UNIVERSITY OF GHANA

UNIVERSITY OF GHANA

Name: Evans Acheampong

Student ID: 10987644

Date of Submission: 20th July, 2023

Lab Index: LAB 5

Design and Simulation of a D Flip Flop in VHDL

I. AIM OF THE EXPERIMENT

The experiment aims to design and implement a D flip-flop using VHDL. It provides a hands-on learning experience in VHDL syntax, behavioral modeling, and clock signal handling. The specific objectives are understanding VHDL, designing the flip-flop, simulating its behavior, and exploring synthesis possibilities. By the end, students should grasp D flip-flop operation, VHDL for basic digital circuits, and simulation techniques as a foundation for more complex designs and hardware implementation projects.

II. ABSTRACT

The lab report focuses on the design and implementation of a D flip-flop using VHDL. The experiment aims to provide a practical understanding of VHDL syntax and behavioral modeling while exploring clock signal handling. Through simulation, the correctness of the flip-flop's behavior is verified. Additionally, students are introduced to synthesis possibilities for hardware implementation on FPGA or ASIC. The experiment's outcomes include a grasp of D flip-flop operation, VHDL for basic digital circuits, and simulation techniques, laying the groundwork for more complex design projects in the future.

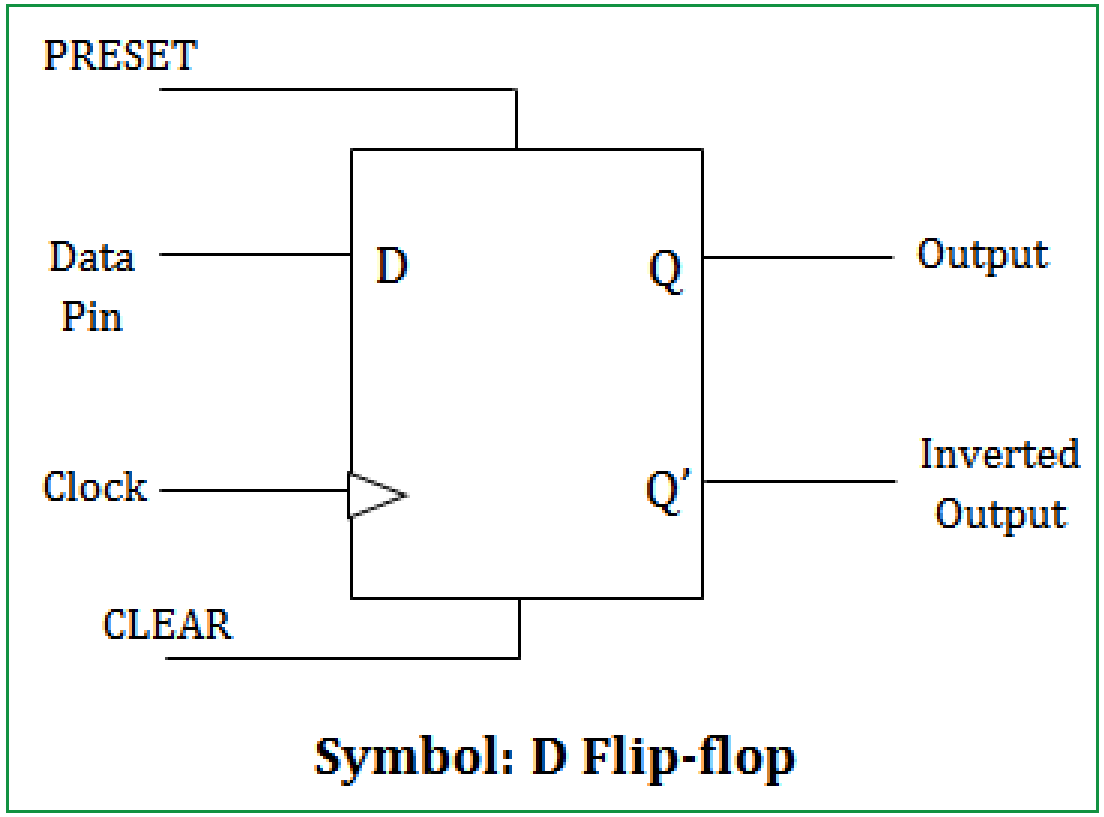
III. INTRODUCTION

This lab report focuses on the design and implementation of a D flip-flop using VHDL (VHSIC Hardware Description Language). VHDL is a powerful language for modeling and simulating digital circuits. The experiment aims to provide hands-on experience in VHDL coding, behavioral modeling, and clock signal handling. Designing the D flip-flop introduces students to fundamental digital circuit principles and sequential logic. Through simulation, students validate their designs, enhancing their proficiency in VHDL and gaining insights into digital circuit behavior. Additionally, the experiment introduces students to synthesis and hardware implementation possibilities, setting the groundwork for more complex digital design projects in the future.

IV. METHODOLOGY

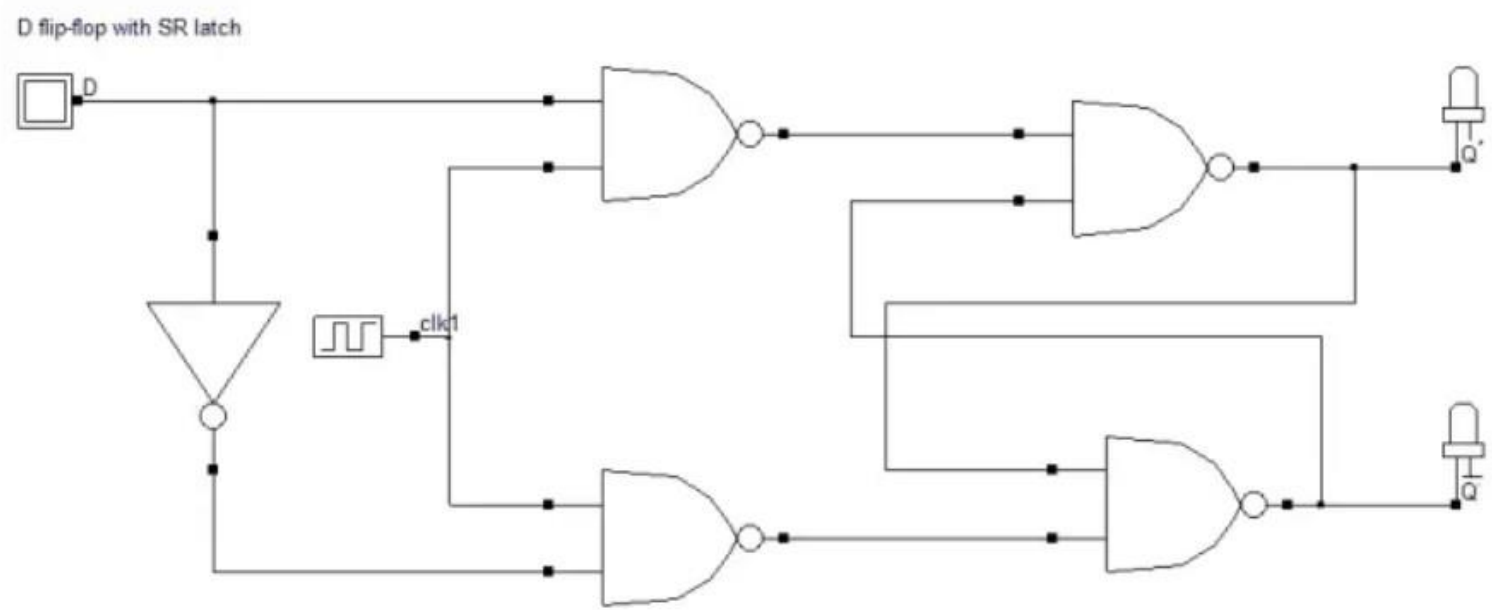
a. Block diagram of entity showing port terminals and datatype:

The block diagram of the entity `D Flip Flop` with its port terminals and datatypes is as follows:



Symbol: D Flip-flop

Circuit Diagram:



D flip-flop using SR

Port Descriptions:

- `clk`: Clock input, type `STD_LOGIC`

- `d`: Data input, type `STD_LOGIC`

- `q`: Output Q, type `STD_LOGIC`

- `not_q`: Output Q', type `STD_LOGIC`

b. Explanation of architecture(s):

The `D Flip Flop` entity is defined in VHDL, and it has a single architecture named `Behavioral`. The architecture describes the behavior of the flip-flop and how it responds to the inputs.

In the `Behavioral` architecture, there is a process sensitive to the `clk` signal. The process contains a conditional statement that checks for the rising edge of the clock (`rising_edge(clk)`). When the rising edge of the clock is detected, the process executes the following code:

d_ff <= d;

Here, `d_ff` is a signal declared inside the architecture, serving as an internal storage element for the flip-flop. It stores the value of the data input `d` when the rising edge of the clock occurs. This represents the behavior of the D flip-flop, where the data input `d` is latched and stored in the flip-flop when the clock rises.

After storing the data, the outputs `q` and `not_q` are updated with the value of `d_ff` and its complement (`not d_ff`), respectively. This behavior ensures that the outputs `q` and `not_q` represent the state of the D flip-flop, following the changes in the data input `d` at each rising edge of the clock.

The `Behavioral` architecture provides an abstract representation of the D flip-flop's behavior, allowing for simulation and verification of the flip-flop's operation before potential hardware implementation.

V. RESULTS AND DISCUSSION

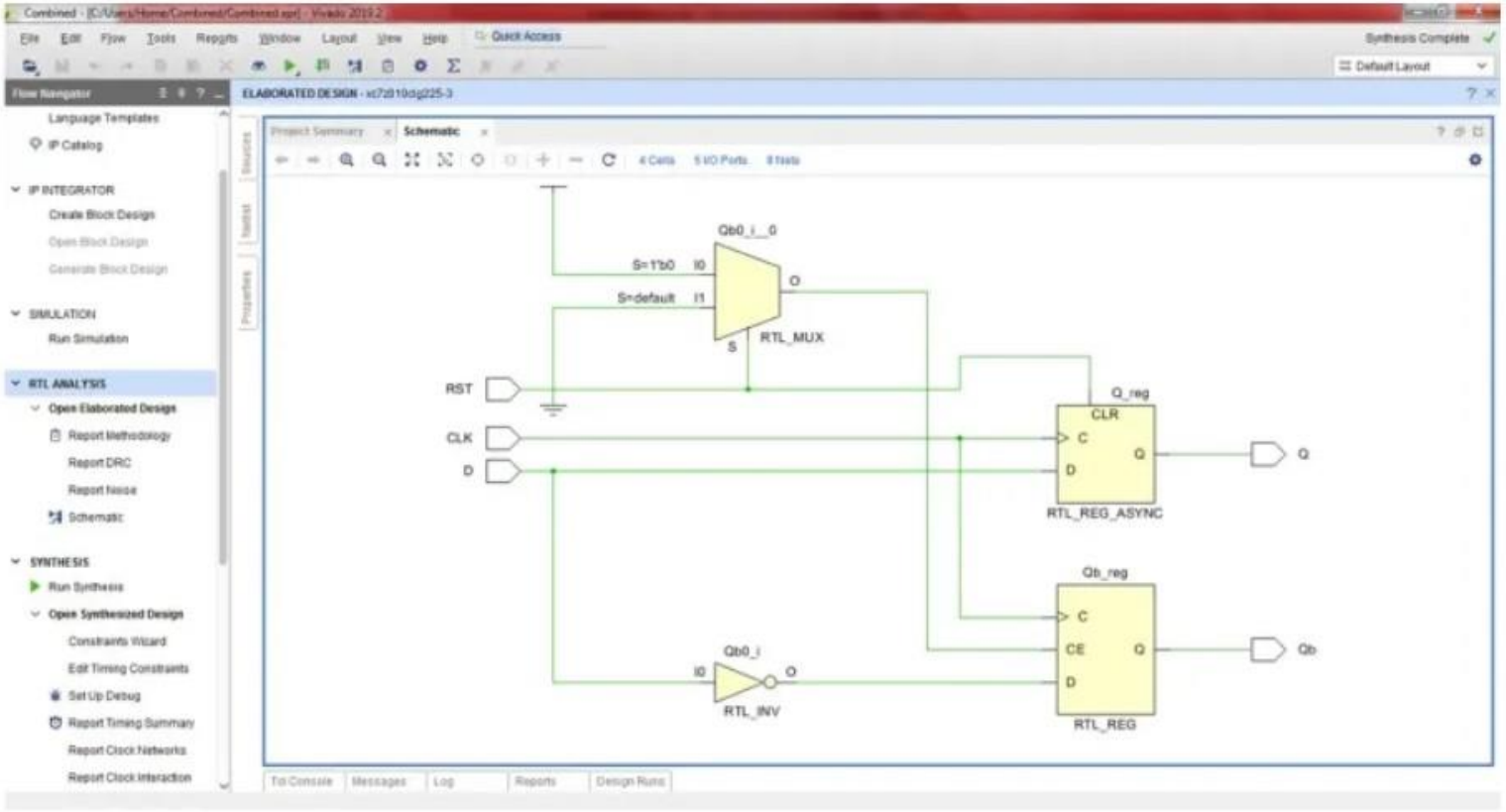
a. Truth Table of Logic Gates:

The truth table for a D flip-flop can be summarized as follows:

Truth table for D flip-flop

CLK	D	Q	Q’
0	x	No change	No change
1	0	0	1
1	1	1	0

RTL Schematic

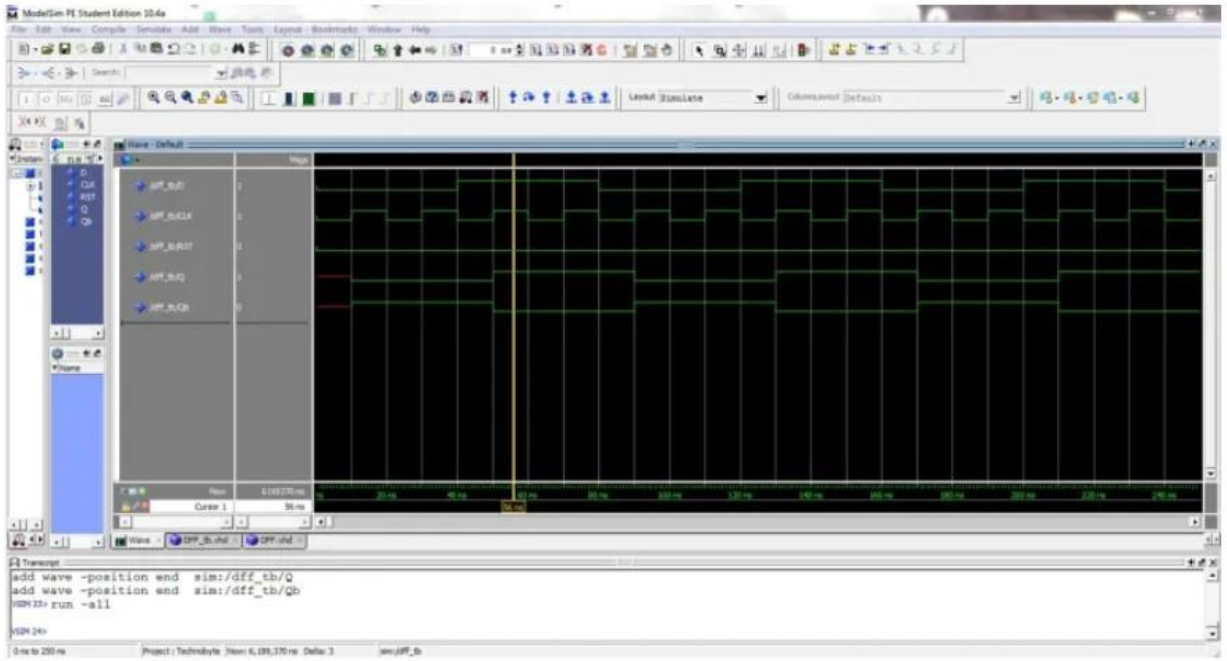


- **Waveform Simulation**

- The waveform simulations of the logic gates were conducted using ModelSim.
- Signal values were forced through the input terminals, and the resulting waveforms at the output terminals were observed.
- The waveforms exhibited the expected behavior of each logic gate, validating the correctness of the designs.

D Flip Flop Waveform Simulation

Simulation Waveform



-- Waveform simulation for D flip-flop

clk:	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
d:	**X**	**0**	**1**	**0**	**1**	**0**	**1**	**1**
q:	**X**	**0**	**0**	**0**	**1**	**1**	**1**	**1**
not_q:	**X**	**1**	**1**	**1**	**0**	**0**	**0**	**0**

- **<u>Comparison of Truth Tables and Simulation Results</u>:**

- The simulation results matched the expected truth tables for each logic gate.
- The outputs of the logic gates corresponded to the specified truth table values for all input combinations.
- The comparison confirmed the accuracy of the VHDL implementations
- When the clock (clk) is low (0), the output Q remains unchanged (q = previous value of q), and the output Q' (not_q) remains the complement of Q.
- When the clock rises from low (0) to high (1), the D flip-flop latches the value of the data input (d) into its internal storage element (d_ff). The output Q then becomes the stored value of d (q = d_ff), and Q' (not_q) becomes the complement of Q.

The simulation results align with the truth table because the VHDL code accurately models the behavior of a D flip-flop. When the clock is low, the outputs don't change, and when the clock rises, the flip-flop captures the data input, reflecting the correct outputs for each input combination.

VI. CONCLUSION

The lab on designing a D flip-flop using VHDL offers valuable insights and practical knowledge for students. Through this experiment, students gain hands-on experience in VHDL coding and behavioral modeling, enabling them to understand and describe digital circuits effectively. The lab focuses on the fundamental principles of digital circuit design, emphasizing sequential logic and clock signal handling, which are essential for more complex circuitry. Moreover, students learn valuable simulation and verification techniques, ensuring the correctness of their designs before potential hardware implementation. The brief introduction to synthesis and hardware implementation provides an awareness of how VHDL designs can be realized on FPGA or ASIC platforms. Overall, this lab lays a strong foundation for students to advance their skills and undertake more sophisticated digital circuit design projects with confidence and proficiency.