CS 2110 Timed Lab 1
Arithmetic Logic Units

The Best TAs

Spring 2020

Contents

1 Timed Lab Rules - Please Read

1.1 General Rules

1. You are allowed to submit this timed lab starting at the moment the assignment is released, until you are checked off by your TA as you leave the recitation classroom. Gradescope submissions will remain open until 7:15 pm - but you are not allowed to submit after you leave the recitation classroom under any circumstances. **Submitting or resubmitting the assignment after you leave the classroom is a violation of the honor code - doing so will automatically incur a zero on the assignment and might be referred to the Office of Student Integrity.**

2. Make sure to give your TA your Buzzcard before beginning the Timed Lab, and to pick it up and get checked off before you leave. **Students who leave the recitation classroom without getting checked off or submit after getting checked off will receive a zero.**

3. Although you may ask TAs for clarification, you are ultimately responsible for what you submit. **The information provided in this Timed Lab document takes precedence.** If in doubt, please make sure to indicate any conflicting information to your TAs.

4. Resources you are allowed to use during the timed lab:
   - Assignment files
   - Previous homework and lab submissions (this includes homework PDFs)
   - Your mind
   - Blank paper for scratch work (please ask for permission from your TAs if you want to take paper from your bag during the Timed Lab)

5. Resources you are **NOT** allowed to use:
   - The Internet (except for submissions)
   - Any resources that are not given in the assignment
   - Textbook or notes on paper or saved on your computer
   - Email/messaging
   - Contact in any form with any other person besides TAs

6. **Before you start, make sure to close every application on your computer.** Banned resources, if found to be open during the Timed Lab period, will be considered a violation of the Timed Lab rules.

7. We reserve the right to monitor the classroom during the Timed Lab period using cameras, packet capture software, and other means.

1.2 Submission Rules

1. Follow the guidelines under the Deliverables section.

2. You are also responsible for ensuring that what you turned in is what you meant to turn in. After submitting you should be sure to download your submission into a brand new folder and test if it works. No excuses if you submit the wrong files, what you turn in is what we grade. In addition, your assignment must be turned in via Gradescope. Under no circumstances whatsoever will we accept any email submission of an assignment. Note: if you were granted an extension you will still turn in the assignment over Gradescope.

3. Do not submit links to files. We will not grade assignments submitted this way as it is easy to change the files after the submission period ends.

1.3 Is collaboration allowed?

Absolutely NOT. No collaboration is allowed for timed labs.

2 Overview

In this timed lab, you will be creating an ALU. This ALU will take in **two 8-bit inputs** and **one 2-bit op code**. It will output **one 8-bit output**.

2.1 Allowed Components

When building this ALU, you may only use:

1. basic logic gates (AND, OR, NAND, NOR, NOT, XOR),
2. decoders,
3. multiplexers,
4. the built-in Circuitsim adders (**NOT** the built-in subtractors),
5. splitters,
6. wires,
7. tunnels,
8. constants,
9. input pins,
10. output pins

IMPORTANT NOTE: YOU DO NOT NEED TO BUILD THE GATES OUT OF TRANSISTORS. PLEASE, FOR YOUR OWN SAKE, DON'T DO IT. USE THE BUILT IN GATES.

IMPORTANT NOTE: You're allowed to use CircuitSim's **default, built-in adders** (in the Arithmetic tab). So please don't try to make your own adders, just use that one. You're not allowed to use anything else from the Arithmetic tab (don't try to use a subtractor; if you need a subtractor, you will need to create your own using the above-mentioned components).

IMPORTANT NOTE: Some operations will also have **additional banned components**. Read the instructions for individual operations carefully.

3 Instructions

3.1 CircuitSim Information

For this assignment, you will be using CircuitSim. The version is the exact same as the one used in Homework 2 and 3, and can be found on either Docker or Canvas under Files → Tools → CircuitSim → CircuitSim.v1.jar. To ensure you are on the correct version, check to see if the title bar says "CircuitSim v1.8.1 2110 edition". **If your file does not open in this version of CircuitSim you will receive a 0**. All changes should be made in the ***tl1.sim*** file. Do not move or rename any or the input or output pins.

3.2 ALU Components

You will create an 8-bit ALU with the following operations, using any of the gates listed above. All numbers should be interpreted as 2's complement.

00. (A & ˜ B) ˆ A	`[(A & ~B) ^ A]`
01. signOfA	`[A == 0 return 0, A < 0 return -1, A > 0 return 1]`
10. −3A	`[-3A]`
11. setBit	`[Ex: If A = 1, return 00000010]`

Below are descriptions to elaborate more on the problems above. If you're still struggling to comprehend what the question is exactly asking after reading below, please ask one of your TA's for clarification.

- For the **signOfA** operation, there are three possible outputs:
  - -1 if A is negative
  - 0 if A is zero
  - 1 if A is positive
- For the **setBit** operation, A will be a value from 0 - 7. In a bit vector of all zeroes, set the Ath bit and return the resulting bit vector.
  - Let's do an example for **setBit**. We'll start out with a bit vector of all 0's, which will be 0b00000000. Now, let's say our input A = 3, which means A = 0b00000011. We want to set ONLY the A^{th} bit in binary and leave the rest of the bits to be 0. In this case, A is the 3^{rd} bit. We want our output to be: 0b00001000.
- Notice that -3A, signOfA, and setBit depend solely on the `A` input. **They should NOT rely on `B` being a particular value.**
- This ALU has two **8-bit** inputs for `A` and `B` and one **2-bit** input for `OP`, the op-code for the operation in the list above. It has one **8-bit** output named `out`.
- The provided autograder will check the op-codes according to the order listed above ((A & ˜ B) ˆ A (`00`), signOfA (`01`), etc.) and thus it is important that the operations are in this exact order.

4 Checking your work

You can locally check your grade by navigating to the directory containing `tl1.sim` and running

```
java -jar tl1-tester.jar
```

5 Deliverables

Please upload the following files onto the assignment on Gradescope:

1. `tl1.sim`