Administrative Details

Instructor: This is a tentative syllabus. The instructor might change, but the book, course objectives, and course description will not.

Office hours: Tuesday, Thursday 3 PM to 4 PM and by appointment

Lecture: Tuesday, Thursday 1:30 P.M to 2:45 PM

Location: Picacho 150

Schedule No: 86827

Description: Microcomputer architecture, instruction set, assembly language programming and debugging, I/O considerations, memory interface, peripherals and busses, exception/interrupt handling.

Prerequisites: Enroll requirements: Pre-requisites: CST 100 and CST 150;

Required Text: There is no required textbook for this course. You could refer to *Computer Organization and Design* 4th or 5th edition, by David A. Patterson and John L. Hennessy, Morgan Kaufmann Publisher

What is this course about?

Welcome to CST 250! This class is the study of microprocessors that are the brains of many of the systems we interact with each day. It includes topics such as number systems, assembly language programming, organization and design of a microcomputer, and interfacing with sensors, A/D converters, etc.

This course introduces you to core, low-level, computing systems concepts including assembly programming and the organization of computing systems. We'll look at tradeoffs in various contemporary architectures (ARM, Intel), practical limitations of computing systems, and some of the rationales behind their designs. Additionally, we'll be interfacing a lot with other systems through sensors and controls, which is what real engineers do. Throughout the course we will work on multiple labs and micro projects some of which will come together as a final class project.

It has been my personal experience as a student and as a teacher for many years that we often think we have understood a concept when someone explains it to us but when we try to apply it to solve a problem or build a project, we realize that we really didn't get it. This course is based on the pedagogy of active learning, where you apply your knowledge to solve problems and reflect on your learning. In that process you gain procedural knowledge and also gain and strengthen conceptual knowledge that is required in the context of your projects and activities. Be prepared to work hard. What you gain from this course will be proportional to how hard you are willing to work. Remember, a 3 credit-hour course entails 9 hours of work/study per week in a 15-week semester, and 1 hours per week in a 7.5 week semester.

Course Objectives

By the end of this course, you should be able to:

- Define basic terminology related to computing (e.g. Program Counter, Stack, Algorithm)
- Describe how programs are stored and operated on by the CPU
- Generate original assembly code using the software development cycle (analyze problem, create algorithm, draw flowchart, write program, and debug program).
- Utilize the advanced features of the CPU such as the stack and interrupts to write modular programs that make function calls
- Identify problems in familiar and unfamiliar programs and correct them.
- Interface with different types of I/O devices with a range of interface protocols (serial, parallel)
- Trace the path of an instruction through the CPU's datapath and identify the control signals that guide this path for that instruction
- Compare different approaches/solutions to a specific assembly programming situation/problem and evaluate which one is better (and why)
- Work with other students to develop teamwork skills and an appreciation for alternate approaches to a problem
- Create and deliver technical presentations that are well designed, easy to follow, and which clearly convey the work that was done

GRADING

COURSE GRADING	
Based on Points (**absolute, fixed, no curve**)	
>= 95.0 <= 100.0	**A+**
>= 87.5 < 95.0	**A**
>=85.0 < 87.5	**A-**
>= 82.5 < 85.0	**B+**
>= 75.0 < 82.5	**B**
>=72.5 < 75.0	**B-**
>= 65.0 < 72.5	**C**
>= 57.5 < 65.0	**D**
< 57.5	**F**

POINTS DISTRIBUTION	
Quizzes:	10 (5 * 2)
Project 0:	5
Project 1:	15
Project 2:	15
Project 3:	20
Final Project:	20
Final Exam:	15
Total Points Available:	100

COURSE TOPICS AND TENTATIVE CALENDAR

Week	Topic	Project/Lab
1	What is Assembly Language? PLP tool: Overview and Familiarity Review of Binary and Hexadecimal Number Systems	Project 0: Bitbucket Tutorial
2	PLP Arithmetic Instructions and Registers Memory Mapped Input Using Switches	Project 1: Simple Calculator
3	Control Flow: Branches and Jumps	
4	Polling, Interrupts and Timers Memory Mapped Output Using the UART	Project 2: UART Number Input
5	Modular Programs and Function Calls	
6	MIPS/PLP ISA: Relationship Between ISA and Hardware: An Overview; Data transfer, Arithmetic	Project 3: Full Calculator
7	Interfacing	
8	**Review for Midterm; Midterm Exam**	
9	High Level Languages to Assembly	
10	Focus on Project 3	
11	**Team Presentations for Full Calculator**	
12	Case study : ARM architecture	Final Project: Sorting Competition
13	Case Study: x86 and Modern Intel Processors	
14	Focus on Final Project	
15	Full Course Review and Project Updates	
16	**Final Team Presentations**	

PEER EVALUATION

Throughout the semester a number of peer evaluations will be administered through CATME. These peer evaluations give each student a chance to rate team member performance and provide comments to the instructor and the student. Peer evaluations may be used to weight **team-based** course grades (the projects). Weighting is at the discretion of the instructor, and is to encourage you to improve your contribution to the team if your teammates deem that you are not contributing enough. Weighting will not fall outside the range of 50% to 150% of your raw **team-based** score. If you have major issues with your team, do not wait, see the instructor sooner rather than later.

CLASS PARTICIPATION AND ATTENDANCE

There is no attendance policy. If you need to skip (or want to skip), feel free. We sometimes have in-class quizzes and assignments that can't be made up however, and you'll miss out on whatever is discussed in class.

You are expected to participate in class such as taking part in team based problem solving, preparing your solution to the homework that is to be discussed in class, taking quizzes, and contributing to in-class discussions in a constructive manner.

Wish you the best in this class and in all your endeavors!