CDA3100 Summer 2013 Syllabus

Instructor:

- Britton Dennis
- Email: dennis@cs.fsu.edu
- Home page: http://ww2.cs.fsu.edu/~dennis
- Office: 105A Love Building
- Office Hours:
  - Thursday, 9:30 AM – 12:30 PM

Class time and location:

- Lectures:
  Monday, Wednesday, Friday, 9:30 AM – 10:45 AM, LOV 103

Course Website:
http://ww2.cs.fsu.edu/~dennis/cda3100_summer_2013

Course Rationale:
This is a core course intended for computer science majors with previous C/C++ Language background. The course introduces fundamental concepts in digital logic design and computer organization. Assembly language programming is used to reinforce understanding of the basic computer structure and machine cycle operation principles. Basic Verilog programming is used to reinforce the understanding of the hardware design principles. The basic interface between the assembly language with C will also be included.

Prerequisites:
Corequisites: COP3330 (Object Oriented Programming) and MAD2104 (Discrete Mathematics I).

Textbook and Class Materials:
Required textbook: "*Computer Organization and Design: The Hardware/Software Interface*," Fourth Edition, D. A. Patterson and J. L. Hennessy, Morgan Kaufmann Publishers, 2009.

In addition, appropriate articles from the literature will be distributed to supplement the textbook and the class notes.

Final Time:
The final is tentatively scheduled for Friday, August 2nd. This is subject to change with proper notifications and reminders.

Course Objectives:
Upon successful completion of this course of study, a student will:

- Understand number representations; be able to compute the represented quantity (value) of a number in a specified representation system and be able to convert number representations between different representation systems.
- Understand algorithms for binary arithmetic operations; be able to perform addition and subtraction using binary numbers.
- Understand floating point number representations and IEEE 754 floating-point standard; be able to encode and decode floating point representations in IEEE 754 standard.
- Understand the MIPS instruction set; be able to encode and decode MIPS instructions.
- Understand MIPS assembly language; be able to read and write functions using the MIPS assembly language.
- Understand MIPS calling conventions; be able to read and write programs consisting of multiple functions in MIPS assembly; be able to show the stack of recursive functions in MIPS assembly.
- Know how to design combinational logic, including decoders, multiplexors, and simple arithmetic-logic units; be able to design decoders, multiplexors, and arithmetic-logic units using basic logic gates.
- Understand sequential logic elements including flip-flops, and memory elements and know to apply them to construct large components; be able to construct the state diagram of sequential logic elements.
- Understand finite state machines; be able to draw state diagrams, implement next state functions, and design and implement finite state machines using basic logic elements.
- Know how to design a datapath and control unit for a processor using a single cycle implementation; be able to design and implement a simple processor using basic elements such as combinational logic, memory elements, and other components.
- Know how to use Verilog to implement simple circuits
- Understand the limitations of single cycle implementations.
- Have basic knowledge of using inline assembly to interface with C.
- Have a solid foundation to further studies in computer organization, operating systems, and compilers.

Assignments:

Assignments will be given along the lectures and they need to be turned in, where some of the assignments require programming using MIPS assembly and some require programming using Verilog. There will be several in-class exercises for feedback and evaluation purposes. There will be a midterm and an accumulative final exam. There will be one assignment designated by the Department of Computer Science for assessment of certain expected outcomes for its degree programs, as required by our accreditation agencies, the University, and the State of Florida. Departmental policy does not permit a final grade of "C-" or better to be assigned unless the student has at earned a grade of "C-" or better on this assignment, regardless of performance on other work in the course.

Grading Policy:

Let S be the number in determining the grade determined by the in-class exercise, homework, midterm, and final. In class exercises: 5%. Homework: 40%. Midterm: 20%. Final: 35%.

Score	Grade	Score	Grade	Score	Grade
$90 \le S$	A	$73 \le S < 76$	B-	$60 \le S < 64$	D+
$85 \le S < 90$	A-	$70 \le S < 73$	C+	$57 \le S < 60$	D
$82 \le S < 85$	B+	$67 \le S < 70$	C	$54 \le S < 57$	D-
$76 \le S < 82$	B	$64 \le S < 67$	C-	$S < 54$	F

Attendance Policy:
The university requires attendance in all classes, and it is also important to your learning. The attendance record may be provided to deans who request it. If your grade is just a little below the cutoff for a higher grade, your attendance will be one of the factors that we consider, in deciding whether to "bump" you up to the higher grade. Missing three or fewer lectures will be considered good attendance. In rare cases, such as medical needs or jury duty, absences may be excused with appropriate documentation. You should let me know in advance, when possible, and submit the documentation I seek. You should make up for any materials missed due to absences.

Missed Exam Policy:
A missed exam will be recorded as a grade of zero. We will follow the university rules regarding missed final exams (see http://registrar.fsu.edu/dir_class/spring/exam_schedule.htm), for all the exams, including the final exam.

Late Penalties:
Unless otherwise specified, assignments have two due times. Assignments turned in before the first due time are considered in time. Assignments turned in before the second due time are considered late and will be penalized by 10 %. Assignments will **NOT** be accepted after the second due time. Usually, the first due time is the beginning of a lecture and the second due time is the beginning of the next lecture.

Academic Honor Policy:
Students are subject to the Academic Honor Code published in the Student Handbook. The Academic Honor System of Florida State University is based on the premise that each student has the responsibility (1) to uphold the highest standards of academic integrity in the student's own work, (2) to refuse to tolerate violations of academic integrity in the university community, and (3) to foster a high sense of integrity and social responsibility on the part of the university community.

In particular, non-group assignments and projects are expected to be the work product of the individual student alone. Group assignments and projects are expected to be the work product of the individual team alone. This means, among other things, that students (teams) are not permitted to read other student's (team's) code (on paper OR on screen) or discuss design or implementation of programming projects with anyone other than with the course staff. There is a line between helping fellow students learn and performing work for someone else and we all know where that line falls. You are responsible for ensuring that your code/documentation/results are adequately protected and not accessible to other students (teams). If you are uncertain about any act of collaboration, please discuss it with course staff before the act. Examples of behaviors NOT allowed include:

- Discuss the solution for a homework question.
- Copy programs for programming assignments.
- Use and submit existing programs/reports on Internet as written assignments.
- Submit programs/reports/assignments done by a third party, including hired and contracted.
- Plagiarize sentences/paragraphs from others without giving the appropriate references. Plagiarism is a serious intellectual crime and the consequences can be very substantial.

Accommodation for Disabilities
Students with disabilities needing academic accommodations should: (1) register with and provide documentation to the Student Disability Resource Center (SDRC); and (2) bring a letter to the instructor

indicating the need for accommodation and what type. This should be done within the first week of class. *This syllabus and other class materials are available in alternative format upon request.*

For more information about services available to FSU students with disabilities, contact the Assistant Dean of Students:

Student Disability Resource Center
(850) 644-9566 (voice)
(850) 644-8504 (TDD)
sdrc@admin.fsu.edu
http://www.disabilitycenter.fsu.edu/

Syllabus Change Policy:
This syllabus is a guide for the course and is subject to change with advance notice.

Tentative Schedule:

Week	**Topic**	**Reading**
Week 1	Introductions and number representations.	Section 2.1 - 2.4 and 3.5.
Week 2	MIPS coding	Section 2.6, 2.7.
Week 3	MIPS function.	Section 2.8, 2.9, 2.10.
Week 4	MIPS function. Floating point with MIPS.	Section 2.8, 2.9, 2.10. Section B.1 – B.4. Section B.7, B.8, B.9, B.10
Week 5	MIPS interrupt. Review of MIPS assembly.	Section B.1 – B.4. Section B.7, B.8, B.9, B.10.
Week 6	Digital logic and midterm.	Appendix C.2, C.3
Week 7	MIPS ALU design.	Appendix C.5.
Week 8	Introduction to Verilog. Memory Elements and counter.	Appendix C.4, C.7, C.8, C.10
Week 9	Finite State Machines. Exercise.	Appendix C.7, C.8, C.10.
Week 10	Datapath and control	Section 4.1 - 4.4
Week 11	Datapath and control.	Section 4.1–4.4
Week 12	Inline assembly. Final review.	See slides.