EECS 470 Lab 1 Assignment

Note:

- You may collaborate with a neighbor on this lab, but must get an individual instructor check-off.
- The lab check-off is due by **Friday, September 5th**

1 Introduction

Lab 1 is an introductory lab. Before starting the assignment, you should log in to GitHub at github.com/EECS-470 **Lab/Project Setup Guide** on the homepage of the repository. This will direct you on how to access the CAEN Linux environment and your lab/project source files.

Afterwards you can start the assignment below. You will complete the assignment by filling out the lab worksheet (accessed on the Course Website) and getting an instructor check-off sometime in-lab or during office hours over the next week. The check-off is either the instructor's signature or a unique 5-digit code if done virtually.

You will then submit the worksheet to Gradescope. We will only be grading these by looking for the instructor check-off, not the work.

2 Assignment

For the lab assignment you will read multiple Verilog files and answer questions about them on the worksheet. After this you will write a simple SystemVerilog module using what you've learned. For students who are familiar with logic design but have not used Verilog before, this lab is meant as an introduction to basic concepts. For students returning to the language, we are hoping to refresh your memory here.

Open the following files by browsing them on GitHub or downloading and opening them in CAEN. Then answer the questions in each file and have an instructor check-off your worksheet.

File: `1-combinational.sv`

Question 1. Which basic logic gate does this module implement?

File: `2-for_loop.sv`

Question 2. Briefly describe what the given module accomplishes. Then, describe how you could change the structure to reduce the latency of producing the output. (Hint: the current structure has O(n) complexity, but you can do better...)

File: `3-state_machine.sv`

Question 3.1. Why do we need a clock and reset in this module but not the previous modules?

Question 3.2. Draw the state transition diagram for this module on your worksheet. Include the enable signal as an input and write the prediction output at each state.

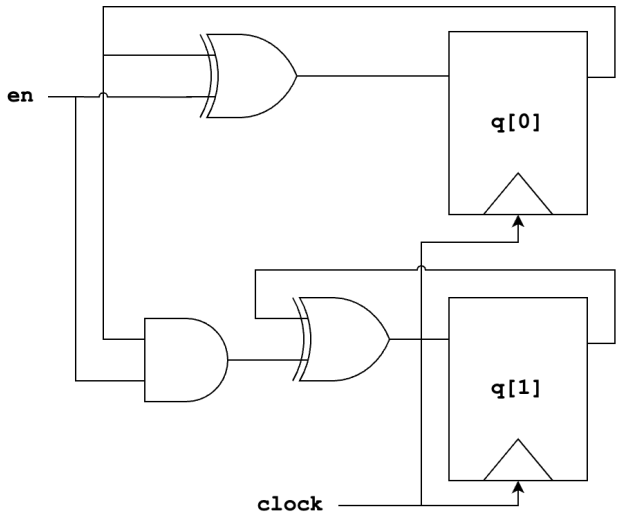
File: `4-register_file.sv`

Question 4.1. If this module were synthesized, how many 1-bit D flip-flops would be required to implement it?

Question 4.2. Briefly describe what internal forwarding is in the context of the module. Then, describe a way internal forwarding could be implemented in hardware?

File: `5-ron.sv`

Question 5. Write synthesizable SystemVerilog that implements the behavior of the hardware schematic below.



Question 5 Bonus. What is the function of this module? How can you simplify the Verilog to describe this functionality more clearly?

Finally, make sure that you can access the CAEN Linux environment by downloading Project 1 repository to CAEN and showing the instructor during your check-off.

3 Submission

In EECS 470, labs are submitted by in-person check-offs during lab or office hours. You will place yourself on the office hours help queue, and an instructor will come by to do the check-off. They will talk to you about the design and have you show them your worksheet and explain some of the concepts you learned.

The lab instructor will give you their signature with the date, or if virtual, a per-student per-lab unique check-off code. You will then submit the worksheet to Gradescope. We will only be grading these by looking for the instructor check-off, not the work.

Place yourself on the help queue during lab or office hours once you're confident you've completed the lab satisfactorily. Upload an image of your signed worksheet to Gradescope by the end of the day of next week's lab.