# KATHMANDU UNIVERSITY

SCHOOL OF ENGINEERING

**DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING**

# LAB REPORT 06


## EEEG-321

Department of Electrical and Electronics Engineering

**By:**

Samyam Shrestha (32056)

**To:**

Santosh Shaha Sir

**Date:**

**4th June, 2024**

# Title: Write VHDL D and T flipflop

**Requirements: Vivado Software, Laptop, FPGA board**

1) **D flipflop**

**VHDL Code**

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity d_flipflop_56 is
port (
din : in std_logic;
clk: in std_logic;
rst: in std_logic;
q: out std_logic);
end d_flipflop_56;

architecture Behavioral of d_flipflop_56 is
begin
dff: process (clk, rst, din) begin
if (rst = '1') then
q <= '0';
elsif (clk' event and clk = '1') then
   q <= din;
end if;
end process dff;
end Behavioral;
```

**Test Bench code:**

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity dflipflop_tb_56 is
end dflipflop_tb_56;

architecture Behavioral of dflipflop_tb_56 is
component d_flipflop_56
port (
din : in std_logic;
clk: in std_logic;
rst: in std_logic;
q: out std_logic);
end component;

signal din: std_logic;
signal clk: std_logic;
signal rst: std_logic;
```

```
signal q: std_logic;

begin
uut: d_flipflop_56  port map ( din => din,
clk => clk,
rst => rst,
q => q);
stimulus: process begin
clk <= '0';
wait for 10ns;
clk <= '1';
wait for 10ns;
end process;
stimuli: process begin
din <= '0';
wait for 10ns;
din <= '1';
wait for 10ns;
end process;
stimuls: process begin
rst <= '0';
wait for 100ns;
rst <= '1';
wait for 10ns;
end process;
end Behavioral;
```

**RTL Schematic Diagram:**

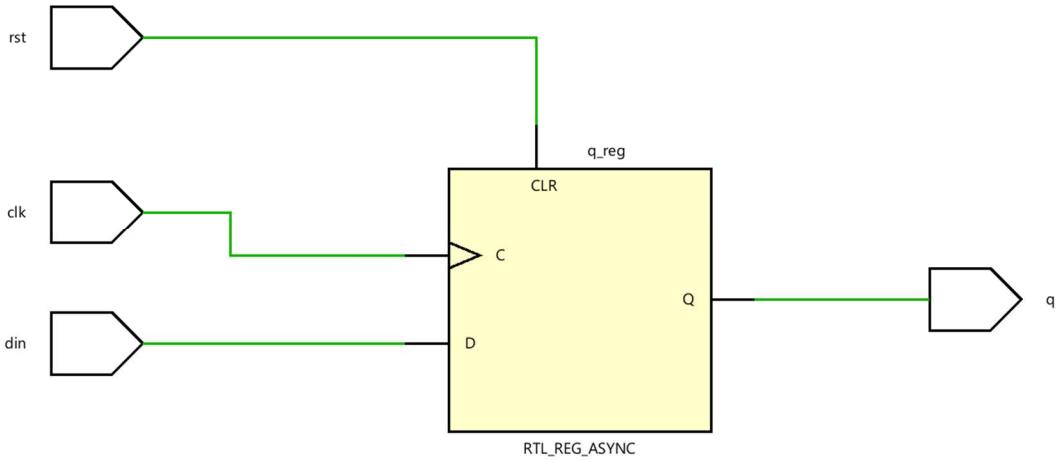


*Figure 1: RTL Schematic of D flip-flop*

**Simulation Model:**

The simulation results are depicted in the figure below, showcasing the D flip-flop output values.

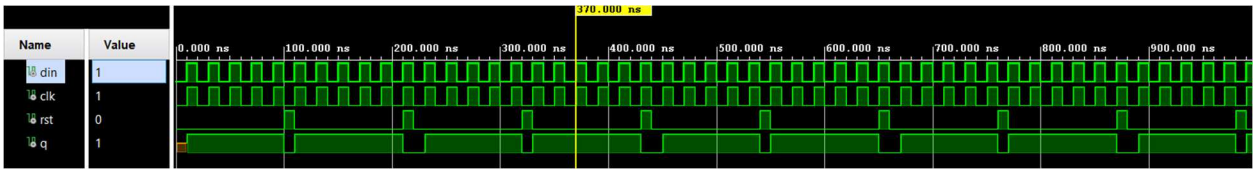


*Figure 2: Simulation Model of D flip-flop*

**XDC File**

*set_property IOSTANDARD LVCMOS33 [get_ports clk]*
*set_property IOSTANDARD LVCMOS33 [get_ports din]*
*set_property IOSTANDARD LVCMOS33 [get_ports q]*
*set_property IOSTANDARD LVCMOS33 [get_ports rst]*

**Device Utilization Report:**

| Resource | Utilization | Available | Utilization % |
|---|---|---|---|
| FF | 1 | 65200 | 0.00 |
| IO | 4 | 210 | 1.90 |

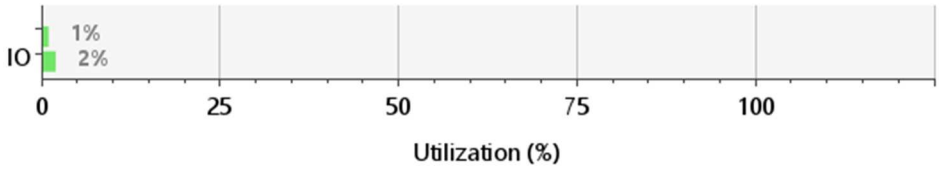


*Figure 3: Device Utilization Report*

**Timing Analysis Report:**

**Design Timing Summary**

| Setup | | Hold | | Pulse Width | |
|---|---|---|---|---|---|
| Worst Negative Slack (WNS): | inf | Worst Hold Slack (WHS): | inf | Worst Pulse Width Slack (WPWS): | NA |
| Total Negative Slack (TNS): | 0.000 ns | Total Hold Slack (THS): | 0.000 ns | Total Pulse Width Negative Slack (TPWS): | NA |
| Number of Failing Endpoints: | 0 | Number of Failing Endpoints: | 0 | Number of Failing Endpoints: | NA |
| Total Number of Endpoints: | 3 | Total Number of Endpoints: | 3 | Total Number of Endpoints: | NA |

**There are no user specified timing constraints.**

*Figure 4: Timing Summary*

## 2) T flip-flop

### VHDL Code

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity t_56 is
port (
tin: in std_logic;
clk: in std_logic;
rst: in std_logic;
q: out std_logic);
end t_56;

architecture Behavioral of t_56 is
begin
tff: process (clk, rst, tin) is
variable m: std_logic := '0';
begin
   if (rst ='1') then
   m := '0';
   elsif (rising_edge(clk)) then
   if (tin = '1') then
   m := not m;
   end if;
   end if;
   q <= m;
   end process tff;
end Behavioral;
```

### Test Bench Code

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity t_tb is
end t_tb;

architecture Behavioral of t_tb is
component t_56
port (
tin: in std_logic;
clk: in std_logic;
rst: in std_logic;
q: out std_logic);
end component;
```

```
signal tin: std_logic;
signal clk: std_logic;
signal rst: std_logic;
signal q: std_logic;
signal m: std_logic;
begin
uut : t_56 port map ( tin => tin,
clk => clk,
rst => rst,
q => q);

stimulus: process begin
clk <= '0';
wait for 10ns;
clk <= '1';
wait for 10ns;
end process;

stimuli: process begin
tin <= '0';
wait for 10ns;
tin <= '1';
wait for 100ns;
end process;
stim: process begin
rst <= '0';
wait for 100ns;
rst <= '1';
wait for 10ns;
end process;
stt: process begin
m <= '0';
wait for 10ns;
m <= '1';
wait for 100ns;
end process;
end Behavioral;
```

## RTL Schematic

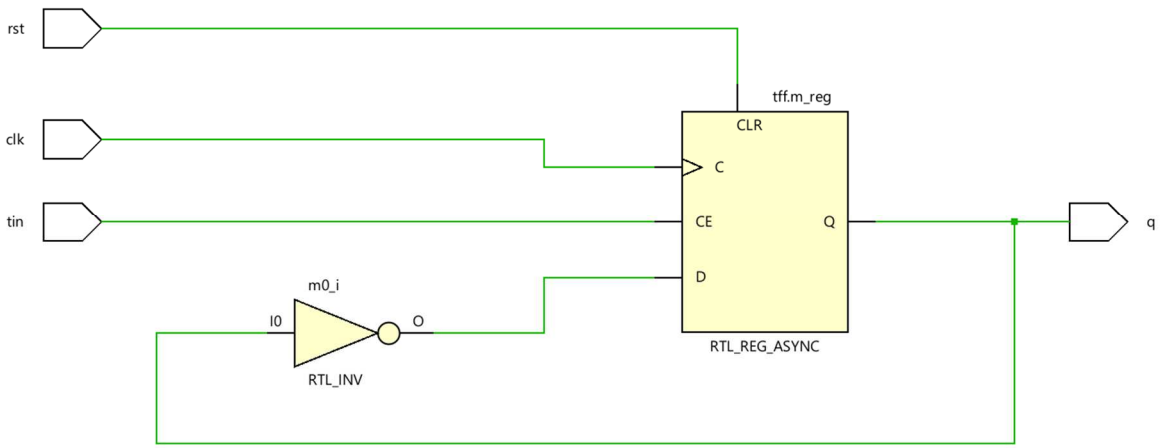


*Figure 5: RTL Schematic of T flip-flop*

## XDC File

*set_property IOSTANDARD LVCMOS33 [get_ports clk]*
*set_property IOSTANDARD LVCMOS33 [get_ports q]*
*set_property IOSTANDARD LVCMOS33 [get_ports rst]*
*set_property IOSTANDARD LVCMOS33 [get_ports tin]*

## Simulation

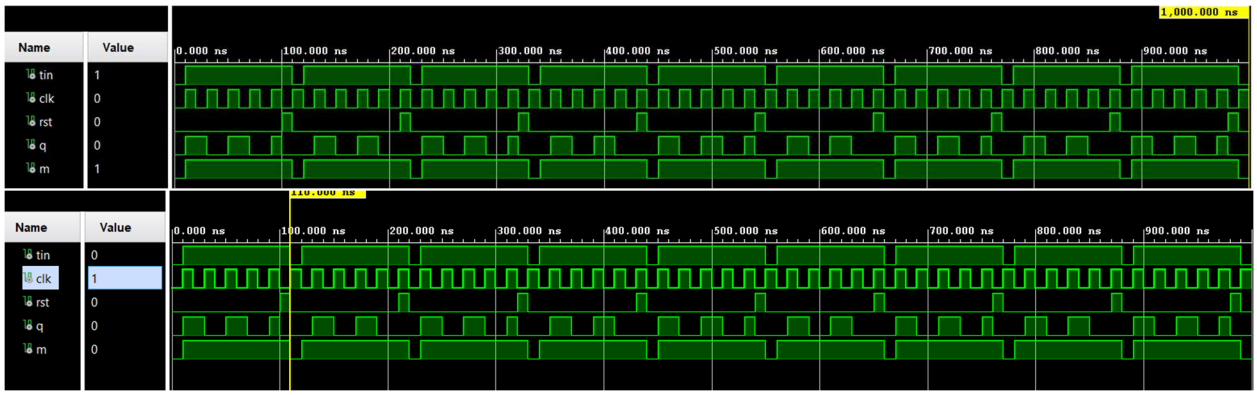


*Figure 6: Timing Diagram for T flip-flop*

## Device Utilization Report

**Summary**

| Resource | Utilization | Available | Utilization % |
|---|---|---|---|
| LUT | 1 | 32600 | 0.00 |
| FF | 1 | 65200 | 0.00 |
| IO | 4 | 210 | 1.90 |

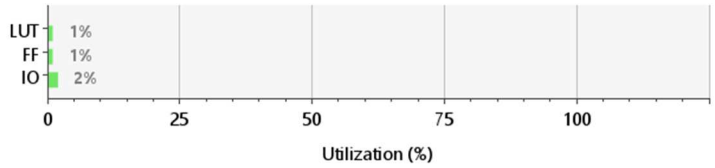


*Figure 7: Device Utilization Report*

## Time Analysis Report

**Design Timing Summary**

| Setup | | Hold | | Pulse Width | |
|---|---|---|---|---|---|
| Worst Negative Slack (WNS): | inf | Worst Hold Slack (WHS): | inf | Worst Pulse Width Slack (WPWS): | NA |
| Total Negative Slack (TNS): | 0.000 ns | Total Hold Slack (THS): | 0.000 ns | Total Pulse Width Negative Slack (TPWS): | NA |
| Number of Failing Endpoints: | 0 | Number of Failing Endpoints: | 0 | Number of Failing Endpoints: | NA |
| Total Number of Endpoints: | 3 | Total Number of Endpoints: | 3 | Total Number of Endpoints: | NA |

**There are no user specified timing constraints.**

*Figure 8: Timing Summary for Full Adder*

## Conclusion

In summary, we developed and tested VHDL code for a D and T flip-flop using Vivado. Additionally, troubleshooting steps were provided to resolve potential file access errors during simulation, such as ensuring no other processes are using the log file, manually deleting the file, and cleaning and rebuilding the project in Vivado. This comprehensive approach ensures that the counter operates correctly and that any issues encountered during simulation are addressed effectively.