Digital Engineering and Computer Architecture (E85)
Fall 2013
Syllabus

Teaching Staff

Professor:	Josef Spjut	josef_spjut@hmc.edu	
	Parsons 2362	Office Hours by Appointment	
Lab Tutors:	Vanessa Ronan	Dong-hyeon Park	Obosa Obazuaye
Lab Graders:	Matthew Tambara		
Homework Graders:	Ivan Wong	Obosa Obazuaye	

Schedule:

Lecture:	SHAN 3460	MW 1:15-2:30
Lab Tutor Hours:	E85 Lab	W 8-10, Th 8-10,10-12
TBP HW Tutor Hours:	Platt	Sat. 7-9, Sun 6-9

Textbook:

Harris & Harris, *Digital Design and Computer Architecture*, 2nd Ed., Morgan Kaufmann, 2012.

Electronic Communication

Class Web Page:	http://www3.hmc.edu/~jspjut/class/e85
Class Email List:	eng-85-l@hmc.edu

Be sure you are on the class mailing list. You should have received email before the beginning of classes. If you did not receive mail, add yourself to the list or risk missing important late-breaking announcements. To subscribe, send email to listkeeper@hmc.edu with one line in the body:

subscribe eng-85-l

Alternatively, if you have an electronic version of the syllabus, you can click this paragraph to have your mail client autocomplete the email for you.

You also will need a Harvey Mudd College computer to complete your labs. If you are not a HMC student, email me your full name and school affiliation and I will request an account for you. You will complete most of your labs in the E85 lab, Parsons B183. See Professor Spjut for the door code. Hardware portions of the labs must be completed in the E85 lab, but other computers including the ones in the ECF may be used for some parts of the labs. If you have issues with the software or computer systems in the lab, contact the system administrator, Willie Drake. He can often be found in his office inside the ECF, and after-hours contact information is typically posted on his door.

Course Objectives:

- To apply the principles of abstraction, modularity, hierarchy, and regularity in digital design
- To develop debugging skills by designing, building, and testing digital circuits using commercially available CAD (computer aided design) tools
- To design, build, and test your own microprocessor
- To understand what's under the hood of a computer

Grading:
Labs: 30%
Problem Sets: 25%
Midterm 1: 20%
Final: 25%

There will be weekly labs and problem sets, a and two midterm exams. The first midterm will be taken in-class and the second will be a take-home exam. The first midterm is worth 20% of your grade and is closed-book, closed-notes. The second midterm is open-book, open-notes (your own notes, including course handouts) and is comprehensive, but biased towards the second half of the class. It is worth 25% of your grade.

The only way to really master the material in this class is to design a microprocessor. The labs in this class build upon each other and culminate in designing your own 32-bit MIPS microprocessor in Labs 10 and 11. You **must** complete these labs and demonstrate a working microprocessor to pass this class.

No late homework or labs will be accepted. However, you are allowed to drop one lab and one homework score. Homework is due **at the beginning** of class. You will lose 50% of the points on late homework turned in by the end of class. Labs are due electronically on Sakai by 3 pm on the due date. Even if you do not complete your microprocessor (labs 10 & 11) on time, you must still submit them before finals week to pass the class.

Your lowest lab and homework score will be dropped before the average is calculated. You are welcome to discuss labs and problem sets with other students or with the professor or lab assistants or tutors after you have made an effort by yourself. However, you must turn in your own work, not work identical to that of another person. Be sure to credit at the top of your assignment anyone with whom you discussed ideas. **It is an honor code violation to simply copy someone else's work.** Reading assignments are intended to be completed before class.

A suggestion for text preparation comes from reddit user **onanym**, "I usually write a cheat sheet, then memorize it so it's all in my head. That way, they can't possibly prove I'm cheating.
Genius." As a response, reddit user **leontes** added, "I used to [do] that, but I found I couldn't remember the details perfectly. So instead... I integrated the information on the cheat sheet into my prior existing knowledge, Coding it perfectly within my preexisting framework.
Not only could they not know I was regurgitating the cheat sheet, but since I had remembered it in terms of [stuff] I already knew, I was able to synthesize complex consequences from the source material so it seemed like original, derivated understanding! They never caught on!" From `http://www.reddit.com/r/AskReddit/comments/11nbmz/im_a_political_science_professor_i_gave_my/c6nzn34`

Tentative Schedule: The schedule below is a tentative plan that may change during the semester. The deadlines, however, are fixed unless otherwise notified; do not assume that they will change just because the lecture schedule changes. Any changes to deadlines will be announced in class and sent to the class mailing list.

Monday	Wednesday	Friday
Sep 2nd Labor Day	4th **1** Binary, logic gates, logic levels 1.1-1.5,A.1-A.2,A.5-A.7	6th
9th **2** transistors; truth tables, Boolean expressions; Boolean algebra 1.6, 1.7,1.9, 2.1-2.3	11th **3** K-maps; Xs and Zs; multiplexers and decoders; priority circuit; timing; hazards; 2.4-2.10	13th **Lab1 Due**
16th **4** **PS1 Due** Last Day to Add Classes sequential circuits: SR latches, D latches, flip-flops, clocking 3.1-3.3.3	18th **5** finite state machines (FSMs) 3.4	20th **Lab2 Due**
23rd **6** **PS2 Due** dynamic discipline, metastability, parallelism 3.5.1-3.5.6,3.6, 3.7	25th **7** System Verilog 1 4.1-4.4	27th **Lab3 Due**
30th **8** **PS3 Due** System Verilog 2 4.5-4.9	Oct 2nd **9** arithmetic: adders, subtractors, comparators, ALUs 5.1-5.2.8	4th **Lab4 Due**
7th **10** **PS4 Due** number systems: fixed & floating 5.3	9th **11** sequential building blocks: counter, shift register, memory arrays: RAMs, ROMs, logic arrays: PLAs, FPGAs 5.4, 5.5, 5.6, 5.7	11th **Lab5 Due**
14th **12** **PS5 Due** Problems/Review	16th **13** **Midterm** In Class	18th
21st Fall Break	23rd **14** C-programming 1 c.1-c.6	25th

Monday	Wednesday	Friday
28th **15** C-programming 2 c.7-c.11	30th **16** MIPs instruction set and registers 6.1-6.3	Nov 1st **Lab6 Due**
4th **17** **PS6 Due** branches & procedure calls, addressing modes 6.4-6.5	6th **18** linking & launching applications 6.6-6.7.1	8th **Lab7 Due**
11th **19** **PS7 Due** single-cycle processor data path and control 7.1-7.3.4	13th **20** multi-cycle processor 7.4.4	15th **Lab8 Due**
18th **21** **PS8 Due** pipelining, hazards and stalls 7.5.1-7.5.5	20th **22** exceptions and advanced microarchitecture 7.7-7.8	22nd **Lab9 Due**
25th **23** **PS9 Due** memory system, performance 8.1, 8.2	27th **24** **Lab10 Due** caches and virtual memory 8.3-8.4	29th Thanksgiving
Dec 2nd **25** **PS10 Due** Review	4th **26** memory-mapped, embedded and PC I/O 8.5-8.7	6th **Lab11 Due**
9th **27** **Final Distributed**	11th **28** MicroPs Project Demo Day	13th Last Day of Classes
16th **Finals Due** Finals Week	18th Finals Week	20th Finals Week