ECE472 Lab 1 **Ashtyn Moehlenhoff**
Torben Rasmussen

Part 1: 16 bit ripple adder

Desciription: In order to implement the 16 bit ripple adder, we simply added on to the 8 bit ripple adder. We increased our array sizes to 16 bits and connected 8 more full adders. You can see in figure 1 that the sum of A and B (for a small sample) is correct.

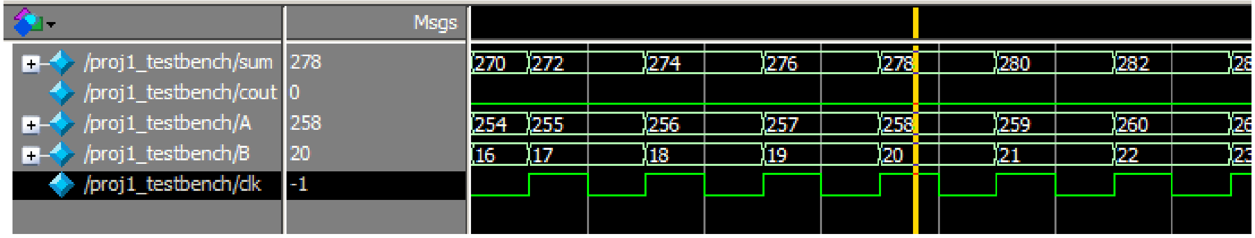

Figure 1: 16 bit ripple adder

Part 2: 16 bit carry-lookahead adder

Desciription: First we got a 4 bit carry-lookahead adder working by making a lookahead module and a fulladder_LA module. We used both these modules in a ripple_adder_LA module to calculate the sum of two 4 bit numbers. In order to create a 16 bit CLA we just strung together four 4 bit CLAs (connecting their carry-in and carry-outs). Figure 2 is a simulation of our 16 bit CLA which adds A and B, you can see the correct sums being calculated for a small sample size.

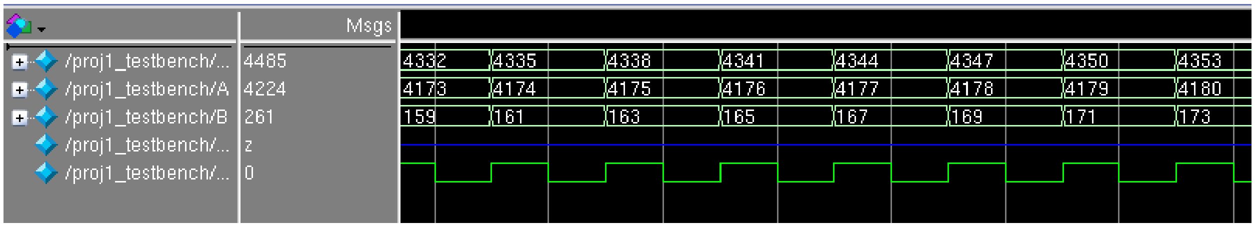

Figure 2: 16 bit carry-lookahead adder

Part 3: Simulating with delay

Description: You can see in Figure 3 and Figure 4 that it takes more time to calculate the sum of A and B (because we added in a delay for each logic function). We have circled the sum of A and B in red in the diagrams. We displayed the worst case for A and B, which is when A = 0xFFFF. This is worst case because it will generate a carry on the addition of each bit.

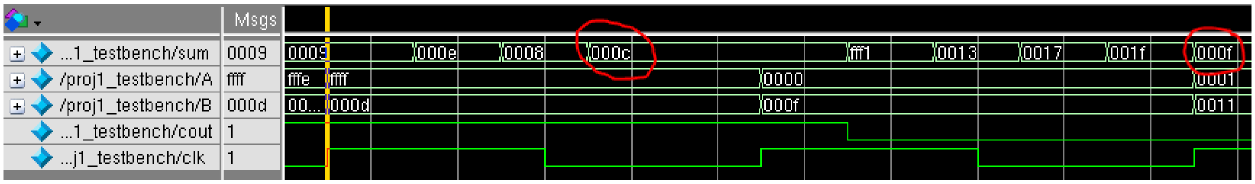

Figure 3: 16 bit ripple adder with delays

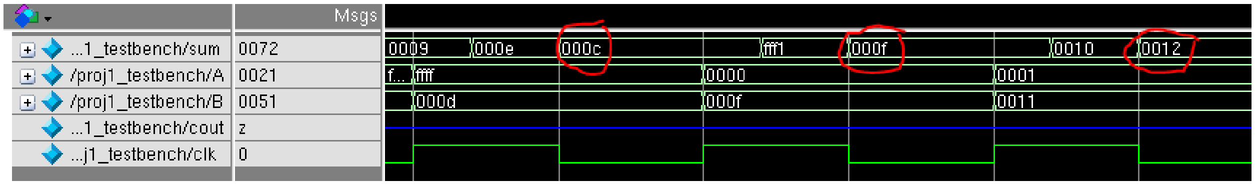

Figure 4: 16 bit carry-lookahead adder with delays

Part4: Reflection

Estimated Time: 6hours

Most Valuable: Getting familiar with Verilog syntax and Modelsim.

Least Valuable: Debugging Modelsim, dealing with crashes in Modelsim on the lab computers, etc…

Suggestions: More information on defining inputs and outputs for each function so we would have had a more clear idea of what we were doing.