16.317: Microprocessor Systems Design I

Fall 2015

Syllabus

Course Meetings

Section 201: MWF 8-8:50 AM, Ball 314

Course Website

Main site: http://mjgeiger.github.io/16317/f15/index.htm

Schedule: http://mjgeiger.github.io/16317/f15/schedule.htm

Course Discussion Group

All course announcements will be posted on the discussion group—you are responsible for checking the board regularly or enabling direct e-mail updates from Piazza.

Sign up link: https://piazza.com/uml/fall2015/16317

Instructor

Dr. Michael Geiger
E-mail: Michael_Geiger@uml.edu
Office: Perry Hall 118A
Phone: 978-934-3618 (x43618 on campus)
Office hours: Monday 9-10:30, Wednesday 9-10:30, Thursday 1:30-3

During the above hours, student questions are my top priority. I will also be in my office MW 10:30-11:45 and F 9-11:45, and I am available by appointment at other times.

Feel free to stop by my office, e-mail me questions, or schedule a one-on-one appointment. Office hours are subject to change.

Textbook

Barry B. Brey, *The Intel Microprocessors: Architecture Programming, and Interfacing*, 2008, Prentice Hall. ISBN: 0135026458 *(optional text for first half of semester)*

Course Overview

Description: This course provides an introduction to microprocessors. It uses assembly language to develop a foundation on the hardware, which executes a program. Memory and I/O interface design and programming. Study of microprocessor and its basic support components, including CPU architecture, memory interfaces and management, coprocessor interfaces, bus concepts, serial I/O devices, and interrupt control devices. Laboratories directly related to microprocessor functions and its interfaces. 3 credits.

Prerequisites: 16.216 (ECE Application Programming) and 16.265 (Logic Design)

Course Overview (cont.)

Course Objectives: By the end of this course, you should understand and/or be able to use all of the following:

1. **Microprocessor Software Architecture:** Data formats, types, and alignment. Memory addressing and organization. Stack operation.
2. **Microprocessor Instructions:** Instruction formats and types: data transfer, arithmetic, logical, shift/rotate, conditional execution, program control, subroutines.
3. **Assembly Language Programming:** Ability to write, modify, and debug programs written in assembly language. Translation of high-level code to assembly language. Programs that integrate assembly and high-level code.
4. **Microprocessor Interfacing:** Memory and I/O interfacing. Bus cycles.
5. **Interrupt Processing:** Hardware and software interrupts.
6. **Microcontroller-based Systems:** Microcontroller architecture and instruction set. Microcontroller programming using both assembly language and high-level code. Design and debug microcontroller-based circuits.

Grading: Grades will be computed on an A to F scale; no A+ grades will be assigned, in accordance with UMass Lowell policy. The weights assigned to the various items are:

Homework/lab assignments	55%
Exam 1	15%
Exam 2	15%
Final	15%

Incomplete grades will only be given in exceptional situations, and the student must be passing the class at the time the grade is requested.

Class participation: You are responsible for all material discussed or announced in class. You are expected to attend class regularly and participate in any in-class discussions, as such exercises are essential to your learning. Although lecture attendance is not explicitly required, regular attendance will improve your understanding of the course concepts.

Exams: Make-up exams will only be offered in exceptional circumstances. You must notify Dr. Geiger as early as possible in order to determine an appropriate make-up date.

Assignment policies: Your assignments will be a mix of typical homework problems, programming assignments, and labs that involve both programming and hardware interfacing. All assignments will be posted on the course web page.

Assignment policies include the following:

- All assignments must be completed individually unless explicitly specified. You may be allowed to work in groups for lab assignments.
- Late assignments are penalized at a rate of 10% per day.
- Some programming assignments may require an instructor to check off the completion of one or more milestones within the assignment.

Academic Honesty

All assignments and exams must be completed individually unless otherwise specified. You may discuss concepts or material covered in class, but may not share any details of your solutions to assigned problems, including algorithms and code. Plagiarism (copying solutions from an outside source) is also unacceptable and will be treated as an instance of cheating.

Students are allowed to discuss assignments in general terms and to help one another fix specific errors. In this case, students are required to note that they received assistance from a classmate by listing that person's name and the nature of their assistance as part of their lab report or homework solution.

Any assignment or portion of an assignment that violates this policy will receive a grade of zero for all parties concerned. Depending on the severity of the infraction, or in cases of repeat violations, additional penalties may be given at the instructor's discretion, up to and including a failing grade in the course.

Further information on the university Academic Integrity policy can be found at:

http://www.uml.edu/Catalog/Undergraduate/Policies/Academic-Policies/Academic-Integrity1.aspx

Course Schedule

This schedule contains a tentative schedule of topics we will cover throughout the term; the course website will contain the most up-to-date version. The web page will also describe which section(s) of the textbook are associated with each lecture.

Please note that the exam dates are fixed—the first exam will be held on **Wednesday, September 30 in class**, the second exam will be held on **Wednesday, November 4 in class**, and the third exam will be held **during finals, at a date/time to be determined**.

Week	Date (M)	Lecture Topics
1	8/31	*No Monday lecture—classes begin 9/1* 1. Course introduction; role of ISA 2. Data storage and addressing
2	9/7	*No Monday lecture—Labor Day* 3. x86 introduction 4. Assembly basics; data transfer instructions
3	9/14	5. Data transfer and arithmetic instructions 6. Arithmetic instructions 7. Logical and shift instructions
4	9/21	8. Rotate and bit test/scan instructions 9. Conditional execution 10. Jump/loop instructions
5	9/28	11. Exam 1 Preview **Wednesday, 9/30: EXAM 1** 12. Subroutines
6	10/5	13. Exam 1 Review 14. HLL and x86 assembly 15. HLL and x86 assembly (continued)
7	10/12	*No Monday lecture—Columbus Day* 16. HLL and x86 assembly examples *(Tue. 10/13)* 17. Interrupts 18. PIC introduction
8	10/19	19. PIC instruction set 20. PIC instruction set (continued) 21. PIC instruction set (continued)
9	10/26	22. PIC assembly programming 23. PIC assembly programming (continued) 24. PIC assembly programming (continued)
10	11/2	25. Exam 2 Preview **Wednesday, 11/4: EXAM 2** 26. PICkit basics

Course Schedule (cont.)

Week	**Date (M)**	**Lecture Topics**
11	11/9	27. Exam 2 Review *No Wednesday lecture—Veterans Day* 28. Working with delay *Friday, 11/13: Last day to withdraw*
12	11/16	29. PIC interrupts 30. Analog to digital conversion 31. Practice problems
13	11/23	32. Topics TBD *No Wednesday, Friday lecture—Thanksgiving Break*
14	11/30	33-35. Topics TBD
15	12/7	36. Topics TBD 37. Exam 3 Preview *Classes end Wednesday, 12/9*
	TBD	**EXAM 3 (DATE/TIME TBD)**