Design and Implementation of Booth's Algorithm for Signed Multiplication

Technical Report – Homework 4

Logan Pride

Course: Contemporary Digital Systems ECE501

Instructor: Dr. Balster

October 28, 2025

Contents

1 Introduction

In this assignment, the student was asked to investigate a well-known VHDL algorithm, called Booth's Algorithm. Booth's Algorithm was derived from the early work of electrical engineer, physicist, and computer scientist Andrew Donald Booth. The structure of Booth's Algorithm has enabled VHDL designers to efficiently perform signed multiplication using minimal hardware resources, reducing computation time and simplifying implementation. In the implementation for Homework 4, the student was then asked to demonstrate their understanding of the algorithm by creating a behavioral VHDL design along with a corresponding testbench, all guided by a set of specifications, parameters, and an input CSV file. Throughout the report, readers are able to follow the objectives, descriptions, processes, and results presented throughout the report, as well as the practical lessons and insights that Booth's Algorithm contributes to VHDL design

1.1 Project Objectives

The primary objectives of this project are:

- Develop VHDL code that implements Booth's Algorithm for signed multiplication.
- Design the multiplier to accept N-bit inputs and produce a 2N-bit output, instantiated with N = 8.
- Implement a start input signal to trigger the multiplication process and a `data_ready` output to indicate completion.
- Create a cohesive testbench to verify the functionality of the design.
- Record results in simulation in ModelSim
- Record results from configuration in Quartus

1.2 Booth's Method/Algorithm

Booth's Algorithm examines pairs of bits from the product register, which contains 2N+1 bits after combining the operands. The algorithm analyzes the last two bits to determine the next operation. If the pair is `"01"`, the value in A_reg— (see Section 2.1, Architecture Overview) is added to the product register. If the pair is `"10"`, the value in S_reg— (see Section 2.1) is added instead, effectively performing a subtraction. If the pair is `"00"` or `"11"`, no addition or subtraction occurs, and the algorithm simply performs a right-shift operation.Below is the chart referenced by the professor and used during analysis to support understanding of the algorithm.

Figure 4.1: Simulation Result

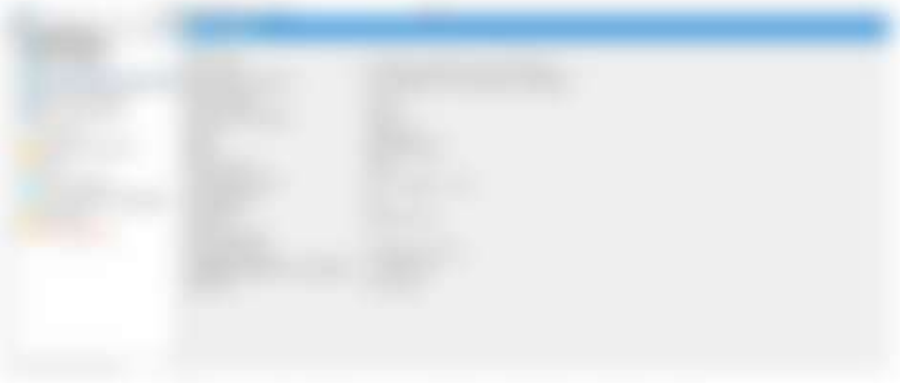


Figure 4.1: Quartus Compilation Results

Compilation Results

- **Status:** Successful - Wed Nov 19 16:42:30 2025
- **Total Variables:** Quartus Prime 64-bit
- **Target Device:** Cyclone IV E EP4CE115F29C7

5 Conclusion

5.1 Summary

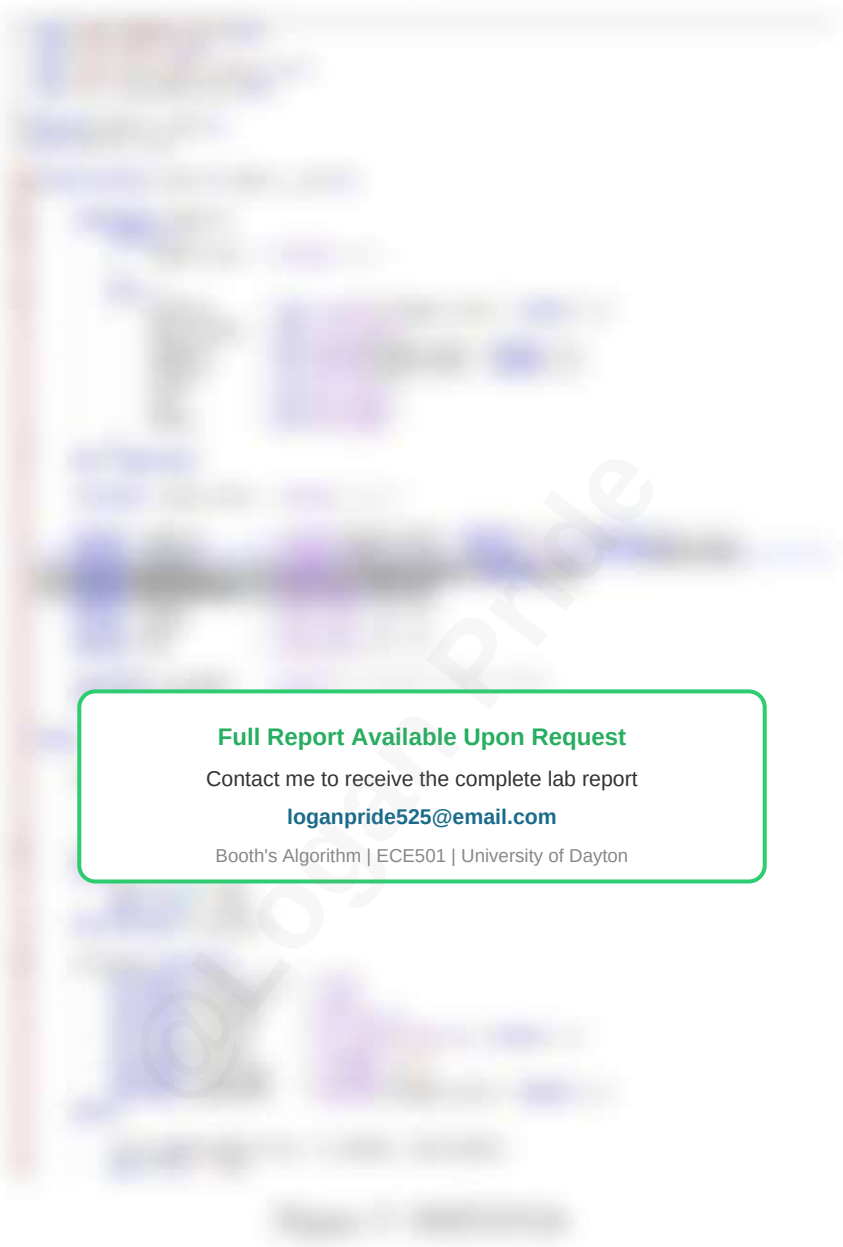
In conclusion, this project demonstrated the successful design, implementation, and verification of a high-performance 16-bit signed multiplier using Booth's algorithm. The multiplier was developed in SystemVerilog and synthesized for the Cyclone IV E FPGA, with results validated through comprehensive testbenches. The design efficiently handles both positive and negative operands, reduces the number of required additions, and achieves a fixed latency of 17 clock cycles. The project successfully met its objectives of creating a functional, optimized, and well-documented multiplier suitable for integration into larger digital systems.

Full Report Available Upon Request

Contact me to receive the complete lab report

loganpride525@email.com

Booth's Algorithm | ECE501 | University of Dayton

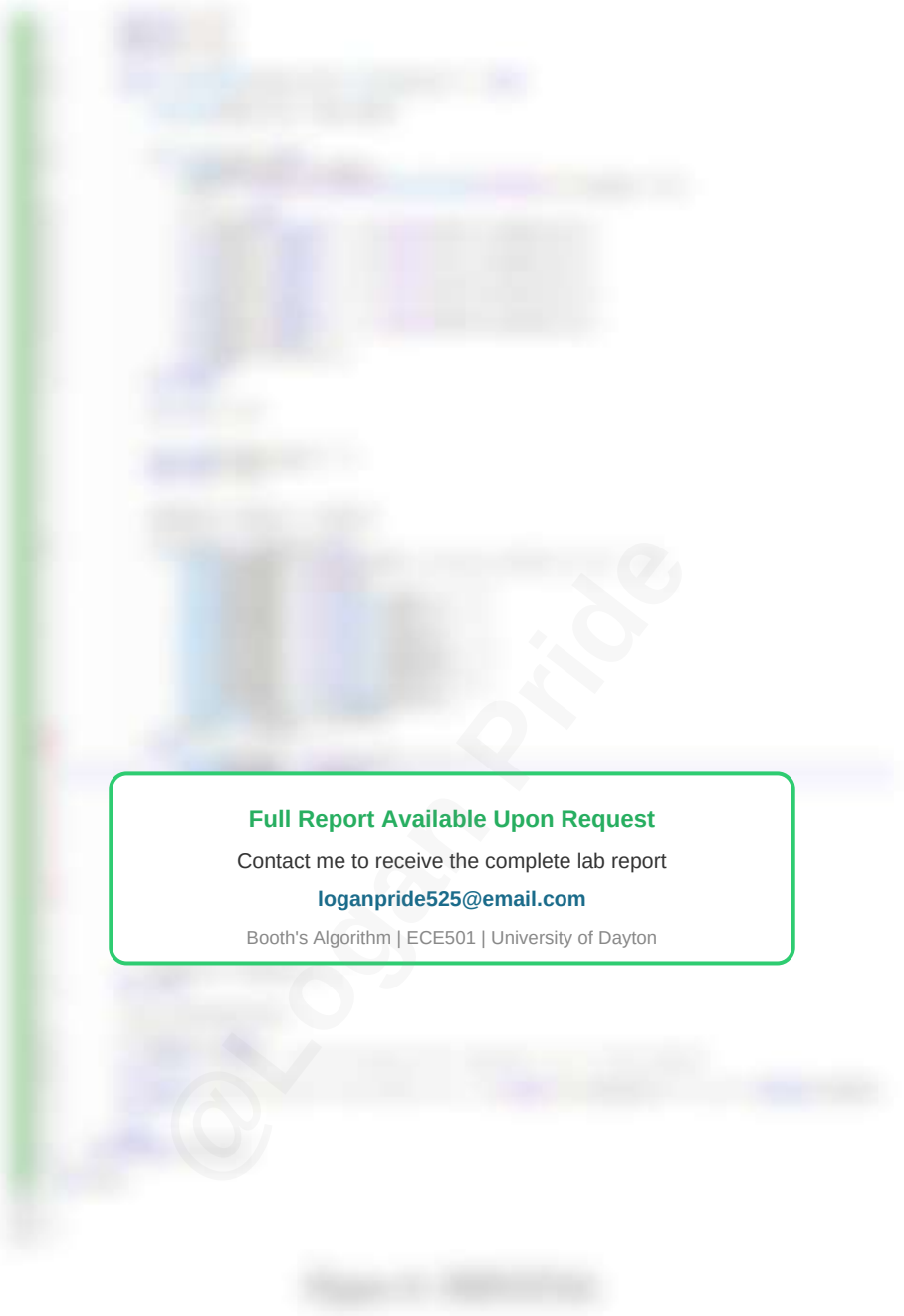


Full Report Available Upon Request

Contact me to receive the complete lab report

loganpride525@email.com

Booth's Algorithm | ECE501 | University of Dayton



Full Report Available Upon Request

Contact me to receive the complete lab report

loganpride525@email.com

Booth's Algorithm | ECE501 | University of Dayton