CS 2521 : Computer Organization and Architecture (4)

Catalog Description:
Internal representation of programs and data. Computer organization and introduction to computer architecture. Machine and assembly language programming. Data and procedural structures. Addressing methods. Systems software including linking and loading. Introduction to hardware performance analysis and measurements.

Textbook: David Patterson and John Hennessy, *Computer Organization and Design: The Hardware/Software Interface, 2nd Ed.*, Morgan Kaufmann, 1997.

References:

Course Goals:
This class is an introduction to computer organization and design. A student who successfully completes this course will understand how to carry out and interpret quantitative performance evaluations of computer systems, how programs and data are represented in a computer, the foundations of computer arithmetic, the design and use of single-cycle, multi-cycle and pipelined processors. In addition, students will be exposed to current issues in computer architecture such as superscalar and VLIW designs.

Prerequisites by Course & Topic
CS 1521: Computer Science II – recursion, linked lists
ECE 1315: Digital System Design – combinational and sequential logic circuits
MA 1296: Calculus I – mathematical maturity, critical thinking

Major Topics Covered in the Course
- Von Neumann model of computing
- Instruction Set Architecture (RISC and CISC)
- Computer Arithmetic
- Single Cycle Architecture
- Multi Cycle Architecture
- Pipelining
- SuperScalar/VLIW

Class/Laboratory Schedule: Lecture: 3 hours per week, Laboratory: 1

Laboratory Projects
- Simulate CPI/MIPS measurements (1)
- Simple assembly language program to sum and find max values (1)
- Recursive solution to Fibonacci numbers (1)
- Booth's Algorithm for Integer multiplication (1)
- Floating point multiplication (2)
- Exception handling (2)
- Comparison of MIPS and EPIC (3)

Course Contribution to Program Objectives and Outcomes:
1. Students further their understanding of mathematics through an in-depth understanding of how integer and floating point computations are actually carried out at the hardware level. (*a*)
2. Students develop a significant understanding of the fundamentals of computer organization and architecture. (*c*)
3. Students do extensive programming at the assembly language level. (*d*)

Estimate CSAB Category Content

	CORE	ADVANCED
Data Structures		
Algorithms		
Software Design		

	CORE	ADVANCED
Computer Organization and Architecture	3	
Concept of Programming Languages	1	

Theoretical Content

- Number System Representation (1 hour)

Problem Analysis

There are 7 laboratory exercises, 6 of which require analysis of a problem in order to formulate an implementation.

Solution Design

There are 7 laboratory exercises, 6 of which require that a solution be designed in order to formulate an implementation.

Coordinator/Prepared by: T. Pedersen