

Nome: Gustavo Pereira Chaves
Matrícula: 19/0014113
Turma: 03A
Data: 05/03/2021

Relatório do Experimento 3

Questão 1:

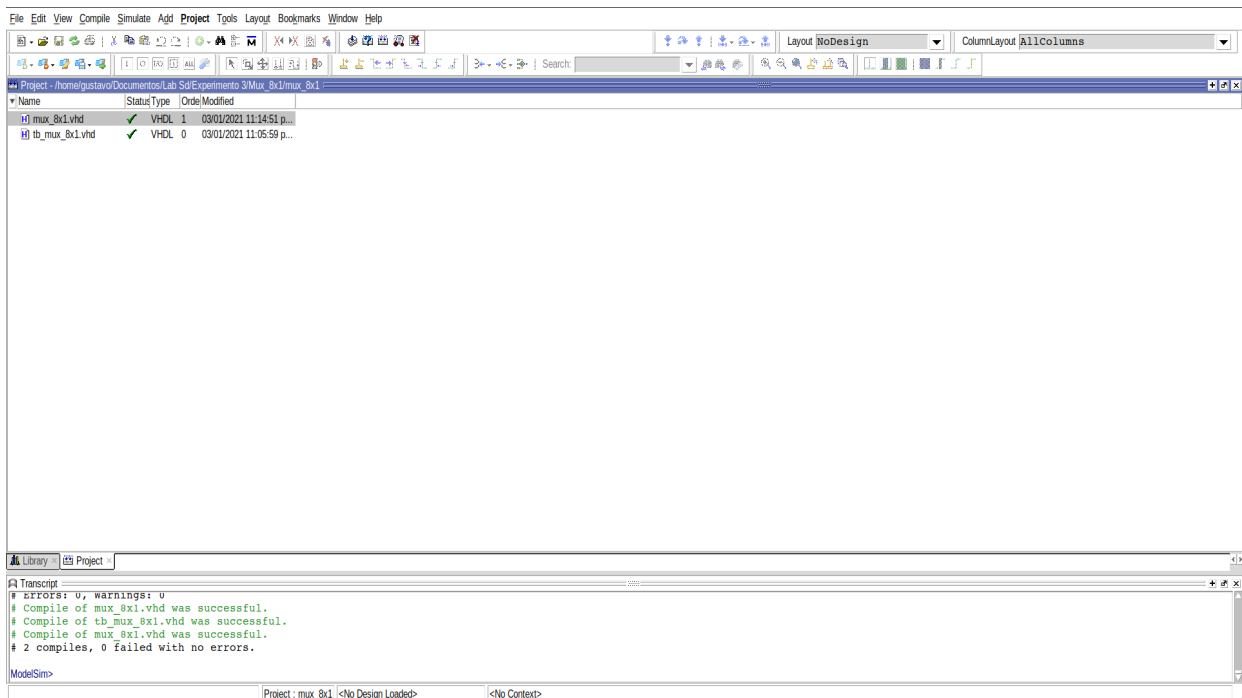
- Documentação:
mux_8x1.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 05/03/2021
6
7  -- Implementação de um Multiplexador 8x1 (8 entradas de dados, 1 saída e 1 seletor de 3 bits)
8
9  -- *****
10 -- Circuito: Multiplexador de 2 entradas:
11 --           D Entrada
12 --           S Seletor
13 --           Y Saída
14 -- *****
15
16 --Package (Pacote)
17 --constantes e bibliotecas
18
19 library IEEE;
20 use IEEE.std_logic_1164.all;
21
22 --Entity (Entidade)
23 --pinos de entrada e saída
24
25 entity mux_8x1 is
26     port (
27         D :in std_logic_vector(7 downto 0);
28         S :in std_logic_vector(2 downto 0);
29         Y :out std_logic
30     );
31 end mux_8x1;
32
33
34 --Architecture (Arquitetura)
35 --implementacoes do projeto
36
37 architecture mux of mux_8x1 is
38
39     -- a definicao inicia por
40     begin
41
42         -- Y = f(D, S)
43         Y <= D(0) when (S = "000") else
44             D(1) when (S = "001") else
45             D(2) when (S = "010") else
46             D(3) when (S = "011") else
47             D(4) when (S = "100") else
48             D(5) when (S = "101") else
49             D(6) when (S = "110") else
50             D(7) when (S = "111");
51
52     end mux;
53
54 -- a definicao termina por end
```

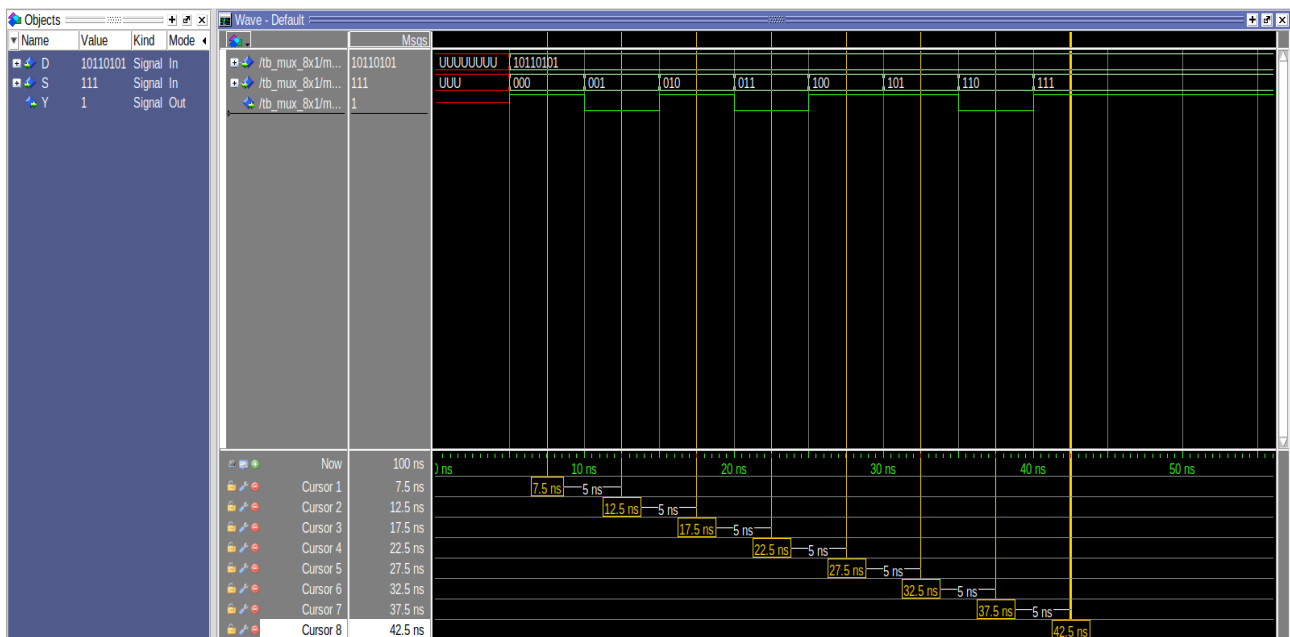
tb_mux_8x1.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 05/03/2021
6
7  -- Implementação de um Multiplexador 8x1 (8 entradas de dados, 1 saída e 1 seletor de 3 bits)
8
9  -- *****
10 -- Testbench para simulação funcional do
11 -- Circuito: Multiplexador de 2 entradas:
12 --           D Entrada
13 --           S Seletor
14 --           Y Saída
15 -- *****
16
17 -- entity (Entidade)
18 -- tb_mux_8x1 uma entidade sem pinos de entrada e saída
19
20 entity tb_mux_8x1 is end;
21
22 --Package (Pacote)
23 --constantes e bibliotecas
24
25 library ieee;
26 use ieee.std_logic_1164.ALL;
27 use std.textio.all;
28
29
30 -- Architecture (Arquitetura)
31
32 architecture testbench of tb_mux_8x1 is
33
34 -- Declaracao do componente mux_8x1, referente a sua arquitetura descrita no arquivo mux_8x1.vhd
35
36 component mux_8x1 is
37     port(
38         D :in std_logic_vector(7 downto 0);
39         S :in std_logic_vector(2 downto 0);
40         Y :out std_logic
41     );
42 end component;
43
44 -- Sinais auxiliares para a interconexao ao
45 -- processo de estimulo
46
47 signal d : std_logic_vector(7 downto 0);
48 signal s : std_logic_vector(2 downto 0);
49
50 -- Instancia do componente mux_8x1 e interconexao do componente ao processo de estimulo
51
52 begin
53     mux: mux_8x1 PORT MAP (D => d, S => s, Y => open);
54
55 -- Implementacao do processo de estimulo
56
57 estimulo: process
58     begin
59
60         wait for 5 ns; d <= "10110101"; s <= "000";
61         wait for 5 ns; s <= "001";
62         wait for 5 ns; s <= "010";
63         wait for 5 ns; s <= "011";
64         wait for 5 ns; s <= "100";
65         wait for 5 ns; s <= "101";
66         wait for 5 ns; s <= "110";
67         wait for 5 ns; s <= "111";
68         wait;
69
70 -- Fim do processo de estimulo
71
72 end process;
73
74 --Fim da definição da arquitetura
75
76 END;
```

- **Compilação:**



- **Simulação do Código:**



Cursor 1: Em 7,5 ns, D = 10110101, S = 000, Y = 1;
 Cursor 2: Em 12,5 ns, D = 10110101, S = 001, Y = 0;
 Cursor 3: Em 17,5 ns, D = 10110101, S = 010, Y = 1;
 Cursor 4: Em 22,5 ns, D = 10110101, S = 011, Y = 0;
 Cursor 5: Em 27,5 ns, D = 10110101, S = 100, Y = 1;
 Cursor 6: Em 32,5 ns, D = 10110101, S = 101, Y = 1;
 Cursor 7: Em 37,5 ns, D = 10110101, S = 110, Y = 0;
 Cursor 8: Em 42,5 ns, D = 10110101, S = 111, Y = 1;

Questão 2:

- Documentação:

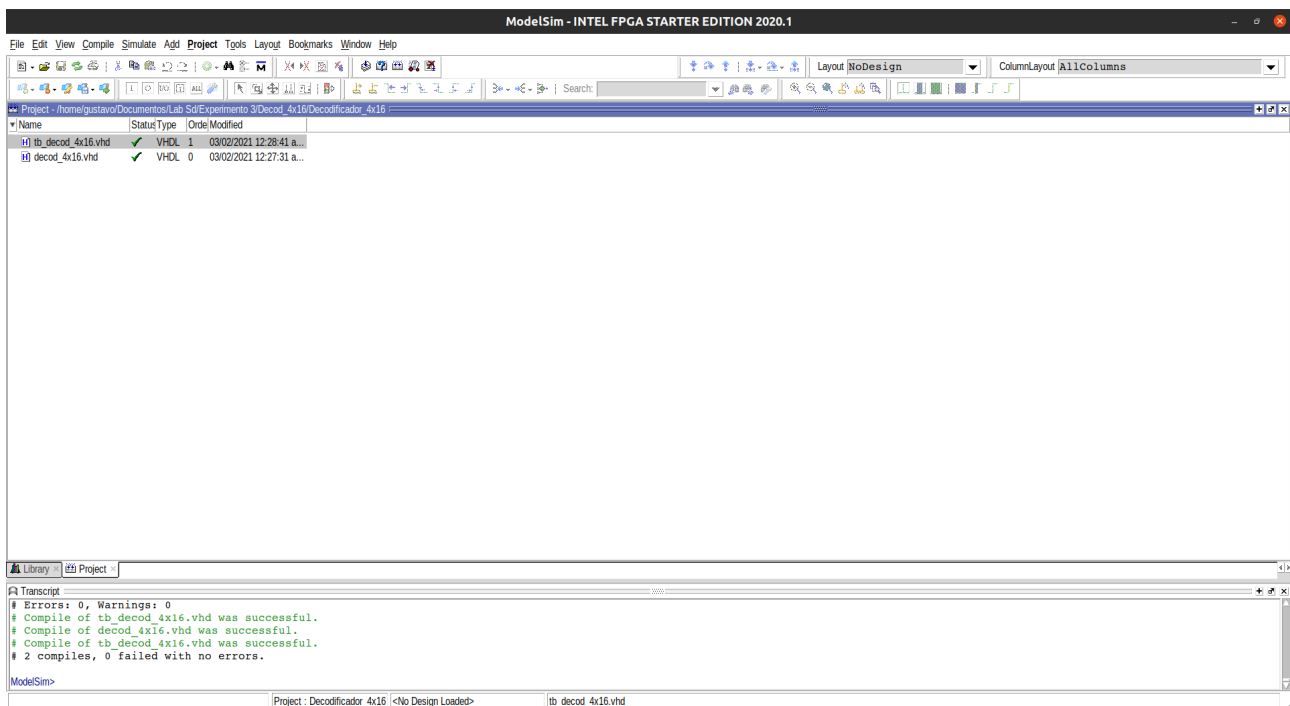
decod_4x16.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 05/03/2021
6
7  -- Implementação de um Decodificador de 4 para 16 bits (1 vetor de 4 bits de entrada e 1 vetor de 16 bits de saída)
8
9  -- *****
10 -- Circuito: Decodificador de 1 entrada:
11 --           A Entrada
12 --           Y Saída
13 -- *****
14
15 --Package (Pacote)
16 --constantes e bibliotecas
17
18 library IEEE;
19 use IEEE.std_logic_1164.all;
20
21 --Entity (Entidade)
22 --pinos de entrada e saída
23
24 entity decod_4x16 is
25     port (
26         A :in std_logic_vector(3 downto 0);
27         Y :out std_logic_vector(15 downto 0)
28     );
29 end decod_4x16;
30
31
32 --Architecture (Arquitetura)
33 --implementacoes do projeto
34
35 architecture decod of decod_4x16 is
36
37     -- a definicao inicia por
38     begin
39
40         -- Y = f(A)
41         with A select
42             Y <= "0000000000000001" when "0000",
43                 "0000000000000010" when "0001",
44                 "0000000000000100" when "0010",
45                 "0000000000001000" when "0011",
46                 "0000000000100000" when "0100",
47                 "0000000001000000" when "0101",
48                 "0000000010000000" when "0110",
49                 "0000000100000000" when "0111",
50                 "0000001000000000" when "1000",
51                 "0000001000000000" when "1001",
52                 "0000010000000000" when "1010",
53                 "0000100000000000" when "1011",
54                 "0001000000000000" when "1100",
55                 "0010000000000000" when "1101",
56                 "0100000000000000" when "1110",
57                 "1000000000000000" when "1111",
58                 "0000000000000000" when others;
59     end decod;
60
61     -- a definicao termina por end
```

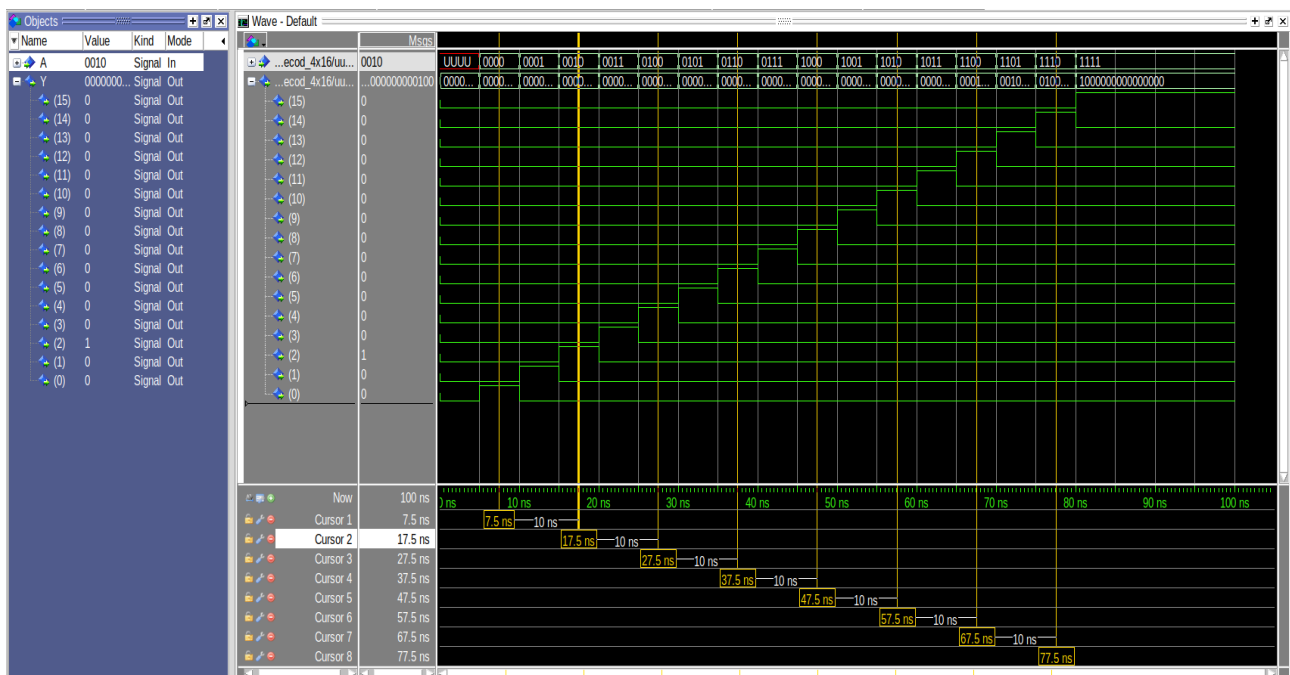
tb_decod_4x16.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 05/03/2021
6
7  -- Implementação de um Decodificador de 4 para 16 bits (1 vetor de 4 bits de entrada e 1 vetor de 16 bits de saída)
8
9  -- *****
10 -- Testbench para simulação funcional do
11 -- Circuito: Decodificador de 1 entrada:
12 --           A Entrada
13 --           Y Saída
14 -- *****
15
16 -- entity (Entidade)
17 -- tb_decod_4x16 uma entidade sem pinos de entrada e saída
18
19 entity tb_decod_4x16 is end;
20
21 --Package (Pacote)
22 --constantes e bibliotecas
23
24 library ieee;
25 use ieee.std_logic_1164.ALL;
26 use std.textio.all;
27 use ieee.numeric_std.all;
28
29
30 -- Architecture (Arquitetura)
31
32 architecture testbench of tb_decod_4x16 is
33
34 -- Declaração do componente decod_4x16, referente a sua arquitetura descrita no arquivo decod_4x16.vhd
35
36 component decod_4x16 is
37     port(
38         A :in std_logic_vector(3 downto 0);
39         Y :out std_logic_vector(15 downto 0)
40     );
41 end component;
42
43 -- Sinais auxiliares para a interconexão ao
44 -- processo de estímulo
45
46 signal a : std_logic_vector(3 downto 0);
47
48 -- Instância do componente decod_4x16 e interconexão do componente ao processo de estímulo
49
50 begin
51     uut: decod_4x16 PORT MAP (A => a, Y => open);
52
53 -- Implementação do processo de estímulo
54
55 estímulo: process
56     begin
57
58         for i in 0 to 15 loop
59             wait for 5 ns;
60             a <= std_logic_vector(to_unsigned(i, 4));
61         end loop;
62         wait;
63
64 -- Fim do processo de estímulo
65
66     end process;
67
68 --Fim da definição da arquitetura
69
70 END;
```

- **Compilação:**



- **Simulação do Código:**



Cursor 1: Em 7,5 ns, A = 0000, Y = 0000000000000001;
 Cursor 2: Em 17,5 ns, A = 0010, Y = 0000000000000100;
 Cursor 3: Em 27,5 ns, A = 0100, Y = 0000000000010000;
 Cursor 4: Em 37,5 ns, A = 0110, Y = 0000000001000000;
 Cursor 5: Em 47,5 ns, A = 1000, Y = 0000000100000000;
 Cursor 6: Em 57,4 ns, A = 1010, Y = 0000010000000000;
 Cursor 7: Em 67,5 ns, A = 1100, Y = 0001000000000000;
 Cursor 8: Em 77,5 ns, A = 1110, Y = 0100000000000000;