UNIVERSITY OF GHANA

Name: Evans Acheampong

Student ID: 10987644

Date of Submission: 22th June, 2023

Lab Index: LAB 3

Design and Simulation of a 4-bit Adder using if-else statements

I. AIM OF THE EXPERIMENT

The aim of the experiment is to implement a Full Adder using VHDL, which involves designing and simulating a digital circuit. This experiment helps participants understand the concept of a Full Adder, gain familiarity with VHDL as a hardware description language, and develop practical skills in circuit design and analysis. The objective is to successfully implement the Full Adder functionality in VHDL, verify its correctness through truth table analysis and waveform simulations, and acquire knowledge and experience in digital circuit design using VHDL.

II. ABSTRACT

This lab report presents the implementation of a Full Adder using VHDL. The Full Adder is an important component in digital circuits as it performs addition on binary numbers, including a carry-in and a carry-out. The VHDL implementation allows for a detailed understanding of the behavior and functionality of the Full Adder. This report provides a comprehensive overview of the lab session, including the methodology, results, and discussions, highlighting the truth table and waveform simulations. The report concludes with the key takeaways from the lab.

III. INTRODUCTION

VHDL (Very High-Speed Integrated Circuit Hardware Description Language) is a hardware description language used to model and design digital systems. It provides a means to describe the behavior and structure of digital circuits, enabling the simulation and synthesis of complex systems. This lab report focuses on implementing a Full Adder using VHDL.

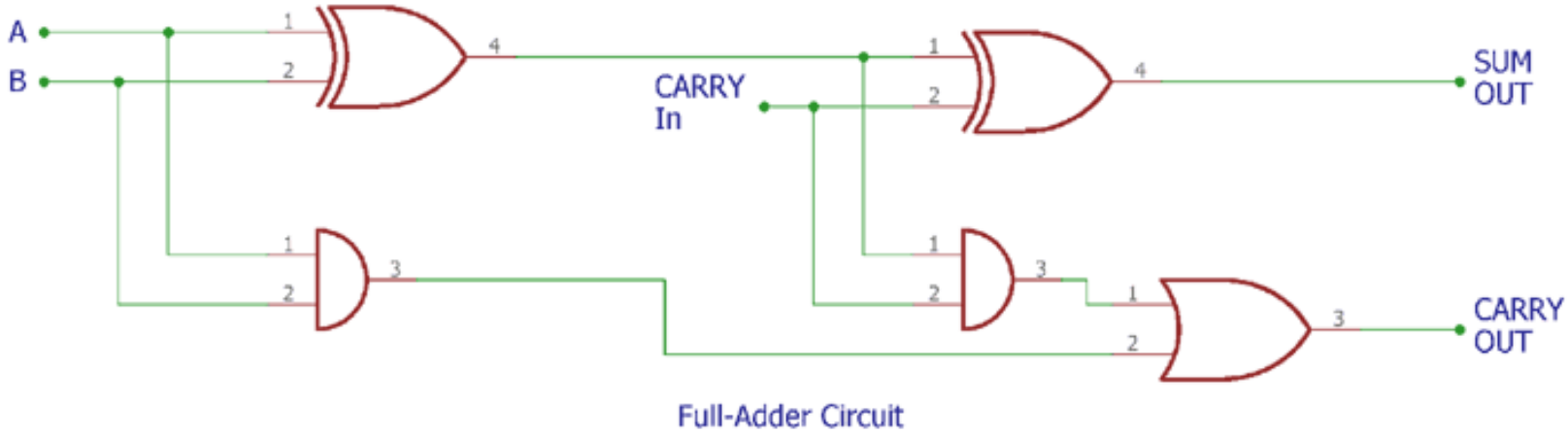
The lab session revolves around understanding the concept of a Full Adder and its importance in digital circuits. The Full Adder performs binary addition on three inputs: A, B, and Cin (carry-in), producing two outputs: Sum and Cout (carry-out). By implementing the Full Adder in VHDL, we gain insights into its functionality and behavior.

IV. METHODOLOGY

The Full Adder is implemented in VHDL using an entity-architecture structure. The entity declaration defines the input and output ports of the Full Adder. In this case, the inputs are A, B, and Cin, while the outputs are Sum and Cout. The datatype used for the ports is STD_LOGIC.

The architecture section of the VHDL code describes the behavior of the Full Adder. It contains a process block that evaluates the input combinations and assigns the appropriate values to the outputs. The process block uses if-else statements to check all possible combinations of input values and determines the corresponding output values.

a. **Block diagram of entity showing port terminals and datatype:** The block diagram of the entity for the simple full adder is as follows:



In the block diagram, the input ports A, B, and Cin are connected to the Full Adder, while the output ports Sum and Cout are generated by the Full Adder.

The datatypes used for the port terminals in this implementation are assumed to be STD_LOGIC in VHDL. The STD_LOGIC type is commonly used to represent digital signals in VHDL, where '0' represents logic low (0) and '1' represents logic high (1).

Therefore, the block diagram shows the input ports A, B, and Cin connected to the Full Adder, and the output ports Sum and Cout generated by the Full Adder, all using the STD_LOGIC datatype.

b. **Explanation of architecture(s):**

The architecture section of the Full Adder VHDL implementation describes the behavior and internal logic of the Full Adder. In this case, the architecture is called "Behavioral."

The Behavioral architecture consists of a process block, which is used to evaluate the input combinations and determine the corresponding output values. The process block is sensitive to changes in the inputs A, B, and Cin.

Inside the process block, if-else statements are used to check all possible combinations of the input values. Each combination corresponds to a specific condition, and the process block determines the appropriate output values based on these conditions.

For example, if the inputs A, B, and Cin are all '0', the if-else statements in the process block will identify this condition and assign the output values Sum and Cout as '0' and '0', respectively. Similarly, for other input combinations, the process block determines the correct output values using if-else statements.

By using if-else statements, the architecture is able to handle all possible input combinations and accurately calculate the sum (Sum) and carry-out (Cout) of the Full Adder.

Overall, the Behavioral architecture provides a behavioral-level description of the Full Adder, specifying its input-output relationship and the conditional logic necessary to perform binary addition.

V. RESULTS AND DISCUSSION

Truth Table of Logic Gates: The truth table for the 4-bit adder can be constructed to illustrate the expected outputs for all possible input combinations. Considering the inputs A, B, and Cin, and the outputs Sum and Cout, the truth table can be represented as follows:

The truth table provides the expected outputs for each combination of inputs A, B, and Cin, demonstrating the logic behavior of the 4-bit adder.

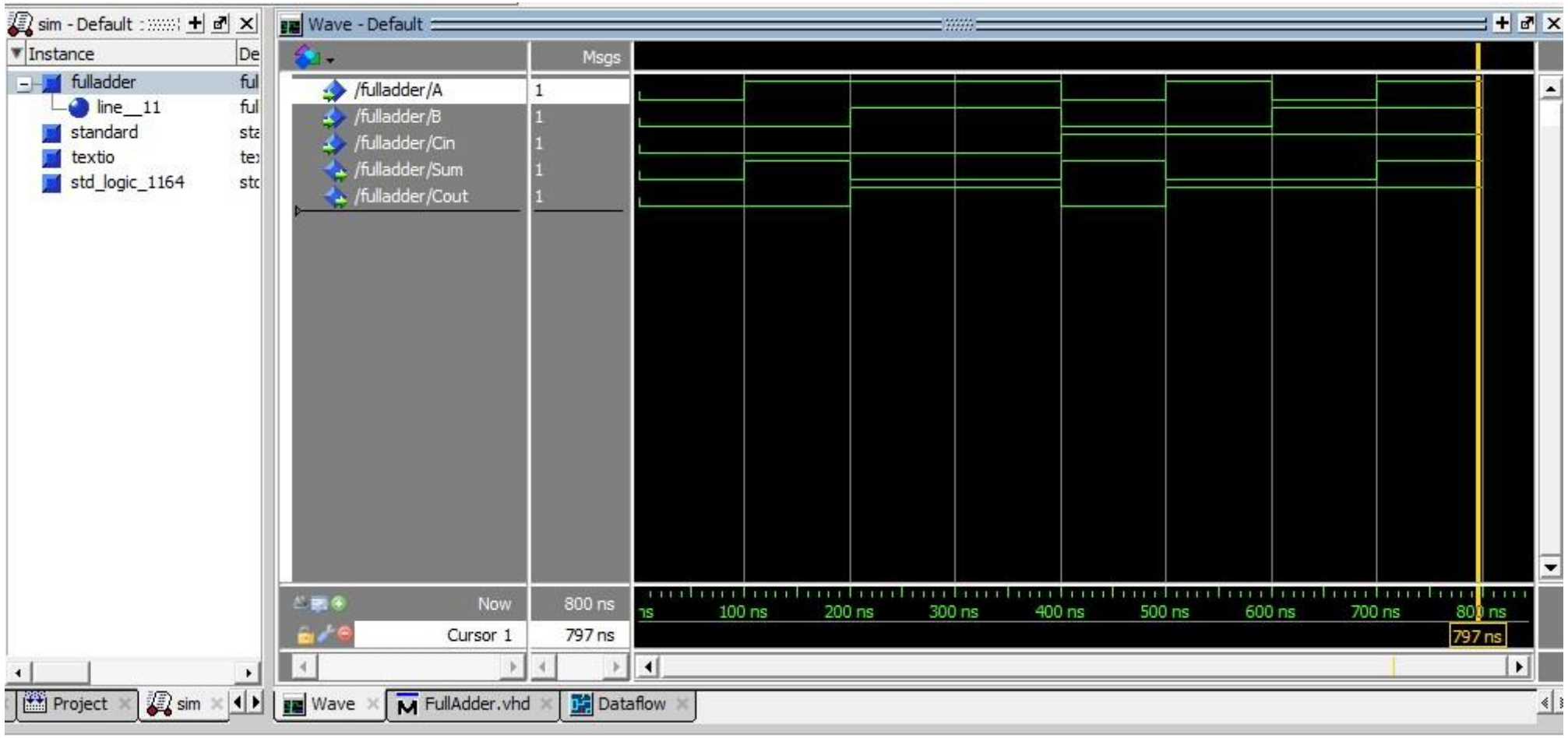
Truth table

Cin	B	A	Cout	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

- **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.

4 Bit Adder Waveform Simulation



- **Comparison of Truth Tables and Simulation Results:**

- 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.

VI. CONCLUSION

In conclusion, the lab session on implementing a Full Adder using VHDL has been insightful and beneficial. By successfully designing and simulating the Full Adder, we have gained a deeper understanding of VHDL and its application in digital circuit design. The Full Adder is an essential component in digital systems, as it enables binary addition with carry-in and carry-out functionalities.

The results of the lab, including the truth table and waveform simulations, have validated the correctness of the Full Adder implementation. The outputs from the simulations align with the expected values, indicating that the VHDL code accurately represents the behavior of the Full Adder.

Through this lab, we have learned how to describe digital circuits using VHDL and apply it to practical designs. This knowledge can be extended to more complex circuit designs, enabling us to create and analyze digital systems with multiple components.

Overall, this lab has provided valuable hands-on experience in VHDL programming and digital circuit design. The implementation of the Full Adder serves as a foundational building block for more advanced digital systems. By mastering VHDL and understanding the principles behind the Full Adder, we are better equipped to tackle future projects in digital circuitry and design.