CSEE W3827 - Fundamentals of Computer Systems
Course Information

Course Call #: 78400
Professor Dan Rubenstein Spring 2013

Course Resources

Contact Information				
Staff	**Office**	**Phone**	**e-mail**	**Office Hours**
Dan Rubenstein (Instructor)	CEPSR 816	(212) 939-7048	danr@cs.columbia.edu	W 10-11am, F 12-1pm or by appt. (backup Th 1-2pm)
David Calhoun (TA)	CEPSR 804	N/A	dmc2202@columbia.edu	Tu 10am-12pm
Andrew Mercer-Taylor (TA)	CS TA Room	N/A	ajm2209@columbia.edu	Tu 4:10-6:10pm
Thomas Rantasa (TA)	CS TA Room	N/A	tr2286@columbia.edu	F 11am-1pm
Lisa Wu (TA)	CSB 468	N/A	lisa@cs.columbia.edu	M 9-11am
Contact Prof and TAs: 3827TA@lists.cs.columbia.edu				

Course URL: http://www.cs.columbia.edu/∼danr/3827

Course meeting time / location: 1:10 pm - 2:25 pm on M,W in NWC 501

Pre-Requisites

An introductory programming course (e.g., COMS 1004 or 1007)

Description

This course explains, from a logic perspective, how computers work, i.e., how 0's and 1's are manipulated to do all the advanced calculations, computations, and services that computers can perform. The first major topic is digital logic, which concerns the design of circuits to implement logic functions using standard components such as AND-gates, OR-gates, and inverters. The circuits might be used to control the flow of data within a computer, or the processing of the data (e.g., arithmetic operations), or to control the overall action of a computer. We will cover how to specify logic functions precisely, to manipulate formal expressions, and to implement them efficiently. We will then cover the design of basic building blocks, including the control, of modern digital computers. Both combinational and sequential circuits will be covered.

The second part of the course involves the structure and software interface of digital computers. Focusing our attention on modern RISC architecture. We will discuss the functional blocks such as the arithmetic unit, register files, and memory. Single-cycle and multiple-cycle implementations will be presented, followed by the concept of pipelining. We will cover the basics of caches and virtual memory. Machine and assembly language programming is a feature of the course. Main memory systems, currently DRAM, will be discussed as well as the operation of magnetic disk drives. Some aspects of I/O will also be introduced.

Grading

Your grade consists of:

40% **Homework:** Homework is **due by the end of class at 2:25** on the date that it is due, unless otherwise specified. **The TAs** will accept assignments until the end of the day, provided you can find them. **Professor Rubenstein will not accept assignments once he has handed them to the TAs for grading.** E-mailed/faxed homework and late assignments will not be accepted unless approved in advance. Approval will only be given under extreme circumstances. You are expected to produce your work in a timely manner. You may discuss and work on questions with other students in the class. However, you should write your solutions on your own, i.e., not copy someone else's solution.

35% **Mid-term:** October 21, in-class, open book, open note, no calculators.

45% **Final:** Probably 12/21 at 9am (determined by the registrar), open book, open note, no calculators.

Note that the total adds to 120%. The lowest score's contribution is reduced by 20%. For instance, if your homework average is 90, your midterm score is a 35, and your final exam score is 70, the midterm's contribution is reduced to 15%, making your average score equal to $0.4 \times 90 + 0.15 \times 35 + 0.45 \times 70 = 72.75$.

A note on effort: The final grade is based on my interpretation of how well you know the material. If all I know about you is your scores on homeworks, midterm and final, then those will directly determine your grade. If, however, you speak up in class and/or attend office hours, this gives me another way to evaluate how much you know and can only help your grade.

Reading / Texts

There is no required text. Recommended texts are:

- *Digital Design and Computer Architecture (2nd edition)*, D. Harris and S. Harris, Morgan Kaufmann. IBSN: 978-0-12-394424-5
- *Logic and Computer Design Fundamentals (4th edition)*, M. Morris Mano and Charles R. Kime, Prentice Hall. ISBN: 978-0-13-198926-9.
- *Computer Organization and Design, The Hardware/Software Interface, 4th edition*, David A. Patterson and John L. Hennessy, Morgan Kaufmann, ISBN 978-0-12-374750-1.

Harris&Harris covers all the material in the course, but not as closely as the other two books. Morris&Kime cover the first half, Patterson&Hennessy the second.

Computing Accounts

The course does not require computing facilities or computing accounts.

Collaboration and Cheating

Collaboration on homework is acceptable, but each student should turn in their own copy. The downside of this collaborative policy is that doing well on the homework doesn't really demonstrate that *you* know the material. To demonstrate this, you need to do well on the midterm and final. The best, time-tested way to do well on the exams: **do your own homework!**

If you cheat, there is a good chance you will do better than you deserve. But if you get caught, there is a good chance the punishment will be more severe than the potential gain from cheating. Please don't cheat.

On contacting Prof. Rubenstein

There are over 130 students in the class at the moment, so I won't be able to respond immediately to each e-mail.

- It doesn't matter if you email me at danr@cs.columbia.edu, danr@ee.columbia.edu, dsr100@columbia.edu or danielr@columbia.edu, they all go to the same place. If the question is of a technical nature, you are better off emailing 3827TA@lists.cs.columbia.edu. This e-mail goes to both myself and the TAs, so the first to get the question can answer.
- Put **3827** in the subject header.
- E-mail is the best way to get in touch with me (unless I'm sitting in my office and you want to see if I'm there). Then you can try my office phone.
- Do a bit of legwork before asking your question. If I've already answered it elsewhere, I might not dig up the answer again.
- I batch e-mails with questions and respond in unison, often via a broadcast FAQ. I do this roughly once or twice a week. This means you may have to wait several days for an answer to your question, so please don't wait until the last minute to start the homework.

Syllabus and Schedule

Note schedule subject to change...

Date	#	Topics/chapters covered	Reading	Assigned	Due
1/22	1	Intro; Overview of Computer Architecture; Definitions (bit,byte,word)	M&K Ch 1		
1/27	2	Binary number representations: 1's complement; 2's complement, adding and subtracting and overflow; floating point representations: overflow and underflow	M&K 4.3-4.4, 10.7, P&H 3.5 skip FP in MIPS	HW #1	
1/29	3	Logic gates; XOR; Boolean Algebra; NAND and NOR gates; Taking complements; DeMorgan's Theorem; Duals	M&K 2.1-2.2, 2.8 2.9		
2/3	4	Standard Forms: minterms, maxterms, sum-of-products, product-of-sums	M&K 2.3	HW #2	HW #1
2/5	5	K-maps: simplification with implicants, Don't-care conditions	M&K 2.4-2.5		
2/10	6	***** Catchup *****		HW #3	HW #2
2/12	7	Combinatorial Circuit Design: Multi-bit output functions; standard combinatorial circuits (enabler decoder, encoder, priority encoder, mux	M&K 3.1, 3.3, 3, 3.6-3.9		
2/17	8	Arithmetic funcs: Adder (half, full, ripple-carry, adder-subtractor); Contraction; Shifter	M&K 4.1-4.2, 4.5 9.4	HW #4	HW #3
2/19	9	Sequential Circuitry: Latch, Flip-Flops, timing issues	M&K 5.1-5.3, 5.6		
2/24	10	Sequential Circuit Analysis & Design: State machines	M&K 5.4-5.5	HW #5	HW #4
2/25	11	PLAs; ROM; Register Design: Load and Transfer	M&K 6.8, 7.1-7.3		
3/3	12	Register Design cont'd: MicroOps and Counters, mux and serial transfer	M&K 7.5-7.6, 7.8-7.9		HW #5
3/5	13	Memory Design	M&K 8.1-8.7		
3/10	14	***** Catchup and/or Midterm review *****			
3/12	16	Processor Design: Datapath, ALU	M&K 9.1-9.5	HW #6	
3/17	–	**SPRING BREAK - NO CLASS!**			
3/19	–	**SPRING BREAK - NO CLASS!**			
3/24	17	Control Word; Simple Arch; Instruction Decoder	M&K 9.6-9.8	HW #7	HW #6
3/26	15	**MIDTERM** (in class)			
3/31	18	Instruction Types and Formats	P&H 2.1-2.6		
4/2	19	Branches, stacks, heaps, immediate addressing	P&H 2.7-2.8, 2.10	HW #8	HW #7
4/7	20	Single Cycle Datapath	P&H 4.1-4.4		
4/9	21	Single Cycle Datapath cont'd		HW #9	HW #8
4/14	22	Pipelining	P&H 4.5-4.6		
4/16	23	Hazards	P&H 4.7-4.8		
4/21	24	Cache & Cache Replacement Policies	P&H 5.1-5.2		
4/23	25	***** Catchup and/or review *****			HW #9
4/28	26	***** Catchup and/or review *****			
4/30	26	***** Catchup and/or review *****			
5/5	26	***** Catchup and/or review *****			
Tentative 5/12		**FINAL EXAM, Time 1:10pm-4:10pm**	Location TBD		