CS152A: Introduction to Digital Design Lab Spring 2020

INSTRUCTOR: Prof. Majid Sarrafzadeh (Office hours: by appointment only)

Lab Location: Online - Zoom Channel (More instructions will be announced on CCLE)

Lab Hours

- **Mohammad Kachuee:** MW [10:00-11:50AM PST]
- **Shayan Fazeli:** TR [10:00-11:50AM PST]
- **Rajas Mhaskar:** TR [12:00-1:50PM PST]
- **Logan Kuo:** MW [12:00-1:50PM PST]
- **Rohan Surve:** TR [12:00-1:50PM PST]
- **Ananya Ravikumar:** MW [2:00-3:50PM PST]

Instructions

In the spring of 2020, due to the major concerns regarding the coronavirus outbreak, CS152A is presented as a project-based and individually-graded online course. You will learn and get experienced with the design and implementation of hardware for Field Programmable Gate-Arrays (FPGAs). You will apply what you've learned in CS51A (combinatorial and sequential logic) and implement designs using modern design tools and Hardware Description Language (HDL).

You will have 5 projects in this course, in which you will prepare and design proper hardware using Verilog and in Xilinx ISE design (the student pack can be downloaded from the Xilinx website).

The difficulty of projects is adjusted in a way that you will find enough time to learn and review the necessary material and put them in action.

Code of Honor

Given that this course is online, cheating of any sort is considered a VERY serious misconduct. Also, sharing your codes (e.g. github) is strongly prohibited.

The Office of the Dean of Students has summarized University policy on academic integrity. Here are the relevant links:

- Student Guide to Academic Integrity: view
- Cheating: view
- Plagiarism: view
- UCLA Student Discipline FAQ:view

These summaries don't specifically address programming assignments in detail, so we state our policy here. Of course, you understand that your work on programming assignments **must be your own**. But we understand that high-level discussions about approaches to a problem have educational value and are acceptable. So where do we draw the line? We'll decide each case on its merits, but here are some categorizations:
Acceptable:

- Clarifying what an assignment is requiring
- Discussing algorithms for solving a problem, perhaps accompanied by pictures, without writing any code
- Helping someone find a minor problem with their code, provided that offering such assistance doesn't require examining more than a few lines of code
- Turning in someone's work without crediting the author of that work, if the source of that work is a CS152A instructor or TA

Unacceptable:

- Turning in any portion of someone's work without crediting the author of that work, if the source of that work is not a CS152A instructor or TA
- Using project solutions from earlier offerings of this or any other class
- Writing for another student a code fragment that solves any portion of a project assignment

- Receiving from another person (other than a CS152A instructor or TA) a code fragment that solves any portion of a programming assignment
- Helping the same person find problems with their code more than a few times for a particular assignment

You must abide both by this policy and the policies expressed in the UCLA Student Conduct Code (view). In accordance with University policy, we will submit cases of suspected cheating to the Dean.

Project Submission

For each project, the following should be submitted:

1. Project code: the Xilinx ISE project folder should be cleaned up (Project > Cleanup Project Files), zipped, and uploaded in the corresponding assignment page on the course website. The filename should be something like: Firstname_Lastname_Project1_code.zip
2. Video demos: You need to prepare a video (screen recording) using the software that you have chosen for the project, and describe the following on your code in details:
    a. Your thought process in designing the top modules
    b. The role of each module, individually
    c. How to integrate different modules
    d. What were the challenges you faced? How did you overcome them?
    e. What results do you see in your simulations, and what can we deduce based on them?

Sample filename is Firstname_Lastname_Project1_videos.zip

3. Lab Report: A **PDF** document that will be submitted as Firstname_Lastname_Project1_report.pdf

Lab Reports

Lab reports should contain the following sections. Percentages are subject to change and can vary based on projects’ needs and specifications.

1. **Introduction and requirement (~10%)**
   - Summarize background information about the lab and the detailed design requirements. It’s very important to make sure you are designing the right thing before starting.
2. **Design description (~15%).**
   - Document the design aspects including the basic description of the design, modular architecture, interactions among the modules, and interface of each major module. You should include schematics for the system architecture. You can also include figures for state machines and Verilog code when needed.
3. **Simulation documentation (~10%).**
   - Document all the simulation efforts (what requirements are tested and what the test cases are), document bugs found during simulation, and provide simulation waveforms. Note that synthesis and implementation report (output from Xilinx ISE) should also be included.
4. **Conclusion (~5%).**
   - Summary of the design. Difficulties you encountered, and how you dealt with them. General suggestions for improving the lab, if any.

Grading

You will be graded on a curve based on your performance in the course. The grading breakdown is as follows:

- Attendance Quizzes (10%)
- Project 1 (15%)
- Project 2 (15%)
- Project 3 (20%)
- Project 4 (20%)
- Project 5 (20%)

Each lab/project is graded based on the following components:

- Lab Report (PDF document) (40%)
- Video Demo (Videos in a Zip file) (60%)