

Nome: Gustavo Pereira Chaves
Matrícula: 19/0014113
Turma: 03A
Data: 26/03/2021

Relatório do Experimento 6

Questão 1:

- Documentação:

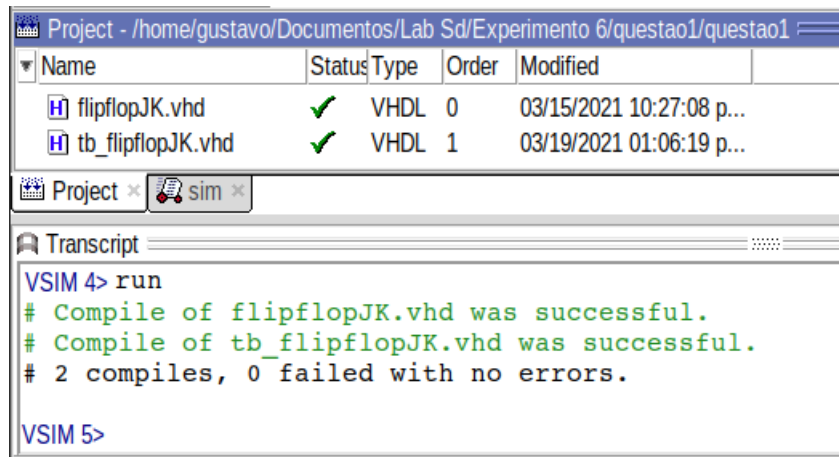
flipflopJK.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 26/03/2021
6
7  -- Implementação de um flip-flop JK gatilhado na borda de subida do clock.
8
9  -- *****
10 -- Circuito: Flip-Flop JK:
11 --           pr  Entrada 1
12 --           clr Entrada 2
13 --           clk Entrada 3
14 --           J   Entrada 4
15 --           K   Entrada 5
16 --           Q   Saída
17 -- *****
18
19 --Package (Pacote)
20 --constantes e bibliotecas
21
22 library IEEE;
23 use IEEE.std_logic_1164.all;
24 use ieee.numeric_std.all;
25
26 --Entity (Entidade)
27 --pinos de entrada e saída
28
29 entity flipflopJK is
30     port (
31         pr, clr, clk, J, K :in std_logic;
32         Q : out std_logic
33     );
34 end flipflopJK;
35
36 --Architecture (Arquitetura)
37 --implementacoes do projeto
38
39 architecture flipflopJK_arch of flipflopJK is
40
41
42 -- Sinal auxiliar para a interconexao
43 signal JK :std_logic_vector(1 downto 0);
44 signal Q_buffer :std_logic := '0';
45
46 -- a definicao inicia por
47 begin
48     JK <= J & K;
49
50 -- Q = f(pr, clr, clk, J, K)
51
52     process(pr, clr, clk)
53     begin
54         if pr = '1' then
55             Q_buffer <= '1';
56         elsif clr = '1' then
57             Q_buffer <= '0';
58         elsif rising_edge(clk) then
59             if JK = "01" then Q_buffer <= '0'; end if;
60             if JK = "10" then Q_buffer <= '1'; end if;
61             if JK = "11" then Q_buffer <= not(Q_buffer); end if;
62         end if;
63     end process;
64     Q <= Q_buffer;
65
66 end flipflopJK_arch;
67 -- a definicao termina por end
```

tb_flipflopJK.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 26/03/2021
6
7  -- Implementação de um flip-flop JK gatilhado na borda de subida do clock.
8
9  -- *****
10 -- Testbench para simulação funcional do
11 -- Circuito: Flip-Flop JK:
12 --           pr Entrada 1
13 --           clr Entrada 2
14 --           clk Entrada 3
15 --           J  Entrada 4
16 --           K  Entrada 5
17 --           Q  Saida
18 -- *****
19
20 -- entity (Entidade)
21 -- tb_flipflopJK uma entidade sem pinos de entrada e saída
22
23 entity tb_flipflopJK is end;
24
25 --Package (Pacote)
26 --constantes e bibliotecas
27
28 library ieee;
29 use ieee.std_logic_1164.ALL;
30 use std.textio.all;
31 use ieee.numeric_std.all;
32
33 -- Architecture (Arquitetura)
34
35 architecture tb_flipflopJK_arch of tb_flipflopJK is
36
37 -- Declaracao do componente flipflopJK, referente a sua arquitetura descrita no arquivo flipflopJK.vhd
38
39 component flipflopJK is
40     port (
41         pr, clr, clk, J, K :in std_logic;
42         Q : out std_logic
43     );
44 end component;
45
46 -- Sinais auxiliares para a interconexao ao
47 -- processo de estimulo
48
49 signal s : std_logic_vector(1 downto 0);
50 signal clk : std_logic := '0';
51 signal jk :std_logic_vector(1 downto 0);
52
53 -- Instancia do componente flipflopJK e interconexao do componente ao processo de estimulo
54
55 begin
56     u1: flipflopJK PORT MAP (pr => s(1), clr => s(0), J => jk(1), K => jk(0), clk => clk, Q => open);
57
58 -- Implementacao do processo de estimulo
59 clk <= not clk after 5 ns;
60
61 estimulo: process
62     begin
63         for i in 0 to 4 loop
64             s <= std_logic_vector(to_unsigned(i, 2));
65             jk <= "00";
66             wait for 10 ns;
67             jk <= "01";
68             wait for 10 ns;
69             jk <= "11";
70             wait for 10 ns;
71             jk <= "10";
72             wait for 10 ns;
73         end loop;
74         wait;
75 -- Fim do processo de estimulo
76
77     end process;
78
79 --Fim da definição da arquitetura
80
81 END;
```

- **Compilação:**



Project - /home/gustavo/Documentos/Lab Sd/Experimento 6/questao1/questao1

Name	Status	Type	Order	Modified
flipflopJK.vhd	✓	VHDL	0	03/15/2021 10:27:08 p...
tb_flipflopJK.vhd	✓	VHDL	1	03/19/2021 01:06:19 p...

Project x sim x

Transcript

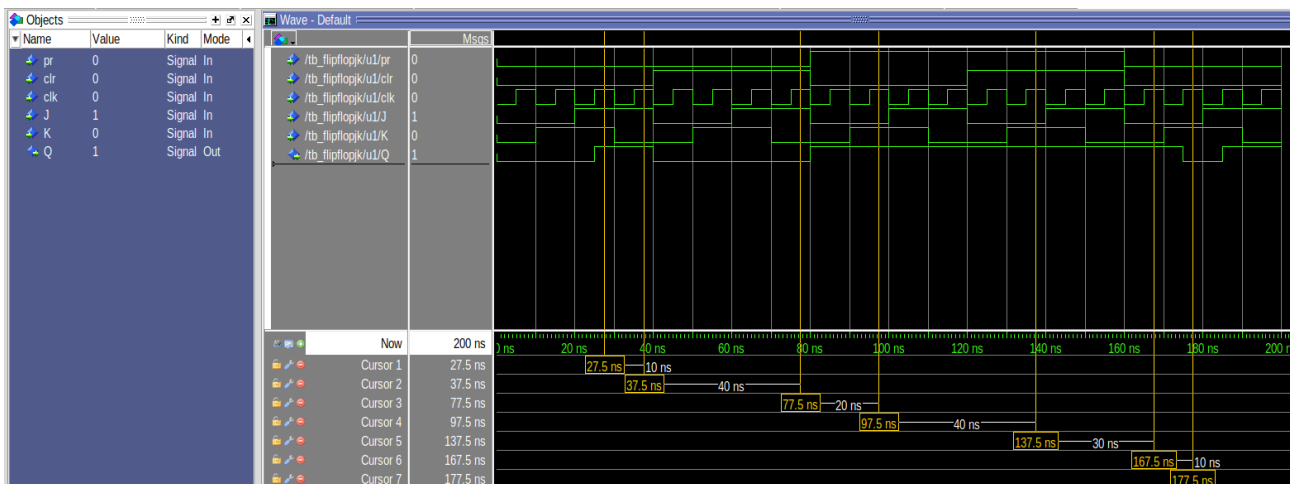
```

VSIM 4> run
# Compile of flipflopJK.vhd was successful.
# Compile of tb_flipflopJK.vhd was successful.
# 2 compiles, 0 failed with no errors.

VSIM 5>

```

- **Simulação do Código:**



Cursor 1: Em 27,5 ns, pr = 0, clr = 0, clk = 1, J = 1, K = 1, Q = 1;
 Cursor 2: Em 37,5 ns, pr = 0, clr = 0, clk = 1, J = 1, K = 0, Q = 1;
 Cursor 3: Em 77,5 ns, pr = 0, clr = 1, clk = 1, J = 1, K = 0, Q = 0;
 Cursor 4: Em 97,5 ns, pr = 1, clr = 0, clk = 1, J = 0, K = 1, Q = 1;
 Cursor 5: Em 137,5 ns, pr = 1, clr = 1, clk = 1, J = 0, K = 1, Q = 1;
 Cursor 6: Em 167,5 ns, pr = 0, clr = 0, clk = 1, J = 0, K = 0, Q = 1;
 Cursor 7: Em 177,5 ns pr = 0, clr = 0, clk = 1, J = 0, K = 1, Q = 0;

Questão 2:

- Documentação:

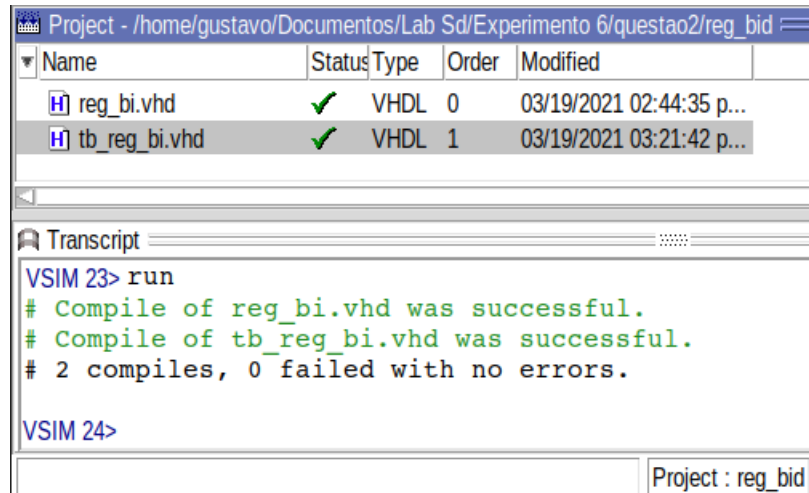
reg_bi.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 26/03/2021
6
7  -- Implementação de um registrador de deslocamento bidirecional de 4 bits gatilhado pela borda de subida do clock.
8
9  -- *****
10 -- Circuito: Registrador de deslocamento bidirecional:
11 --          clk  Entrada 1
12 --          rst  Entrada 2
13 --          load  Entrada 3
14 --          dir  Entrada 4
15 --          D    Entrada 5
16 --          L    Entrada 6
17 --          R    Entrada 7
18 --          Q    Saída
19 -- *****
20
21 --Package (Pacote)
22 --constantes e bibliotecas
23
24 library IEEE;
25 use IEEE.std_logic_1164.all;
26 use ieee.numeric_std.all;
27
28 --Entity (Entidade)
29 --pinos de entrada e saída
30
31 entity reg_bi is
32     port (
33         clk, rst, load, dir, L, R :in std_logic;
34         D : in std_logic_vector(3 downto 0);
35         Q : out std_logic_vector(3 downto 0)
36     );
37 end reg_bi;
38
39 --Architecture (Arquitetura)
40 --implementacoes do projeto
41
42 architecture reg_bi_arch of reg_bi is
43
44
45     -- Sinal auxiliar para a interconexao
46     signal Q_buffer :std_logic_vector(3 downto 0);
47
48     -- a definicao inicia por
49     begin
50
51     -- Q = f(clk, rst, load, dir, D, L, R)
52
53         process(clk)
54         begin
55             if (rising_edge(clk)) then
56                 if (rst = '1') then Q_buffer <= "0000";
57                 elsif (load = '1') then Q_buffer <= D;
58                 elsif (dir = '0') then Q_buffer <= std_logic_vector(unsigned(Q_buffer) sll 1); Q_buffer(0) <= L;
59                 elsif (dir = '1') then Q_buffer <= std_logic_vector(unsigned(Q_buffer) srl 1); Q_buffer(3) <= R;
60                 end if;
61             end if;
62         end process;
63         Q <= Q_buffer;
64
65     end reg_bi_arch;
66     -- a definicao termina por end
```

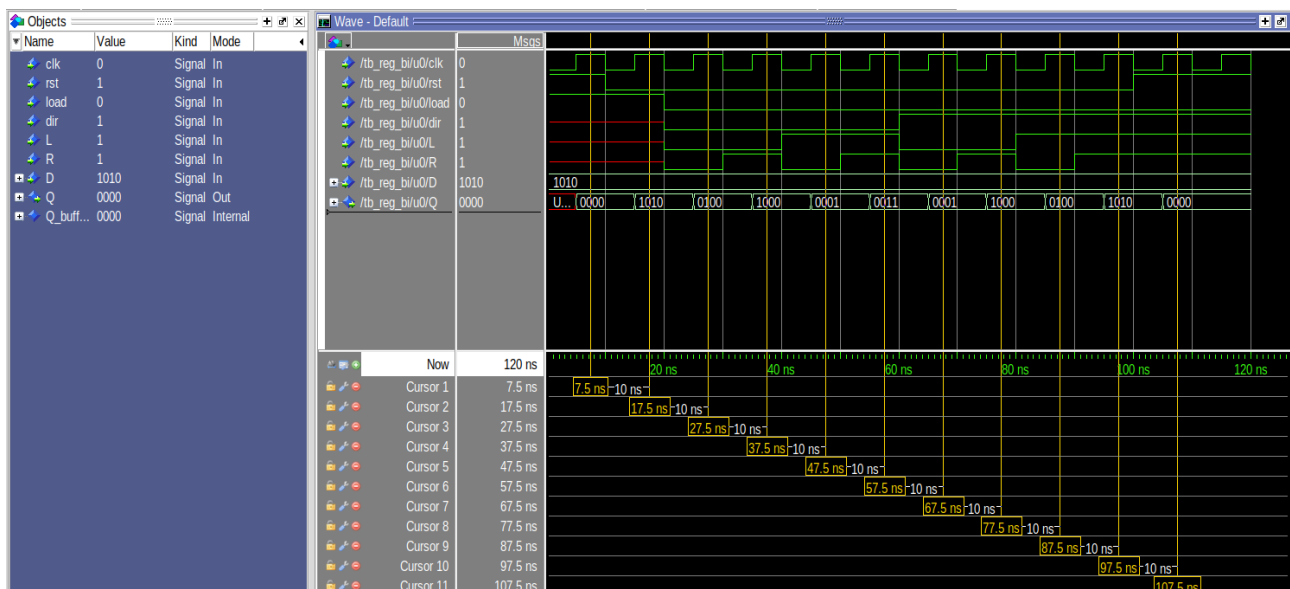
tb_reg_bi.vhd

```
1  -- Universidade de Brasília
2  -- Laboratório de Sistemas Digitais
3  -- Autor : Gustavo Pereira Chaves
4  -- Matrícula: 19/0014113
5  -- Data : 26/03/2021
6
7  -- Implementação de um registrador de deslocamento bidirecional de 4 bits gatilhado pela borda de subida do clock.
8
9  -- *****
10 -- Testbench para simulação funcional do
11 -- Circuito: Registrador de deslocamento bidirecional:
12 --          clk  Entrada 1
13 --          rst  Entrada 2
14 --          load  Entrada 3
15 --          dir  Entrada 4
16 --          D    Entrada 5
17 --          L    Entrada 6
18 --          R    Entrada 7
19 --          Q     Saida
20 -- *****
21
22 -- entity (Entidade)
23 -- tb_reg_bi uma entidade sem pinos de entrada e saída
24
25 entity tb_reg_bi is end;
26
27 --Package (Pacote)
28 --constantes e bibliotecas
29
30 library ieee;
31 use ieee.std_logic_1164.ALL;
32 use std.textio.all;
33 use ieee.numeric_std.all;
34
35 -- Architecture (Arquitetura)
36
37 architecture tb_reg_bi_arch of tb_reg_bi is
38
39 -- Declaracao do componente reg_bi, referente a sua arquitetura descrita no arquivo reg_bi.vhd
40
41 component reg_bi is
42     port (
43         clk, rst, load, dir, L, R :in std_logic;
44         D : in std_logic_vector(3 downto 0);
45         Q : out std_logic_vector(3 downto 0)
46     );
47 end component;
48
49 -- Sinais auxiliares para a interconexao ao
50 -- processo de estimulo
51
52 signal clk : std_logic := '0';
53 signal rst, load, dir : std_logic;
54 signal LR : std_logic_vector(1 downto 0);
55 signal d : std_logic_vector(3 downto 0);
56
57 -- Instancia do componente flipflopJK e interconexao do componente ao processo de estimulo
58
59 begin
60     u0: reg_bi port map (clk, rst, load, dir, LR(1), LR(0), D, Q => open);
61
62 -- Implementacao do processo de estimulo
63 clk <= not clk after 5 ns;
64
65 estimulo: process
66 begin
67     D <= "1010"; rst <= '1'; load <= '1';
68     wait for 10 ns;
69     rst <= '0';
70     wait for 10 ns;
71
72     dir <= '0'; load <= '0';
73     for i in 0 to 3 loop
74         LR <= std_logic_vector(to_unsigned(i, 2));
75         wait for 10 ns;
76     end loop;
77
78     dir <= '1';
79     for i in 0 to 3 loop
80         LR <= std_logic_vector(to_unsigned(i, 2));
81         wait for 10 ns;
82     end loop;
83
84     rst <= '1';
85     wait;
86 -- Fim do processo de estimulo
87 end process;
88
89 --Fim da definição da arquitetura
90 END;
```

- **Compilação:**



- **Simulação do Código:**



Cursor 1: Em 7,5 ns, clk = 1, rst = 1, load = 1, dir = 0, L = 0, R = 0, D = 1010, Q = 0000;
 Cursor 2: Em 17,5 ns, clk = 1, rst = 0, load = 1, dir = 0, L = 0, R = 0, D = 1010, Q = 1010;
 Cursor 3: Em 27,5 ns, clk = 1, rst = 0, load = 0, dir = 0, L = 0, R = 0, D = 1010, Q = 0100;
 Cursor 4: Em 37,5 ns, clk = 1, rst = 0, load = 0, dir = 0, L = 0, R = 1, D = 1010, Q = 1000;
 Cursor 5: Em 47,5 ns, clk = 1, rst = 0, load = 0, dir = 0, L = 1, R = 0, D = 1010, Q = 0001;
 Cursor 6: Em 57,5 ns, clk = 1, rst = 0, load = 0, dir = 0, L = 1, R = 1, D = 1010, Q = 0011;
 Cursor 7: Em 67,5 ns, clk = 1, rst = 0, load = 0, dir = 1, L = 0, R = 0, D = 1010, Q = 0001;
 Cursor 8: Em 77,5 ns, clk = 1, rst = 0, load = 0, dir = 1, L = 0, R = 1, D = 1010, Q = 1000;
 Cursor 9: Em 87,5 ns, clk = 1, rst = 0, load = 0, dir = 1, L = 1, R = 0, D = 1010, Q = 0100;
 Cursor 10: Em 97,5 ns, clk = 1, rst = 0, load = 0, dir = 1, L = 1, R = 1, D = 1010, Q = 1010;
 Cursor 11: Em 107,5 ns, clk = 1, rst = 1, load = 0, dir = 1, L = 1, R = 1, D = 1010, Q = 0000;