UNIVERSITY OF NEBRASKA AT OMAHA
COURSE SYLLABUS/DESCRIPTION

Department and Course Number	CSCI 3710
Course Title	Introduction to Digital Design and Computer Organization
Course Coordinator	Hassan Farhat
Total Credits	3
Date of Last Revision	February 16, 2014

1.0 Course Description

1.1 Overview of content and purpose of the course (catalog description)
This course starts with the basics of the digital design and progresses to arithmetic units design, register files and datapath design, and computer organization. Topics covered from digital design include: Boolean algebra and minimization; primitive gates and the RC model for gate delay; decoders, encoders, multiplexers and demultiplexers; memory primitives; sequential circuits design and analysis procedures; counters and registers. Topics covered from computer organization include number representations, arithmetic, logic and shift unit design for unsigned and signed numbers; register-files single-output and multiple-output ports organization and design; memory organization and design DRAM, SRAM, ROM and flash; the three computer organizations stack, AC and general-purpose; and assembly programming.

1.2 For whom course is intended
The course is intended for undergraduate computer science majors.

1.3 Prerequisites of the course (courses)
CSCI 3320 (could be taken concurrently)

1.4 Prerequisites of the course (topics)
1.4.1 Knowledge of programming and data structure

1.5 Unusual circumstances of the course
None

2.0 Objectives

2.1 Study of gate deigns at switch level
2.2 Study of Boolean algebra, minimization, and combinational circuit design
2.3 Study of primitive memory elements, analysis and design of sequential circuits
2.4 Study of number representations
2.5 Study of computer organization at the register level
2.6 Study of the design of arithmetic, logic and shift units: signed and unsigned
2.7 Study the register-files and bus design
2.8 Study of memory design
2.9 Study assembly instruction formats for different architectures
2.10 Study of converting high-level constructs to assembly constructs

3.0 Content and Organization

		Contact hours
3.1	Gate designs at the switch level	1
3.2	RC circuits and gate delays	2
	3.2.1 RC time constant and effect on transition time	
	3.2.2 Rise and fall time	
	3.2.3 Gate propagation delays	
3.3	Boolean algebra, minimization, and design	4
3.4	Decoders, multiplexers, encoders and demultiplexers design	2
3.5	Computer data representations	3
	3.5.1 Unsigned and signed data	
	3.5.2 Range of data	
	3.5.3 Floating-point	
3.6	Computer arithmetic logic units efficient designs	5
	3.6.1 Design of arithmetic units using multiplexers (programmable logic design)	
	3.6.2 Design of arithmetic units using minimization techniques (ASIC design)	
	3.6.3 Design of logic units	
	3.6.4 Complexity of adder designs	
	3.6.5 Shifter design	
	3.6.5.1 The three forms of shifts: arithmetic, logic, rotate	
	3.6.6 Case study datapath ALU and shift unit design	
	3.6.6.1 Unsigned	
	3.6.6.2 Signed	
	3.6.6.3 Status bits	
3.7	Sequential circuits	3
	3.7.1 Primitive memory elements	
	3.7.2 Counters and Registers	
3.8	Sequential circuit design procedures and analysis	4
3.9	Register-Files	3
	3.9.1 Organization: single and multi-port input/output	
	3.9.2 Design	
3.10	Datapath organization	3
	3.10.1 AC based	
	3.10.2 General purpose register based	
	3.10.3 Cases study	
	3.10.3.1 Datapath design	
	3.10.3.2 Control word format	
3.11	Memory organization and design at gate and switch level	4
	3.11.1 SRAM	
	3.11.2 DRAM	
	3.11.3 ROM	
	3.11.4 Flash memory	
	3.11.5 Case study:	
	3.11.5.1 SRAM design	
3.12	The three architectures	2
	3.12.1 Stack based	

3.12.2 AC based
3.12.3 General purpose register based
3.12.4 Instruction formats and fields
3.12.5 Addressing modes
3.12.6 The datapath and control unit registers

3.13 Register Transfer Language and realization 1
3.14 Sample computer organization 7

3.14.1 Instruction codes
3.14.2 Computer registers
3.14.3 Machine instruction set
3.14.3.1 Instruction set completeness
3.14.3.2 Register-reference instructions
3.14.3.3 Memory-reference instructions
3.14.3.4 Input-output instructions
3.14.4 Instruction cycle
3.14.4.1 Fetch, decode and execute
3.14.5 Designer view and user view of accessible registers
3.14.6 Assembly language and translation to binary
3.14.6.1 Simple programs
3.14.6.1.1 add, subtract and store
3.14.6.1.2 logic and shift
3.14.6.2 Converting high-level constructs to assembly
3.14.6.2.1 If-then-else programs
3.14.6.2.2 Loop programs
3.14.6.2.3 Multiplication programs

4.0 Teaching Methodology

4.1 Methods to be used

The primary teaching methods will be lecture, in-class demonstrations, and lab assignments.

4.2 Student role in the course

The student will attend lectures and demonstration, participate in discussion on assigned readings, complete assigned homework, and complete required examinations

4.3 Contact hours

Three hours per week

5.0 Evaluation

5.1 Type of student projects that will be the basis for evaluating student performance, specifying distinction between undergraduate and graduate, if applicable. For Laboratory projects, specify the number of weeks spent on each project).

Students will complete a sequence of case studies that are tailored towards realization of the different units of computer architecture. This is in addition to quizzes and examinations.

5.2 Basis for determining the final grade (Course requirements and grading standards) specifying distinction between undergraduate and graduate, if applicable.

Component	Grading
Exams	80%
Homework/ Lab Assignments	15%
Participation	5%

5.3 Grading scale and criteria.

Points	*Grade*
97-100%	A+
93-96%	A
90-92%	A–
87-89%	B+
83-86%	B
80-82%	B–
77-79%	C+
73-76%	C
70-72%	C–
67-69%	D+
63-66%	D
60-62%	D–
0-59%	F

6.0 Resource Material

6.1 Textbooks and/or other required readings used in course

See 6.4 for possible texts

6.2 Other suggested reading materials, if any

Handouts on RC circuits

6.3 Other sources of information

None

6.4 Current bibliography of resource for student's information

6.4.1 M. Mano, *Computer System Architecture*, 3rd Edition, Prentice Hall, 1993.

6.4.2 H. Farhat, *Digital Computer Design and Organization*, Pearson Custom Publishing, 2000

6.4.3 J. Hayes, *Introduction to Digital Logic Design*, Addison-Wesley, 1993.

6.4.4 R. Tocci and N. Widmer G. Moss, *Digital Systems Principles and Applications*, Prentice Hall, 3rd edition, 2008

6.4.5 A. Clements, *The Principles of Computer Hardware*, 3rd edition, Oxford, 2000.

6.4.6 G. Karam and J. Bryant, *Principles of* Computer *Systems*, Prentice Hall, 1992

6.4.7 M. Morris Mano and Michael D. Ciletti, *Digital Design*, 4th edition,Prentice Hall, 2007

6.4.8 M. Morris Mano and C. Kime, *Logic and Computer Design Fundamentals*, 4th edition, Prentice Hall, 2008.

6.4.9 Daniel D. Gajski, *Principles of Digital Design*, Prentice Hall, 1997
6.4.10 Victor P. Nelson, H. Troy Nagle, Bill D. Carroll, David Irwin, *Digital Logic Circuit Analysis and Design*, Prentice Hall, 1995
6.4.11 Ken Coffman, *Real World FPGA Design with Verilog*, Prentice Hall, 2000
6.4.12 John Vyemura, *A first course in Digital System Design*, Brook/Cole Publishing, 2000
6.4.13 John Wakerly, *Digital Design Principles and Practices*, Prentice Hall, 4th edition, 2008
6.4.14 Steve Waterman, *Digital Logic Simulation with CPLD Programming*, Prentice Hall, 2003
6.4.15 William Stallings, *Computer Organization and Architecture: Designing for Performance*, 6th edition, Prentice Hall, 2003.
6.4.16 Randal E. Bryant, David R. O'Hallaron, *Computer Systems: A Programmer's Perspective*, Prentice Hall, 2003
6.4.17 G. Karam and J. Bryant, *Principles of* Computer *Systems*, Prentice Hall, 1992.
6.4.18 J. Carpinelli, *Computer Systems* Organization *and Architecture*, Addison-Wesley, 2001.
6.4.19 V. Hamacher, Z. Vranesic and S. Zaky, *Computer Organization*, 4th edition, McGraw-Hill, 1996.
6.4.20 V. Heuring and H. Jordan, *Computer System Design and Architecture*, Addison-Wesley, 1997.
6.4.21 J. Hayes, *Computer Architecture* and *Organization* 3rd edition, McGraw-Hill, 1998.
6.4.22 M. Murdocca and V. Heuring, *Principles of Computer Architecture*, Prentice Hall, 2000.
6.4.23 J. Hennessy and D. Patterson, *Computer Architecture: A Quantitative Approach*, 2nd edition, Morgan Kaufmann, 1996.
6.4.24 J. Hennessy and D. Patterson, *Computer Organization and Design: The Hardware/Software Interface*, 3rd edition, Morgan Kaufmann, 2007.
6.4.25 W. Stallings, Computer *Organization and Architecture: Designing for Performance, 7th Edition*, Prentice Hall, 2007
6.4.26 Andrew S. Tanenbaum, *Structured Computer Organization*, 5th edition, Prentice Hall, 2006.
6.4.27 James Evans, *Itanium Architecture for Programmers: Understanding 64-Bit Processors and EPIC Principles*, Prentice Hall 2003.
6.4.28 N. Weste and D. Harris, "CMOS VLSI Design: A Circuits and Systems Perspective 4th edition", Addison Wesley, 2011.
6.4.29 S. Brown and Z. Vranesic, *"Fundamentals of Digital Logic with VHDL Design with CD-ROM*, 3rd edition", McGraw-Hill, 2009.
6.4.30 T. Floyd, "Digital Fundamentals: A System Approach", Prentice Hall, 2013.
6.4.31 W. Kleitz, "Digital Electronics A Practical Approach with VHDL, 9th edition", Prentice Hall, 2012.
6.4.32 M. Mano and M. Cilette, "Digital Design, 5th edition", Prentice Hall, 2013.

6.4.33 R. Tocci, N. Widmer and G. Moss, “Digital Systems Principles and Applications, 11th edition”, Prentice Hall, 2011.

6.4.34 J. Hennessy and D. Patterson, “Computer Architecture: A Quantitative Approach 5th Edition”, Morgan Kaufmann, 2011.

6.4.35 D. Harris and S. Harris, “Digital Design and computer Architecture”, 2nd edition, Morgan Kaufman, 2013.

6.4.36 D. Patterson and J. Hennessy, “Computer Organization and Design, Revised 4th edition”, Morgan Kaufmann, 2011.

7.0 Computing Science Accreditation Board Category Content (class time in hours)

CSAB Category	*Core*	*Advanced*
Data Structures		
Computer Organization and Architecture	35	
Algorithm and Software Design	3	
Concepts of Programming Languages	3	

8.0 Oral and Written Communications

Every student is required to submit at least __0___ written reports (not including exams, tests, quizzes, or commented programs) to typically _____ pages and to make ___0__ oral presentations of typically ___0__ minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

9.0 Social and Ethical Issues

No coverage

10.0 Theoretical content

The course is considers theoretical aspects of Boolean algebra, and finite state machine design and minimization.

11.0 Problem analysis

The course is an introduction to computer architecture. As a result, design problems from Register Transfer Languages are analyzed. The designed solutions are considered in the analysis aspect of the design.

12.0 Solution design

The solution design includes translating the word problem into a formal description in the context of state machines, ASM charts, and microoperations realization.

CHANGE HISTORY

Date	***Change***	***By whom***	***Comments***
06/19/2003	Initial ABET version	Farhat	
06/19/2003	Cleanup	Wileman	
10/07/2008	Modified objectives and contents	Farhat	
10/07/2008	Undated Resource material	Farhat	
10/07/2008	Filled mapping table (Outcomes vs. Objectives)	Farhat	
11/24/2008	Combined the syllabus for the two courses (CSCI 2710 and CSCI 3710)	Farhat	
2/16/2014	Updated Resource Material	Farhat	