# Lab 5 – Counter with External Input

## CS1050 Computer Organization and Digital Design

**Name: Oshadi Saumya Perera**

**Index no: 200458M**

## ➢ Lab Task

- Using the Excitation table of a D Flip-Flop, complete the Excitation table for the counter.
- Identify the inputs to D Flip-Flops $D_0$, $D_1$ and $D_2$ using Karnaugh Maps.
- Building a D Flip-Flop.
- Building the slow down clock.
- Building a 3-bit counter with an external input using D Flip-Flops. Count in clockwise and anticlockwise directions based on an external input.
- Verify its functionality via simulation and on the development board.

## ➢ Excitation table

| Button | Current State ($Q_t$) | | | Next State ($Q_{t+1}$) | | | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|---|---|
| B | $Q_2$ | $Q_1$ | $Q_0$ | $Q_2$ | $Q_1$ | $Q_0$ | | | |
| 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 |

## ➢ Karnaugh Maps

$D_0$

| $Q_1Q_0$<br>$BQ_2$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | X | 1 |
| 01 | 0 | X | 0 | 0 |
| 11 | 0 | X | 1 | 1 |
| 10 | 0 | 0 | 1 | X |

$D_0 = B'.Q_2' + B.Q_1$

$D_1$

| $Q_1Q_0$<br>$BQ_2$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | 1 | X |
| 01 | 0 | X | 1 | 0 |
| 11 | 1 | X | 1 | 1 |
| 10 | 0 | 0 | 0 | X |

$D_1 = B.Q_2 + B'.Q_0 + Q_2.Q_0$

$D_2$

| $Q_1Q_0$<br>$BQ_2$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 1 | X |
| 01 | 0 | X | 1 | 1 |
| 11 | 1 | X | 0 | 1 |
| 10 | 1 | 0 | 0 | X |

$D_2 = B'.Q_1 + Q_1.Q_0' + B.Q_0'$

- VHDL codes

- Design Source Code – D Flip-Flop (D_FF.vhd)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/09/2022 10:37:59 AM
-- Design Name:
-- Module Name: D_FF - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
```

```
entity D_FF is
    Port ( D : in STD_LOGIC;
           Res : in STD_LOGIC;
           Clk : in STD_LOGIC;
           Q : out STD_LOGIC;
           Qbar : out STD_LOGIC);
end D_FF;

architecture Behavioral of D_FF is

begin

    process (Clk)
    begin
        if (rising_edge(Clk)) then
            if Res = '1' then
                Q <= '0';
                Qbar <= '1';
            else
                Q <= D;
                Qbar <= NOT(D);
            end if;
        end if;
    end process;

end Behavioral;
```

- Test Bench Code – D Flip-Flop (D_FF_Sim.vhd)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/09/2022 10:44:09 AM
-- Design Name:
-- Module Name: D_FF_Sim - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
```

```
entity D_FF_Sim is
--  Port ( );
end D_FF_Sim;

architecture Behavioral of D_FF_Sim is
component D_FF
   Port ( D : in STD_LOGIC;
         Res : in STD_LOGIC;
         Clk : in STD_LOGIC;
         Q : out STD_LOGIC;
         Qbar : out STD_LOGIC);
end component;
signal D,Res,Q,Qbar : STD_LOGIC;
signal Clk : STD_LOGIC := '0';
begin
   UUT : D_FF
   PORT MAP(
      D => D,
      Res => Res,
      Clk => Clk,
      Q => Q,
      Qbar => Qbar
   );

   process
   begin
      wait for 20ns;
      Clk <= NOT(Clk);
```

```
    end process;

    process
    begin
        Res <= '1'; --Q <= 0
        wait for 50ns;
        Res <= '0';

        D <= '0';
        wait for 50ns;

        D <= '1';
        wait for 74ns;

        D <= '0';
        wait for 80ns;

    end process;

end Behavioral;
```

- Design Source Code – Slow down clock (Slow_Clk.vhd)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/09/2022 02:47:51 PM
-- Design Name:
-- Module Name: Slow_Clk - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
```

```
entity Slow_Clk is
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
end Slow_Clk;

architecture Behavioral of Slow_Clk is

signal count : integer :=1;
signal clk_status : STD_LOGIC := '0';

begin
    --For 100MHz input clock this generates 1Hz clock
    process (Clk_in)
    begin
        if(rising_edge(Clk_in)) then
            count <= count+1;   --Increment counter
            if(count = 50000000) then     --Count 50M pulses (1/2 of period)
            --if(count = 4) then
                clk_status <= NOT(clk_status);    --Invert clock status
                Clk_out <= clk_status;
                count <= 1;    --Reset counter
            end if;

         end if;
    end process;



end Behavioral;
```

- Test Bench Code – Slow down clock (Slow_Clk_Sim)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/14/2022 03:41:15 PM
-- Design Name:
-- Module Name: Slow_Clk_Sim - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
```

```
entity Slow_Clk_Sim is
--  Port ( );
end Slow_Clk_Sim;

architecture Behavioral of Slow_Clk_Sim is
component Slow_Clk
    port(
        Clk_in : in STD_LOGIC;
        Clk_out : out STD_LOGIC
    );
end component;
signal Clk_100 : STD_LOGIC := '0';
signal Clk_25 : STD_LOGIC;
constant Clk_100_period : time := 20ns;

begin
    UUT : Slow_Clk
    PORT MAP(
        Clk_in => Clk_100,
        Clk_out => Clk_25
    );
    Clk_100_process :process
    begin
        Clk_100 <= '0';
        wait for Clk_100_period/2;
        Clk_100 <= '1';
        wait for Clk_100_period/2;
```

```
  end process;
  stim_proc :process
  begin
    wait for 100ns;
    wait for Clk_100_period*10;
    wait;
  end process;
end Behavioral;
```

- Design Source Code – Counter (Counter.vhd)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/09/2022 02:55:39 PM
-- Design Name:
-- Module Name: Counter - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
```

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Counter is
   Port ( Dir : in STD_LOGIC;
          Res : in STD_LOGIC;
          Clk : in STD_LOGIC;
          Q0 : out STD_LOGIC;
          Q1 : out STD_LOGIC;
          Q2 : out STD_LOGIC);
end Counter;

architecture Behavioral of Counter is
   component D_FF
   Port ( D : in STD_LOGIC;
          Res : in STD_LOGIC;
          Clk : in STD_LOGIC;
          Q : out STD_LOGIC;
          Qbar : out STD_LOGIC
   );
   end component;
   component Slow_clk
```

```
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
    end component;

    signal D0,D1,D2 : STD_LOGIC;  --Internal signals
    signal Clk_slow : STD_LOGIC;  --Internal clock
    signal Q_0,Q_1,Q_2 : STD_LOGIC;  --Internal signals
begin

    Slow_Clk0: Slow_Clk
        PORT MAP(
            Clk_in => Clk,
            Clk_out => Clk_slow
        );
    D_FF0: D_FF
        PORT MAP(
            D => D0,
            Res => Res,
            Clk => Clk_slow,
            Q => Q_0
        );

    D_FF1: D_FF
         PORT MAP(
            D => D1,
            Res => Res,
            Clk => Clk_slow,
            Q => Q_1
         );
```

```
  D_FF2: D_FF
      PORT MAP(
          D => D2,
          Res => Res,
          Clk => Clk_slow,
          Q => Q_2
      );
  D0 <= (NOT(Dir) AND NOT(Q_2)) OR (Dir AND Q_1); --OR (NOT(Q_2) AND Q_1)   --DC
  D1 <= (Dir AND Q_2) OR (NOT(Dir) AND Q_0) OR (Q_2 AND Q_0);    --DB
  D2 <= (NOT(Dir) AND Q_1) OR (Q_1 AND NOT(Q_0)) OR (Dir AND NOT(Q_0));   --DA
  Q0 <= Q_0; --QC
  Q1 <= Q_1; --QB
  Q2 <= Q_2; --QA
end Behavioral;
```

- Test Bench Code – Counter (Counter_sim.vhd)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/09/2022 03:24:17 PM
-- Design Name:
-- Module Name: Counter_sim - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
```

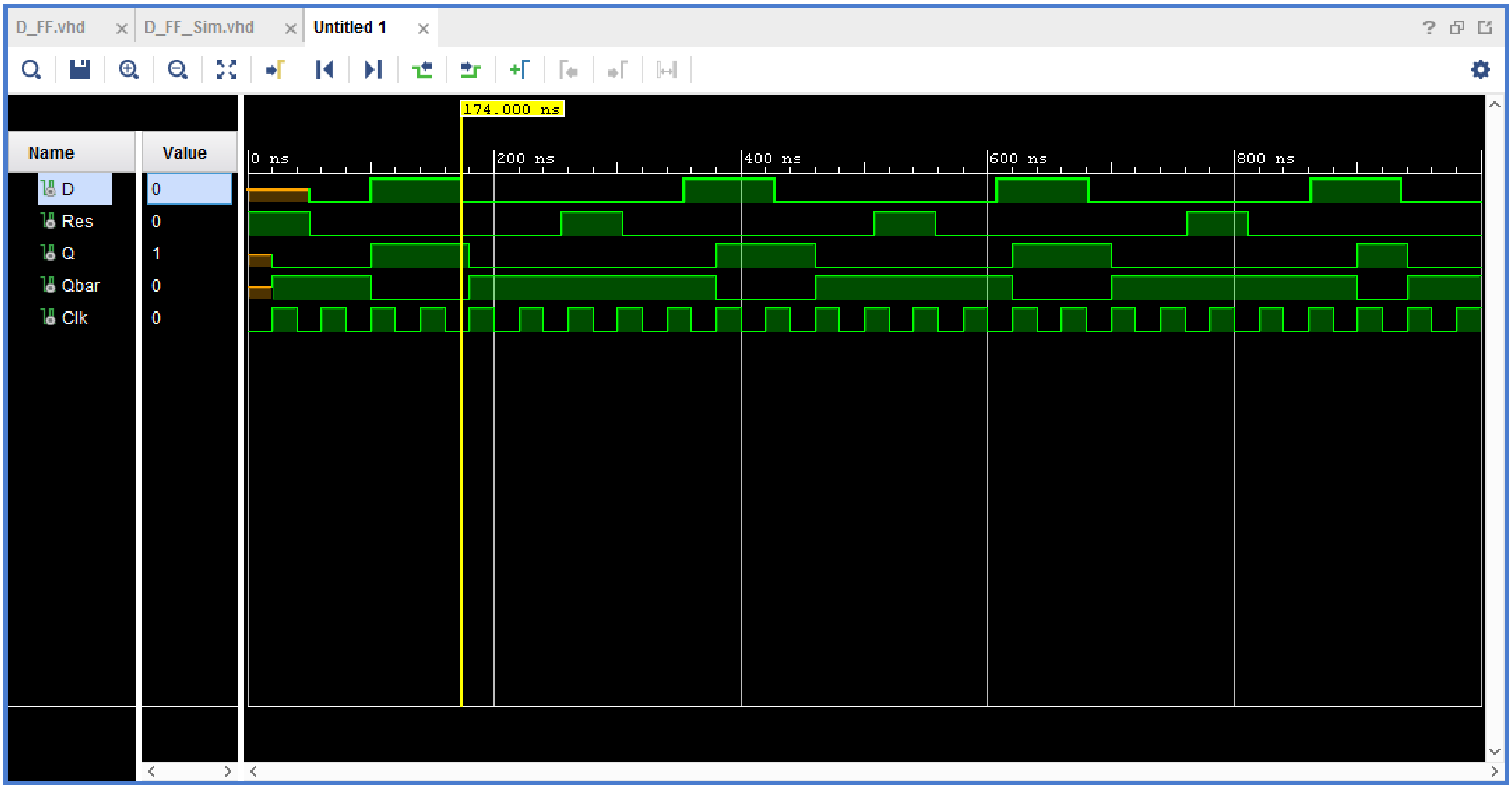
```
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Counter_sim is
--  Port ( );
end Counter_sim;

architecture Behavioral of Counter_sim is
component Counter
    Port ( Dir : in STD_LOGIC;
           Res : in STD_LOGIC;
           Clk : in STD_LOGIC;
           Q0 : out STD_LOGIC;
           Q1 : out STD_LOGIC;
           Q2 : out STD_LOGIC);
```

```
end component;
signal Clk : STD_LOGIC := '0';
signal Dir,Res,Q0,Q1,Q2 : STD_LOGIC;

begin
  UUT : Counter
    PORT MAP(
      Clk => Clk,
      Dir => Dir,
      Res => Res,
      Q0 => Q0,
      Q1 => Q1,
      Q2 => Q2
    );
  process
  begin
    wait for 5ns;   --100MHz
    Clk <= NOT(Clk);
  end process;

  process
  begin
    Res <= '1';
    wait for 100ns;

    Res <= '0';
    Dir <= '0';
    wait for 300ns;
```

```
        Dir <= '1';

        wait for 300ns;


    end process;

end Behavioral;
```

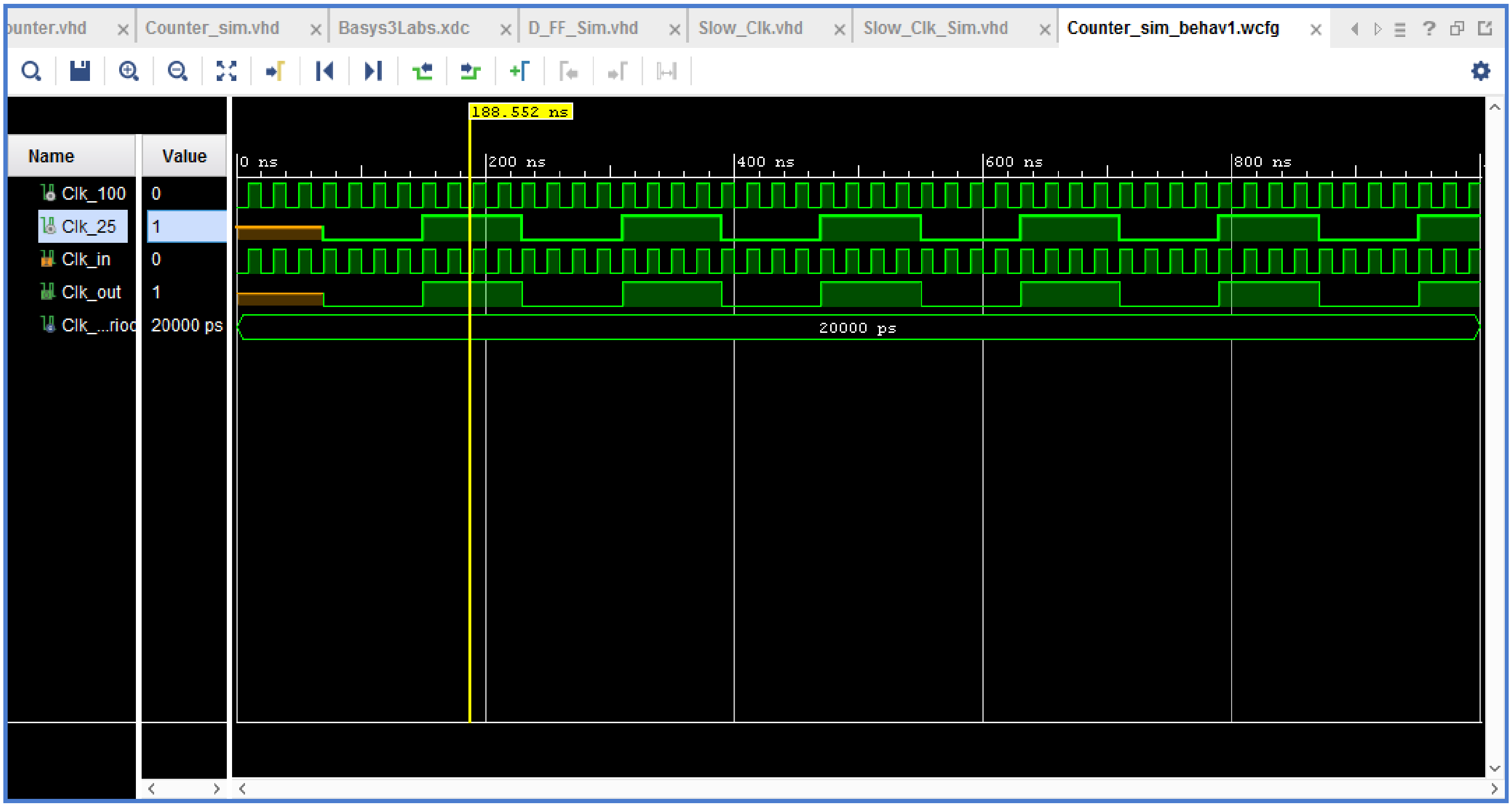
- ## Timing Diagrams

  - ### D Flip-Flop



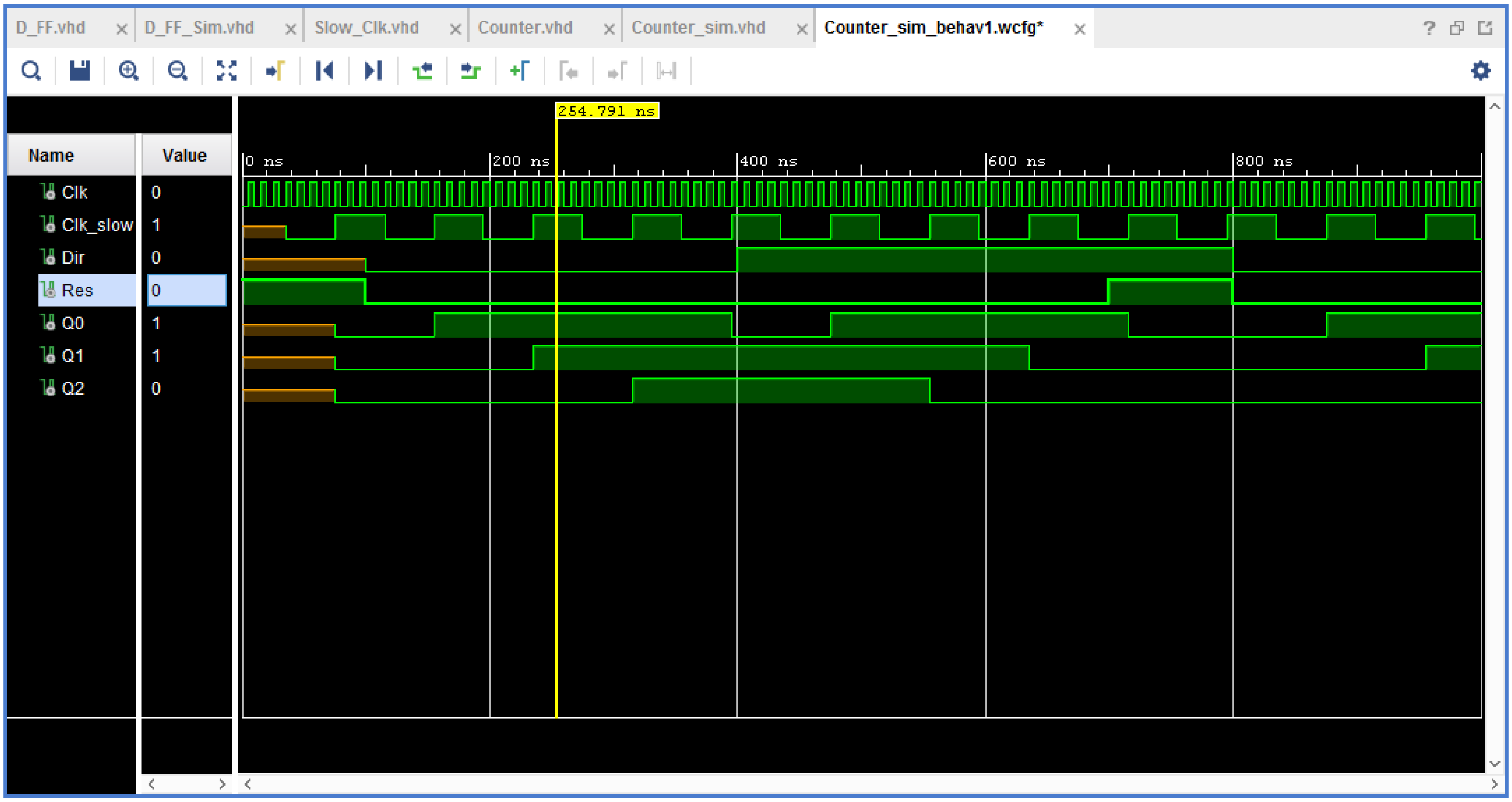
| Name | Value |
|---|---|
| D | 0 |
| Res | 0 |
| Q | 1 |
| Qbar | 0 |
| Clk | 0 |

- Slow down clock



| Name | Value |
|---|---|
| Clk_100 | 0 |
| Clk_25 | 1 |
| Clk_in | 0 |
| Clk_out | 1 |
| Clk_...riod | 20000 ps |

- Counter



| Name | Value |
|---|---|
| Clk | 0 |
| Clk_slow | 1 |
| Dir | 0 |
| Res | 0 |
| Q0 | 1 |
| Q1 | 1 |
| Q2 | 0 |

## Conclusions from the lab

- ✓ Counter is a register that goes through a predetermined sequence of states.
- ✓ Counter can be designed using D Flip-Flops and logic gates.
- ✓ We can control the direction of counting (clockwise or anticlockwise) by using an external input.
- ✓ Output changes only as a rising edge of a clock signal.
- ✓ All D Flip-Flops using the same clock.