Assignment 2 - Arithmetic Logic Unit

COEN 122L - Fall 2018
By: Ryan Khodi (rkhodi@scu.edu)
TA Office Hours: By appointment.
Professor: Dr. Weijia Shang (wshang@scu.edu)

Description

The Arithmetic Logic Unit (ALU) is a hardware component that is in charge of handing various arithmetic operations. The ALU decides which operation to perform based on the value of the 4-bit Opcode. Our ALU will need to be able to perform the following operations on its two 32-bit inputs: Add, Increment, Negate (2's complement), Subtract, and Pass A. In addition, the ALU is also in charge of setting both the ZERO and NEGATIVE control flags depending on whether the output is either a zero or negative value.
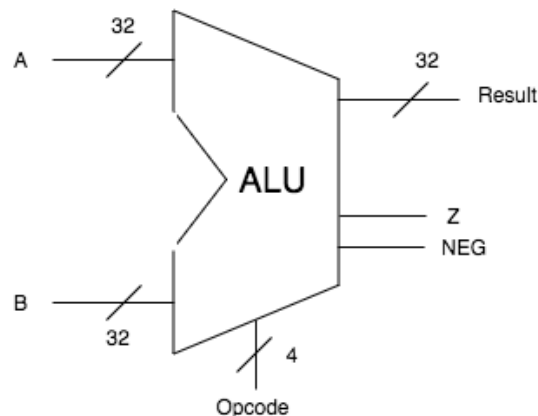

Figure 1: ALU

Assignment

In this lab, you will create an ALU using Verilog. The inputs of this component should include: two 32-bit data, and one 4-bit Opcode (even though 3 bits is enough to cover the number of operations we need we are still going to use 4). The output of this component should have: one 32-bit result, two 1-bit flags (ZERO and NEG). As shown in figure 1.

Helpful notes:
1) The negate operator in Verilog is ˜
2) The NEGATE operation should return the 2's complement. Therefore, it is not just as simple as using the negate operator. Finding the 2's complement of a number takes two steps: flipping all the bits, and then something else.
3) When subtracting, make sure you use the 2's complement of A instead of the negation (same as note 2).
4) Don't forget to set the ZERO and NEG flags after you complete the logic for the output.

The logic for each of the of the operations goes as follows:

Operation	Opcode	Logic
ADD	0000	A + B
INCREMENT	0001	B + 1
NEGATE	0010	0 + ~A
SUBTRACT	0011	B + ~A
PASS A	0100	A + 0

Figure 2: Truth Table for ALU

Deliverables

To receive full credit, you will need to demo your working code. In addition, you must submit your source code (commented), your test-bench code (commented), and a screenshot of your waveform. To submit online, make sure everything is in a zipped folder (name the folder firstname_lastname.zip) and turn it into Camino. Please copy your code into individual .txt files and include those in the folder.