

Digital Design Lab
ECE 315

Lab/Project #5B
Multiplier

Group #7

Karysse Hay, Nikeem

Dunkelly-Allen and Dibyanshi Mishra

Zeinab Ramezani, TA

University of Miami

4/1/2022

Overview

The purpose of this lab was to design a 4-Bit Multiplier using the add and shift algorithm. This would allow for two 4-Bit numbers (X and Y) to be multiplied to produce an 8-Bit result. In the first clock cycle, X (first input) is added to P (multiplicand which starts at zero). If the least significant bit of Y (second input) is '1'. X is shifted left while Y is shifted right. This process was repeated in subsequent cycles and completed when Y=0. It took no more than 4 clock cycles on the rising edge. It used an 8-Bit Register and a 4-Bit Adder to produce an 8-Bit result. The technique is simply one of long multiplication.

The 4-Bit Multiplier was combined with circuits from previous labs such as the Counter circuit, BCD circuit and seven segment display circuit to build a Top Level circuit. This final circuit effectively displayed the output on the 7 segment displays of the DE1-SoC Board. In the DE1-SoC board, X and Y were set up as the inputs while the last two segments represented the product of X and Y. The circuit was efficiently able to display the multiplication when tested for multiplication.

Equipment

- Quartus Prime Software
- AND gates
- OR gates
- NOT gates
- ModelSim
- Input Pins
- Output Pins
- Orthogonal Node
- Diagonal Node
- DE1-SoC Board

Description

A project using the Quartus Prime software was created. After the project was launched, New Project Wizard was selected. The location and name of the file was chosen, an empty project was selected, and the flowing wizard was left blank. For the next selection, the

DE1 SoC board was chosen. Under simulation the Tool Name was changed to simulations and the format to VHDL. After the window changed from the file menu, a new Block Diagram/Schematic option was chosen. The Four Bit Multiplier was then created.

The Four Bit Multiplier is a combinational circuit that consists of the 4-Bit Adder from Lab 3, the 8-Bit Register from Lab 5A, four AND gates and a 4:1 Mux. There were four inputs, Y_3 , Y_2 , Y_1 and Y_0 connected to the 4:1 Mux which selected one bit at a time and was denoted as Q. The output from the Mux was connected to 4 AND gates, with four new inputs, X_3 , X_2 , X_1 and X_0 . The four AND gates gave the output A_3 , A_2 , A_1 and A_0 . These outputs and QE, QF, QG, and QH (outputs from Register) were inputted into a 4- Bit Adder that was created in Lab 3. The five outputs from the 4-Bit Adder were connected to the 8-Bit Register. The 8-Bit register was also fed inputs Reset, Clock, Load, and three inputs QB, QC and QD (outputs from Register). The eight outputs of the 8-Bit register were labeled QA, QB, QC, QD, QE, QF, QG and QH respectively with QA representing the most significant number . After all the components were connected, the logic diagram was compiled and simulated and a symbol block was created for it to be used in the next part of the lab.

Once the 4-Bit Multiplier was converted into a symbol, it was used in the Top Level Circuit. The Top Level circuit consisted of the 4-Bit Multiplier, the BCD circuit from Lab 4, the Counter from Lab 5A and the Seven Segment Display Circuit from Lab 2. Eight new inputs, X_0 , X_1 , X_2 , X_3 , Y_3 , Y_2 , Y_1 and Y_0 along with a Clock and Reset were connected to the 4-Bit Multiplier. A Counter was added and connected to Reset and inverted Clock. The Counter's outputs were connected to the 4-Bit Multiplier. The outputs (QA-QH) from the Multiplier were then fed into the BCD circuit, which connected to the Seven Segment Display Circuit. The display circuit gave 14 outputs D00-D06 and D10-D16. These were assigned to pins so that it can be displayed on the DE1-SoC board.

The DE1-SoC board was set up using the pin planner where a switch was assigned for Reset (PIN_AE12) and a Pulse switch was assigned for the Clock (PIN_Y16). Toggle switches were also assigned for X_3 , X_2 , X_1 , X_0 , Y_3 , Y_2 , Y_1 and Y_0 . The first four segment displays were assigned to X and Y inputs while the last two segments displayed the product. The circuit was compiled, simulated and downloaded to the board. The display was tested for several combinations of X and Y, and the output of the product was correctly displayed each time.

Design Synthesis

Truth Table:

PS	NS
CBA	$C^+B^+A^+$
000	001
001	010
010	011
011	100
100	000
101	000
110	000
111	000

Table 1: Truth Table for 3-bit counter

	CB 00	01	11	10
A 0	1	1	0	0
1	0	0	0	0

Table 2: K-map for A^+

	CB 00	01	11	10
A 0	0	1	0	0
1	1	0	0	0

Table 3 K-map for B^+

	CB 00	01	11	10
A 0	0	0	0	1
1	0	1	0	0

Table 4: K-map for C^+

Boolean Equations:

$$A^+ = \overline{A} \overline{C}$$

$$B^+ = A \overline{B} \overline{C} + \overline{A} B \overline{C}$$

$$C^+ = A B \overline{C} + \overline{A} \overline{B} C$$

Complete Logic Diagram:

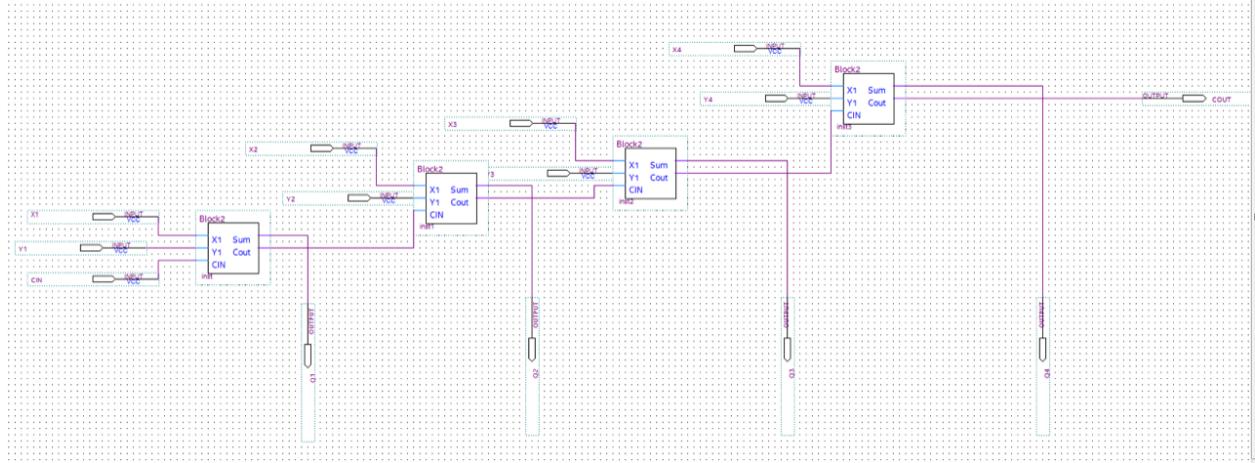


Figure 1: Logic Diagram for 4-Bit Full Adder From Lab 3

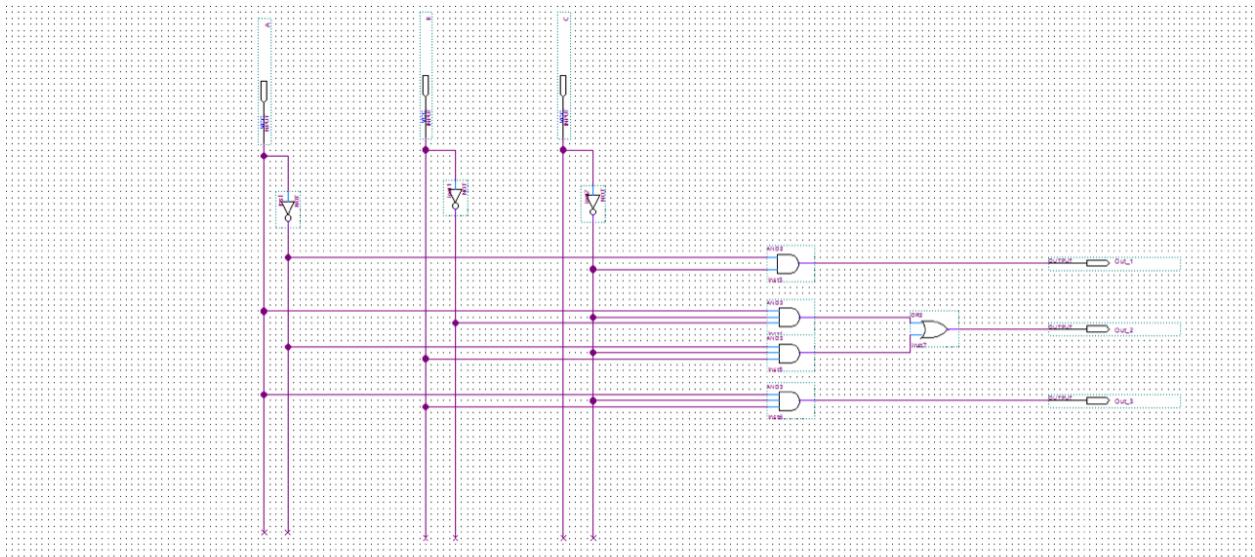


Figure 21: 3-Bit Up Counter Logic Diagram From Lab 5A

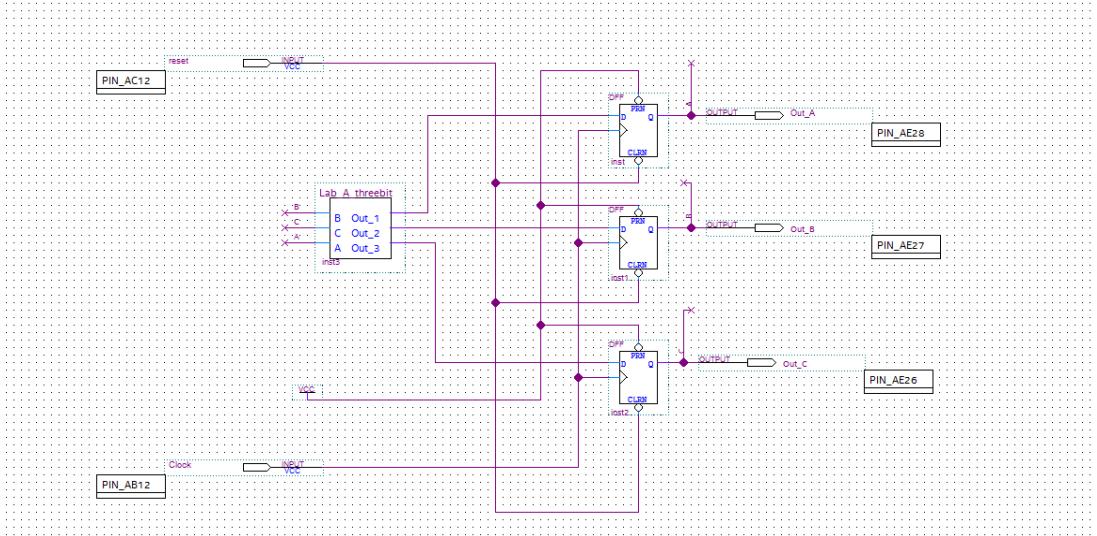


Figure 3: Counter Logic Diagram From Lab 5A

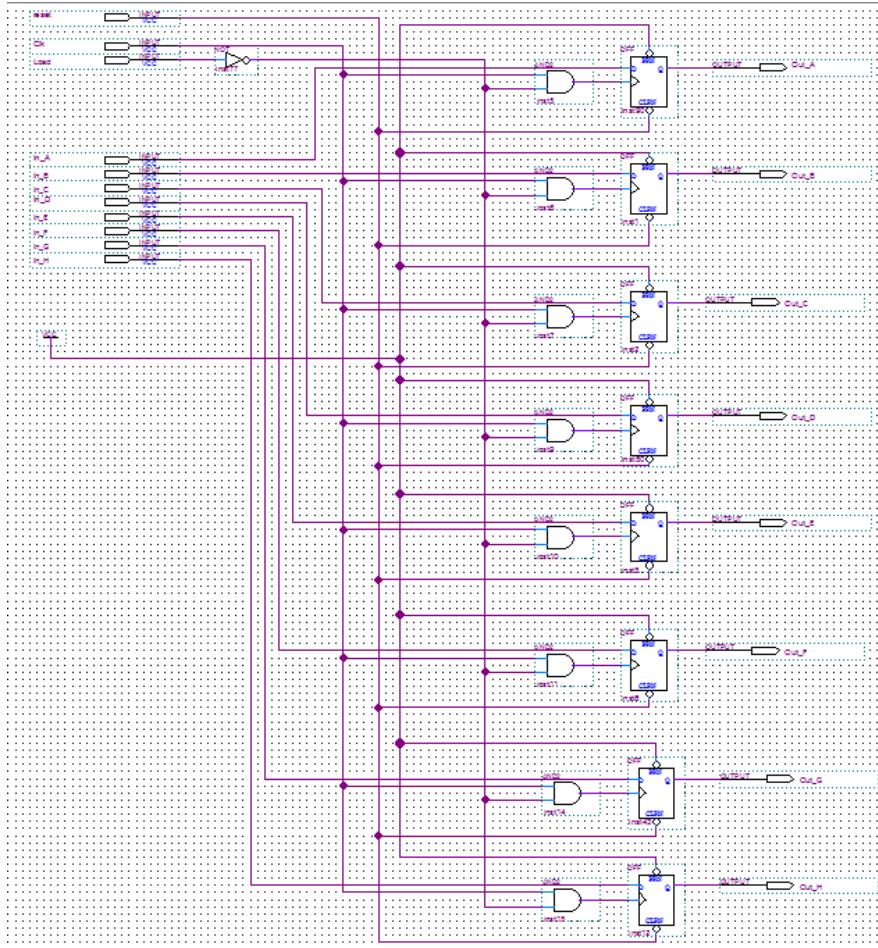


Figure 4: 8- Bit Register Logic Diagram From Lab 5A

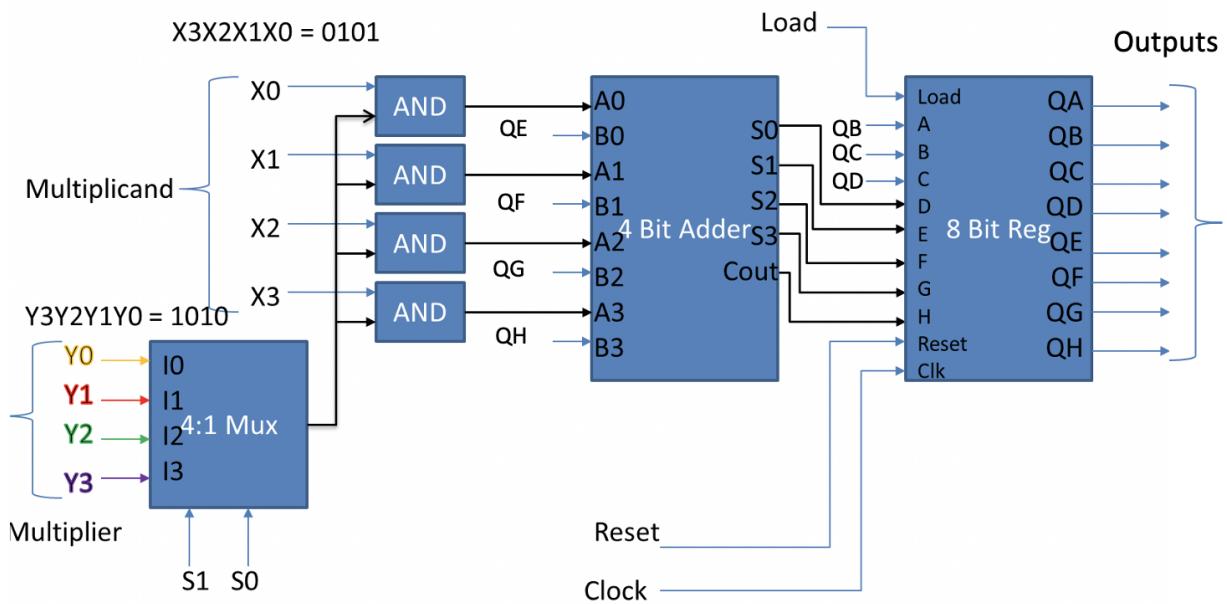


Figure 5: 4-Bit Multiplier Block Diagram

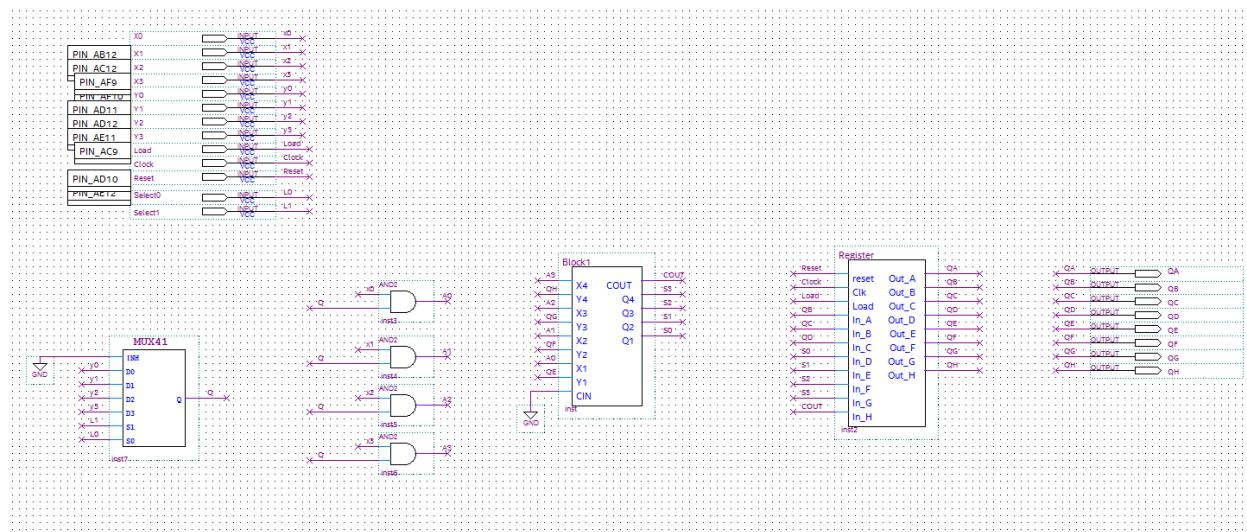
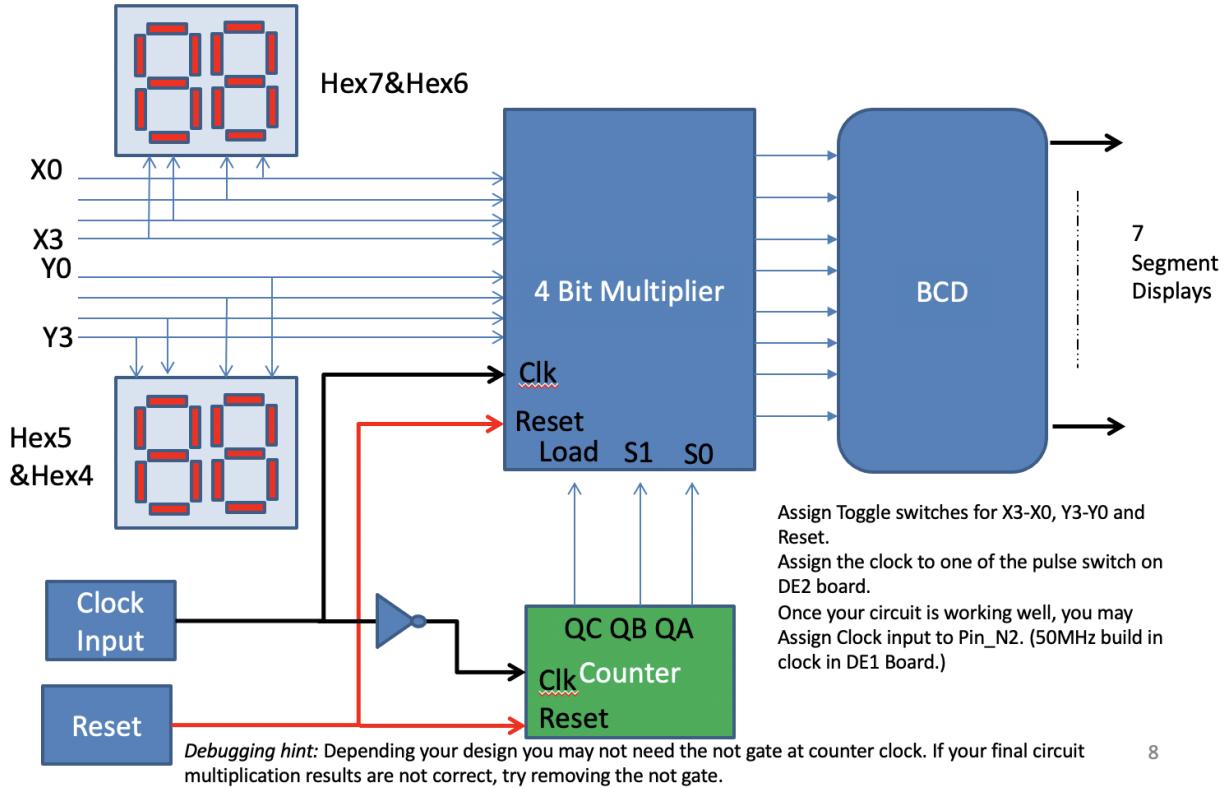


Figure 6: 4-Bit Multiplier Logic Diagram



8

Figure 7: Top Level Block Diagram

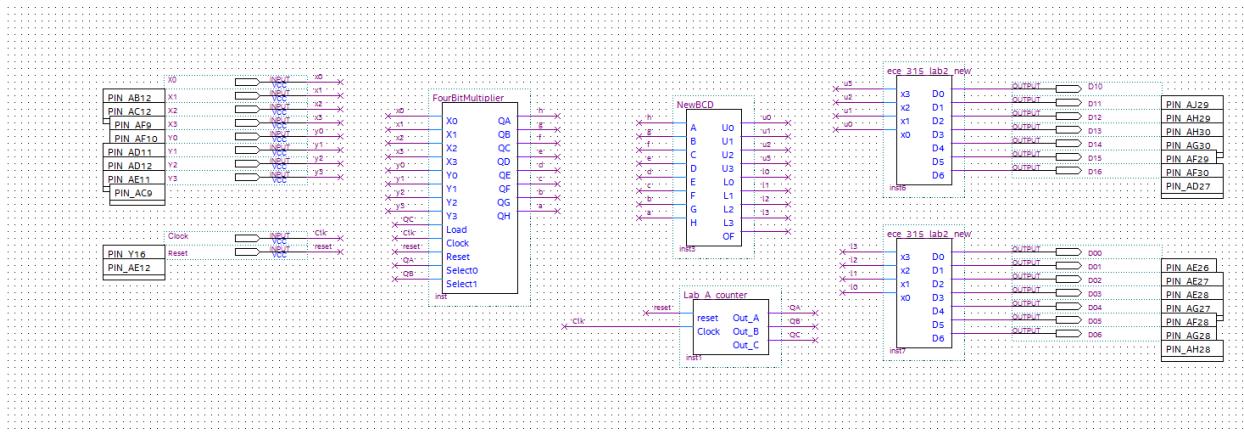


Figure 8: Logic Diagram for Top Level Circuit

Results and Simulations

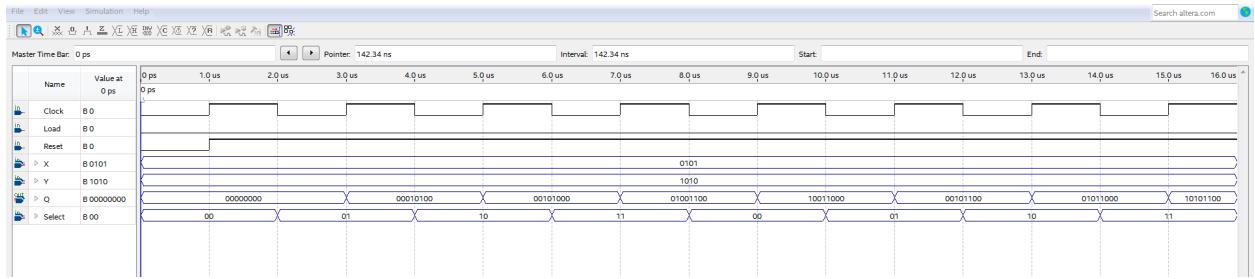


Figure 9: 4-Bit Multiplier Simulation

The 4-bit multiplier goes through four shift phases before showing the correct product in 8 bits. On the fourth rising clock cycle the product of the multiplicand and multiplier is shown. The multiplicand is 0101 and the multiplier is 1010 whose values are 5 and 10 respectively in decimal. The product (Q) shows 50 in binary.

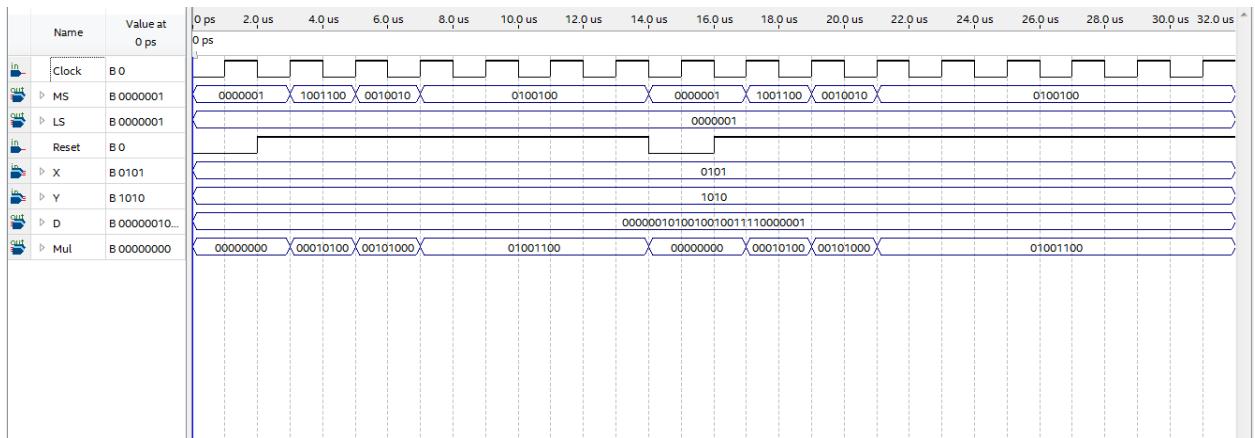


Figure 10: Top Level Circuit Simulation

Likewise with the 4-bit multiplier simulation, the multiplier goes through four shift phases before showing the correct product of the multiplicand and multiplier. This simulation shows the output of the seven-segment display using two displays: MS and LS. Most significant is labeled as MS and least significant as LS. Using the same multiplicand and multiplier from the 4-bit simulation (Figure 9), after the fourth rising clock cycle MS shows 5 and LS shows 0, combined together is 50 thus the correct product. This can be shown on the DE1-SoC board in Figure 11.

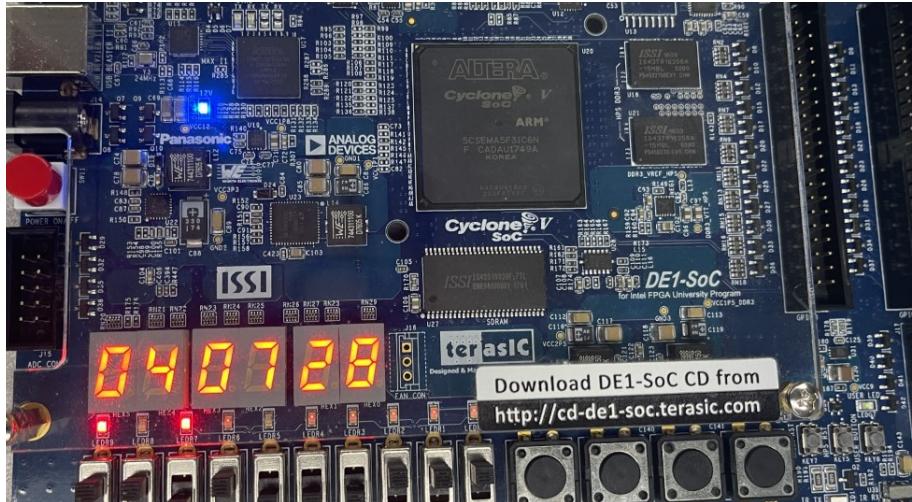


Figure 11 : Multiplier downloaded on DE1-SoC board

The Multiplier was downloaded to the board. The first four segment displays represent X and Y (inputs). The last two seven segment displays represent the product of X and Y. The first switch was assigned to Reset and the first Pulse button was assigned to the Clock.

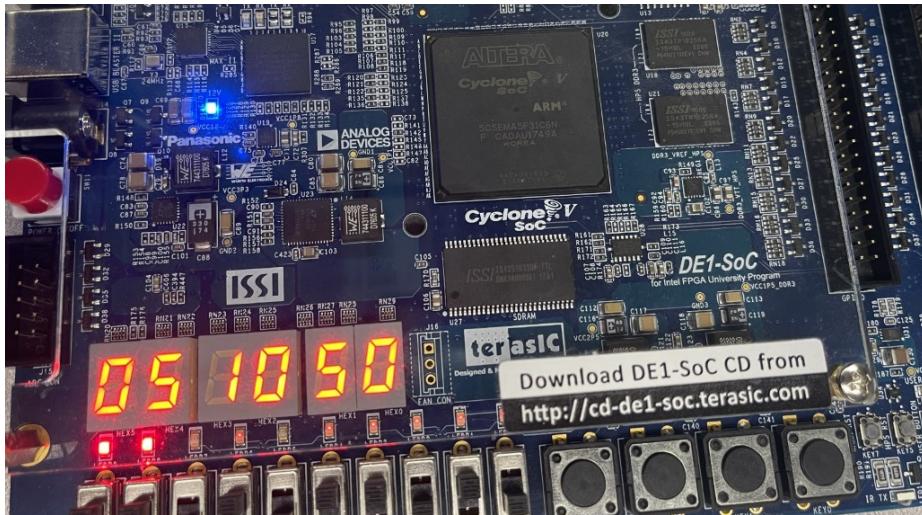


Figure 12: Multiplier downloaded on DE1-SoC board

This is another example of The Multiplier. The Figure above shows the multiplication of $X=5$ and $Y=10$ which is 50.

The purpose of the 3-Bit Up Counter was to perform a single multiplication between the 4 Bit input X and a single bit of input Y. This is because only one value can be processed at a time by the 4-Bit Adder and shifted to produce the entire solution of the multiplication of X and Y.

Conclusion

The Four Bit Multiplier was a combination of circuits from previous labs, 4 AND gates and a 4:1 MUX. This was then connected to the Top Level Circuit. This consisted of the 4-Bit Multiplier, along with the Seven Segment Display Circuit, BCD circuit and Counter circuit which allowed for the two inputs and the product of the inputs to be displayed on the DE1 board with the fourth rising edge of the clock.

The original 3-bit up counter originated from Lab 5A counted from 0, 1, 2, 3, 4 and then returned to the number 0. We found this to be faulty and modified it to hold at 4 instead of immediately returning to the number 0. When immediately returning to zero it would display the wrong product, changing this gave us the correct answer.

The pulse button was assigned for the Clock, and the first switch was assigned to Reset. The pulse button assigned to the clock had to be pressed four times to show the correct product. Each press of the button showed the shifting process until the fourth push where the correct answer was shown. The first four segment displays represented the X and Y inputs while the last two segments displayed the output of the product. There were no limitations of the circuit built for the lab as seen in the results. When X=5 and Y=10, the board correctly displayed the product of 50.

The most difficult part of our lab was the debugging process. The multiplier was outputting the incorrect answer on the seven segments even though the simulation on Quartus Prime was correct. We learned how to use different techniques to efficiently debug our program and fix the issues that arose.