c1, c2, c3       : IN  CHARACTER;
 4          o0, o1, o2, o3, o4   : OUT BIT);
 5  END cas_sel3;
 6
 7  ARCHITECTURE correta OF cas_sel3 IS
 8    SIGNAL   sel_bit : BIT_VECTOR(1 DOWNTO 0);
 9    SIGNAL   sel_stg : STRING(1 TO 3);
10    CONSTANT cnt     : STRING := "eta";
11  BEGIN
12    sel_bit <= b1 & b0;
13    WITH sel_bit SELECT      -- Correto: possivel determinar os valores de retorno da expressao
14     o0 <= i0 WHEN "00",     --          no contexto da expressao (00, 01, ... , 11)
15           i1 WHEN OTHERS;
16
17    sel_stg <= c2 & c1 & c0;
18    WITH sel_stg SELECT      -- Correto: possivel determinar os valores de retorno da expressao
19      o1 <= i0 WHEN "abc",   --          no contexto da expressao (... aaa, aab, aac ...)
20            i1 WHEN OTHERS;
21
22    WITH sel_stg SELECT      -- Correto: valor constante
23      o3 <= i0 WHEN cnt,     -- constante cnt := "eta"
24            i1 WHEN OTHERS;
25
26    WITH c3 SELECT
27      o4 <= i0 WHEN 'E',   -- Correto: valor constantte
28            i1 WHEN OTHERS;
29  END correta;
```

Quadro 4.7.7 Descrição corrigida: " cas_sel3 ".

4.8 Principais pontos abordados

- Na construção " IF ELSE ", a seqüência das condições define a prioridade, pois a primeira condição verdadeira encontrada na lista estabelece quais comandos devem ser executados.
- A construção " WHEN ELSE " é uma transferência condicional de sinal concorrente, e a sua correspondente construção seqüencial é " IF ELSE ".
- Na construção " CASE WHEN ", as condições devem ser mutuamente exclusivas; todas as condições devem ser especificadas, e elas possuem a mesma prioridade.
- A construção " WITH SELECT " é uma transferência condicional de sinal concorrente, e a construção seqüencial correspondente é " CASE WHEN ".
- Sempre que possível, deve-se dar preferência a descrições empregando a construção " CASE WHEN " em detrimento da construção " IF ELSE ".
- O comando " WAIT " suspende a execução de um processo. Três tipos de construções podem ser empregados: " WAIT ON ", " WAIT UNTIL ", e " WAIT FOR ".
- O comando " NULL " não realiza nenhuma operação.

4.9 Exercícios

4.9.1 Apresente o comportamento da descrição do Quadro 4.2.1 alterando a linha 8 de duas formas. A primeira substituindo por " PROCESS (sa) " e a segunda por " PROCESS (sa, sb, sc) ". Descreva as iterações que devem ocorrer para cada caso.

4.9.2 A descrição do Quadro 4.9.1 contém erros. Apresente possíveis soluções para os erros e comente as razões.

```
1  ENTITY erro1 IS
2    PORT (a, b :  IN  BIT;
3          c    :  IN  BOOLEAN;
4          x, y, z, k    :  OUT BIT);
5  END erro1;
6
7  ARCHITECTURE teste OF erro1 IS
8  BEGIN
9    abc: PROCESS (a, b, c)
10   BEGIN
11     IF a THEN x <= '1';
12     END IF;
13     IF c THEN z <= '1';
14     END IF;
15     IF a AND b = '1' THEN y <= '0';
16     END IF;
17     IF c AND b = '1' THEN k <= '1';
18     END IF;
19   END PROCESS abc;
20 END teste;
```

Quadro 4.9.1 Código com erros.

4.9.3 A descrição do Quadro 4.9.2 contém erros. Apresente possíveis soluções para os erros e comente os motivos.

```
1  ENTITY erro2 IS
2    PORT (x   :  IN  INTEGER RANGE 0 TO 15;
3          s   :  OUT INTEGER RANGE 0 TO 15);
4  END erro2;
5
6  ARCHITECTURE teste OF erro2 IS
7  BEGIN
8    abc: PROCESS (x)
9    BEGIN
10     CASE x IS
11       WHEN 7 TO 10     => s <= 0;
12       WHEN 3 DOWNTO 0  => s <= 1;
13       WHEN 4 | 5 | 9   => s <= 2;
14       WHEN 15          => s <= 3;
15     END CASE;
16   END PROCESS abc;
17 END teste;
```

Quadro 4.9.2 Código com erros.

4.9.4 A descrição do Quadro 4.9.3 corresponde ao circuito de seleção com uma alteração na lista de sensibilidade. Verifique e comente os problemas acarretados pela alteração.

```
1  ENTITY mux_4b IS
2    PORT (i0, i1, i2, i3  :  IN  BIT;  -- entradas
3          s0, s1          :  IN  BIT;  -- selecao
4          ot              :  OUT BIT); -- saida
5  END mux_4b;
6
7  ARCHITECTURE errada OF mux_4b IS
8  BEGIN
9    abc: PROCESS (i0, i1, i2, i3)  -- sinais s0 e s1 removidos da lista
10   BEGIN
11     IF    s1 & s0 = "00" THEN  ot <= i0;
12     ELSIF s1 & s0 = "01" THEN  ot <= i1;
13     ELSIF s1 & S0 = "10" THEN  ot <= i2;
14     ELSE                       ot <= i3;
15     END IF;
16   END PROCESS abc;
17 END errada;
```

Quadro 4.9.3 Circuito de seleção alterado.

4.9.5 Proponha duas descrições para o esquema da Figura 4.9.1: uma empregando a construção " IF ELSE " e outra com a construção " CASE WHEN ". Três condições estabelecem uma operação lógica entre as entradas " x " e " y ". A condição de maior prioridade, " a=b ", define uma operação "e"; a condição " b=c " define a operação "ou"; e a última condição, " a=c ", de menor prioridade, define uma operação "ou-exclusivo".

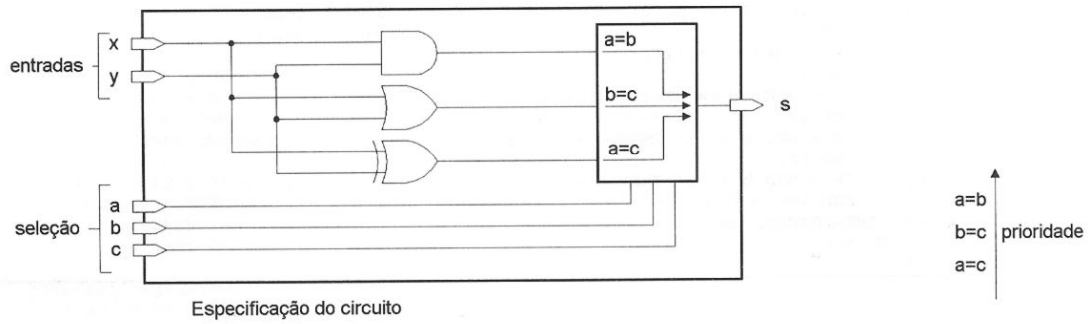


Figura 4.9.1 Proposta de um circuito com problemas na especificação.

4.9.6 Refaça o Exercício 3.10.2 empregando a construção " IF ELSE ".

4.9.7 Refaça o Exercício 3.10.2 empregando a construção " CASE WHEN ".

4.9.8 Refaça o Exercício 3.10.4 empregando a construção" IF ELSE ".

4.9.9 Refaça o Exercício 3.10.4 empregando a construção " CASE WHEN ".

4.9.10 Refaça o Exercício 3.10.6 empregando a construção " IF ELSE ".

4.9.11 Refaça o Exercício 3.10.6 empregando a construção " CASE WHEN ".

4.9.12 Refaça o Exercício 3.10.8 empregando comandos seqüenciais.

4.9.13 Refaça o Exercício 3.10.9 empregando comandos seqüenciais. Não utilize blocos para divisão da descrição; divida a descrição em dois processos: um contendo o decodificador, e o outro contendo o circuito de seleção.

4.9.14 A Figura 4.9.2 apresenta o diagrama de um circuito de deslocamento rápido, *barrel shift*, para discussão do exercício. As entradas " d1 " e " d0 " definem o deslocamento dos bits da entrada " v " para a saída " s ". O deslocamento pode variar entre zero a três bits para a esquerda, conforme descrito na figura.

Apresente duas descrições para a solução empregando as construções " IF ELSE " e " CASE WHEN " aninhadas, e outras duas soluções sem construções aninhadas. Sintetize as descrições e comente os resultados.

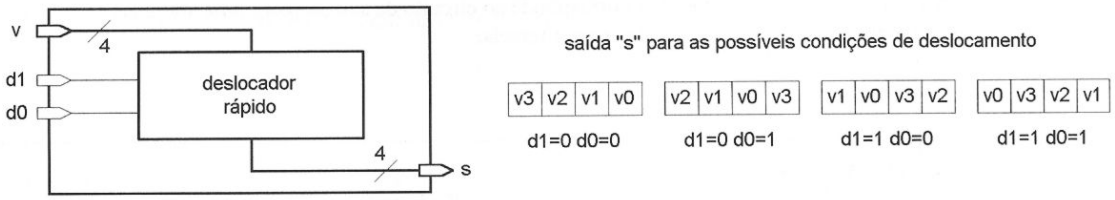


Figura 4.9.2 Diagrama de um deslocador rápido de 4 bits.

4.9.15 Sintetize as descrições dos Quadros 4.3.4 e 4.9.3, entidades " mux_4a " e " mux_4b ", respectivamente. Note que na entidade " mux_4b " dois sinais foram removidos da lista de sensibilidade do processo. Verifique como a ferramenta de síntese trata esse problema. Compare os diagramas gerados no nível RTL e verifique as mensagens geradas pela ferramenta.

4.9.16 A descrição do Quadro 4.9.4 contém erros. Corrija-os e proponha alternativas quando necessário, considerando as restrições quanto à expressão de escolha de uma construção " CASE WHEN ".

```
ENTITY cas_sel6 IS
  GENERIC (gen : INTEGER := 7);
  PORT (bt      : IN  BIT;
        btv     : IN  BIT_VECTOR(2 DOWNTO 0);
        bol     : IN  BOOLEAN;
        chr     : IN  CHARACTER;
        int     : IN  INTEGER;
        rl      : IN  REAL;
        st      : IN  STRING(1 TO 3);
        tm      : IN  TIME;
        o0,o1,o2,o3,o4,o5,o6,o7,o8, o9 : OUT INTEGER);
END cas_sel6;

ARCHITECTURE teste OF cas_sel6 IS
  CONSTANT cst : INTEGER := 7;
BEGIN
  PROCESS(bt, btv, bol, chr, int, rl, st, tm)
  BEGIN
  CASE bt IS
    WHEN   '0'  => o0 <= 0;
    WHEN OTHERS => o0 <= 1;
  END CASE;

  CASE btv IS
    WHEN "000"  => o1 <= 0;
    WHEN OTHERS => o1 <= 1;
  END CASE;

  CASE bol IS
    WHEN FALSE  => o2 <= 0;
    WHEN OTHERS => o2 <= 1;
  END CASE;

  CASE chr IS
    WHEN 'a'    => o3 <= 0;
    WHEN OTHERS => o3 <= 1;
  END CASE;

  CASE int IS
    WHEN   7    => o4 <= 0;
    WHEN OTHERS => o4 <= 1;
  END CASE;

  CASE rl IS
    WHEN  7.0   => o5 <= 0;
    WHEN OTHERS => o5 <= 1;
  END CASE;

  CASE st IS
    WHEN  "abc" => o6 <= 0;
    WHEN OTHERS => o6 <= 1;
  END CASE;

  CASE tm IS
    WHEN  50 ns => o7 <= 0;
    WHEN OTHERS => o7 <= 1;
  END CASE;

  CASE int IS
    WHEN  gen   => o8 <= 0;
    WHEN OTHERS => o8 <= 1;
  END CASE;

  CASE int IS
    WHEN  cst   => o9 <= 0;
    WHEN OTHERS => o9 <= 1;
  END CASE;
  END PROCESS;
END teste;
```

Quadro 4.9.4 Descrição com erros.