# CprE 381 – Computer Organization and

# Assembly-Level Programming

# Proj-A Report

Lab Partners Chimzim Ogbondah & Parth

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Section / Lab Time: 4 Thursday 4:10-6 pm

***Submit a typeset pdf version of this on Canvas by the due date. Refer to the highlighted language in the Proj-A instructions for the context of the following questions****.*

1. [Part 0] Updated Project Team Contract (if applicable). With your project group members, create a list of best practices / tips for designing, compiling, and testing VHDL modules based on your experiences so far with these labs, both working individually and as a group.
   1. Delegating who will work on what, Setting time a side to make sure that either partner is not struggling or if they are how to help them move pass the roadblock
   2. Coming together to explain how each component works and the working together to integrate the design
2. [Part 1 (a)] Draw a schematic for a 1-bit ALU that supports the following operations: add/sub (both signed and unsigned), slt, and, or, xor, nand, and nor. What are the inputs and outputs that are needed?
3. [Part 1 (b)] In your project writeup, describe your design in terms of the VHDL coding style you chose and the control signals that are required.
4. [Part 1 (c)] Describe how the execution of the different operations corresponds to the Modelsim waveforms in your writeup.
5. [Part 2 (a)] Draw a simplified schematic for this 32-bit ALU. Consider the following questions: how is Overflow calculated? How is Zero calculated? How is slt implemented?
6. [Part 2 (b)] In your writeup, describe what challenges (if any) you faced in implementing this module.
7. [Part 2 (c)] Describe how the execution of the different operations corresponds to the Modelsim waveforms in your writeup.
8. [Part 3 (a)] Describe the difference between logical (srl) and arithmetic (sra) shifts. Why does MIPS not have a sla instruction?
   1. Shift left logical shifts the data to the left and pads with zeros while arithmetic pads with 1s
9. [Part 3 (b)] In your writeup, briefly describe how your VHDL code implements both the arithmetic and logical shifting operations.
   1. It implements both arithmetic and logical shifting by using a 2-1 mux which is given an input of 1 and 0. If they LogArth signal is set to 0 it takes zero and 1 when the signal is one. It then uses stages of shifting 1 bit, 2 bits, 4 bits, 8 bits, 16 bits and at each step adding the LogArth to the amount of bits being shifted at the end. (1, 2, 4, 8, 16) each stage feeds into the next where it either takes logArth if its to be shifted or the output from the previous bit stage shift and then continues along with the shifting pattern where the bit data is moved to the right by the amount corresponding to the shift stage.
10. [Part 3 (c)] In your writeup, explain how the right barrel shifter from part b) can be enhanced to also support left shifting operations.
    1. It allows left shifting by shifting both right and shifting left (which follows the same process of shifting as the right shifting except it flips the process) there is a Signal for shifting that is connected to a mux at the end where if 0 is selected it gives the output right shifting and if it is 1 it gives the output left shifting.
11. [Part 3 (d)] Describe how the execution of the different shifting operations corresponds to the Modelsim waveforms in your writeup.

1. [Part 4(b)] Justify why your test plan is comprehensive. Include waveforms that demonstrate your test program is functioning.
2. [Part 5(c) BONUS] Justify why your test plan is comprehensive. Include waveforms that demonstrate your test program is functioning.
3. [Feedback] You must complete this section for your lab to be graded. Please complete each column **separately** for each team member; I expect it to take roughly 10 minutes (do not take more than 20 minutes).
   1. How many hours did you spend on this lab?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Task** | **During lab time** | | | **Outside of lab time** | | |
| **Team Initials** |  |  | CO |  |  | CO |
| Reading lab |  |  | .2 |  |  | .2 |
| Pencil/paper design |  |  | .1 |  |  | .1 |
| VHDL design |  |  | 1.0 |  |  | .5 |
| Assembly coding |  |  |  |  |  |  |
| Simulation |  |  | .2 |  |  | .5 |
| Debugging |  |  | .5 |  |  | .5 |
| Report writing |  |  | 0 |  |  |  |
| Other: |  |  |  |  |  |  |
| Total |  |  | 2 |  |  | 1.8 |

* 1. If you could change one thing about the lab experience, what would it be? Why?
  2. What was the most interesting part of the lab?
     1. I thought implementing a shifter only using 2-1 muxs was really cool