

COUNTER WITH EXTERNAL INPUT

LAB 05 REPORT

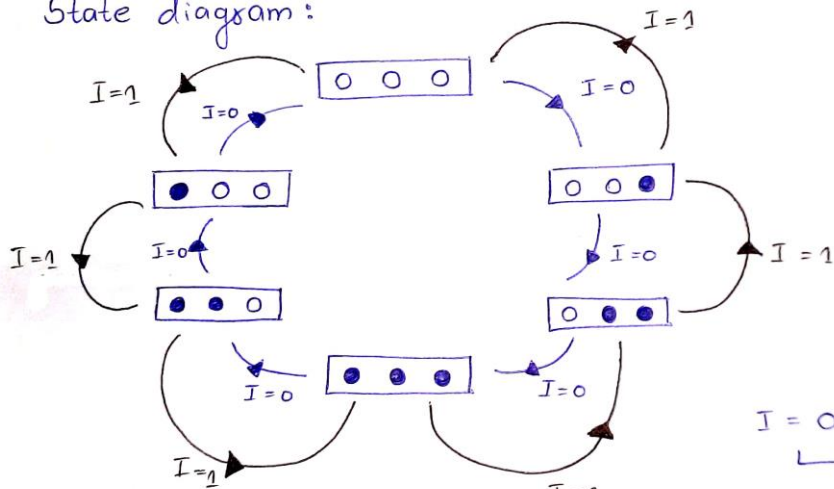
Task Assigned:

- Identify the functionality of the 3 bit counter according to the state diagram given and construct it's excitation table using D flipflop's excitation table.
- Using Karnaugh maps, obtain the simplified logic expression for each bit in the counter.
- Design and develop the counter in Vivado software according to following steps:
 - Develop a source code for a D flipflop in VHDL and verify it's functionality via a simulation.
 - Develop a source code for a slow clock in VHDL and verify it's functionality via a simulation.
 - Combining above two circuits (a slow clock and 3 D flipflops), develop a source code for the 3 bit counter and verify it's functionality via a simulation

Excitation table for the 3 bit counter:

I (Button)	Current Stage (Q_t)			Next Stage (Q_{t+1})			D_2	D_1	D_0
	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0			
0	0	0	0	0	0	1	0	0	1
0	0	0	1	0	1	1	0	1	1
0	0	1	0	X	X	X	X	X	X
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	0	0	0	0
0	1	0	1	X	X	X	X	X	X
0	1	1	0	1	0	0	1	0	0
0	1	1	1	1	1	0	1	1	0
1	0	0	0	1	0	0	1	0	0
1	0	0	1	0	0	0	0	0	0
1	0	1	0	X	X	X	X	X	X
1	0	1	1	0	0	1	0	0	1
1	1	0	0	1	1	0	1	1	0
1	1	0	1	X	X	X	X	X	X
1	1	1	0	1	1	1	1	1	1
1	1	1	1	0	1	1	0	1	1

State diagram:



$I = 0$
 \hookrightarrow Count upwards
 $I = 1$
 \hookrightarrow Count downwards

Derived Karnaugh maps & simplified logic expressions:

$I \backslash Q_2 \backslash Q_1 \backslash Q_0$	00	01	11	10
00	0	0	1	X
01	0	X	1	1
11	1	X	0	1
10	1	0	0	X

$$D_2 = \bar{I} \cdot Q_1 + \bar{Q}_0 \cdot I + Q_1 \cdot \bar{Q}_0$$

$I \backslash Q_2 \backslash Q_1 \backslash Q_0$	00	01	11	10
00	0	1	1	X
01	0	X	1	0
11	1	X	1	1
10	0	0	0	X

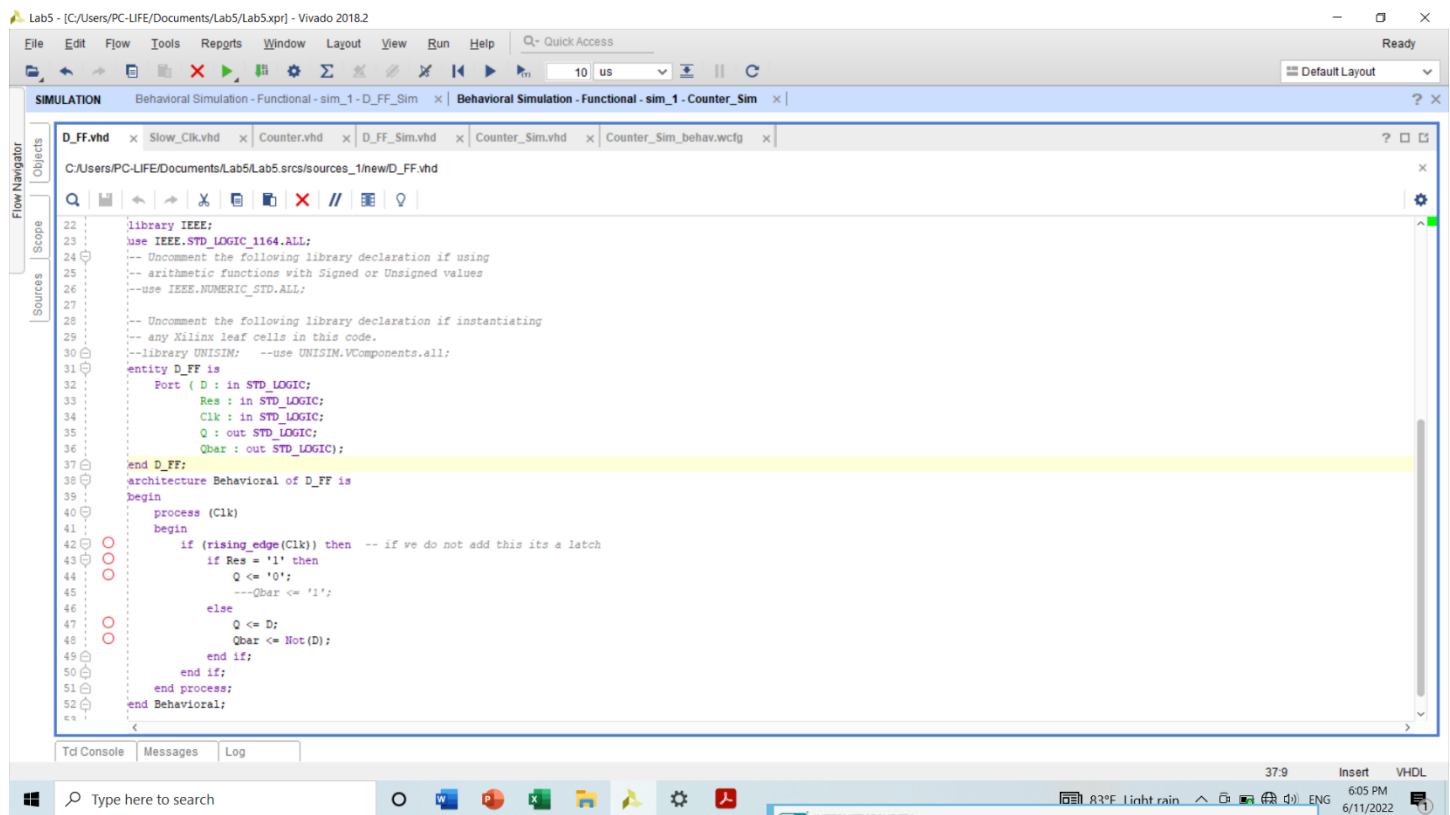
$$D_1 = Q_0 \cdot \bar{I} + Q_0 \cdot Q_2 + I \cdot Q_2$$

$I \backslash Q_2 \backslash Q_1 \backslash Q_0$	00	01	11	10
00	1	1	1	X
01	0	X	0	0
11	0	X	1	1
10	0	0	1	X

$$D_0 = Q_1 \cdot I + Q_1 \cdot \bar{Q}_2 + \bar{I} \cdot \bar{Q}_2$$

VHDL Codes:

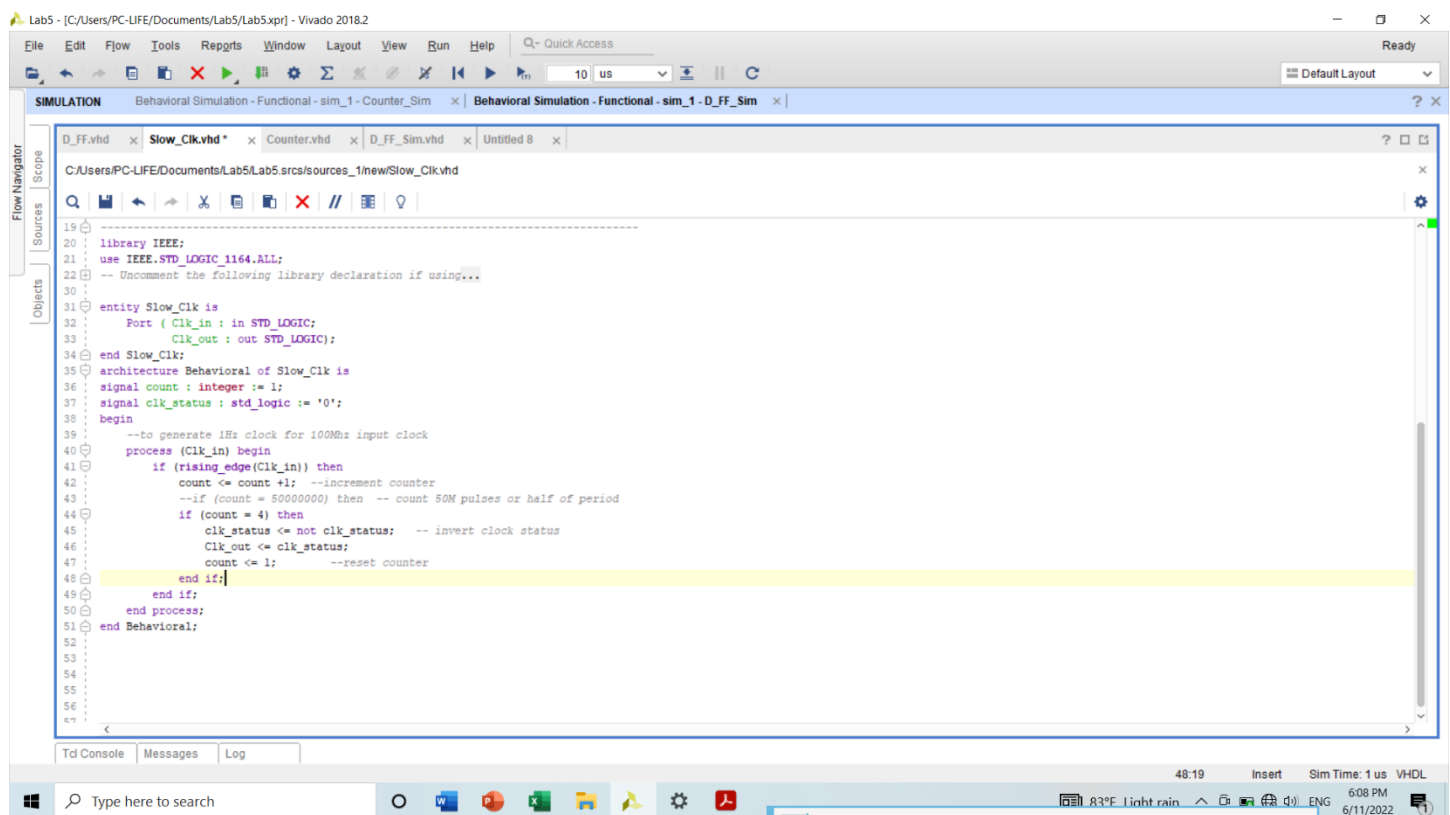
01)VHDL SOURCE CODE FOR D FLIPFLOP



The screenshot shows the Vivado 2018.2 IDE with the VHDL source code for a D Flip-Flop. The code is located in the file `C:/Users/PC-LIFE/Documents/Lab5/Lab5.srcs/sources_1/new/D_FF.vhd`. The code defines an entity `D_FF` with inputs `D`, `Res`, `Clk`, and `Q`, and an output `Qbar`. The architecture is behavioral and implements a D flip-flop with a reset and a clock input.

```
1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;
3 -- Uncomment the following library declaration if using
4 -- arithmetic functions with Signed or Unsigned values
5 --use IEEE.NUMERIC_STD.ALL;
6
7 -- Uncomment the following library declaration if instantiating
8 -- any Xilinx leaf cells in this code.
9 --library UNISIM; --use UNISIM.VComponents.all;
10
11 entity D_FF is
12     Port ( D : in STD_LOGIC;
13           Res : in STD_LOGIC;
14           Clk : in STD_LOGIC;
15           Q : out STD_LOGIC;
16           Qbar : out STD_LOGIC);
17 end D_FF;
18
19 architecture Behavioral of D_FF is
20 begin
21     process (Clk)
22     begin
23         if (rising_edge(Clk)) then -- if we do not add this its a latch
24             if Res = '1' then
25                 Q <= '0';
26                 Qbar <= '1';
27             else
28                 Q <= D;
29                 Qbar <= Not(D);
30             end if;
31         end if;
32     end process;
33 end Behavioral;
```

02)VHDL SOURCE CODE FOR SLOW CLOCK



The screenshot shows the Vivado 2018.2 IDE with the VHDL source code for a Slow Clock generator. The code is located in the file `C:/Users/PC-LIFE/Documents/Lab5/Lab5.srcs/sources_1/new/Slow_Clk.vhd`. The code defines an entity `Slow_Clk` with inputs `Clk_in` and `Clk_out`. The architecture is behavioral and implements a slow clock generator that divides the input clock by 500,000.

```
19 library IEEE;
20 use IEEE.STD_LOGIC_1164.ALL;
21 -- Uncomment the following library declaration if using...
22
23 entity Slow_Clk is
24     Port ( Clk_in : in STD_LOGIC;
25           Clk_out : out STD_LOGIC);
26 end Slow_Clk;
27
28 architecture Behavioral of Slow_Clk is
29     signal count : integer := 1;
30     signal clk_status : std_logic := '0';
31 begin
32     --to generate 1Hz clock for 100MHz input clock
33     process (Clk_in) begin
34         if (rising_edge(Clk_in)) then
35             count <= count + 1; --increment counter
36             --if (count = 50000000) then -- count 50M pulses or half of period
37                 if (count = 4) then
38                     clk_status <= not clk_status; -- invert clock status
39                     Clk_out <= clk_status;
40                     count <= 1; --reset counter
41                 end if;
42             end if;
43         end process;
44 end Behavioral;
```

03)VHDL SOURCE CODE FOR 3 BIT COUNTER

Lab5 - [C:/Users/PC-LIFE/Documents/Lab5/Lab5.xpr] - Vivado 2018.2

File Edit Flow Tools Reports Window Layout View Run Help Q- Quick Access Ready

Behavioral Simulation - Functional - sim_1 - Counter_Sim x Behavioral Simulation - Functional - sim_1 - D_FF_Sim x

D_FF.vhd x Slow_Clk.vhd * Counter.vhd * D_FF_Sim.vhd x Untitled 8 x

C:/Users/PC-LIFE/Documents/Lab5/srcs/sources_1/new/Counter.vhd

```
19 library IEEE;
20 use IEEE.STD_LOGIC_1164.ALL;
21 -- Uncomment the following library declaration if using
22 -- arithmetic functions with Signed or Unsigned values
23 --use IEEE.NUMERIC_STD.ALL;
24
25 -- Uncomment the following library declaration if instantiating
26 -- any Xilinx leaf cells in this code.
27 --library UNISIM;
28 --use UNISIM.VComponents.all;
29
30 entity Counter is
31     Port ( Dir : in STD_LOGIC;
32           Clk : in STD_LOGIC;
33           Res : in STD_LOGIC;
34           Q0 : out STD_LOGIC;
35           Q1 : out STD_LOGIC;
36           Q2 : out STD_LOGIC);
37 end Counter;
38
39 architecture Behavioral of Counter is
40
41     component D_FF
42     Port ( D : in STD_LOGIC;
43           Res : in STD_LOGIC;
44           Clk : in STD_LOGIC;
45           Q : out STD_LOGIC;
46           Qbar : out STD_LOGIC);
47 end component;
```

Lab5 - [C:/Users/PC-LIFE/Documents/Lab5/Lab5.xpr] - Vivado 2018.2

File Edit Flow Tools Reports Window Layout View Run Help Q- Quick Access Ready

Behavioral Simulation - Functional - sim_1 - Counter_Sim x Behavioral Simulation - Functional - sim_1 - D_FF_Sim x

D_FF.vhd x Slow_Clk.vhd * Counter.vhd * D_FF_Sim.vhd x Untitled 8 x

C:/Users/PC-LIFE/Documents/Lab5/srcs/sources_1/new/Counter.vhd

```
49 component Slow_Clk
50     Port ( Clk_in : in STD_LOGIC;
51           Clk_out : out STD_LOGIC);
52 end component;
53
54 SIGNAL D0,D1,D2 : STD_LOGIC;
55 SIGNAL Q_0,Q_1,Q_2 : STD_LOGIC;
56 SIGNAL Clk_slow : STD_LOGIC;
57
58 begin
59     Slow_Clk0 : Slow_Clk port map (
60         Clk_in => Clk,
61         Clk_out => Clk_slow);
62     D_FF_0 : D_FF PORT MAP (
63         D => D0,
64         Res => Res,
65         Clk => Clk_slow,
66         Q => Q_0);
67     D_FF_1 : D_FF PORT MAP (
68         D => D1,
69         Res => Res,
70         Clk => Clk_slow,
71         Q => Q_1);
72     D_FF_2 : D_FF PORT MAP (
73         D => D2,
74         Res => Res,
75         Clk => Clk_slow,
76         Q => Q_2);
77
78     --Removing timing hazards
79     --D0 <= (Q_1 AND Dir) OR (Q_1 AND NOT(Q_2)) OR (NOT(Dir)AND NOT(Q_2));
80     --D1 <= (Q_0 AND NOT(Dir)) OR (Q_0 AND Q_2) OR (Dir AND Q_2);
81     --D2 <= (Q_1 AND NOT(Dir)) OR (Q_1 AND NOT(Q_0)) OR (Dir AND NOT(Q_0));
82
83     --Without considering timing hazards
84     D0 <= (Q_1 AND Dir) OR (NOT(Dir)AND NOT(Q_2));
85     D1 <= (Q_0 AND NOT(Dir)) OR (Dir AND Q_2);
86     D2 <= (Q_1 AND NOT(Dir)) OR (Dir AND NOT(Q_0));
87     Q0 <= Q_0;
88     Q1 <= Q_1;
89     Q2 <= Q_2;
90 end Behavioral;
```

Td Console Messages Log

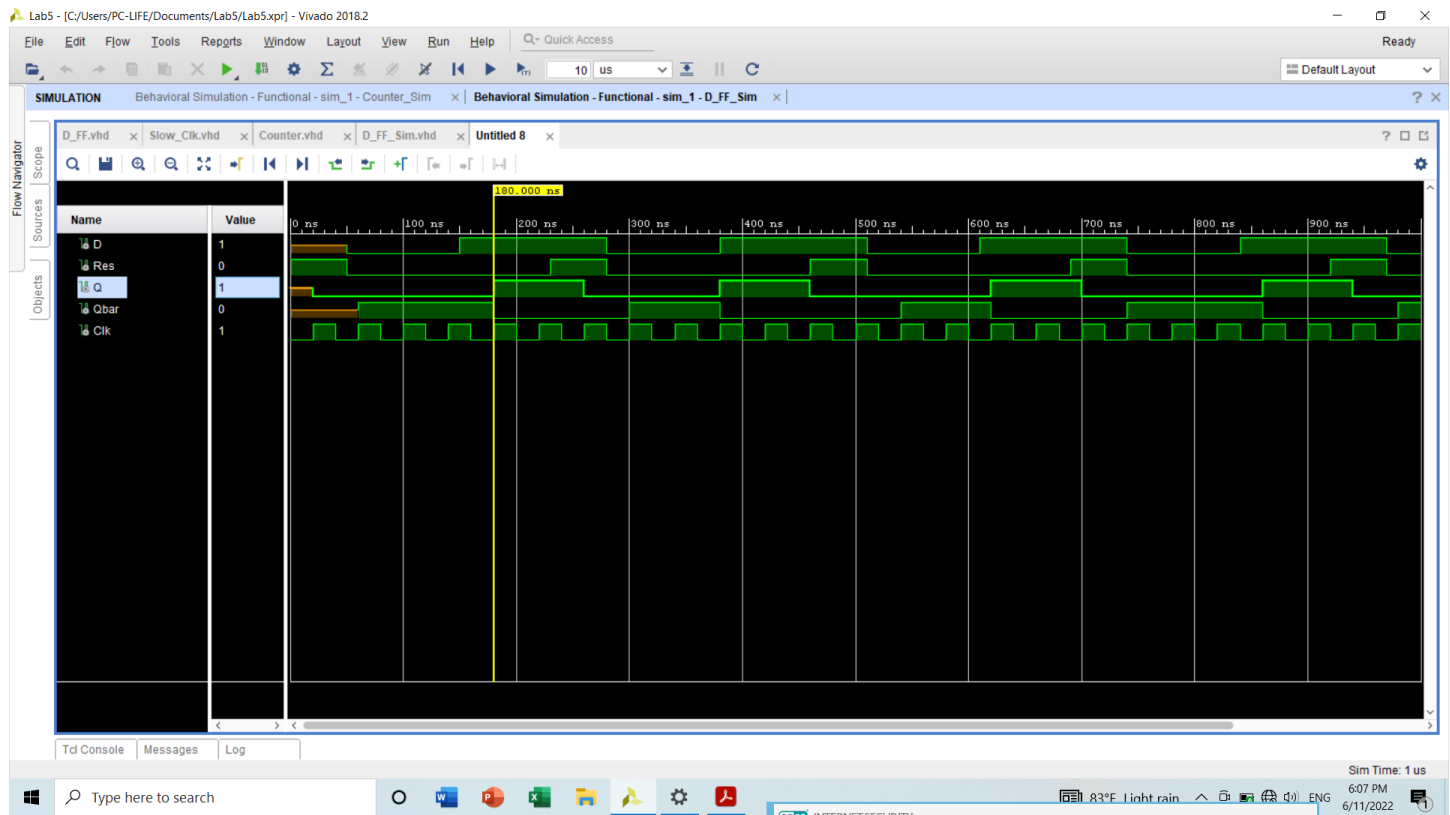
33:30 Insert VHDL

Type here to search

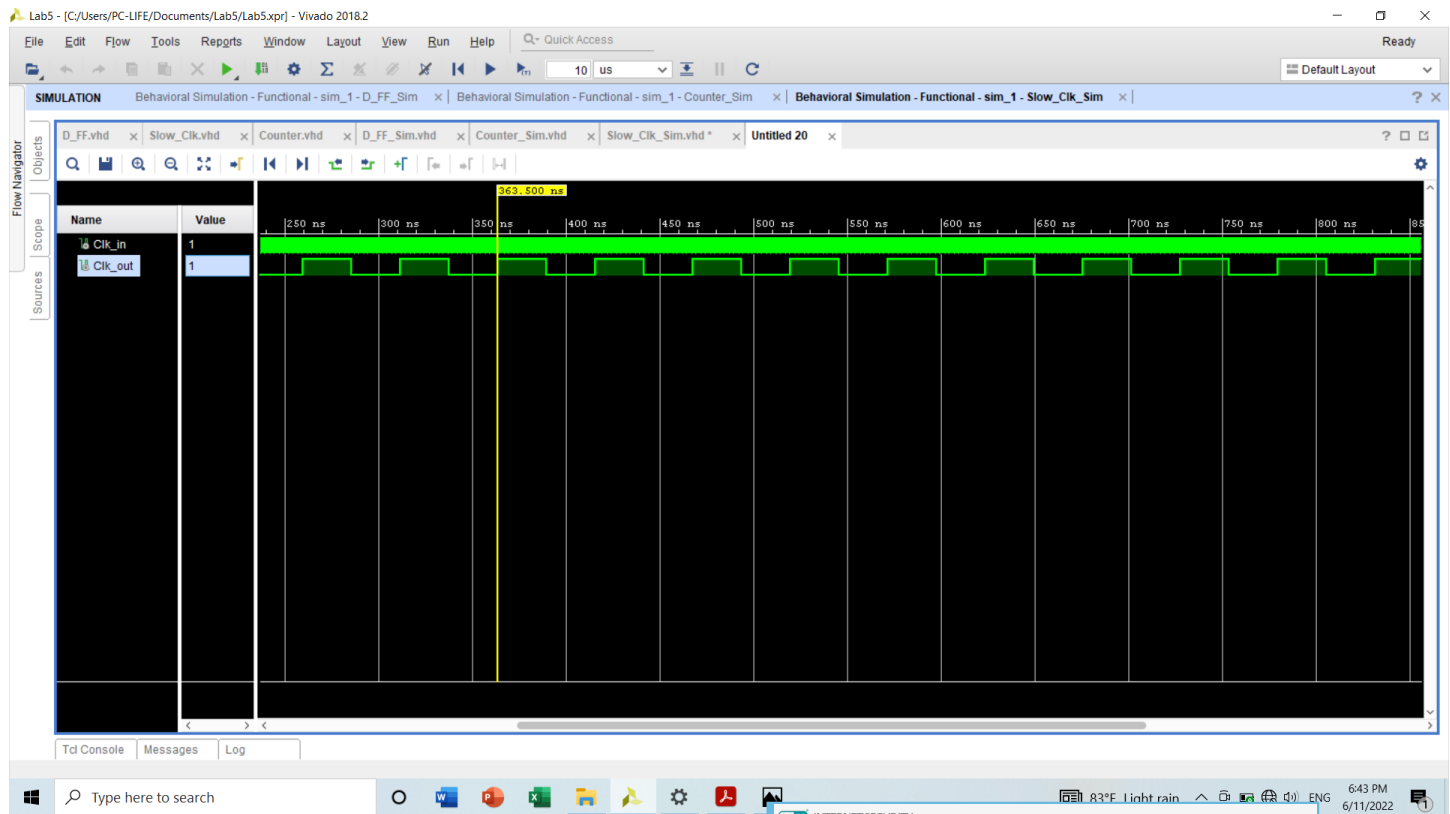
83°F Light rain 6:11 PM 6/11/2022

Timing Diagrams:

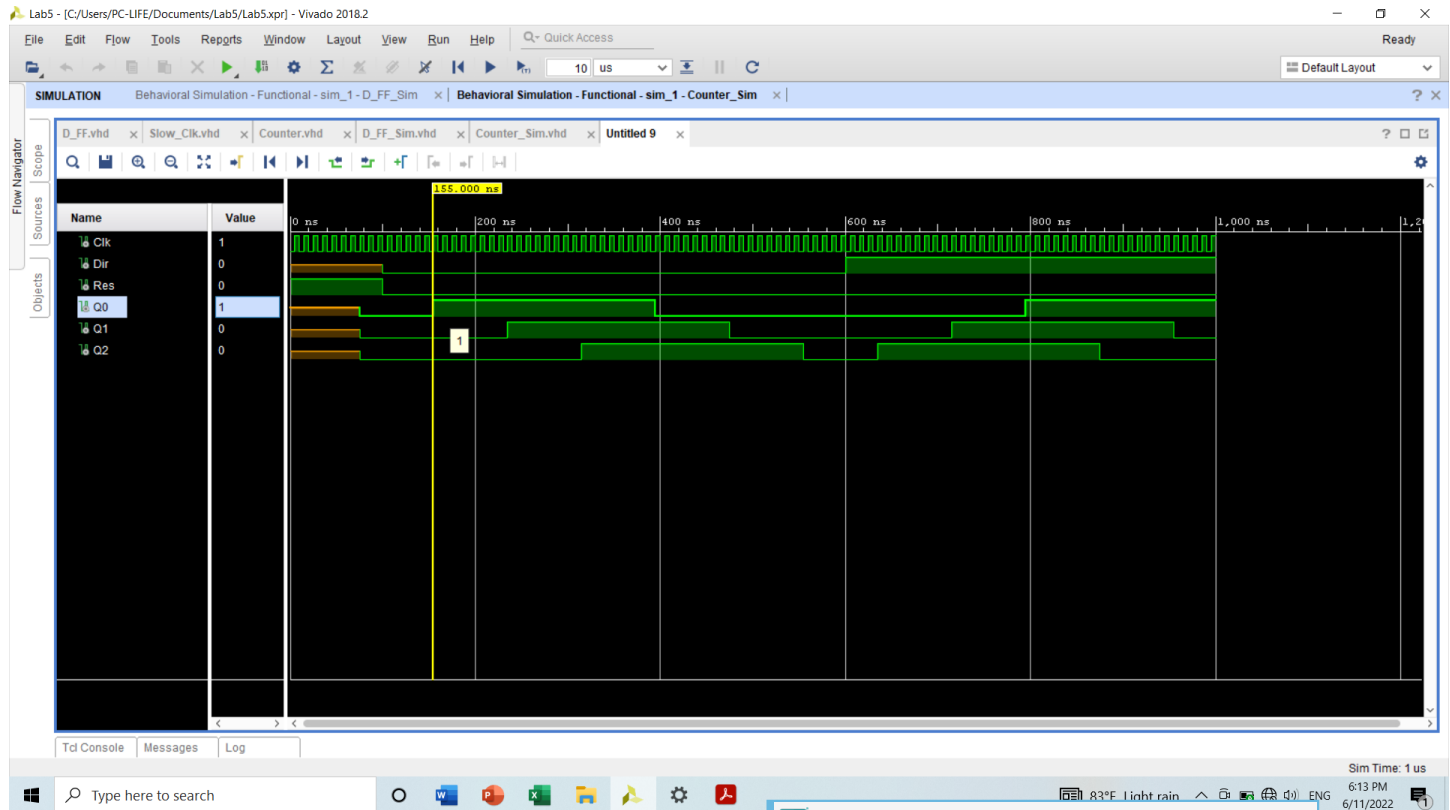
01)TIMING DIAGRAM FOR D FLIPFLOP



02)TIMING DIAGRAM FOR SLOW CLOCK



03)TIMING DIAGRAM FOR 3 BIT COUNTER



Test bench codes for simulation:

01)TB FOR D FLIPFLOP'S SIMULATION

Lab5 - [C:/Users/PC-LIFE/Documents/Lab5/Lab5.xpr] - Vivado 2018.2

File Edit Flow Tools Reports Window Layout View Run Help Q- Quick Access Ready

Behavioral Simulation - Functional - sim_1 - Counter_Sim x | Behavioral Simulation - Functional - sim_1 - D_FF_Sim x |

D_FF.vhd x Slow_Clk.vhd x Counter.vhd x D_FF_Sim.vhd x Untitled 8 x

C:/Users/PC-LIFE/Documents/Lab5/src/tbnewD_FF_Sim.vhd

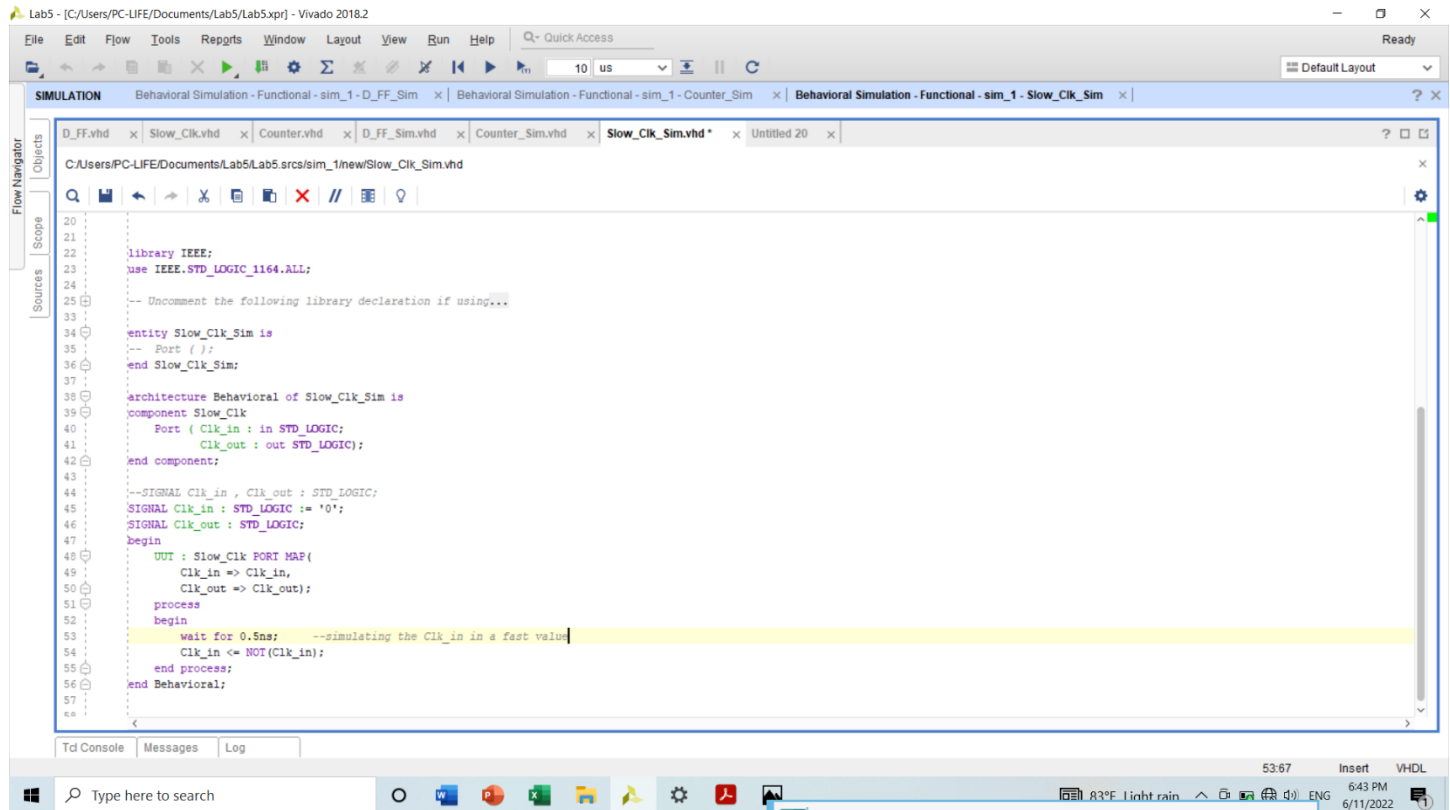
```
22 library IEEE;
23 use IEEE.STD_LOGIC_1164.ALL;
24
25 -- Comment the following library declaration if using
26 -- arithmetic functions with Signed or Unsigned values
27 --use IEEE.NUMERIC_STD.ALL;
28
29 -- Comment the following library declaration if instantiating
30 -- any UTLIB user cells in this code.
31 --library UNISIM;
32 --use UNISIM.VComponents.all;
33
34 entity D_FF_Sim is
35     -- Port (I/O)
36     port (
37         D : in STD_LOGIC;
38         Res : in STD_LOGIC;
39         Clk : in STD_LOGIC;
40         Q : out STD_LOGIC;
41         Qbar : out STD_LOGIC
42     );
43 end entity D_FF_Sim;
44
45 -- SIGNAL D, Res, Q, Qbar : std_logic;
46 SIGNAL Clk : std_logic := '0';
47
48 begin
49     uut: D_FF_Sim port map (
50         D => D,
51         Res => Res,
52         Clk => Clk,
53         Q => Q,
54         Qbar => Qbar);
55
56 --simulation for clock signal
57 process
58     begin
59         wait for 20ns;
60         Clk <= NOT(Clk);
61     end process;
62
63 process
64     begin
65         Res <= '1'; -- D <= '0'
66         wait for 50ns;
67         Res <= '0';
68         D <= '0';
69         wait for 100ns;
70         D <= '1';
71         wait for 80ns;
72         D <= '0';
73         --wait for 100ns;
74     end process;
75 end process;
76 end Behavioral;
```

Tcl Console Messages Log

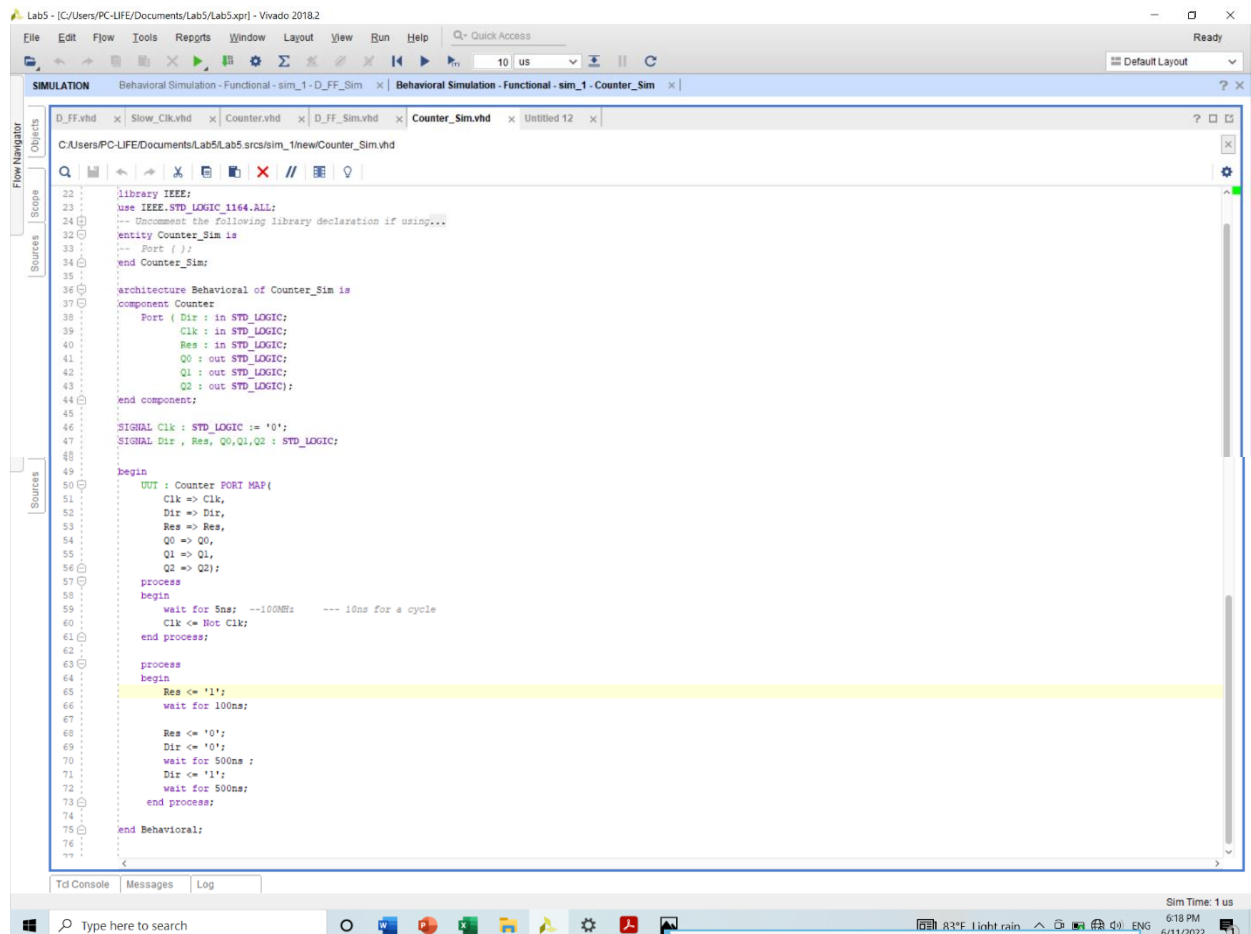
Sim Time: 1 us

6:08 PM 6/11/2022

02)TB FOR SLOW CLOCK'S SIMULATION



03)TB FOR 3 BIT COUNTER'S SIMULATION



Conclusions:

- Sequential logic circuits (**We used D flipflops for the lab**) allow it's output to be depend not only on the external input given but also on the past sequence of outputs.

In the lab, a 3 bit counter is developed using 3 D flipflops which could maintain a memory of past outputs and also could response to the external input on the counting direction whether it's clockwise or anticlockwise.

- Clock divider (**Slow_Clk in our lab**) allows to take an input signal of frequency f_{in} and generate an output signal of a frequency $f_{out} = \frac{f_{in}}{n}$ where n is an integer.
- Vivado's simulation allows to verify the functionality of the source codes of the circuits designed.