ECE 128: FPGA Laboratory

Lehigh University
Department of Electrical and Computer Engineering
Jonah Gibson

12 September 2023

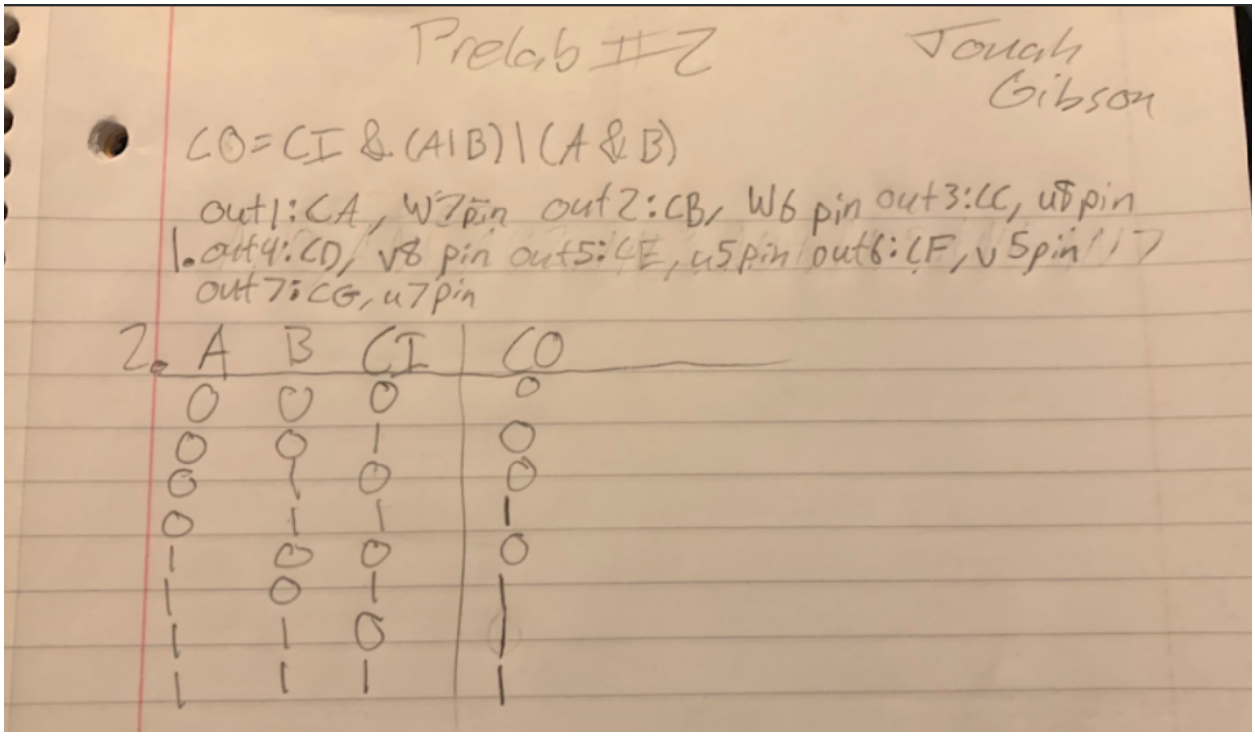
Lab #2: One Bit Full Adder

1. Objectives

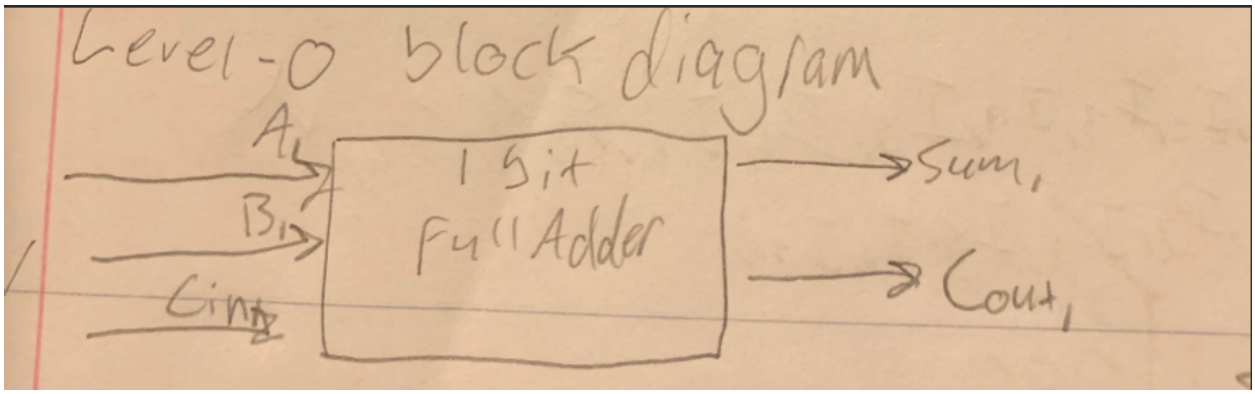
The objective of this lab is to be able to confidently use the Xilinx and Vivado software to design, build, and troubleshoot a one bit full adder with three inputs, A, B, and a carry in CI, as well as two output for the sum and the carry out on our provided boards. At the end of the lab, the board should be able to be interacted with to demonstrate the adder's functions. This lab is important because it allows the students to gain a better understanding of how to implement a vital piece of hardware that will be used throughout their careers, as well as seeing an adder in real life, as opposed to the theory we are taught in the classroom. While it is not the easiest lab we will tackle, it is one of the most important labs, as it contains many fundamental topics of the course.

2. Introduction

To begin this lab, it was necessary to draw a block diagram and gate level model to get a better understanding of what the adder was and how it would be translated to the software and hardware in the lab. The written form of these diagrams can be seen below.



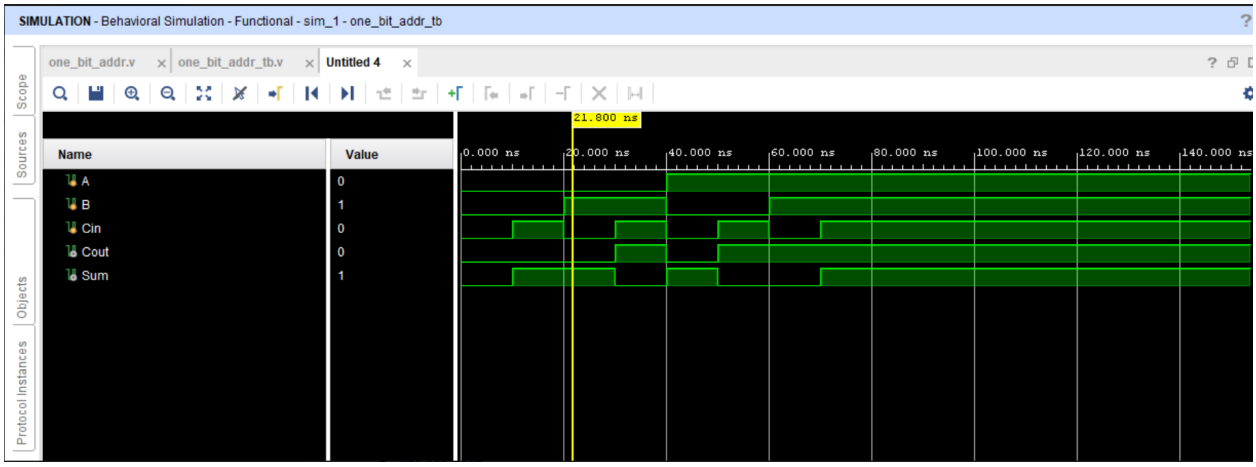
Pictured: Truth table and seven segment display pin assignments for a full adder. The truth table uses the three inputs A, B and CI, as well as 1 output for the CO result.



Pictured: Gate level model for a one bit full adder. You can see the same three inputs going into the adder, and the outputs Cout and Sum.

3. Results

In the end, we were able to get our project working as intended, and ultimately got it approved by the Professor.



Pictured above: the functioning behavioral simulation of the adder.

The only difficulties I encountered were when my partner and I were attempting to run an implementation of the design. We were confused, as the project was synthesizing fine, but when we tried implementing it, our testbench disappeared from the file navigator. We tried many different approaches to remedy this, but eventually, we figured out that in our testbench, we had mistakenly called the name of our testbench file when using the uut function. We should have used the name of the original file. Once we identified the source of the problem, we were able to swiftly remedy the situation and complete the lab without a hitch.

4. Conclusions

In the end, I was able to get every part of this lab working as intended. As mentioned in the “Results” section, I had a few hiccups along the way, but I was able to overcome them by getting some help from Professor Naher, as well as looking deep into the program with my partner. This resulted in me being in the lab longer than I had hoped, but I was able to work through it in a timely fashion nonetheless.

5. Contributions

I worked on this lab in collaboration with Jonah Burd, and we took turns using the keyboard to make edits and changes to our file. Seeing as this was a pretty straightforward and quick lab, it was difficult to truly split the work, so I would consider our contributions to be roughly the same.

6. Appendices

The code for our project is available on GitHub.