Digital Design Lab
EEN 315

Project 2
Four-bit Multiplier

Group 2
Andrew O'Neil-Smith
(Partner: Austin Clifton and Amy Zamon)

Sayan Maity, TA

University of Miami
November 18, 2013

Abstract

This lab is to move forward with digital implementation and make a four-bit multiplier using the add and shift concept. The main point of the project was to use the external FPGA board and get used to assigning pins and downloading the files to the board.

Table of Contents

Overview

Previous knowledge required was how to use Quartus II software, basic logic circuitry, design using Truth tables, flip flops, and multiplexors. We needed to know about the adder and half-adder, subtraction, multiplication, and ALU.

Objectives

Introduce design of arithmetic circuits, understand advantages of register storage, concepts of logic control, and programmable logic. Learn how to use an FPGA.

Equipment

Quartus II software

Description	Quantity
74194 4 bit register	2
74163	1
MUX 4:1	1
7483	1
74198	1
AND gate	4
Inverter	2

Description

First we made our block diagram layout file in Quartus II. Since there were a lot of components with a lot of pins, it was challenging to wire them together. Each four-bit shift register was a number to be multiplied. A modulo N counter was used for the "Add and Shift" algorithm. The 8-bit shift register was used to handle four Add and Shift operations, as well as maintaining the final output. We designed and implemented a four-bit multiplier using the conventional "add and shift" algorithm.

Then we simulated the Vector Waveform File. After ten clock pulses, we got the right answer. The final step was to download it to the DE2 board. We had to figure out how to assign the pins from our Block Diagram File. Once we assigned the pins to pins on the FPGA, we were able to download it to the board and set switches and LEDs. Both inputs to the multiplier are fed from two different sets of switches. We used a reset switch and a start switch. The clock was assigned to a push button for manual triggering. We had trouble with confusing the multiplicand and multiplier, but once we used the right set of switches, our problem went away.

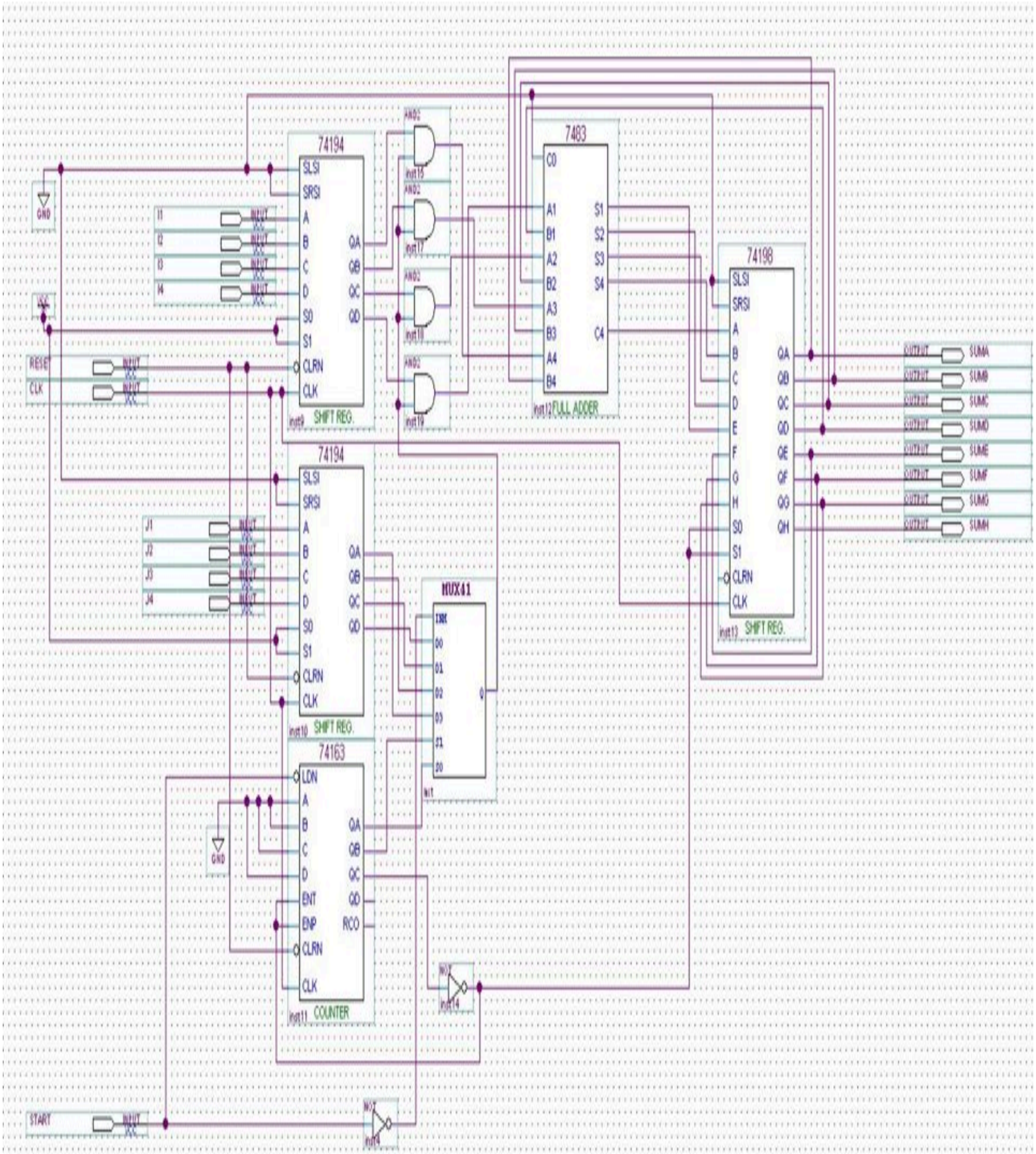
Specifications

Take Multiplier as 9 (1001) and Multiplicand as 10 (1010). Also take Multiplier as 5(0101) and Multiplicand as 13 (1101). Do the Multiplication in pen and paper way also using add and shift algorithm. Simulate using Quartus II Software and show that each and every state generated is same as the state generated in add and shift algorithm. Download the program implemented in Quartus II Software in the DE2 board as explained in the reference material. Assign 4 input pins for multiplier input and 4 input pins for multiplicand input and 4 input pins for the counter (loadable) input. Assign an input pin load to input the multiplier and multiplicand in the registers and then reset all the multiplier and multiplicand inputs as zero. Use a Key button on the board to manually handle every event going to taken place in every clock pulse. Assign all the 8 bits of the output (result of the multiplication) in 7 consecutive LED's on the DE board and show that in every clock pulse the change in the output is according to the states generated in the add and shift algorithm. Do it for the 2 combinations of Multiplier and Multiplicand mentioned.

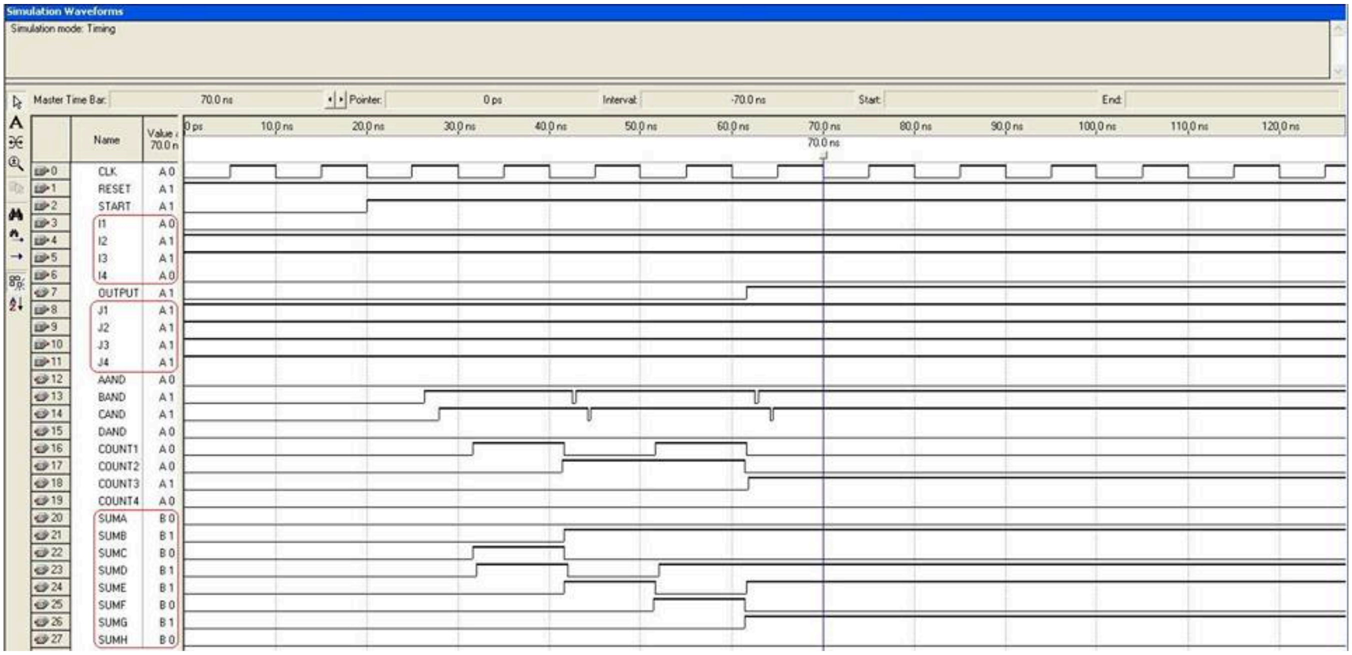
Design Synthesis

We did not have to design much logic for this project. The point of the project was to test our knowledge of the "add and shift" algorithm, as well as using adders and shift registers.

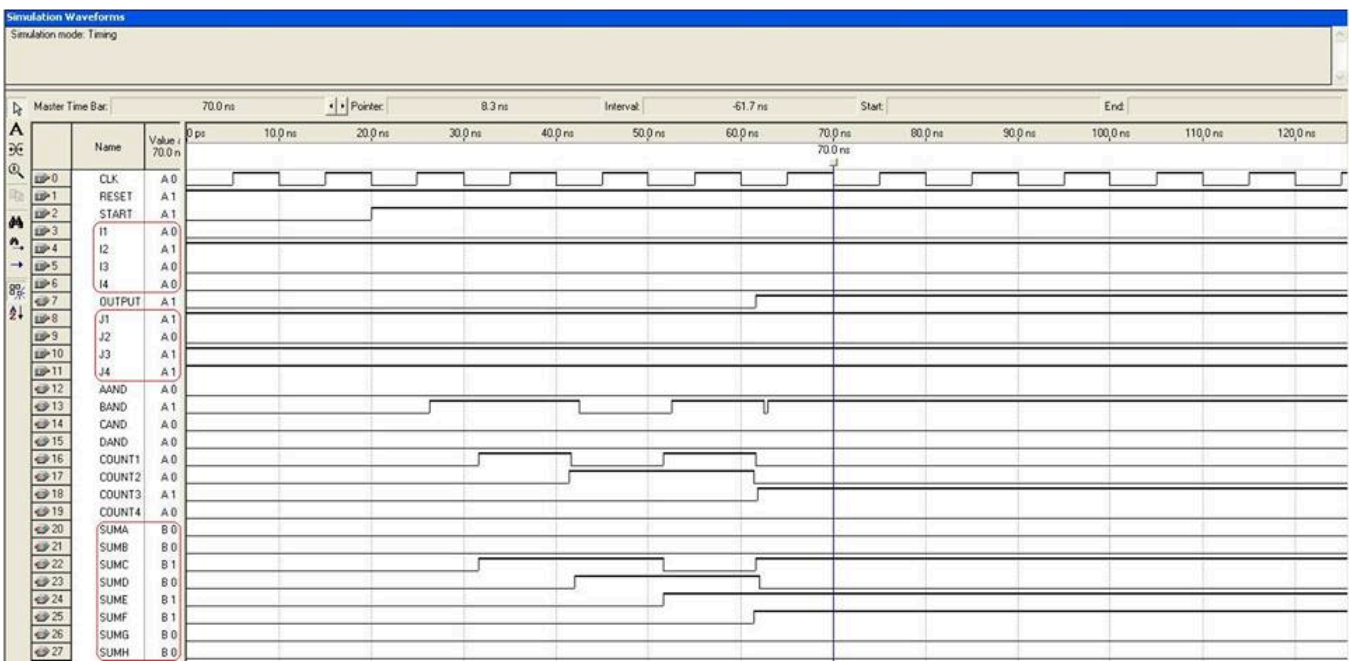
Complete Logic Diagram



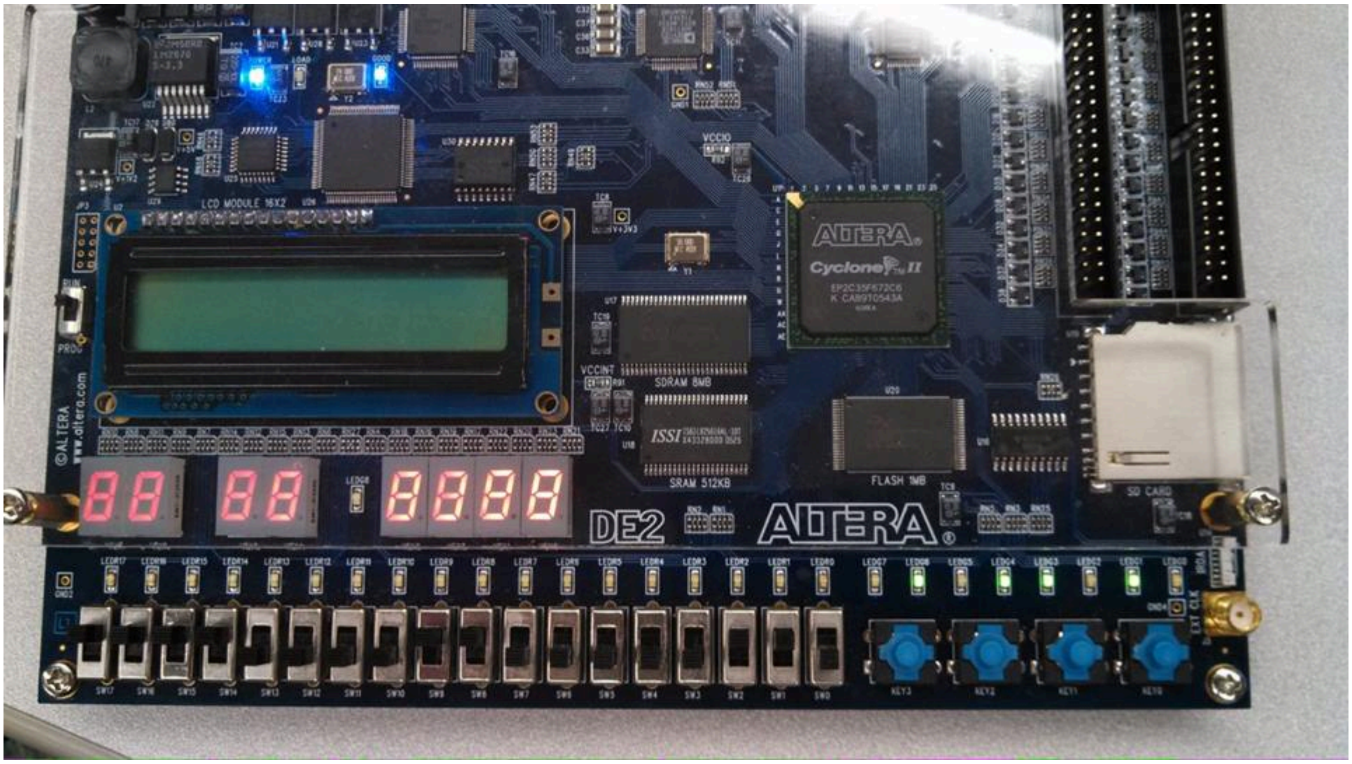
Results and Simulations



9x10



5x13



1111 x 0110 = 01011010

Answers to the questions in the lab handout

none

Conclusion

We were able to successfully implement the circuit in both Quartus II and on the board. We were able to use alternative logic in our block diagram. We initially had problems getting the board to display the correct result, but it was simply a matter of switching the multiplier and multiplicand.

Works Cited

none