**Lab 5 - Counter**

CS1050 Computer Organization and Digital Design

Name : **Dissanayake D.M.A.K.**

Index No. : **220135N**

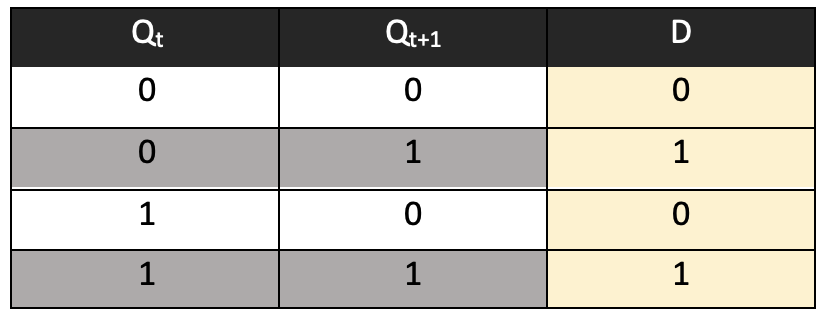
Group : **44**

# Lab Task:

Designing a 3 bit up / down counter using D flip-flops and slow down clock. First we completed the truth table and got expressions for D0, D1, D2. Then we created D flip flop and slow down clock. By using these D flip flops and slow down clock we created the counter.

**Introduction**

We initiate the lab by examining D flip-flop excitation tables and streamlining equations using Karnaugh Maps. Following this, we progress to VHDL modeling of D flip-flops. Additionally, we establish a slowing counter model to effectively regulate clock frequencies, ensuring optimal visibility.

The central objectives of the lab involve constructing the 3-bit counter circuit, incorporating the derived Boolean expressions, and integrating the slowing counter to manage clock pulses efficiently. Rigorous simulation is employed to validate the counter's performance and functionality before advancing to hardware verification using the BASYS 3 board.

## Excitation Table of D Flip-Flop

## State Table of Counter

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Qt | | | Button | Qt+1 | | | D2 | D1 | D0 |
| Q2 | Q1 | Q0 | Q2 | Q1 | Q0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | X | X | X | X | X | X |
| 0 | 1 | 0 | 1 | X | X | X | X | X | X |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | X | X | X | X | X | X |
| 1 | 0 | 1 | 1 | X | X | X | X | X | X |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |

## Karnaugh Maps for three Inputs

We can derive the boolean expressions for the inputs D2, D1 and D0 based Qt and B.

**For D0**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q2Q1  Q0B | 00 | 01 | 11 | 10 |
| 00 | 1 | X | 0 | 0 |
| 01 | 0 | X | 1 | 0 |
| 11 | 0 | 1 | 1 | X |
| 10 | 1 | 1 | 0 | X |

D0 = Q1.B + Q’2.B’

**For D1**

D1 = Q2.B + Q0.B’

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q2Q1  Q0B | 00 | 01 | 11 | 10 |
| 00 | 0 | X | 0 | 0 |
| 01 | 0 | X | 1 | 1 |
| 11 | 0 | 0 | 1 | X |
| 10 | 1 | 1 | 1 | X |

**For D2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q2Q1  Q0B | 00 | 01 | 11 | 10 |
| 00 | 0 | X | 1 | 0 |
| 01 | 1 | X | 1 | 1 |
| 11 | 0 | 0 | 0 | X |
| 10 | 0 | 1 | 1 | X |

D2 = Q’0.B + Q1.B’

# 1)D Flip Flop

## Design Source File

**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;

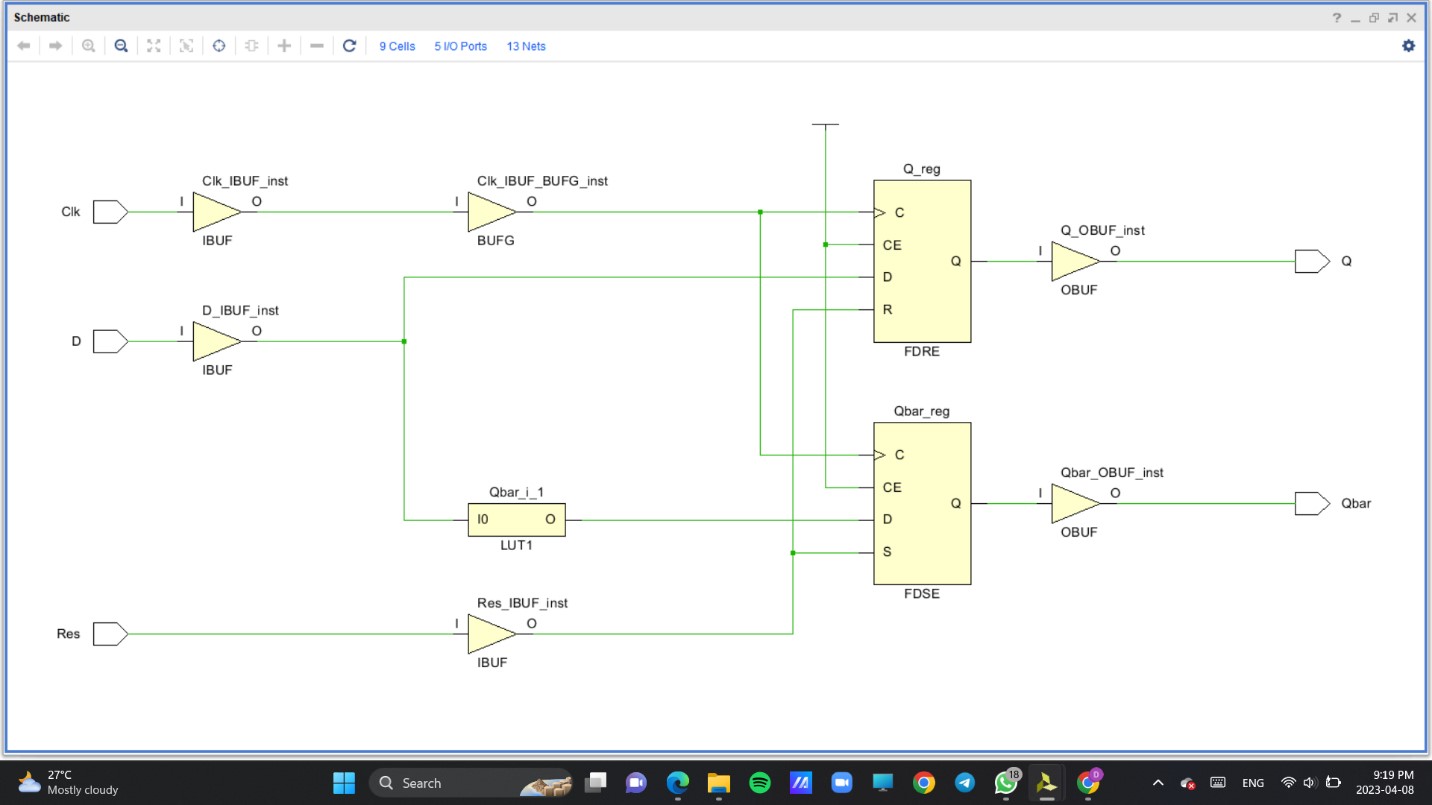
**end** if;

**end** if;

**end** process;

**end** Behavioral**;**

## Elaborated design schematic.



**Simulation source file**  
**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.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 **:** STD\_LOGIC**;**

**Signal** Res **:** STD\_LOGIC**;**

**Signal** Clk **:** STD\_LOGIC**;**

**Signal** Q **:** STD\_LOGIC**;**

**Signal** Qbar **:** STD\_LOGIC**;**

**begin**

UUT**:** D\_FF **port** **map(**

D **=>** D**,**

Res **=>** Res**,**

Clk **=>** Clk**,**

Q **=>** Q**,**

Qbar **=>** Qbar**);**

Clk\_process**:**

**process**

**begin**

Clk **<=** '0'**;**

**wait** **for** 50ns**;**

Clk **<=** '1'**;**

**wait** **for** 50ns**;**

**end** **process;**

**process**

**begin**

D **<=** '0'**;** Res **<=** '1'**;** **wait** **for** 100 ns**;**

Res **<=** '0'**;** D **<=** '0'**;** **wait** **for** 100 ns**;**

Res **<=** '0'**;** D **<=** '1'**;** **wait** **for** 100 ns**;**

Res **<=** '1'**;** D **<=** '1'**;** **wait** **for** 100 ns**;**

Res **<=** '0'**;** D **<=** '1'**;** **wait** **for** 100 ns**;**

Res **<=** '1'**;** D **<=** '0'**;** **wait** **for** 100 ns**;**

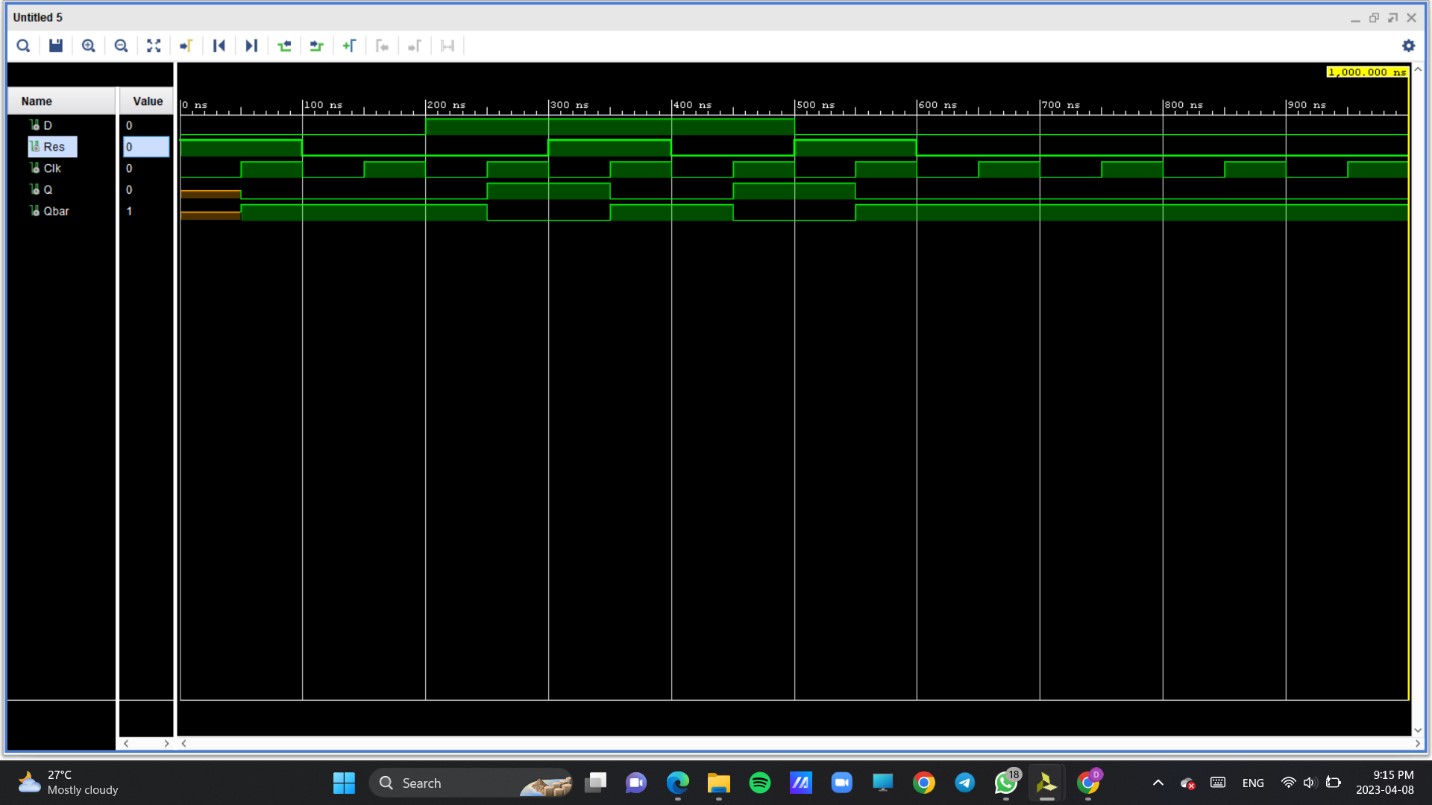
Res **<=** '0'**;** D **<=** '0'**;** **wait** **for** 100 ns**;**

**wait;**

**end** **process;**

**end** behavioral**;**

**Timing Diagram**



# 2) Slow Clock

## Design Source File

**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**

**process** **(**Clk\_in**)** **begin**

**if** **(rising\_edge(**Clk\_in**))** **then**

count **<=** count **+** 1**;**

**if(**count **=** 5**)** **then**

clk\_status **<=** **not** clk\_status**;**

Clk\_out **<=** clk\_status**;**

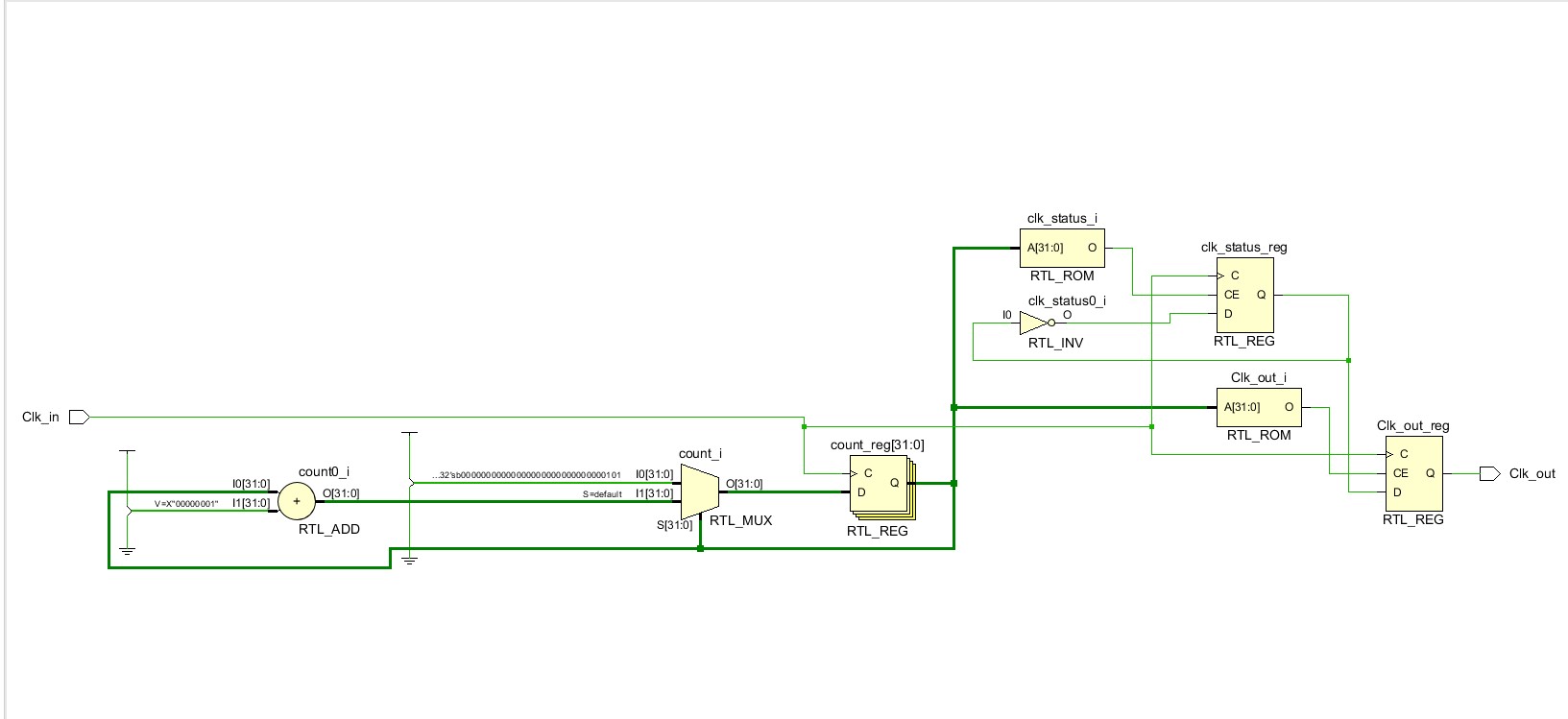
count **<=** 1**;**

**end** **if;**

**end** **if;**

**end** **process;**

**end** Behavioral**;**

**Elaborated design schematic**

**Simulation File**

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**entity** Slow\_Clk\_Sim **is**

-- Port ( );

**end** Slow\_Clk\_Sim**;**

**architecture** Behavioral **of** Slow\_Clk\_Sim **is**

**signal** Clk\_in**:** STD\_LOGIC **:=** '0'**;**

**signal** Clk\_out**:** STD\_LOGIC**;**

**begin**

-- Instantiate the Unit Under Test (UUT)

UUT**:**Slow\_Clk **port** **map(**

Clk\_in **=>** Clk\_in**,**

Clk\_out **=>** Clk\_out

**);**

-- Clock process definitions

Clk\_process**:**

**Process**

**begin**

Clk\_in **<=** '0'**;**

**wait** **for** 10ns**;**

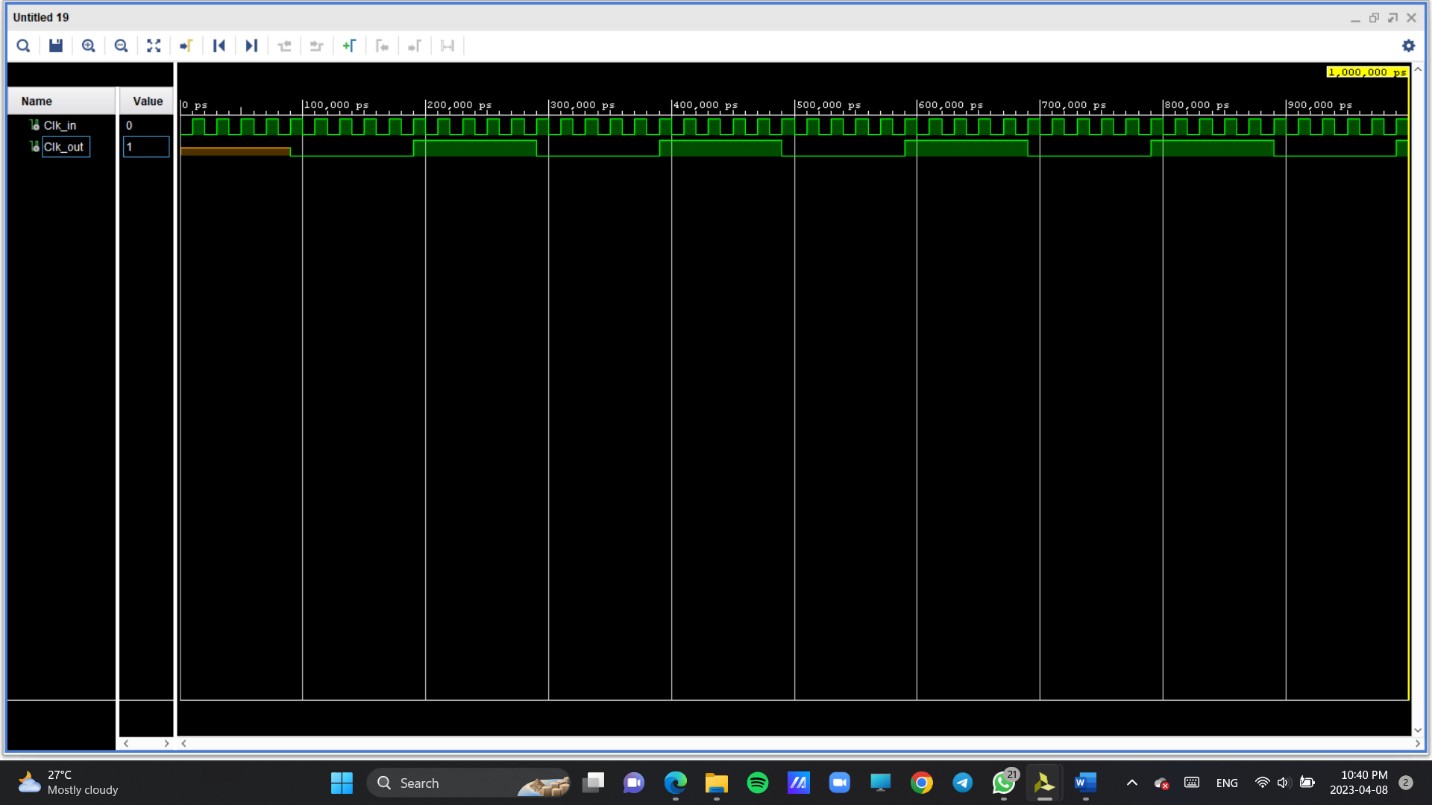
Clk\_in **<=** '1'**;**

**wait** **for** 10ns**;**

**end** **process;**

**end** Behavioral**;**

**Timing Diagram**



# 3) Counter

## Design Source File

**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**;**

Q **:** **out** STD\_LOGIC\_VECTOR **(**2 **downto** 0**));**

**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** Q0**,** Q1**,** Q2 **:** std\_logic**;** -- Internal signals

**signal** Clk\_Slow **:** std\_logic**;** -- Internal clock

**begin**

Slow\_Clk0 **:** Slow\_Clk

**port** **map** **(**

Clk\_in **=>** Clk**,**

Clk\_out **=>** Clk\_slow**);**

D0 **<=** **((not** Q2**)** **and** **(not** Dir**))** **or** **(**Q1 **and** Dir**);**

D1 **<=** **((not** Dir**)** **and** Q0**)** **or** **(**Dir **and** Q2**);**

D2 **<=** **((not** Dir**)** **and** Q1**)** **or** **(**Dir **and** **(not** Q0**));**

D\_FF0 **:** D\_FF

**port** **map** **(**

D **=>** D0**,**

Res **=>** Res**,**

Clk **=>** Clk\_slow**,**

Q **=>** Q0**);**

D\_FF1 **:** D\_FF

**port** **map** **(**

D **=>** D1**,**

Res **=>** Res**,**

Clk **=>** Clk\_slow**,**

Q **=>** Q1**);**

D\_FF2 **:** D\_FF

**port** **map** **(**

D **=>** D2**,**

Res **=>** Res**,**

Clk **=>** Clk\_slow**,**

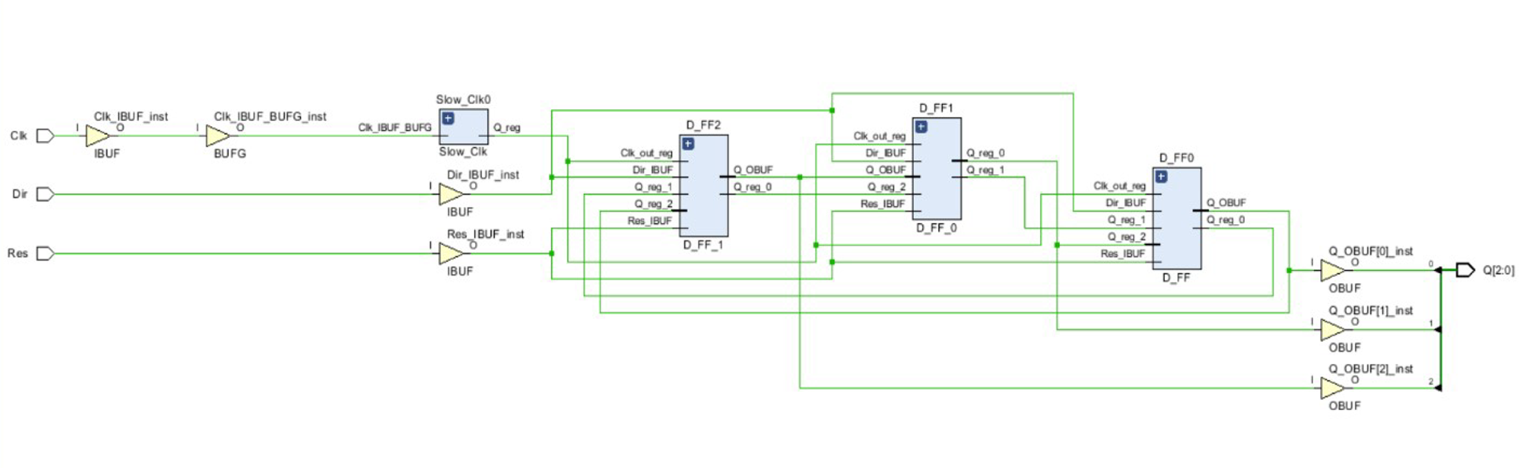
Q **=>** Q2**);**

Q**(**0**)** **<=** Q0**;**

Q**(**1**)** **<=** Q1**;**

Q**(**2**)** **<=** Q2**;**

**end** Behavioral**;**

**Elaborated design schematic**

**Simulation File**

**library** IEEE**;**

**use** IEEE**.**STD\_LOGIC\_1164**.ALL;**

**entity** Counter\_sim **is**

-- Port ( );

**end** Counter\_sim**;**

**architecture** Behavioral **of** Counter\_sim **is**

-- Component declarations

**component** Counter

**Port** **(** Dir **:** **in** STD\_LOGIC**;**

Res **:** **in** STD\_LOGIC**;**

Clk **:** **in** STD\_LOGIC**;**

Q **:** **out** STD\_LOGIC\_VECTOR **(**2 **downto** 0**));**

**end** **component;**

-- Signal declarations

**signal** Dir **:** **in** STD\_LOGIC**;**

**signal** Res **:** **in** STD\_LOGIC**;**

**signal** Clk **:** **in** STD\_LOGIC**;**

**signal** Q **:** **out** STD\_LOGIC\_VECTOR **(**2 **downto** 0**));**

**begin**

UUT**:** Counter

**Port** **map** **(**Dir **=>** Dir**,**

Res **=>** Res**,**

Clk **=>** Clk**,**

Q **=>** Q**);**

Clk\_process**:**

**process**

**begin**

Clk **<=** '0'**;** **wait** **for** 10ns**;**

Clk **<=** '1'**;** **wait** **for** 10ns**;**

**end** **process;**

**process**

**begin**

Res **<=** '1'**;** Dir **<=** '0'**;** **wait** **for** 100ns**;** Dir **<=** '0'**;** Res **<=** '0'**;** **wait** **for** 100ns**;**

Res **<=** '0'**;** **wait** **for** 100ns**;**

Res **<=** '1'**;** Dir **<=** '1'**;** **wait** **for** 100ns**;**

Dir **<=** '1'**;** Res **<=** '0'**;** **wait** **for** 100ns**;**

Res **<=** '1'**;** **wait** **for** 100ns**;**

Res **<=** '0'**;** Dir **<=** '1'**;** **wait** **for** 100ns**;**

Dir **<=** '0'**;** Res **<=** '0'**;** **wait** **for** 100ns**;**

Res **<=** '1'**;** **wait** **for** 100ns**;**

Res **<=** '0'**;** Dir **<=** '1'**;** **wait** **for** 100ns**;**

Dir **<=** '1'**;** Res **<=** '1'**;** **wait** **for** 100ns**;**

Res **<=** '0'**;** **wait** **for** 100ns**;**

Res **<=** '0'**;** Dir **<=** '0'**;** **wait** **for** 100ns**;**

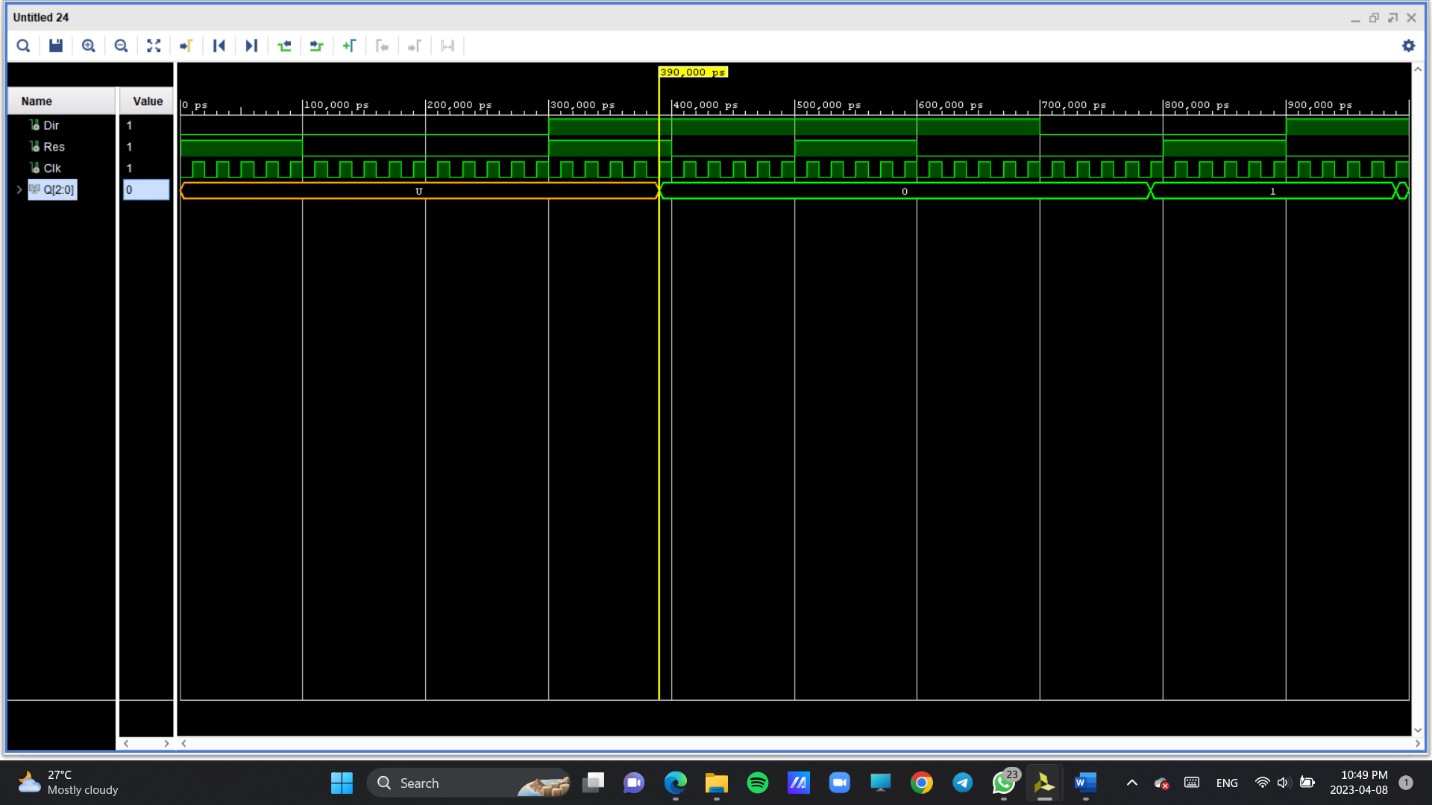
Dir **<=** '0'**;**

**wait;**

**end** **process;**

**end** Behavioral**;**

**Timing Diagram**



**Constraints File**

## Clock signal

set\_property PACKAGE\_PIN W5 [get\_ports Clk]

set\_property IOSTANDARD LVCMOS33 [get\_ports Clk]

create\_clock -add -name sys\_clk\_pin -period 10.00 -waveform

{0 5} [get\_ports Clk]

## Switches

set\_property PACKAGE\_PIN V17 [get\_ports {Dir}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Dir}]

## LEDs

set\_property PACKAGE\_PIN U16 [get\_ports {Q(0)}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Q(0)}]

set\_property PACKAGE\_PIN E19 [get\_ports {Q(1)}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Q(1)}]

set\_property PACKAGE\_PIN U19 [get\_ports {Q(2)}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Q(2)}]

##Buttons

set\_property PACKAGE\_PIN U17 [get\_ports Res]

set\_property IOSTANDARD LVCMOS33 [get\_ports Res]

**Conclusion**

A counter is like a digital gadget that can either increase or decrease its number with each tick of a clock. You can decide if it should count up or down using an external switch. It basically goes through different stages, and each time the clock ticks, the number it shows changes. Depending on what kind of counter it is, you can make it change by giving it a clock tick or using other buttons. People use counters in lots of electronic things to measure time or how often something happens.

-End-