Computer Architecture I

Teaching Staff

Instructor: David Black-Schaffer — Office Hours: See website
TA: Muneeb Khan — Office Hours: See website

If you cannot make these times please send us a message and we will arrange another time.

Email

Please use the course website on Canvas for all electronic contact with the teaching staff. You can find a link to the website from either studentportalen or the IT department page.

- If you have general questions please post them under the Discussions section of the website.
- If you have private questions for the teaching staff, go to your inbox on the class website and send a message to the instructor or TAs.
- Questions about course administration should be sent to the instructor.
- Lab questions should be sent to the TAs.

We will try to respond as soon as possible, but no later than our next scheduled office hours.

Course Contents

This is an introductory course to computer architecture and will cover MIPS assembly language, the MIPS processor architecture, the memory hierarchy, input/output, performance analysis, the basic design and implementation of a the data path, and the basics of parallelism in modern multicore processors. The exam for the course will cover the reading material and the lecture material. *Note that neither the reading nor the lectures contain all the material for the exam: you will need to do both.* The expected coursework consists of reading the book, viewing online lectures and completing individual assessment quizzes, participating in in-class practice problems, completing labs in groups of two, and a written final exam.

Prerequisites

Formally: Programmeringsteknik II. Students are expected to have a background in imperative programming, e.g., C/C++/Java. You should be familiar with loops, if/then/else, arrays, and walking through strings.

Grading

This course is graded as U, 3, 4, 5, with the final grade weighted 40% for participation and 60% for the final. You must pass both portions to pass the course. The participation portion of the grade is intended to encourage you to work on the material before the exam and is not optional. To receive credit for the participation portion you must complete the lectures on time, participate in the practice sessions, and complete all labs. The participation grade is based on the average grade of your labs, but will be U if you do not complete the lectures or participate in the practice sessions. See below for the details.

To provide students with flexibility, each student will receive 8 late days to use during the course. Students may use these as they see fit for a total of 8 days of extension on their lab due dates and lecture watching. Doing so requires notifying the teaching assistant when you wish to use a late day. We will keep track of how many late days you have remaining. Note that using a late day will not increase the amount of time you have until the next assignment.

Why so much work?

This course has required readings, lectures, quizzes, practice sessions, and labs. That may seem like a lot of work, but the motivation for it is simple: to help you learn. Our experience has been that

non-required reading is never done, and that without the practice sessions, students do not get enough hands-on exposure to the material to do well on the exam. Our sincere hope is that by making these required, we will improve how much you learn in the class.

Reading

"Computer Organization & Design: The Hardware/Software Interface" by Patterson and Hennesy. 4th Edition, Morgan Kaufman 2007. *The third edition is not acceptable*. This book is very clear and easy to read, if a bit long-winded. You will need to have access to the CD content of the book for appendices C and D. Reading should be done before you come to the practice session. Remember that the exam will include material present in the assigned reading that is not in the lectures!

Lecture	**Reading** *before coming to class*
1. Introduction	1.1-1.9 (may be done after class, obviously)
2. Instruction Sets 1	2.1-2.7
3. Instruction Sets 2	2.8-2.10, 2.13, 2.16, 2.18-2.19
4. Arithmetic	3.1-3.5, 3.8-3.9
5. Logic	C.1-C.3, C.5-C.8, C.10, D.1-D.2
6. Processor 1: Control and Datapath	4.1-4.4
7. Processor 2: Pipelining	4.5-4.6
8. Processor 3: Hazards and Forwarding	4.7
9. Processor 4: Branching	4.8-4.9, 4.13-4.14
10. I/O and Memory	6.1-6.7, 6.10, 6.12-6.13, C.9 (skip error correction in C.9)
11. Caches	5.1-5.3
12. Virtual Memory	5.4 (skip handling of faults in 5.4), 5.5, 5.7, 5.11-5.12
13. Parallelism	2.11, 3.6, 4.10, (4.11 optional), 5.8, 6.9, 7.1-7.5
14. Roofline	7.10-7.13
15. Review	

Lectures

All lectures will be provided online through the course website. Lectures will contain short, non-graded, self-assessment quizzes designed to help you understand how well you understood the material and tell us what we should focus on in the practice sessions. These lectures are not optional and you are expected to have viewed them and answered the quizzes before the scheduled practice session. We will try to post lectures at least one week before their due dates so you have flexibility as to when to watch them. Your performance on the quizzes will not affect your grade, so you may re-take any quizzes you get wrong if it helps you learn. To receive credit for the participation portion of the course you must watch all lectures and take all quizzes before their practice session dates. (We will track this on the website.)

Practice Sessions (Local)

Practice sessions will be held once or twice a week as standard 2x45 minute sessions with the instructor and TAs. During these sessions we will review the material from the assigned reading and lectures and focus on areas where students had difficulty as seen in the quiz results. Most of the time will be devoted to interactive problem solving in pairs. Lack of participation in the practice sessions will result in a U for the participation portion of the course.

Practice Sessions (Distance)

Practice sessions will be provided in the form of assigned problems based on the results of the lecture quizzes. The problems will be posted on the course website and while students are required to submit solutions, they will not be graded. Students are expected to work either alone or in pairs to solve these problems. The TAs will then provide feedback on the solutions and help with the

problems via online discussions. Lack of participation in the practice sessions will result in a U for the participation portion of the course.

Labs

The course will have five labs. Students should work in pairs to complete the labs and one lab assignment should be submitted for each pair. If you cannot find partner within the first week please post on the class website and we will find one for you. The TAs will provide online tutorials for the labs to get you started with the software, answer questions on the course website, and will be present in the lab for several hours every week to help you with any problems you may have.

	Topic	TA Tutorial
1	Introduction to MIPS assembly	Introduction to SPIM simulator
2	Adder and ALU logic	Introduction to LogicSIM
3	Datapath logic	
4	MIPS I/O	
5	Caching effects	Introduction to cache simulator

All lab assignments must be submitted to receive any credit for the labs. Students are required to submit their lab by midnight on the specified date. Graded labs will be returned within a week. You may request a re-grade of the lab within one week of receiving the grade. No re-grades will be offered after that time. Note that re-grades may result in lowered grades. Missed labs may only be turned in *after the course is finished*, but any late missed labs will limit the maximum grade lab portion of the course to 3. If you wish to re-submit a new version of a lab for re-grading you may do so after the course is finished, but you will not be able to raise your grade for the labs above 3 as with missed labs.

Submitted labs will be graded as follows:

1 point	Poor quality and/or largely non-functional
2 points	Poor quality and not completely functional
3 points	Acceptable quality but not completely functional
4 points	Acceptable quality and completely functional
5 points	Good quality and completely functional

Quality includes the clarity of the solution, e.g., efficiency of implementation, clarity of comments, variable names, layout, etc. The grade for the lab portion of the course will be determined by the average score across all labs. Averages under 3 will receive a U for the lab portion of the course. Note that it is essential that you turn in your lab by the due date to get any credit for the lab portion of the course, even if it is not fully functional.

You can ask questions about the lab solutions during the office hours after the lab has been submitted.

Grading Summary

60%: written final exam.

40%: participation, which is determined as follows:

- Average of lab grades. (An average less than 3.0 will result in a U.)
- U if lectures and quizzes are not completed on time.*
- U if no/insufficient participation in practice sessions.
- U if not all labs turned in on time.*

*You may use your late days to extend these deadlines, but not to skip the assignments.

Honor Code

The standard honor code for the department applies. You are expected to have read and understood this document. (http://www.it.uu.se/edu/cheating.pdf). The key definition of cheating is: “Students who ... by forbidden aids or in any other way attempt to mislead/deceive during a test or other student assignment which is to be assessed.”

In the context of this course, the following general rules should be respected. This is not a complete list. If you have questions please contact us for clarification.

- Students work together in pairs on lab assignments and are expected to contribute equally to each lab. Each lab group is expected to submit their own work. If this is not working as expected please contact the teaching staff.
- Lab assignments must be completed without examining existing solutions. This does not prohibit you from obtaining assistance, but you may not consult solutions to the same problem. E.g., in a lab about building ALUs, it is okay to search for “how does a MUX work” but not okay to search for “how do I use a MUX in an ALU”. In a lab about building a MUX, the first search would not be acceptable.

Since all of labs are submitted electronically, we may use automated tools to detect plagiarism.

Any instance of cheating will be pursued to the fullest extent of the department and university policies. Don’t cheat. If you are in doubt, ask.