EC413: Computer Organization – Fall, 2021

Basics

Instructor: Prof. Martin Herbordt – Office: PHO 333 -- Office Hours: T 3-5, W 3-5 and by appointment
Office Phone: x3-9850 -- Email: herbordt@bu.edu

Course web page: http://learn.bu.edu

TAs : Chunshu Wu, Ben Leone, Ian Chadwick, Kellen Jay, Sara Fagin, TBD
Email: {happycwu,bleone90,ianjc,kbjay,sfagin}@bu.edu
TF Lab Hours in PHO305: M: 7:30-8 / T: 7-9 / W: 7-9 / Th: 7-10 / Fr: 3:00-6:00
Lab: **PHO305**, PHO307

Discussion Sections: meet in PHO 307 – M1:25-2:15, M4:40-5:30, M6:30-7:20

Mission Statements: "From gates to programs'' and "How computers really work"

Course Description: *(From programs to gates and back again)*
Introduction to the fundamentals and design of computer systems. The starting points are your basic knowledge of logic design and high-level language programming. The ending points include your ability (i) to create a working computer using standard HDL design practices, (ii) to program that computer in assembly language, and (iii) to be able to understand and evaluate computer system design options at multiple levels. A fundamental outcome is to understand, through the logic level, how programs run on computers and approaches to writing code for performance. Topics include computer instruction sets, assembly language programming, logic and algorithmic design of arithmetic operations, design of sequential logic with registers and buses, CPU design (data path, control, integrating datapath and control, pipelining), performance evaluation, memory devices, memory systems (including caching and virtual memory), and I/O. In parallel there is a lab where the focus is on in-depth understanding of selected comp org topics including HDL design using Verilog and standard system design methods.

Course Style: EC413 has both theoretical and practical aspects.

Prerequisites: EC311, Introduction to Logic Design
Familiarity with Xilinx CAD tools
High-Level Language Programming, preferably in EC327. Literacy with **C** is helpful.

Texts: Patterson & Hennessy, *Computer Organization & Design: The Hardware/Software Interface*, 6th Edition (NEW), Morgan Kaufmann, 2020 – This is the MIPS edition.
Readings from: Mano & Kime, *Logic and Computer Design Fundamentals*, Prentice Hall (to be posted)
Various articles and tutorials (to be posted)

Course Mechanics

- **Exams:** There will be two midterms and a final exam. Exams are closed book/notes with a single standard-sized sheet of notes allowed (hand written, front and back).
- **Quizzes:** Most semesters there are ~4-6 short quizzes. Their purpose is to ensure that everybody keeps up.
- **Attendance:** *speaking of attendance ….* Attendance is essential – much of what I cover in this course will be found nowhere else.
- **Readings:** … are also essential. There is no way we can (or would want to) go through all of the essential details of any topic during class. Good news - the textbook is a standard so you can expect your future colleagues to have learned Computer Organization in similar way.

- **Homework:** There are ~8 homework assignments. Unless stated otherwise, homework is due by *the beginning of class* on the date specified using the course web site. Homework will not be accepted after that. Homework must be readable, so typed is preferred. But scanning or photo (with conversion to pdf) is OK.
  Academic honesty wrt HWs: You are encouraged to work together to learn the material and to discuss approaches to solving homework problems. However, *you must come up with and write up the solutions on your own. See academic honesty policy.*

Labs

- **Overall:** There will be ~9 labs. Much more about these later! Some of the labs (3-6) have a substantial workload, requiring perhaps 10 hours or more for many students. (You may want to plan accordingly.) It is therefore essential that you get an early start. It is also essential that you understand thoroughly the underlying material before you start to code. For most labs there is a "prelab" during discussion section; is generally a to-be-turned-in-and-graded assignment that will help you get started.
  Academic honesty wrt labs: The rules for collaboration are the same as for homework: *these are all to be done with your group alone.*
- **Lab Practicalities:** The labs are open-ended in the sense that there is not a specified time during which the labs must be done. Rather, labs are assigned about a week before they are due and it is your responsibility to get them done on time, generally late Friday afternoons. The TFs will be in the lab about 12-15 hours during weeks when labs are due. Please note that the amount of time that it takes any particular student to complete any particular lab takes can vary by an order of magnitude (from a few hours to much more than that). The greatest determinants of duration are how well prepared you are before you start coding and how well you understand the tools. Much more about this later, especially during discussion section.
- **Lab grading mechanism:** A large part of each lab is the demo. Because there are many more students that TAs, you are urged to get the labs done early. For labs 3 on, 5% bonus for finishing by Thursday, 10% bonus for finishing Wednesday or earlier. There is a 10% penalty for being late one business day (usually the following Monday) and a severe penalty for being later than that.
- **Discussion Sections**: The discussion sections are run by the TFs and serve two purposes: a pre-lab and/or an extension of the lectures. Attendance is not required, but is *very strongly* recommended, especially to do the pre-labs and for exam reviews (this is most of the discussion sections).

Grading

Grades are based on a weighted average of HW, quiz, exam, and lab scores. *Each category is curved independently.* Since we update all evaluation mechanisms every semester, I never know in advance exactly how the scores in each category will relate to a letter grade, but here is the general idea:

HWs and Labs. The expectation is that all (passing) students will make a significant effort on all HW assignments, and will have working demos of all labs and the project. There is also some opportunity for extra credit on the labs. Therefore, the cut-offs between A/B, B/C, etc. are generally *higher* than the traditional 90, 80, etc. For example, for an A, a student should get at least 90s on HWs and 95s on the labs.

Exams and Quizzes. On the other hand, exams and quizzes are often hard and unpredictable, so the reverse is true here. Last year, the A-/B+ cut-off for the mid-terms was an 84 and for the final a 75.

Weighting the categories.

- HWs, Quizzes = ~1pt each
- Labs = ~4pts each
- Exams = 18pts, 18pts, 24pts

Please note: *You are not competing with each other on grades*. There have been semesters when half the class has gotten A- or better. FYI, the median is usually around a B, but this is purely statistical. No student is "average"!

Administration

- **Office Hours:** My office hours are listed above and on the web site. Normally the best time to catch me (otherwise) is between class halves or after class; the worst time is right before class when I am trying to get set up. On occasion I will be traveling and so unable to keep office hours. I will announce this well in advance.
- **Email:** You are required to periodically check your email since that is the way some notifications (assignments and coreecttions) will be distributed. Questions via email are always good. If the question/answer has general interest, I will broadcast it to the class (leaving the questioner anonymous); if the solution is very involved, we may need to go over it in person.
- **Course Web Site:** I use the course web site to post class notes, lab and homework assignments, homework solution sketches, additional readings, and other course information.
- **Incompletes:** Incompletes are granted only in accordance with university policy, which (broadly) requires a major personal (non-academic) crisis near the end of the semester.
- **If you have an academic crisis of any kind:** These happen. Please note that they are much easier to mitigate or resolve if you talk to me early – don't wait.
- **Academic Conduct:** https://www.bu.edu/academics/policies/academic-conduct-code - Please read the university academic honesty policy. If something is not clear, then ask. In particular, plagiarism is really serious! See the notes wrt HW & Labs.
- **Course Notes:** Because of the nature of the material, Computer Organization is almost always taught with slides, although I try to mix things up by doing examples on the slides themselves. A version of the slides will always available well before class, although the final version might not be available until just before class.
- **Distractions:** What was that? Except for annotating notes, please keep all electronic devices off during class. This will help me, you, and the students sitting around you.
- **Punctuality:** I start class promptly so please be here on time (see "distractions").
- **Instructer Errosr:** Please don't be shy! If you see me make a mistake, please let me know right away. If you are not sure, that's even better – it might give me a chance to clarify something.
- **Your success** is something I really care about! All job markets are incredibly competitive, especially for the really excellent jobs, but students who do well in this class have a very high probability of working at the best companies (and becoming researchers, doctors, executives, etc.).

For all courses at BU – university policies

- **COVID 19 & BU Community Health Expectations:** Masks are required and face coverings must be worn over the mouth and nose at all times when in public spaces on campus, including classrooms. Students should be prepared to show proof that they are compliant with health attestations and testing in order to attend class. All students are expected to follow all university guidelines with respect to daily symptom checks, testing, social distancing, and mask wearing when they leave their dorm or home. For a detailed description of official BU policies regarding COVID, please visit: http://www.bu.edu/dos/policies/lifebook/covid-19-policies-for-students/
- **Inclusion:** I consider this classroom to be a place where you will be treated with respect, and I welcome individuals of all ages, backgrounds, beliefs, ethnicities, genders, gender identities, gender expressions, national origins, religious affiliations, sexual orientations, ability – and other visible and nonvisible differences. All members of this class are expected to contribute to a respectful, welcoming, and inclusive environment for every other member of the class.
- **Accommodations for Students with Documented Disabilities:** If you are a student with a disability or believe you might have a disability that requires accommodations, requests for accommodations must be made in a timely fashion to Disability & Access Services, 25 Buick St, Suite 300, Boston, MA 02215; 617-353-3658 (Voice/TTY). Students seeking academic accommodations must submit appropriate medical documentation and comply with the established policies and procedures http://www.bu.edu/disability/accommodations/

Keys to success in this (and most other) course(s)

- **Attendance.** You must attend class. Much of the material (and perspective) will be found nowhere else.
- **Do the readings.** Other material is found only in the readings.
- **Read actively**. Work out the examples as you read. If you are not positive that you understand something completely, try inventing and solving your own problems.
- **Take notes.** In particular, annotate the lecture notes during class.
- **Reread the notes.** Perhaps surprisingly (or not), retention is better from reading than from listening, and even better for writing, and still better when you explain it to someone else.
- **Participate.** Ask questions; talk with your fellow students. Be active.
- **Keep up.** Before each class, (at least) study the notes from the previous class. This course comprises a number of disparate topics, each with its own terminology and axioms.
- **Allocate enough time!** Much of the material is time-consuming to master. There is often a big difference between thinking you understanding a subject and *really* understanding it. In computer organization and architecture topics interact in subtle ways; mastery of any one requires a good understanding of all of them.
- **How do you know that you know the material?** A good metric is whether you would feel comfortable standing in front of a class explaining it. Another is whether you think that you could explain it during a job interview!