ECE 199JL: Introduction to Computer Engineering **Fall 2012**

Douglas L. Jones and Steven S. Lumetta, Instructors
Geoffrey Herman, Discussion Leader
Sai Ma and Seyed Ahmadyan, TAs

Course Syllabus

Lecture	Date	Topics	Book Sections
1	8/27	Overview and orientation; computer systems organized as a systematic set of transformations	Ch. 1
2	8/29	Representation using bits: unsigned and signed integers	§ 2.1−2.4
3	8/31	Operations on bits: arithmetic	§ 2.5
4	9/5	Operations on bits: logical	§ 2.6
5	9/7	Other representations: ASCII, floating point; hexadecimal notation	§ 2.7
6	9/10	C programming: basic data types, program structure	Ch. 11
7	9/12	C programming: basic I/O, operators, control structures	§ 11.5, 12.1−12.3, 13.1−13.3
8	9/14	C programming: examples, problem solving	§ 12.4, 13.4
	9/17	**OPTIONAL REVIEW SESSION FOR MIDTERM #1**	
9	9/19	Logic gates	§ 3.1, 3.2
10	9/21	Boolean algebra: truth tables, Boolean expressions	
11	9/24	Boolean algebra: logical completeness and two-level design	
12	9/26	Boolean algebra: canonical forms, minimization	
13	9/28	Boolean algebra properties	
14	10/1	Gates and combinational logic design examples	
15	10/3	Combinational logic design examples	
16	10/5	Sequential logic components: latches, flip-flops	§ 3.4
17	10/8	Sequential logic components: registers	
18	10/10	Finite state machines: abstraction	
19	10/12	Clock synchronous design	
	10/15	**OPTIONAL REVIEW SESSION FOR MIDTERM #2**	
20	10/17	FSM example mapping to digital logic	§ 3.6
21	10/19	FSM example mapping to digital logic	
22	10/22	Combinational logic structures: decoders, muxes	§ 3.3
23	10/24	Memory	§ 3.5
24	10/26	FSM design example	
25	10/29	FSM design example	

26	10/31	FSM design example	
27	11/2	The von Neumann Model: basic components	§ 4.1, 4.2
28	11/5	The von Neumann Model: instruction processing, sequencing	§ 4.3 — 4.5
29	11/7	The LC-3 Instruction Set Architecture: operates, data movement, control	§ 5.1 — 5.4
30	11/9	Example program in LC-3 machine language	§ 5.5, 5.6
	11/12	**OPTIONAL REVIEW SESSION FOR MIDTERM #3**	
31	11/14	Problem solving using systematic decomposition	§ 6.1
32	11/16	More programming examples, debugging	§ 6.2
33	11/26	Assembly language: introduction	§ 7.1, 7.2
34	11/28	Assembly language: examples	
35	11/30	Assembly language: how assemblers work	§ 7.3
36	12/3	Design of the LC-3 datapath; optimization of elements such as ALU	
37	12/5	Control unit design as FSM (microcode)	App. C
38	12/7	Implementation of the LC-3	App. C
39	12/10	I/O in hardware and software (assembly)	Ch 8
40	12/12	Wrap-up and advice	

Supplemental notes will also be provided, particularly for the second and third parts of the class.