CS M152A Introductory Digital Design Laboratory, Winter 2019

Administrative

Instructor: Miodrag Potkonjak (miodrag@cs.ucla.edu)
Instructor's Office Hours: By appointment

Lab / Office Hour Location: BH3424

Instructed Lab Hours:

Lab 1	MW 10:00am - 11:50am	Hongxiang Gu	hxgu@cs.ucla.edu
Lab 3	TR 10:00am - 11:50am	Jia Guo	jia@cs.ucla.edu
Lab 4	MW 12:00pm - 1:50pm	Mohammad Kachuee	mkachuee@cs.ucla.edu
Lab 5	TR 14:00pm - 3:50pm	Shayan Fazeli	shayan@cs.ucla.edu
Lab 6	TR 12:00pm - 1:50pm	Migyeong Gwak	mgwak@cs.ucla.edu

Open Lab/Office Hours (Tentative):

T 4:00pm - 6:00pm	Jia Guo
W 4:00pm - 6:00pm	Mohammad Kachuee
F 4:00pm - 6:00pm	Migyeong Gwak
Rest TBD	

Introduction

In this lab, you will get hands-on experience with design implementation on Field-Programmable-Gate-Arrays (FPGAs). You will apply what you've learned in CS51A (combinatorial and sequential logic) and implement some designs using modern design tools and Hardware Description Language (HDL).

We have designed several projects for you to implement using the Xilinx ISE design and implementation environment. These projects are intended as tutorials for you to learn how to put designs together, implement them in Verilog HDL, generate the programming file, as well as proving the design both in simulation and on real hardware.

The projects are not designed in such a way that you will spend a significant amount of time in the lab while keeping yourself awake with lots of caffeinated drinks. Instead, you should treat the lab sessions as a time to work with the hardware and interact with the TAs. **You will find that working on the designs on your own computer using the free**

version of ISE (Webpack), and sufficiently simulating the designs ahead of time before testing them on the FPGA boards makes one's life much easier in this course.

Team Work

You will be working in two (or three) person teams. The team is formed at the beginning of the quarter and is not subject to any change throughout the quarter. The teammates will also share the same score for the course.

Each team will be assigned a workbench with an FPGA board and a PC workstation. Keep in mind that your team will be sharing the same workbench with three other teams from the other sessions. Try to keep the workload balanced within the team.

Lab Conducts

1. Treat each other with the courtesy and decency that you would treat a colleague.
2. Respect the lab equipment that you are assigned to and be considerate to the other teams that share the same workstation with you.
3. Keep your assigned workstation clean. Keep the lab free of trash.
4. No food or drinks allowed in the lab.
5. Under no circumstances should you take the lab equipment outside of the lab.
6. If you must use the cell phone, kindly step outside of the lab.

Lab Access

The lab is only accessible to the students in a TA's presence. You are strongly encouraged to work outside of the lab to find all the bugs first before testing the design in a lab session.

In general, you should only attend your assigned instructed lab session. You may attend other instructed lab session. However, the students who are assigned to that session have priority over the hardware.

You are free to attend any open lab hours. Try to stick with the same workstation that you have always used unless the workstation is already occupied.

Computer Use

The PC workstation is only for coursework-related purposes, and they have restricted internet access. Do not leave your design files on the computer. Do not clutter the screen

with files. Use a USB thumb drive or a cloud-based file storage service to transfer files between your own computer and the workstation.

No grades will be given to a team if the design files are lost on the lab computers without a backup.

Lab time is limited. Use your best judgment on what's the best use of your allotted lab time.

Academic Integrity

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

- Student Guide to Academic Integrity: http://www.deanofstudents.ucla.edu/Student%20Guide%20to%20Academic%20Integrity.htm
- Cheating: http://www.deanofstudents.ucla.edu/Better%20of%20an%20Exam.pdf
- Plagiarism: http://www.deanofstudents.ucla.edu/Begin%20that%20Paper.pdf
- UCLA Student Discipline FAQ: http://www.deanofstudents.ucla.edu/faq2.htm

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 (http://www.deanofstudents.ucla.edu/SCC98.htm). In accordance with University policy, we will submit cases of suspected cheating to the Dean.

Project Submission

For each lab, 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
2. Lab Report (Electronic Version): the lab report should be uploaded in the corresponding assignment page
3. Lab Report (Paper Version): the paper version of the lab report should be printed out on both sides and handed in on the assigned date

Lab Reports

Lab reports should contain the following sections. The percentage in the brackets represents the percentage score for the section.

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.
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

Your will be graded on a curve based your performance in the four labs and attendance:

- Attendance (10%)
- Lab 0 (0%)
- Lab 1 (15%)
- Lab 2 (20%)
- Lab 3 (25%)
- Lab 4 (30%)

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

- Lab reports (40%). See the Lab Report sections.
-
- Demo (60%). The followings are the rules for demo points. Note that before the deadline, only complete and correct demos are accepted. If unfortunately, you can't finish the project on time, you have to demo whatever accomplished by the end of the next session.

  Demo early: 65%

  Demo on time: 60%

  One-session late demo: 45% x (correctness), where correctness refers to the percent of the demo you are able to finish by the late session