

Rutgers University, Department of Electrical and Computer Engineering
ABET COURSE SYLLABUS
COURSE: 14:332:333

Course Catalog Description:	14:332:333 – Computer Architecture Lab (1) Assembly language programming on an emulated 32-bit microprocessor. Design, simulation, and evaluation of a simple datapath and control using VHDL or Verilog.
Pre-Requisite Courses:	14:332:231, 14:332:252
Co-Requisite Course:	14:332:331
Pre-Requisite by Topic:	<ol style="list-style-type: none">1. Basic logic and state machine design2. Programming fundamentals (subroutines and data structures)3. Number systems.
Textbook & Materials:	Laboratory manual supplied by the instructor.
References:	<i>VHDL Starter's Guide</i> , by Sudhankar Yalamanchili, Prentice-Hall, 1998, or, CD Verilog Tutorial included with the textbook.
Overall Educational Objective:	This lab class is intended to train the students on both assembly language programming and processor design and simulation. It will also help the students to enforce their understanding of the knowledge learned in the co-requisite course 14:332:331(Computer Architecture and Assembly Language).
Course Learning Outcomes:	A student who successfully fulfills the course requirements will have demonstrated: <ol style="list-style-type: none">1. an ability to define assembly language variables and subroutines, and to define components within a hardware and the interfaces between these components.2. an ability to write assembly language programs that consist of subroutines or recursive subroutines, and to write VHDL/Verilog programs that describe the definition and architecture of a complex hardware.3. an ability to debug the assembly language programs by checking the control flow of the program and evaluating the contents of the internal registers.4. an ability to debug the VHDL/Verilog programs following the divide-and-conquer method taught in class.5. an ability to analyze the performance of an assembly language program in terms of the number of instructions it generates, and to analyze the performance of a

complex hardware in terms of its delay.

6. an in depth understanding of the working of a 32-bit architecture, with a special focus on the processor design including datapath and control.
7. an ability to compute the minimum clock cycle length as well as the allowable clock rate of a particular processor design.
8. an ability to optimize the design of the datapath and control to increase the clock rate given a fixed set of gates.

How Course Outcomes are Assessed:

- Attendance 10%
- Project 1 20%
- Project 2 30%
- Project 3 40%

N = none S = Supportive H = highly related

Outcome	Level	Proficiency assessed by
(a) an ability to apply knowledge of Mathematics, science, and engineering	H	Lab projects
(b) an ability to design and conduct experiments and interpret data	H	Lab Projects
(c) an ability to design a system, component or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability	N	
(d) an ability to function as part of a multi-disciplinary team	S	Lab Work
(e) an ability to identify, formulate, and solve ECE problems	H	Lab projects
(f) an understanding of professional and ethical responsibility	N	
(g) an ability to communicate in written and oral form	S	Lab project reports
(h) the broad education necessary to understand the impact of electrical and computer engineering solutions in a global, economic, environmental, and societal context	N	
(i) a recognition of the need for, and an ability to engage in life-long learning	S	Lab projects
(j) a knowledge of contemporary issues	N	
(k) an ability to use the techniques, skills, and modern engineering tools necessary for electrical and computer engineering practice	H	Lab projects
Basic disciplines in Electrical Engineering	H	Lab projects
Depth in Electrical Engineering	S	Lab projects
Basic disciplines in Computer Engineering	H	Assembly programming
Depth in Computer Engineering	H	Lab projects
Laboratory equipment and software tools	H	Lab projects
Variety of instruction formats	N	

Topics Covered week by week:

- Week 1:** Introduction to MIPS architecture and instruction set.
- Week 2:** Introduction to SPIM and MIPS programming.
- Week 3 and 4:** Project 1, MIPS programming with recursive subroutine.
- Week 5:** Introduction to VHDL and Verilog programming.
- Weeks 6 to 9:** Project 2, Designing a 32-bit ALU using VHDL or Verilog
- Weeks 10 to 15:** Project 3, Designing a 32-bit datapath (single-cycle or multi-cycle) and its control

Computer Usage: Each of the projects requires a computer that installs SPIM and Synopsis VSS.

Laboratory Experiences: Students need to do three laboratory projects.

Design Experiences: laboratory reports

Independent Learning Experiences: NA

Contribution to the Professional Component:

(a) College-level mathematics and basic sciences: 0.25 credit hours

(b) Engineering Topics (Science and/or Design): 0.75 credit hours

(c) General Education: 0 credit hours

Total credits: 1

Prepared by: Y. Zhang and G. Burdea

Date: August 2007